Investigating host-microbiota cooperation with gap-filling optimization problems

Clémence Frioux

THESE DE DOCTORAT DE

L'UNIVERSITE DE RENNES 1
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
Mathématiques et Sciences et Technologies
de l'Information et de la Communication
Spécialité : *Informatique*

Par
Clémence FRIOUX

Investigating host-microbiota cooperation with gap-filling optimization problems

Thèse présentée et soutenue à Rennes, le 19 novembre 2018
Unité de recherche : Institut de Recherche en Informatique et Sytèmes aléatoires (IRISA)

Rapporteur·rice·s avant soutenance :

Oliver Ebenhöh	Professeur, Heinrich-Heine-Universität Düsseldorf, Allemagne
Marie-France Sagot	Directrice de recherche Inria, LBBE Lyon

Composition du Jury :

Présidente :	Mireille Ducassé	Professeure, INSA Rennes
Examinateurs :	Samuel Chaffron	Chargé de recherche CNRS, LS2N Nantes
	Vincent Fromion	Directeur de recherche, INRA Jouy-en-Josas
	Philippe Potin	Directeur de recherche CNRS, Station Biologique de Roscoff
	Laurent Simon	Professeur, LABRI Bordeaux
Dir. de thèse :	Anne Siegel	Directrice de recherche CNRS, IRISA Rennes

The task is, not so much to see what no one has yet seen;
but to think what nobody has yet thought,
about that which everybody sees.
—
Arthur Schopenhauer
Parerga und Paralipomena *(in German), 1851*

Remerciements

I would like to sincerely thank Oliver Ebenhöh and Marie-France Sagot for reviewing my manuscript. I also thank Samuel Chaffron, Mireille Ducassé, Vincent Fromion, Philippe Potin and Laurent Simon for accepting to be part of my jury.

Je remercie Oliver Ebenhöh et Marie-France Sagot d'avoir accepté d'être rapporteur et rapportrice de mon manuscrit de thèse. Je remercie également Samuel Chaffron, Mireille Ducassé, Vincent Fromion, Philippe Potin et Laurent Simon d'avoir accepté de participer à mon jury de soutenance.

Je tiens à remercier chaleureusement Anne Siegel, mon encadrante de stage M2 en 2015 puis ma directrice de thèse ces trois dernières années. Merci de m'avoir proposé cette thèse et encouragée tout au long de cette aventure. Merci aussi pour tes conseils, ton coaching et ton soutien : ils m'ont énormément apporté.

Je remercie Inria, l'UMR IRISA, l'Ecole Doctorale MATHSTIC et l'Université de Rennes pour m'avoir permis de faire ce doctorat. Merci à l'Université de Rennes 1 et à l'ENSAI de m'avoir permis de découvrir l'enseignement ; et à mes étudiants, de m'avoir fait aimer enseigner.

Merci à Torsten Schaub, Philipp Wanko et Sebastian Schellhorn. Merci pour votre accueil pendant mes deux mois à Potsdam, pour toute ton aide Torsten et pour vos réponses à mes innombrables questions Philipp et Sebastian. Mes remerciements vont aussi à Simon Dittami, j'ai adoré travailler sur *Ectocarpus* et ton aide et ta disponibilité ont été précieuses.

Que serait une thèse sans une bonne équipe et des formidables collègues ? Merci à tous les membres de Dyliss, mon équipe d'accueil mais aussi à tous les symbiotes pour les pauses, la bonne humeur, l'aide en cas de besoin. Un merci tout particulier à la Plateforme Genouest : Genostack et GoDocker m'ont bien dépannée en période de rush et bien sûr à leurs administrateurs respectifs pour leur patience et leurs bons et loyaux services (Matéo, Olivier-s).

Quelques remerciements personnels ensuite, mais l'exhaustivité de la liste n'est pas garantie. Merci à Enora, je suis ravie d'avoir pu travailler avec toi pendant ces six mois, j'en ai beaucoup appris. À Jeanne pour les discussions du matin. À Arnaud pour ton aide sur les projets, pour le débugage Python 3 et les heatmaps ! Merci à Lucas pour plein de choses : ton aide, tes outils, mais aussi et surtout ces moments plein de bonne humeur et de bons mots. Merci à mes cobureaux, j'ai adoré partager mes plantes avec vous.

Je remercie aussi Camille. Nos discussions pendant tes jeudis à Rennes et même les autres jours ont beaucoup compté pendant ma thèse. Merci pour tes remarques toujours pertinentes, tu as contribué à améliorer mon esprit critique et la qualité de mon travail et merci pour ton amitié. Merci à toi Méziane pour tellement de choses : ton aide au travail bien sûr mais pas que ! Ton amitié, nos fous rires, tes attentions, l'entrainement à avoir des réflexes... Merci Cervin pour ces trois très bonnes années (!), c'était un plaisir de t'avoir en compagnon de thèse, en co-réalisateur également, j'ai bien ri pendant trois ans grâce à toi. Marie et Chloé, il y a tellement de mercis pour vous aussi : les discussions, la bonne humeur, les soirées et tout le reste. J'ai adoré le Chili avec toi Marie, où j'y ai fait ta connaissance au tout début de ces trois ans et puis tous les bons moments qui ont suivi. Chloé ta présence a été très importante pour moi : les lundi midi, les patates, le taiso et bien sûr les vacances !

Marie L, tu as ta place dans mes remerciements également. Malgré la distance tu es toujours là en cas de besoin. Tu es une amie en or, mais tu le sais déjà.

Merci à tous mes amis. Vous vous reconnaîtrez et si ces trois ans se sont bien passés, à Rennes, en soirées, en vacances ou ailleurs, c'est aussi grâce à vous. Petite pensée aussi pour les paresseux du Costa Rica que j'ai croisés pile au bon moment de ma thèse.

Un immense merci à mes parents. Vous m'avez toujours encouragée, et sans vous je n'aurais pas pu faire mes études dans de si bonnes conditions. Sans votre aide je n'aurais jamais été jusqu'au doctorat. Amélie je te remercie bien sûr également, tu as une confiance en moi qui relève de l'extraordinaire et ton soutien m'est plus précieux que ce que tu peux imaginer.

Et enfin, *last but not least*, merci à toi Ludo de m'avoir apporté ton soutien à toute épreuve dans cette aventure. Tu as rempli ta part du contrat et bien plus encore. Je voulais ajouter que sans toi cette thèse n'aurait pas été la même ou n'aurait pas été tout court, et ces derniers mots de remerciements sont pour toi.

Contents

Résumé en Français

Résumé en Français

Des métabolismes non modèles en écologie des systèmes

La meilleure compréhension de la physiologie des organismes est un des objectifs de la biologie des systèmes. Elle passe notamment par l'intégration des connaissances et des données dans des modèles de systèmes biologiques [Kitano, 2002a]. La biologie, la chimie ou encore les sciences environnementales peuvent désormais s'associer aux sciences de l'informatique pour créer des nouvelles disciplines à leur interface. C'est le cas de la bioinformatique qui est un champs essentiel de la biologie des systèmes pour intégrer données et connaissances, et modéliser les mécanismes biologiques afin d'expliquer les observations et prédire les réponses.

Le métabolisme est un domaine d'intérêt en biologie des systèmes : il étudie les transformations biochimiques des composés sous l'activité de protéines appelées enzymes. Ces dernières sont exprimées à partir du matériel génétique de la cellule. Elles sont par ailleurs régulées par divers signaux et composés cellulaires, mais aussi par leur environnement. Le métabolisme est ainsi impacté par de nombreux mécanismes physiologiques au sein de la cellule ou de l'organisme. En bioinformatique, les réseaux métaboliques regroupent la représentation des capacités métaboliques d'une cellule, d'un organe ou encore d'un organisme (réseaux à l'échelle génomique). Ils lient également cette activité métabolique à l'information génétique. Ainsi, le métabolisme peut être abordé en intégrant les observations issues de nombreuses expériences dites "omiques" : génomique, transcriptomique, métabolomique, protéomique etc. De multiples méthodes computationnelles pour modéliser et explorer les réseaux métaboliques ont été développées pour transformer les données liées au métabolisme des espèces en prédictions de leur physiologie.

Cette thèse soulève des questions liées à l'étude du métabolisme chez les organismes non modèles et tente d'y apporter des solutions. Ces espèces ont la particularité d'avoir été peu étudiées du point de vue expérimental, mais de l'information biologique, notamment sous forme de séquences génétiques, apparaît pour ces organismes grâce à la génération de données à haut débit en biologie. Pour les organismes dits "modèles", des observations et expériences ont été faites pendant des décennies voire des siècles. Au contraire, la connaissance de la physiologie de ces organismes non modèles, qui sont généralement distants phylogénétiquement des premiers, est limitée [Russell et al., 2017].

L'état de l'art de la littérature liée à l'analyse bioinformatique du métabolisme et plus généralement à la biologie des systèmes démontre un basculement depuis des organismes modèles, étudiés individuellement, vers des organismes non modèles, étudiés collectivement au sein de microbiomes. Cette évolution est liée à celle des techniques omiques qui génèrent un afflux de données inédit en biologie des systèmes. Les génomes de ces organismes précédémment peu ou pas étudiés deviennent disponibles. Plusieurs limites à leur étude émergent alors. Les données et modèles qui en découlent sont plus sujets à l'incomplétude ou aux erreurs, en raison du manque de résultats expérimentaux pour orienter la modélisation. Par ailleurs, le rôle des méthodes automatiques et informatiques dans les études prend une importance d'autant plus grande que la littérature contient peu d'informations sur ces organismes. Cela implique d'améliorer les modèles et leur capacité de résilience à l'incomplétude des données. D'autre part, il est nécessaire de donner aux expert·e·s des domaines et aux biologistes une place importante dans la validation des prédictions afin qu'ils·elles puissent choisir et orienter les hypothèses à valider par l'expérimentation.

Du métabolisme individuel vers les interactions des microbiotes

La question de l'étude des organismes non modèles est également au centre des problématiques soulevées par la place importante que prend la recherche sur les microbiomes. Ce terme regroupe à la fois les micro-organismes (dont l'ensemble forme le microbiote), leurs génomes et leur environnement [Marchesi and Ravel, 2015]. La relation entre deux organismes, qualifiés de symbiotes, est appelée symbiose. En effet, les organismes sont considérés comme des éléments d'une communauté qui interagit positivement ou négativement sur leur physiologie [Cavaliere et al., 2017]. Ces interactions interviennent notamment à l'échelle métabolique à travers l'échange de métabolites. Les microbiotes marins, la rhizosphère des plantes, le microbiote intestinal humain et beaucoup d'autres sont particulièrement étudiés pour mettre en évidence les dépendances entre hôtes et micro-organismes et à terme, exploiter la connaissance tirée de ces études pour des applications en santé, environnementales ou industrielles. L'amélioration continue des techniques de séquençage donne accès à des données génétiques concernant des espèces qui ne sont pas forcément cultivées, voire même non cultivables, notamment en isolation de leurs symbiotes. Néanmoins, il est souhaitable de parvenir à comprendre au moins partiellement la physiologie de ces organismes pour contribuer à élucider un objectif général qu'est l'organisation des microbiomes. Les méthodes bioinformatiques doivent ainsi s'adapter à l'analyse de ces données massives malgré la connaissance imparfaite des organismes considérés et le potentiel limité d'expérimentation pour valider les hypothèses.

Cette thèse s'intéresse particulièrement aux applications de biologie marine et à l'algue brune *Ectocarpus siliculosus* pour laquelle les dépendances envers son microbiote sont confirmées [Tapia et al., 2016, Dittami et al., 2014a] mais dont la nature reste encore à élucider. De nombreuses méthodes sont développées pour appréhender les interactions, mais leur applicabilité aux organismes non modèles reste limitée [Gottstein et al., 2016]. Ces méthodes diffèrent entre elles sur plusieurs critères, dont le principal est la sémantique utilisée pour modéliser la fonctionnalité du métabolisme, ou productibilité des métabolites. Ces sémantiques peuvent être généralisées avec une notion d'activation de réaction à partir de métabolites disponibles appelés graines et représentant généralement le milieu de culture ou l'environnement. L'activation se dérive, avec les sémantiques de productibilité existantes, en deux activations qui peuvent être complémentaires : l'activation topologique (approche graphe) [Ebenhöh et al., 2004] et l'activation flux (approche contraintes stoechiométriques) [Orth et al., 2010]. Ces définitions sont un fil rouge de la thèse, la première s'adapte facilement aux données incomplètes et la seconde est quantitative et modélise la fonctionnalité avec davantage de précision.

L'objet de la présente thèse est de démontrer l'applicabilité des méthodes combinatoires et de programmation logique à deux problèmes. Le premier concerne l'étape de complétion lors de la reconstruction des réseaux métaboliques. Elle vise à rafiner les modèles afin qu'ils permettent la réalisation d'objectifs métaboliques selon des sémantiques définies d'activation des réactions. L'ajout de cette information manquante est réalisée en sélectionnant des réactions métaboliques au sein de bases de connaissances, avec des critères propres à chaque méthode de complétion. La première partie de la thèse s'intéresse à ce problème pour des contraintes liées aux organismes non modèles. Dans sa seconde partie, cette thèse s'applique à résoudre le problème de la sélection de communautés minimales dans un microbiote avec les contraintes suivantes : minimiser d'une part le nombre de symbiotes choisis et d'autre part le nombre d'interactions ou échanges nécessaires pour atteindre l'objectif métabolique choisi. La dernière exigence imposée au problème est de ne pas proposer une unique solution mais au contraire d'explorer toutes les communautés pour prévenir une perte d'information potentiellement pertinente et ainsi proposer aux biologistes l'intégralité des modèles optimaux.

Première partie : avancées pour la complétion de réseaux métaboliques

La première partie des résultats développe des travaux liés à la complétion de réseaux métaboliques. Le **Chapitre 2** s'attache à la **validation de Meneco** (travaux publiés dans [Prigent et al., 2017]), une méthode de **complétion topologique utilisant la programmation par ensembles réponses** (ASP). Cette évaluation se fait avec une analyse comparative de ses résultats et de ceux de méthodes basées sur une activation flux [Satish Kumar et al., 2007, Vitkin and Shlomi, 2012, Thiele et al., 2014]. Des réseaux métaboliques dégradés de la bactérie *Escherichia coli* et plusieurs bases de données de réactions sont utilisés pour les expérimentations. Le critère de parcimonie adopté par Meneco se révèle être un avantage en pratique dans la mesure ou l'étape de complétion se veut être une étape de suggestions à valider par des expert·e·s. Cela vise à limiter l'ajout de réactions faux-positifs non associés à des gènes dans le réseau. Ainsi, la capacité de Meneco d'échantillonner tout l'espace des solutions et de proposer des sets minimaux de réactions est un avantage. Parallèlement nous montrons que, en dépit de sa sémantique basée sur les graphes, Meneco est en mesure de sélectionner des complétions qui satisfont les contraintes des sémantiques flux (Flux Balance Analysis [FBA]) dans une majorité des cas pour lesquels la dégradation du réseau métabolique est modérée.

Pour des plus grands taux de dégradation, les réseaux complétés par Meneco sont moins fonctionnels selon le critère de la FBA, en comparaison avec fastGapFill. Dans le **Chapitre 3** (travaux publiés dans [Frioux et al., 2017]), il est proposé d'étendre la définition de la complétion topologique en l'associant à des contraintes de programmation linéaire satisfaisant la FBA pour proposer une **complétion de réseaux métaboliques hybride**. Techniquement, l'association ASP et contraintes linéaires est résolue par l'utilisation d'un propagateur de contraintes de programmation linéaire (LP) qui permet la vérification de la satisfaisabilité des modèles proposés par ASP avec un solveur LP. Des interactions entre solveurs SAT et modules de théories avaient déjà été développées dans le domaine de l'étude du métabolisme [Peres et al., 2014], mais jamais pour l'ASP. Appliquée à la complétion de réseaux métaboliques, la combinaison de l'ASP et de la LP (Cplex) permet de proposer des ensembles minimaux de réactions satisfaisant les deux sémantiques d'activation. Cette complétion hybride est implémentée dans Fluto. Sa capacité à restaurer la fonctionnalité des réseaux métaboliques a été validée sur les mêmes données d'*Escherichia coli* que pour Meneco.

Les deux méthodes de complétion ci-dessus ont ensuite étudiées dans le cadre d'**application à des données concrètes** dans le **Chapitre 4**. Premièrement, l'impact de la complétion par Meneco sur la fonctionnalité (flux et topologique) a été démontré lors de la reconstruction d'EctoGEM, le réseau métabolique de l'algue brune *Ectocarpus siliculosus* dans le cadre de la ré-annotation de son génome (travaux publiés dans [Aite et al., 2018]). Il a été observé à cette occasion que l'utilisation de méthodes complémentaires lors de la reconstruction de réseaux, notamment eucaryotes, permet d'optimiser l'utilisation des données disponibles. Ainsi, il est possible de restaurer la fonctionnalité du réseau vis-à-vis de la production de biomasse, mais également de compléter des voies métaboliques. Une fois reconstruit, un réseau métabolique de qualité peut contribuer à la reconstruction d'autres réseaux. C'est ce qui est montré pour l'algue rouge *Chondrus crispus*. Son réseau métabolique n'était pas fonctionnel pour la production de l'acide aminé alanine. Fluto a été utilisé pour trouver dans le réseau d'*E. siliculosus* la réaction manquante, ce qui a restauré la productibilité en flux du composé. En pratique, la complétion hybride peut donc s'appliquer efficacement au déblocage de points spécifiques du réseau métabolique. Enfin, la théorie de la complétion de réseaux métaboliques a été dérivées pour l'étude des complémentarités métaboliques entre EctoGEM et le réseau d'une bactérie symbiotique de l'algue brune, mais non cultivable, *Candidatus* Phaeomarinobacter ectocarpi (travaux publiés dans [Prigent et al., 2017]). Meneco a été utilisé pour déterminer les métabolites dont la production chez l'algue est à même d'être facilitée par la bactérie. Ces interactions hypothétiques ont ensuite été soigneusement rafinées par un biologiste pour éliminer les faux-positifs et ainsi retourner un ensemble de 19 métabolites qui sont susceptibles d'être au cœur d'interactions algo-bactériennes. Cela démontre que la recherche d'interactions entre organismes est un problème qui peut être abordé par des méthodes combinatoires comme la complétion de réseaux. Par ailleurs, la reconstruction de réseaux métaboliques en considérant l'organisme isolément peut être source d'erreurs dans la mesure où certaines réactions non associées à des gènes peuvent en réalité n'être uniquement catalysées que par des symbiotes. C'est donc un paramètre à prendre en compte lors de la reconstruction, mais également lors de l'analyse d'interactions elle-même, en ôtant par exemple ces réactions du réseau considéré.

Seconde partie : sélection de communautés dans des microbiotes

Formalisation et Modélisation du problème de sélection de communautés

Les travaux sur l'analyse des communautés se sont poursuivis dans la **seconde partie** de la thèse, publiés dans [Frioux et al., 2018a]. La base théorique de la complétion et la sémantique basée sur les graphes ont été appliquées à la formalisation de méthodes de sélection de communautés résolue par des approches ASP. Le problème est celui du choix de communautés de symbiotes dans un (possiblement grand) microbiote afin de satisfaire un objectif métabolique commun ou spécifique de l'hôte. Les communautés sélectionnées sont minimales en taille et en en matière d'interactions requises pour atteindre l'objectif, c'est-à-dire en nombre d'échange de métabolites. Pour cela, une approche en deux étapes est proposée, pour mieux adresser la complexité du problème. Dans un premier temps, uniquement la taille de la communauté est minimisée, ce qui permet d'utiliser un formalisme modélisant un méta-organisme dont la résolution est facile en ASP. Ce type de modélisation a été utilisé par [Eng and Borenstein, 2016] dans un outil qui propose une unique solution de communauté optimale. Ici, les communautés satisfaisant ce critère peuvent être énumérées ou l'union des symbiotes qui y apparaissent peut être directement obtenue. Dans un deuxième temps, un formalisme compartimentalisé, dans lequel les échanges sont pris en compte est utilisé et ces échanges peuvent être minimisés. Les travaux de [Julien-Laferrière et al., 2016] utilisent des optimisations proches dans un problème de sélection de consortiums parmi des petits ensembles de bactéries. Cette seconde minimisation est davantage complexe mais la réduction de l'espace de recherche réalisée à la première étape la rend calculable en pratique. Ces méthodes ont été encapsulées en Python et implémentées dans Miscoto.

Miscoto a été testé sur les données du Human Microbiome Project [Human Microbiome Project, 2012] en sélectionnant des communautés permettant de transformer un métabolite graine en un métabolite cible. En ressort l'évidence d'une redondance fonctionnelle dans le microbiote, reflétée par le grand nombre de communautés minimales associées à chacune des fonctions testées. Néanmoins, la combinaison des deux étapes de minimisation réduit significativement le nombre de communautés minimales et le nombre de bactéries apparaissant dans leur union. Ces communautés ont vocation à être *a posteriori* filtrées sur des critères additionnels, aussi il est crucial d'accéder à l'ensemble d'entre elles et non pas en proposer arbitrairement une unique à l'utilisateur. En effet, dans le cadre de la sélection de communautés pour diverses applications (industrielle, santé, etc.), ce raffinement des solutions proposées peut permettre d'appliquer des critères tels que l'incompatibilité de croissance de certaines espèces, des cas de compétitions - phénomène non modélisé ici -, des difficultés à travailler avec certaines souches etc.

Applications de la sélection de communautés

Le travail de sélection de communautés a été appliqué dans le **Chapitre 6**. Dans un premier temps un système composé du **réseau métabolique humain et de 773 réseaux métaboliques du microbiote intestinal** a été considéré dans des conditions imitant la culture cellulaire d'entérocytes (cellules absorptives intestinales). Les communautés de taille minimale permettant au réseau humain de produire un maximum de composés cytosoliques ont été énumérées. 89 bactéries apparaissent dans 381 communautés de taille 3. Elles ont été analysées avec des méthodes de classification, d'études de graphes et de power graphes. Cela a permis de discriminer les bactéries et de mettre en évidence des ensembles de bactéries équivalentes dans le contexte de l'étude. Puis les échanges au sein des communautés ont été calculés pour déterminer la capacité de chaque bactérie à débloquer la productibilité des cibles métaboliques chez l'humain. Cela a permis de mieux comprendre les dépendances des cibles aux groupes bactériens issus de la classification précédente.

Un travail similaire a été réalisé pour **sélectionner des communautés bactériennes susceptible d'aider *Ectocarpus siliculosus* à réaliser des fonctions métaboliques**. En effet, comme *Ca.* P. ectocarpi, la bactérie étudiée dans le **Chapitre 4** sur l'application des méthodes de complétion, n'est pas cultivable, il est judicieux d'exploiter la redondance fonctionnelle des microbiotes et de chercher des bactéries aux capacités similaires parmi des bactéries isolées auprès de l'algue brune et cultivables. Les communautés prédites à l'aide de Miscoto ont été testées expérimentalement par des collègues de Sorbonne Université à la Station Biologique de Roscoff (Bertille Burgunter-Delamare, Simon Dittami). Les résultats préliminaires montrent des effets positifs significatifs des bactéries sur la croissance de l'algue par rapport à celle de l'algue isolée. Des analyses métabolomiques retrouvent 7 des 8 composés testés uniquement dans les cultures non-axéniques de l'algue. Cette collaboration entre bioinformatique et biologie sur la prédiction de communautés pour les algues est prometteuse et forme une base intéressante pour de futures expérimentations.

Conclusions générales de la thèse

Cette thèse montre l'applicabilité des techniques combinatoires à l'étude du métabolisme des organismes non modèles. Elle établit que les méthodes reposant sur une activation topologique du métabolisme sont adaptées à la sélection et à l'échantillonnage de l'espace des solutions grâce à l'efficacité des méthodes de résolutions en programmation logique et notamment en ASP. Ces sémantiques passent plus facilement à l'échelle et supportent davantage les données incomplètes que les méthodes basées sur l'analyse de flux, ce qui les rend particulièrement adaptées à des analyses de premier ordre sur les organismes non modèles. Les résultats peuvent dans un deuxième temps être raffinés avec des sémantiques flux ou une expertise humaine. Par ailleurs, cette thèse montre que la combinaison des deux sémantiques en ASP est applicable en pratique à des problèmes tels que la complétion de réseaux métaboliques. Enfin, les méthodes combinatoires peuvent être utilisées pour des objectifs de prédiction à des fins de validations expérimentales, telle que présentées pour la sélection de communautés au sein de microbiotes.

Publications liées à la thèse

Les travaux présentés dans cette thèse ont pour une grande partie été publiés. La validation de Meneco (Chapitre 2) et l'étude de la complémentarité métabolique entre *Ectocarpus siliculosus* et sa bactérie symbiotique (partie du Chapitre 4) ont été publiée dans *PLOS Computational Biology* [Prigent et al., 2017], tout comme le travail sur la place de la complétion métabolique et la traçabilité des reconstructions [Aite et al., 2018] (Chapitre 4). Le développement de la complétion hybride (Chapitre 3) a été présenté à la 14eme conférence *Logic Programming and Non-Monotonic Reasoning LPNMR* [Frioux et al., 2017]. Ce papier a été élu "meilleur article d'étudiant" et choisi pour une publication étendue dans le journal *Theory and Practice of Logic Programming* (accepté [Frioux et al., 2018b]). Enfin, les travaux sur la sélection de communautés (Chapitres 5 et 6) ont été acceptés pour présentation à la 17eme conférence *European Conference on Computational Biology ECCB* dont les actes sont publiés dans *Bioinformatics* [Frioux et al., 2018a].

Contributions logicielles

Le **Chapitre 7** présente les outils développés pendant la thèse du point de vue logiciel. La question de la nécessité d'encapsuler les outils basés sur ASP pour faciliter leurs utilisations et installations est abordée. Durant ce doctorat, MeNeTools pour l'analyse de réseaux métaboliques, Fluto pour la complétion hybride et Miscoto pour la sélection de communautés ont été développés. Parallèlement, un travail commun sur la reconstruction de réseaux métaboliques a été effectué avec le développement d'AuReMe. Cette plateforme conteneurisée sur Docker permet l'intégration de logiciels hétérogènes et le suivi des modifications qu'ils apportent au réseau métabolique. Ainsi l'utilisateur n'a pas à gérer les installations individuelles des outils et peut produire des réseaux dont la reproductibilité et la traçabilité des modifications sont assurées. Par ailleurs, une vue utilisateur locale sous forme de wiki est proposée en lien avec cette plateforme, afin de visualiser et explorer le réseau et ses métadonnées à chaque étape de sa reconstruction.

Introduction

Better understanding the physiology of organisms comes under the objectives of systems biology through the integration of knowledge and data, into models of biological systems [Kitano, 2002a]. Biology, chemistry and environmental sciences have met computer science and have led to the rise of new interfacing fields. Computational biology (or bioinformatics) is one of them. It can be at the service of systems biology to model biological mechanisms in order to integrate observations, explain behaviours and predict responses.

Metabolism is a field of choice in systems biology: it consists in studying the transformation of compounds under the activity of proteins called enzymes. The latter are expressed from genetic material; they are regulated by various signals and components of the cell and its environment, and thus the metabolism is impacted in many ways by the physiology of the cell. In computational biology, metabolic networks gather the metabolic capabilities of cell, organs or full organisms and link activity to genetic information. Accordingly, metabolism can be tackled by integrating information that comes from many "omics" experiments. Therefore, computational methods for modeling and exploration have been developed to turn data into knowledge and accurate predictions [O'Brien et al., 2015].

This thesis addresses questions raised by studies performed on non-model organisms [Russell et al., 2017]. They are species that were previously poorly studied but for which various material, including genetic sequences is now available thanks to the high-throughput production of data in biology. Contrary to model organisms on which experiments have been carried for decades or centuries, the level of knowledge is low for this genetically-distant organisms. This encompasses two main limitations. The data and their associated models are more likely to be error-prone or incomplete. Secondly, the place taken by automatic methods and prediction computations is higher due to the lack of existing knowledge in literature. This entails to improve the models and their ability to be resilient to sparse data. Furthermore, it requires to put experimenters and experts at the center of the predictions, so that they can rely on the whole puzzle to choose hypotheses and make decisions for experimentation.

Non model organisms are more and more frequently addressed, for a second reason that is the rise of microbiome-centered studies. Marine microbiota, plant rhizosphere, human gut microbiota and many others are extensively analyzed to capture the dependencies between their associated hosts and the microorganisms. Sequencing advances give access to data about species that are not necessarily cultivated nor cultivable and yet the physiology of these non-model organisms is meant to be understood to unravel the bigger picture that is the organization of the microbiomes [Cavaliere et al., 2017]. Computational methods have to be adapted to analyze this massive data despite the incomplete knowledge on the organisms and the limited experimental potential for validation.

Review of literature (Chapter) related to metabolism in computational biology and systems biology in general, shows that the **shift between individually-studied model organisms and collectively-studied Non-Model Organisms (NMOs)** has been happening. It is promoted by the dramatic evolution of omics techniques and its associated decreased cost that produces a data flood in systems biology. Thus, gene sequences become available for many organisms that were previously poorly or not studied and that for some of them cannot be cultured in the lab. In addition, organisms are understood to be part of a whole community that involves other species interacting together, for the best or for the worst. These interactions can occur at the metabolic level through exchanges of metabolites. In particular, this thesis focuses on marine biology and the brown algae *Ectocarpus siliculosus* for which the understanding of its dependencies to microbiota is an open challenge [Dittami et al., 2014a, Tapia et al., 2016]. Many methods get developed for understanding interactions although limitations reside in their applicability to NMOs. These methods can be classified in many ways but mainly on the semantics they use for modeling the functionality of the metabolism: graph-based semantics or constraint-based one. The lower level of knowledge about them, the lower-quality models of their metabolism. There thus is still plenty of room for the development of methods applicable to such organisms and their communities.

The concern of this thesis is to demonstrate the **applicability of combinatorial methods and logic programming** to two problems. The first objective addresses the **advances in gap-filling** steps of metabolic network reconstruction. It aims at refining the models so that they can handle metabolic objectives under particular semantics. The addition of missing information is performed via a selection within knowledge databases under various criteria. We address the question in the context of NMOs. In the second part, this thesis undertakes to tackle the problem of **community selection within a large microbiota**. To address the high combinatorics of the problem, a two-step incremental heuristic is chosen. This leads to proposing communities that minimize the number of involved symbionts and required exchanges while preventing the loss of information that occurs if only a single solution to the problem is provided.

Chapter 2, "Flexibility and accuracy of graph-based gap-filling", will endeavour to place the graph-based semantics of parsimonious combinatorial gap-filling in the landscape of completion methods. The place of Meneco [Prigent et al., 2017], a recent gap-filling method relying on the graph-based semantics of metabolic activation will be established with respect to its constraint-based counterparts [Satish Kumar et al., 2007, Vitkin and Shlomi, 2012, Thiele et al., 2014]. Methods and semantics will be compared using a large benchmark of gap-filling experiments. **Chapter 3, "Hybrid gap-filling reconciles graph-based and constraint-based formalisms"**, will undertake to reconcile the two major semantics of producibility in metabolism to create a hybrid gap-filling that satisfies their constraints [Frioux et al., 2017], by combining Answer Set Programming (ASP) with Linear Programming (LP) theory. The combination of logic programming with theory for computational biology can be very valuable, it has been done before using SAT solvers [Peres et al., 2014].

The application of the two gap-filling methods to algae is presented in the following **Chapter 4, "Application of gap-filling to non-model organisms"**. It demonstrates how the flexibility of combinatorial gap-filling is an asset in practice in the context of reconstructing the new version of genome-scale metabolic model (GSM) for *Ectocarpus siliculosus*. Then, hybrid gap-filling is shown to be useful to tackle the unblocking of specific parts of metabolic networks, for a red alga. The latest application of gap-filling described in this thesis bridges to the second main objective that is the study of communities of organisms. Combinatorial gap-filling is applied to explore complementarity between the brown alga *E. siliculosus* and one of its symbiotic bacterium *Candidatus* Phaeomarinobacter ectocarpi. Identification of putative metabolic interaction is automatically and systematically performed. Collaboration with biologists enables to remove false positive through the enhancement of genome annotations; and to select a number of metabolic functions that are not expected to be carried on by the alga alone based on existing genomic knowledge, but could be met provided cooperation with its symbiont.

The second part of the results, "**Scalability and combinatorics of community selection**", focuses on the community selection problem in microbiotas. Organisms live in interaction with others forming communities whose roles are not easy to capture in lab experimentation. Yet the role of interactions is crucial and deserves to be understood both to appreciate the dependencies of an organism of interest towards its microbiota, and to exploit cooperation between species for industrial applications for instance. **Chapter 5, "Formalism and combinatorics of community selection"**, proposes formalisms and a workflow for the selection of communities of bacteria within microbiotas for meeting objectives of interest. It describes how the issues of community selection can be addressed, and the implementation of a tool for this purpose. Notably, this chapter demonstrates that ASP solving assets and heuristics can be valuable to switch from a large microbiota to a small community in a two-step approach. The first step uses a similar simplification of the modeling as in [Eng and Borenstein, 2016] although with ASP it is possible to retrieve all optimal solutions and not just a single one. This is useful for the second step that is more precisely modeled, as it has been done before in a similar work [Julien-Laferrière et al., 2016]. The proposed workflow is then benchmarked on bacterial data from the gut microbiota. Finally, **Chapter 6, "Application of community selection algorithms"**, applies the developed methods to the study of optimal communities once again in the gut microbiota but with a focus on the human host. Finally, selection of communities for a brown alga is computed and experimentally tested in the lab by collaborators of Roscoff Biological Station.

This thesis covers the projects I was involved in during my PhD. In addition, this manuscript contains a description of the software projects I contributed to, presented in **Chapter 7, "Integrating heterogenous bioinformatics software in traceable workflows"**. My thesis research also encompasses published works. The gap-filling part of the thesis relies on the publication of the combinatorial parsimonious method in *PLOS Computational Biology* [Prigent et al., 2017] (Chapter 2). The hybrid gap-filling was presented at the *Logics Programming and Non-Monotonic Reasoning* Conference of 2017 [Frioux et al., 2017] (Chapter 3). It was awarded best student paper and elected for fast publication of an extended version of the paper in *Theory and Practice of Logic Programming* (in production). The applications of gap-filling (Chapter 4) were presented in *PLOS Computational Biology* [Aite et al., 2018] and [Prigent et al., 2017]. The selection of community described in the second part (Chapter 5 and 6) is accepted at the *European Conference on Computational Biology* of 2018 and the associated paper published in *Bioinformatics* [Frioux et al., 2018a].

Chapter 1

Metabolic studies at the service of non-model organisms and microbiotas

This chapter presents the state of the art regarding research on the different topics treated in this thesis. We will first introduce the shift between system biology that focuses on individual organisms to systems ecology that extends the latter in the context of communities of non-model organisms and microbiotas. We will present the main application domain of this thesis that is marine biology and particularly brown algae with *Ectocarpus siliculosus*. This seaweed's physiology is dependent to its microbiota and is thus interesting to study, in addition to being a model for brown algae. We will then describe existing methods to study interactions, notably metabolic ones, in microbiotas and how to select communities. Then a focus on metabolic producibility semantics and metabolic network reconstruction (Genome-Scale Model (GSM)) will be made before describing the particularities of gap-filling and its interest for Non-Model Organisms (NMOs) and microbiota studies.

1.1 From systems biology to systems ecology

1.1.1 Systems biology and the challenge of omics data integration to elucidate metabolism

The complexity of biological systems results from the interactions between biological elements that create complex behaviours [Kitano, 2002a]. Systems biology aims at examining organisms as a whole rather than isolate parts of them for individual studies [Kitano, 2002b]. The discipline is directly tied to the expression "the whole is greater than the sum of the parts".

Systems biology was introduced as a MeSH term (controlled vocabulary thesaurus used for indexing articles for PubMed) in the NCBI (National Center for Biotechnology Information) in 2005 and is defined as the following [1]: *"Comprehensive, methodical analysis of complex biological systems by monitoring responses to perturbations of biological processes. Large scale, computerized collection and analysis of the data are used to develop and test models of biological systems"*. As a comparison, **computational biology** was introduced in 1997: *"A field of biology concerned with the development of techniques for the collection and manipulation of biological data, and the use of such data to make biological discoveries or predictions. This field encompasses all computational methods and theories for solving biological problems including manipulation of models and datasets[2]"*.

Systems biology combines discovery science and hypothesis-driven science [Ideker et al., 2001]. The former aims at identifying all elements of a system for further digging information within it; an example was the sequencing of the human genome [Venter et al., 2001], forming a database of nearly 3 billion base-pairs. The latter, hypothesis-driven science, aims at formulating hypotheses that will be confirmed or contradicted experimentally.

Such systemic approaches were first carried out a hundred years ago with studies on homeostasis [Hood et al., 2008]. Contemporary systems biology has been on the rise for the last two decades due to the production of high-throughput data, notably with the sequencing of whole organisms [Ellegren, 2014] becoming easier and cheaper.

Therefore, a major concern of systems biology is to integrate data, extract biological knowledge and provide hypotheses to be tested. This is a challenge that entails the need of adequate methods in bioinformatics. They tackle the "omics" data that enable the study of components and interactions within the cell [Joyce and Palsson, 2006]. Notably, the field of metabolism can integrate a lot of this data by combining genomics, transcriptomics and proteomics [Oksman-Caldentey and Saito, 2005, Yizhak et al., 2010, Fiehn, 2001]. It consists in the study of the chemical transformations occurring in a cell via the action of proteins called enzymes. The combination of all these transformations forms a metabolic model, or metabolic network, and has many applications. Through adequate simulations, modeling the metabolism can for instance, guide metabolic engineering for synthetic biology purposes, form a basis for drug discovery or be used for studying interspecies interactions [Oberhardt et al., 2009]. Metabolism is thus a field of choice in systems biology to better understand the physiology of organisms.

[1] `https://www.ncbi.nlm.nih.gov/mesh/68049490`, accessed in June 2018

[2] `https://www.ncbi.nlm.nih.gov/mesh/68019295`, accessed in June 2018

1.1.2 Non-model organisms is the new standard

The first applications of newly developed methods related to biology often concern species called **model organisms**. They are species that were extensively studied notably for experimentation-based practical reasons, low complexity of the organism [Russell et al., 2017], relative closeness to human or representativity amongst a taxon. Among them we can cite the bacterium *Escherichia coli*, the yeast *Saccharomyces cerevisiae*, the worm *Caenorhabditis elegans*, the rat *Rattus norvegicus*, the mouse *Mus musculus*, the fruitfly *Drosophila melanogaster* or the plant *Arabidopsis thaliana*. The use of the expression "model organism" has since be applied to species that are widely used in specific fields [Leonelli and Ankeny, 2013] but we will only consider its first meaning here.

Together with the human, model organisms have been the very first species whose knowledge about benefited from high-throughput methods, in particular sequencing. With the enhancement and decreasing costs of sequencing technologies, genomes of thousands of organisms are available; yet the degree of knowledge about them is far below the previously cited ones. An organism with these characteristics is called a **Non-Model Organism (NMO)**. Most of the presently sequenced species are NMOs, with some taxonomic groups favoured such as mammals (> 1% of all species sequenced) over plants (0.01%) or insects (0.001%) [Ellegren, 2014].

As pointed out by [Leonelli and Ankeny, 2013], model organisms are considered representative of species beyond themselves. This is why in the field of metabolism for instance, the proximity between an NMO and a model organism is exploited to predict similar functions in the former (orthology). On the other hand, the prediction of functions that are specific to NMOs is more difficult due for instance to the difficulty to cultivate, isolate or genetically engineer them. Consequently, studies performed on NMOs are often performed by perturbing their environment rather than perturbing them locally as it can be done in routine for model organisms (e.g. gene modifications). This advocates for global modeling of the link between a phenotype and its environment. Systems biology is thus adequate to better understand NMOs. Yet methods need to be adapted to deal with the incompleteness of data and the lower amount of knowledge on these organisms.

To sum up

Systems biology aims at integrating resources and large scale data for the comprehensive study of biological systems. Doing so requires adequate methods provided by computational biology. Metabolism can be a level of study for biological systems through the integration of several "omics" fields.
Studying non-model organisms (NMOs), for which the level of knowledge is low, is an important application of systems biology. The integration of all the information and data about an NMO is a key point to address the challenge of understanding its physiology.

1.1.3 They don’t live alone: changes in organisms study paradigm

Organisms interact within the communities they form together, and also with their environment. The combination of all these interactions explains their physiology and leads to variations into the phenotypes they harbour. The interests of the scientific community for the study of microbiomes has been drastically rising over the last fifteen years, as shown on Figure 1.1. In 2017, more than 4,000 scientific publications were indexed in Pubmed with a title containing the words "microbiome" or "microbiota". This evolution is also directly tied to the enhancement and cost decreasing of high-throughput sequencing methods as explained earlier.

[Marchesi and Ravel, 2015] provide definitions for the main terms related to this field of research. **Microbiota** describes the microorganisms of a defined environment. The collection of their genomes is called the **metagenome**. And finally, the combination of these both, in addition to their habitat and environmental conditions is defined as the **microbiome**. We define as **symbionts** micro-organisms that belong to the microbiota of a host species, regardless of the type of their relationship; the latter is called **symbiosis**, a concept introduced by DeBary in 1879 [Moran, 2006]. We will use the term **holobiont** to consider the system composed by a host and its symbionts [Webster, 2014] regardless of the evolutionary considerations that are sometimes associated to this term. Such associations of organisms are sometimes called **metaorganisms** [Bosch and McFall-Ngai, 2011].

[Faust and Raes, 2012] and [Zuñiga et al., 2017] recap the different pairwise interactions that can occur within members of a microbiota or between a host and its symbionts. Mutualism occurs when both organisms gain benefit from the interaction. If only one benefits from it, it is qualified as commensalism, the other species having no benefit nor losses; or parasitism if the interaction is detrimental for the second species. If the interaction is detrimental for both, it is described as competition (Figure 1.2). Finally, neutralism occurs if the interaction is neutral for both partners.

The access to the huge amount of microbiomes data caused a change in the study of organisms. They are no longer considered as individuals but rather as a member of a community or a microbiota in which interactions occur. Plants (rhizosphere) [Glick, 1995, Van Der Heijden et al., 2008, Bais et al., 2006] and human microbiota [Gibson et al., 2004, Bäckhed et al., 2005, Sekirov et al., 2010, Tremaroli and Bäckhed, 2012] are among the most studied host-microbiome systems. Other well-studied systems include insects such as the pea aphid [Douglas, 1992, Brinza et al., 2009, Gauthier et al., 2015], bees [Engel et al., 2016], arthropods and *Wolbachia* bacteria [Werren et al., 2008] marine organisms (sponges [Thacker and Freeman, 2012], coral [Blackall et al., 2015] etc.) Studying the organisms individually, although convenient for certain species, is reductionist as they naturally do not live without biotic interactions [Cavaliere et al., 2017].

NMOs are the subject of a lot of microbiomes studies, either as hosts or symbionts. As a matter of facts, the study on some NMOs is empeded by growth constraints as it is sometimes impossible to grow them in axenic cultures [Provasoli and Pintner, 1980, Tapia et al., 2016]. This demonstrates the dependencies towards biotic interactions on one hand but also prevents a lot of experimental studies. Observations from synthetic design studies showed that interactions are expected to occur for metabolically expensive resources [Johns et al., 2016]. Therefore new solutions are needed to study these organisms and provide biological hypotheses to be tested. For instance, artificial and controlled communities to replace the native microbiome can be useful for elucidating the physiology of such organisms.

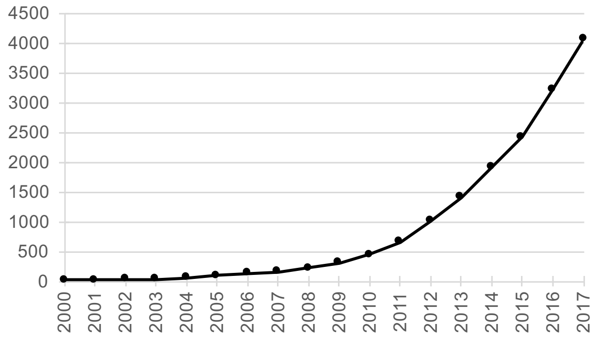


Figure 1.1: ***Evolution of the number of scientific publications treating microbiome or microbiota***

Number of publications with the words "microbiome" or "microbiota" in their title (y axis) by year (x axis), as indexed on Pubmed. The advanced search performed used the following query: ((microbiome[Title] OR microbiota[Title]) AND ("2017/01/01"[Date - Publication] : "2017/12/31"[Date - Publication]))

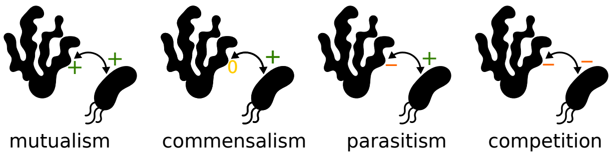


Figure 1.2: ***Classical ecological pairwise interactions***

"+" and "-" signs describe positive (win) and negative (loss) interactions respectively. A favourable, respectively detrimental, interaction for both species is a mutualism, resp. a competition. In case one partner benefits from it and not the other, it is qualified as commensalism provided it is neutral for the second partner, parasitism otherwise.

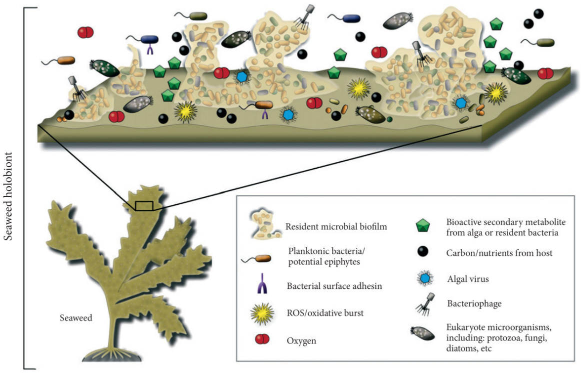

Seaweed holobiont
Seaweed
Resident microbial biofilm
Planktonic bacteria/
potential epiphytes
Bacterial surface adhesin
ROS/oxidative burst
Oxygen
Bioactive secondary metabolite
from alga or resident bacteria
Carbon/nutrients from host
Algal virus
Bacteriophage
Eukaryote microorganisms,
including: protozoa, fungi,
diatoms, etc

Figure 1.3: ***Dependencies of seaweeds to their biotic environment***

Picture from [Egan et al., 2013]. Seaweeds host and interact with a large range of organisms including bacteria and eukaryotes.

1.1.4 Marine biology: microbiomes and non-model organisms

The main biological domain of application of this thesis is marine biology and particularly brown algae. Brown algae are photosynthetic eukaryote seaweeds. They are distant from green plants like the well-studied *Arabidopsis thaliana*. The model of brown algae is the filamentous *Ectocarpus siliculosus*. Its genome has been sequenced in 2010, paving the way to a better understanding of these organisms [Cock et al., 2010]. The genome-scale metabolic model (GSM) of *Ectocarpus siliculosus* was reconstructed from this data four years later by [Prigent et al., 2014]. The latest version of *Ectocarpus siliculosus* GSM was published in [Aite et al., 2018] following a thorough work of re-annotation and enhancement on the brown algal genome by [Cormier et al., 2017].

The dependency of seaweeds towards their microbiome has been known for more than thirty years [Provasoli and Pintner, 1980]. It can notably occur through metabolic interactions such as for phytohormones production [Goecke et al., 2010]. The seaweed physiology is tightly related to its associated organisms [Egan et al., 2013], the whole holobiont forming an ecosystem that is both crucial and delicate to fully understand. Figure 1.3 extracted from [Egan et al., 2013] presents the complexity of the ecosystem on algal wall.

A first insight into the interest of studying *Ectocarpus siliculosus* together with its associated bacteria was shown by [Dittami et al., 2014a]: *Candidatus* Phaeomarinobacter ectocarpi, a bacterium frequently found with brown algae, was sequenced and its metabolic capabilities show complementarities with the ones of *E. siliculosus*. In 2016, [Tapia et al., 2016] demonstrated that, cultured in axenic conditions, *Ectocarpus* sp. displayed an altered phenotype: it no longer grows as filaments (Figure 1.4). The same year, [Dittami et al., 2016] shed light on the effect of interactions between the alga and its microbiota on the acclimation to salinity variations. *Ectocarpus* sp. appears to be a good candidate to investigate dependencies of brown algae to their microbiota, especially at the scale of metabolism [Dittami et al., 2014b]. A solution to do this is to rely on some bacteria, known to grow in association to seaweeds, and cultivable [KleinJan et al., 2017], to experimentally test complementarities between metabolisms.

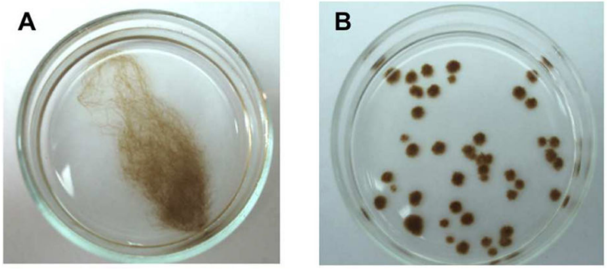


Figure 1.4: ***Dependencies of Ectocarpus sp. to its microbiota***

Photos from [Tapia et al., 2016]. ***A****. Ectocarpus sp. grown in native conditions displays a typical branched morphology.* ***B****. Culture in an axenic medium displays an altered phenotype.*

To sum up

It is widely acknowledged that symbioses play an important role in organisms physiology. Studies need to take theses interactions into account, which makes the unraveling of non-model organisms biology an even more complex subject.

In marine biology, brown algae physiology can be studied with *Ectocarpus siliculosus*. In particular, this seaweed presents strong dependencies to its microbiota that need to be further explored to be fully understood.

1.2 Investigating ecosystems and communities

1.2.1 Genomics studies of microbiota diversity

Studies relative to microbiota mainly rely on the genetic information that can be obtained for their members. Bacteria are the main group of microorganisms that are studied in microbiotas. A first approach to answer the question "who is there?" is to perform sequencing of **16S ribosomal RNA genes** whose high degree of conservation within species can be used to decipher Operational Taxonomic Units (OTUs). Although relatively cheap, this technique, also called **genomic survey**, has strong limitations as it does not enable to access the genetic contents of the organisms nor to identify organisms at the strain level [Xu and Zhao, 2018]. This is addressed by **metagenomics** studies that aims at sequencing bacteria within microbiota [Hornung et al., 2018]. They are completed by **metatranscriptomics** ones to gain information on the genes that are actually expressed by the species. The combination of both is useful. Indeed, the existence of a gene in a genome does not entail an actual expression of this gene and thus the effective presence of the enzyme, which is a limitation to be kept in mind when studying genome-scale models of metabolism. The bottleneck related to metagenomics and metatranscriptomics data is the assembly and assignment of genes to species which may have as a consequence non-complete genomes for some species and some genomic information that cannot be assigned [Xu and Zhao, 2018]. Yet, these techniques are undoubtly strong assets to study microbiota in which a large majority of species are not cultivable.

1.2.2 Co-occurrence and metabolic scores as markers of interactions

Starting from this genomic data, several types of computational analyses can be performed to better understand the functioning of interactions in microbiomes. [Li et al., 2016] surveyed the range of computational approaches to achieve this goal. **Co-occurrence networks** can be designed using metagenomics or 16S gene sequencing data [Friedman and Alm, 2012, Zelezniak et al., 2015]. Resulting networks display positive or negative relationships between taxa, provided the pair is expected to be associated, respectively not associated, in these conditions. Time-series or space conditions data [Faust and Raes, 2012] can be taken into account. [Berry and Widder, 2014] associates co-occurrence metrics to Lotka-Volterra numerical models to detect keystones species.

Many methods exploit the functions associated to genes to assess the interactions between species: this entails using **metabolic models** built from genomic data. Then, given the overlaps or complementarity between models, it is possible to evaluate the competition, respectively cooperation, potential within the community. [Kreimer et al., 2012] calculate the effective metabolic overlap (EMO) between two organisms, by comparing the topology of their metabolic models, to evaluate the competition between them. [Levy and Borenstein, 2013, Levy et al., 2015] describe the relationship between a host species and a parasitic or commensal one by two metrics. The first one is the biosynthetic support [Borenstein and Feldman, 2009] that aims at identifying the set of exogenously acquired metabolites (also called seeds) [Borenstein et al., 2008]. The more similar two sets of seeds are, the more likely the species compete. On the other hand, the metabolic complementarity index quantifies the cooperation score using the same sets of compounds. The higher number of seeds of one species being present in the metabolic model of the second one (but not in its seeds), the more likely cooperation will occur between them. A similar approach is used by [Cottret et al., 2010]. [Zelezniak et al., 2015] employ a similar scoring system, yet based on fluxes and not topology, with their metabolic resource overlap (MRO) and metabolic interaction potential. The first provides hints into competition potentials whereas the second aims at quantifying the cooperation through metabolic complementarity. [Freilich et al., 2011] compare expected growth rates of species in individual culture conditions to the rates in co-culture to qualify interactions within the pair of species. The work presented in [Mendes-Soares et al., 2016] is a first step towards the connection between pairwise interaction networks and metabolic modeling.

These scoring methods can be viewed as a first step to approach microbiota data and primarily express hypotheses on the type of pairwise interactions between species. They are completed by other methods that model the metabolism of the system for a deeper understanding of the processes involved.

1.2.3 Investigating ecosystems behaviours at the metabolic scale

Many questions arise when considering a community of organisms. Studying the metabolic interactions is of particular interest and can help to answer some of these questions through the use of metabolic models. However, the matter of the scale at which the modeling has to occur is crucial. Natural communities can involve thousands of species and identifying precise metabolic interactions at this scale is not currently feasible. It is more tractable when reducing the community to a few organisms forming synthetic communities [Blasche et al., 2017]. The scale issue also appears in the metabolic models themselves. They can be genome-scale or only account for the metabolic core of the metabolism. The precision and resolution of hypotheses made with the modelings depend on the scale of the study, notably for computational reasons.

Quantitative study of communities The first computational metabolic analysis of mutualistic interactions in a community was performed by [Stolyar et al., 2007]. The authors focused on *Desulfovibrio vulgaris* and *Methanococcus maripaludis*. They analyzed both metabolic networks and predicted flux distribution for a co-culture model. Through exchanges of methane and hydrogen, the community is expected to efficiently produce methane and acetate. [Klitgord and Segrè, 2010] reused this work to design growth medium that could guide small communities towards certain types of interactions (mutualism, neutralism etc.). In the meantime, their algorithm enables to identify putative exchanges within pairs of species. [Succurro and Ebenhöh, 2018, Gottstein et al., 2016] review the improvements and new developments made in the field of quantitative modeling for communities since this first study. [Zomorrodi and Maranas, 2012], [Khandelwal et al., 2013], [Henry et al., 2016], and [Budinich et al., 2017] have proposed methods for applying mathematical modeling to communities. Such methods have been applied to concrete case-studies [Ankrah et al., 2017, Koch et al., 2016]. Community models can be visualised using graphs [Granger et al., 2016].

Dynamics of communities Previous methods can be derived to dynamic analyses (see [Widder et al., 2016, van der Ark et al., 2017] for reviews). This can apply to small communities. In most cases, models are descriptive rather than predictive. [Zhuang et al., 2011] derived a model from their experimental data. [Zomorrodi et al., 2014] developed d-OptCom to observe evolutions of shared resources concentrations over time. [Hanemaaijer et al., 2017] use coarse-grained metabolic networks and experimental data to predict interaction fluxes between two bacteria. Altogether, dynamic modeling of communities needs experimental data

[Steinway et al., 2015] and is hardly applicable to large scale communities and NMOs. As stated by [Gottstein et al., 2016], applying such quantitative methods comes with limitations: models have to be of high quality in order to obtain reliable results. The quality of metabolic networks however is not an easy objective to meet, especially for NMOs. [Mendes-Soares and Chia, 2017] further emphasizes the fact that mathematical modelings are computationally demanding and are not suitable for the analysis of large and complex communities. An alternative is to study metabolism functionality with a **graph-based modeling of communities**. [Ofaim et al., 2017] used a graph-based metabolic activity applied to metagenomics (rhizosphere). [Opatovsky et al., 2018] studied the possible pairwise interactions in between a non-model organism, the whitefly *Bemisia tabaci*, and five of its symbionts. The study was performed using graph-derived techniques [Ebenhöh et al., 2004, Kreimer et al., 2012].

To sum up

A landscape of methods coexists for studying interactions in communities, many of them using metabolic networks. Their applicability to non-model organisms and scalability vary and are related to the resolution of the modeling as well as the needed data (quality of network reconstructions, experimentations).

1.2.4 Selection of communities

A natural follow-up of microbiota studies is to select communities of interest that are able to meet a metabolic objective [Johns et al., 2016]. This can occur in several contexts ranging from synthetic community design for industrial purposes [Julien-Laferrière et al., 2016] to a reduction of microbiota complexity for understanding cooperation in specific functions. Once again the levels of modeling differ and a general pattern follows a resolution-scale paradox: in order to scale to large microbiotas, constraints have to be lowered whereas they can be strict when few species are at stake.

Compartmentalization in communities [Henry et al., 2016, Faria et al., 2016] present several levels of accuracy in analyzing communities. In a compartmentalized system, organisms are independent and exchanges need to be characterized. In a mixed-bag or "soup" system, a metaorganism concentrates the metabolic capabilities of all organisms. No interactions are taken into account, everything is considered shareable at no cost. The first system is accurate but computationally demanding, the second one can be solved easily but the price to pay is a risk to overestimate the community capabilities.

As an example, the non-compartmentalized formalism choice was chosen by [Greenblum et al., 2012] to evaluate the effect of obesity and inflammatory bowel disease on the gut microbiota metabolic capabilities. Similarly [Abubucker et al., 2012] propose a workflow to build metabolic models starting from short DNA sequence reads. Although the information about "who is there" is available, the final model merges all metabolic pathways in a supra-organism. The resulting dataset provides information about the metabolic pathways present in the microbiota of each condition as well as their relative abundances.

Community selection in a mixed-bag framework [Eng and Borenstein, 2016] developed a method, CoMiDA, for the selection of communities within large microbiota under a defined objective that is the achievement of a metabolic function and defined culture conditions, represented by the existence of available metabolites. They base their algorithm upon a network-flow inspired modeling and use Integer Linear Programming (ILP) to select a minimal-size community that achieves the objective. They tested their algorithm by studying 10,000 functions of the Human Microbiome Project (HMP) [Human Microbiome Project, 2012]. Such function is the combination of two metabolites, one input/substrate and one output/product. The metabolic objective is reached if the product is synthesizable from the substrate using the metabolic capabilities of the selected community. These functions were tested under a non-compartmentalized framework, that is to say that chosen bacteria share all their metabolic capabilities which is an asset to support large datasets such as the HMP, that contains 2,051 bacteria and their associated networks and 2,252 unique metabolites. The authors showed that a community satisfies the problem in less than 3% of cases if allowing multiple reactants/products by reactions. Otherwise, if reactions are simplified, a solution exists in nearly 40% of cases, mostly of size one, which means that a community is not needed and the function is intrinsically met by a single bacterium.

Community selection in a compartmentalized framework [Julien-Laferrière et al., 2016] developed MultiPus for the purpose of designing synthetic microbial consortia starting from sets of available bacteria and the possibility to genetically engineer species with a database of metabolic functions (reactions). They present a algorithm that solves the NP-hard Directed Steiner Tree problem and eventually solve the community selection problem with Answer Set Programming (ASP) with the following optimizations:

– minimize the number of exogenous reactions to be added to the community
– minimize transports in the model, ie required exchanges between members of the selected community

In practice, reactions of the system are weighted. Transport reactions and exogenous reactions are more costly that reactions belonging the the individual metabolic models of the organisms. No optimization is made on the number of species involved in the community. They applied the tool to the selection of a community for antibiotics production within a consortium of 4 species. Then they find solutions to produce 1,3-propanediol and methane starting from two bacteria and a database of exogenous reactions.

To sum up

Studies on ecosystems range from the identification of interactions to the selection of communities within microbiotas. Metabolic networks seem to be a good solution to do so but trade-offs between resolution in modeling and scaling have to be taken into account. This is particularly true for non-model organisms for which knowledge levels are low. How are metabolic networks built and how is functionality expressed? Are there specificities in their reconstruction in the context of systems ecology? These questions will be adressed in the following sections.

1.3 Modeling the activity of metabolism

1.3.1 Metabolic networks

A majority of the methods described earlier rely on the study of metabolism to analyse ecosystems and communities. Here I provide descriptions of the objects and formalisms associated to mathematical abstractions of metabolism.

Genes encode a lot of information including sequences corresponding to protein called enzymes that catalyze metabolic reactions. These enzymes result from the translation of mRNA into peptides and proteins that will be matured, and the mRNA itself comes from the transcription of DNA. Metabolic models describe metabolic capabilities of these enzymes: which reactions they are able to catalyze. The reactions and the metabolic compounds they involve are at the center of metabolic modeling. Metabolic models, also called metabolic networks, range from pathways that are sets of reactions associated to a particular metabolic function, to genome-wide or genome-scale metabolic models (GSM). The latter aim to gather all metabolic reactions occurring in the cell or the organisms. The first GSM was reconstructed in 1999 for *Haemophilus influenzae* [Edwards and Palsson, 1999] using annotation of its genome. Ever since, hundreds of models have been constructed and refined for a wide range of living organisms [Kim et al., 2012].

Definition 1.1 Metabolic network *A metabolic network can be defined as a* ***labelled directed bipartite graph*** *G*

$$G = (R \cup M, E, s)$$

where R and M are sets of nodes standing for reactions and compounds (also called metabolites), respectively. When $(m,r) \in E$ or $(r,m) \in E$ for $m \in M$ and $r \in R$, the metabolite m is called a reactant or a product of reaction r, respectively. Importantly metabolites and reactions nodes can both have multiple ingoing and outgoing edges. For any $r \in R$, define

$$reactants(r) = \{m \in M \mid (m,r) \in E\}$$

$$products(r) = \{m \in M \mid (r,m) \in E\}$$

The edge labelling $s : E \to \mathbb{R}$ gives the ***stoichiometric coefficients of a reaction's reactants and products****, respectively, i.e., their relative quantities involved in the reaction. Finally, the activity rate of reactions or "flux" is bound by lower and upper bounds, denoted by $lb_r \in \mathbb{R}_0^+$ and $ub_r \in \mathbb{R}_0^+$ for $r \in R$, respectively.*

Whenever clear from the context, we refer to metabolic networks with G (or G', etc) and denote the associated reactions and compounds with M and R (or M', R' etc.), respectively.

Note that another representation of metabolic models exists, based on hypergraphs. We will only focus in this thesis on the bipartite representation in formalisms. Pleease refer to [Cottret and Jourdan, 2010] for details on hypergraphs. Notice that some figures of the thesis will adopt an hypergraph-like representation by abstracting the reaction nodes for the sake of simplification.

Example In the small example of Figure 1.5, the metabolic model comprises four reactions: $R = \{r_1, r_{2f}, r_{2b}, r_3\}$. For instance, reaction r_1 has two reactants $reactants(r_1) = \{A, B\}$. By

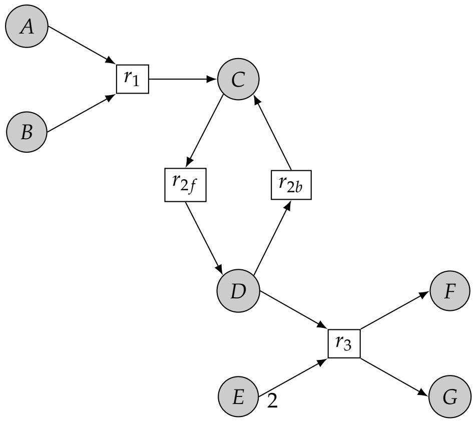


Figure 1.5: ***A small metabolic network - bipartite graph representation***

Circles are compounds nodes, squares are reactions nodes. Edges describe reactant (resp. product) relationships between compounds and reactions (resp. reactions and compounds). Labelling of the edges describe the stoichiometries of the metabolites in the reaction. Absence of labelling accounts for a stoichiometry of 1 by default.

default, if no other indication, the stoichiometric coefficient of reactants and compounds are 1. In reaction r_3 however, the stoichiometric coefficient of reactant E is 2 $s(E, r_3) = 2$, meaning that two molecules of E and one molecule of D produce one molecule of F and one molecule of G: $1D + 2E \rightarrow 1F + 1G$.

Producibility and functionality in metabolic networks Based on the contents of a GSM in terms of reactions and metabolites, there are several methods to model the functioning of the metabolism. Indeed, comparing the models based on their contents only is one static way to exploit the data; but another way is to model producibility of metabolites, accessibility of reactions in the networks, based on ad-hoc definitions and formalisms. Concretely, modeling producibility aims at creating a sub-network containing metabolites and reactions that can be reached given specific conditions such as the growth medium composition. It can also be used to ensure that a particular target reaction or set of metabolites is accessible. There are **two main formalisms to describe producibility in metabolic models: graph-based and constraint-based semantics** that are complementary one to each other: the first one is well adapted to non stationary states whereas the second is fitted to model stationary states [Kruse and Ebenhöh, 2008].

1.3.2 Graph-based semantics

Definition The study of the graph connectivity, shape and topology has been widely used for biological networks, including GSMs [Raymond and Segrè, 2006, Goldford and Segrè, 2018] The notion of producibility under the graph-based modeling lies on the definition of **scope** that was introduced by [Ebenhöh et al., 2004] and [Handorf et al., 2005]. Numerical parameters such as stoichiometry, enzyme kinetics, or reaction fluxes are ignored in topological modeling. The focus is made on the topology of the graph and the connectivity of the nodes [Manor et al., 2014]. Therefore, let us consider a metabolic network $G = \{R \cup M, E\}$ without the stoichiometric information that was included in the general definition (Def. 1.1).

The computation of the **Graph-Based (GB) producibility** requires to consider a set of metabolites $S \subseteq M$ called **seeds** that represents available compounds, for instance the ones composing the growth medium. In these definitions, seeds are given as inputs, they can consist in growth medium metabolites for instance. They are not the seeds defined by [Borenstein et al., 2008] and [Carr and Borenstein, 2012] extracted from strongly connected components of the graph. Two kinds of seeds can be distinguished here.

- *initiation seeds*, that is, compounds initially present due to experimental evidence and given as inputs
- *boundary seeds*, another set of compounds that is assumed to be activated by default due to the graph topology. Some reactions may have no reactants although they have products in metabolic models. In this case, such products can be considered as seeds. These *boundary compounds* are defined as:

$$S_b\,(G) = \{m \in M \mid r \in R, m \in products(r), reactants(r) = \emptyset\} \tag{1.1}$$

For simplicity, we will consider a single set of seeds S, that is defined according to one or both previous definitions. The set of reachable metabolites from S, called *scope*, is written $\Sigma_G(S)$ and is recursively computed. A metabolite $m \in M$ is *reachable* from S if $m \in S$ or if $m \in products(r)$ for some reaction $r \in R$ where all $m' \in reactants(r)$ are reachable from S.

Definition 1.2 Scope definition *Starting from the seeds, the scope can be iteratively updated until it reaches a fixed point.*

$$\Sigma_G(S)_{i+1} = \Sigma_G(S)_i \cup products(r) \; for \; \{r \in R \mid m \in Reactants(M_i), \forall m, (r,m) \in E\}), \; with \; M_O = S$$

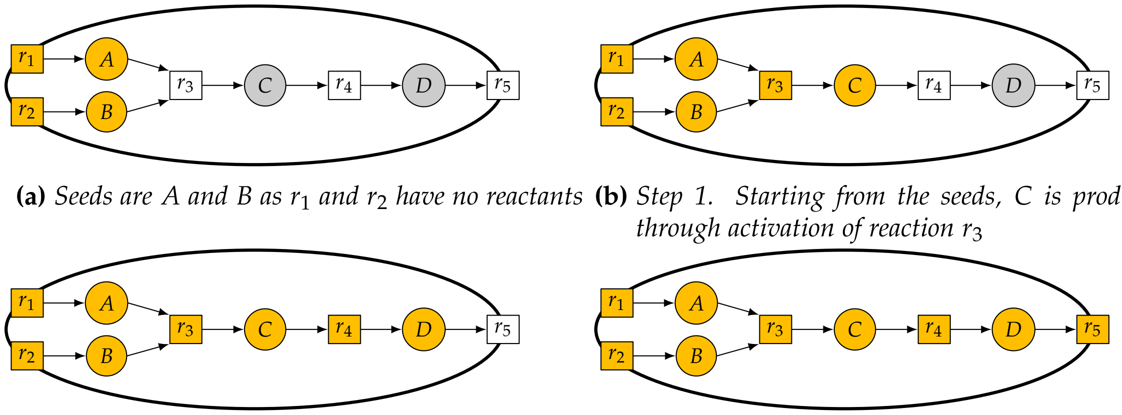
Hence, following the scope definition, a metabolite is **reachable** if it belongs to the scope of the seeds $m \in \Sigma_G(S)$. More generally, with a focus on reactions rather than metabolites, we can define the **activation of a reaction**. A reaction is topologically-activated if all of its reactants belong to the scope of the seeds.

Definition 1.3 Graph-based activation of a reaction. *Given an objective reaction $r_{obj} \in R_{obj}$, a metabolic network $G = (R \cup M, E)$ and a set of seeds S, topological activation is defined as follows:*

$$r_{obj} \in active^t_G(S) \;\; iff \;\; reactants(r_{obj}) \subseteq \Sigma_G(S).$$

Figure 1.6 describes the activation of reactions and the producibility of compounds through the scope computation in a small metabolic model.

Applications Graph-based semantics has been used in complementarity to constraint-based methods in [Laniau et al., 2017]. In terms of applications for the reconstruction of metabolic models, [Collet et al., 2013] expanded the GSM of *Ectocarpus siliculosus* and this gap-filling technique was implemented into the Meneco package in [Prigent et al., 2017]. Topological studies of graphs have been applied to deciphering microbial interactions [Greenblum et al., 2012]. More importantly, in the context of microbiota study, topological producibility semantics has not only been applied to the modeling of co-occurrence or supra-organisms as stated by [Heinken et al., 2016]; it is suitable to model interactions within compartmentalized organisms [Julien-Laferrière et al., 2016].



(a) *Seeds are A and B as r_1 and r_2 have no reactants*

(b) *Step 1. Starting from the seeds, C is producible through activation of reaction r_3*

(c) *Step 2. C is the only reactant of r_4, which enables its activation and the producibility of its product D.*

(d) *Step 3. D is the only reactant of r_5, which enables its activation: D can be exported out of the system.*

Figure 1.6: ***Graph-based "scope" producibility in metabolic networks.***

The ellipse describes the system: an organism or a cell that comprises 5 reactions and 4 metabolites. Reactions without reactants (r_1 and r_2) are imports: A and B can be imported into the system: they are seeds. r_5 has no products, it symbolizes an export: D can be exported out of the system.

Cycle activation The GB-modeling of GSMs can be interpreted as a modeling of the initial state of the system. Indeed, the activation of every reaction has to be initiated starting from the seeds which means that prior to this, the whole system is considered as turned-off, non-functional. This is an asset as a stringent definition of reaction activation gives credit to the hypothesis that it is truly activable physiologically, in nature or in lab experiments. Yet the strict nature of the definition also leads to problems in modeling, in particular when observing the cycles of metabolic models. Such case is presented in Figure 1.7. In this example, the seed set, composed of $\{S_1, S_2, S_3\}$ activates reactions $\{r_0, r_1, r_6\}$ which results in the accessibility of the metabolites $\{S_1, S_2, S_3, A, B, G\}$. Notice the cycle composed of reactions $\{r_2, r_3, r_4\}$. By applying the strict definition of GB-producibility, the cycle can never be initiated since reaction r_2 requires C in addition of S_2, the former being itself produced by the activation of the cycle. Once activated, provided S_2 is available at an unlimited rate, the cycle can always sustain its activity since the production of 2 molecules of C can both feed the cycle and supply reaction r_5. The crucial point thus lies in the first activation of the cycle at the initial state. Two solutions exist for this purpose.

- Either the strict initial state is considered; in this case the cycle can never be activated unless a reaction is missing in the system and would produce one metabolite of the cycle, such as D or E. This can happen as annotation can be error-prone, especially for non-model organisms. Thus it could make sense to add a reaction if it exists in databases or related organisms and if a gene can be associated to it.
- Either we consider that the cycle has been activated somehow because the current modeling does not aim at explaining the initial state. In this case, a solution is to model a pseudo-initial state and consider that some metabolites are expected to be producible in the system, that they for sure can be produced naturally in the organism. This is often the case for a certain class of metabolites called cofactors. They are small molecules that participate to many reactions and which equilibrium relies on cycles. For example, ATP, NADH or CoA etc. belong to this category. These molecules are considered as already producible in some experimentations. For instance [Eng and Borenstein, 2016] used the currency metabolites identified by [Greenblum et al., 2012] and excluded from the reac-

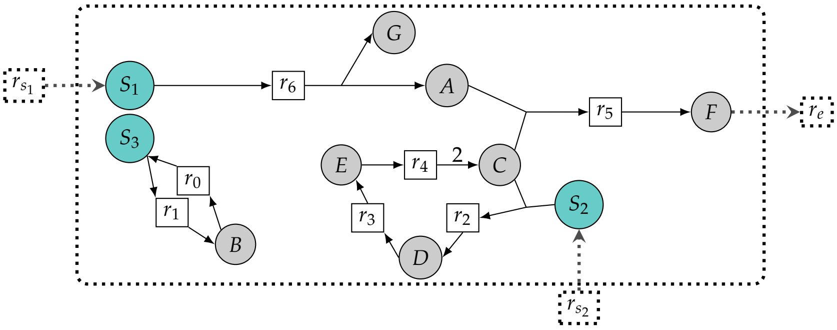


Figure 1.7: ***Effect of cycles on the graph-based concept of producibility***

Blue circles are the seeds of the system. Following the definition of seeds provided above, S_1 and S_2 are boundary seeds (cf. Equation 1.1). S_3 belongs to the second type of seeds, the initiation seeds, as there is no reaction producing it from nothing in the model. The cycle composed of metabolites $\{S_2, C, D, E\}$ and reactions $\{r_2, r_3, r_4\}$ is not activable because S_2 needs to be associated to C, that is itself a product of the cycle, to trigger reaction r_2.

tions [Greenblum et al., 2012]; [Cottret et al., 2010, Julien-Laferrière et al., 2016] ignored the side-compounds. Another modeling solution is to include them in the seeds, in this case the *initiation seeds* that were described above. In the example of Figure 1.7, unblocking the cycle could be achieved by adding D or E in the seeds set, after a study of their biological role.

To sum up

Graph-based semantic can efficiently describe the producibility in a metabolic model. It ignores stoichiometry, which is an asset when GSMs are incomplete or non-precise as it is sometimes the case in the middle of GSM generation or for non-model organisms.

1.3.3 Constraint-based semantics

A second way of modeling the functionality of a metabolic model is to rely on **Constraint-Based (CB)** methods.

Definition Each reaction $r \in R$ is associated to a value, the flux v_r that models its activity rate; usually millimoles per gram dry weight per hour (mmol.gDW^{-1}.hr^{-1}). Each reaction has two bounds to constraint its flux distribution: a *lower bound* (LB) and an *upper bound* (UB). Stoichiometry is taken into account in CB modeling, its precision is even essential as the model is mathematically represented as a stoichiometric matrix S [Maranas and Zomorrodi, 2016]. The latter has the reactions (rxn) of the model as columns and its compounds (cpd) as lines.

$$\begin{array}{c c} \text{S} & \begin{array}{cccc} rxn_1 & rxn_2 & \dots & rxn_n \end{array} \\ \begin{array}{c} cpd_1 \\ cpd_2 \\ \vdots \\ cpd_n \end{array} & \begin{pmatrix} s(cpd_1, rxn_1) & s(cpd_1, rxn_2) & \dots & s(cpd_1, rxn_n) \\ s(cpd_2, rxn_1) & s(cpd_2, rxn_2) & \dots & s(cpd_2, rxn_n) \\ \vdots & \vdots & \ddots & \vdots \\ s(cpd_n, rxn_1) & s(cpd_n, rxn_2) & \dots & s(cpd_1, rxn_n) \end{pmatrix} \end{array}$$

The metabolic network example of Figure 1.5 contains the following four reactions:

$$\begin{aligned} r_1 &: & 1A + 1B &\rightarrow 1C \\ r_{2f} &: & 1C &\rightarrow 1D \\ r_{2b} &: & 1D &\rightarrow 1C \\ r_3 &: & 1D + 2E &\rightarrow 1F + 1G \end{aligned}$$

The resulting stoichiometric matrix S is:

$$\begin{array}{c} \\ A \\ B \\ C \\ D \\ E \\ F \\ G \end{array} \begin{array}{c} \begin{array}{cccc} r_1 & r_{2f} & r_{2b} & r_3 \end{array} \\ \begin{pmatrix} -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ +1 & -1 & +1 & 0 \\ 0 & +1 & -1 & -1 \\ 0 & 0 & 0 & -2 \\ 0 & 0 & 0 & +1 \\ 0 & 0 & 0 & +1 \end{pmatrix} \end{array}$$

With this formalism in mind, it is possible to model flux distribution in the reactions of the network to optimize defined objectives. Flux Balance Analysis (FBA) is addressing this question.

Flux Balance Analysis FBA is a mathematical optimization of flux distribution in a network. The purpose is to find a distribution of each reaction flux in the model such that the flux z of a specified objective function r_{obj} is maximized (or minimized). This is a solvable problem in Linear Programming (LP) provided the system has reached the steady-state, that is to say the amount of each compound being produced is equal to its amount being consumed [Orth et al., 2010]. The following definition [Maranas and Zomorrodi, 2016] presents the optimization problem.

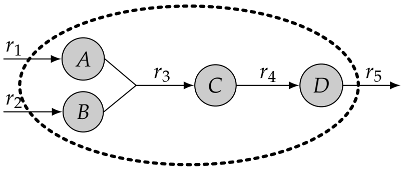
Definition 1.4 FBA optimization problem. *The FBA optimization formulation is the following (matrix display on the left, numerical on the right):*

$$\begin{aligned} \text{maximize (or minimize) } z = v_{r_{obj}} \qquad & z = \sum_{r_{obj} \in R_{obj}} c_{r_{obj}} v_{r_{obj}} & \\ \text{subject to } S.v = 0 \qquad & \sum_{r \in R} S_{mr} v_r = 0, \forall m \in M & (1.2) \\ LB \leq v \leq UB \qquad & LB \leq v_r \leq UB & (1.3) \\ v \in \mathbb{R} \qquad & v_r \in \mathbb{R} & \end{aligned}$$

Figure 1.8 applies the FBA optimization to a small example, the same one as in Figure 1.6 for graph-based scope. FBA can be used to assess the stoichiometric or CB-activation of a reaction the same way the scope can be used to assess its topological or GB-activation.

Definition 1.5 Constraint-based activation of a reaction. *Given an objective reaction $r_{obj} \in R_{obj}$, a metabolic network $G = (R \cup M, E, s)$ and a set of seeds S, stoichiometric activation is defined as follows:*

$$r_{obj} \in active^s_G(S) \;\; \textit{iff} \;\; v_{r_{obj}} > 0 \textit{ and Equations } (1.2) \textit{ and } (1.3) \textit{ hold for } M \textit{ and } R.$$



$$\begin{array}{c|ccccc|} & r_1 & r_2 & r_3 & r_4 & r_5 \\ A & +1 & 0 & -1 & 0 & 0 \\ B & 0 & +1 & -1 & 0 & 0 \\ C & 0 & 0 & +1 & -1 & 0 \\ D & 0 & 0 & 0 & +1 & -1 \end{array} = S$$

$$S.\begin{pmatrix} v_{r_1} \\ v_{r_2} \\ v_{r_3} \\ v_{r_4} \\ v_{r_5} \end{pmatrix} = 0$$

$$\text{maximize}\, z = v_{r_4}$$
$$0 \leq v \leq 2$$

Optimization leads to
the following distribution of fluxes

$$\begin{pmatrix} v_{r_1} \\ v_{r_2} \\ v_{r_3} \\ v_{r_4} \\ v_{r_5} \end{pmatrix} = \begin{pmatrix} 2 \\ 2 \\ 2 \\ 2 \\ 2 \end{pmatrix}$$

$$\frac{d[A]}{dt} = 0 \Leftrightarrow \times v_{r_1} - 1 \times v_{r_3} = 0$$
$$\frac{d[B]}{dt} = 0 \Leftrightarrow 1 \times v_{r_2} - 1 \times v_{r_3} = 0$$
$$\frac{d[C]}{dt} = 0 \Leftrightarrow 1 \times v_{r_3} - 1 \times v_{r_4} = 0$$
$$\frac{d[D]}{dt} = 0 \Leftrightarrow 1 \times v_{r_4} - 1 \times v_{r_5} = 0$$

Figure 1.8: ***Application of FBA to a small example.***

Adapted from [Kim and Lun, 2014]. The system consists in five reactions, represented as an hypergraph. r_1 and r_2 model the production of A and B the two boundary seeds. r_5 aims at preventing the accumulation of metabolite D. The associated stoichiometric matrix can be built. Using FBA, the objective is to maximize flux in reaction r_4 that produces D. The solving of the equation $S.v = 0$ that models the steady-state, together with the bounds of the reactions flux gives a single solution to the system: all fluxes set to 2 mmol.gDW^{-1}.hr^{-1}. The flux in the objective reaction r_4 being positive, it is constraint-based or stoichiometrically activated.

Note that the condition $v_{r_{obj}} > 0$ strengthens the flux condition for $r_{obj} \in R$ in the second part. More generally, observe that activated target reactions are not directly related to the network's seeds S. However, the activation of targets highly depends on the boundary compounds in $S_b(G)$ for which (3.1) is always satisfied and thus initiates the fluxes. Since boundary compounds are produced by at least one reaction without prerequisite, an arbitrary amount might be produced. Therefore, the incoming flux value always balances the sum of the flux values associated to outgoing edges. Intuitively, boundary compounds are nutrients that are expected to be available in the system (growth medium) for the consumption by the metabolic network, thus initiating the reactions within.

A relaxed definition of activation can be defined by slightly modifiying the constraint-based activation dependencies. It consists in allowing the accumulation of metabolites in the model, thus relaxing the steady-state assumption with the following:

$$\sum_{r \in R} S_{mr} v_r \geq 0, \forall m \in M \tag{1.4}$$

The relaxed constraint-based definition becomes:

Definition 1.6 Relaxed constraint-based activation of a reaction. *Given an objective reaction $r_{obj} \in R_{obj}$, a metabolic network $G = (R \cup M, E, s)$ and a set of seeds S, stoichiometric activation is defined as follows:*

$$r_{obj} \in active^{r}_{G}(S) \;\; \textit{iff} \;\; v_{r_{obj}} > 0 \textit{ and Equations } (1.4) \textit{ and } (1.3) \textit{ hold for } M \textit{ and } R.$$

Steady-state vs initial state modeling As pointed above, FBA relies on the steady-state assumption, which entails that the CB-activation of reactions relies on it as well. This is a major difference with the GB-based activation which modeled on the contrary the initial state of the system. This difference is well-observed in Figure 1.7. In this example, r_{s_1} and r_{s_2} are the reactions (producing the boundary seeds in the GB-modeling) that connect the system with the external compartments. Yet, because of the steady-state assumption, the cycle is balanced and activated. Isolated in a smaller system in Figure 1.9 in which the consumption of C by r_5 is replaced by an export reaction, we can design the stoichiometric matrix and its associated equations under the steady-state assumption described by Equation 1.2. The stoichiometric matrix corresponding to Figure 1.9 is the following:

$$\begin{array}{c} \\ S_2 \\ C \\ D \\ E \end{array} \begin{array}{c} \begin{array}{ccccc} r_{s_2} & r_2 & r_3 & r_4 & r_e \end{array} \\ \begin{pmatrix} +1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & +2 & -1 \\ 0 & +1 & -1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 \end{pmatrix} \end{array} = S$$

Under the steady-state assumption, equation 1.2 holds thus $S.v = 0$ with $v = \begin{pmatrix} v_{rs_2} \\ v_{r_2} \\ v_{r_3} \\ v_{r_4} \\ v_{r_e} \end{pmatrix}$. The following equations can be extracted for each metabolite:

$$\begin{aligned} \frac{d[S_2]}{dt} &= 0 \Leftrightarrow v_{r_{S_2}} - v_{r_2} = 0 \\ \frac{d[C]}{dt} &= 0 \Leftrightarrow 2v_{r_4} - v_{r_2} - v_{r_e} = 0 \\ \frac{d[D]}{dt} &= 0 \Leftrightarrow v_{r_2} - v_{r_3} = 0 \\ \frac{d[E]}{dt} &= 0 \Leftrightarrow v_{r_3} - v_{r_4} = 0. \end{aligned}$$

This system of equations is easily solvable: $v_{r_{S_2}} = v_{r_2} = v_{r_3} = v_{r_4} = v_{r_e}$. Values of the fluxes could be precised using the bounds constraints of the fluxes (see Eq. 1.3). The conclusion of this example is that the cycle is activated under CB-modeling, whereas it was not under GB-modeling.

Thermodynamically infeasible cycles The previous kind of cycles must not be mistaken for Thermodynamically Infeasible Cycles (TIC) that involve only internal fluxes and can also be

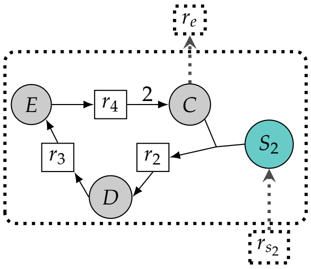
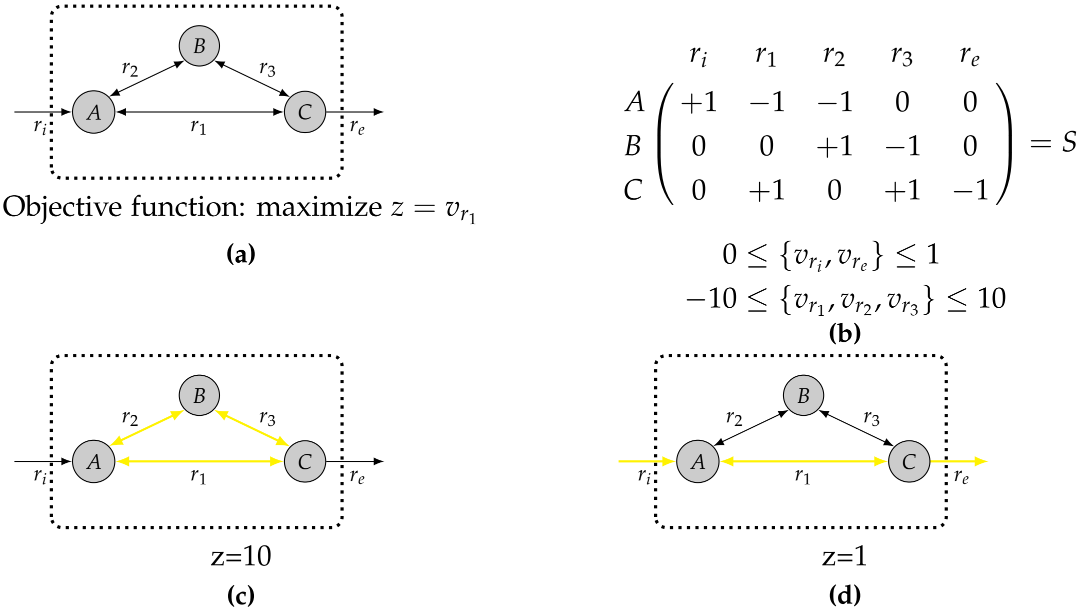


Figure 1.9: ***Effect of cycles on the constraint-based concept of producibility***

The cycle og this model is extracted from the model of Figure 1.7. At steady-state, the presence of the boudary seed S_2 and the production of C via the cycle is sufficient to consider the cycle as activable. This is not the case with a GB modeling of producibility.

functional in FBA. As only stoichiometry constraints flux distribution in FBA, it can occur that a cycle is functional without external inputs of energy which violates the second law of thermodynamics [Maranas and Zomorrodi, 2016, Desouki et al., 2015]. As stated by [Schellenberger et al., 2011a], there cannot be a net flux in a closed cycle. An example of TIC is provided in figure 1.10.



Objective function: maximize $z = v_{r_1}$

(a)

$$\begin{array}{c} \\ A \\ B \\ C \end{array} \begin{array}{c} \begin{array}{ccccc} r_i & r_1 & r_2 & r_3 & r_e \end{array} \\ \begin{pmatrix} +1 & -1 & -1 & 0 & 0 \\ 0 & 0 & +1 & -1 & 0 \\ 0 & +1 & 0 & +1 & -1 \end{pmatrix} \end{array} = S$$

$$0 \leq \{v_{r_i}, v_{r_e}\} \leq 1$$
$$-10 \leq \{v_{r_1}, v_{r_2}, v_{r_3}\} \leq 10$$

(b)

z=10

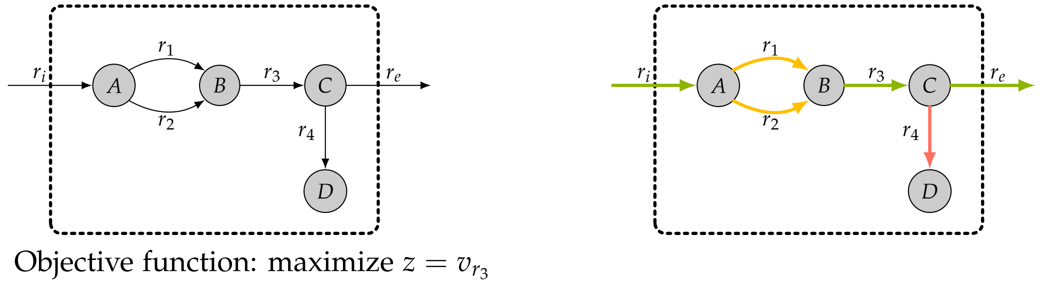
(c)

z=1

(d)

Figure 1.10: ***Example of a thermodynamically infeasible cycle***

Reproduced and adapted from [Schellenberger et al., 2011a] (a) Structure of the system. It is composed of three metabolites and 5 reactions, among which one import and one export for connection with the environment. Notice the reversibility of reactions $\{r_1, r_2, r_3\}$. They could each be replaced by two irreversible reactions, one forward, one backward. Reaction flux to be maximized is v_{r_1}. (b) Stoichiometric matrix describing the model and bounds for reactions fluxes. (c) FBA optimization proposes a flux value of 10 in r_1 which is possible is fluxes are distributed with the $\{r_1, r_2, r_3\}$ cycle. However this cycle is a close loop which is thermodynamically infeasible. (d) A thermodynamically feasible distribution of fluxes is through reactions $\{r_i, r_1, r_e\}$ that leads to a flux value of 1.



Objective function: maximize $z = v_{r_3}$

(a)

(b)

Figure 1.11: ***Flux Variability Analysis***

The objective function of this example is to maximize the flux in reaction r_3. There are several flux distributions in the system that enable such maximization. FVA calculates these distributions and classifies reactions based on the flux range they harbour in all distributions. $\{r_i, r_2, r_e\}$ are essential reactions (green): the flux through them is always positive to enable the optimization of the objective function. $\{r_1, r_2\}$ are alternative reactions (orange). Flux can flow through one or the other or be splitted in the two of them: there exist distributions in which flux of the alternative reaction is positive and others in which it is null. Finally r_4 is a blocked reaction (red). Having flux in this reaction would lead to an accumulation of D which is prohibited at steady state. Thus it is never activated.

Flux Variability Analysis There might be several flux distributions in the model such that the flux in the objective function is maximum (or minimum). The Flux Variability Analysis (FVA) problem [Mahadevan and Schilling, 2003] derives from FBA and aims at finding the range of flux distributions for each reaction of the model under the previous optimization. This results in a range of fluxes for each reaction, which then can be classified into three categories:

– *essential reaction*: the flux range is strictly positive (or negative if the reaction is reversible and occurs in the backward direction)
– *blocked reaction*: the flux through the considered reaction is always null, meaning the reaction cannot be active for the maximization (or minimization) of flux in the objective function.
– *alternative reaction*: the flux can be either positive (or negative if the reaction is reversible and occurs in the backward direction) or null. This entails that a production path involving this reaction exists but another path without this reaction also exists.

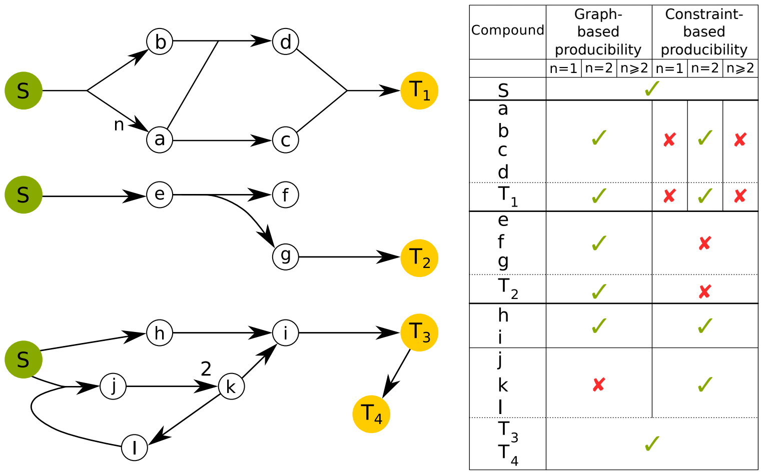
Figure 1.11 describes a minimal example with reactions of the three categories.

To sum up

Constraint-based producibility allows to quantitatively model the functionality of the metabolism. It relies on the modeling of the steady-state.

1.3.4 Comparison of graph-based and constraint-based producibility definitions

The two definitions of activation and reachability can be compared. FIgure 1.12 describes the producibility of metabolites in small networks according to the chosen formalism. It confirms



Compound	Graph-based producibility			Constraint-based producibility		
	n=1	n=2	n≥2	n=1	n=2	n≥2
S	✓					
a b c d	✓			✗	✓	✗
T_1	✓			✗	✓	✗
e f g	✓			✗		
T_2	✓			✗		
h i	✓			✓		
j k l	✗			✓		
T_3 T_4	✓					

Figure 1.12: ***Differences between GB and CB producibility***

Seeds and targets are S and T circles, respectively. The objective functions are formed by a reactions consuming the ensembles $\{T_1\}$, $\{T_2\}$ and $\{T_3, T_4\}$. Arrows represent reactions. The labels of the reactions $S \mapsto na + b$ and $J \mapsto 2k$ depict their stoichiometry. Crosses indicate that metabolites cannot be produced. Check marks indicate that metabolites can be produced. The compound T_1 can always be produced according to graph-based criteria whereas the variation of the stoichiometric coefficient n can block the production of T_1 according to a balanced-mass stoichiometric framework: Flux Balance Analysis (FBA). By blocking the production of T_1, a variation of n can also block the production of all metabolites downstream. The compound T_2 can be produced according to graph-based criteria whereas the fact that f cannot be accumulated blocks the production of T_2 according to a balanced-mass stoichiometric framework. On the other hand, k remains FBA-producible through the cycle involving j, k and l whereas it is not producible according to our graph-based criteria. T_3 and T_4 are producible by both criteria. Figure and legend adapted from [Prigent et al., 2017]

that the constraint-based producibility is very sensitive to the accuracy of stoichiometric coefficients, contrary the the graph-based one. The production of $\{i, j, k\}$ enlightens the differences between modeling of the initial state and modeling of the steady state: the producibility is not GB-initiated for these three metabolites.

The topological semantics has been compared to the constraint-based one (described in the following subsection) in [Kruse and Ebenhöh, 2008]. Despite some compounds such as the cofactors [Ebenhöh et al., 2006], it is realistic to expect that metabolites are synthesized "de novo" and depend from external inputs. The authors conclude that the computation of the scope, especially when assuming the presence of some cofactors, results to coincide with the prediction of FBA. Thus GB methods are a good alternative to determine producibility as they are **computationally efficient** and have similar results to constraint-based methods. The definition of the scope is compatible with the hypothesis that the cell is constantly growing or reproducing, ie in a non-stationary state, contrary to flux-based methods that rely on steady-state assumptions.

1.3.5 Solving optimization problems for metabolism

The two previous definitions of producibility and their associated problems use each a different solving strategy. GB modeling is a combinatorial problem that does not use numbers contrary to CB modeling, that is solved using linear programming.

Linear Programming As stated by [Maranas and Zomorrodi, 2016], "*mathematical optimization (programming) systematically identifies the best solution out of a set of possible choice with respect to a pre-specified criterion*". An optimization problem is presented with FBA in Definition 1.4 and can be described in a general manner [Maranas and Zomorrodi, 2016] by:

$$\begin{aligned} \text{minimize (or maximize) } f(x) \\ \text{subject to} \\ g(x) \leq a \\ h(x) > b \\ x \in \mathbb{S} \end{aligned}$$

- x is a vector of variables
- the maximization or minimization of $f(x)$ is the objective function (maximize the flux in the objective reaction in FBA)
- $g(x) \leq a$ and $h(x) > b$ are constraints applied to the objective functions. They can be equalities or inequalities.
- $\mathbb{S}$ is the set of feasible values for x, the reals $\mathbb{R}$ for the FBA optimization problem of Definition 1.4.

In linear programming, both the objective and constraint functions are linear [Maranas and Zomorrodi, 2016] and the variables in x are continuous. Such problems can be solved with ad-hoc solvers among which CPLEX [1]. The first algorithm solving such problem without examining all feasible solution is the simplex method introduced by [Dantzig and Orden, 1955].

Combinatorial solving with Answer Set Programming (ASP) GB problems can be efficiently modeled and solved using Answer Set Programming (ASP). ASP is a declarative problem solving approach. A lot of work in this thesis relies on this approach. This paragraph provides a small descrption of ASP, additional information can be obtained in the book written by [Gebser et al., 2012]. The main distinctive feature of ASP is that it is declarative. "*What to solve*" is described rather than "*How to solve it*". ASP enables to express search problems in NP [Gebser et al., 2012] ASP can be compared to Prolog. Main differences lie in the strict separation of logic and control in the former. ASP benefited from Satisfiability Testing (SAT) (Boolean constraints) solving assets, but its expressivity is higher than the one of its boolean counterpart. Moreover, ASP considers all propositions as false unless they are proved otherwise (close world reasoning).

Before the solving step, a grounding step in necessary in ASP in order to convert the model representation made by the user/modeler into a finite propositional format. The set of atoms satisfying the constraints are called answer sets. A widespread ASP suite is Clingo [Gebser et al., 2016b] (that combines the previously separated Gringo for grounding and Clasp for solving). The plain-lined elements of Figure 1.13 present the process of solving a problem with ASP, from the modeling of the problem to the interpretation of solutions. A logic programm in ASP is a set of rules of the following form:

$$\underbrace{a_0}_{head} :- \underbrace{a_1, ..., a_m, \text{ not } a_{m+1}, ..., \text{ not } a_n.}_{body}$$

[1] ibm.com/products/ilog-cplex-optimization-studio accessed on July 19, 2018

$\{a_0, ..., a_n\}$ are atoms, $: -$ can be interpreted as a "if". Hence the rule can be intuitively expressed as: "the head of the rule is true if the body of the rule holds". In particular: a_0. is always true, it is a fact; and $: - a_0$ is always false. The negation is an important element of ASP. An atom that never belongs to the head of a rule (after grounding) cannot be true. Consequently,

$$p.$$

$$: -p,\ notq.$$

has no answer set because the p atom never belongs to the head of a rule, whereas:

$$p.$$

$$q : -p.$$

has $\{p, q\}$ as an answer set.

Applied to metabolic modeling, ASP can express reactions, metabolites and producibility. An example is given with the following:

```
scope(M) :- seed(M).
scope(M) :- reaction(R); product(M,R); scope(N) : reactant(N,R).
```

The first line expresses that seeds belong to the scope. The second one expresses the recursivity of the scope definition: the product of a reaction belongs to the scope only if the reactants of this reaction are themselves in the scope. Therefore, considering the small metabolic network of Figure 1.6 with the following input:

```
seed("A").
seed("B").
reaction("r1").
product("A").
reaction("r2").
product("B").
reaction("r3").
reactant("A","r3").
reactant("B","r3").
product("C","r3”).
reaction("r4").
reactant("C","r4").
product("D","r4").
reaction("r5").
reactant("D","r5").
```

returns the following answer set:

```
scope("A") scope("B") scope("C") scope("D").
```

Hybrid linear and combinatorial programming Hybridization of the two programmings can be performed via ASP using the latest versions of Clingo [Gebser et al., 2016b]. This is achieved with a linear programming (LP) constraint propagator that interfaces between the ASP solver and the LP solver and the use of LP theory that forms a grammar to construct and express the linear constraints in the logic problem [Kaminski et al., 2017] [Janhunen et al.,

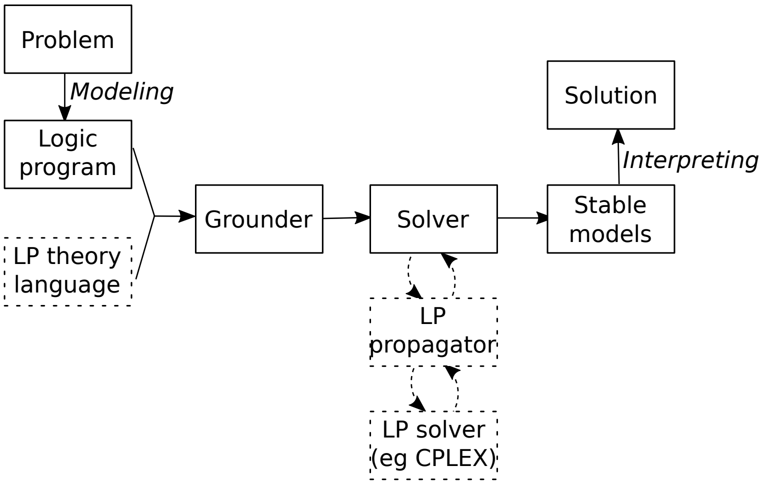


Figure 1.13: ***ASP and hybrid LP-ASP solving of a problem***

Figure adapted from [Gebser et al., 2012]. Plain-lined boxes represent the typical process of ASP solving. A problem is modeled into a logic program that will be grounded ie instantiated in first order variables prior to actual solving. The latter return stable models that can be interpreted by the user. The addition of linear programming (LP) to ASP (dashed lines) involves the use of the LP theory language that enables the expressivity of lienar constraints. Then at the solving phase, calls to the LP solver are made through the existence of a constraint propagator.

2017]. Figure 1.13 presents the process of solving a hybrid problem. Dashed lines describe the elements that distinguish hybrid solving from the regular ASP solving of a problem. The combination of logic programming and linear programming had been already performed using SAT solving in the context of computational biology [Morterol, 2016, Peres et al., 2014].

To sum up

Graph-based and Constraint-based semantics of producibility applied to metabolic models appear complementary for initial state and steady state studies. Their associated solving methods are also complementary which entails the possible combination of both.

1.4 Role of activation semantics in GSM reconstruction

The existence of several semantics of activation to model the functionality of a GSM has a strong impact on its reconstruction. It is for example involved in the answer to the question "when is a GSM reconstruction considered achieved?". A general answer is that it is over when the GSM is functional with respect to a metabolic objective (often the production of biomass) and a semantics (often the constraint-based activation $active^{s}_{G}(S)$ [Definition 1.5]). Of course this objective can be difficult to meet in certain GSM reconstructions, notably for non-model organisms in which cases another semantics or objective can be set.

1.4.1 Metabolic network reconstruction and limitations for non-model organisms

GSM are obtained after a wide reconstruction process that aims to capture all the reactions that are likely to be catalyzed in the organism based on its genomic information. [Thiele and Palsson, 2010] proposed a protocol to build high-quality models based on 96 steps. This protocol relies on an important step of manual curation and refinement that is expected to last from months to a year. It has been used to build some of the most highly curated GSMs like the one of the bacterium *Escherichia coli* for which several versions have been produced and enhanced over the years [Reed et al., 2003, Feist et al., 2007, Orth et al., 2011, Monk et al., 2017]. The first stage of reconstruction in Thiele's protocol is to create a draft version of the model from genomic information. This step is the easiest to be fully automatized.

The difficulty related to building a GSM depends on the organism that is being studied. Models for prokaryotes are easier to reconstruct that model for eukaryotes. In addition, NMOs are more difficult to model due to their phylogenetic distance to other organisms: annotation of genes is more challenging for poorly-studied organisms. Consequently, a draft for a NMO is likely to be smaller than for a model organism. The model refinements stage in Thiele's protocol relies a lot on study of the literature and existing experimental studies. Once again, it is not suitable for NMOs, notably for instance when the model is built for a non-cultivable bacterium.

Below are detailed the main steps in GSM reconstruction and some major tools that are used for this purpose.

Main steps In the last eight years, following the publication of [Thiele and Palsson, 2010]'s protocol, a lot of platforms have been developed to fully automatize the reconstruction of GSMs. They combine several methods that can be classified in two categories: building a draft and refining it until reaching model functionality. The functionality is usually assessed by evaluating the ability of the model to mathematically sustain growth. To that purpose, a biomass reaction [Feist and Palsson, 2010] is generally added in the model and its contents in terms of reactants, products and their associated stoichiometry is customized for the studied organism. As this is the key element to decipher functionality, precision regarding its design is needed [Chan et al., 2017]. A key point is the balance of cofactors in the system, that often requires adjustments and are complex elements of the models [Xu et al., 2017, Xavier et al., 2017]. There exists universal biomass compositions that can be adapted to the species under study [Henry et al., 2010] although published models display biomass functions that can differ considerably from this universal model [Xavier et al., 2017].

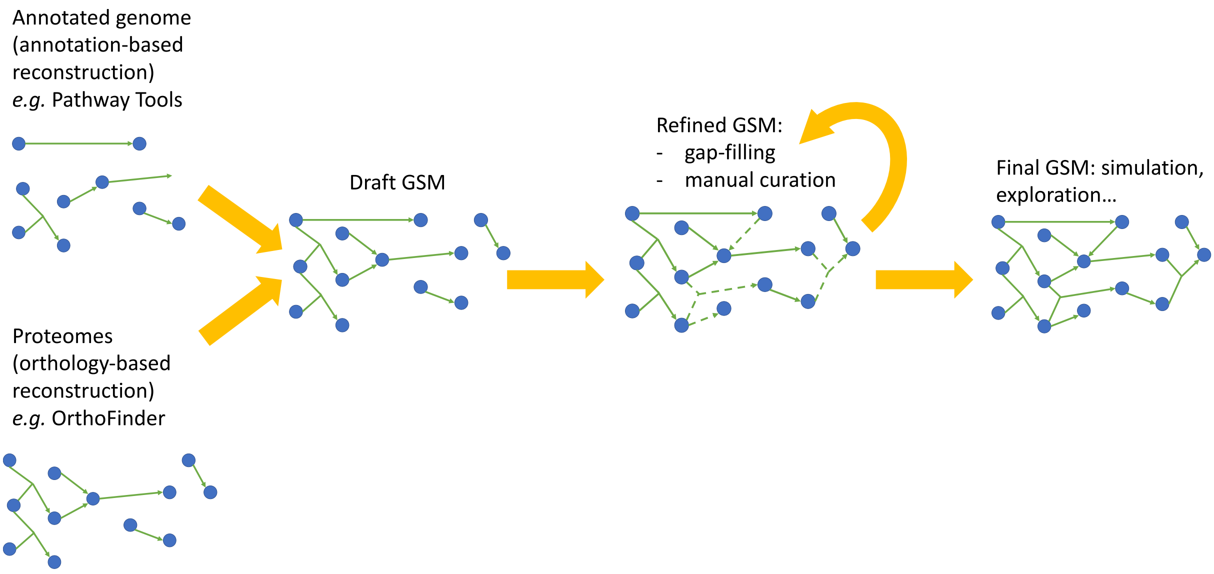


Figure 1.14: ***An example of GSM reconstruction process***

A draft GSM is obtained, for example by merging GSMs created from annotation of the genome or orthology with templates. Such models can be obtained with individual tools (e.g. OrthoFinder for orthology [Emms and Kelly, 2015]) or platforms (e.g. PathwayTools [Karp et al., 2016]). The draft is iteratively refined through gap-filling or manual curation until a satifiable GSM is obtained and can be used for simulations for instance.

Pathway Tools [Karp et al., 2016] provides a whole set of tools and methods to build a model starting from annotation. ModelSeed [Henry et al., 2010, Devoid et al., 2013] is dedicated to the automatic online reconstruction of GSMs, possibly combined with RAST annotation pipeline [Aziz et al., 2008], for microbes and plants. Kbase [Arkin et al., 2016] is an online platform that facilitates sharing and collaboration for the reconstruction of GSMs. Finally, the RAVEN Toolbox runs within MATLAB and provides a complete pipeline dedicated to reconstruction, analysis and visualization of models.

An important fact about these platforms is their dependencies to knowledge repositories. There are several major databases in the field of metabolic modeling that each possess its own identifiers for components of GSMs: reactions, metabolites, pathways and even full models. BiGG [King et al., 2016] is freely available and contains 6,203 metabolites, 17,812 reactions and 85 GSMs. MetaCyc [Caspi et al., 2018], used as a basis for the Pathway Tools software, is academic-free and contains 14,847 metabolites, 14,971 reactions, 2,642 metabolic pathways and 13,076 models called Pathway/Genome Databases (PGDB) (available on subscription). The SEED [Henry et al., 2010] contains 15,734 metabolites and 34,702 reactions. Finally, KEGG [Kanehisa et al., 2016] contains 18,332 metabolites and 10,920 reactions but is not freely available. These major databases are not easily reconcilable but some tools, among which MetaNetX [Moretti et al., 2016] provide promising work in this direction. Concretely, once a model is built on a platform, it is tighly linked to the associated database and it can be difficult in practice to compare it with a model linked to another database. This is a problem when using orthology [Loira et al., 2015, Emms and Kelly, 2015] methods to enhance the GSM of a species with the GSMs of template organisms by aligning their proteomes sequences: the resulting set of reactions to be added to the first model need to be "translated" to its own database.

In addition to the major platforms previously cited, a wide range of methods and bioinformatics tools [Vijayakumar et al., 2017] exist in the field of GSM reconstruction. They can be used during the reconstruction process or as a curation/refinement step and/or after, for analyzing the model [Ebrahim et al., 2013, Schellenberger et al., 2011b, Steffensen et al., 2016].

An important step of the refinement stage in GSM reconstruction processes is the gap-filling. It aims at selecting reactions from a database in order to restore functionality in the model. There are several techniques and their associated tools for this purpose. A shared criterion is to make a minimal number of modifications to the model, hence a minimization of the reactions to be added. A focus on gap-filling will be given in a following subsection.

Figure 1.14 presents a example of GSM reconstruction process. Depending of the available data, the pipeline of reconstruction can vary. However, a general pattern is to create a draft, refine it and eventually obtain a final model.

Flexibility and reproducibility of GSM reconstructions In [Aite et al., 2018], we showed that a large number of published GSMs rely on several platforms, tools and even databases. This means that building a GSM is a flexible process and the method employed for the reconstruction is variable; it depends on the data available, the organism that is studied and the manual curation that needs to be performed. [Heavner and Price, 2015] advocate for transparency reproducibility and quality criteria in GSM reconstruction.

The possible issues raised by the previous observation if of two kinds. First **traceability** needs to be ensured by knowing the steps that led to the addition of objects in the model. Knowing whether a reaction was added thanks to the annotation, orthology step, gap-filling or manual curation can be important for the study of functions of interest. The second aspect is **reproducibility**. Few models provide the full and detailed reconstruction process with the chaining of tools and inputs used. More importantly, traces of manual curation - addition, removal or modification of reactions in the model - are absent or poorly described [Aite et al., 2018]. This is an issue if someone wants to apply the exact same reconstruction process to a taxonomically close organism for example.

To sum up

> The reconstruction of GSMs is a highly flexible process that is adapted to the organism under study. Reconstruction of models can be done fully automatically following one or several semantics of activation, but refinements and revisions made to the GSM are important to produce good quality models. In any case, the flexibility in the process has to be supported by strong efforts on traceability and reproducibility.

1.4.2 Specificity of the gap-filling step

Gap-filling is an important step in metabolic model reconstruction. Gaps exist in GSMs for several reasons: limitations of the reconstruction tools, incomplete knowledge that leads to non-annotated genes, biased knowledge that led to add a false-positive reaction creating an upstream gap etc. These gaps can be identified in the whole network [Satish Kumar et al., 2007, Thiele et al., 2014] or in specific pathways to unblock metabolites of interest [Prigent et al., 2017]. The second step is the proposition of reactions of interest by a dedicated algorithm. The ultimate stage is to assign genes to the suggested reactions [Pan and Reed, 2018, Orth and Palsson, 2010, Plata et al., 2012, Chitale et al., 2016]. Here we will discuss the first and second step and present a definition for the gap-filling problem and algorithms for its solving.

Definition of a gap-filling problem Metabolic network completion, or gap-filling, can be about ensuring that a set of target reactions (reaction r_5 in Fig. 3.1) follows a certain definition of activation (eg graph-based, constraint-based, relaxed constraint-based etc.) from the seed compounds in S by possibly extending the metabolic network with reactions from a reference network It can also be focused on ensuring that metabolites become reachable but these two definitions are non incompatible. The definition can additionally be extended to the activation of all reactions of the model, respectively the reachability of all metabolites.

Definition 1.7 A definition of the gap-filling problem *Given a metabolic network $G = (R \cup M, E, s)$, a set $S \subseteq M$ of seed compounds such that $S_b(G) \subseteq S$, a set $R_T \subseteq R$ of target reactions, and a reference network $(R' \cup M', E', s')$, the* metabolic network completion problem *is to find a set $R'' \subseteq R' \setminus R$ of reactions of minimal size such that $R_T \subseteq active_{G''}(S)$ where*

$$
\begin{aligned}
G'' &= ((R \cup R'') \cup (M \cup M''), E \cup E'', s'')\ ,\\
M'' &= \{m \in M' \mid r \in R'', m \in reactants(r) \cup products(r)\}\ ,\\
E'' &= E' \cap ((M'' \times R'') \cup (R'' \times M'')), \textit{ and}\\
s'' &= s \cup s'.
\end{aligned}
$$

We call R'' a *completion* of $(R \cup M, E, s)$ from $(R' \cup M', E', s')$ with respect to S and R_T. The existence of several concepts of activation, among which the graph-based, the constraint-based and the relaxed constraint-based, allows different biological paradigms to be captured. Indeed, the gap-filling step is strongly dependent to the formalism employed for modeling producibility in the GSM. As the aim of this step is to restore functionality by adding reactions into the model, possible variations happen into the solutions based on the producibility definition harboured by the gap-filling method.

A landscape of gap-filling methods

We introduce here four gap-filling tools (described in [Prigent et al., 2017]). Figure 1.15 describes the algorithms used by the four tools and Table 1.1 sums up their main characteristics [Prigent et al., 2017].

Meneco [Prigent et al., 2017] is a topological parsimonious gap-filling tool relying on the graph-based producibility of metabolites and the **graph-based activation** of reactions $active^t_G(S)$ in a metabolic network G initiated by seeds S (Def. 1.3). It proposes a minimal number of reactions to be added to the draft GSM such that it can produce metabolic targets starting from seeds metabolites [Collet et al., 2013, Schaub and Thiele, 2009a]. These metabolic targets can be any compounds of the GSM. In particular, having in mind the definitions of activated reactions, the reactants of the objective reaction can be set-up as targets for applying Meneco. Meneco uses ASP to solve the combinatorial gap-filling problem. In particular, ASP is

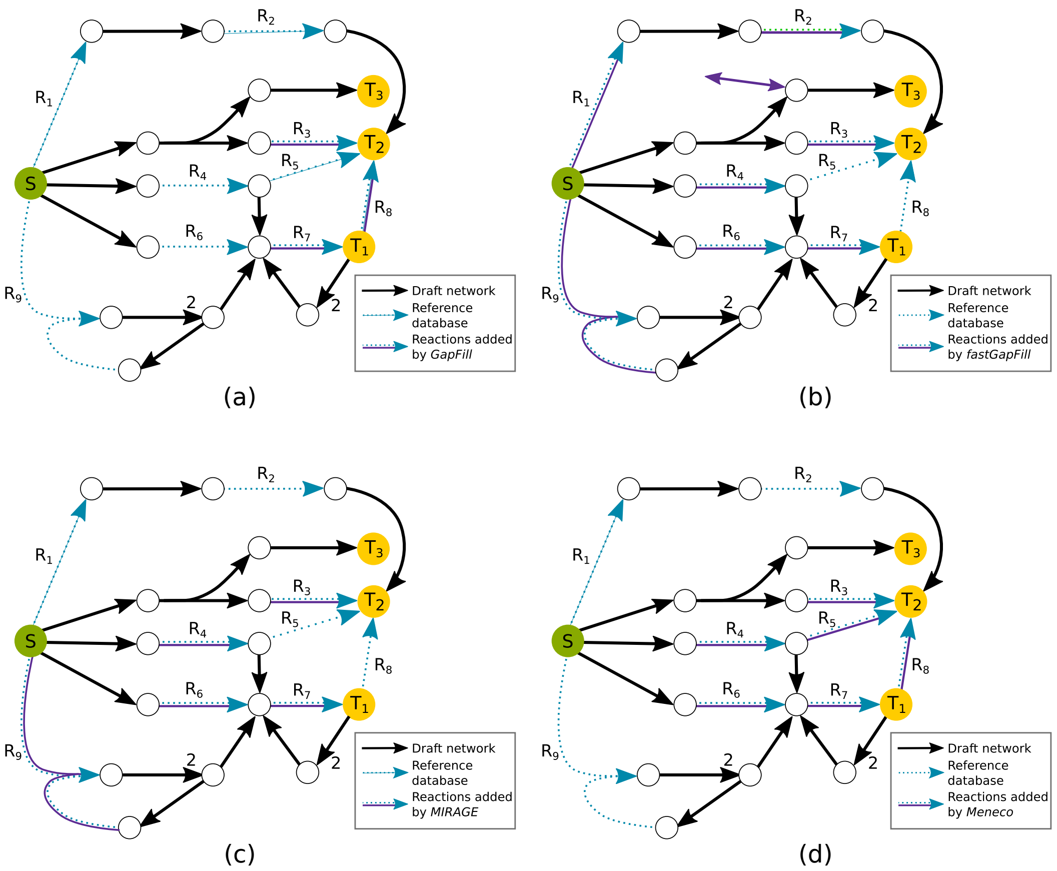

R$_1$
R$_2$
R$_3$
R$_4$
R$_5$
R$_6$
R$_7$
R$_8$
R$_9$
S
T$_1$
T$_2$
T$_3$
2
Draft network
Reference database
Reactions added by GapFill
Reactions added by fastGapFill
Reactions added by MIRAGE
Reactions added by Meneco
(a)
(b)
(c)
(d)

Figure 1.15: ***Gap-filling of metabolic networks with different heuristics***

The reactions of the initial network are depicted as black arrows. Seeds (e.g. growth medium) and targets are S and T circles, respectively. The labels on the arrows depict the stoichiometry of the reactions. Blue dotted arrows represent reactions that can be added to the network (reference database). The purple arrows represent reactions proposed by different gap-filling tools. GapFill *(a) reported two reactions as a minimal completion and two different combinations to produce biomass from T_1, T_2 and T_3, $\{R_3, R_7\}$ and $\{R_7, R_8\}$.* fastGapFill *(b) reports one unique set of seven reactions to unblock all reactions of the example: $\{R_1, R_2, R_3, R_4, R_6, R_7, R_9\}$. It also add an import/export reaction for the reactant of the reaction producing T_3. In additions, 100 runs of* MIRAGE *(c) without scoring of reactions reported the following set of five reactions: $\{R_3, R_4, R_6, R_7, R_9\}$. Finally,* Meneco *(d) reported that three reactions are needed to restore the topological producibility of the three targets, with five different combinations. Therefore, the output of* Meneco *is the set of six reactions $\{R_3, R_4, R_5, R_6, R_7, R_8\}$. Figure and legend adapted from [Prigent et al., 2017]*

an asset here as it enables to provide a full exploration of the solution space: Meneco can enumerate all optimal gap-filling solutions. As enumeration can be computationally demanding in some cases, a solution is to use the intersection (reactions occurring in all optimal solutions) and union (reactions occurring in at least one optimal solution), that are both efficiently computed through the use of adequate ASP solving heuristics. The inputs to the GB gap-filling problem as solved by Meneco are the following:

- a draft GSM $G_{draft} = (R_{draft} \cup M, E)$;
- a set $M_{target} \subset M$ of target metabolites that are expected to be producible by the organism of interest;
- a set $M_{seed} \subset M$ of seed nutrients found in the growth medium;
- a database of metabolic reactions available to fill the network that can be considered as

a metabolic model itself $G_{database} = (R_{database} \cup M, E)$.

The Meneco gap-filling problem is stated as follows:

$$\text{Minimize } size(R_{fill}) \text{ s.t } \begin{cases} R_{fill} \subset R_{database} \\ M_{target} \cap scope_{R_{fill} \cup R_{draft}}(M_{seed}) \text{ is maximal.} \end{cases}$$

The scope is the set of metabolites reachable from the seeds, as explained in Definition 1.2. Applied to the example depicted in Figure 1.15, the five different sets of reactions restoring the topology-based producibility of T_1, T_2 and T_3 are: $\{R_3, R_4, R_7\}$, $\{R_3, R_6, R_7\}$, $\{R_4, R_5, R_7\}$, $\{R_6, R_7, R_8\}$, $\{R_4, R_7, R_8\}$. Meneco missed the reactions R_1 and R_2, because of the parsimonious criteria they used. On the contrary, Meneco reported the reactions R_5 and R_8 because it checked that T_2 could be simply produced by adding R_8 as soon as T_1 was produced. Finally due to the cycle, Meneco missed the reaction R_9. This could have been circumvented by adding one of the metabolites of this cycle to the seeds, like explained in the description of initiation seeds (see 1.3.2)

GapFill uses a parsimonious bottom-up strategies [Satish Kumar et al., 2007, Benedict et al., 2014], which enriches the draft metabolic network until the targeted properties are satisfied. More precisely, GapFill detects a minimal number of reactions from the reference database to enable the synthesis of the targeted compounds according to a **relaxed constraint-based** activation $active^r_G(S)$ (presented in Definition 1.6). Consequently, GapFill allows the accumulation of internal compounds in the model. Applied to the toy example in Figure 1.15, GapFill reports a minimum of two reactions to enable the biomass synthesis. There exist two alternative sets of two reactions that do so: $\{R_3, R_7\}$ and $\{R_7, R_8\}$. The union of reactions does not contain the reactions R_1 and R_2 since the parsimonious assumption omits alternative long pathways. The reaction R_7 is used by GapFill to fill the cycle it is part of and to produce T_1. Then GapFill uses either the reaction R_8 or the reaction R_3 to produce T_2 since it considers T_3 as already producible despite a possible accumulation of the reactant of R_3. One limitation of this approach is that it reports a bounded (parameterized) number of solutions to the problem, although there is no a priori estimation of the number of iterations needed to solve the problem [Prigent et al., 2017].

fastGapFill [Thiele et al., 2014] is an extension of the GapFill approach combined with the Fastcore algorithm, which eliminates the focus on the biomass production by identifying a single set of reactions which unblocks all reactions within the draft metabolic network. The price to pay is to allow import and export fluxes, especially to resolve issues related to the activation of isolated reactions in the GEM. This constitutes a heuristics to solve the gap-filling problem with the **constraint-based** definition of activation $active^s_G(S)$ (presented in Definition 1.5). When applied to our toy example, fastGapFill reported seven reactions from the reference database and one import/export reaction of the reactant of the reaction producing T_3 that was not present in the database to be added to the model. The reactions R_1 and R_2 are introduced to produce T_2 in the absence of R_3. Since no accumulation of the reactant of R_3 is possible, fastGapFill adds an import/export reaction to enable the production of T_3 without using the previous reactions of this pathway. The reactions R_4, R_6, and R_9 are equivalent to produce T_1 and were all chosen in order to unblock all fluxes going through all reactions present in the draft model (including the vertical reaction going from the product of R_4 to the reactant of R_7). Finally, R_7 is added to enable the production of T_1.

MIRAGE In contrast to parsimonious approaches, top-down approaches start from all available information and remove reactions without added-value to the solution of the problem [Pharkya et al., 2004, Reed et al., 2006, Christian et al., 2009]. Among them, MIRAGE [Vitkin and Shlomi, 2012] aims at relaxing the minimality condition over the number of added reactions by identifying all subset minimal sets of reactions which enable the **constraint-based activation** $active^{s}_{G}(S)$ (presented in Definition 1.5) of the biomass reaction. A subset-minimal reaction set might include a higher number of reactions but it is minimal in the sense that it loses its capability to restore biomass functionality as soon as any reaction is removed from the set. As the number of such sets of reactions increases dramatically with the size of the reference database, MIRAGE samples the space of solutions by randomly iterating the search algorithm. In our toy example, the algorithm was iterated 100 times without applying any *a priori* scores to reactions of the database. Four different sets of reactions were obtained {R_3, R_7}, {R_3, R_6, R_7}, {R_3, R_4, R_7} and {R_3, R_7, R_9}. The reaction R_3 is mandatory to produce T_3 and it also produces T_2. Therefore, no other reaction producing T_2 is necessary, so that R_1, R_2, R_5 and R_8 were never used. Given the cycle, the reaction R_7 would be the minimal completion to produce T_1 from a constraint-based activation point of view, but the subset-minimality criterion also enables the MIRAGE algorithm to find the reactions R_4, R_6 and R_7.

In the first chapter of Part 1 of this thesis, I will present the validation of Meneco with respect to the other gap-filling methods presented above.

To sum up

The definition of activation together with the heuristics used to solve the gap-filling problem highly impact the four reconstruction procedures that all report different solutions on a small case-study.

Gap-filling non-model organisms The drawback of CB gap-filling is that its applicability is dependent to the balance of the whole system in terms of stoichiometry. An accumulated metabolite can alter the evaluated producibility of another metabolite despite the existence of adequate reactions. This is particularly challenging for NMOs. Their GSMs are not necessarily well-balanced and applying automatic methods to balance them is complex given the poor knowledge available to built their objective functions. In this case, using topological methods to assess the activation of reactions and the producibility of metabolic targets is an interesting alternative.

1.4.3 Genome-scale metabolic model reconstruction in a microbiota context

The beginning of GSM reconstruction for members of microbiota is the same as for classical individual organisms. As long as reactions are supported by genetic information they can undoubtly be added to the network. Difficulty may arise during the gap-filling step. Indeed, the risk at this step is to add orphan reactions, that miss associated genes [van der Ark et al., 2017]. This can happen as a side effect of overfitting the model: activate dead-end metabolites to prevent accumulation for instance [Pan and Reed, 2018].

This occurs because most GSMs are refined and curated under the hypothesis of self sustainability of the considered organism, it can live by itself thus its GSM has to sustain biomass production. Yet, this is not always experimentally validated for NMOs for which culture in

Table 1.1: *Characteristics of gap-filling methods (extracted from [Prigent et al., 2017]). Gap-filling methods mainly differ with respect to the set of modifications to the system they enable (approximating the criteria of producibility, changing the reversibility of reactions, modification of import/export reactions). Therefore, they can be classified according to (i) the set of compounds whose producibility should be restored; (ii) the criteria they optimize and (iii) the number of solutions set they return.*

	Which aim?	Which problem is solved?	Which exploration of the search space?
GapFill	The *stoichiometry-based* production of a *single target* is enabled by adding a *minimal number* of reactions from the reference database.	The problem is modeled by a Mixed Integer Linear Programming (MILP) optimization problem which forces the production of target fluxes, encoded in a GAMS program.	The algorithm reports a *bounded (parameterized) number* of solutions to the gapfilling problem, ordering them by the number of reactions they contain.
Meneco	The *graph-based* simultaneous production of a set of *multiple targets* is enabled by adding a *minimal number* of reactions from the reference database.	The problem is approximated by a combinatorial optimization problem describing topological constraints for the production of a metabolite and solved with Answer Set Programming technologies.	The algorithm reports an *exhaustive enumeration* of all *solutions of minimal size* for the complete set of targets. It also reports the global solution set consisting of all reactions appearing in at least one solution.
fastGapFill	All reactions from the draft model are *unblocked* by selecting a *minimal number* of *reactions from the reference database* or *import/export fluxes* for internal metabolites.	The problem is solved by computing a near-minimal set of reactions that need to be added to the draft metabolic network to render it flux consistent (FastCore algorithm). The search is modeled by MILP optimization problems solved with Cplex.	The algorithm reports a *single solution*. The network, both enriched with additional reactions from the reference database and modified according to novel import fluxes, has no blocked reactions with respect to its core set.
Mirage	The flux of a set of *multiple reactions* (including biomass) is enabled by adding a set of reactions from the reference database. The algorithm favors reactions whose presence is supported by additional data, when available.	The top-down algorithm randomly identifies a set of reactions from the reference database in which all reactions have a non-zero flux. An iterative procedure selects reactions to be removed until the model is no more functional. A ranking of reactions according to their impact of flux distributions is obtained by applying the procedure a parameterized number of times.	The algorithm reports a *single set of reactions* which enables the flux production of all target fluxes. The model is no longer functional when removing any of the reported reaction (*subset minimality*). As the algorithm is *not deterministic*, the reported solution may change at each application of the algorithm.

lab conditions can be difficult or impossible. In the individual growth hypothesis, if gaps remained in the GSMs, it made sense to add reactions and if no associated genes were found, the reason was probably that a gene was missed by experimenters in the genome. This can be an erroneous hypothesis given the fact that some species cannot display a normal physiology without the presence of their microbiota and that the precise dependencies are not easy to be identified and replaced by adequate supplementations in the growth medium [Tapia et al., 2016]. In some cases, biological experimentations can provide leads to identify these interactions [Amin et al., 2015].

The study of NMOs, and particularly in communities, in practice imposes to deal with incomplete data and incomplete GSMs. This entails that some methods become less adapted to this range of studies, like FBA-based methods [Johns et al., 2016]. It is possible that the non functional metabolic activities in a GSM depend on metabolic interactions with other organisms within a community. A solution can be for instance to remove from the networks all reactions that are not backed by an identified gene and work with non gap-filled models. If metabolic dependencies in interactions can be precisely identified, another solution is to consider as metabolic inputs (seeds), the concerned metabolites; however this is difficult to assess when organisms cannot be grown experimentally. Gap-filling can also be performed among the metabolic models of the microbiota of the studied organisms, leaving to experimenters the task to validate the exchanges in practice.

To sum up

Gap-filling is tightly related to the study of microbiotas' NMOs from the metabolic point of view. It can constitute from one hand a drawback in reconstructing such GSMs by forming a risk of overfitting without gene support. On the other hand, it can be exploited in the perpective of digging microbiota's metabolic capabilities to explore putative metabolic dependencies between species.

Conclusion

Non-model organisms and microbiotas. Techniques of sequencing together with improvements in handling large datasets make it possible to study and attempt to elucidate the physiology of the wide range of organisms. Among them we find species that are qualified as non-model (NMOs) for which the level of knowledge is lower than for the model counterparts. These species cannot be given the same amount of attention than model organisms for several reasons. The high-throughput generation of data makes it necessary to automatically exploit the information whereas decades ago, integration, curation and refinement were done by experts, often "manually". In addition, not all NMOs can be cultured in the lab for experimental testing and validation of hypothesis. Thus modeling of their physiology sometimes rely only on sequencing in natural conditions. This occurs frequently in the context of studying microbiotas. Organisms do not live isolated to each other in nature and interaction can occur at many scales, including the metabolic one that is at the center of this thesis.

Studying metabolic interactions. The study of interactions, especially at the metabolic scale, in microbiotas or communities of organisms ranges from the definition of metabolic scores based on complementarity between genome-scale metabolic networks (Genome-Scale Models (GSMs)) to dynamic modeling over time. As expected, the more precise the community modeling is, the more data and knowledge about the protagonists is needed. This is a challenge as experimentation-based data is difficult to acquire for NMOs. Yet, a lot of species found in microbiotas belong to this category. This makes the study of identifying metabolic interactions in microbiotas a still open challenge. Selecting communities of interest in microbiotas is of high interest for synthetic microbiol consortia with industrial applications, or simply for evaluating metabolic dependencies towards a specific objective in the lab. This field of research suffers from the impossibility to compute precise behaviours at the scale of microbiotas, thus necessiting to lower constraints and simplify the functionality of communities during selection.

Choice of functionality semantics in metabolism. We presented two complementary semantics for defining producibility of metabolites or activation of reactions in metabolic networks. We chose to derive a generic concept of reaction activation to graph-based and constraint-based formalisms. The first relies on external entries (seeds) to the model to recursively computes reachable metabolites in GSMs. It can be a good solution to model non-stationary states in the cell. It can also support the conjecture that not all metabolites need a *ab initio* production by establishing that some cofactors can be presumably producible. Constraint-based semantics on the other hand is particularly adapted to model stationary states. Activation of reactions in a GSM can be assessed using Flux Balance Analysis (FBA). More generally, flux-based methods can provide information about the roles of reactions with respect to a particular objective, that is often for a GSM to produce biomass. Constraint-based and flux-based methods are not incompatible from the theoretical point of view. From a technical aspect, we show that their respective programming paradigms can be conciliated, notably with the use of linear constraints propagators together with Answer Set Programming (ASP) solving strategies.

Role of gap-filling in genome-scale models reconstruction. Activation semantics are also involved in the reconstruction processes of metabolic networks. We presented generalities about GSM reconstruction. Notably, the ability to use GSMs for the study of microbiotas relies on the possibility to obtain quality GSMs for such purpose. We showed that the process of reconstructing high-quality GSMs still necessitates to combine platforms and individual tools, as well as sometimes databases of metabolic knowledge, with manual refinements made by experts. This raises issues for reproducibility and traceability if reconstructions have to be repeated or extended to other organisms. In particular, gap-filling is a recurrent step performed in the process of GSM reconstruction. It aims to ensure the feasability of a metabolic objective (biomass production generally) by the GSM by adding reactions to the network. We showed that a landscape of methods coexists for this purpose and that not one definition of gap-filling exists, but several ones, depending on the exact problem solved and the chosen semantics of activation for functionality. In the context of NMOs and microbiotas, the risk of gap-filling is to add reactions without gene supports for organisms that possiby cannot sustain the metabolic objective by their own. On the other hand, gap-filling can be viewed as an interesting tool to search metabolic interactions within communities by replacing metabolic knowledge databases by the metabolic networks of symbionts.

part I

Advances in gap-filling for non-model organisms

In this first part I will present the work I pursued on gap-filling for metabolic models. As presented in the previous chapters, gap-filling is an important step in reconstructing Genome-Scale Models (GSMs). It occurs after the automatic draft generation. The general purpose of gap-filling is to add reactions to the model such that it reaches functionality. Constraints for the addition of reactions are of two kinds. The model is wanted to be minimally altered, that is to say the number of reactions to be added is expected to be minimal. Secondly, the reactions are expected to be meaningful with respect to the biology of the considered organism.

These constraints are challenging when it comes to study a non-model organism: literature related to it is sparse, annotation is less accurate and thus validating the reactions to be added is delicate. In the meantime, the design of the biomass reaction and its adequate associated stoichiometries, that is the main objective to be optimized through gap-filling, is troublesome, particularly for eukaryotes. Even for bacteria, the challenge remains as these functions are often derived from the ones of model organisms (e.g. *E. coli*) that are sometimes taxonomically distant from the considered species. All these limitations entail the need for a certain cautiousness and flexibility with respect to the gap-filling step and and to give the user the opportunity to get involved in the process. The latter can occur through the better sampling of the solution space to catch all solutions or elements of the solution space (union, intersection). The former flexibility criterion can be met by adapting the objective of gap-filling through the use of target metabolites and graph-based definition of producibility rather than reactions with precise stoichiometry and the associated constraint-based definition of producibility. Here we present two gap-filling algorithms and their testing, with a strong emphasis on application to non-model organisms (NMOs).

Chapters 2 and 3 present works that were respectively published in [Prigent et al., 2017] and [Frioux et al., 2017]. Chapter 2 focuses on the topological or graph-based gap-filling method Meneco and particularly to its validation and comparison to constraint-based methods. Chapter 3 presents the extension of the method to a hybrid formalism that reconciles graph-based and constraint-based gap-filling theories. Finally, chapter 4 presents applications of these works, some of them being published in [Aite et al., 2018] and [Prigent et al., 2017], through the reconstruction of GSMs for NMOs, the use of hybrid gap-filling in the final stages of reconstructing GSMs and the indirect use of the gap-filling theory to find metabolic complementarities between members of an holobiont.

Chapter 2

Flexibility and accuracy of graph-based gap-filling

In this chapter I present the validation and testing of Meneco, a graph-based gap-filling method. The aim of Meneco is to restore the producibility of targets in a Genome-Scale Model (GSM) starting from available metabolites called seeds. These targets can be any set of metabolites of the model such that the algorithm applies to GSMs regardless their state of reconstruction. It is particularly fitted to suit non-model organisms (NMOs) for which a biomass reaction may not be available. It can also be applied to restore the producibility of biomass reactions by setting as targets the reactants of this reaction. Meneco relies on logic programming for the selection of minimal-size solutions within a possibly large database of reactions. Indeed, handling a gap-filling step for NMOs entails to be able to support large generic databases that contain metabolic reactions for a wide range of organisms. It also entails to provide solutions of high quality that can be analyzed and refined by experts *a posteriori*. In this work, Meneco is compared to several existing tools of the literature: fastGapFill, GapFill, and MIRAGE, three constraint-based gap-filling methods, to assess the potentiality of graph-based methods with respect to these criteria. The benchmark is established on a well known model organism, the bacterium *Escherichia coli*, for the sake of method validation. This work has been published in *PLOS Computational Biology* [Prigent et al., 2017].

Parts of this chapter's text and figures were extracted from the paper Sylvain Prigent, I, and others coauthored in ***PLOS Computational Biology*** entitled "*Meneco, a topology-based gap-filling tool applicable to degraded genome-wide metabolic networks*" [Prigent et al., 2017].

Meneco is a gap-filling tool that aims to select a minimal number of reactions, picked from a database, to restore the producibility of a set of target metabolites starting from seed ones, in a genome-scale model (GSM). It relies on Answer Set Programming (ASP) to solve this combinatorial problem. Contrary to flux-based methods, Meneco uses a graph-based definition of producibility, based on the scope of a set of seeds that usually are metabolites of the growth medium (for more details, see Subsection 1.4.2). It was applied to the reconstruction of EctoGEM, the GSM of *Ectocarpus siliculosus* [Prigent et al., 2014, Collet et al., 2013]. The goal of the following work is to evaluate the performances of Meneco with respect to constraint-based gap-filling methods. In a first section we compare the accuracy of two parsimonious gap-filling methods. Afterwards, we demonstrate the interest of parsimonious gap-filling among the wide range of methods with respect to scaling to large databases.

2.1 A benchmark of 10,800 gap-filling cases to compare GSM completion methods

2.1.1 Estimation of gaps in automatically-reconstructed draft GSMs

BioCyc [Caspi et al., 2014, Caspi et al., 2016] is a wide database of thousands of GSMs. The repository is split in three "*Tiers*" that each received different amounts of curation. *Tier* 1 database contains a few GSMs for model organisms (6 species and the general MetaCyc) that received at least the equivalent of one year of literature-based analysis for curation. *Tier* 2 GSMs, 42 models in version 22.0, underwent an automatic reconstruction but also received some manual curation to remove false positives reactions. Finally, *Tier* 3 contains several thousands (13,004 GSMs today in version 22.0, mostly non-model organisms [NMOs]) of models that were automatically built and were not validated by scientists.

When comparing the number of reactions in the BioCyc repository version 19.5, we noticed that the 7,296 automatically reconstructed bacterial networks (Tier 3) contained on average 8% fewer reactions than the 27 curated bacterial metabolic networks contained in the manually curated repositories (Tier 1 & Tier 2). This can be seen as a hint onto the average effect of refinements, review and manual curation of automatically-obtained GSMs. ***This tends to place gap-filling and manual curation effects - the following steps to automatic reconstruction - as minor with respect to the size of the model in terms of reactions, yet crucial for the quality and functionality of the model.***

2.1.2 Creation of the benchmark based on FVA properties

We benchmarked gap-filling tools using several versions of *E. coli* MG1655's GSM, from the Bigg database [King et al., 2016]:

- $iJR904$ [Reed et al., 2003]

- $iAF1260$ [Feist et al., 2007]
- $iJO1366$ [Orth et al., 2011].

Based on the observation that refinements to model seem to add a restricted amount of reactions, we generated a large-scale benchmark of 3,600 degraded GSMs for each GSM version as follows. Ninety biomass reactions for the $iJR904$ *E. coli* GSM [Reed et al., 2003] were randomly generated by altering the initial biomass reaction of the model. The reactants of these biomass reactions will constitute the targets of Meneco as the tool requires a set of metabolites rather than a reaction for gap-filling. Then, forty GSMs per model were obtained by removing 10%, 20%, 30%, or 40% of the GSM. Degradation occurred through all types of pathways, including the central ones, such as the Tri-Carboxylic Acid cycle (TCA) for instance. 105 networks out of 120 (88%) had a degraded TCA cycle, with the number of missing reactions ranging from 1 to 7.

Reactions of each *E. coli* GSM were classified according to their functionality, computed using Flux Variability Analysis (FVA) with respect to biomass production. As a reminder, a reaction r is defined as essential when a non-zero biomass production implies a non-zero flux through this reaction. Otherwise, if a pathway can produce the biomass without always involving reaction r, then the latter is considered alternative. Finally, if the flux through reaction r is always zero, this reaction is classified as blocked (for more details about FVA, refer to Subsection 1.4.2). It is important to note that the classification of reactions into essential, alternative and blocked is related to and highly dependent on the corresponding biomass reaction. Importantly, the degradation of *E. coli* networks carried out in our benchmarks was such that essential, alternative, and blocked reactions, with respect to each of the 90 different biomass functions, were uniformly removed from the initial network (e.g. see Table 2.1 for $iJR904$). Together, we obtained three benchmarks of 3,600 (40*90) gap-filling test cases. No GSM was capable of producing the biomasses with respect to graph-based and constraint-based definition of producibility (cf. Def. 1.3 and 1.5). The total number of gap-filling experiments is thus 10,800 for the combination of the three models. Figure 2.1 sums up the degradation pipeline.

2.1.3 Two types of completion experiments

In order to complete these degraded metabolic networks, we distinguished two use-cases.

i) As a first benchmark experiment, we gap-filled the models using **the initial *E. coli* models as the repair database** to compare two parsimonious tools: Meneco and GapFill. This enables to compare the functionality of models gap-filled by the two tools, notably by checking the FVA status of reactions prior to and after gap-filling.

ii) In the second use-case, we selected **MetaCyc as a reference database** (version 18.5) [Caspi et al., 2014], motivated by both its wide content (eukaryotic and prokaryotic reactions) and its accessibility (freely downloadable). This benchmark represents the use-case where the networks that must be gap-filled depict organisms that cannot be naturally associated to a precise phylogenetic taxa for which databases of reactions exist, requiring to explore all possible metabolic reactions to fill the network. The identifiers of $iJR904$ were mapped to identifiers of the MetaCyc database and both files were merged to create a complete reference database which contains MetaCyc and all the reactions removed from the original GSM. The addition of the original GSM of *E. coli* to the database ensures that a solution to the problem exists, which is important for the validity of the benchmark.

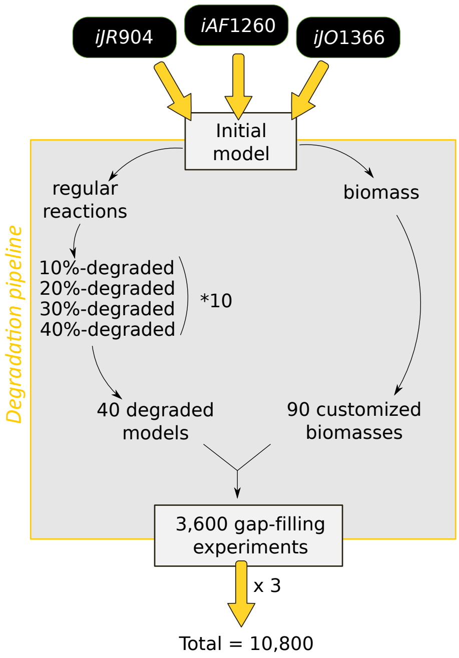


Figure 2.1: ***Degradation pipeline of*** **E. coli** ***GSMs***

For each E. coli model*, between 10 and 40% of reactions are removed; by respecting proportions of essential, alternative and blocked according to the FVA formalism. 10 models of each degradation are created. In parallel, 90 biomass reactions are built by altering the initial one (removal some of its reactants). The combination of each degraded network for each original E. coli GSM and each biomass ends up in 10,800 gap-filling experiments.*

The purpose of this two experiments was to compare Meneco to its constraint-based counterparts: fastGapFill, GapFill and MIRAGE. GapFill uses a relaxed constraint-based semantics that enables accumulation of metabolites (relaxed constraint-based activation of reaction). This enables to solve the gap-filling in practice but entails a risk not to restore biomass producibility under constraint-based activation. fastGapFill aims to restore the possibility to have flux in every blocked reaction of the model, not only a targeted reaction. Finally, MIRAGE is a top-down approach that relaxes the minimality condition over the number of added reactions by identifying all subset minimal sets of reactions which enable the functionality of the biomass reaction. For more details on the three methods, refer to subsection 1.4.2 and Table 1.1. As most methods of the benchmark are based on constraint-based semantics and that is is the common criterion to validate GSM, we decided to evaluate the success of the gap-filling step by assessing flux in the biomass reaction using FBA (constraint-based activation of the biomass reaction). Consequently, further references to successful completions and/or producibility in this chapter refer to the constraint-based definition of producibility (Definition 1.5).

2.2 Functionality analysis of two parsimonious gap-filling methods

To test the relevance of the reactions added by Meneco, we set-up a benchmark for comparison with another parsimonious method GapFill. The purpose is to estimate the impact of the choice of the producibility criterion (scope vs flux) over the parsimonious gap-filling procedures. Meneco and GapFill were applied to complete each of the 10,800 degraded GSMs by using the

Table 2.1: *Characteristics of the 3,600 networks in the benchmark of E. coli i*JR904 *degraded GSMs. Between 10% and 40% of reactions were removed from three E. coli reference networks in order to block the constraint-based production of 40 different biomass functions. The distribution of the removed reactions is presented according to the degradation rate of the model. The initial model is composed of 1,075 reactions and 1,800 metabolites.*

***iJR*904**	average	10%	20%	30%	40%
Removed reactions					
min	101	101	203	307	414
max	446	117	226	343	446
mean	269	109	216	325	430
Essential reactions removed					
min	0	0	0	0	0
max	21	7	13	18	21
mean	5.18	2.19	4.67	6.47	7.83
Blocked reactions removed					
min	15	15	32	59	75
max	112	28	61	81	112
mean	55	22	44	66	89
Alternative reactions removed					
min	72	72	144	227	304
max	349	92	177	276	349
mean	208	84	166	252	332

networks prior to degradation as a reference database. The main motivation for changing the reference database was to be able to analyze how the classification of reactions into essential, alternative and blocked with respect to biomass production evolved along with the completion process.

2.2.1 Parsimonious tools Meneco and GapFill propose comparable sizes of solutions

For each combination of a degraded *E. coli* network and a biomass composition, Meneco and GapFill were applied to complete the network in order to produce all compounds included in the biomass. Note that, in particular, GapFill was not used to produce all blocked metabolites present in the network, as it could have been done using the GapFind algorithm that is provided with GapFill, but only those involved in the biomass of the network.

The GapFill benchmark was run as an assembly of single target completions since it was impossible to be run on a global biomass reaction while being exhaustive. This is due to the combinatorial characteristics of the problem: the number of solutions when filling the networks based on a complete biomass was too high (mandatory bounding of the number of solutions with GapFill) and so was the computation time. Hence, we used as output for GapFill the union of the completion sets (maximal 30) for each individual target. We noticed

Table 2.2: *Characteristics of the 10,800 networks in the benchmark of E. coli degraded GEMs and their completion with Meneco and GapFill. Between 10% and 40% of reactions were removed from three E. coli reference networks in order to block the constraint-based production of 40 different biomass functions. For each degraded network, both the Meneco and the GapFill tools were used to restore the producibility of the biomass. The same table is available for the the 10 percent degraded models*

Reference network	***iJR*904**	***iAF*1260**	***iJO*1366**
Characteristics			
reactions	1075	2383	2582
compounds	1800	1967	2129
Reactions added by Meneco			
min	0	4	0
max	64	90	90
mean	23	35	30
Reactions added by GapFill			
min	0	0	0
max	63	126	107
mean	23	48	37

that only 68 completion experiments reached the limit of 30 completion sets for individual targets among the 10,800 degraded networks. The results of these gap-filling procedures (Table 2.2 for the complete benchmark, Table 2.3 for 10%-degradation rates) show that although a large number of reactions was removed from the network, both Meneco and GapFill returned solution sets of relatively small size (from 0 to 64 reactions for the $iJR904$ network, from 0 to 126 reactions for $iAF1260$, from 0 to 107 reactions for $iJO1366$). On average, Meneco or GapFill added only 23 reactions (2.1% of the initial size of the GSM) to the $iJR904$ network, 41.5 reactions (1.7%) to the $iAF1260$ network, and 33.5 reactions (1.3%) to the $iJO1366$ network. On average, Meneco returned 1.6% fewer reactions (*i.e.* 4) than GapFill. Interestingly, although Meneco and GapFill solutions are comparable in sizes, they only share 45.3% of the reactions in their contents (data not shown) meaning that both tools complete different pathways for the production of the targets. We analyzed the size of GapFill and Meneco completion with regards to the degradation rates of draft networks. There is a correlation between both: the size of the completion tends to grow when the degradation rate rises.

Highlights

On average, GapFill returned 1.6% more reactions (*i.e.* 4) than Meneco, which is negligible. Yet, although Meneco and GapFill solutions are comparable in sizes, they only share 45.3% of the reactions.

2.2.2 Meneco surpasses GapFill for restoration of biomass producibility

The capacity of the completed models to produce biomass was tested using FBA and the constraint-based definition of producibility (Def. 1.5). Results are shown in Fig. 2.2. They evi-

Table 2.3: *Characteristics of the 2,800 networks with 10% degradation rate completed either with Meneco or GapFill using the original GSM as a database.*

Reference network	*iJR*904	*iAF*1260	*iJO*1366
Characteristics			
reactions	1075	2383	2582
compounds	1800	1967	2129
Reactions added by Meneco			
min	0	4	0
max	10	18	11
mean	6.99	9.09	6.43
Reactions added by GapFill			
min	0	0	0
max	11	18	12
mean	4.44	19.14	7.26

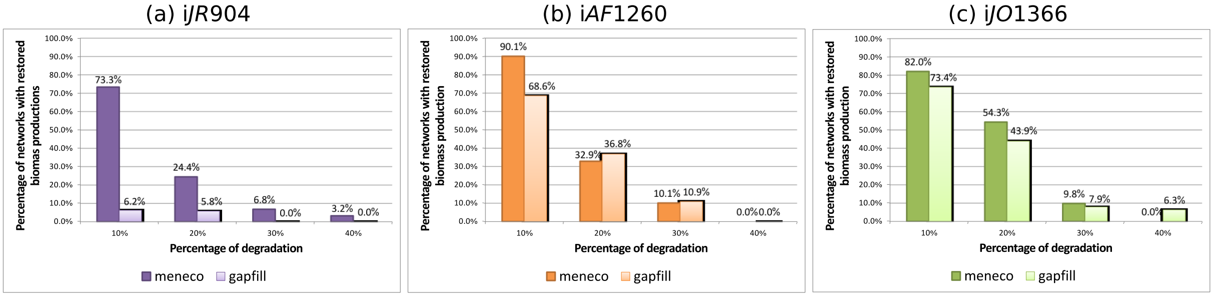
dence that Meneco is capable of restoring biomass production for 3,488 of the 10,800 degraded networks (32.3%) while GapFill restores the biomass production of 2,338 networks (21.6%).

A major difference between Meneco and GapFill can be observed for 10%-degraded networks: Meneco succeeded in restoring biomass production for 2,209 of the 2,700 degraded networks (81.8%) while GapFill restored the biomass production of 1,334 (49.4%) of them. Based on the aforementioned comparative analysis of BioCyc networks with different levels of manual curation, a 10% degradation rate can be considered realistic for automatically-reconstructed draft GSMs.

The results also suggest that the quality of the network has a strong impact on the performance of Meneco and GapFill: For the $iJR904$ network, both methods restored biomass production in 85 (2.3%) cases and they both failed in 2,602 (72.2%) cases. Meneco succeeded while GapFill failed in 890 (24.7%) cases whereas Meneco was outperformed by GapFill in 23 (0.6%) cases among the 3,600 ones (Fig. 2.2 (a)). For the $iAF1260$ and the $iJO1366$ networks, the capabilities of Meneco and GapFill are much more comparable (Fig. 2.2 (b) and (c)). This could be explained by the fact that $iAF1260$ and $iJO1366$ networks are more recent and of higher quality and robustness than $iJR904$, in turn suggesting that the capability of GapFill to restore biomass production depends on the network stoichiometry and topology whereas Meneco is more tolerant to inconsistencies.

Highlights

The relaxed constraint-based modeling used in GapFill strongly impairs its ability to restore flux in biomass reactions. Graph-based approximation of metabolism outperforms constraint-based approximation for parsimonious gap-filling tools when evaluating quality using Flux Balance Analysis.



Number of functional networks after gap-filling

	gapfill failed	gapfill succeeded	Total
meneco failed	2602	23	**2625**
10% degradation	240	0	**240**
20% degradation	652	23	**675**
30% degradation	839	0	**839**
40% degradation	871	0	**871**
meneco succeeded	890	85	**975**
10% degradation	604	56	**660**
20% degradation	196	29	**225**
30% degradation	61	0	**61**
40% degradation	29	0	**29**
Total	**3492**	**108**	**3600**

Number of functional networks after gap-filling

	gapfill failed	gapfill succeeded	Total
meneco failed	2304	98	**2402**
10% degradation	89	0	**89**
20% degradation	527	77	**604**
30% degradation	788	21	**809**
40% degradation	900	0	**900**
meneco succeeded	250	948	**1198**
10% degradation	194	617	**811**
20% degradation	42	254	**296**
30% degradation	14	77	**91**
40% degradation	0	0	**0**
Total	**2554**	**1046**	**3600**

Number of functional networks after gap-filling

	gapfill failed	gapfill succeeded	Total
meneco failed	2119	166	**2285**
10% degradation	104	58	**162**
20% degradation	411	0	**411**
30% degradation	761	51	**812**
40% degradation	843	57	**900**
meneco succeeded	297	1018	**1315**
10% degradation	135	603	**738**
20% degradation	94	395	**489**
30% degradation	68	20	**88**
40% degradation	0	0	**0**
Total	**2416**	**1184**	**3600**

Figure 2.2: ***Impact of Meneco and GapFill on the restoration of biomass production***

Percentages and numbers of degraded networks capable of producing biomass after gap-filling with Meneco and GapFill for three different initial reference networks (iJR904, iAF1260, iJO1366). Among the complete benchmark of networks degraded with a rate of 10%, Meneco restored the biomass production of 81.8% of networks, while GapFill restored the biomass production of 49.4% of networks.

2.2.3 Meneco recovers essential reactions of the three *E. coli* GSMs

The reactions of *iJR*904 were classified according to their functionality with respect to the production of its associated 90 random biomass functions. According to this classification, we tested how many essential, blocked and alternative reactions were recovered in the networks filled by Meneco and GapFill. Our analysis shows that Meneco was able to recover 97.5% of the essential reactions on average among the 10,800 experiments, and few blocked reactions were included in the networks filled by this tool.

To gain better insights into the importance of essential and alternative reactions in the gap-filling procedure, we classified each gap-filling experiment according to four categories:

(i) the model is functional after gap-filling

(ii) the model has recovered all essential reactions after gap-filling but it is not functional

(iii) the gap-filling procedure missed one essential reaction

(iv) the gap-filling procedure missed more than one essential reaction

Results are depicted in Fig. 2.3. They confirm that in 9,529 completions (88.2%), Meneco recovered all essential reactions of the reference network. When failing to restore network functionality, it still recovered all essential reactions in 6,041/7,312 completions (82.3%). In the cases in which Meneco did not recover all essential reactions, it generally missed a single essential reaction, and at most 3 (in only 90/10,800 cases - 0.8%). The same study applied to the GSMs completed with GapFill shows different results (Fig. 2.4): in average 68.3% of them contain all the essential reactions. Shifts between the three models occur more clearly with GapFill as only 52.6% of *iJR*904 models recovered all essential reactions and nearly 20% miss more than one. This shows again that GapFill is less robust on this particular version of the *E. coli* metabolic model. Finally, the number of essential reactions missed by model by GapFill is increased compared to Meneco: a maximum of three was missed by the graph-based method, the maximum is of 21 for the constraint-based one.

Failure to restore biomass production is mainly explained by missing alternative pathways which were not restored by the gap-filling procedures. This was confirmed by analyzing the status of reactions in the 3,488 networks reconstructed with Meneco and capable of producing biomass: among the reactions that were essential in the reconstructed network, on average 40% were classified as alternative in the reference network. Similarly, 47% of the blocked reactions in gap-filled networks were classified as alternative in the initial one.

Highlights

This data demonstrates that although both Meneco and GapFill are parsimonious, the graph-based one presents better results when tested on constraint-based criteria. This indicates that both producibility semantics are not incompatible but on the contrary that they might be complementary one to another.

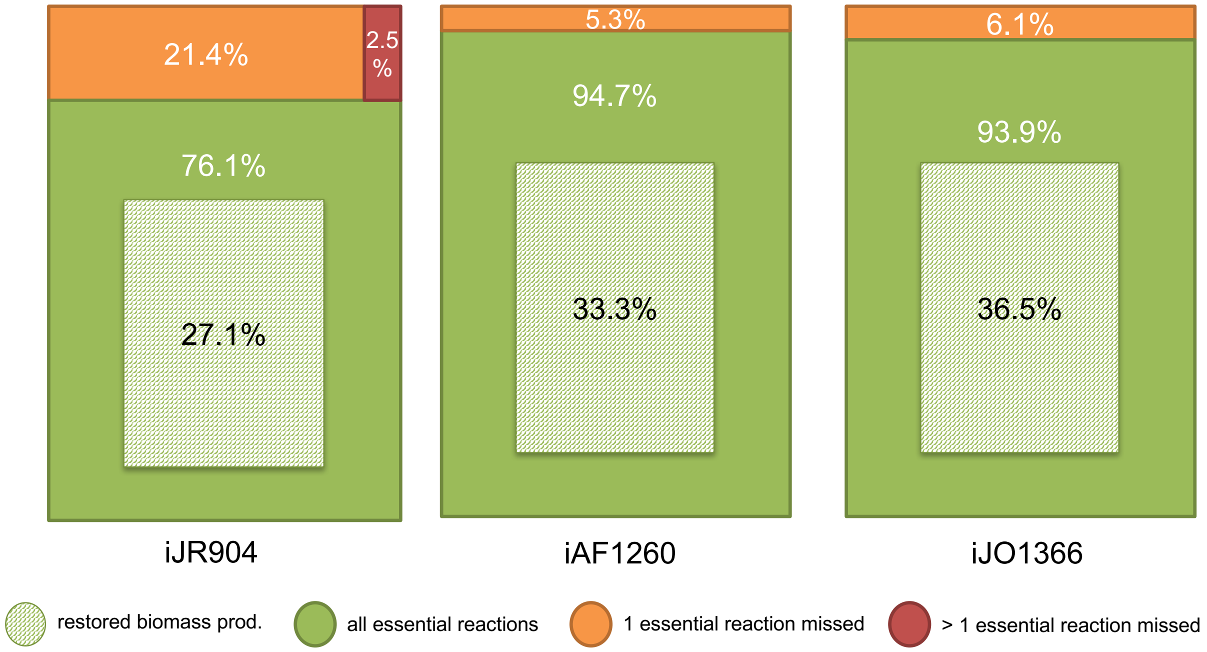


Figure 2.3: ***Biomass restoration and recovery of essential reactions due to completion of 10,800 degraded networks by Meneco.***

*For the 10,800 degraded iJR*904*, iAF*1260 *and iJO*1366 *networks, the gap-filling results were classified according to their status: (i) restored biomass production (green and white stripes), (ii) recovery of all essential reactions (green), (iii) exactly one missed essential reaction (orange) and (iv) more than one missed essential reaction (red).*

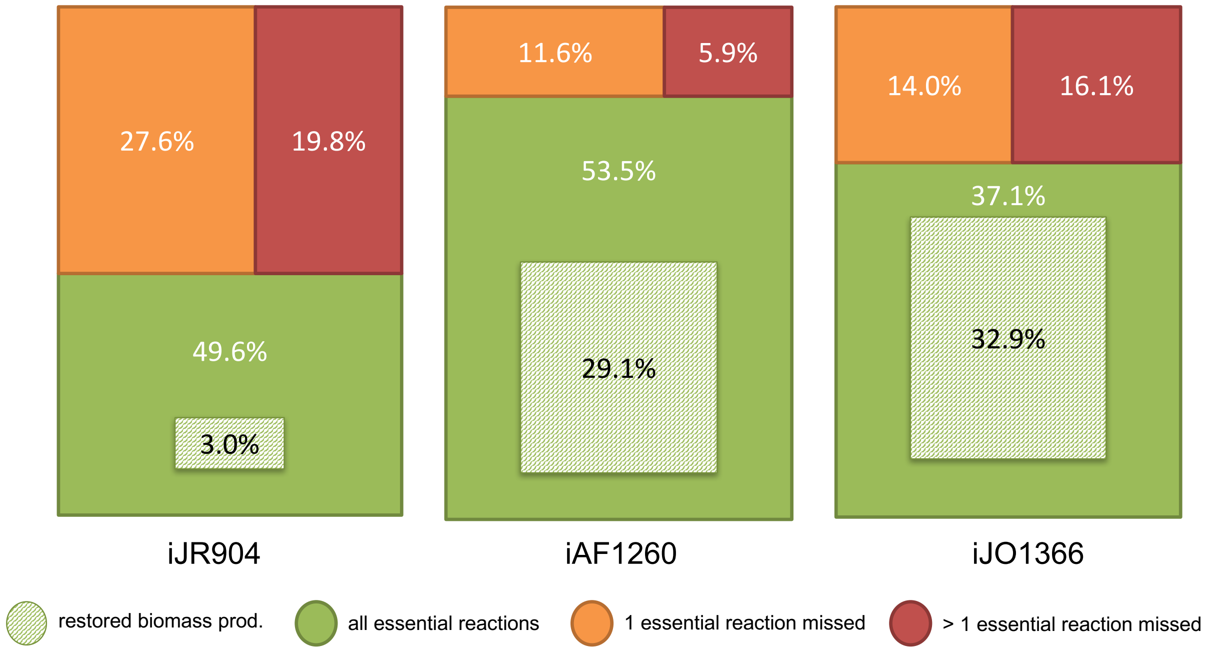


Figure 2.4: ***Biomass restoration and recovery of essential reactions due to completion of 10,800 degraded networks by GapFill.***

For the 10,800 degraded iJR904, iAF1260 and iJO1366 networks, the gap-filling results were classified according to their status: (i) restored biomass production (green and white stripes), (ii) recovery of all essential reactions (green), (iii) exactly one missed essential reaction (orange) and (iv) more than one missed essential reaction (red).

2.2.4 Parsimonious topological gap-filling conserves the role of reactions

We further investigated the networks after their gap-filling by Meneco and compared them to their reference network, to check whether the gap-filling process changed the classification of reactions. As FVA can only be applied if biomass can be quantitatively produced, we studied the 3,488 networks (among the 10,800) capable of biomass production after their gap-filling. To do so, we compared the classification of reactions before degradation to the classification obtained after degradation and gap-filling. In all cases, reactions that were initially blocked with respect to biomass production remained blocked after the completion. In the same way, 98.6% of the reactions initially essential remained essential after the completion, the others becoming alternative. This slight difference could be explained by rounding errors made by the solver when computing FVA.

The study of reactions that were initially alternative is most interesting. Since the completion by Meneco is parsimonious, one can expect some of the initially alternative reactions to become either blocked or essential, by blocking some long production pathways and making the shortest one essential for biomass production. Examples of such situations are depicted in Figure 2.5. 63.3% of the reactions remained alternative, while 34.7% became blocked and 2% essential. On average, for each reconstructed network, 30 initially alternative reactions became essential. From our point of view, this number is low enough to enable a manual curation of the completion results, once more demonstrating that Meneco is a good decision support tool. It is also interesting to note that the quantity of reactions that change classification is similar for each of the three tested networks.

Meneco restores the essential and blocked features of the reactions according to the available information. Due to the minimality criterion, it prefers shorter pathways, thus turning some alternative reactions into either blocked or essential reactions. A change from essential to blocked for a reaction would indicate a failure in the gap-filling process. Since an essential reaction is mandatory to restore biomass production in FBA, gap-filling methods always retrieve functional pathways retaining reactions that were initially essential. While keeping the essential characteristic of the reactions is important, the fact that blocked reactions remain blocked is a sign that performing the gap-filling did not create new functional pathways.

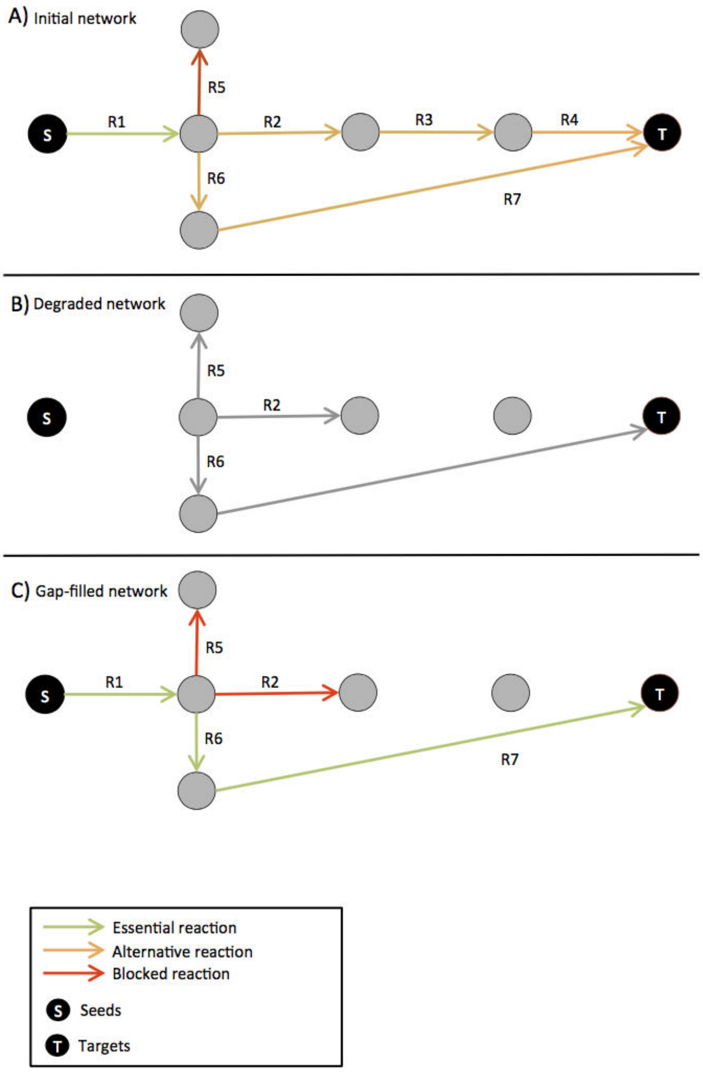


Figure 2.5: ***Possible FVA status changes after completion with respect to the initial network***

The figure depicts a simple network, with compounds as circles and reactions as arrows. Essential reactions are green, alternative ones orange and blocked ones are red. No FVA can be performed on the degraded network (grey arrows) as biomass production is no longer possible. After gap-filling, R_1 *stays essential and* R_5 *stays blocked. However, the former alternative* R_6 *ans* R_7 *became essential. The former alternative* R_2 *became blocked.*

Highlights

Altogether these features confirm that the topological over-approximation, when compared to stoichiometric criteria, is relevant for the parsimonious gap-filling of a new draft metabolic network not only in terms of performances but also in terms of functional accuracy.

2.3 Meneco handles a real-scale reference completion database

The previous results described the interest of using using a graph-based criterion of producibility over a constraint-based one for parsimonious gap-filling. This section presents the ability of Meneco to scale to large databases and the comparison to other gap-filling methods, not necessarily parsimonious, in terms of completion size, restoring constraint-based producibility of the biomass, and accuracy in the relevance of added reactions with respect to the latter.

2.3.1 Meneco and fastGapFill scale to the large-scale MetaCyc database

Gap-filling methods show variable performances on large databases The four tools were tested for computational feasibility on the 3,600 $iJR904$ GSMs of our benchmark with the MetaCyc reference database, to check which methods were applicable in practice.

- Meneco can sample the whole space of solutions and compute their enumeration, union and intersection, which is an asset of the combinatorial ASP solving. The enumeration of all solutions to the parsimonious topological gap-filling problem with Meneco ended in 3,326 cases over the 3,600 studied cases. Our computation reported that there are 1,798 solution sets on average (minimum: 1, maximum: 829,440), suggesting that many combinations of reactions may restore the production of all targets and that considering a single solution to the gap-filling problem is too restrictive. Consequently we defined the output of Meneco to be the set of all reactions appearing in at least one of the minimal sets. Using an efficient ASP solving strategy, such output can be computed in three minutes on average on a single core.
- The computational time of fastGapFill to report a single set of reactions to be added to each of the 3,600 GSM was, on average, between one and two minutes. This computational time to obtain an unique solution was equivalent to those of Meneco as soon an efficient Mixed Integer Linear Programming (MILP) solver was used.
- MIRAGE algorithm has to be run 100 times to rank the reactions. This ranking was used in a last run of the algorithm to report a final solution to the MIRAGE gap-filling problem. In average, one iteration of the algorithm per GSM lasted around 30 minutes when completing with MetaCyc. Limited by computational performances, such outputs were computed for a sample of 360 GSMs of our benchmark.
- GapFill The number of enumerated solutions to the topological parsimonious problem confirmed that GapFill can not be used to perform an exhaustive gap-filling of the GSMs in practice. This is due to complexity of the MILP problem solved that forced us to bound the number of solutions reported by GapFill by a too low parameter value to be

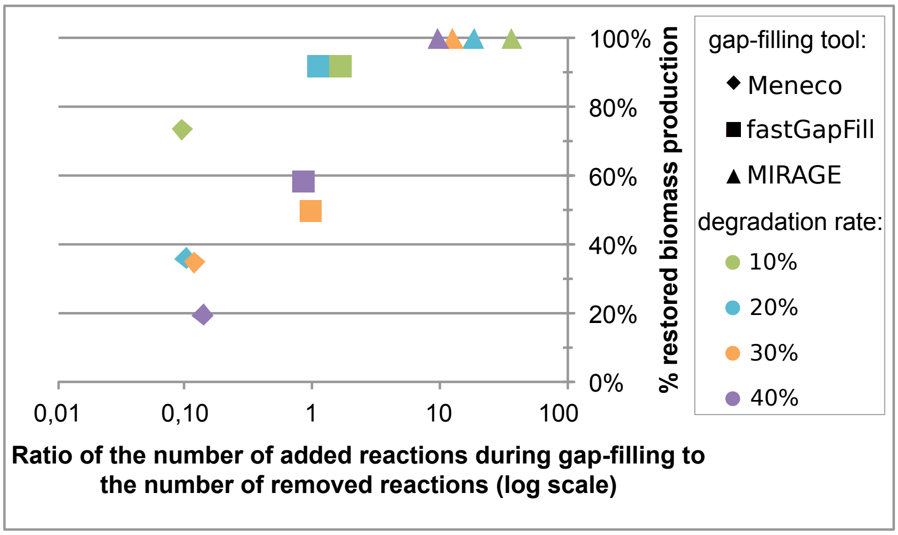


Figure 2.6: ***Comparison of the sizes of the output of the three gap-filling methods Meneco, fastGapFill, MIRAGE***

From 360 (MIRAGE) to 3,600 degraded GSMs (fastGapFill, Meneco) were completed with the gap-filling algorithms using the MetaCyc reference database. GSMs were gathered according to their initial size (90%, 80%, 70% and 60% of the iJR904 E. coli GSM). The number of reactions introduced in each GSM to restore its functionality is compared to the number of reactions removed from the original network and the capability of the completed GSM to restore the producibility of biomass (FBA).

significant. Because of this negative bias for GapFill, this tool was not included in the comparison of the methods using the MetaCyc database.

Highlights

From the applicability point of view, differences occur between the four gap-filling methods. Meneco and fastGapFill algorithms support large databases whereas GapFill does not and will not be compared to the others. MIRAGE hardly scales and was used on a sample of the benchmark.

Meneco and fastGapFill efficiently complete GSMs using reference database of a realistic size. Meneco was thus compared to fastGapFill and MIRAGE in this experiment, as their computational performances enable scaling up to a database made of more than 10,000 metabolic reactions. Results are depicted in Table 2.4 and Figure 2.6.

The sizes of the unions of solutions provided by MIRAGE were too large to be interpreted and manually curated, with an average of 4,029 reactions (minimum=3,481, maximum=4,228) for a metabolic network with an initial size of 1,075 reactions (Fig. 2.6). Note that the average size of one solution among the 100 ones proposed by the union is itself wide: 2,976 (±1,151) reactions. This analysis suggests that MIRAGE is not well-suited to be used with minimal data (draft, seeds, targets and no a priori scoring on the database) for gap-filling and that the algorithm needs all the recommended data (phylogenetic and/or transcriptomic scores) to perform correctly and gain robustness, thus reducing the size of the output. However this kind of information is often sparse or nonexistent for NMOs, to which Meneco aims to be applied. However, as expected given the large sizes of the proposed completions, 100% of GSMs filled by the union of MIRAGE results recovered the biomass synthesis.

For each run, the output of fastGapFill contained on average 87 import or export reactions (minimum = 72, maximum = 108) which were not contained in the reference database. However, in the case that all available metabolites are known (*i.e.* growth medium), it might seem non-relevant to import other internal compounds to unblock fluxes. To be able to compare the fastGapFill results with the other methods we therefore removed import and export reactions from the solutions. 72.88% of the GSMs completed by fastGapFill recovered their biomass synthesis ability, with very high rates for 10% and 20% degraded networks. Altogether, fastGapFill added 273 reactions in average (minimum = 150, maximum = 388). This is slightly more than the number of reactions initially removed from the network (+14% in average) and highly more (+64%) for 10% degraded networks (Fig. 2.6). This suggests that fastGapFill is very efficient to restore the functionality of a network, but that a large number of reactions proposed by fastGapFill should be manually curated before being usable.

A characteristic of Meneco is the very small size of its outputs, which contained from 0 to 110 reactions (32 on average), in line with the parsimonious criteria used. This is less than 15% of the number of reactions removed from the original networks (Fig. 2.6). Yet despite the very low number of added reactions, in average, 40.83% of the networks completed with Meneco recovered the capability of synthesizing biomass. Altogether, 73% of GSMs with a 10% degradation rate in our benchmark became functional after gap-filling. This suggests that the Meneco tool finds a reasonable trade-off between the size of the output to enable a manual curation and the biological significance for relatively poorly degraded GSMs.

Highlights

To sum-up, on average, Meneco returned answers 8 times smaller than fastGapFill and 125 times smaller than MIRAGE. This could enable an easier and faster manual curation of the results. Nevertheless this greatly reduced number of reactions proposed comes with a cost: Meneco restores less functionality, especially when it comes to highly degraded networks. The large number of enumerated solutions to the topological parsimonious problem confirmed that GapFill could not be used to perform an exhaustive gap-filling of the GSMs in practice. Finally analyses suggests that MIRAGE is not suited to be used with minimal data and does not apply to our case-study.

2.3.2 When successful, Meneco improves the accuracy of the reactions added to filled GSMs compared to fastGapFill

The previous subsection showed that Meneco and fastGapFill can scale up to large databases, which is frequently required when working on NMOs, and that fastGapFill presents better results in biomass functionality restoring, especially at high degradation rates. Here we focus on gap-filling cases in which both tools were successful. In order to estimate the accuracy of Meneco and fastGapFill with regard to the proposition of reactions to produce targets, all reactions added to a functional GSM for gap-filling were classified as essential, blocked or alternative with respect to the production of biomass using FVA. The analyses are depicted in Figure 2.7. On average, 63% of the reactions added by fastGapFill were blocked in the completed GSM towards the production of the biomass, *i.e.* the production of the target com-

Table 2.4: *Size of the completions and capability to produce biomass of GSMs in the benchmark of 3,600 degraded E. coli GSMs after their completion with Meneco, fastGapFill and MIRAGE using the MetaCyc database. Between 10% and 40% of reactions were removed from E. coli iJR904 reference network in order to block the constraint-based producibility of 40 different biomass functions. For each degraded GSM, both the Meneco and fastGapFill tools were used to restore the producibility of the biomass by picking reactions from the MetaCyc database. For a sample of 10% of the degraded GSMs in the benchmark, the MIRAGE tool was also tested. The 10%-degradation column is highlighted as it corresponds to an reasonable rate of missing reactions between the automatic draft reconstruction stage and the final GSM.*

Degradation rate of GSMs in the tested benchmark					
	All	10%	20%	30%	40%
Reactions added by Meneco					
min	0	0	4	6	13
max	110	23	36	62	110
mean	32	10	22	38	59
Reactions added by fastGapFill					
min	150	150	213	284	339
max	388	209	256	350	388
mean	273	180	237	311	366
Reactions added by MIRAGE					
min	3,481	3,517	3,678	3,687	3,481
max	4,228	3,951	4,048	4,145	4,228
mean	4,029	3,916	4,005	4,094	4,131
Percentage of filled GSMs able to synthesize biomass					
Meneco	41%	73%	36%	35%	19%
fastGapFill	73%	92%	92%	50%	58%
MIRAGE	100%	100%	100%	100%	100%

pounds. This high rate of blocked reactions was independent of the initial rate of degradation of the considered GSM. This is consistent with the main purpose of fastGapFill which aims at unblocking *all* reactions in the GSM core set (in this case the whole model) without any focus on a set of targeted reactions. Conversely, only 12% of reactions added by Meneco were blocked with respect to the biomass reaction in the completed GSMs, and even less for 10% and 20% degraded GSMs. This confirms that fastGapFill does not solve the exact same problem as Meneco does, that is proposing sets of reactions to unblock unproducible compounds. Since the problem fastGapFill solves is a larger one, it is more difficult to filter, refine and manually curate the proposed solution as it is often required after gap-filling, notably for NMOs.

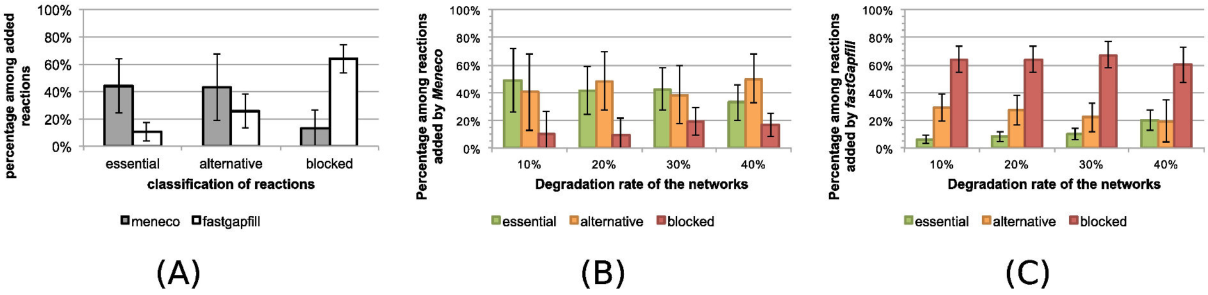


Figure 2.7: ***Classification of reactions added by Meneco and fastGapFill in functional completed GSMs.***

For each degraded GSM which recovered its ability to synthesize biomass after gap-filling, the functional classification (essential, alternative, blocked with respect to to biomass production) of reactions added to the GSM was calculated. (A) Comparison of Meneco and fastGapFill over the complete benchmark in terms of biomass production restoration. (B). Classification of reactions added in functional GSMs filled by Meneco. (C). Classification of reactions added in functional GSMs filled by fastGapFill.

Highlights

Meneco is a relevant tool for the preliminary completion of a new draft metabolic network with large databases when limited phenotypic information is available. Its parsimonious feature allows suggesting reactions directly related to the production of the targets and manual curation is facilitated through a study of the whole space of solutions. This suggests that the topological parsimony criterion used in Meneco provides a good trade-off in terms of scalability with respect to the size of the reference database used for completion: the output of Meneco remains reasonable in terms of size (for an *a posteriori* refinement), with a reasonable loss of impact on the restoration of biomass production for poorly degraded models compared to other approaches.

Conclusion

This chapter compared the graph-based method Meneco to several constraint-based methods (fastGapFill, MIRAGE, GapFill) for gap-filling applications. The first section focused on **two parsimonious methods: Meneco and GapFill** that were tested in a large benchmark of 10,800 completions belonging to three *E. coli* degraded GSMs. Meneco outperformed GapFill on the number of models with restored biomass production. It displayed very good results on 10% degraded models, which is expected to be the degradation ratio of models automatically reconstructed with respect to final models. More importantly, when using FVA to study the types of reactions, Meneco is showed to have excellent results for restoring essential reactions. Altogether the results of this chapter demonstrate that **graph-based gap-filling is appropriate to perform targeted completions of GSMs, even against large databases as it is expected to occur with NMOs**. A limitation is nevertheless visible for some moderately and a large proportion of highly degraded networks. Meneco's performances decrease when the degradation rate of the metabolic network increases. For low-degradation GSMs, Meneco facilitates the model refinements thanks to its parsimonious criterion and the study of the whole solution space. Finally graph-based gap-filling also seems compatible to constraint-based methods as networks completed with Meneco display good results when tested with FBA.

The second section demonstrated that **Meneco and fastGapFill can both scale-up to large databases of reactions**. They were compared using FBA on the biomass reaction, a constraint-based criterion that is acknowledged important in the validation of GSMs reconstructions. Yet both tools differ in their objectives as fastGapFill does not take a particular set of target metabolites or reactions but rather aims at unblocking the whole model. On the contrary, Meneco is flexible and can complete only portions of interest in the model. These differences get clearer when comparing the type of reactions added to the models: a lot of reactions added by fastGapFill are blocked with respect to the biomass production. This leads to large solutions sets that complicate the a posteriori refinement that is needed after gap-filling. **Meneco however produced reduced outputs as it is a parsimonious method, and has the advantage of studying the whole space of completion solutions** by computing very efficiently the union of them in routine. In terms of flux restorations in the objective function, **fastGapFill surpasses Meneco but the latter nevertheless seems to be a good trade-off between the size of the completions and the FBA scores**.

This first study advocates for the use of parsimonious methods for gap-filling and more generally for the use of a targeted gap-filling such that completions can be explored, evaluated and refined by experts. It also demonstrates that despite the fact that Meneco is a graph-based method, it produces good completion results when models are evaluated using a constraint-based criterion. Meneco is the only gap-filling tool that can efficiently restore targeted functions with poor data availability. Its competitors either target every gap of the GSM (fastGapFill), or have less interesting results on restoring functionality (GapFill), or provide non refinable solutions (MIRAGE).

Chapter 3

Hybrid gap-filling reconciles graph-based and constraint-based formalisms

The previous chapter demonstrated the interest of using parsimonious graph-based methods for the gap-filling of metabolic models. It is flexible (definition of targets) and efficient in average for low degradation models. It also pinpointed the importance of comparison with constraint-based methods, in particular Flux Balance Analysis (FBA) to assess the functionality of the reconstructed model. In this chapter I present a hybrid method, Fluto, that enables to complete a model with sets of reactions that satisfy both formalisms. I will present the formalism of the method, its implementation and its benchmarking, extracted from the work published in [Frioux et al., 2017] at the Logic Programming and Nonmonotonic Reasoning (LPNMR) in 2017 and awarded *Best Student paper*.

This chapter's content was for most parts extracted from the paper I coauthored with Sebastian Schellhorn, Anne Siegel, Torsten Schaub and Philipp Wanko; entitled "*Hybrid metabolic network completion*" and presented at the **14th International Conference on Logic Programming and Nonmonotonic Reasoning, LPNMR 2017**, in Espoo, Finland [Frioux et al., 2017]. This paper was awarded **Best student paper** and chosen for a **fast-track publication** of an extended version of the article in the journal TPLP (Theory and Practice in Logic Programming) [Frioux et al., 2018b].

We evidenced in the previous chapter that the Meneco ASP-based method partly restores the bio-synthetic capabilities of a large proportion of moderately degraded GSMs: it fails to restore the ones of both some moderately degraded and most of highly degraded metabolic GSMs. Meneco relies on a recursive graph-based activation of reactions and reachability of compounds for querying a database and select sets of reactions that can restore an observed bio-synthetic behaviour. The main reason for the work we present in this chapter is that the Meneco purely qualitative approach misses quantitative constraints accounting for the law of mass conservation, a major hypothesis about metabolic networks, and particularly important for their validation. Hence, the qualitative ASP-based approach fails to tell apart solution candidates with correct and incorrect stoichiometry balance and therefore may reports inaccurate results for some degraded networks.

We address this in the present chapter by proposing a hybrid approach to metabolic network completion that integrates our qualitative ASP approach with quantitative techniques from FBA, the quantitative approach for capturing reaction rates in metabolic networks. We accomplish this by taking advantage of recently developed theory reasoning capacities for the ASP system *clingo* [Gebser et al., 2016a]. More precisely, we use an extension of *clingo* with linear constraints over reals, as dealt with in Linear Programming (LP [Dantzig, 1963]). This extension provides us with an extended ASP modeling language as well as a generic interface to alternative LP solvers, viz. *cplex*, for dealing with linear constraints.

3.1 Activation-based metabolic network completions

We will present in this section the hybrid activation of reactions in metabolic networks and the associated metabolic completion (gap-filling) problem. Hybrid activation combines graph-based and constraint-based activations that were previously introduced in this thesis but that we will present again here for the sake of clarity.

For illustration of the three semantics and their associated gap-filling, we will consider the metabolic model in Figure 3.1. The network consists of 9 reactions, r_{s_1}, r_{s_2}, r_e and r_0 to r_5, and 8 compounds, $A, \dots, F$, S_1, S_2 and S_3. Here, the seeds are $S = \{S_1, S_2, S_3\}$, S_1 and S_2 being the two boundary compounds of the network. Dashed rectangle describes the boundary of the system, outside of which is the environment of the organism. Consider reaction $r_4 : E \rightarrow 2C$ transforming one unit of E into two units of C (stoichiometric coefficients of 1 are omitted in the graphical representation. We have $\mathit{reactants}(r_4) = \{E\}$, $\mathit{products}(r_4) = \{C\}$, along with $s(E, r_4) = 1$ and $s(r_4, C) = 2$.

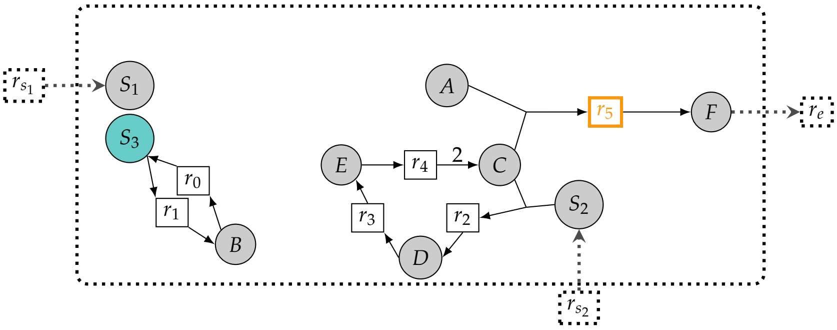


Figure 3.1: ***Example of a metabolic model***

Compounds and reactions are depicted by circles and rectangles respectively. Dashed reactions are reactions involving the boundary between the organism's metabolism and its environment. r_5 is the target or objective reaction. S_1 and S_2 are boundary seeds. S_3 is assumed to be an initiation seed. Numbers on arrows describe the stoichiometry of reaction (default value is 1).

In biology, several concepts have been introduced to model the activation of reaction fluxes in metabolic networks, or to synthesize metabolic compounds. In Subsection 1.4.2, we introduced a function *active* that given a metabolic model G takes a set of seeds $S \subseteq M$ and returns a set of activated reactions, $active_G(S) \subseteq R$, either from a graph-based or a constraint based criterion (Definitions 1.3 and 1.5). As a reminder, the graph-based activation is defined with the following:

Definition 3.1 Graph-based activation of a reaction (already introduced in Def 1.3.) *Given an objective reaction $r_{obj} \in R_{obj}$, a metabolic network $G = (R \cup M, E)$ and a set of seeds S, topological activation is defined as follows:*

$$r_{obj} \in active^t_G(S) \ \ \mathit{iff} \ \ reactants(r_{obj}) \subseteq \Sigma_G(S).$$

with $\Sigma_G(S)$ being the scope of the network starting from the seeds.

The constraint-based activation relies on the steady-state assumption, presented in the following equation, that is another presentation of Equation 1.2:

$$\textstyle\sum_{r \in R} s(r,m) \cdot v_r + \sum_{r \in R} -s(m,r) \cdot v_r = 0 \qquad \text{for } m \in M. \tag{3.1}$$

and on the existence of bounds for fluxes values v_r:

$$LB \leq v_r \leq UB \tag{3.2}$$

Definition 3.2 Constraint-based activation of a reaction (already introduced in Def 1.5.) *Given an objective reaction $r_{obj} \in R_{obj}$, a metabolic network $G = (R \cup M, E, s)$ and a set of seeds S, stoichiometric activation is defined as follows:*

$$r_{obj} \in active^s_G(S) \ \ \mathit{iff} \ \ v_{r_{obj}} > 0 \text{ and (3.1) and (3.2) hold for } M \text{ and } R.$$

The steady-state equation 3.1 can be relaxed with:

$$\textstyle\sum_{r \in R} s(r,m) \cdot v_r + \sum_{r \in R} -s(m,r) \cdot v_r \geq 0 \qquad \text{for } m \in M. \tag{3.3}$$

allowing accumulation of metabolites and leading to the relaxed constraint-based activation:

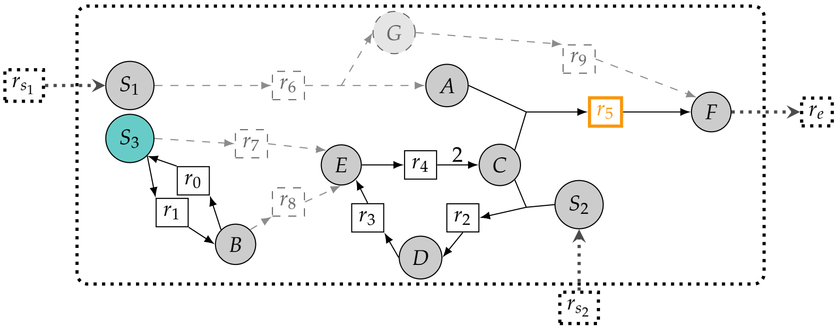


Figure 3.2: ***Parsimonious metabolic model completion problem***

The purpose of its solving is to select the minimal number of reactions from a database (dashed shaded reactions) such that activation of target objective reaction r_5 is restored from boundary and/or initiation seeds. We focus here on three formalisms for activation of target reaction: stoichiometric, topological and hybrid.

Definition 3.3 Relaxed constraint-based activation of a reaction (already introduced in Def 1.6.) *Given an objective reaction $r_{obj} \in R_{obj}$, a metabolic network $G = (R \cup M, E, s)$ and a set of seeds S, stoichiometric activation is defined as follows:*

$$r_{obj} \in active^{r}_{G}(S) \;\; \textit{iff} \;\; v_{r_{obj}} > 0 \textit{ and } (3.3) \textit{ and } (3.2) \textit{ hold for } M \textit{ and } R.$$

In the same subsection, the paragraph *Definition of a gap-filling problem* gave a general definition of gap-filling based on the *active* function. It consists into picking reactions (parsimoniously or not) in a database such that the objective function becomes active. Accordingly, different formulations of metabolic network completion can be characterized: the stoichiometric (or constraint-based), as well as the relaxed stoichiometric, the topological (or graph-based), and finally the hybrid one that will combine features from graph-based and constraint-based semantics. We elaborate upon their formal characterizations in the following subsections to clarify the domain of our work and the results presented afterwards.

3.1.1 Stoichiometric Metabolic Network Completions

Flux distributions are formalized in terms of a system of equations relying on the stoichiometric coefficients of reactions. Reactions stoichiometry is governed by the *law of mass conservation* under a steady state assumption; in other words, the mass of the system remains constant over the reaction. The input and output fluxes of reactions consuming and producing a metabolite are balanced, which are formalized in equation 3.1. The constraint-based or stoichiometric activation of reactions is defined above in Definition 3.3.

Application to the toy example In our draft network G, consisting of all non-dashed nodes and edges depicted in Figure 3.2 or Figure 3.1 (viz. reactions r_{s_1}, r_{s_2}, r_e and r_0 to r_5 and compounds $A, \ldots, F$, S_1, S_2, and S_3 and r_5 the single target reaction). The reference network G', consists of the shaded part of Fig 3.2, (viz. reactions r_6 to r_9 and metabolite G). A strict stoichiometry-based completion aims to obtain a solution with $r_5 \in active^{s}_{G''}(\{S_1, S_2, S_3\})$ where v_{r_5} is maximal. This can be achieved by adding the completion $R''_1 = \{r_6, r_9\}$ (Figure 3.3). The cycle made of compounds E, C, D and the boundary seed S_2 is already balanced and notably self-activated. Indeed, initiation of D and E producibility requires the producibility of C (in addition to the presence of the boundary seed S_2) that itself depends on D and E. Yet, according the flux conditions, that models steady state conditions, the cycle is activated. Such self-activation of cyclic pathways is an inherent feature of purely stoichiometric approaches to network completion. This is a drawback of the semantics, already discussed in Subsection 1.3.2, because the effective activation of the cycle requires the additional (and unchecked) condition that at least one of the compounds was present as the initial state of the system. This could be the case provided there exist another way to enable the production of one or several components of the cycle (here an activable reaction producing E for instance) [Prigent et al., 2017].

Relaxed stoichiometric activation To solve metabolic network completion parsimoniously with flux-balance activated reactions, Linear Programming can be used to maximize the flux rate $v_{r_{obj}}$ of the objective reaction provided that the linear constraints are satisfied. Nonetheless, this problem turns out to be hard to solve in practice and existing approaches scale poorly to real-life applications (cf. [Orth et al., 2010]). This motivated the use of approximate methods. The relaxed problem is obtained by weakening the mass-balance equation 3.1 with the equation 3.3 This is used in the concept of *relaxed constraint-based (or stoichiometric) activation* presented earlier in this section (Definition 3.3). The resulting problem can now be efficiently solved with Linear Programming and has been implemented in GapFill [Satish Kumar et al., 2007]. Existing systems addressing strict stoichiometric network completion either cannot guarantee optimal solutions [Latendresse, 2014] or do not support a focus on specific target reactions [Thiele et al., 2014]. Other approaches either partially relax the problem [Vitkin and Shlomi, 2012] or solve the relaxed problem based on Equation 3.3, like the system GapFill [Satish Kumar et al., 2007]. Applied to the network of Figure 3.2, the minimal completion under the relaxed stoichiometric activation is $R_1'' = \{r_6\}$ (Figure 3.4) but does not carry flux because of the accumulation of metabolite G, allowed by Equation 3.3. Note however that for strict steady-state modeling an *a posteriori* verification of solutions is needed to warrant the exact mass-balance equation 3.1.

Highlights

There can be several parsimonious approaches for gap-filling based on constraint-based activation of reactions. Here we present two of them. The first $active^s_G(S)$ is compliant with FBA and ensures the gap-filled model has flux in its objective reaction. The second one $active^r_G(S)$ is the one GapFill implements and enables accumulation of metabolites which facilitates solving in practice.

3.1.2 Topological Metabolic Network Completion

This definition of completion is the one employed in Meneco that uses the definition of graph-based activated reactions (Definition 3.1). Note that this semantics avoids self-activated cycles by imposing an external entry sufficient to initiate all cycles (in Figure 3.1 S_3 is not enough to activate the cycle as it does not activate one of its reaction on its own). The resulting network completion problem can be expressed as a combinatorial optimization problem and effectively solved with ASP [Schaub and Thiele, 2009b].

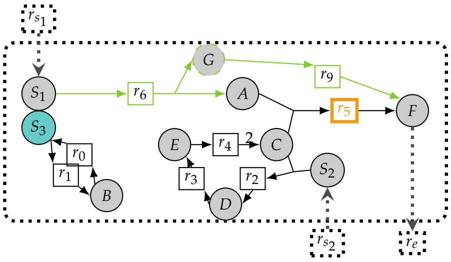


Figure 3.3: ***Stoichiometric activation and metabolic network completion***

Solution to parsimonious metabolic network completion under stoichiometric activation hypothesis in order to satisfy Equations 3.2, 3.1 and Definition 3.2. Within this network, there exists at least one flux distribution which activates r_5.

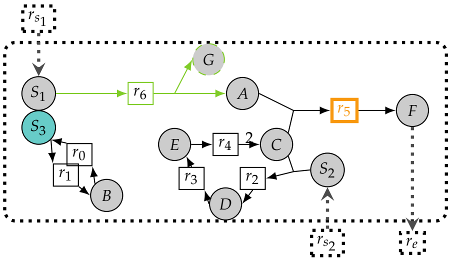
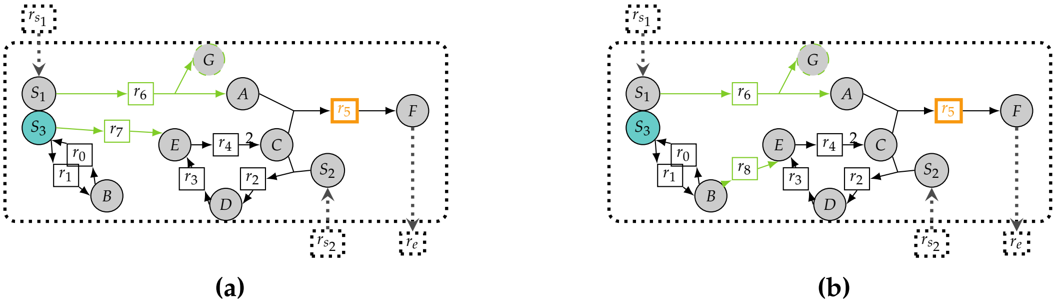


Figure 3.4: ***Relaxed-stoichiometric activation and metabolic network completion***

Solution to parsimonious metabolic network completion under relaxed stoichiometric activation hypothesis in order to satisfy Equations 3.2, 3.3 and Definition 3.3. Notice that within this completed network, there exist no flux distribution allowing the reaction r_5 to be activated due to the accumulation of the G metabolite.



(a)

(b)

Figure 3.5: ***Topological activation and metabolic network completion***

Solutions to metabolic network completion under topological activation hypothesis satisfying Definition 3.1. The production of C cannot be explained by a self-activated cycle and requires an external source of compounds via S_3 and reaction r_7.

Application to the toy example For illustration, consider again the draft and reference networks G and G' in Figure 3.1 and Figure 3.2. We get $\Sigma_G(\{S_1, S_2, S_3\}) = \{S_1, S_2, S_3, B\}$, indicating that target reaction r_5 is not activated from the seeds with the draft network because A and C, its reactants, are not reachable. This changes once the network is completed. Valid minimal completions are $R''_2 = \{r_6, r_7\}$ (Figure 3.5 (a)) and $R''_3 = \{r_6, r_8\}$ (Figure 3.5 (b)) because $r_5 \in active^t_{G''_i}(\{S_1, S_2\})$ since $\{A, C\} \subseteq \Sigma_{G''_i}(\{S_1, S_2\})$ for all extended networks G''_i obtained from completions R''_i of G for $i \in \{2, 3\}$.

Highlights

A second method to complete metabolic networks relies on the graph-based activation of reactions $active^t_G(S)$. It corresponds to the model adopted by Meneco considering that it tales as targets the reactants of objective reactions.

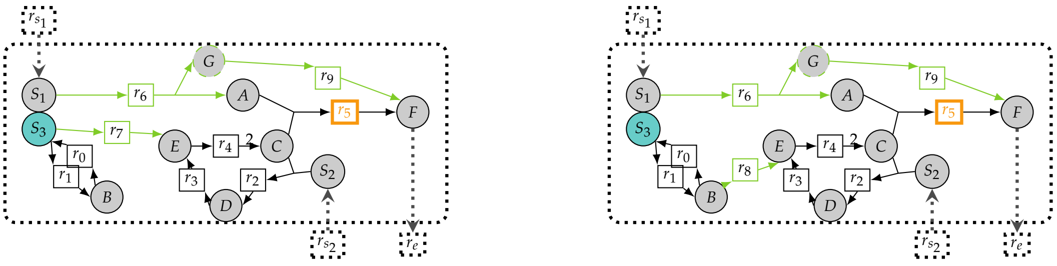
3.1.3 Hybrid Metabolic Network Completion

Hybrid activation and application to the toy example The idea of hybrid metabolic network completion is to combine the two previous activation semantics: the topological one accounts for a well-founded initiation of the system from the seeds and the stoichiometric one warrants its mass-balance. We thus aim at network completions that are both topologically functional and flux balanced (without suffering from self-activated cycles). More precisely, we obtain the following definition:

Definition 3.4 Hybrid activation of a reaction *A reaction $r_{obj} \in R_{obj}$ is* hybridly activated *from a set S of seeds in a network G, if both criteria apply:*

$$r_{obj} \in active^h_G(S) \;\; iff \;\; r_{obj} \in active^s_G(S) \; and \; r_{obj} \in active^t_G(S).$$

Applying this to our example in Figure 3.2, we get the (minimal) hybrid solutions $R''_4 = \{r_6, r_7, r_9\}$ (Figure 3.6 (a)) and $R''_5 = \{r_6, r_8, r_9\}$ (Figure 3.6 (b)). Both (topologically) initiate paths of reactions from the seeds to the target, ie. $r_5 \in active^t_{G''_i}(\{S_1, S_2, S_3\})$ since $\{A, C\} \subseteq \Sigma_{G''_i}(\{S_1, S_2, S_3\})$ for both extended networks G''_i obtained from completions R''_i of G for $i \in$



(a) **(b)**

Figure 3.6: ***Solutions to hybrid metabolic network completion***
Two solutions under hybrid activation hypothesis satisfying Definition 3.4 (that is Definitions 3.3 and 3.1).

$\{4,5\}$. Both solutions are as well stoichiometrically valid and balance the amount of every metabolite, hence we also have $r_5 \in active^s_{G''_i}(\{S_1, S_2, S_3\})$.

Highlights

Hybrid activation $active^h_G(S)$ ensures that flux flows in the objective reactions (stoichiometric activation) and that their reactants are graph-based producible (topological activation). It enables to make the model functional at both initial and steady states.

3.1.4 Union of Metabolic Network Completions

Union of solutions is useful in metabolic models gap-filling As depicted in the toy examples for the topological (Figure 3.5) and hybrid (Figure 3.6) activation, several minimal solutions to one metabolic network completion problem may exist. There might be dozens of minimal completions, depending on the degradation of the original draft network, hence leading to difficulties for biologists and computational biologists to discriminate the individual results. One solution to facilitate this curation task is to provide, in addition to the enumeration of solutions, their union. This has been done previously for the topological completion (Chapter 2 and the associated publication in PLOS Computational Biology [Prigent et al., 2017]).

Notably, the concept of "union of solutions" is particularly relevant from the biological perspective since it provides in a single view all possible reactions that could be inserted in a solution to the network completion problem. Additionally, verifying the union according to the desired (stoichiometric and hybrid) activation semantics, offers a way to analyze the quality of approximation methods (topological and relaxed-stoichiometric ones). If individual solutions contradict a definition of activation that the union satisfies, it suggests that the family of reactions contained in the union, although possibly non-minimal, may be of interest. Thus providing merit to the approximation method and their results.

Stability of the union of solutions with the three activations Importantly, we notice that the operation of performing the union of solutions is stable with the concept of activation, although it can contradict the minimality of the size of completion. Indeed, as shown in the following propositions, the union of solutions to the topological network completion problem is itself a (non-minimal) solution to the topological completion problem. Similarly, the union of minimal stoichiometric solutions always displays the stoichiometric activation of the target reaction(s). In fact, adding an arbitrary set of reactions to a metabolic network still maintains stoichiometric activation, since flux distribution for the newly added reactions may be set to zero. Consequently, the union of minimal hybrid solutions always displays the hybrid activation in the target reaction(s).

The following propositions (Propositions 1, 2 and 3) are a formalization of the stability of the union of solutions with respect to the three concepts of activation of a set of objective reactions R_{obj}.

The union $G = G_1 \cup G_2$ of two metabolic networks $G_1 = (R_1 \cup M_1, E_1, s_1)$ and $G_2 = (R_2 \cup M_2, E_2, s_2)$ is defined by

$$\begin{aligned} G &= (R \cup M, E, s), \\ R &= R_1 \cup R_2, \\ M &= M_1 \cup M_2, \\ E &= E_1 \cup E_2, \\ s &= s_1 \cup s_2. \end{aligned}$$

Proposition 1. *Let G_1 and G_2 be metabolic networks with a set of objective reactions R_{obj}. If $R_{obj} \subseteq active^t_{G_1}(S)$, then $R_{obj} \subseteq active^t_{G_1 \cup G_2}(S)$.*

Proof. The proof is given by monotonicity of the union and the monotonicity of the closure. Thus it can never be the case that having more reactions disables reachability. More formal, $R_{obj} \subseteq active^t_{G_1}(S)$ holds iff $reactants(r_{obj}) \subseteq \Sigma_{G_1}(S)$. Furthermore, we have $\Sigma_{G_1}(S) \subseteq \Sigma_{G_1 \cup G_2}(S)$ by the definition of the closure. This implies $reactants(r_{obj}) \subseteq \Sigma_{G_1 \cup G_2}(S)$. Finally, we have $R_{obj} \subseteq active^t_{G_1 \cup G_2}(S)$. □

Proposition 2. *Let G_1 and G_2 be metabolic networks. If $R_{obj} \subseteq active^s_{G_1}(S)$, then $R_{obj} \subseteq active^s_{G_1 \cup G_2}(S)$.*

Proof. First, we define following bijective functions

$$\begin{aligned} f :& R_1 \to \{1, \dots, l\} \subseteq \mathbb{N}, \\ & r \mapsto f(r) = i \\ g :& M_1 \to \{1, \dots, k\} \subseteq \mathbb{N}, \\ & m \mapsto g(m) = j \\ f' :& R_1 \cup R_2 \to \{1, \dots, l'\} \subseteq \mathbb{N}, \\ & r \mapsto f'(r) = \begin{cases} f(r) & \text{, if } f(r) \text{ is defined} \\ i & \text{, otherwise} \end{cases} \\ g' :& M_1 \cup M_2 \to \{1, \dots, k'\} \subseteq \mathbb{N} \\ & m \mapsto g'(m) = \begin{cases} g(m) & \text{, if } g(m) \text{ is defined} \\ j & \text{, otherwise} \end{cases} \end{aligned}$$

for $k = |M_1|$, $l = |R_1|$, $k' = |M_1 \cup M_2|$ and $l' = |R_1 \cup R_2|$ regarding G_1 and $G_1 \cup G_2$, respectively. Now, we rewrite the system of (3.1) regarding G_1 as a matrix equation $Av = 0$ of form

$$\begin{pmatrix} a_{11} & \dots & a_{1l} \\ \vdots & \ddots & \vdots \\ a_{k1} & \dots & a_{kl} \end{pmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_l \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$$

where A is a $k \times l$ matrix with coefficients

$$a_{g(m)f(r)} = \begin{cases} s_1(r,m) & ,(r,m) \in E_1 \\ -s_1(m,r) & ,(m,r) \in E_1 \\ 0 & \text{, otherwise} \end{cases}$$

and v consists of variables $v_{f(r)}$ for $r \in R_1$. By $L = \{v \mid Av = 0\}$ we denote the set of solutions induced by $Av = 0$.

Furthermore, we represent the system of linear equations of (3.1) regarding $G_1 \cup G_2$ as a matrix equation $A'v' = 0$ of form

$$\begin{pmatrix} a_{11} & \dots & a_{1l} & a_{1l+1} & \dots & a_{1l'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ a_{k1} & \dots & a_{kl} & a_{kl+1} & \dots & a_{kl'} \\ 0 & \dots & 0 & a_{k+1l+1} & \dots & a_{k+1l'} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & \dots & 0 & a_{k'l+1} & \dots & a_{k'l'} \end{pmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_l \\ v_{l+1} \\ \vdots \\ v_{l'} \end{pmatrix} = \begin{pmatrix} 0 \\ \vdots \\ 0 \end{pmatrix}$$

where A' is a $k' \times l'$ matrix with coefficients

$$a_{g'(m)f'(r)} = \begin{cases} s(r,m) & ,(r,m) \in E_1 \cup E_2 \\ -s(m,r) & ,(m,r) \in E_1 \cup E_2 \\ 0 & \text{, otherwise} \end{cases}$$

where $s = s_1 \cup s_2$ and v' consists of variables $v_{f'(r)}$ of (3.1) for $r \in R_1 \cup R_2$. Note that A' can always be written in this form, since switching columns and rows will not change solutions. By $L' = \{v' \mid A'v' = 0\}$ we denote the set of solutions induced by $A'v' = 0$.

Since $A'v' = 0$ is homogeneous, $L \subseteq L'$ holds by extending L with zeros for $v_{f'(r)}$ with $r \in R_2 \setminus R_1$. Thus $\{v \mid v \in L, \forall r_{obj} \in R_{obj}, v_{f(r_{obj})} > 0\} \subseteq \{v \mid v \in L', \forall r_{obj} \in R_{obj}, v_{f'(r_{obj})} > 0\}$ by extending the first set with zeros for $v_{f'(r)}$ with $r \in R_2 \setminus R_1$. From $R_{obj} \subseteq active^s_{G_1}(S)$, we know that the homogeneous system of linear equations from (3.1) regarding G_1 is non-trivial satisfiable, which finally implies that $R_{obj} \subseteq active^s_{G_1 \cup G_2}(S)$. □

Proposition 3. *Let G_1 and G_2 be metabolic networks. If $R_{obj} \subseteq active^h_{G_1}(S)$, then $R_{obj} \subseteq active^h_{G_1 \cup G_2}(S)$.*

Proof. Follows directly by the definition of hybrid activation together with Theorem 1 and Theorem 2. More formal, $R_{obj} \subseteq active^h_{G_1}(S)$ holds iff $R_{obj} \subseteq active^t_{G_1}(S)$ and $R_{obj} \subseteq active^s_{G_1}(S)$. From Theorem 1 and $R_{obj} \subseteq active^t_{G_1}(S)$ follows $R_{obj} \subseteq active^t_{G_1 \cup G_2}(S)$. Analogously, from

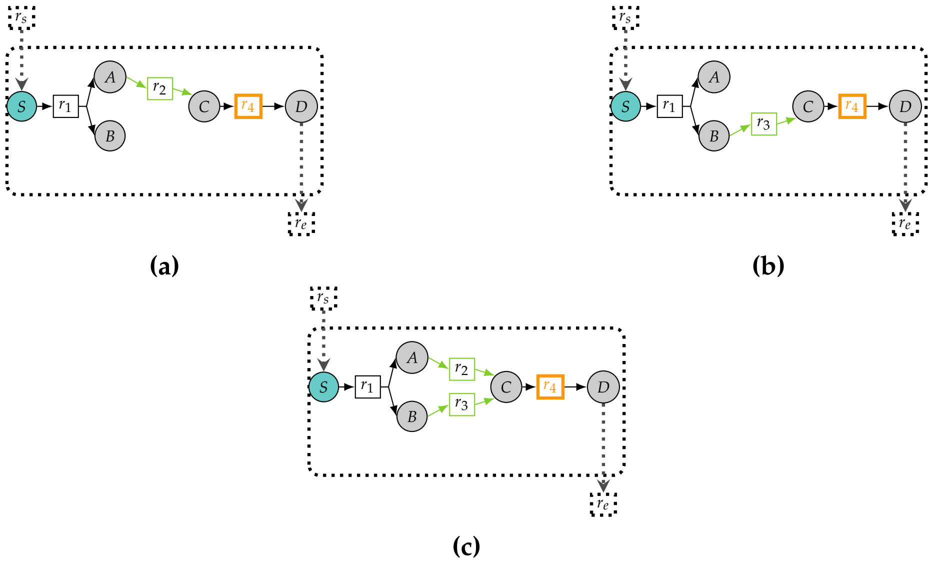


Figure 3.7: ***Union of no-flux solutions carries flux***

(a) Topological completion $R_1 = \{r_2\}$ satisfies $r_4 \in active^t_{G_1}(\{S\})$, but carries no flux, due to accumulation of compound B that contradicts Eq. 3.1. (b) Topological completion $R_2 = \{r_3\}$ satisfies $r_4 \in active^t_{G_2}(\{S\})$ and carries no flux as well, due to accumulation of compound A that contradicts Eq. 3.1. (c) Completion with the union $R_1 \cup R_2 = \{r_2, r_3\}$. $G = G_1 \cup G_2$ satisfies $r_4 \in active^h_G(\{S\})$ and thus is flux-balanced.

Theorem 2 and $R_{obj} \subseteq active^s_{G_1}(S)$ follows $R_{obj} \subseteq active^s_{G_1 \cup G_2}(S)$. Finally, this implies $R_{obj} \subseteq active^h_{G_1 \cup G_2}(S)$. □

In particular, studying the union in case of topological modeling can pinpoint interesting cases. Individual solutions satisfying the topological activation can additionally satisfy the stoichiometric and thus the hybrid activation semantics. A union including such a solution will also adhere to the hybrid standard. In some cases, the union of solutions will display the stoichiometric activation whereas the individual solutions only satisfy the topological activation. Figure 3.7 displays an example of topological metabolic network completions that do not satisfy stoichiometric (and hybrid) activation whereas their union does. Figure 3.8 provide an example of minimal topological completions that do not satisfy stoichiometric (and hybrid) activation and for which the union does not satisfy it either. Both observations induce that in general we cannot derive anything about activation of reactions in a graph resulting from the union of two or more graphs. And similarly, we cannot infer about the activation of reactions in subgraphs arbitrarily derived from a graph in which these reactions are activated.

Highlights

Considering the union of gap-filling solutions is useful for parsimonious methods. It enables the users to consider reactions occurring in all solutions and possibly refine them according to additional criteria. We showed that for the topological, stoichiometric and hybrid formalisms, the union is stable with the activation definitions.

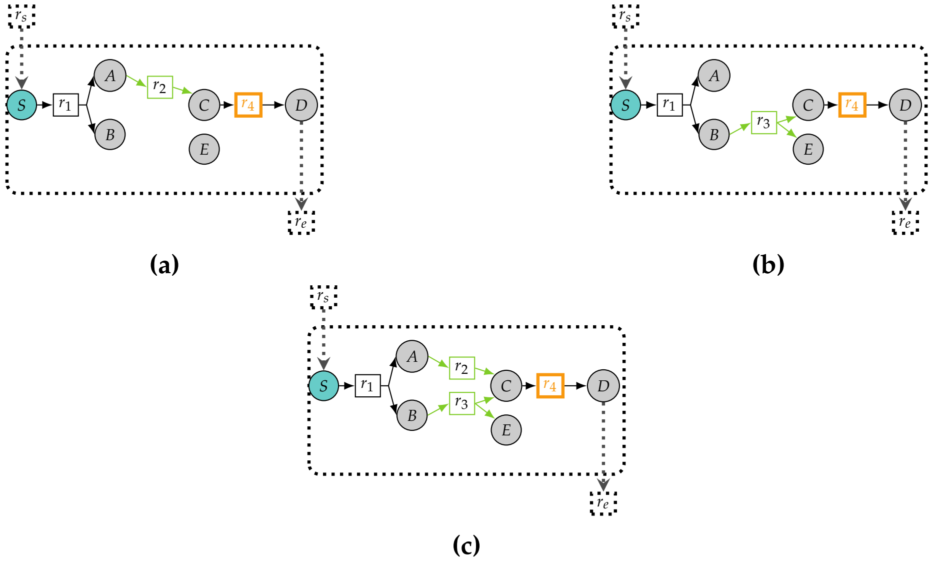


Figure 3.8: ***Union of no-flux solutions carries no flux***

(a) Topological completion $R_1 = \{r_2\}$ satisfies $r_4 \in active^t_{G_1}(\{S\})$, but carries no flux, due to accumulation of compound B that contradicts Eq. 3.1. (b) Topological completion $R_1 = \{r_3\}$ satisfies $r_4 \in active^t_{G_2}(\{S\})$, but carries no flux, due to accumulation of compounds A and E that contradicts Eq. 3.1. (c) Completion with the union $R_1 \cup R_2 = \{r_2, r_3\}$. $G = G_1 \cup G_2$ satisfies $r_4 \in active^t_G(\{S\})$, but contradicts minimality and carries no flux $r_4 \notin active^s_G(\{S\})$, due to accumulation of compound E that contradicts Eq. 3.1.

3.2 Linear Constraints to extend ASP-based gap-filling

For encoding our hybrid problem, we rely upon the theory reasoning capacities of the ASP system *clingo* that allows us to extend ASP with linear constraints (LC) over reals (as addressed in Linear Programming). We confine ourselves below to features relevant to our application and refer the interested reader for details to [Gebser et al., 2016a].

As usual, a *logic program* consists of *rules* of the form

$$\texttt{a}_0 \texttt{ :- } \texttt{a}_1\texttt{,}\ldots\texttt{,a}_m\texttt{,not a}_{m+1}\texttt{,}\ldots\texttt{,not a}_n\texttt{.}$$

where each $\texttt{a}_i$ is either a *(regular) atom* of form $\texttt{p(t}_1\texttt{,}\ldots\texttt{,t}_k\texttt{)}$ where all $\texttt{t}_i$ are terms or a *linear constraint atom* of form[1] '$\texttt{\&sum\{w}_1\texttt{*x}_1\texttt{;}\ldots\texttt{;w}_l\texttt{*x}_l\texttt{\} <= k}$' that stands for the linear constraint $w_1 \cdot x_1 + \cdots + w_l \cdot x_l \leq k$. All $\texttt{w}_i$ and $\texttt{k}$ are finite sequences of digits with at most one dot[2] and represent real-valued coefficients w_i and k. Similarly all $\texttt{x}_i$ stand for the real-valued variables x_i. As usual, `not` denotes (default) *negation*. A rule is called a *fact* if $n = 0$.

Semantically, a logic program induces a set of *stable models*, being distinguished models of the program determined by stable models semantics [Gelfond and Lifschitz, 1991]. Such a stable model X is an *LC-stable model* of a logic program P,[3] if there is an assignment of reals to all real-valued variables occurring in P that (i) satisfies all linear constraints associated with linear constraint atoms in P being in X and (ii) falsifies all linear constraints associated with

[1] In *clingo*, theory atoms are preceded by '&'.

[2] In the input language of *clingo*, such sequences must be quoted to avoid clashes.

[3] This corresponds to the definition of T-stable models using a *strict* interpretation of theory atoms [Gebser et al., 2016a], and letting T be the theory of linear constraints over reals.

linear constraint atoms in P being not in X. For instance, the (non-ground) logic program containing the fact 'a("1.5").' along with the rule '&sum{R*x} <= 7 :- a(R).' has the stable model {a("1.5"),&sum{"1.5"*x}<=7}. This model is LC-stable since there is an assignment, e.g. $\{x \mapsto 4.2\}$, that satisfies the associated linear constraint '$1.5 * x \leq 7$'. We regard the stable model along with a satisfying real-valued assignment as a solution to a logic program containing linear constraint atoms. For a more detailed introduction of ASP extended with linear constraints, illustrated with more complex examples, we refer the interested reader to [Janhunen et al., 2017].

To ease the use of ASP in practice, several extensions have been developed. First of all, rules with variables are viewed as shorthands for the set of their ground instances. Further language constructs include *conditional literals* and *cardinality constraints* [Simons et al., 2002]. The former are of the form a:$\mathtt{b}_1$,...,$\mathtt{b}_m$, the latter can be written as s{$\mathtt{d}_1$;...;$\mathtt{d}_n$}t, where a and $\mathtt{b}_i$ are possibly default-negated (regular) literals and each $\mathtt{d}_j$ is a conditional literal; s and t provide optional lower and upper bounds on the number of satisfied literals in the cardinality constraint. We refer to $\mathtt{b}_1$,...,$\mathtt{b}_m$ as a *condition*. The practical value of both constructs becomes apparent when used with variables. For instance, a conditional literal like a(X):b(X) in a rule's antecedent expands to the conjunction of all instances of a(X) for which the corresponding instance of b(X) holds. Similarly, 2{a(X):b(X)}4 is true whenever at least two and at most four instances of a(X) (subject to b(X)) are true. Finally, objective functions minimizing the sum of weights w_i subject to condition c_i are expressed as #minimize{w_1:c_1;...;w_n:c_n}.

In the same way, the syntax of linear constraints offers several convenience features. As above, elements in linear constraint atoms can be conditioned, viz. '&sum{$\mathtt{w}_1$*$\mathtt{x}_1$:$\mathtt{c}_1$;...;$\mathtt{w}_l$*$\mathtt{x}_l$:$\mathtt{c}_n$} <= k' where each $\mathtt{c}_i$ is a condition. Moreover, the theory language for linear constraints offers a domain declaration for real variables, '&dom{lb..ub} = x' expressing that all values of x must lie between lb and ub. And finally the maximization (or minimization) of an objective function can be expressed with &maximize{$\mathtt{w}_1$*$\mathtt{x}_1$:$\mathtt{c}_1$;...;$\mathtt{w}_l$*$\mathtt{x}_l$:$\mathtt{c}_n$} (by minimize). The full theory grammar for linear constraints over reals is available at https://potassco.org.

Highlights

ASP can be extended with linear constraint to counter its original weaknesses regarding the management of real numbers. Linear constraints can be added to the ASP model thanks to a theory grammar that will be interpreted by a LP-solver.

3.3 How to solve hybrid metabolic network completion in practice?

Inputs to the problem In this section, we present our hybrid approach to metabolic network completion. We start with a factual representation of problem instances. A metabolic network G with a typing function $t : M \cup R \rightarrow$ {d,r,s,t}, indicating the origin of the respective entities, is represented as follows:

$$\begin{aligned} F(G,t) = \; & \{\texttt{metabolite(}m\texttt{,}t(m)\texttt{)} \mid m \in M\} \\ & \cup \{\texttt{reaction(}r\texttt{,}t(r)\texttt{)} \mid r \in R\} \\ & \cup \{\texttt{bounds(}r\texttt{,}lb_r\texttt{,}ub_r\texttt{)} \mid r \in R\} \cup \{\texttt{objective(}r\texttt{,}t(r)\texttt{)} \mid r \in R\} \\ & \cup \{\texttt{reversible(r)} \mid r \in R, \mathit{reactants}(r) \cap \mathit{products}(r) \neq \emptyset\} \\ & \cup \{\texttt{rct(}m\texttt{,}s(m,r)\texttt{,}r\texttt{,}t(r)\texttt{)} \mid r \in R, m \in \mathit{reactants}(r)\} \\ & \cup \{\texttt{prd(}m\texttt{,}s(r,m)\texttt{,}r\texttt{,}t(r)\texttt{)} \mid r \in R, m \in \mathit{products}(r)\} \end{aligned}$$

While most predicates should be self-explanatory, we mention that `reversible` identifies bidirectional reactions. Only one direction is explicitly represented in our fact format. The four types `d`, `r`, `s`, and `t` tell us whether an entity stems from the **d**raft or **r**eference network, or belongs to the **s**eeds or **t**argets.

In a metabolic network completion problem, we consider a draft network $G = (R \cup M, E, s)$, a set S of seed compounds, a set R_T of target reactions, and a reference network $G' = (R' \cup M', E', s')$. An instance of this problem is represented by the set of facts $F(G,t) \cup F(G',t')$. In it, a key role is played by the typing functions that differentiate the various components:

$$t(n) = \begin{cases} \texttt{d}, & \text{if } n \in (M \setminus (T \cup S)) \cup (R \setminus (R_{S_b} \cup R_{obj})) \\ \texttt{s}, & \text{if } n \in S \cup R_{S_b} \\ \texttt{t}, & \text{if } n \in T \cup R_{obj} \end{cases} \qquad \text{and} \qquad t'(n) = \texttt{r},$$

where $T = \{m \in \mathit{reactants}(r) \mid r \in R_{obj}\}$ is the set of target compounds and $R_{S_b} = \{r \in R \mid m \in S_b\,(G), m \in \mathit{products}(r)\}$ is the set of reactions related to boundary seeds.

ASP encoding Our encoding of hybrid metabolic network completion is given in Listing 1. Roughly, the first 10 lines lead to a set of candidate reactions for completing the draft network. Their topological validity is checked in lines 12–16 with regular ASP, the stoichiometric one in lines 18–24 in terms of linear constraints. (Lines 1–16 constitute a revision of the encoding in [Schaub and Thiele, 2009b].) The last two lines pose a hybrid optimization problem, first minimizing the size of the completion and then maximizing the flux of the target reactions.

In more detail, we begin by defining the auxiliary predicate `edge/4` representing directed edges between compounds connected by a reaction. With it, we calculate in Line 4 and 5 the scope $\Sigma_G(S)$ of the **d**raft network G from the seed compounds in S; it is captured by all instances of `scope(M,d)`. This scope is then e**x**tended in Line 7/8 via the reference network G' to delineate all possibly producible compounds. We draw on this in Line 10 when choosing the reactions R'' of the completion (cf. Section 3.1) by restricting their choice to reactions from the reference network whose reactants are producible. This amounts to a topological search space reduction.

The reactions in R'' are then used in lines 12–14 to compute the scope $\Sigma_{G''}(S)$ of the **c**ompleted network. And R'' constitutes a topologically valid completion if all targets in T are producible by the expanded draft network G'': Line 16 checks whether $T \subseteq \Sigma_{G''}(S)$ holds, which is equivalent to $R_{obj} \subseteq \mathit{active}^t_{G''}(S)$. Similarly, R'' is checked for stoichiometric validity in lines 18–24. For simplicity, we associate reactions with their rate and let their identifiers take real values. Accordingly, Line 18 accounts for Equation 1.3 by imposing lower and upper bounds on each reaction rate. The mass-balance equation 3.1 is enforced for each metabolite

```
 1 edge(R,M,N,T) :- reaction(R,T), rct(M,_,R,T), prd(N,_,R,T).
 2 edge(R,M,N,T) :- reaction(R,T), rct(N,_,R,T), prd(M,_,R,T), reversible(R).

 4 scope(M,d) :- metabolite(M,s).
 5 scope(M,d) :- edge(R,_,M,T), T!=r, scope(N,d):edge(R,N,_,T’), N!=M, T’!=r.

 7 scope(M,x) :- scope(M,d).
 8 scope(M,x) :- edge(R,_,M,_), scope(N,x):edge(R,N,_,_), N!=M.

10 { completion(R) : edge(R,M,N,r), scope(N,x), scope(M,x) }.

12 scope(M,c) :- scope(M,d).
13 scope(M,c) :- edge(R,_,M,T), T!=r, scope(N,c):edge(R,N,_,T’), T’!=r, N!=M.
14 scope(M,c) :- completion(R), edge(R,_,M,r), scope(N,c):edge(R,N,_,r), N!=M.

16 :- metabolite(M,t), not scope(M,c).

18 &dom{L..U} = R :- bounds(R,L,U).

20 &sum{ IS*IR : prd(M,IS,IR,T), T!=r;  IS’*IR’ : prd(M,IS’,IR’,r), completion(IR’);
21      -OS*OR : rct(M,OS,OR,T), T!=r; -OS’*OR’ : rct(M,OS’,OR’,r), completion(OR’)
22     } = "0" :- metabolite(M,_).

24 &sum{ R } > "0" :- reaction(R,t).

26 &maximize{   R : objective(R,t) }.
27 #minimize{ 1,R : completion(R)  }.
```

Listing 1: *Encoding of hybrid metabolic network completion*

`M` in lines 20–22; it checks whether the sum of products of stoichiometric coefficients and reaction rates equals zero, viz. `IS*IR`, `-OS*OR`, `IS’*IR’`, and `-OS’*OR’`. Reactions `IR`, `OR` and `IR’`, `OR’` belong to the draft and reference network, respectively, and correspond to $R \cup R''$. Finally, by enforcing $r_{obj} > 0$ for $r_{obj} \in R_{obj}$ in Line 24, we make sure that $R_{obj} \subseteq active^{s}_{G''}(S)$.

In all, our encoding ensures that the set R'' of reactions chosen in Line 10 induces an augmented network G'' in which all targets are activated both topologically as well as stoichiometrically, and is optimal with respect to the hybrid optimization criteria.

Highlights

The implementation of the hybrid gap-filling model uses basis from the Meneco encoding and extends it with linear constraints satisfying the constraints of Flux Balance Analysis. It ensures that the solution respects both topological and stoichiometric activations.

3.4 Benchmarking hybrid gap-filling

In this section, we introduce Fluto, our new system for hybrid metabolic network completion, and empirically evaluate its performance. The system relies on the hybrid encoding described in Section 3.3 along with the hybrid solving capacities of *clingo* [Gebser et al., 2016a] for implementing the combination of ASP and LP. We use *clingo* 5.2.0 incorporating as LP solvers either *cplex* 12.7.0.0 or *lpsolve* 5.5.2.5 via their respective Python interfaces. We describe the details of the underlying solving techniques in a separate paper [Janhunen et al., 2017] and focus below on application-specific aspects.

The output of Fluto consists of two parts. First, the completion R'', given by instances of predicate `completion`, and second, an assignment of floats to (metabolic flux variables v_r for) all $r \in R \cup R''$. In our example, we get

$$\begin{aligned} R'' &= \{\texttt{completion}(r_6), \texttt{completion}(r_8), \texttt{completion}(r_9)\} \\ \text{and } &\{r_{s_1} = 49999.5, r_9 = 49999.5, r_3 = 49999.5, r_2 = 49999.5, \\ &\quad r_e = 99999.0, r_6 = 49999.5, r_5 = 49999.5, r_4 = 49999.5\}. \end{aligned}$$

Variables assigned 0 are omitted. Note the flux value $r_8 = 0$ even though $r_8 \in R''$. This is to avoid the self-activation of cycle C, D and E. By choosing r_8, we ensure that the cycle has been externally initiated at some point, but activation of r_8 is not necessary at the current steady state.

We analyze (i) the quality of Fluto's approach to metabolic network completion, and (ii) we compare the quality of Fluto's solutions with other approaches. To have a realistic setting, we re-used the gap-filling benchmark of the previous chapter that was created to test Meneco and was published in [Prigent et al., 2017]. We use degradations of a functioning metabolic network of *Escherichia coli iJR*904 [Reed et al., 2003] comprising 1075 reactions. The network was randomly degraded by 10, 20, 30 and 40 percent, creating 10 networks for each degradation by removing reactions until the target reactions were inactive according to *Flux Variability Analysis* [Becker et al., 2007]. 90 target reactions with varied reactants were randomly chosen for each network, yielding 3600 problem instances in total [Prigent et al., 2017]. The reference network consists of reactions of the original metabolic network.

We ran each benchmark on a Xeon E5520 2.4 GHz processor under Linux limiting RAM to 20 GB. At first, we investigate two alternative optimization strategies for computing completions of minimum size. The first one, *branch-and-bound* (BB), iteratively produces solutions of better quality until the optimum is found and the other, *unsatisfiable core* (USC), relies on successively identifying and relaxing unsatisfiable cores until an optimal solution is obtained. Note that we are not only interested in optimal solutions but if unavailable also solutions activating target reactions without trivially restoring the whole reference network. In *clingo*, BB naturally produces these solutions in contrast to USC. Therefore, we use USC with stratification [Ansótegui et al., 2013], which provides at least some suboptimal solutions. We will focus here on the functionality of models gap-filled by Fluto and compare it to the two other parsimonious approaches Meneco and GapFill.

3.4.1 Quality of models gap-filled with Fluto

Impact of solver options on the gap-filling of variously degraded models Now, we examine the quality of the solutions provided by Fluto. Table 3.1 gives the number of optimal and suboptimal obtained by Flutoin its default setting within 20 minutes for BB, USC and the best of both (BB+USC), individually for each degradation and overall. Each obtained best solution was checked for stoichiometric activation of the objective reaction with *cobrapy* 0.3.2 [Ebrahim et al., 2013], that implements FBA tests. As expected all solutions found by Fluto passed the verification test with *cobrapy*, which validates that the combination of LP constraints and ASP is functional and fits our objective. Note that default settings for Fluto include the default configurations for *clingo* and *cplex*. The data was obtained in using networks with 10, 20, and 30 percent degradation. For 94.3% of the instances Fluto(BB+USC) found a solution within the time limit and 82.3% of them were optimal. We observe that BB provides overall more useful solutions but USC acquires more optima, which was to be expected by the nature of the

Table 3.1: *Comparison of qualitative results for Fluto under several solver configurations. Number of optimal solutions and suboptimal ones obtained by Fluto in its default setting within 20 minutes for* BB*,* USC *and the best of both (*BB+USC*), individually for each percentage of degradation and overall. Altogether, 94.3% of the instances Fluto(*BB+USC*) found a solution within the time limit and 82.3% of them were optimal.*

		Branch-and-bound		Unsatisfiable core		Best of BB and USC	
		BB		USC			
Degradation rate	Number of instances	optimal solutions	solutions	optimal solutions	solutions	optimal solutions	solutions
10%	900	900	900	892	892	900	900
20%	900	669	830	769	793	814	867
30%	900	88	718	344	461	382	780
overall	2,700	1,657 (61.4%)	2,448 (90.7%)	2,005 (74.3%)	2,146 (79.5%)	2,096 (77.7%)	2,547 (94.3%)

Table 3.2: *Comparison of Fluto and Meneco solutions for 10 percent degraded networks. All solutions to each instance were enumerated and individually tested using FBA.*

	Fluto			Meneco		
	min	average	max	min	average	max
solutions per instance	1	2.24	12	1	1.88	6
reactions per solution	1	6.66	9	1	6.24	9
verified solutions			100%			73.39%
instances with only verified solutions			100%			72.94%
instances without verified solutions			0%			26.61%
instance with some verified solutions			0%			0.45%

optimization techniques. Additionally, each technique finds solutions to problem instances where the other exceeds the time limit, underlining the merit of using both in tandem. In detail, Fluto found a smallest set of reactions completing the draft network for 77.7% in 20 minutes, a suboptimal solution for 16.7%, and no solution for 5.6% of the problem instances.

Highlights

For 94.3% of the degraded *E. coli* instances Fluto found a solution within the 20 minutes time limit and 77.7% of them were optimal. This means that Fluto performs well when using a database of reactions of a reasonable size.

3.4.2 Comparison to Meneco and GapFill

Enumeration of solutions with Fluto and Meneco We compare the quality of Fluto with Meneco 1.4.3 [Prigent et al., 2017] that uses topological activation $active^t_G(S)$ and was tested in the previous chapter; and GapFill that uses a relaxed constraint-based activation $active^r_G(S)$.

Table 3.3: *Comparison of Fluto, Meneco and GapFill unions for 10 percent degraded networks The union of solutions were computed and tested using FBA. Unverified solutions means individual solutions that never satisfy the constraints of FBA. Verified solutions verify these constraints and partially verified solutions means that some individual solutions satisfy them within the enumeration but not all of them.*

	Fluto	Meneco	GapFill
verified union	100%	73.39%	6.20%
verified union of verified solutions	100%	72.94%	NA
verified union of unverified solutions	0%	0.00%	NA
verified union of partially verified solutions	0%	0.45%	NA

[1] [Satish Kumar et al., 2007]. [2] Both Meneco and GapFill are systems for metabolic network completion. While Meneco pursues the topological approach, GapFill applies the relaxed stoichiometric variant using Inequation 3.3. We performed an enumeration of all minimal solutions to the completion problem under the topological (Meneco), the relaxed stoichiometric (GapFill), and hybrid (Fluto) activation semantics for the 10 percent degraded networks of the benchmark set (900 instances to be completed).

First, we compare the quality of individual solutions of Fluto and Meneco. [3] Results are displayed in Table 3.2. The first two rows give the minimum, average and maximum number of solutions per instance, and reactions per solution, respectively, for Fluto and Meneco. While Fluto finds 19% more solutions on average and twice as many maximum solutions per instance compared to Meneco, the numbers of reactions in minimal solutions of both tools are similar. The next four rows pertain to the solution quality as established by *cobrapy*. First, what percent solutions over all instances could be verified, second, what percent of instances had verified solutions exclusively, third, how many instances had no verified solutions at all, and finally, percent of instances where only a portion of solutions could be verified. All of Fluto's solutions could be verified, compared to the 72.04% of Meneco across all solutions and 72.94% of instances that were correctly solved. Interestingly, Meneco achieves hybrid activation in some but not all solutions for 0.45% (4) of the instances. Fluto does not only improve upon the quality of Meneco, but also provides more solutions per instances without increasing the number of relevant reactions significantly.

Union of solutions to replace enumeration To empirically evaluate the properties established in Section 3.1.4, and be able to compare to GapFill, for which only the union of reactions was available, we examine the union of minimal solutions provided by all three systems and present the results in Table 3.3. The four rows show, first, for what percent of instances the union of solutions could be verified, second, how many instances had only verified solutions and their union was also verified, third, the percentage of instances where the union of solutions displayed activation of the target reactions even though all individual solutions did not provide that, and forth, instances where the solutions were partly verifiable and their union could also be verified. While again 100% of Fluto's solutions could be verified, only 73.3% and 6.2% are obtained for Meneco and GapFill, respectively, for 10 percent degraded networks. As reflected by the results, the ignorance of Meneco regarding stoichiometry leads to possi-

[1] Update of 2011-09-23 see http://www.maranasgroup.com/software.htm

[2] The results for Meneco and GapFill are taken from previous work [Prigent et al., 2017], where they were run to completion with *no* time limit.

[3] There was no data available for the individual solutions of GapFill.

bly unbalanced networks. Still, the union of solutions provided a useful set of reactions in almost three quarters of the instances, showing merit in the topological approximation of the metabolic network completion problem. Interestingly, although we demonstrated in Subsection 3.1.4 and Figure 3.7 that the union of individually no-flux solutions can carry flux, the results on Meneco show that this case concern very few instances (0.45%). On the other hand, the simplified view of GapFill in terms of stoichiometry misguides the search for possible completions and eventually leads to unbalanced networks even in the union. Moreover, GapFill's ignorance of network topology results in self-activated cycles. By exploiting both topology and stoichiometry, Fluto avoids such cycles while still satisfying the stoichiometric activation criteria. The results support the observations made in Section 3.1.4. For both Fluto and Meneco all instances, for which the complete solution set could be verified, the union is also verifiable, as well as all unions for instances where Meneco established hybrid activation for a fraction of solutions.

Highlights

As expected, Fluto performs better than Meneco and GapFill and enables to take the most out of the two formalisms for gap-filling metabolic models that respect topological and stoichiometric activation of the objective reactions.

Conclusion

We presented the **first hybrid approach to metabolic network completion** by combining graph-based and constraint-based formalisms in a uniform setting to get the most out of the two of them. To this end, we elaborated a formal framework capturing different semantics for the activation of reactions. Based upon these formal foundations, we developed a hybrid ASP encoding reconciling disparate approaches to network completion. The resulting system, Fluto, thus combines the advantages of both approaches and yields superior results compared to purely quantitative or qualitative existing systems. Our experiments show that Fluto scales to more highly degraded networks than graph-based gap-filling and produces useful solutions in reasonable time. In fact, all of Fluto's solutions passed the biological gold standard that is Flux Balance Analysis. The exploitation of the network's topology guides the solver to more likely completion candidates, and furthermore avoids self-activated cycles, as possibly obtained in constraint-based approaches. Also, unlike other systems, Fluto allows for establishing optimality and addresses the strict stoichiometric completion problem without approximation.

Fluto takes advantage of the **hybrid reasoning capacities of the ASP system *clingo*** for extending logic programs with linear constraints over reals. This provides us with a practically relevant application scenario for evaluating this hybrid form of ASP.

From the computational biology perspective, hybrid gap-filling can be viewed as an intermediary step between the graph-based gap-filling - that might not satisfy stoichiometric constraints but scales very efficiently - and the final refinements to be made to the GSMs. Such an application of the method will be presented in the next chapter. As a gap-filling method that satisfies the strict steady-state condition of the constraint-based semantics, there can be cases in which Fluto does not find a satisfiable solution due to the accumulation of some metabolites and the absence of reactions in the database to eliminate them. The possibility to create exports of such metabolites has been implemented into Fluto. Default setting does not enable them but they can be configured if needed.

Chapter 4

Applications of gap-filling to non-model organisms

The present chapter focuses on biological applications of the previous two chapters. Firstly I describe the role of gap-filling in the reconstruction of a Genome-Scale Model (GSM) for a Non-Model Organism (NMO), a work presented in *PLOS Computational Biology* [Aite et al., 2018]. I will show how graph-based methods were used into the process, and why they are of interest in the field. I then detail a use-case of hybrid gap-filling for *Chondrus crispus* that fits in the general purpose of model refinement adressed by this hybrid technique. I will finish this chapter by describing the application of graph-based gap-filling methods to evaluate the cooperation potential between *Ectocarpus siliculosus* and a frequently associated bacterium of this alga: *Candidatus* Phaeomarinobacter ectocarpi. The latter work was published in *PLOS Computational Biology* [Prigent et al., 2017]

4.1 Graph-based and hybrid gap-filling: assets in GSM reconstruction processes for non-model organisms

The reconstruction of *Ectocarpus siliculosus*'s GSM was partly extracted from the paper I, Méziane Aite, Marie Chevallier, Camille Trottier (four first coauthors) and others coauthored, published in ***PLOS Computational Biology*** and entitled *Traceability, reproducibility and wiki-exploration for "à-la-carte" reconstructions of genome-scale metabolic models* [Aite et al., 2018].

Gap-filling is placed between the automatic generation of draft GSMs and their final refinements by experts who curate false positive reactions and add literature information in the model. The pipelines for GSM reconstruction vary a lot because their purpose is to catch and exploit all available information for producing the highest quality models. Yet, a step of gap-filling is recurrent among these multiple pipelines as automatic methods never catch the entirety of an organism's metabolism. This section relies on the reconstruction of the second version of *Ectocarpus siliculosus*'s GSM using a workspace (AuReMe) that integrates multiple tools for ensuring traceability and reproducibility in GSMs [Aite et al., 2018].

Ectocarpus siliculosus is a model for the study of brown algae. Its genome has been sequenced in 2010 [Cock et al., 2010] and its GSM, EctoGEM, was reconstructed four years later [Prigent et al., 2014]. This first version contained 1,866 reactions and 2,020 metabolites. The re-annotation of the genome [1] motivated a new version of the model in [Aite et al., 2018]. The newest version contains 1,977 reactions and 2,132 metabolites and is updated to a more recent version of the MetaCyc database.

4.1.1 Reconstruction protocol of EctoGEM v2

The reconstruction of EctoGEM v2 was tight to the definition of a biomass reaction (30 reactants) and a set of target metabolites (biomass reactants and additional compounds known to be produced by the seaweed). EctoGEM v2 (referred to as EctoGEM in the following) was built using the second version of its genome annotation and orthology with the plant model *Arabidopsis thaliana*.

Figure 4.1 depicts the reconstruction process. It involves several tools and steps that needs to be fully documented to keep tracks of the modifications made to the model. Annotation-based reconstruction with Pathway Tools [Karp et al., 2016] was done in parallel to the orthology based one [Loira et al., 2015]. Table 4.1 depicts the characteristics of the sub-model obtained at each step of the reconstruction. A large majority of the model's final reactions are found using annotation. Yet the impact on functionality is low: only 5 targets are producible (graph-based and constraint-based criteria) out of the 50. This means that there exist gaps in the models that annotation does not allow to fill.

The models resulting from the annotation and orthology are merged. They share a lot of their reactions thus having a small impact on the size of the combined model. The functionality of the merged model remains the same as the one of the annotation-based model.

[1] http://gem-aureme.irisa.fr/ectogem

Table 4.1: *Added value of each GSM reconstruction step to the quality of the model. The GSM of* E. siliculosus *was reconstructed using the annotation of its genome and orthology with the plant model* A. thaliana. *For each step of the pipeline, qualitative descriptions are provided.*

Organism GSM	Subnetworks of the reconstruction pipeline	number of metabolites	number of reactions	number of genes	ratio of reactions associated to genes	Number of biological compounds of interest (targets)	number of topologically producible targets	number of flux-activated targets	FBA growth rate
E. siliculosus	annotation-based network	2118	1834	2281	83%	50	5	5	0
	orthology with A. thaliana	650	442	593	89%		1	0	0
	merging annotation and orthology-based models	2118	1887	2281	83%		5	5	0
	Final network after gap-filling and manual curation	2132	1977	2281	79%		50	50	3.02

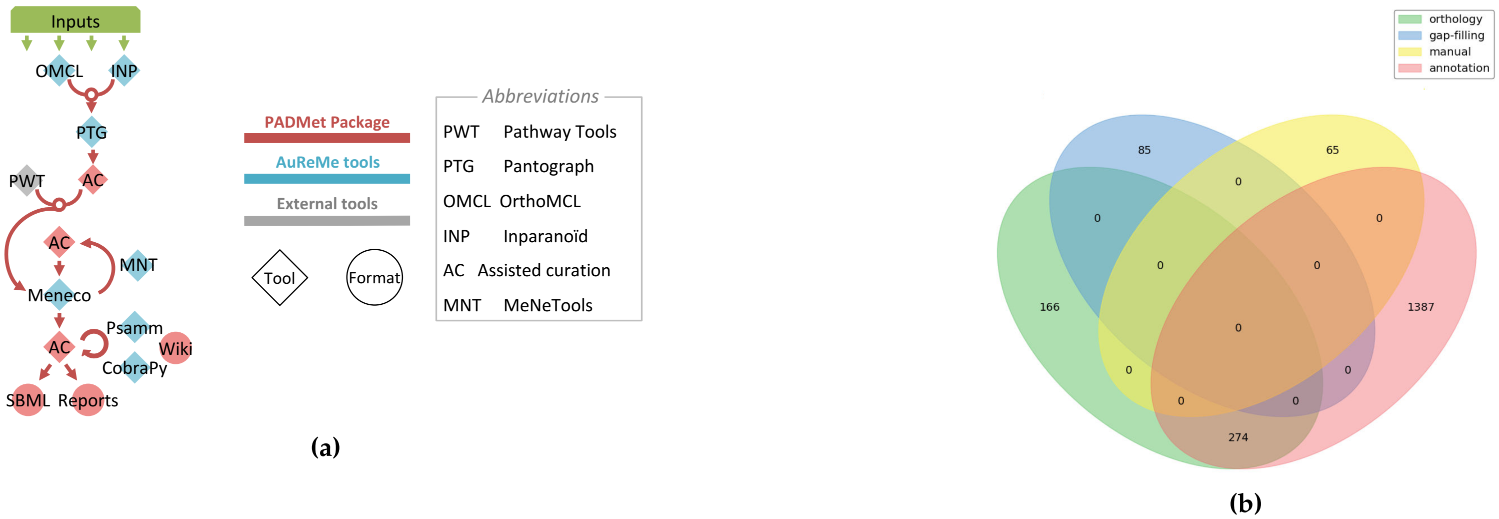


Figure 4.1: ***Reconstruction of EctoGEM v2***

(a) Reconstruction pipeline of EctoGEM. The GSM of Ectocarpus siliculosus *was reconstructed using Pathway Tools [Karp et al., 2002, Karp et al., 2016], the annotation-based reconstruction tool associated to the MetaCyc database. Parallely, orthology using Pantograph [Loira et al., 2015] was performed to catch reactions of* Arabidopsis thaliana*'s model that are associated to genes orthologous to the algal ones. The two draft GSMs were combined into a merged one. The latter was then gap-filled with* Meneco*. Manual refinements to the model were then performed, together with a monitoring of the the graph-based producibility of EctoGEM, using the topology analysis package MeNeTools (detailed in Chapter 7) and other tools dedicated to the constraint-based analysis of GSMs (CobraPy [Ebrahim et al., 2013], PSAMM [Steffensen et al., 2016]). As a last stage, the final supports of the model were produced: SBML files [Hucka et al., 2003], reports on the model contents and a wiki (http://gem-aureme.irisa.fr/ectogem) for browsing the model. (b) Origin of the reactions in EctoGEM. Venn diagram representing the origin of the 1977 reactions in EctoGEM: orthology with* A. thaliana*, annotation, gap-filling or manual curation. The last category includes reactions that were added for modeling reasons such as import reactions for the growth medium.*

Gap-filling with Meneco enables to restore the producibility of the targets through adding 85 reactions to the model (4.3% of the size of the GSM).

> *Highlights*
>
> Despite the low number of reactions added by the gap-filling step, its impact on the functionality of the model is large. For *E. siliculosus*, gap-filling added the equivalent of 4.3% of the size of the model and unblock the producibility of 90% of the targets.

4.1.2 Multiple sources of information and gap-filling recover complete pathways

Once the reconstruction is achieved, each pathway of the model can be compared to the full pathway in the database (MetaCyc) to obtain completion rates and study possibly missing reactions. This is another way, static, to explore functionality. When doing such analyses on EctoGEM, we noted the impact of gap-filling and the other reconstruction steps on the completion of pathways.

The benefit of combining orthology, annotation and gap-filling is also noticeable when analyzing the metabolic pathways. Complementary methods that exploit all available data can retrieve several reactions from pathways, resulting in the reconstruction of complete or near exhaustive pathways. The wiki page [1] associated with a given pathway describes all the methods of the pipeline providing evidence for the presence of each pathway reaction in the GSM and allows browsing databases to search for evidences in other species. An example of such pathway completion can be observed during the reconstruction process of EctoGEM (Figure 4.2).

The 6-hydroxymethyl-dihydropterin diphosphate biosynthesis I (PWY-6147) and the following tetrahydrofolate biosynthesis pathway (PWY-6614) identified in algal metabolism include in total eight reactions leading to the production of a necessary metabolite, a tetrahydrofolate (THF-GLU-N) starting from GTP. In this example (Figure 4.2), the need to combine approaches is illustrated by the functional characterization of the pathway after each step of the selected pipeline. Genome-annotation and orthology-based tools identified respectively 3 and 4 reactions of these pathways, including one that was identified by both. Combining information is an essential step leading to a partial reconstruction of the pathways (7/8 reactions) in the merged model. As mentioned on the MetaCyc website, the database groups reactions and metabolites into classes using an ontology tree structure. In our example, a 7,8 dihydrofolate (DIHYDROFOLATE-GLU-N) is a metabolite class comprising subclasses and instances. The 7,8-dihydrofolate monoglutamate (dihydrofolate) compound is one of these instances. Originally, reaction DIHYDROFOLATE-SYNTH-RXN produces DIHYDROFOLATE while the following reaction in the pathway consumes DIHYDROFOLATE-GLU-N. Performing gap-filling with an extended version of the MetaCyc database (provided in the file *metabolic-reactions.sbml* of the database) allows to retrieve an instantiated version of the DIHYDROFOLATEREDUC-RXN (namely DIHYDROFOLATEREDUCT-RXN-THF/NADP//DIHYDROFOLATE/NADPH/PROTON.37.) that takes DIHYDROFOLATE as a reactant. This enables to restore the producibility of the final compounds of the pathway starting from its inputs. A second reaction is added by gap-filling in PWY-6147: H2NEOPTERINALDOL-RXN. Application of these heterogeneous methods allows the completion of the entire dihydrofolate biosynthesis pathway from GTP to be present in EctoGEM as it is

[1] http://gem-aureme.irisa.fr/ectogem/index.php/Category:Pathway

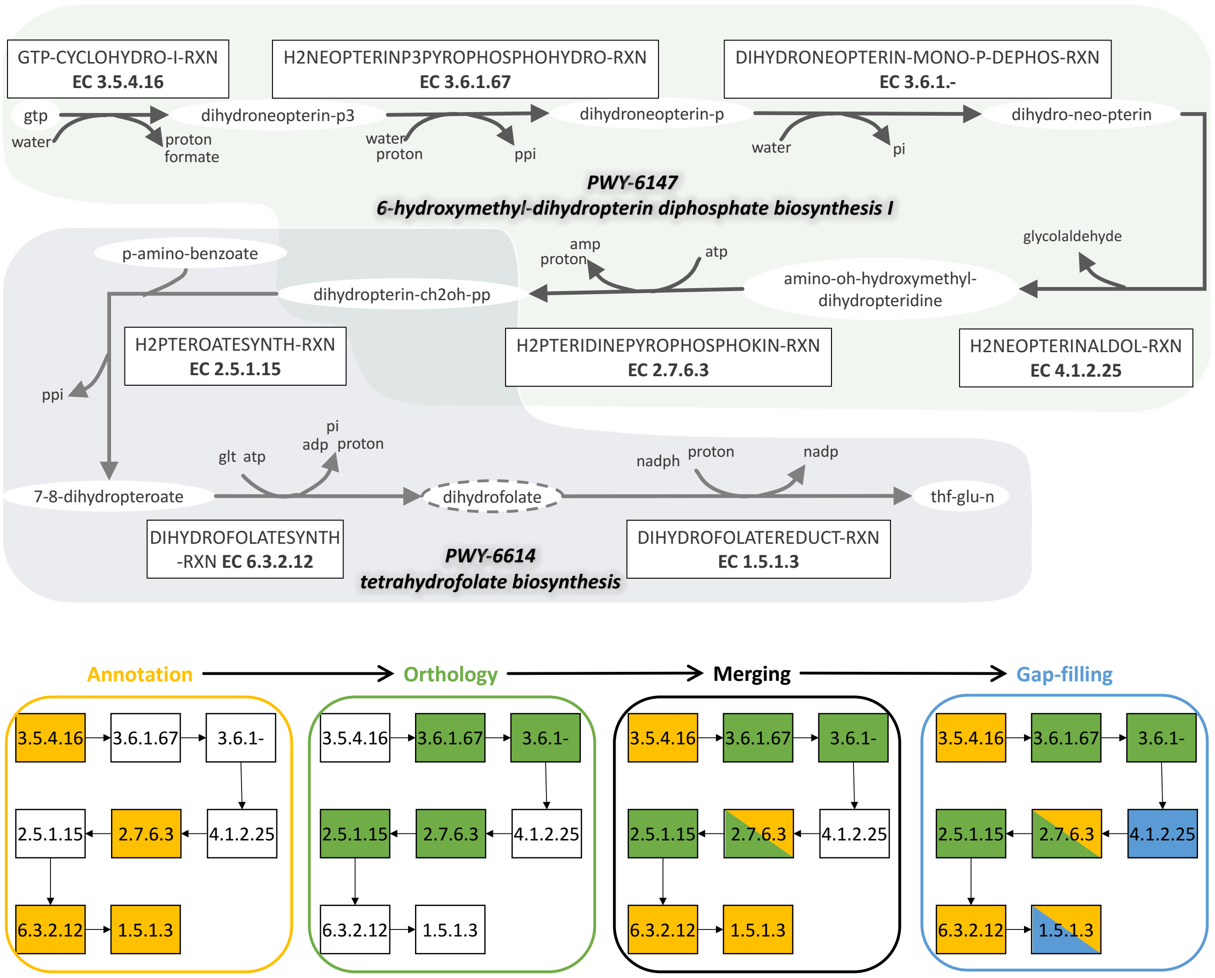


Figure 4.2: ***Interest of heterogeneous methods in pathway completion and filling thanks to tracking of process metadata***

Completion of the 6-hydroxymethyl-dihydropterin diphosphate biosynthesis I and the tetrahydrofolate biosynthesis pathways in E. siliculosus via the combination of annotation (yellow), orthology (green) and gap-filling (blue). The dihydrofolate compound with the dotted line is an instance of the DIHYDROFOLATE-GLU-N *class, following MetaCyc classes ontology structure. The class compound is the original reactant of the* DIHYDROFOLATEREDUCT-RXN *reaction retrieved with annotation, whereas the previous reaction of the pathway (*DIHYDROFOLATESYNTH-RXN*) produces the instance* DIHYDROFOLATE. *Hence the gap-filling step that, using an extended version of MetaCyc, selects an instantiated version of* DIHYDROFOLATESYNTH-RXN *that consumes the instance* DIHYDROFOLATE.

described in the MetaCyc database.

Highlights

Studying the coverage of pathways during GSM reconstruction is helpful to monitor the evolution of the model. In particular, it sheds light on the interest of using several sources of information and methods, among which the gap-filling has an important place.

4.1.3 Impact of the gap-filling step in the reconstruction process

The reconstruction of EctoGEM v2 was achieved using several steps and methods, among which the graph-based gap-filling performed with Meneco. The functionality of models can be assessed either dynamically using graph-based or constraint-based criteria for defining producibility, or in a static way by studying the global topology of the model and particularly the existence and completion of metabolic pathways.

85 reactions were added by gap-filling, which represents 4.3% of the size of the final model. The balance between manually removed and manually added reactions is positive: 65 reactions belong to the latter category in the final model. Yet, a large majority of them (56/65) consist in artefact and artificial reactions whose purpose is to model the import of metabolites in the boundaries and the extracellular compartment of the system. The addition of the biomass reaction of EctoGEM is also considered as manual curation.

We observe that the impact of gap-filling is crucial, despite the enhancement of *Ectocarpus siliculosus*'s annotation since the first version of the GSM. Indeed, the functionality of the model through the producibility of the various targets and the biomass is at a very low rate (5/50) prior to gap-filling, whereas it is fully functional afterwards. The acquisition of the union of parsimonious solutions with Meneco enables to cover the space of solutions and it can be filtered *a posteriori* to remove false positive reactions, as it has been done with EctoGEM during manual curation. Moreover, another asset of graph-based gap-filling is the possibility to use not only a reaction (the biomass function) but also scattered additional compounds as targets. This widens the possible objectives to be given to gap-filling and enhances its flexibility.

Highlights

Altogether the reconstruction of EctoGEM v2 demonstrates the interest of graph-based gap-filling in practice, for the reconstruction of GSMs for NMOs through its good results on functionality restoration but also on the static aspect of pathway completion.

4.1.4 Application of hybrid gap-filling to *Chondrus crispus*

Once a model is reconstructed and is of satisfying quality, it can be use as a template for helping reconstructing GSMs of more or less taxonomically related species. This is the case here for *Ectocarpus siliculosus* that was used to enhance the GSM of *Chondrus crispus* using hybrid gap-filling with Fluto. Fluto is a tool that provides minimal sets of reactions such that the completed model satisfies both graph-based (scope) and constraint-based (FBA) activation of its objective reaction(s). Fluto has been described in Chapter 3.

Reconstruction of *Chondrus crispus*'s metabolic model *Chondrus crispus* is a red alga with industrial interests for instance in food through its production of carrageenans. Its genome was sequenced in 2013 by [Collén et al., 2013]. The following case-study was performed with Enora Fremy and Méziane Aite in the project of reconstructing the GSM of *C. crispus* in collaboration with Gabriel Markov (Station Biologique de Roscoff). CcrGEM[1] is the name of the model. It was reconstructed using the classical AuReMe pipeline [Aite et al., 2018]: merging of annotation-based and orthology-based (with EctoGEM v2 as a template) models, gap-filling with Meneco. The model has 85 targets, among which 30 are reactants of the biomass reaction. After the gap-filling step, 29 reactants of the biomass are producible under the constraint-based semantics. L-α-alanine (called alanine later in this section) is the only reactant in which flux does not flow. Consequently the model requires at least one additional reaction in order to produce alanine or to restore the balance of the fluxes within alanine's production pathway through degradation reactions for instance.

Hybrid gap-filling to unblock the constraint-based producibility of alanine Tests were performed using EctoGEM: applying the biomass reaction of CcrGEM into the brown alga model led to a positive flux into alpha-alanine. Thus the missing reaction(s) in CcrGEM can be found in Ectogem and added to CcrGEM in order to restore flux. This is a typical gap-filling step that can be performed with Fluto given since the constraints to be restored are flux-based.

Fluto was run with CcrGEM (at the post Meneco gap-filling stage) as a draft model and EctogGEM as the repair database. The minimal size of the solution is one reaction and several solutions exist. The chosen one is ALANINE-DEHYDROGENASE-RXN that consumes alanine, NAD and water and produced pyruvate, ammonium, NADH and H^+. The selected reaction was associated to the gene CHC_T00008930001 based on sequence and proteic domains comparison with *Saccharina japonica* and *Ectocarpus siliculosus*. The analysis of this reaction and the other possible solutions that could be proposed by Fluto confirm that the blocked production of alanine was caused by an accumulation of metabolites in the model. In the chosen reaction the alanine is degraded, enabling to balance the fluxes. Other solutions degrade alanine with different metabolites as coreactants or degrade other substrates upstream in the alanine biosynthesis pathways.

Analysis of essential reactions for alanine production FVA performed on the model after addition of the missing reaction enables to classify the reactions of CcrGEM according to their status with respect to the production of alanine. Only 16 reactions are essential. On the other hand, 462 reactions are alternative, meaning that there are numerous synthesis pathways that

[1] http://gem-aureme.irisa.fr/ccrgem/index.php/

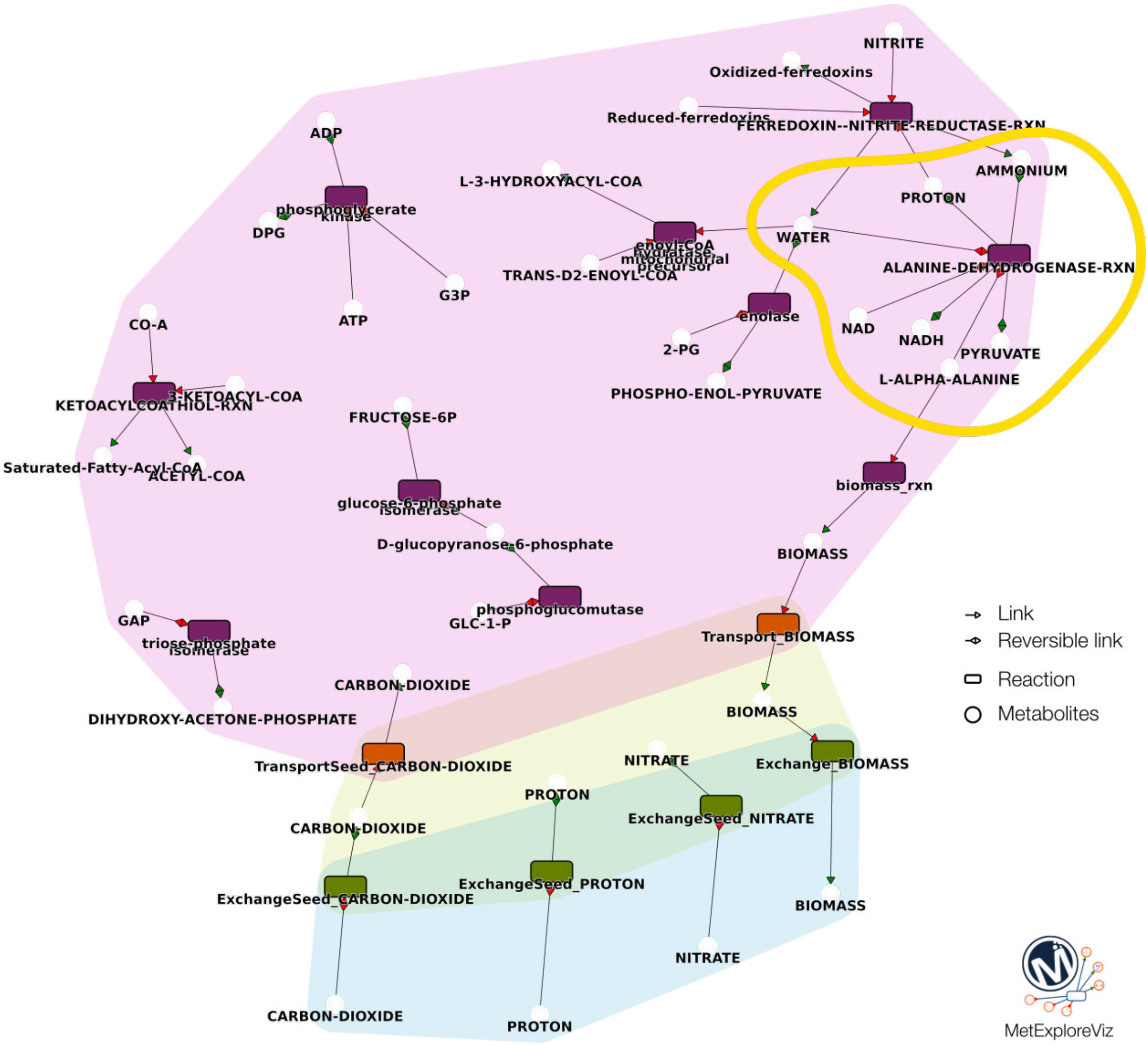


Figure 4.3: ***Essential reactions for the production of L-α-alanine in Chondrus crispus GSM (CcrGEM)***
The GSM of the red alga was reconstructed using annotation, orthology and graph-based gap-filling but missed one reaction to produce alanine under a constraint-based modeling. reaction ALANINE-DEHYDROGENASE-RXN (yellow outline) was retrieved using hybrid gap-filling (Fluto) and associated to a gene. The essential reactions for producing alanine are displayed along with the compartments of the metabolites they involved. Cytosol is displayed in pink, extracellular in green and the boundary compartment in blue. There are 462 reactions in the model that are alternative for alanine production, but not displayed here. The graph was obtained using Metexplore [Cottret et al., 2018] and MetExploreViz [Chazalviel et al., 2017]

coexist in the model to produce this amino acid. Figure 4.3 describes the essential reactions for alanine production. The biomass reaction of CcrGEM was altered to possess only one reactant, the alanine, with its stoichiometry unchanged (26.6) in order to capture the adequate classification of reactions in FVA. Among the 16 essential reactions, 6 are artificial ones set up to model the growth medium. We observe with this 6 reactions that carbon dioxide, nitrate and protons are the only seeds that occur in every production path of alanine, having in mind that there might be other alternative ones. Notably the transport reactions between the extracellular space and the cytosol for the protons and nitrate are not essential, they do not appear on the figure. This means that there exist transport reactions of the model, that probably transport other metabolites along with protons and/or nitrate; and these reactions can be used in the production paths of alanine. The biomass reaction is also essential, as expected since it is the objective function. The reaction added by Fluto for flux balance and alanine degradation is also essential, as expected since flux is broken in its absence. It is outlined in yellow on Figure 4.3. Globally, the essential cytosolic reactions in alanine production pathway of CcrGEM are scaterred and not chained one to each others (except with universal metabolites such as water, protons or ammonium). This makes understandable the high number of alternative reactions found in FVA: there are many "holes" in the pathway between the essential reactions, hence large combinatorics.

Highlights

This case-study demonstrates the interest of hybrid gap-filling in practice. It does not aim to replace the topological gap-filling that can restore flux in some targets by itself. On the contrary it aims to refine the gap-filling by unblocking the remaining targets, by using specialized databases or template GSMs for close organisms as repair databases. By doing so, it facilitates the understanding of some production pathways of interest in the considered GSM.

4.2 Gap-filling as a means to study host-symbiont systems

Parts of this section's text and figures were extracted from the paper Sylvain Prigent, I, and others coauthored in ***PLOS Computational Biology*** entitled "*Meneco, a topology-based gap-filling tool applicable to degraded genome-wide metabolic networks*" [Prigent et al., 2017].

Gap-filling is performed most of the times on a universal database of reactions or one that is dedicated to the type of organism that is considered (plant, bacterium etc.). Yet, having in mind that organisms do not live in isolation, it seems natural to consider symbiont organisms as metabolic suppliers for their hosts. In this direction, it is relevant to examine the gap-filling as a way to identify putative host-symbionts interactions, always in light of expert knowledge and curation of hypotheses. Here we present the application of Meneco to *Ectocarpus siliculosus* using the GSM of its symbiont *Ca.* P. ectocarpi as a database of reactions.

4.2.1 The bacterial GSM supports the algal GSM for target producibility

The Meneco tool was applied to the analysis of candidate algal-bacterial interactions in the brown algal model *E. siliculosus*. The bacterium used for this study is *Candidatus* Phaeomarinobacter ectocarpi, that was sequenced together with the alga [Cock et al., 2010] and whose GSM was reconstructed [Dittami et al., 2014a]. The analysis pipeline is summarized in Figure 4.4. We considered, as a fixed set of seeds, molecules found in the seawater medium and commonly used to grow the algae [Prigent et al., 2014]. In parallel, we designed a large set of targets by identifying all reactions in MetaCyc 19.0 for which at least one of the associated genes was supported by the presence of one or more expressed sequence tag (EST) within the dataset of 90,637 ESTs published in the original *Ectocarpus* genome paper [Cock et al., 2010]. For each of these expressed reactions, we considered that both reactants and products should be available in the network and defined them as targets. This formed a set of 1,125 metabolites. Using Meneco we then identified which of these targets were producible in the alga using the bacterial GSM as a reactions database. The GSM used for the alga was the first version of EctoGEM without any gap-filling in order to ensure that every reaction in the model was supported by genetic information (*i.e.* no false positive reactions). This allowed the identification of the set of targets that became producible when adding the metabolic capacities of the *Ca.* P. ectocarpi model to the algal one. These targets and the metabolic pathways enabling their production were then manually examined with the help of Simon Dittami who has a biological expertise on the studied alga. In total 83 previously non-producible algal targets became producible according to the Meneco semantics when combining the metabolic capacities of the bacterium with those of the alga.

Highlights

By setting as metabolic targets the metabolites involved in reactions derived from transcriptomics data, it is possible to explore the part of *E. siliculosus*'s metabolism that is activated in given conditions. Complementarity between the model of the alga and *Ca.* P. ectocarpi's one entails the producibility of 83 additional targets compared to the alga alone.

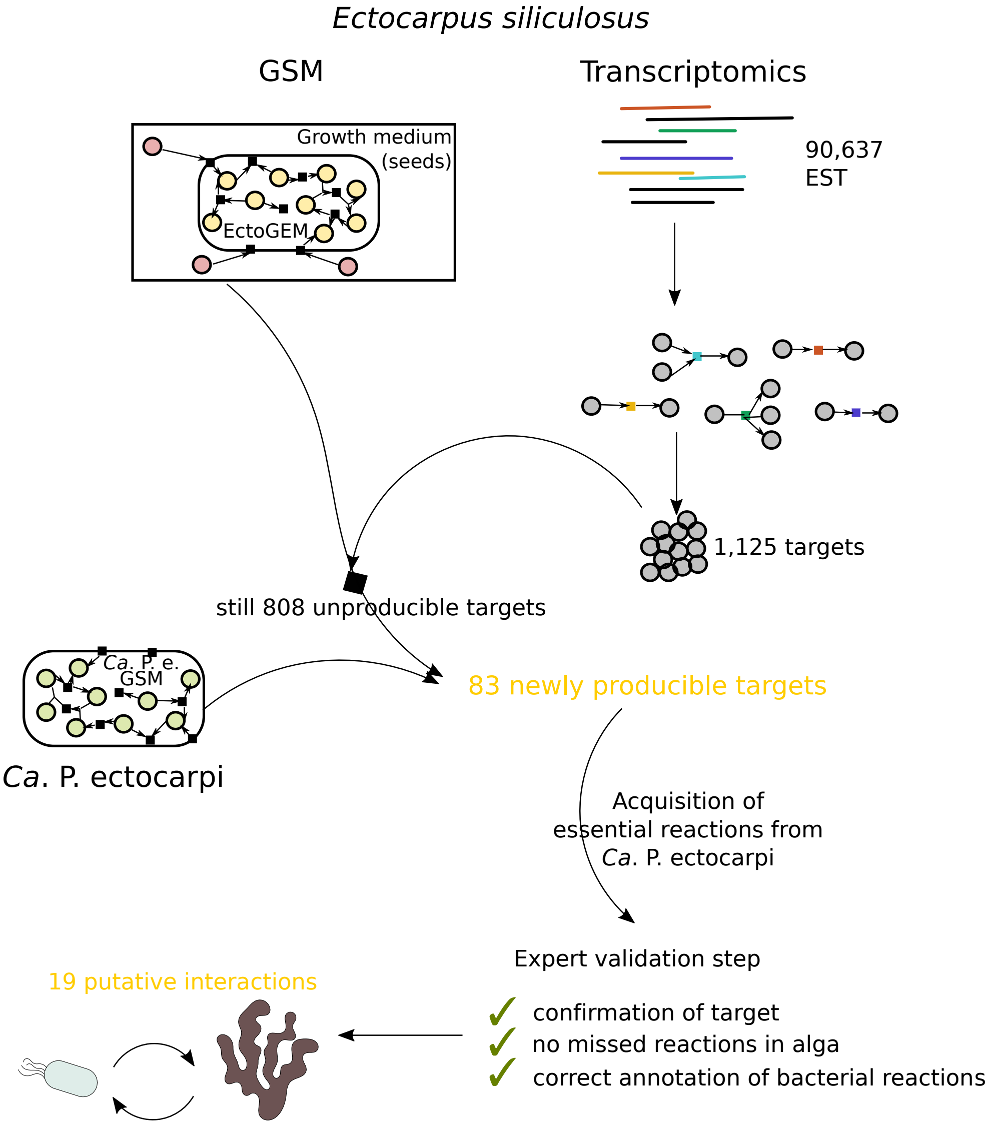


Figure 4.4: ***Application of graph-based gap-filling to an algal bacterial system, design of the study***
Meneco was applied to the study of metabolic complementarities between Ectocarpus siliculosus *and a frequently associated bacterium:* Candidatus *Phaeomarinobacter ectocarpi. Sequenced ESTs of* E. siliculosus *were annotated to retrieve a set of 1,125 targets. The bacterial GSM was used as a database of reactions to decipher newly producible targets for the alga. Essential reactions from the graph-based criterion were studied for the 83 newly producible compounds according to three criteria.*

4.2.2 Validation process of the interactions

For each of these 83 targets, the essential reactions allowing its producibility by *Ca.* P. ectocarpi were studied, as well as the target and its related ESTs in *E. siliculosus*. The essential reactions in this context are the ones provided by Meneco that is to say reactions that are mandatory to restore the producibility of the target, even in non-parsimonious completions. The purpose was to determine whether the production of the target by the alga could depend on an interaction with the bacterium.

The production of a target was considered the result of a possible interaction between the two species if:

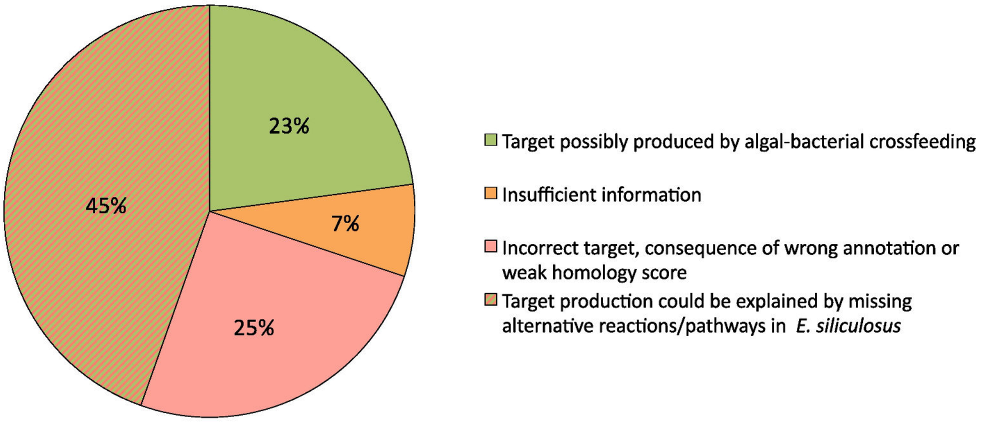


Figure 4.5: ***Study of the 83 newly producible targets when combining E. siliculosus and Ca. P. ectocarpi metabolic networks.***

When merging E. siliculosus and Ca. P. ectocarpi metabolic networks, 83 targets previously non-producible by E. siliculosus became producible. The 93 essential reactions related to their production by Ca. P. ectocarpi GSM as well as the E. siliculosus ESTs were studied to assess whether the new producibility may be the result of a real interaction between the two organisms. This pie chart describes the classification of the putative interactions and false positive ones based on the expert validation.

(i) the algal EST related to the target definition was correctly annotated *i.e.* confirmation of the target

(ii) the target was not producible by the algal itself through the presence of other ESTs or genes

(iii) the bacterial reaction(s) necessary for the target production was (were) correctly annotated.

These individual examinations revealed some errors or missing annotations in *E. siliculosus* as well as some cases of weak sequence homology leading us to no longer consider 21 (25%) of the 83 compounds as targets. Improvements in genome annotations also allowed recovering alternative reactions or pathways in *E. siliculosus* related to the production of 37 (45%) of the compounds. Nineteen compounds (23%) however are likely to be part of an exchange between the alga and the bacterium (Figure 4.5). The details of these examinations is presented in Appendix A.

Highlights

A thorough manual curation of the 83 targets enables to fix annotation errors in both algal and bacterial genomes and to obtain a final set of 19 targets whose producibility putatively results from metabolic interactions between the alga and the bacteria.

4.2.3 Identifying candidate pathways for algal-bacterial interactions

Several of the reactions identified by Meneco were the result of potentially interesting interactions. For example, *E. siliculosus* on its own is probably incapable of producing histidine or histidinol because a histidinol-phosphatase (EC 3.1.3.15) is missing from its genome. Meneco has previously identified this mandatory reaction to complete the histidine biosynthetic pathway [Prigent et al., 2014]. Recent genomic data from other strains of *E. siliculosus* (Dittami, Tonon, personal data), as well as the published genome of *Saccharina japonica* [Ye et al., 2015], confirm the absence of a histidinol-phosphatase in other brown algae, while a corresponding enzyme is present in diatoms. As a proteinogenic amino acid, histidine is essential for all living organisms. Brown algae have therefore either evolved an alternative way of producing histidinol or histidine, or they acquire at least one of these substances from their environment. Analyses indicate that histidinol or histidine may be provided by symbiotic bacteria such as *Ca.* P. ectocarpi, which encodes all enzymes of the histidine biosynthetic pathway.

A second example is vitamin B5 (pantothenic acid), which is an important vitamin for the formation of coenzyme-A. *E. siliculosus* is capable of producing vitamin B5 from β-alanine via the activity of a pantoate-β-alanine ligase (Esi0070_0043), but it lacks a biosynthetic pathway to produce β-alanine (Figure 4.6). Our analysis suggests that *E. siliculosus* may rely on external sources of either β-alanine or vitamin B5. The bacterium *Ca.* P. ectocarpi is able to provide both of these compounds via the phosphopantothenate biosynthesis pathway.

A third example is agmatine, an aminoguanidine, which is well-studied in human where it modulates for instance receptors of neurotransmitters or ion channels. While the role of agmatine in *Ectocarpus* still remains unknown, the *E. siliculosus* GSM possesses two metabolic reactions (MetaCyc ID: agmatin-deiminase-rxn, E.C. 3.5.3.12 and MetaCyc ID: agmatin-rxn, E.C. 3.5.3.11) for the degradation of agmatine and polyamine synthesis. These reactions in *Ectocarpus* are well-supported, despite the fact that it lacks the metabolic capacity to produce agmatine, as no arginine decarboxylase (E.C. 4.1.1.19) is present in its genome. As in the case of histidine synthesis described above, this latter observation also holds true for other sequenced *Ectocarpus* strains and the *S. japonica* genome. A plausible explanation is that *E. siliculosus* (and other brown algae) may use external sources of agmatine to produce polyamines. *Ca.* P. ectocarpi is capable of producing agmatine, which constitutes a potential interaction point between both organisms. Our current working hypothesis is that when external or bacterial agmatine is available, agmatine-derived polyamines may complement or replace ornithine-based polyamine synthesis in *E. siliculosus*.

Highlights

Histidine, agmatine and β-alanine producibity in *E. siliculosus* possibly depend on bacterial interactions.

4.2.4 False positive interactions

The aforementioned three examples highlight promising candidate pathways for algal-bacterial interactions. For other identified metabolites and reactions, however, sequence information was insufficient to determine the exact specificity of the enzymes and reactions, and thus to draw conclusions without additional experimentation. Furthermore, in Figure 4.5, we analyzed the 58 targets leading to false positive results (70%), i.e. candidate algal-bacterial interactions either due to missing gene annotations or false/overly precise annotations. One example for each of these cases is detailed below: one missing annotation in the algal genome falsely leading the assumption that the bacterium was needed to provide riboflavin, and one wrong assignment of EC numbers to an algal gene leading to predict a reaction involved in

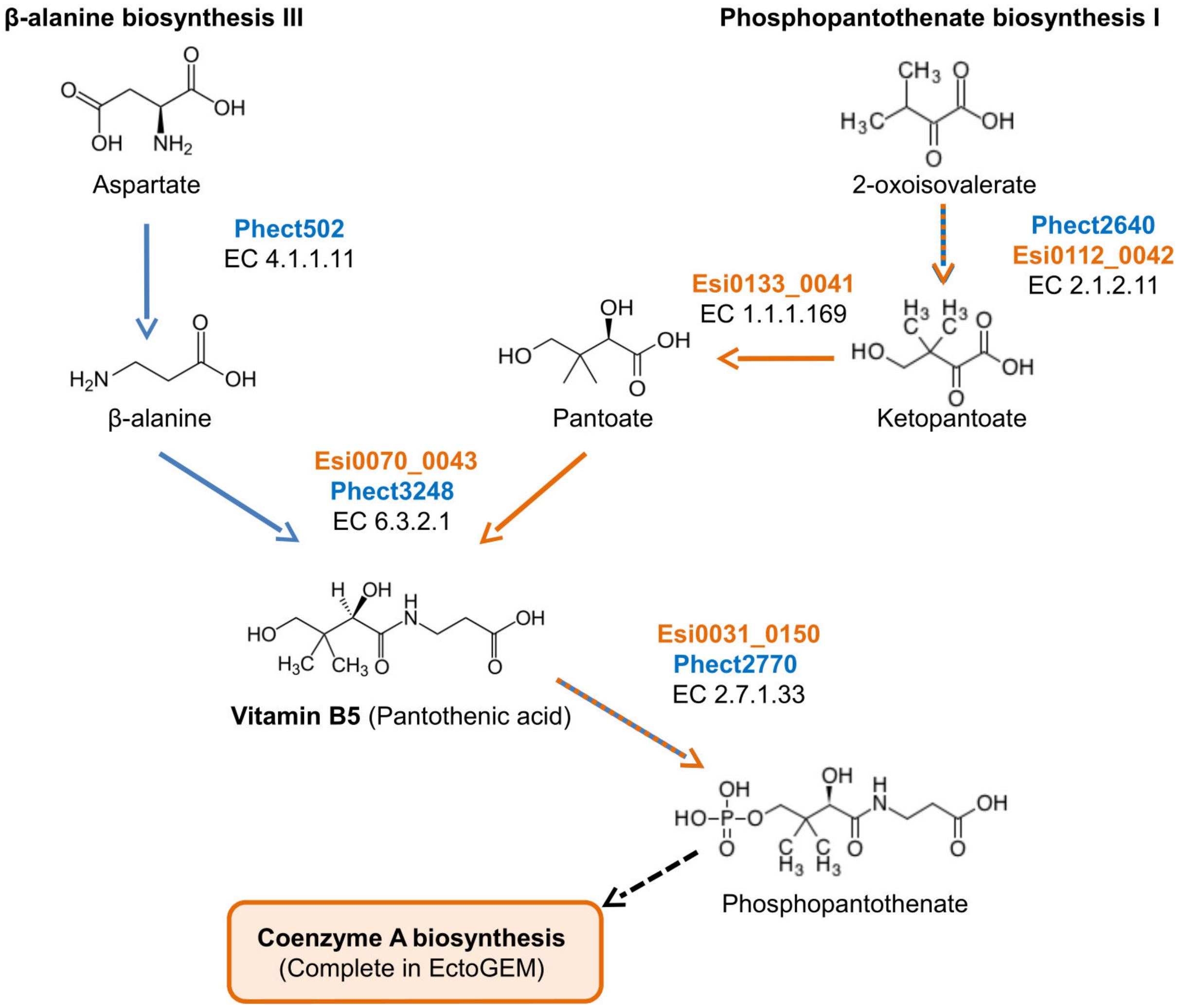


Figure 4.6: ***Vitamin B5 biosynthesis in E. siliculosus and Ca. P. ectocarpi***

Orange labels designate enzymes from the alga, blue labels correspond to enzymes from the bacterium. The alga does not possess enzymes to transform aspartate in beta-alanine whereas the bacterium does. Yet the alga possesses an enzyme that catalyzes the transformation of beta-alanine into vitamin B5 in the coenzyme A biosynthesis pathway. A putative interaction between its associated bacteria could explain the presence if beta-alanine in the algal metabolism.

peptidoglycan synthesis.

The first case (37 targets; 45% of the cases), i.e. the impact of missing annotations, is best illustrated by the case of riboflavin. In the algal network, riboflavin as well as several of its derivatives (FMN, FMNH2, FAD, FADH2), could not be produced because a RIBOPHOSPHAT-RXN reaction was not found. This phosphatase reaction, however, is only poorly characterized; relevant genes have so far only been identified in *E. coli*. It is thus probably carried out by one of the many phosphatases present in the *E. siliculosus* genome and this type of reaction should typically be added by a gap-filling method, as done by Meneco in the context of the analysis of the algal network. In *Ca.* P. ectocarpi, homologs of the *E. coli* phosphatase were found, and the corresponding reaction was predicted. Without manual validation this analysis would have therefore indicated that bacteria may provide riboflavin to the alga. Riboflavin, is a substrate for the synthesis of FAD, a co-factor in a number of other algal reactions, falsely giving the impression that a large number of algal targets depends on the metabolic input of the bacterium. One of the strong points of Meneco with this respect is that it directly identifies essential reactions necessary to produce targets in the network. During the manual curation of the Meneco results, it was therefore possible to group together all targets that require riboflavin and the RIBOPHOSPHAT-RXN, and to treat them all at the same time.

An example for the second case (21 targets; 25% of the cases), i.e. the effect of false or overly precise annotations, is UDP-N-acetyl-*alpha*-D-muramoyl-L-alanyl-*gamma*-D-glutamyl- meso-2,6-diaminopimeloyl-D-alanyl-D-alanine. This compound is an intermediate in the synthesis of peptidoglycans (bacterial cell wall polysaccharides) and was added to the list of targets because the PHOSNACMURPENTATRAN-RXN (E.C. 2.7.8.13) reaction was predicted in the genome and the corresponding gene expressed in the algal network. The corresponding gene (Esi0111 0044), however, is most likely to correspond to EC 2.7.8.15, and has been annotated as such. The reason this gene was also associated with EC 2.7.8.13 in the algal network appears to be a mistake during the manual assignment of the GO term. Thus, despite the results obtained with Meneco, so far there is no indication that brown algal cell walls contain peptidoglycans, or that brown algae rely on bacteria to produce them. Altogether it demonstrates that false negative interactions predicted using gap-filling are also ways to analyze deeper the models and enhance the knowledge we have about them.

Highlights

This work on a algal-bacterial system constitutes a first step towards the shift to community interactions studies using gap-filling derived methods. Together with a strong biological expertise it helps to evidence putative interactions that can be the basis of future experimental validation. In this particular application yet, *Ca.* P. ectocarpi is not cultivable nor can it be isolated so this entails to find new bacterial models to validate the dependency hypotheses.

Conclusion

This chapter described **three applications of gap-filling in the context of studying the metabolism of NMOs** for a better understanding of their physiology. The first section was dedicated to **monitoring the impact of graph-based gap-filling throughout the reconstruction** of a second version of *Ectocarpus siliculosus*'s GSM. We showed that the impact is positively noticeable on functionality at two different scales. First it helps restoring the **producibility** of a large majority of targets, both under graph-based and constraint-based modelings. This is done through the addition of a small number of reactions that can be checked and validated. Secondly graph-based gap-filling impacts the static view of GSM functionality too. It contributes to the **completion of pathways** the same way the other reconstruction steps do. We showed their respective impact on the completion of two pathways for the production of tetrahydrofolates starting from GTP. This view of functionality is static as the activation of the reactions through the seeds and upstream reactions of the model is not considered here, only the existence of reactions in the pathway and the model is studied.

The second application presents a **use-case of Fluto's hybrid completion method on the red alga *Chondrus crispus***. It can **complement the graph-based gap-filling** performed by Meneco for the cases in which the latter does not restore flux into specific targets. Here the producibility of L-α-alanine was blocked and Fluto was applied using the GSM of *E. siliculosus* as a database. One reaction enabled to unblock the producibility of L-α-alanine, that we studied deeper by performing a Flux Variability Analysis (FVA) of its production in the final GSM.

Finally the third application constitutes a **bridge between the first part of this thesis and the following one**. It shows that **graph-based gap-filling can be derived** from its original purpose to **find metabolic complementarities between a host and a symbiont**. We applied it to *E. siliculosus* and one of its frequently associated bacteria *Candidatus* Phaeomarinobacter ectocarpi, the latter GSM's being use as a database for gap-filling a large set of transcriptomics-based algal targets. Meneco has a feature that provides graph-based essential reactions from the database for each target. These reactions from the bacterial GSM were manually studied in order to validate or reject each of the 83 targets that could become producible by the alga using bacterial reactions. This expert validation enabled to correct algal and bacterial erroneous annotations and to shed light on 19 putative interactions between the organisms among which some related to the histidine and vitamin B5 biosynthesis pathways, and agmatine degradation. This paves the way to another kind of gap-filling that, instead of risking the addition of reactions without genetic support to the model, can **capitalize on the symbionts metabolic capabilities to suggest interactions among ecosystems**.

part II

Scalability and combinatorics of community selection

The first part of this thesis presented advances of gap-filling methods dedicated to enhance metabolic modeling of non-model organisms. We presented the results of benchmarking for Meneco, a graph-based gap-filling method. Then we described the design, development and testing of Fluto, a hybrid graph/constraint-based gap-filling methods. Finally we applied these tools to biological use-cases. Notably, we applied graph-based gap-filling to study putative metabolic dependencies between *Ectocarpus siliculosus* and *Candidatus* Phaeomarinobacter ectocarpi. We found various interaction hypotheses that could not be contradicted using biological expertise on genome annotation and literature analyzing. Yet testing *in vitro* these hypotheses is excluded due to the impossibility to cultivate and isolate the considered bacterium. In this second part of the thesis, we present the work related to community selection within microbiotas that will be applied to the holobiont of *E. siliculosus* in order to test its metabolic dependencies towards other bacteria than the uncultivable *Candidatus* Phaeomarinobacter ectocarpi.

Chapter 5 presents the formalism of community selection and its implementation and then illustrates this work by benchmarking the metabolic complementarities within the gut microbiota. Finally, Chapter 6 presents two applications of community selection, one in the human gut microbiota, the second for *E. siliculosus*. This part includes in Chapter 5 and in the first section of Chapter 6 parts of the paper accepted to the ECCB Conference [Frioux et al., 2018a].

Chapter 5

Formalism and modeling of the community selection problem

This chapter proposes a two-step optimization for community selection within large microbiota. Starting from the observation that existing methods have limits, we propose a solution that gets through them and can be applied in the context of large-scale screening of microbiota. A first limit of existing methods lied into the scale-resolution duality. Methods that can scale to large microbiota require lowering the modeling level of communities by ignoring exchanges and boundaries. On the other hand, methods that take the latter into account do not scale to large sets of bacteria. The combination of both techniques seems natural but entails to get through the second limit that is the single solution to the community selection problem that is given in existing scalable methods. A single solution may not be representative of the problem if the space of solutions is very large. Moreover, performing a two step optimization requires to keep all information at the intermediary step to prevent biases in the following one. Finally, as the community selection problem often aims to provide experimenters hypotheses for lab testing, it is likely that the criteria taken into account into the selection are not sufficient with respect to many biological criteria that impact cultures (growth incompatibilities or difficulties, competition etc.). In the light of this, it becomes natural to access all optimal solutions for an a posteriori filtering that takes experiment-specific growth criteria into account.

We then present a first application of this work through a benchmark based on the Human Microbiome Project data. [Eng and Borenstein, 2016] produced Genome-Scale Models (GSMs) for 2,051 bacteria of this dataset and tested on it their mixed-bag inspired community selection algorithm. More precisely, they selected pairs of metabolites, that we will call functions, within the set of metabolites occurring in the GSMs. In each pair, one metabolite was considered an input (seed), the other one an output (target) and the objective was to find a minimal-size community of bacteria able to catalyze the transformation of the input (seed) into the output (target). They tested 10,000 pairs of metabolites this way and concluded that a immense majority of them (>97%) could not be solved, whereas the solvable ones were often achievable by size 1 communities, that is to say a single bacterium was intrinsically capable of transforming the input into the output with its own metabolic reactions. Starting from this dataset and these results, we propose to apply our two-step algorithm to the problem and to scale-up to screen the entire set of functions.

This chapter's content is for most parts extracted from the paper I coauthored with Enora Fremy, Camille Trottier and Anne Siegel that was accepted as a Proceeding paper at the ***European Conference on Computational Biology 2018***, published in *Bioinformatics* [Frioux et al., 2018a].

5.1 Formalism and modeling of the community selection problem

In this section, we formalize and implement, using Answer Set Programming (ASP), the frameworks of large-scale community selection inspired by the works of [Eng and Borenstein, 2016] and [Julien-Laferrière et al., 2016]. The implementation of community selection by the former belongs to the category of mixed-bag or gene-soup modeling as described by [Faria et al., 2016, Henry et al., 2016]. [Eng and Borenstein, 2016] implemented this using a network-flow like modeling and Integer Linear Programming (ILP). Mixed-bag modeling can lead to overestimate the possible interactions between organisms as it does not enable to precisely identify them [Faria et al., 2016]. Yet, [Eng and Borenstein, 2016] showed that this method can scale.

[Julien-Laferrière et al., 2016] on the other hand proposed a graph-based algorithm that solves the Directed Steiner Hypertree problem with the purpose of designing microbial consortia starting from a small set of species of interest. The aim is to select reactions with a minimal total cost, either from bacteria, transports, or from a database of exogeneous ones, the latter two being penalized by a higher weight. They use ASP in practice to study the possible solutions to their problem.

A **community of species** is formally defined as a family of metabolic networks $\{G_1, \dots G_N\}$.The producibility of target in models has been previously defined and used in this thesis under a graph-based criterion. It can be applied to a community too through the concept of scope introduced by [Ebenhöh et al., 2004] and [Handorf et al., 2005]. As a reminder, a metabolite is considered producible, ie in the scope of a model, if it belongs to the seeds or if it is a product of a reaction whose reactants are producible (graph-based semantics). The scope can thus be derived to communities and supports several modelings of communities. It was notably used by [Julien-Laferrière et al., 2016]. The concepts of this section will be illustrated with the toy example of Figure 5.1.

5.1.1 Mixed-bag target producibility

Following the definition of [Henry et al., 2016], the mixed-bag or gene-soup modeling of communities considers a boundary-free meta-organism. Organisms' capabilities are no longer examined individually but collectively in a virtual compartment. Therefore, the associate metabolic network is defined as

$$\text{mxdBag}(G_1..G_N) = (\cup G_i, \cup R_i, \cup E_i).$$

In this chapter we rely on a graph-based definition of reaction activation and reachability of metabolites, based on the scope of metabolic networks associated to seeds. The scope of the

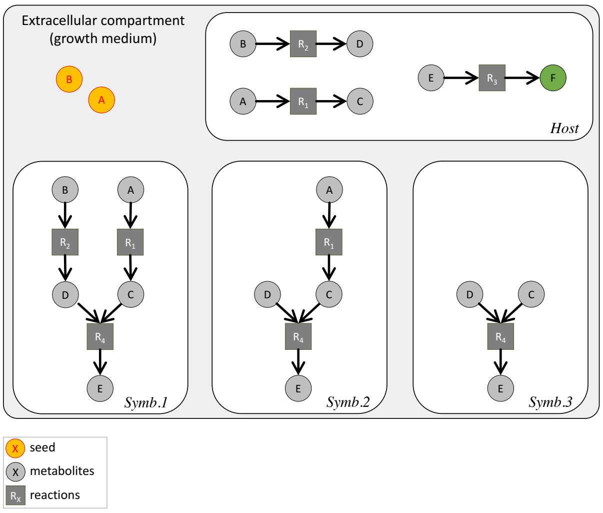


Figure 5.1: ***Toy example for community selection***

One host and three symbionts GSMs are defined. The objective of the host is to produce the F metabolite. The purpose is to select a minimal community to ensure this producibility. The medium consists in two metabolites, A and B.

mixed-bag community from a set of seeds S is naturally defined as the scope of S according to the non-compartmentalized metabolic network:

$$\text{mxdbagScope}(G_1..G_N, S) = \text{scope}(S, \text{mxdBag}(G_1..G_N)). \tag{5.1}$$

In Figure 5.2, the mixed-bag scope covers all the metabolites of the community model. A target t is producible if it belongs to the mixed-bag community scope of the meta-organism. This is the case for F, the target of Figure 5.2. The three symbionts possess the reaction of interest r_4 that would enable the producibility of the target by the host.

5.1.2 Compartmentalized target producibility

Considering that all organisms in the community actually have boundaries and correspond to different compartments requires establishing a family of allowed exchanges between organisms. A first solution is to consider exchangeable a metabolite between two organisms provided it belongs to both of their GSM. However it drastically rises with the number of organisms and the similarity between models (which is expected to be high for automatically built models of NMOs). This solution was adopted by [Julien-Laferrière et al., 2016] since their number of considered models is low. Another solution is to rely on producibility to refine the set of exchangeable compounds, this is the chosen solution here. We denote by

$$\text{exchg}(G_1..G_N) = \{(r_m, i, j) | m \in M_i \cap M_j,\ i \neq j\}.$$

the set of reactions enabling the exchange of each compound m, which belong to pairs of metabolic networks G_i and G_j. Consider a set of exchange reactions $\mathcal{E} \subset \text{exchg}(G_1..G_N)$. The compartmentalized metabolic network associated with the community $G_1..G_N$ and the family of exchanges $\mathcal{E}$ is defined as

$$\text{cptModel}(G_1..G_N, \mathcal{E}) = (\overline{M}, \overline{R}, \overline{E})$$

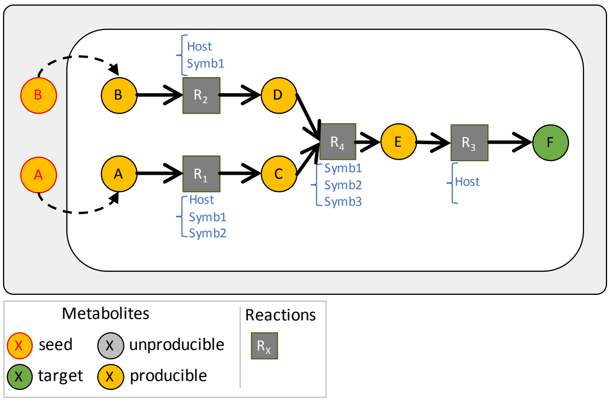


Figure 5.2: ***Mixed-bag selection of community***

In a mixed-bag modeling, all resources are shareable with no cost. Thus exchanges and precise interactions will not be taken into account in the community selection. Here, Host is the only species owning reaction R_3 to produce F from E. Symbionts 1, 2 and 3 all possess the R_4 reaction to produce the precursor E of the target. The minimal community is of size two and three solutions exist: Host & Symb1, Host & Symb2 or Host & Symb3.

for which components are compartmentalized with an index: metabolites are denoted by (m, i), reactions are denoted by (r, j) and edges are denoted by (e, i, j), i and j being the indices identifying the organisms. More formally, we have

$$\overline{M} = \cup_{i=1..N}(M_i \times \{i\}).$$

$$\overline{R} = \mathcal{E} \cup_{i=1..N} (R_i \times \{i\}).$$

$$\begin{aligned}\overline{E} = \{[(m,i),(e_m,i,j)] \,|\, (e_m,i,j) \in \mathcal{E}\} \\ \cup \{[(e_m,i,j),(m,j)] \,|\, (e_m,i,j) \in \mathcal{E}\} \\ \cup_{i=1..N} \{[(m,i),(r,i)] \,|\, (m,r) \in E_i\} \\ \cup_{i=1..N} \{[(r,i),(m,i)] | (r,i) \in E_i\}. \qquad (5.2)\end{aligned}$$

The edges of the community GSMs include the edges symbolizing consumption and production of metabolites of the models and the exchange reactions for the selected exchanged metabolites (Equation 5.2.) Given a set of substrate metabolites S, we allow each of the seed in S to be imported into each considered organisms by creating a set of compartmentalized seeds:

$$\text{cptSeed}(G_1..G_N, S) = \cup_{1..N}(S \cap M_i \times \{i\}).$$

These compartmentalized seeds model the fact that not all organisms possess all the seed metabolites in their GSMs (as reactants or products of at least one of their reactions). Figure 5.3 illustrates the compartmentalized modeling on the small community example.The media compounds A and B, which both belong to the cytosolic metabolic networks of host and symbiont 1, are considered their compartmentalized seeds. Symbiont 2 has only A as seed and symbiont 3 has no internal seeds, because neither A nor B are compounds of its metabolic network.

By extension, we say that a metabolite m belongs to the scope of the compartmentalized metabolic model if there is a metabolic model G_k that contains m and has it in its scope:

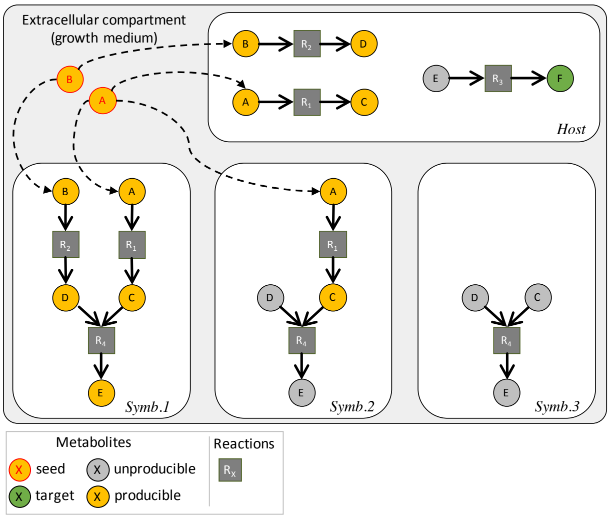


Figure 5.3: ***Compartmentalized modeling of community***

In a compartmentalized modeling, sharing metabolites has a cost and requires exchange reactions. In this case, the compartmentalized scope without exchanges is shown with yellow metabolites. F does not belong to the scope of Host, the only organism which possess it in its GSM. The exchanges that can be considered will concern metabolites that are shared between models and that can be given by the provider, ie belong to its scope. For instance, Symbiont 3 cannot provide metabolite E to others at this step as it does not belong to its scope. However, Symbiont 1 can.

Definition 5.1 Compartmentalized scope *The scope of a community of organisms under compartmentalized modeling is defined as:*

$$\text{cptScope}(G_1..G_N, \mathcal{E}, S) = \{m \in \cup_{1..N} M_i | \exists k, (m,k) \in \text{scope}(\text{cptSeed}(G_1..G_N, S), \text{cptModel}(G_1..G_N, \mathcal{E}))\}.$$

Based on Definition 5.1, in Figure 5.3, the scope of the four-species community can be calculated. Compound E belongs to the compartmentalized scope of $\{A, B\}$ because it can be produced in symbiont 1. On the contrary, target F does not belong to the compartmentalized scope of $\{A, B\}$ even though it belongs to the mixed-bag scope of $\{A, B\}$. The host species is the only one with the enzymatic capability of producing F. Yet there is not way of producing the precursor E in the host species according to the compartmentalized metabolic network: a transport of E to the host is needed. This exchange would ensure F is in the compartmentalized scope of the community.

Given a set of compounds T considered as relevant targets, we define a set of exchanges $\mathcal{E}$ to be consistently associated with a mixed-bag community in agreement with T when it makes it possible to produce, in the compartmentalized model, all the compounds from T that used to be producible in the mixed-bag model, i.e.:

$$T \cap \text{cptScope}(G_1..G_N, \mathcal{E}, S) = T \cap \text{mxdbagScope}(G_1..G_N, S). \tag{5.3}$$

Equation 5.3 entails that there exist exchanges to enable a compartmentalized producibility of the targets within a community if these targets can be produced under the mixed-bag modeling.

Highlights

Two levels of producibility can be used to study communities. Either a meta-organism consisting of all individuals can be considered, or on the contrary it is possible to take exchanges between organisms into account in a compartmentalized model. In any case, both formalisms can be modeled using the graph-based criterion of producibility.

5.1.3 Parsimonious mixed-bag selection of community

Selecting a sub-community following the mixed-bag model entails picking a minimal number of organisms, such that they can collectively meet an objective regardless of transport reactions and exchanges. The search space $\mathcal{C}$ is the set of the 2^N sub-families of the whole community. We select organisms in the mixed-bag community to be added to an empty system (or a system with a host) such that a maximum of its targets are producible, by maximizing the size of

$$T \cap \text{mxdbagScope}(G_{i_1}..G_{i_L}, S)$$

among all possible sub-communities $\{G_{i_1}..G_{i_L}\} \subset \{G_1..G_N\}$. This selection of reactions must occur in a minimal number of organisms, hence the minimization of the number of species L in a second step. Optimal community is defined by successively solving these two problems:

Definition 5.2 Mixed-bag selection of community *The mixed-bag community is defined as the solution to the following optimization problem:*

$$\text{mxdbagCnity}(S, T, G_1..G_N) = \underset{\{G_{i_1}..G_{i_L}\}\subset\{G_1..G_N\}}{\arg\min} \left(\begin{array}{l} \textit{size}\,(T \setminus \text{mxdbagScope}(G_{i_1}..G_{i_L}, S))\ , \\ \textit{size}\{\ G_{i_1}..G_{i_L}\ \}\ . \end{array} \right.$$

For instance, in Figure 5.2, there are three different ways to produce target T from the medium $\{A, B\}$ according to a mixed-bag framework: the host is the only one to possess reaction R_4, it will necessarily belong to the minimal community. It does not possess reaction R_3 so another symbiont is needed. The host can be combined with any of the three symbionts 1, 2 and 3, which all possess the reaction R_4, producing E from C and D.

As the mixed-bag formalism is subjected to a small amount of constraints due to the absence of transport or exchanges modeling, it can be used as a first study of the community and to pinpoint members of interest for experimenters. [Eng and Borenstein, 2016] implemented CoMiDA, an ILP algorithm for solving this problem with a network-flow formalism. They tested it on 10,000 random pairs of seed and target singletons to identify a minimal community. However, CoMiDA, like many other problem-solving techniques, proposes a single solution to the gene-soup sub-community problem, thus preventing the experimenter from catching the global and possibly complex combinatorics of the problem. Topological modeling with Answer Set Programming (ASP) [Gebser et al., 2012] has solving assets, as it enables us to sample the whole space of the solutions and thus provide enumeration of optimal solutions or their intersubsection-union. The foundations of an ASP-encoding enabling the identification of optimal communities is depicted in Listing 2.

```
{sel_orga(B) : orga(B)}.
mixedbagScope(M) :- seed(M).
```

```
3       mixedbagScope(M) :- sel_orga(B); product(M,R); reaction(R,B);
            mixedbagScope(M2) : reactant(M2,R).
4       #minimize { 1@2,T : target(T), not mixedbagScope(T)}.
5       #minimize { 1@1,B : sel_orga(B)}.
```

Listing 2: *Selection of minimal-size communities in a mixed-bag framework*

Line 1 entails that the selected organisms of the community belong to the available organisms. The following two lines define the mixed-bag scope. A compound belong to such scope if it is a seed or if if is a product of a reaction belonging to a selected organism and whose reactants are themselves in the mixed-bag scope. The optimizations aim to minimize first the number of unproducible targets in the community, that is to say the targets that do not belong to the mixed-bag scope. Then the number of selected organisms is minimized too.

Highlights

A first solution to select communities within a microbiota is to adopt a mixed-bag modeling and perform a twofold optimization. First the number of unproducible targets in the community is minimized, then so is the number of organisms involved in this community. This type of optimization is not greedy thanks to the mixed-bag modeling and can be used to discriminate a large number of species in a microbiota.

5.1.4 Parsimonious exchange-based selection of community

The mixed-bag modeling should be considered as a method for globally studying the metabolic complementarity of community members. A limitation is that it does not take into account the required exchanges needed when setting back the boundaries of the metabolic models. A natural motivation is that it is energetically demanding to export or import metabolites, hence a parsimonious hypothesis of exchange dependencies in organisms. [Julien-Laferrière et al., 2016] have introduced an algorithm for the selection of synthetic community based on the minimization of exogenous reactions and transport reactions. Without much information on precise transportable mechanisms, as it is the case for many non-model organisms, the size of the search space consisting of all possible exchanges is $4^{\sum_{i<j} \text{size}(M_i \cap M_j)}$. Consequently we restrict the definition of exchangeable compounds by preventing to consider all pairs of metabolites between all organisms (see Equation 5.2 and line 2 of Listing 3).This makes the search for minimal exchanges an intractable problem for microbiomes with a high number of organisms, and particularly if their models contain a large set of shared metabolites. This occurs a lot in models that are built automatically from metagenomics as they cannot be individually curated and improved.

To solve this issue, we introduce a heuristic to approximate the exchange minimization problem, which entails filtering minimal size communities with criteria based on the number of exchange compounds. This is modeled using an optimization problem chaining three combinatorial optimizations: maximizing the number of produced targets in the community under mixed-bag assumption (as seen before), minimizing the size of the community (as seen before) and, then, minimizing the number of exchanges by considering organisms boundaries again. Therefore, the family of optimal communities in the compartmentalized setting is formally defined as follows:

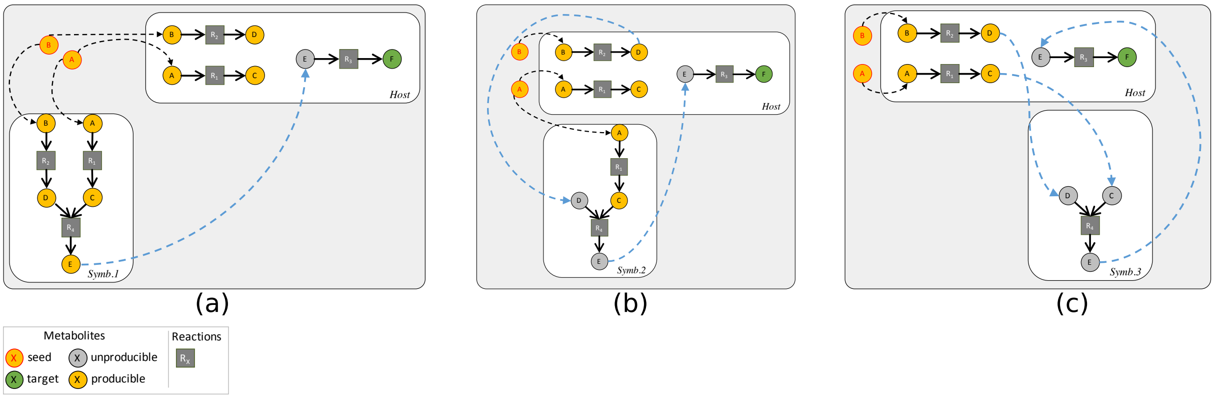

(a) (b) (c)

Figure 5.4: ***Impact of exchange requirements in community selection***

(a) Exchange of E from Symb1 to Host is sufficient for the Host & Symb1 community. (b) Host & Symb2 community requires exchange of D and E metabolites: first, exchange of D so that E can be in the scope of Symb2 and be in turn exchanged to Host. (c) Host & Symb3 community requires exchange of C, D and E metabolites. The unique solution with minimal size and minimal exchanges is Host & Symb1.

Definition 5.3 Compartmentalized selection of community *The compartmentalized community is defined as the solution to the following optimization problem:*

$$\mathrm{cptCnity}(S, T, G_1..G_N) = \underset{\substack{\{G_{i_1}..G_{i_L}\}\\ \subset \{G_1..G_N\}}}{\arg\min} \left(\begin{array}{l} \mathit{size}\,(T \setminus \mathrm{mxdbagScope}(G_{i_1}..G_{i_L}, S))\,, \\ \mathit{size}\{\, G_{i_1}..G_{i_L}\, \}\,, \\ \mathit{size}\{\, \mathcal{E} \subset \mathrm{exchg}(G_{i_1}..G_{i_L}) \mid \\ \qquad T \cap \mathrm{cptScope}(G_{i_1}..G_{i_L}, \mathcal{E}, S) = \\ \qquad T \cap \mathrm{mxdbagScope}(G_{i_1}..G_{i_L}, S) \ \}\,. \end{array} \right.$$

Such optimal communities with their associated minimal sets of exchanges can be identified using the ASP programming paradigm with an extension of the previous encoding, and applied to communities pre-selected with the mixed-bag framework. The predicate *escope* is introduced to recursively compute the scope of the compartmentalized model associated with a selected family of organisms and exchange reactions.

```
1     {sel_orga(B) : orga(B)}.
2     {exchanged(M,O1,O2) : metabolite(M,O1); metabolite(M,O2); escope(M,O1);
          sel_orga(O1); sel_orga(O2); O1!=O2}.
3     escope(M,O) :- seed(M); sel_orga(O).
4     escope(M,O) :- exchanged(M,_,O), sel_orga(O).
5     escope(M,O) :- product(M,R); reaction(R,O); sel_orga(O); escope(M2,O) :
          reactant(M2,R).
6     #minimize{1@3,M : target(M), not escope(M,_)}.
7     #minimize { 1@2,B : sel_orga(B)}.
8     #minimize{1@1,M,O1,O : exchanged(M,O1,O)}.
```

Listing 3: *Selection of minimal-size communities in a compartmentalized framework, together with a minimal size of exchange reactions*

Line 2 describes the set of exchangeable compounds in which will be selected the minimal exchanges. A metabolite is exchangeable if it is producible by the provider (possibly after it received exchanged metabolites itself) *i.e* belongs to its *escope*, if the recipient does not possess it in itw own escope and if both organisms are selected at the end. In the absence of a host, the targets have to belong to at least one selected organism's scope. In the presence of a host, the targets (that are expected to belong to the host) will be expected to be either produced by the host, either finally exchanged to the host by a symbiont.

This approach discriminates the three solutions obtained with the mixed-bag modeling in Figure 5.2 that all are detailed in Figure 5.4. Cooperation between the host and symbiont 1 requires the latter to provide the compound *E* (that belongs to the individual scope of symbiont 1) to the host in order to activate the production of *F* (Figure 5.4 (a)). Alternatively, the host can be combined with symbiont 2 by providing it the *D* compound in order to produce *E*, which can be transferred to the host in return, resulting in two exchanges (Figure 5.4 (b)). Finally, the host can be combined with symbiont 3, by providing it with *C* and *D* compounds in order to produce *E*, which activates the production of *F* in a total of three exchanges (Figure 5.4 (c)). The host-symbiont 1 community is thus the optimal one to restore producibility of *T* when minimizing both the size of the community and the involved exchanges.

Highlights

A second way to address community selection resides in a triple optimization that considers the exchanges. The number of non-producible targets is minimized, as well as the size of the community and eventually the number of required exchanges. It is important to notice that the optimizations are prioritized in this order, so the size of community is favored over the number of exchanges. This entails that this community selection can be applied after the previously presented one (minimal size only) and preserves the size of the community.

5.1.5 Implementation

The Miscoto Python tool[1] (MIcrobiome Screening and COmmunity selection using TOpology) encapsulates the previous ASP encodings and takes as inputs:

- a family of metabolic networks (in SBML format) each corresponding to a symbiont
- a metabolic network associated with a main species called *host* (optional)
- a set of seeds depicting the medium compounds
- a set of targets depicting the expected products

For the sake of an exhaustive target study, the family of targets can be set to be equal to all cytosolic compounds of the host, or the whole set of compounds in the microbiome and host. In this direction, Miscoto has a functionality to compute the added value of the microbiome over the host by studying the whole scope of its metabolites. For the sake of time saving in the case a lot of runs have to be carried out with small changes in the problem (e.g. change of a target), it is possible to prevent the reading of the SBML models all over again and run Miscoto directly with the ASP instance. It is easily human readable and modifiable (with GNU sed for instance) and it can be generated with a specific command of Miscoto.

[1] https://github.com/cfrioux/miscoto

5.1.6 Discussion: divide the problem to better address its complexity

This section addressed the large-scale community selection and modeling problem with two steps; one that aims to reduce the number of species to be considered by retrieving all minimal communities for a given objective, and the second one that models more precisely the system in order to keep only the species that minimize exchanges. We chose to perform these two steps because of the large combinatorics of putative exchanges that need to be considered if applying such modeling to hundreds or thousands of symbionts. As stated in Subsection 5.1.3, transports would be considered between each pair of organisms, that is to say 2^N pairs, with N being the number of organisms, because we would consider transports from a organism A to an organism B and also the reciprocal ones. Then for each of these pairs, a transport between each of the metabolites they both own in their GSMs would be examined. Consequently, the more the GSMs share metabolites, the more putative transports to consider. Formally, if no a priori definition of exchangeable metabolites exist, the family of transports to be examined is 2^K, with K the sum of metabolites shared between each pair of GSMs multiplied by 2 to consider the reverse exchange (Equation 5.4). The power of 2 is the space of research to be examined. For each exchange, a binary condition occurs: either the exchange is kept (1), either it is rejected (0). If the community is described by $\{G_1...G_N\}$ with $G_i = \{M_i \cup R_i, E_i\}$:

$$K = 2 * \sum_{i<j} |M_i \cap M_j|. \tag{5.4}$$

We can apply this formula to the toy example of Figure 5.1. $K = 44$ with six pairs considered $\{Host, Symbiont1\}$, $\{Host, Symbiont2\}$, $\{Host, Symbiont3\}$, $\{Symbiont1, Symbiont2\}$, $\{Symbiont1, Symbiont3\}$, $\{Symbiont2, Symbiont3\}$. The number of existing sets of selected exchanges is 2^{44}. In the HMP benchmark studied in the next section, due to the large overlap of metabolites in the 2,051 models integrated into the benchmark, examining exchanges during the community selection step without any a priori on the bacteria requires considering $2^{3.10^9}$ possible transports and makes it impossible to identify minimal communities. Reducing the search space with the mixed-bag modeling is compatible with the compartmentalized modeling, and does not lose information provided the union or enumeration of optimal mixed-bag solutions is carried out. Thus the number of exchanges to be considered is reduced. However, in our minimal example (Figure 5.3), the mixed-bag formalism does not enable to reduce the search space. In addition we compute exchangeable metabolites on-the-flight using a producibility heuristic so that not all pairs of metabolites need to be considered. For instance in the example (Figure 5.4), exchanges of A and B between $Host$ and $Symbiont1$ (and respectively between $Symbiont1$ and $Host$) will never be considered as they both are already producible in both models.

Performances depend on the complexity and amount of species involved in the experiment. As an example, the first step combined with the enumeration of minimal-size communities (mixed-bag) lasted around 300 seconds for the study of Recon and the 773 bacteria described in the results (Chapter 6) on a personal computer. The combination of the second and third steps lasted 100 seconds in average for each individual target. As runs are independent they can easily be parallelized. Finally, the identification of minimal exchanges (and their union) lasted around 200 seconds per target, less than four hours for all targets combined in one run.

Highlights

Switching from a large microbiota to minimal communities relies on a two-level "divide to conquer" strategy. First the mixed-bag modeling reduces the number of organisms to be considered by working on a system in which metabolic enzymes are shared and exchanges are costless. This is motivated by computational concerns: a direct computation of relevant communities by minimizing both the number of bacteria and the number of metabolic transports among them is not conceivable due to the high combinatorics of the problem. Secondly, minimal communities of the first step can be individually studied by computing their minimal exchanges or minimal-exchanges communities can be computed from the union of previous solutions.

5.2 Benchmarking the bacterial complementarity within the HMP dataset

5.2.1 Use of community selection tools within large-scale screening workflows

The Miscoto tool can be included into workflows for large scale and systematic screening of microbiotas. It can be used for a set of individual set in order to classify them and study their dependencies towards species for producibility. For the sake of efficiency, the reading of the SBML files for each GSM can be performed only once in the workflow through the production of the ASP instances that can be a posteriori altered for testing the system with different targets for instance. The latter can be easily implemented using shell or other languages and enables to save time for large computations. This section and the following chapter present two types of applications using Miscoto-based workflows. In this subsection we present an example of a four step workflow (Figure 5.5) that was applied on the HMP data.

A first step of the analysis of the microbial consortium (see application in Figure 5.5) can a *feasibility analysis*. It entails identifying the added-value of the symbionts for the production of metabolic compounds of the system. Targets are classified according to three criteria: *unsolvable target function* (the target is not producible by the host associated with all its symbionts (mixed-bag model) – *trivial target function* (the target is producible by the host or, if no host is provided, a single species of the community) – *community target function* (the target is producible by the host only when it is combined with one or more symbionts in the community or, if no host is provided, two or more symbionts). To that end, a mixed-bag framework is sufficient.

As a second step, for each community function, the algorithm depicted in listing 1 enables the *computation of the minimal size of the community* required to activate the function. A third step can be an *estimation of functional robustness*. Intuitively, we can expect that the more complex (poor distribution of enzymes of interest among the bacteria, large set of targets) the metabolic objective is, the lower the number of minimal communities there should be to activate a function within a microbial community. In addition, enumerating solutions in synthetic community design can provide experimenters with alternatives for countering biological incompatibilities between bacteria [Julien-Laferrière et al., 2016]. To that end, reasoning-modes of ASP are used to estimate the functional robustness associated with a family of selected functions: first, the union of species involved in at least one minimal community required to activate a function is easily computed with a brave enumeration mode to estimate the range of redundancy associated to the function. All minimal communities associated with the targeted function can then be enumerated if needed.

As a final step, the *identification of minimal exchange communities* among the ones previously identified in step 2 can be performed. Minimal-exchange communities can also be enumerated for a biological-expert analysis. This allows to select communities based on a minimality criterion for transports required in order to decrease the expert-based analysis of exchanges needed to assess their biological feasibility.

More generally, as the study of communities can be very dependent to the data and biological context, we advocated for the design of a generalist tool by developing Miscoto. We wanted it to be usable in flexible workflows, either in the case of specific targeted functions are already known or in the cases in which no information is available on the system and the purpose is a global screening. In the same direction, we wanted it to support the presence or the absence (empty model) of a host. We place Miscoto and its associated workflows as a first step in the study of ecosystems, to target species and functions of interest that can be studied deeper after identification. We see it as a decision-aided tool, in order to furnish the whole information to expert, who in the end, make the final judgment to evaluate the results.

Highlights

> Besides the identification of communities for a precise target or set of targets, Miscoto can be used in workflows for global analyses of cooperation within a microbiota through a general screening of functions.

5.2.2 Design of the analysis workflow

The HMP Consortium stool sample [Human Microbiome Project, 2012] was the first large dataset to describe the abundance and variety of bacterial functions in the gut. It represents a valuable resource for exploring microbial cooperation in the gut metabolism. [Eng and Borenstein, 2016] produced metabolic models for the HMP data. Each of the 2,051 bacteria has its own metabolic model, the union of all consisting of 3,606 unique reactions. The average size of each metabolic model is 1,096 reactions. A benchmark was established to validate their flow-inspired algorithm by selecting a minimal community of bacteria to produce a target metabolite from a seed one. Ten thousand random **pairs of two metabolites - one seed, one target - that we call functions**, were randomly tested. The authors used a ILP algorithm and several formalisms, of which the most constrained was a bipartite graph modeling, equivalent to the mixed-bag framework introduced in Subsection 5.1.3. The goal of the algorithm was to enable the production of the targeted compound, assuming that the seed was considered to be the only available compound for the bacteria in the consortium to activate their reactions. The authors showed that a minimal community could be exhibited to produce the desired target from the unique seed in less than 5% of the cases in the benchmark, thus concluding that in most cases among the ones tested, no path exists between the seed and the target.

Miscoto allowed the analysis of [Eng and Borenstein, 2016] to be pursued by designing a pipeline that enables an exhaustive exploration of the previously benchmarked dataset (available on demand to the authors). The pipeline is depicted in Figure 5.5. We first calculated the metabolic feasibility by a microbial community of all possible 4,948,400 *functions* (seed/target pairs of metabolites). We then studied the functional redundancy by sampling functions associated with a community - i.e., requiring at least two bacteria to be met - and exhaustively enumerated all minimal solutions for each function that required at least 3 bacteria. Ultimately, we focused on cooperation processes and showed that exchange computation can discriminate between size-minimal communities by identifying minimal metabolic exchanges required within the selected bacteria.

5.2.3 Exhaustive feasibility study of the HMP seed/target functions

15% of the HMP functions rely on bacterial cooperation to be met Figure 5.6 depicts the results of the exhaustive study of feasibility of all of the 4,948,400 functions in the HMP dataset, that is, all possible pairs combining any single seed from the microbiome compounds associated with any single target from the microbiome compounds. We classified functions in

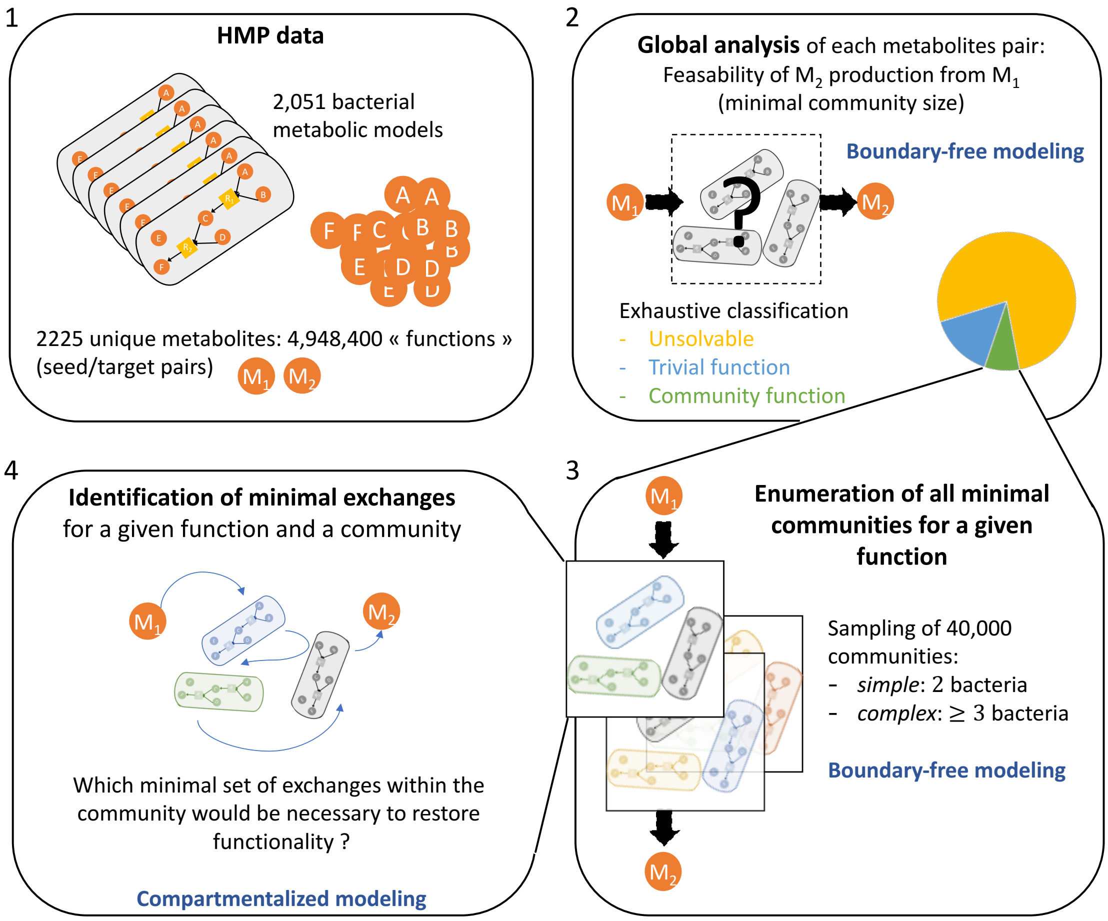

1
HMP data
2,051 bacterial metabolic models
2225 unique metabolites: 4,948,400 « functions »
(seed/target pairs)
M1
M2
2
Global analysis of each metabolites pair:
Feasability of M2 production from M1
(minimal community size)
Boundary-free modeling
M1
M2
Exhaustive classification
- Unsolvable
- Trivial function
- Community function
3
M1
Enumeration of all minimal communities for a given function
Sampling of 40,000 communities:
- simple: 2 bacteria
- complex: ≥ 3 bacteria
Boundary-free modeling
M2
4
Identification of minimal exchanges for a given function and a community
M1
M2
Which minimal set of exchanges within the community would be necessary to restore functionality ?
Compartmentalized modeling

Figure 5.5: *Workflow applied to the HMP*

4,948,400 pairs of seed/target metabolites (i.e., functions) were tested according to their capability to be produced from the microbiome. (1) Function feasibility*: pairs were classified in three categories: unsolvable, trivial or community functions. (2)* Minimality community size computation*: the minimal size of the community allowing us to restore the producibility of 40,000 community functions was computed. (3)* Exhaustive enumeration of minimal communities*: 5,301 pairs that required a community of three or more bacteria were explored in more depth by enumerating all the possible minimal communities allowing us to restore the producibility. (4)* Sorting according to minimal exchanges*: for the 5,301 pairs, a minimal set of exchanges explaining the predicted cooperation was computed. The complete set of minimal exchanges was computed for all communities obtained for a subset of complex functions.*

three categories:

- unsolvable functions that can never be reached regardless of the selected community
- trivial functions that can be met by a single bacteria
- community functions that can only be met through bacterial cooperation (two or more bacteria)

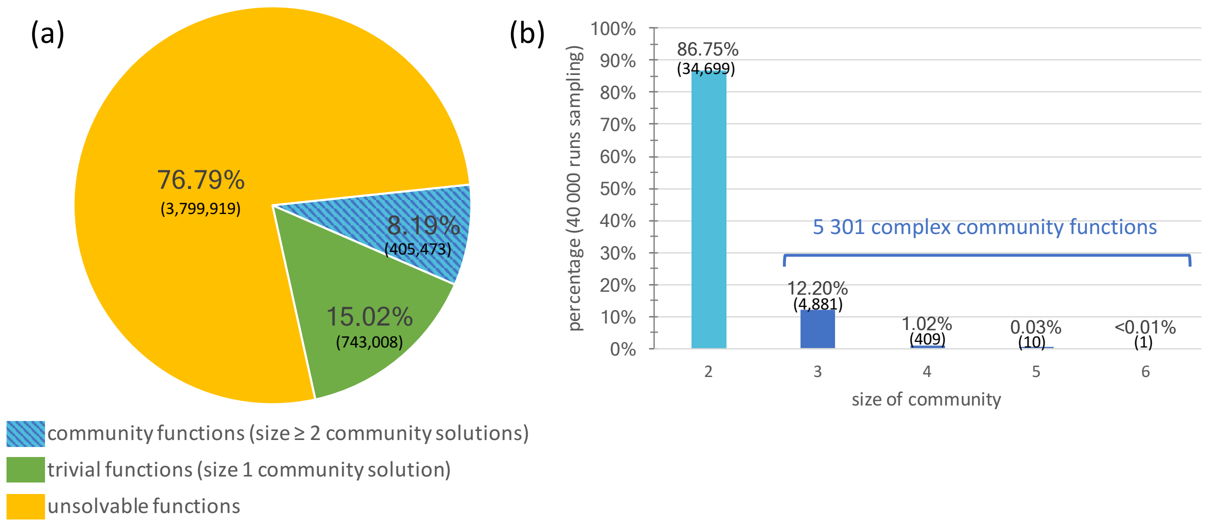
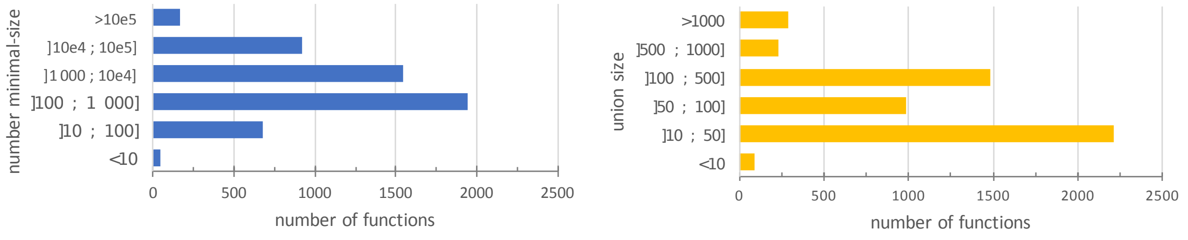


Figure 5.6: ***Feasibility and community-size computation of HMP functions***

(a) Exhaustive study of the feasibility for the 4,948,400 seed/target functions associated with the HMP dataset. (b) Community size computation for 10% of community functions (i.e., 40,000 seed/target pairs associated with a community of size 2 or more): all communities have size less than 6, the most frequent case being size 2.



(a) **(b)**

Figure 5.7: ***Enumeration and union of minimal-size solutions***

Minimal-size solutions (communities) for each of the 5,301 complex functions associated with 3 or more bacteria were enumerated.(a) Distribution of the enumeration size and (b) Distribution of the number of bacteria involved in at least one solution (union size).

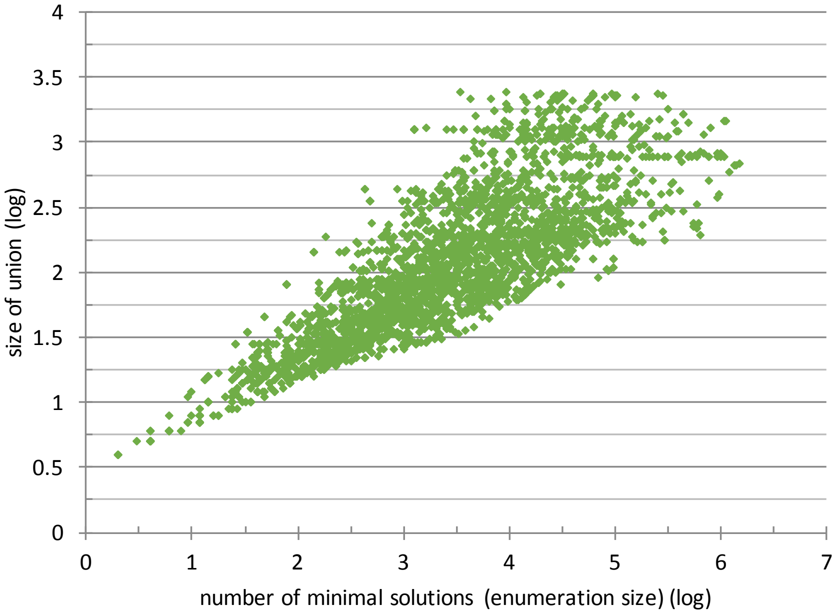


Figure 5.8: ***Relationship between size of union and number of solutions***

For each function, the size of its enumeration of minimal-size solutions and the size of the union of bacteria involved in at least one solution were plotted.

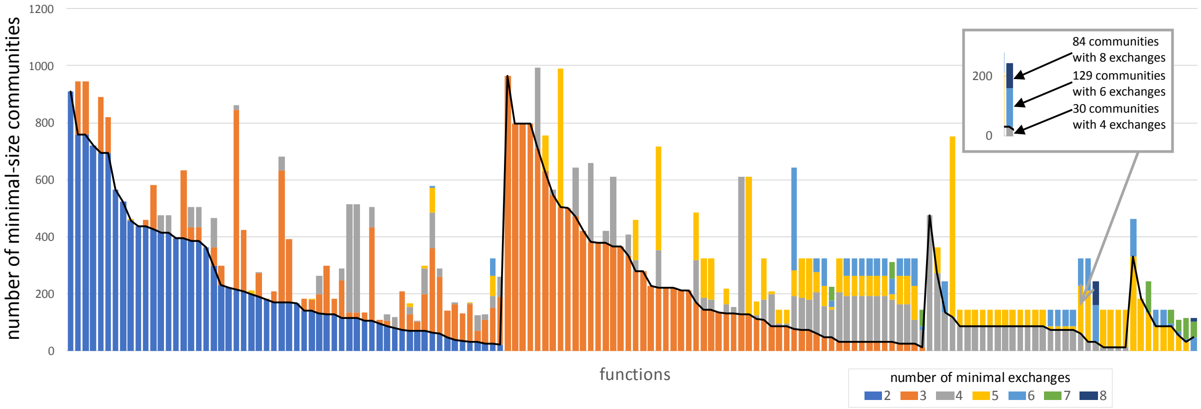


Figure 5.9: ***Minimal-exchanges computation for each minimal-size solution***

150 seed/target functions associated with less than 1,000 minimal communities of three bacteria were selected. For each minimal community (53,081 in total), the minimal number of exchanges required to make effective the target producibility in a compartmentalized framework was computed. Each vertical bar stands for a seed/target function. Colors depict the number of minimal communities associated with a set of exchanges of a given size for each function. As an example, the producibility of the C13629 target from the C00214 seed is made feasible by 243 communities of three bacteria. 30 of these communities have a minimal number of 4 exchanges, 129 communities of 6 exchanges and 84 communities of 8 exchanges. Focusing on communities with minimal-exchanges (below the black line) reduces the total number of communities by a ratio of 45% and the size of the union of bacteria involved in solutions by 24%.

Our analysis demonstrated that 76.8% of functions are unsolvable in the benchmark, which is expected, as we only allow one metabolite to initiate metabolic reactions, i.e., the unique seed. We notice however that feasible functions among the complete family of seed/target functions are more than five times more frequent than estimated in the random benchmark of [Eng and Borenstein, 2016], militating in favor of exhaustive studies of feasibility with a complete view of the producibility capabilities of microbiomes. Our tests also evidenced that 15.0% of the functions are trivial i.e., intrinsically met in one metabolic model. The remaining 8.2%, or 405,473 functions, depend on two or more bacteria: they are community functions (Figure 5.6(a)).

Computation of community size in a sample of community functions To have a better insight of the size of the community associated with community functions, we randomly selected 10% of them (40,000 community functions) and used both our Miscoto tool and the network-flow CoMiDA algorithm [Eng and Borenstein, 2016] to identify a minimal community of bacteria that enables us to complete them. Both tools reported identical results, confirming that the scope-based semantics encoded in Miscoto, in mixed-bag settings, and the bipartite-graph model implemented in CoMiDA are equivalent. We observed that 86.8% of community functions are simple: two bacteria cooperating are enough to restore the function (Figure 5.6(b)). The maximal size of communities needed to restore a function is equal to 6. There are 5,301 (13.2%) complex community functions of size three or more, with most of the functions depending on three bacteria (12.2% of the benchmark, i.e., 4,881 functions, and 420 being of size 4 or more). Together, this analysis suggests that a very low percentage (1% provided that the sampling of complex community functions was representative, at most 7% in all cases) of the whole set of seed/target functions is made feasible with communities of 3 or more bacteria.

Highlights

Miscoto can efficiently classify functions by ensuring an exhaustive exploration of a large-scale microbiome and point out complex community functions. In the HMP, as expected, most functions are unreachable or met by a single bacterium. The remaining functions require a community whose minimal size is rarely superior to 3.

5.2.4 Exploring the whole space of solutions demonstrates functional redundancy of bacterial metabolism

We used the enumeration capability of ASP-based methods implemented in Miscoto to enumerate minimal community solutions for all complex functions (associated with 3 or more bacteria) identified in the 40,000 functions sample, i.e., 5,301 ones. The number of minimal communities per function ranged from 2 to 1,506,662. As shown in Figure 5.7(a), **86.5% of functions generated more than 100 solutions, of which 49.8% generated more than one thousand solutions. The median is 977 solutions per function**. This illustrates the high combinatorics of the community solving problem. In terms of biological interpretation, the number of minimal communities associated with a function can be regarded as a pointer to functional redundancy: a function associated with a large number of minimal community solutions in the microbiome is more likely to be effective than one with a very few number of solutions because several bacteria are able to play an equivalent role with respect to the function restoration [Moya and Ferrer, 2016].

To confirm this hypothesis, for each seed/target function in our sample, we computed the number of bacteria involved in at least one minimal community restoring the function, i.e., the union of bacteria (Figure 5.7(b)). Our analysis suggests that the number of solutions is linked with the number of species involved in a solution with a polynomial relation (Figure 5.8). In this respect, computing the latter is much less time-consuming than the former and is easier to study than a large set of enumerated solutions (Figure 5.7(b)) as the median size of the union is 68.5. This promotes the use of the quantity of bacteria involved in at least one minimal solution to the function restoration problem. When screening a large-set of targeted functions, this criterion enables us to sort functions according to their redundancy and determine whether it is worth enumerating the whole set of minimal communities worth being performed.

5.2.5 Identification of exchanges to discriminate minimal-size communities associated with a given function

In order to get a better insight into the variability of cooperation processes involved in minimal-size communities, we further developed the analysis by selecting a panel of 150 functions for which the minimal size of communities was 3 bacteria and which displayed a number of community solutions ranging from 100 to 1,000, which is the main range identified in the enumeration study. We computed the exchanges in each community of minimal size from the 150 functions (53,081 runs in total) We noticed that the number of minimal exchanges ranges from 2 to 8 for communities optimizing the producibility of one function (Figure 5.9). For 94% of all functions, there is at least one community solution with at most 4 exchanges. 38.7% of the functions can be met by at least one community associated with two exchanges only: they correspond to the simplest exchanged-based functioning for a community of three species, where two bacteria each provide a single precursor to a third one. The functioning of 60.9% of the remaining functions can be explained with three exchanges, suggesting that exchanges within several bacteria (e.g. the production of two precursors by a bacterium or a cycle system between two bacteria) are needed to make the function effective. The other cases correspond to more complex cases with multiple exchanges within the consortium.

As shown in Figure 5.9, only 17% of the functions studied are associated with minimal-size community that all depict the same number of exchanges (monocolor vertical bars in the picture). For the 83% non-homogeneous functions, **focusing on communities with a minimal number of exchanges reduces the number of communities to be explored by 45%**; these minimal-exchanges communities are depicted with the lowest color segment in each vertical bar. Concentrating on these minimal-exchange communities allows us to reduce the average family of bacteria (union) involved in the possible communities from 43 to 30.

Highlights

There are hundreds or thousands different bacterial community that can restore each function. It is thus important to capture their diversity. To do so, a solution is to compute the set of bacteria that belong to these solutions, it is easily calculated and can be used as inputs for exchange optimization.
Adding an exchanged-based criterion to the community-size optimization criteria may facilitate the selection of strains associated with a targeted function by reducing the number of relevant species to be investigated closely.

Conclusion

This chapter presented the **modeling of the community selection optimization problem** and its testing using the HMP dataset. We present two types of modelings, **mixed-bag and compartmentalized**, that can be **chained to face the high combinatorics of symbiont selection** and interactions identification within communities. The two modelings are formalized using the **scope for graph-based producibility**. The first one considers a single compartment in which all enzymatic capabilities are shared, which is an overestimation but yet enables to reduce the number of symbionts to be studied when addressed with parsimonious optimization. The second modeling take organisms' boundaries into account to compute exchanges. It provides a definition of exchangeable compounds that prevents considering a putative transport for each pair of metabolites that belong to two different species.

The two community selection tools are **implemented into a Python package relying on ASP** for an efficient solving, that enables the computation of one solution, the union of solutions or the enumeration. The purpose of the tool is to facilitate the study of ecosystems at the metabolic scale by providing to biologists and experimenters hypotheses to be tested. We place Miscoto and its derivative workflows as **decision-support tools to eliminate the non-optimal hypotheses** and be able to classify all the remaining ones with additional biological criteria, without loss of information. Other modeling methods, notably constraint-based, can be applied to deeper study communities raised by Miscoto, prior to experimental studies for example.

A question that is not addressed in this chapter is the **validation of transports** for exchanges proposed by the modelings. It is taken into account in Miscoto as a priority is given to exchanges for which transports (export from the provider, import to the receiver) are characterized in the model. Yet it is not discussed here as we realized that it poorly applies in practice. Indeed, transport reactions are rare in GSMs and more importantly they often are artificial ones, not supported by associated genes. In this direction, we did not present the existing-transport optimization here and did not take it into account into the applications of the next chapters, for which it would not have changed the results. Yet we strongly advocate for the search of transporters [Elbourne et al., 2017, Griesemer et al., 2018] after using Miscoto to validate the hypotheses raised by the tool.

We propose to use Miscoto within workflows that can be adapted to the objective of various microbiota studies. It can be use to large-scale screening without a priori on the putative cooperation or more more targeted objectives by restraining the studied metabolites for instance. In the second part of this chapter we illustrated the former and screened metabolic cooperation potentials within data of the Human Microbiome Project. We thereby described the **scalability and screening capabilities of Miscoto** to test i) the **feasability of functions** within a microbiota, ii) assess the **minimal size of communities** to perform such functions, iii) compute **minimal-size communities under the mixed-bag modeling** and obtain all optimal solutions and finally iv) perform a **second optimization that minimizes exchange costs** within the previously selected set of solutions to ultimately retain a final set of bacteria that optimize both criteria. We screened the entirety of functions within the HMP and showed that contrary to what the sample examined by [Eng and Borenstein, 2016] seemed to predict, more functions are achievable and a quite large amount of them require bacterial cooperation to be performed. A interesting result of ours is the large number of optimal communities per functions, especially at the first step (86.5% of functions display more than 100 solution, 49.8% more than 1,000) that expresses a **large redundancy of functions within the gut** [Moya and Ferrer, 2016]. **Functional redundancy** has been widely studied, particularly for the gut microbiota, and is linked to the concept of resilience in case of perturbation, homeostasis and metabolic plasticity [Lozupone et al., 2012]. Bacteria belonging to various groups can contribute to similar ecological processes and can substitute for one another [Ha et al., 2014] [Allison and Martiny, 2008]. In particular, [Comte et al., 2013] hypothesize that all bacterial communities have a potential for functional redundancy but this potential expresses or not given environmental conditions and forcing. In any case, this **redundancy advocates for the importance of studying of the whole space of solutions when selecting communities**, so it can be captured. We also showed that computing the union of bacteria involved in one solution is helpful as it is more tractable for analysis. Indeed, it is easy to filter the bacteria of the union and remove any solution using a particular non-desired bacterium. Finally, we showed the added-value of the second optimization as it enable to reduce by 45% the number of solutions compared to the first optimization, and by 24% the size of the union of bacteria.

Chapter 6

Applications of community selection algorithms

The previous chapter presented methods for deciphering microbial communities within large microbiotas. They were implemented into the Miscoto package and enabled a wide screening of metabolic cooperation in metabolic models from the Human Microbiome Project. In this chapter I describe applications on more realistic data (multiple seeds/targets). First I apply Miscoto to the study of global cooperation between the human Genome-Scale Model (GSM) Recon2.2 and 773 gut microbial GSMs. We globally study the added-value of bacteria for the human metabolism and study the combinatorics of bacteria involved in all minimal communities with clustering techniques and individual target studies. A second application brings us back to Non-Model Organisms (NMOs) and in particular to marine biology with the study of *Ectocarpus siliculosus* holobiont. We present a project that aims to shed light on metabolic dependencies of the algal towards its microbiota, through the selection of microbial communities within a set of cultivable marine bacteria.

6.1 Recon 2.2 and gut microbiota complementarities

The applications of my thesis are mainly directed towards non-model organisms, in particular brown algae. However, this entails working with data that is not complete, nor perfectly suitable to benchmark a tool. Consequently I chose the human gut ecosytem, that was extensively studied and for which datasets are curated, to test Miscoto onto the deciphering of interactions between a host and its microbiota.

This section's content is partly extracted from the paper I coauthored with Enora Fremy, Camille Trottier and Anne Siegel that was accepted as a Proceeding paper at the ***European Conference on Computational Biology 2018*** published in Bioinformatics [Frioux et al., 2018a].

6.1.1 Data and models of the human metabolism

When GSMs became available for more and more organisms, it became clearer that a human GSM was a major step to be reached in the field. The human GSM, called *Recon 1* was constructed in 2007 based on a thorough evaluation of annotation and literature [Duarte et al., 2007]; it contains 2,766 metabolites and 3,311 metabolic and transport reactions. Since then the model has evolved, followed by several versions, among which the community-driven version 2 [Thiele et al., 2013a] (5,063 metabolites and 7,440 reactions) and the version 2.2 [Swainston et al., 2016] (5,324 metabolites and 7,785 reactions). In 2018, Recon 3D is published with 4,140 metabolites and 13,543 reactions.

Models derived from the human GSM for each organ and distinguished between women and men are soon available [Thiele et al., 2018]. In parallel, as the number of studies on the gut microbiota rose; models for some gut bacteria became available and studied along with the human metabolism [Heinken et al., 2013, Heinken and Thiele, 2015, Magnúsdóttir and Thiele, 2018]. In 2016, GSMs for 773 bacteria of the human were built by [Magnúsdóttir et al., 2016] paving the way to large-scale analysis of human-bacterial interactions in the gut.

6.1.2 Application of Miscoto to the gut microbiota data

For the sake of application and illustration of large-scale community selection with the Miscoto tool, we applied it to the study of the cooperation potential between Recon 2.2 GSM of the human metabolism [Swainston et al., 2016] and 773 gut microbial models ([Magnúsdóttir et al., 2016]), available on: vmh.uni.lu). The purpose of this experiment is to study microbial cooperation in more realistic conditions than the HMP benchmark through the use of multi-compounds sets of seeds and targets.

The intestinal environment is difficult to model as it is highly dependent to the diet. In order to globally analyze and screen the cooperation within these organisms, we designed a lab experimentation-like modeling setup consisting in the cultivation of enterocytes (intestinal absorptive cells) with bacteria under fixed nutritional conditions. These nutrients were thus voluntarily restricted to 51 compounds of the Dulbecco's Modified Eagle's growth medium (DMEM). This medium mimics a type of cell culture conditions that can suit enterocytes and has been shown to enable some bacterial growth [Biedermann et al., 2014]. These compounds compose the seeds set of the experiment.

Regarding the targets, we decided to study complementarity within the holobiont at the largest scale so we set all cytosolic metabolites of Recon 2.2 as metabolic targets, i.e., a set of 1,920 compounds, having in mind that this target list could be refined in the future following an intestinal version of the human metabolic network [Thiele et al., 2018].

First analyses on the human GSM alone were performed. We noticed that 831 cytosolic compounds are producible from the DMEM growth medium, whereas 1,451 metabolites are producible from native modeling conditions of Recon 2.2 using as seeds all boundary compounds for which imports to the extracellular space and cytosol exist. This confirms that we used stringent conditions but this also ensured a larger control on the metabolites that were available for growth.

A feasibility analysis with Miscoto highlighted that cooperation with the 773 gut microbes may facilitate the producibility of **46 additional target metabolites** with the DMEM growth medium. The same analysis using the native boundary conditions of Recon led to 24 newly producible cytosolic compounds. This first result shows that bacteria have an increased added value on the host in a limited growth medium rather than in the native modeling conditions.

The minimal-size community identification implemented in Miscoto evidenced that, in cooperation with Recon, **three bacteria are sufficient** to ensure the host producibility of the 46 targets. The enumeration of all optimal solutions led to a total of **381 equivalent communities** of three bacteria for this metabolic objective. Only 11.5% of the 773 bacteria, that is to say 89 bacteria (Table 6.1), play a role in at least one of these 381 communities with a cooperation potential.

We computed the minimal exchanges for all 381 communities. They ranged from 42 to 48 exchanges to make the producibility of the targets effective, mostly from the bacteria to the host. Since exchanges do not vary significantly between solutions, we decided to keep the 381 communities and study them deeper.

Highlights

Cooperation between the human GSM and the 773 microbial models enable the producibility of 46 additional metabolites in the human metabolic model. They can be producible through cooperation with any of 381 minimal communities of three bacteria.

6.1.3 Complementarity within the 381 bacteria communities

Clustering of the 89 bacteria involved in minimal communities We identified a unique partition of the 89 bacteria in three disjoint clusters (Table 6.1) of 58, 15 and 16 species under the two following conditions:

- each of the 381 minimal communities comprises exactly one species of each cluster
- a bacterium belongs to one and only one cluster

Table 6.1 depicts the 89 bacteria belonging to at least one of the 381 equivalent minimal communities. Figure B.1 in Appendix B presents the bacteria as a taxonomic tree. All bacteria of Cluster 1 are *Bacteroidetes*, with a large prevalence of *Prevotella* species. Bacteria of Clusters 2 and 3 are *Proteobacteria* and *Firmicutes*. The differentiation of these two clusters based on taxonomy is less obvious than for Cluster 1, with notably *Bacillus* members in both clusters. Globally the taxonomic analysis of clustering strongly differentiates Cluster 1 from the two others. This means that functions performed by these bacteria are specific to the *Bacteroidetes* phylum.

Association of bacteria within communities The graph in Figure 6.1 depicts the links between bacteria within clusters. Nodes describe each of the 89 bacteria (see Table 6.1 for association between bacterial identifier and bacterial names), with the color of the node indicating the cluster they belong to. Edges between nodes indicate that the two bacteria meet in at least one minimal community among the 381.

With this graph, we notice that the distribution of bacteria within each cluster is not homogeneous: each species in a cluster is associated with only few species in the two others. This is even true for Cluster 1 that was differentiated from the two others from the taxonomic point of view. This suggests that bacteria do not have equivalent roles regarding the individual producibility of targets. We can see on this graph that some bacteria seem to be specifically associated to others.

Notice the existence of some subgraphs with distinguishable patterns. For instance, there exist an obvious biclique between bacteria 14, 27 and 44 of Clusters 2 and 3 and other bacteria, among which 33 bacteria of Cluster 1 (right part of the graph) that only associate with this three bacteria. The latter means that 66 solutions out of the 381 involve bacterium 27 for Cluster 3, either bacterium 14 or 44 from Cluster 2 and one of the 33 bacteria for Cluster 1.

Evidence of equivalent bacteria with power graph analysis The previous graph (Figure 6.1) shows the existence of bicliques patterns. This information can be retrieved using a power graph analysis [Royer et al., 2008, Bourneuf and Nicolas, 2017, Shannon et al., 2003] that highlights such patterns in a graph. Figure 6.2 is the power graph describing the association of bacteria.

Power graphs were described in [Royer et al., 2008]. Power nodes are sets of nodes and can be linked by power edges. Consider two power nodes linked by a power edge. This entails that all nodes of the first power node are connected to all nodes in the second one. An example of power node is *PWRN-1-18-1* that contains bacteria 9 and 10. This power node is linked to bacterium 25 meaning that two solutions among the 381 minimal ones involve respectively bacteria $\{25, 9\}$ and bacteria $\{25, 10\}$. A power node is outlined in blue (resp. green, orange) if all the bacteria it contains belong to Cluster 1 (resp. Cluster 2, Cluster3).

The largest biclique of the graph involves *PWRN-1-1-1* and *PWRN-1-1-2* and accounts for 116 out of the 381 minimal community solutions: any of the 58 bacteria of *PWRN-1-1-1* (bacteria from Cluster 1) with bacterium 27 and any of the 2 bacteria of *PWRN-1-17-1*.

Bicliques symbolize equivalency of bacteria within the communities. Notice that some bacteria belong to several power nodes, such as bacteria 32 (*Capnocytophaga ochracea*) and 33 (*Capnocytophaga sputigena*) for instance. We observe that they belong to the same genus thus their equivalency is not surprising. These two bacteria belong to three other power

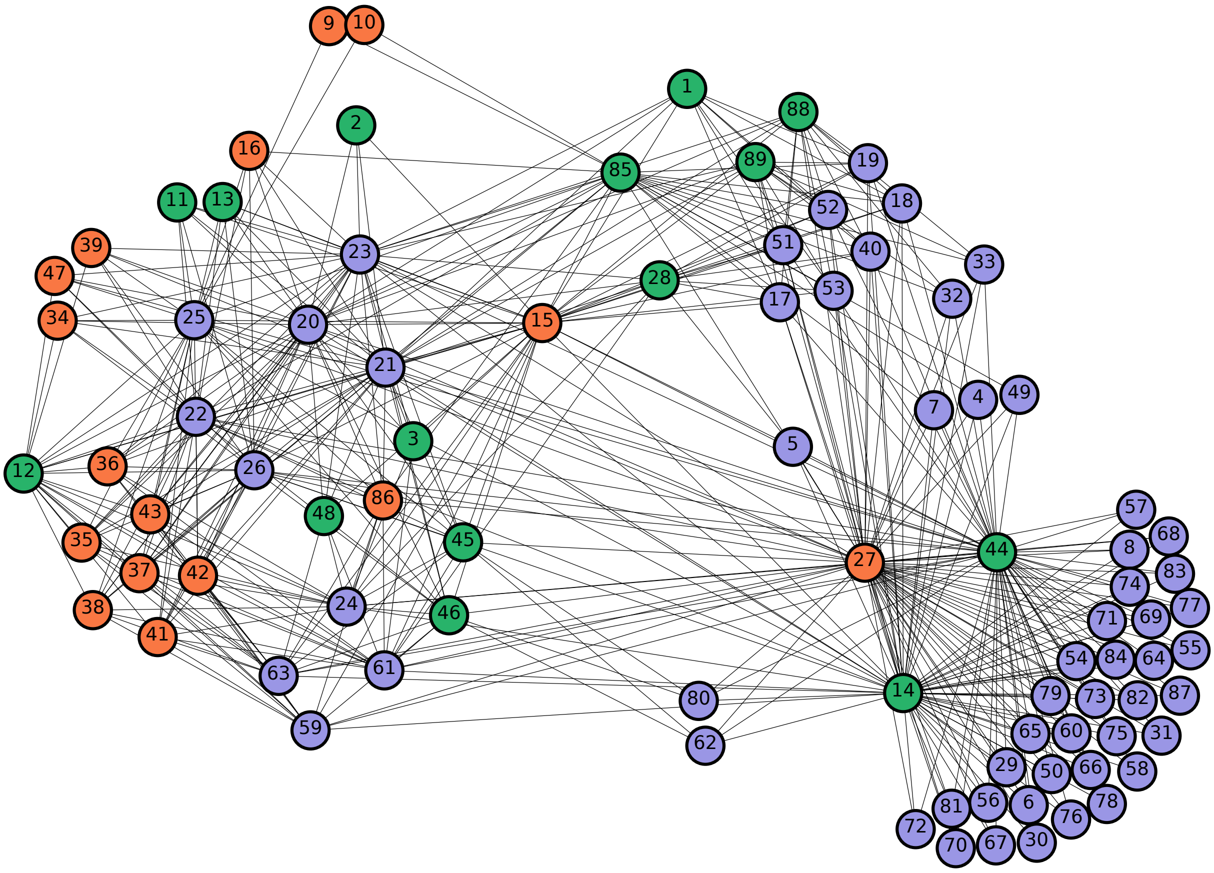


Figure 6.1: ***Connections between bacteria among the 381 minimal communities enabling the production of 46 targets of the human metabolism***

Nodes are the 89 bacteria involved in the 381 communities. The matching between node numbers and bacterial names is depicted in Table 6.1. Bacteria were separated in three clusters such that each of the 381 minimal communities comprises exactly one species of each cluster and a bacterium belongs to one and only one cluster. Blue nodes bacteria belong to Cluster 1, green ones to Cluster 2 and orange ones to Cluster 3. Edges between two nodes indicate that the two corresponding bacteria co-exist in at least one minimal community.

nodes together with other species. The inner power node of a group of bacteria describe the strict equivalency. The outer power nodes describe equivalency with respect to a subset of minimal communities. For instance, bacteria 32 and 33 are strictly equivalent but they also are equivalent to any of the bacteria $\{4, 7, 49, 17, 18, 1940, 51, 52, 53\}$ when it comes to associate with bacterium 27 from Cluster 3 and bacterium 85 from Cluster 2.

Highlights

The study of the bacteria involved in minimal communities together with the enumeration of each community enables to shed light on combinatorics within the associations of bacteria to form communities. It can identify groups of equivalent organisms that can be of interest for experimental procedures in which one bacteria could possibly be replaced by other equivalent species.

Table 6.1: *Identification of the 89 bacteria belonging to the 381 minimal communities. Their names, identifiers (ID) in Fig. 6.1 and cluster affiliations are depicted.*

Bacteria	ID	Cluster
Alistipes finegoldii DSM 17242	4	1
Alistipes indistinctus YIT 12060	5	1
Alistipes onderdonkii DSM 19147	6	1
Alistipes shahii WAL 8301	7	1
Alloprevotella tannerae ATCC 51259	8	1
Bacteroides clarus YIT 12056	17	1
Bacteroides faecis MAJ27	18	1
Bacteroides fluxus YIT 12057	19	1
Bacteroides graminisolvens DSM 19988	20	1
Bacteroides massiliensis B846dnLKV334	21	1
Bacteroides nordii CL02T12C05	22	1
Bacteroides oleiciplenus YIT 12058	23	1
Bacteroides pyogenes DSM20611	24	1
Bacteroides salyersiae WAL 10018	25	1
Bacteroides timonensis AP1	26	1
Butyricimonas synergistica DSM 23225	29	1
Butyricimonas virosa DSM 23226	30	1
Capnocytophaga granulosa ATCC 51502	31	1
Capnocytophaga ochracea DSM 7271	32	1
Capnocytophaga sputigena ATCC 33612	33	1
Dysgonomonas gadei ATCC BAA 286	40	1
Odoribacter laneus YIT 12061	49	1
Parabacteroides goldsteinii dnLKV18	50	1
Parabacteroides gordonii DSM 23371	51	1
Paraprevotella clara YIT 11840	52	1
Paraprevotella xylaniphila YIT 11841	53	1
Porphyromonas asaccharolytica DSM 20707	54	1
Porphyromonas endodontalis ATCC 35406	55	1
Porphyromonas somerae DSM 23386	56	1
Porphyromonas uenonis 60 3	57	1
Porphyromonas uenonis DSM 23387	58	1
Prevotella albensis DSM 11370	59	1
Prevotella bivia DSM 20514	60	1
Prevotella brevis ATCC 19188	61	1
Prevotella bryantii B14	62	1
Prevotella bryantii C21a	63	1
Prevotella buccae ATCC 33574	64	1
Prevotella conceptionensis 9403948	65	1
Prevotella corporis DSM 18810	66	1
Prevotella denticola DSM20614	67	1
Prevotella denticola F0289	68	1
Prevotella disiens FB035 09AN	69	1
Prevotella disiens JCM 6334	70	1
Prevotella enoeca JCM 12259	71	1
Prevotella intermedia 17	72	1
Prevotella intermedia ATCC 25611	73	1
Prevotella loescheii DSM 19665	74	1
Prevotella melaninogenica ATCC 25845	75	1
Prevotella nanceiensis DSM 19126	76	1
Prevotella nigrescens ATCC 33563	77	1
Prevotella oralis ATCC 33269	78	1
Prevotella pallens ATCC 700821	79	1
Prevotella ruminicola 23	80	1
Prevotella shahii JCM 12083	81	1
Prevotella timonensis 4401737	82	1
Prevotella timonensis CRIS 5C B1	83	1
Prevotella veroralis DSM 19559	84	1
Tannerella forsythia ATCC 43037	87	1
Acinetobacter haemolyticus NIPH 261	1	2
Acinetobacter junii SH205	2	2
Acinetobacter pittii ANC 4052	3	2
Bacillus clausii KSM K16	11	2
Bacillus endophyticus 2102	12	2
Bacillus halodurans C 125	13	2
Bacillus mojavensis RO H 1	14	2
Brevibacterium casei S18	28	2
Hafnia alvei BIDMC 31	44	2
Klebsiella pneumoniae pneumoniae MGH78578	45	2
Klebsiella sp 1 1 55	46	2
Methylobacterium radiotolerans JCM 2831	48	2
Serratia liquefaciens ATCC 27592	85	2
Vibrio fluvialis 560	88	2
Vibrio furnissii NCTC 11218	89	2
Bacillus cereus AH187 F4810 72	9	3
Bacillus cereus G9842	10	3
Bacillus mycoides DSM 2048	15	3
Bacillus timonensis JC401	16	3
Brevibacillus brevis NBRC 100599	27	3
Cedecea davisae DSM 4568	34	3
Citrobacter amalonaticus Y19	35	3
Citrobacter freundii ATCC 8090	36	3
Citrobacter freundii UCI 31	37	3
Clostridium difficile CD196	38	3
Clostridium sordellii ATCC 9714	39	3
Escherichia albertii KF1	41	3
Escherichia albertii TW07627	42	3
Escherichia fergusonii ATCC 35469	43	3
Kluyvera ascorbata ATCC 33433	47	3
Staphylococcus cohnii hu 01	86	3

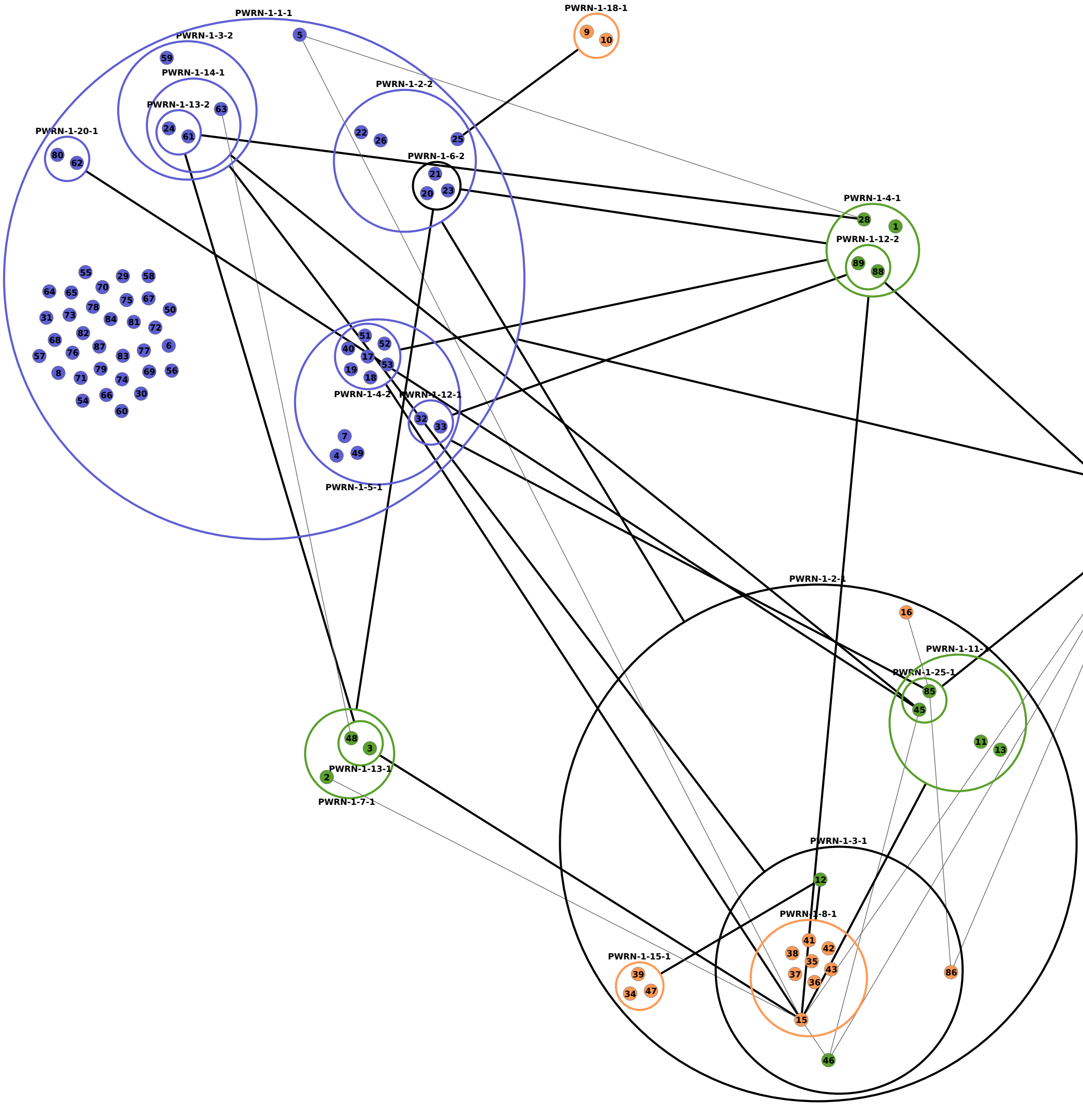


Figure 6.2: ***Power graph analysis of the combinatorics between the 89 gut bacteria***

Each bacterium belonging to one the the 381 minimal solutions is depicted by a node. The present graph displays the bicliques motifs within the graph of Figure 6.1 and preserves its colors describing the clustering of bacteria. A power node is outlined in blue (resp. green, orange) if all the bacteria it contains belong to Cluster 1 (resp. Cluster 2, Cluster3). The creation of this graph was made with PowerGrASP (`https://github.com/aluriak/powergrasp`) [Bourneuf and Nicolas, 2017] and it was visualized using Cytoscape [Shannon et al., 2003].

6.1.4 Functional roles of bacteria inside minimal communities

In order to elucidate both the role of the three clusters of bacteria and the individual role of species in each group, we screened the impact of the 89 bacteria over the 46 individual targets with Miscoto. We performed a feasibility analysis to compute the size of a minimal community to produce each target. The producibility is assessed using the graph-based definition of activation that ensure the target is in the scope of the human GSM provided exchanges occur within a community.

This highlighted that **the producibility of each individual target can be restored by a sin-**

gle bacterium. Figure 6.3 depicts the connection between each target and each bacterium. A yellow spot describes the case when the number of associated exchanges between the human host and the bacterium is not minimal with respect to the considered target. Red spots depict bacteria which do not restore the considered target producibility. Finally, bacteria enabling the producibility of the corresponding target are depicted by a green spot when the required number of exchanges is minimal among all bacteria associated with the target. This heatmap highlights that some targeted compounds can be produced by only very few species, leading to further insights into the 381 optimal communities identified by the Miscoto tool.

A first discriminating compound to produce is D-glucosamine (*gam*), which can only be produced by the species from the Cluster 1 (*Bacteroidetes*, mostly *Prevotella*, *Bacteroides* and *Porphyromonas* bacteria). The second discriminating family of three compounds involving allantoin (*alltt, alltn, C11821*). They can only be produced by species from the Cluster 2. A third family of two discriminating compounds is related to the hydroxy-proline producibility (*X1p3h5c* and *4hpro_LT*): they can either be produced by a subfamily of Cluster 3, or by *Bacillus endophyticus* from the Cluster 2. In the latter case, ADP-glucose (*adpglc*) and methanethiol (*ch4s*) have to be produced by a sub-family of Cluster 3. Provided that these eight compounds are made producible, the producibility of the 38 remaining compounds in ensured by restricting to some homogeneous groups of bacteria (*Porphyromonas*, *Vibrio*, *Capnocytophaga*, *Allistipes*, *Paraprevotella*, *Citrobacter*, *Escherichia*, *Bacteroides*, *Prevotella*) which have similar impact over the producibility of the 46 targets (their associated lines are identical). Some groups represented by a large number of strains may have few differences in terms of target production, possibly explained by gaps in metabolic networks due to differences in genome annotation. Nevertheless, these differences have no impact on the selection of optimal communities.

Highlights

Taken together, these analyses illustrate that the role of the different bacteria in the production of a multi-target function by an optimal community can be elucidated by using the screening of individual target feasibility with Miscoto. An interesting feature of this study would have been to access a human GSM derived for enterocytes. This is going to be available soon as a preprint paper by Thiele and al presents such a work, for a large collection of human male and female organs [Thiele et al., 2018].

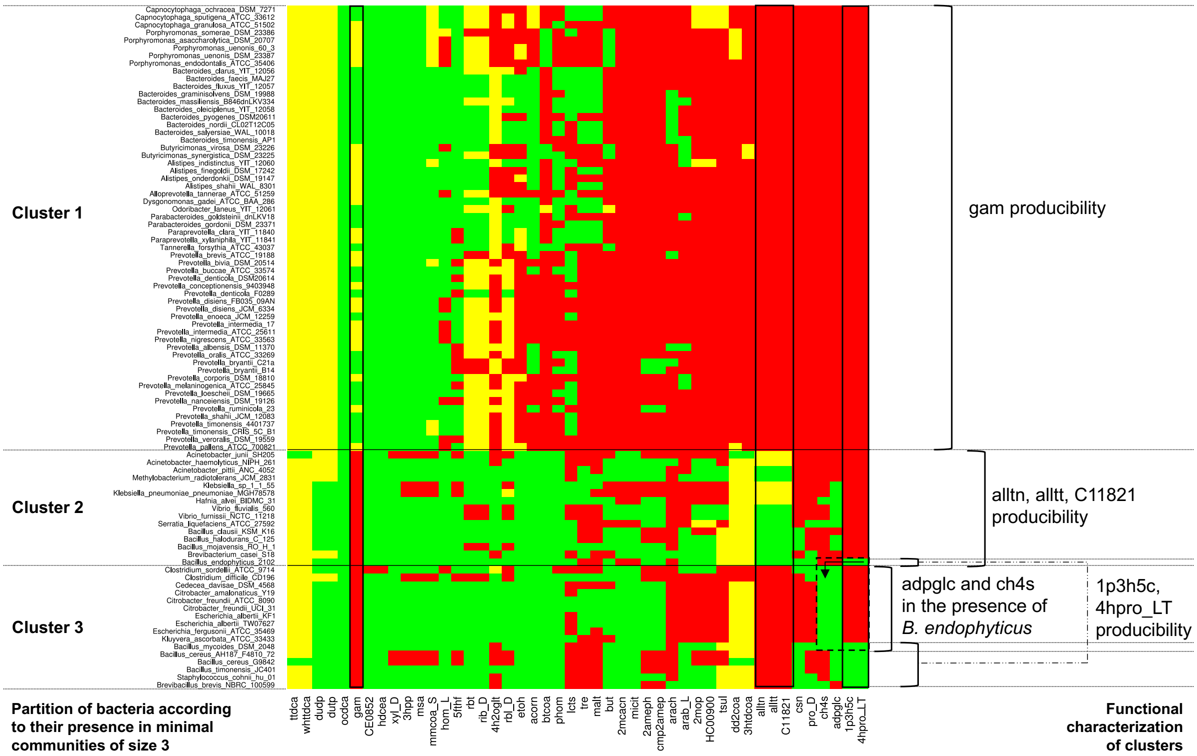


Figure 6.3: ***Study of functional redundancy in the gut microbiota***

Feasibility analysis of a family of 89 bacteria (lines) involved in optimal cooperations with Recon according to the producibility of 46 Recon cytosolic compounds (columns, identifiers from the BiGG database). Bacteria enabling the producibility of the corresponding target are depicted by a green spot when the required number of exchanges is minimal among all bacteria associated with the target. A yellow spot describes the case when the number of associated exchanges is not minimal with respect to the considered target. Red spots depict bacteria which do not restore the considered target producibility. The 89 bacteria are partitioned into three clusters such that each of the 381 optimal communities enabling the producibility of the whole set of 46 targets comprises a bacterium in each cluster. These clusters can be discriminated by the producibility of eight compounds.

6.2 Selection of communities for *E. siliculosus*

The following work was a joint project with Simon Dittami and Bertille Burgunter-Delamare of Roscoff Biological Station (Laboratory of Integrative Biology of Marine Models [a] - UMR 8227) and Enora Fremy, during her Masters degree internship at the IRISA laboratory.

[a] `http://www.sb-roscoff.fr/en/laboratory-integrative-biology-marine-models`

6.2.1 Context of the study

The physiology and reproduction of the brown algal model *Ectocarpus siliculosus* are dependent to its microbiota [Tapia et al., 2016, Dittami et al., 2016] (for additional details, refer to Subsection 1.1.4) but so far the precise mechanisms underlying these interactions are not known. Hypotheses about complementarity between the metabolisms of *Ectocarpus siliculosus* and *Candidatus* Phaeomarinobacter ectocarpi have been discussed in [Dittami et al., 2014a] and in this thesis (Section 4.2) using gap-filling methods, a work published in [Prigent et al., 2017]. This bacterium was co-sequenced with the alga but unfortunately it is not isolable nor cultivable, hence the impossibility to experimentally validate the predicted metabolic interactions.

6.2.2 Study of individual bacterial metabolisms

Previous works of this thesis related to the gut microbiota has shown a strong redundancy of functions among the bacteria of the intestines. This suggests the possibility to consider the function performed by the micro-organism rather than the bacteria by itself. And in particular, it advocates for the possible switch between bacteria with equivalent metabolic capabilities in the same microbiota [Allison and Martiny, 2008, Comte et al., 2013]. Applied to the problem of identifying dependencies between NMOs and their microbiota, it supports the idea of finding bacteria whose metabolic capabilities are close to the ones of *Ca.* P. ectocarpi in order to observe the interactions in the holobiont. This can be done by selecting bacteria among species that were isolated in *Ectocarpus* sp. cultures, or even cultures of other algae such as *Laminaria* sp.

Ten bacteria associated to *Ectocarpus* sp. were isolated and cultivated (See Fig. 6.4 for their taxonomic information). Their genomes have been sequenced and their draft metabolic networks were reconstructed using Pathway Tools [Karp et al., 2016]. Table 6.2 describes the characteristics of their GSMs.

Differences between their functional metabolism can already be exposed by computing the scope of each GSMs, i.e. the number of metabolites that they can produce starting from a set of seeds. The last column of Table 6.2 depicts the size of the scope for each bacterium using the seeds of the alga. It ranges from 112 metabolites to 563. At first sight, there seem to be a positive correlation between the size of the GSM (number of reactions and associated genes) and the size of the scope. However the results for *Imperialibacter* sp. R6 contradict this hypothesis with an average size of GSM and a poor size of scope. One explanation could be the absence of annotation for one or several enzymes whose associated reactions produce one or several important metabolite(s) that is(are) crucial to unblock the producibility of many other metabolites.

The size of the scope of the combination of the alga combined with each of the bacteria can also be computed as well as the ability to produce a set of targets. The chosen set of targets consists in 160 metabolites that become producible in the alga provided its metabolic capabilities are completed with the ones of the ten bacteria collectively. Concretely, it consists into using Miscoto to maximize the number of algal metabolites that are producible. This list was initially bigger but reduced by our colleague Simon Dittami following manual curation. Obvious false positives were removed, among which metabolites that had been already screened when studying the complementarities between *Ectocarpus siliculosus* and *Ca.* P. ectocarpi in [Prigent et al., 2017] (see Section 4.2 of this thesis for details). Merging the metabolic models of the alga and each of the ten bacteria gives insights into the complementarity among each couple. The original scope of the alga contains 357 metabolites. The scope of the ten meta-organisms is displayed in Table 6.3. The second column indicates the number of producible targets for *Ectocarpus siliculosus* provided the associated bacterium and the alga can exchange

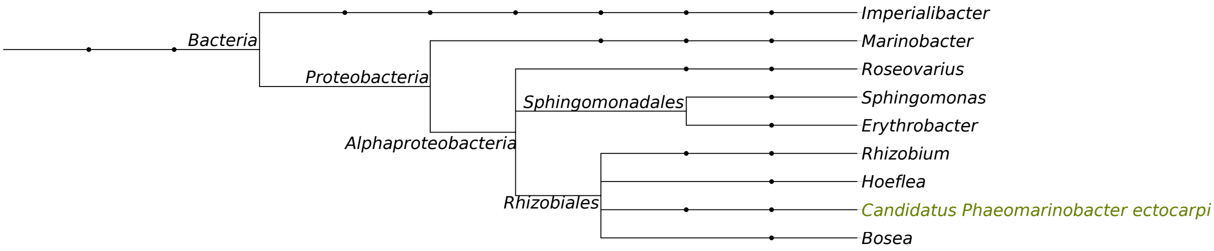


Figure 6.4: ***Taxonomic tree of the 10 bacteria associated to Ectocarpus sp. used for community selection.***

Ten bacteria from the microbiota of the alga are cultivable and can be used for community selection in order to test the interactions predictions obtained previously with Candidatus *Phaeomarinobacter ectocarpi. The latter is also displayed on the tree. The tree was created using iTOL [Letunic and Bork, 2016]*

Table 6.2: *Characteristics of the ten bacterial GSMs used for community selection. The scope is calculated using the algal growth medium as seeds.*

	reactions	genes	metabolites	scope size
Bosea sp. 5a	1753	1765	2004	563
Hoeflea sp. 425	1728	1509	1985	549
Rhizobium sp. 404	1664	1266	1935	538
Roseovarius sp.134	1547	1341	1812	507
Roseovarius sp.420	1541	1334	1808	507
Marinobacter sp. HK15	1551	1225	1807	395
Sphingomonas sp.391	1514	1167	1811	130
Erythrobacter sp. 430	1380	897	1661	119
Sphingomonas sp.361	1379	961	1654	114
Imperialibacter sp. R6	1597	1325	1879	112

metabolites (computed with Miscoto). In this table we observe that *Bosea* sp. 5a, *Hoeflea* sp. 425, *Rhizobium* sp. 404 and the two *Roseovarius* sp. display again the largest scopes when combined to the algal GSM. *Imperialibacter* sp. R6 is no longer the least functional GSM, meaning that maybe *E. siliculosus* owns enzymes that were missing when considering the bacterium alone. Additionally, Table 6.3 shows that no single bacterium is enough to make the alga able to produce the 160 metabolic targets. This means that the optimal bacterial community is of size greater than 1 bacterium. The number of producible algal targets varies less than the range of the scope among the couples: between 129 and 146 targets can become producible in the alga provided cooperation with a bacterium. This entails that the choice of bacteria to be combined to *E. siliculosus* could lead to communities involving a large proportion of the 10 bacteria (in the union of solutions) and that only a few targets might discriminate the added-value of each species.

Highlights

Studies of symbionts metabolic models individually or simply combined with the host as pairs already gives information about the completeness and complementarity of the models.

Table 6.3: *Functional characteristics of each combination of the alga and one bacterium. The size of the scope was computed as well as the ability to produce 160 targets of interest for the alga.*

	size of meta-organism scope	number of targets producible by the alga (no minimal exchanges)
E. siliculosus & *Bosea* sp. 5a	1017	146
E. siliculosus & *Hoeflea* sp. 425	997	149
E. siliculosus & *Rhizobium* sp. 404	983	147
E. siliculosus & *Roseovarius* sp.134	943	143
E. siliculosus & *Roseovarius* sp.420	943	143
E. siliculosus & *Sphingomonas* sp.391	942	140
E. siliculosus & *Imperialibacter* sp. R6	916	137
E. siliculosus & *Marinobacter* sp. HK15	914	139
E. siliculosus & *Sphingomonas* sp.361	878	131
E. siliculosus & *Erythrobacter* sp. 430	859	129

6.2.3 Selection of communities

The use of *Ectocarpus siliculosus*'s GSM prior to gap-filling enables to guarantee the limitation of false-positive reactions in the model. Indeed, as discussed before in this thesis, the main drawback of gap-filling GSMs is the risk to include reactions with no-genetic support that are not catalyzed by the organism but rather whose products are acquired from the environment through metabolic cooperation. Miscoto was run using the GSMs of all ten bacteria and *E. siliculosus*, a set of seeds matching the growth medium of the alga and the set of targets. **The minimal size of the community required to enable the algal producibility of 160 targets is 3 bacteria. There are 6 equivalent solutions and the union of all solutions consists in 6 bacteria out the 10 initially available.** The six solutions are the following:

Solution 1

- *Hoeflea* sp. 425
- *Roseovarius* sp.420
- *Marinobacter* sp. HK15

Solution 2

- *Hoeflea* sp. 425
- *Roseovarius* sp.134
- *Marinobacter* sp. HK15

Solution 3

- *Hoeflea* sp. 425
- *Roseovarius* sp.420
- *Imperialibacter* sp. R6

Solution 4

- *Hoeflea* sp. 425
- *Roseovarius* sp.134
- *Imperialibacter* sp. R6

Solution 5

- *Bosea* sp. 5a
- *Roseovarius* sp.420
- *Marinobacter* sp. HK15

Solution 6

- *Bosea* sp. 5a
- *Roseovarius* sp.134
- *Marinobacter* sp. HK15

The combinatorics of the solutions can be analyzed. Each solution contains either *Bosea* sp. 5a or *Hoeflea* sp. 425 as a first bacterium, and is completed by one of the *Roseovarius* sp. bacteria as a second member. The third one is either *Marinobacter* sp. HK15 or *Imperialibacter* sp. R6. Yet *Bosea* sp. 5a and *Hoeflea* sp. 425 do not play equivalent roles, contrary to the two *Roseovarius* sp.. By mapping the individual targets resolved by each alga-bacterium couple using the six selected bacteria appearing in the previous solutions, we obtain the heatmap presented in Figure 6.5. The figure confirms the equivalency between the two *Roseovarius* sp. species. The need for one species between *Bosea* sp. 5a and *Hoeflea* sp. 425 is explained by the producibility of nine metabolites on the right of the heatmap. Then small variabilities exist between the two pairs: the *Rhizobiales Hoeflea* sp. 425 and *Bosea* sp. 5a on one side, and *Imperialibacter* sp. R6 and *Marinobacter* sp. HK15 on the other side. Minimal exchanges were computed for each solution. Interestingly, all solutions displayed the same number of minimal exchanges (57), thus this criterion does not enable to distinguish between the six of them. All exchanges are directed towards the algal host. No exchanges between bacteria are needed in this case. This means that the bacteria complete some very particular pathways of the alga and that, regardless the chosen solution, the production pathways for all targets require specific reactions that are activable in bacteria from the selection.

Highlights

Six communities of three bacteria enable the alga to produce the 160 tested metabolites based on our analysis. The six community solutions involve only six bacteria, drawing combinatorial patterns that exhibit metabolic similarities between the species.

6.2.4 Experimental design and results

These six solutions were provided to the colleagues of Roscoff in the context of Bertille Burgunter-Delamare's internship to set-up an experimental design whose purpose is to establish the effects of some of the previously computed communities on the algal growth.

They set-up the following experiment. Three co-cultures were designed by Bertille Burgunter-Delamare and coworkers:

- *Marinobacter* sp. HK15, *Hoeflea* sp. 425, *Roseovarius* sp.420 (Solutions 1-2)
- *Marinobacter* sp. HK15, *Bosea* sp. 5a, *Roseovarius* sp.420 (Solutions 5-6)
- *Marinobacter* sp. HK15, *Hoeflea* sp. 425, *Imperialibacter* sp. R6

Two controls were added to the experiment:

- *Ectocarpus siliculosus* in native growth conditions without inoculation
- *Ectocarpus siliculosus* in a growth medium completed with antibiotics without inoculation

In parallel, cultures of the alga with a single strain of bacteria were made: *Sphingomonas* sp., *Erythrobacter* sp. 430, *Bosea* sp. 5a, *Hoeflea* sp. 425, *Imperialibacter* sp. R6, *Marinobacter* sp. HK15, *Roseovarius* sp.. The cultures lasted 28 days with a visual evaluation of algal growth and diverse measurements; and were followed by 16S RNA sequencing (metabarcoding) and metabolomics analyses to test the presence of eight metabolites:

- spermidine

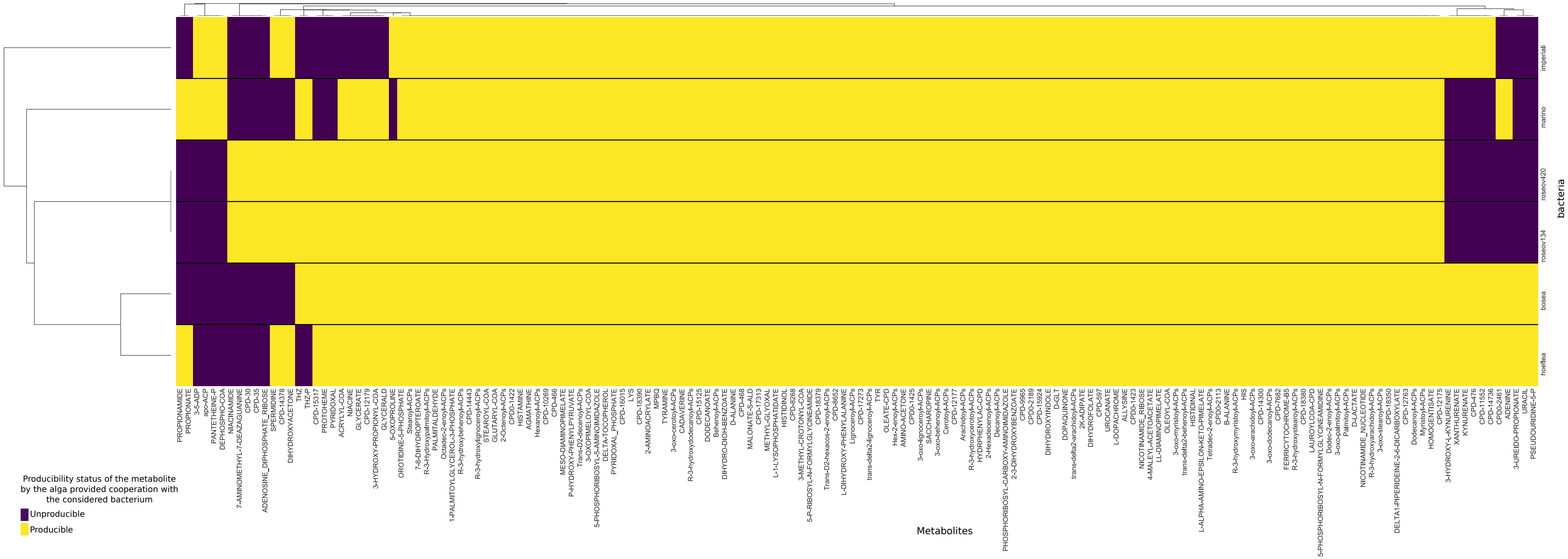
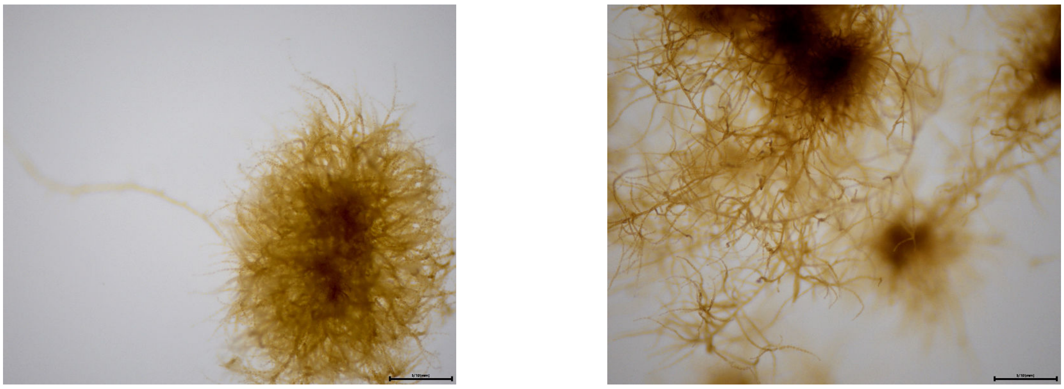


Figure 6.5: ***Dependencies of targets producibility by alga towards cooperation with the six selected bacteria***
The ability of Ectocarpus siliculosus and each of the six bacteria (Bosea sp. 5a, *Hoeflea* sp. 425, *Marinobacter* sp. HK15, *Imperialibacter* sp. R6, *Roseovarius* sp.*134 and Roseovarius* sp.*420) to cooperate such that the alga produces target metabolites was tested. Purple spots indicate that the alga cannot produce the metabolite when associated to the bacterium. A yellow spots indicates the producibility of the target compound. The heatmap shows that the two Roseovarius* sp. *species display the same behaviour when associated to the alga. Hoeflea* sp. 425 *and Bosea* sp. 5a, *the two* Rhizobiales *are globally similar, and Imperialibacter sp. R6 and Marinobacter sp. HK15 are more distinguishable.*



(a) **(b)**

Figure 6.6: ***Differences of morphology in Ectocarpus siliculosus after 28 days of culture***
Pictures taken by Bertille Burgunter-Delamare. Horizontal rule symbolizes 500 µm. (a) culture in control condition without bacterial inoculation. (b) culture with bacterial inoculation composed of Marinobacter sp. HK15, Bosea sp. 5a, Roseovarius sp.

- putrescine
- nicotinic acid (vitamin)
- folic acid (vitamin)
- auxin (hormone for plant growth control)
- L-histidine (amino-acid)
- β-alanine (amino-acid)
- preQ1

6.2.5 Experimental results

We briefly present here the results obtained by Bertille Burgunter-Delamare.

Beneficial effect of bacteria on the algal growth The inoculation of bacteria increases the algal growth although after four weeks it does not impact the number of individuals as it was observed by [Tapia et al., 2016] after six weeks. The morphology of the alga varies between non-inoculated cultures and inoculated ones, as shown by [Tapia et al., 2016]. Figure 6.6 illustrates the changes of morphology after 28 days of culture between the algae inoculated with *Marinobacter* sp. HK15, *Bosea* sp. 5a, *Roseovarius* sp.co-culture and the non-inoculated treated algae.

Identification of bacterial OTUs in cultures Interestingly *Hoeflea* sp. 425 is found in every culture, even the controls. An hypothesis is that the bacterium was not reached by the antibiotics. There are bacteria in the cell wall of the alga, it is not impossible that some are intracellular; in both cases they would be harder to reach with antibiotics. However *Hoeflea* sp. 425 was isolated from the algal growth medium so no clue enable to hypothesize its presence within the seaweed so far. On the contrary, *Imperialibacter* sp. R6 and *Erythrobacter* sp. 430's corresponding OTUs were not found in the cultures in which they were expected. This could be due, for instance, to competition between bacterial species [Egan et al., 2013].

Metabolomics assay Regarding the metabolomics study, all compounds were identified in at least one alga-bacteria co-culture. On the contrary, only preQ1 was found in the antibiotics without inoculation control culture. Biologists observed that no culture presented the whole set of tested compounds despite the prediction. Several hypotheses can be established among which the consumption of the metabolites. The fact that not all predicted metabolites appear in every co-culture could also be explained by competition events. Nevertheless, metabolomics confirms the hypothesis of metabolic cooperation for the all metabolites but PreQ1. Competition is not easy to evaluate in the communities. It is expected to result from the consumption of similar resources by several species, which could be assessed using the topology of the metabolic models [Kreimer et al., 2012]. Other reasons for competition have been isolated: the effect of antibiotics, which is plausible here as the medium has been perturbed by antibiotics during the experiments, and also a diminution of bacterial secretions that promote mutualism, both described in [Coyte et al., 2015].

Highlights

The first lab experiments performed based on Miscoto predictions present interesting results for the better understanding of interactions between *Ectocarpus siliculosus* and its microbiota. Not all predictions were experimentally observed, yet the beneficial effect of added bacteria is clearly visible. Competition that is not taken into account into our model could constitute an explanation to the differences observed between modeling and *in vitro* tests.

Conclusion

This chapter explored metabolic complementarity within the human gut ecosystem. We computed the added-value of bacterial metabolic capabilities over the human producibility of metabolites under strict culture conditions. We showed that 381 minimal communities of three bacteria enabled the human metabolism to produce 46 additional metabolites. These communities involved a set of 89 bacteria. For the 46 newly producible human metabolites, we studied their dependencies towards the 89 bacteria and thus explored the **functional redundancy of functions within the system**. The small number of newly producible metabolites can have a twofold explanation. First the modeled culture conditions are stringent through the use of DMEM medium as a set of seeds. Secondly the human model was highly curated and the GSM used in the experiment is the fully functional one, possibly gap-filled using reactions with no gene-evidence [Thiele et al., 2013b]. For instance, there are 3,043 reactions without indicated gene association in Recon 2.2, some of them being expected as they are imports or exports of metabolites.

The second biological application of this chapter is directed to **better understand the metabolic interactions between the brown alga *Ectocarpus siliculosus* and its microbiota**. In Chapter 4 we identified putative interactions between the seaweed and *Candidatus* Phaeomarinobacter ectocarpi. Unfortunately, the impossibility to grow the bacterium in the lab made it impossible to validate the hypotheses. Yet, we showed on the gut that redundancy occurs within the functions carried out by the microbiota and that eventually, the existence of a specific function can be initially studied regardless the identity of the provider. Relying on this makes it possible to look for functions similar to the ones of *Ca.* P. ectocarpi in other bacteria that can be found in *E. siliculosus*'s environment. We had a set of 10 bacteria that are cultivable and in which we were able to pick minimal communities. Six different minimal communities were predicted to enable the producibility of 160 metabolites in the alga. Three communities were **experimentally tested**, along with cultures of individual bacteria. The beneficial effect of restoring the bacteria after antibiotics treatment was observed on the alga. 7 out of 8 tested metabolites were observed only in algal cultures supplemented with bacteria. Nonetheless not all cultures displayed the expected metabolic behaviour. An explanation could be found into competition between bacteria [Egan et al., 2013]. The impact of the antibiotics, that do not enable a complete elimination of the bacteria, has also to be taken into account as it strongly perturbs the microbiota prior to inoculation with the desired communities.

Despite the promising results obtained in this first experimental application, it is important to retain the putative weaknesses in modeling that can affect the experimental testing. The **bacterial models have been build automatically** and were not curated thus it is possible that some reactions are false positive and highly probable that some other reactions are missing due to annotation limitations. Regarding the alga, weaknesses reside in the absence of gap-filling prior to community modeling together with the graph-based modeling that can miss the activation of some cycles at steady-state. It is thus expected that a certain proportion of the target metabolites we tested are indeed producible by the alga itself. Yet **graph-based modeling is also an asset** as it enables to work on non fully functional models, which is inevitable in the field of NMOs. In addition, the work presented here modeled only the putative cooperation and not the **competition** events that can also happen. Altogether, this work on the selection of communities for experimental application on algae is a promising step in the field of bridging modeling and experimentation for NMOs metabolic studies.

part III

Software, discussion and perspectives of the thesis

This part contains a Chapter related to the software development performed during the thesis. The methods to encapsulate and distribute Answer Set Programming (ASP)-based tools in computational biology are discussed. A project on a workspace for traceable and reproducible reconstructions of Genome-Scale Models (GSMs) is presented. This Chapter is followed by the conclusions and perspective of the thesis.

Chapter 7

Integrating heterogeneous bioinformatics software in traceable workflows

During my PhD, I developed or contributed to the development of several software projects related to the modeling of metabolism in organisms or communities. I will present here this software and how it was applied to the various projects of my PhD. The AuReMe workspace aims to ensure traceability and reproducibility of "à la carte" Genome-Scale Models (GSMs) reconstructions. Despite the existence of major platforms for this purpose, it is noticeable that many GSMs rely on several pipelines, tools and database rather than on one single platform. This heterogeneity provides good quality models but as the tools used for reconstruction poorly interact one with another, it causes a loss of traceability and reproducibility. Most models do not provide information on the reasons why elements of the GSM were added to the model, nor the method that led to this addition. AuReMe is an adaptable workspace that accepts inputs of these major platforms and proposes a large set of refinement and analysis tools that can be chained into pipelines. All modifications to the initial model will be traced and tracked so that the final GSM is fully reproducible and all adequate metadata is stored and made accessible in a user-friendly wiki. This chapter describes also three other tools I have developed or designed during my PhD. Their domains of application are the study of GSMs functionality under graph-based definition of producibility (MeNeTools), hybrid graph-based constraint-based gap-filling (Fluto) and selection of communities within microbiota (Miscoto).

7.1 Software for combinatorial metabolic modeling

User-friendly implementation and release of ASP-based tools Answer Set Programming (ASP) is a powerful programming paradigm for modeling and solving combinatorial problems. This is particularly true in the field of metabolic modeling, notably when using graph-based definition of producibility. We nevertheless showed in Chapter 3 that ASP also supports constraint-based modeling through the use of linear programming propagators. This is implemented in Fluto.

In practice, hiding the calls to ASP solvers to the users through the encapsulation in generic programming languages is preferable in computational biology. It enables to provide packages that are easy to install and use, in programming languages such as Python.

During my PhD, I developed several ASP-based tools for the purpose of studying metabolism; they are described in the present section. The first one is MeNeTools (Metabolic Network Tools) that performs graph-based functionality analyses of metabolic models. For instance, it computes the scope (set of producible compounds) of GSMs starting from available metabolites. Details about the MeNeTools are available in Appendix C. I also developed Miscoto (Microbiome Screening and COmmunity selection using TOpology) for selecting minimal communities of organisms within microbiotas.

These two tools rely on the PyASP (`https://github.com/sthiele/pyasp`) Python wrapper for ASP solvers. It installs the grounder (Gringo) and solver (Clasp) binaries and enables to call them from Python modules. More recently, Gringo and Clasp have been merged in a standalone Clingo whose latest versions enable the support of propagators [Gebser et al., 2016b]. In the case a propagator is needed, the use of PyASP is no longer an option and can be replaced by the Python package associated to Clingo. The latter is available on Conda (`https://anaconda.org/potassco/clingo`). If users already own Clingo on their system and notably in their PATH, the use of Clyngor as a wrapper (`https://github.com/Aluriak/clyngor`) is possible and supports propagators and PyASP-like features.

As Fluto requires a constraint propagator, the use of PyASP was not possible and I used Conda to package the tool so that the user only has to additionnally install its licenced CPLEX Python module (free for academics). This enables to ensure Clingo and its Python module do not have to be installed by the user. By default, the package comes with a promotional version of Cplex that is sufficient to solve a toy gap-filling model provided as an example.

7.1.1 MeNeTools

MeNeTools is a Python 3 package. It is a toolbox that enables to investigate the properties of a GSM with respect to the graph-based definition of functionality. All MeNeTools rely on ASP for the analysis of the model topology. The link between Python and ASP is ensured by the use of the PyASP package (`https://pypi.org/project/pyasp/`). The package includes four functionalities:

- *menescope* provides the scope of a metabolic model and associated seeds, that is to say the set of metabolites that can be reached from the seeds. It takes two SBML files as inputs: a metabolic model and seeds.
- *menecheck* carries out the first step of analysis that is performed by Meneco the gap-filling

tool. It analyzes the producibility of targets and classifies them accordingly. Its inputs are a metabolic model, seeds and targets, all in SBML format.

– *menepath* aims at explaining the producibility of target compounds by giving a subset of the model that is sufficient to produce the target from the seeds. It can isolate the reactions that are essential, based on the graph-based definition of producibility, to produce a metabolite. It takes the same inputs as menecheck.
– *menecof* gets the minimal set of cofactors or metabolites that enables to maximize the number of producible targets. The purpose is to find metabolites that, if become producible or are added to the seeds, trigger the producibility of targets. These metabolites can be found within the metabolites of the model or within a set of predefined cofactors. The selection can be made using weights, in this case the metabolites are classified by their occurrences in the models (or another scoring given by the user). The highest weights are privileged in the selection of cofactors. Inputs are metabolic model, seeds and targets as SBML and optionnally a set of defined cofactors for the selection.

Availability The MeNeTools are available on Github (`https://github.com/cfrioux/menetools`) and as a Pypi package (`https://pypi.org/project/MeneTools/`) for Python 3. More details and example for the four tools are given in Appendix C.

Application The MeNeTools rely on the graph-based definition of reaction activity $active^t_G(S)$. They are used to test this activity from the metabolites point of view. Notably, they are applied in the context of reconstructing GSMs for non-model organisms, such as EctoGEM, the model of *Ectocarpus siliculosus* described in section 4.1 of Chapter 4. They are useful to measure the added value of each reconstruction step to the quality of the model and to examine precisely the mechanisms underlying the producibltiy of some metabolites. The MeNeTools have also been applied to other metabolic networks that are not mentioned in this thesis: diatoms, bacteria, micro-algae.

7.1.2 Fluto

Fluto is a Python 3 package to perform hybrid gap-filling. This method has been introduced in Chapter 3. It uses ASP combined to Linear Programming (LP) to propose minimal sets of reactions that ensure the objective of the metabolic model satisfies graph-based and constraint-based producibility. Cplex solver (free for academic usage) is linked to the ASP solver Clingo through a constraint propagator. It takes as inputs a draft GSM and a repair GSM in which reactions will be chosen. It can also take optional topological seeds that will not be considered for constraint-based modeling.

Availability Fluto is available on Github (`https://github.com/cfrioux/fluto`) and as a Conda package (`https://anaconda.org/cfrioux/flutopy`).

Application Fluto has been applied to unblock the constraint-based activity $active^s_G(S)$ of *Chondrus crispus'* metabolic model, as described in Subsection 4.1.4 of Chapter 4.

7.1.3 Miscoto

Miscoto is a Python 3 package for community selection in microbiota. It includes the methods described in Chapter 5. The objective is to explore microbiomes and select minimal communities within them. It uses ASP to optimize community selection. Inputs are metabolic models, seeds (growth medium) and metabolic targets. Computations can be performed with a set of symbionts or a set of symbionts and a host. In the latter case, targets will be produced by the host, whereas in the former they will be produced by any member of the microbiome. It combines several tools:

- *miscoto_instance*. In a screening benchmark context, it is likely that a large number of individual targets have to be tested, which was the case in the Human Microbiome Project benchmark of Chapter 5. In such experiment, reading thousands of models at each step is not efficient since only small modifications to the ASP model will be made (change of one seed and one target for instance). A possibility is to read the data once and produce the ASP instance using *miscoto_instance*. It is human readable and can be edited a posteriori using bash commands such as sed. Inputs are models, seeds and optional targets.
- *miscoto_mincom* computes a community from a microbiome. Inputs are SBML models (symbionts and, possibly empty, host), seeds, targets or an instance pre-created with *miscoto_instance*. Options define the type of modeling that is performed:
  - "*soup*": minimal size community in a mixed-bag framework
  - "*minexch*": minimal size and minimal exchange community.

  In any case, it is possible to compute one minimal solution and/or the union, the intersection and enumeration of all minimal solutions.
- *miscoto_scope* computes the scope and target produciblity of a host (possibly empty) and the added- value of a microbiome regarding the scope and target producibility. The microbiome result part gives the targets and compounds that are producible providing cooperation occurs within the whole community and that were not producible with the host alone. Computation is made from SBML models or an instance pre-created with *miscoto_instance*.

Availability Miscoto is available on Github (`https://github.com/cfrioux/miscoto`) and as a Pypi package (`https://pypi.org/project/miscoto/`) for Python 3.

Application Miscoto has been used for the exploration of microbiotas and the selection of communities. In Chapter 5 we applied it to bacteria of the Human Microbiome Project (Section 5.2). In Section 6.1 of Chapter 6, we used Miscoto to screen the added value of gut bacteria with respect to the human host metabolism. Finally, in the Section 6.2 of the same Chapter, it was used to select communities for *Ectocarpus siliculosus*. Apart from the work presented in this thesis, Miscoto is being applied to bacterial communities of diatoms.

7.2 Integrating heterogeneous software in traceable workflows

Part of this section is extracted from the paper I, Méziane Aite, Marie Chevallier, Camille Trottier (four first coauthors) and others coauthored, published in ***PLOS Computational Biology*** and entitled *Traceability, reproducibility and wiki-exploration for "à-la-carte" reconstructions of genome-scale metabolic models* [Aite et al., 2018].

7.2.1 AuReMe workspace

Reconstructing high quality genome-scale models is a difficult task that entails using variable sources of data, extended literature analysis and several tools and methods [Aite et al., 2018]. This heterogeneity of databases and software that can be used is even more important for non-model organisms whose GSMs benefit from the exploitation of all available sources of data. In [Aite et al., 2018] we showed these dependencies and notably we observed that most published GSMs use widespread platforms for building a good quality draft GSM and afterwards refinements are made either manually or through the use of other heterogeneous methods. This provides high quality GSMs but they lack traceability: information related to the reasons why reactions were added to the model or with which method are absent. This prevents reproducibility of the GSMs. Additionally, when re-using models as templates for building GSMs of taxonomically related species, having access to traceability information is useful to support the studies. Apart from reactions added by various methods, the least standardized modifications made to models are manual refinements or manual curation. These modifications are often crucial for the functionality of the model but most of the time lost among the other reactions. A special attention needs to be given to manual refinements when they are performed outside main platforms.

In this direction, we advocated for the creation of an environment dedicated to host the "à la carte" reconstruction and refinement of GSMs, locally or remotely. This led to the development of the AuReMe workspace (`http://aureme.genouest.org/`). The objective is to start from a draft GSM build using major platforms (eg Pathway Tools [Karp et al., 2016]) or from scratch and enhance the model by using additional methods such as orthology, gap-filling or facilitated manual curations.

Docker integration for heterogeneous software The distribution of the workspace had to respond to several criteria:

- local or remote usage
- multi-platform support
- no installation by the user of the individual tools, some of which having many dependencies to be dealt with.

The chose solution is to distribute AuReMe as a Docker (`https://docker.com/`) image (`https://hub.docker.com/r/dyliss/aureme-img/`). Starting from the image, the user can create a container that will host the GSM reconstruction while communicating with the user's system in which input and output data will be available.

Internal management of traceability and metadata Metadata stored when using AuReMe are of two kinds.

- **Process metadata**. They respond to the questions "What? When? Why? How?" regarding the modifications made to the metabolic model. Types and names of tools that are used are stored and associated to the reactions they added to the model. A log file preserves the outputs of these tools and another one stores the commands performed by the user. The latter file can be used a posteriori to reproduce the GSM reconstruction starting from the same inputs. Manual curation is supervised both to facilitate modifications of the models from the user point of view and also for traceability reasons. Forms can be filled to add/delete or create a new reaction. The user can add comments and associated genes to each added reaction.
- **Biological metadata**. They concern reactions, metabolites and pathways of the model. Databases used for reconstructing models provide biological data for each object: synonyms, formulas, identifiers in other databases, molecular weights, expected taxa etc. All this information is retrieved from the database and stored in the model under creation.

Metadata are stored in a internal format that does not aim to replace existing formats for exchanging metabolic models (SBML, stoichiometric matrices) but rather to efficiently gather and retrieve information when modifying the model. This format is named PADMet (Python library for hAndling metaData ofMETabolism). It notably enables to store metadata and homogenize the treatments for several metabolic databases among which the most used MetaCyc [Caspi et al., 2016] and BiGG [King et al., 2016].

Chaining tools in reproducible GSM reconstruction pipelines Automation of GSM reconstruction is easy to be set-up. Inputs (seeds, targets, templates for orthology, proteomes etc.) are organized in a specific architecture of repositories. Based on this, pipelines that chain tools and methods can be developed using makefiles. A default pipeline exists. It includes orthology-based reconstruction that will be merged with an annotation-based model from Pathway Tools and gap-filled with Meneco (see Chapter 2 for details about Meneco). This default pipeline can be adapted to the data available for each reconstruction case.

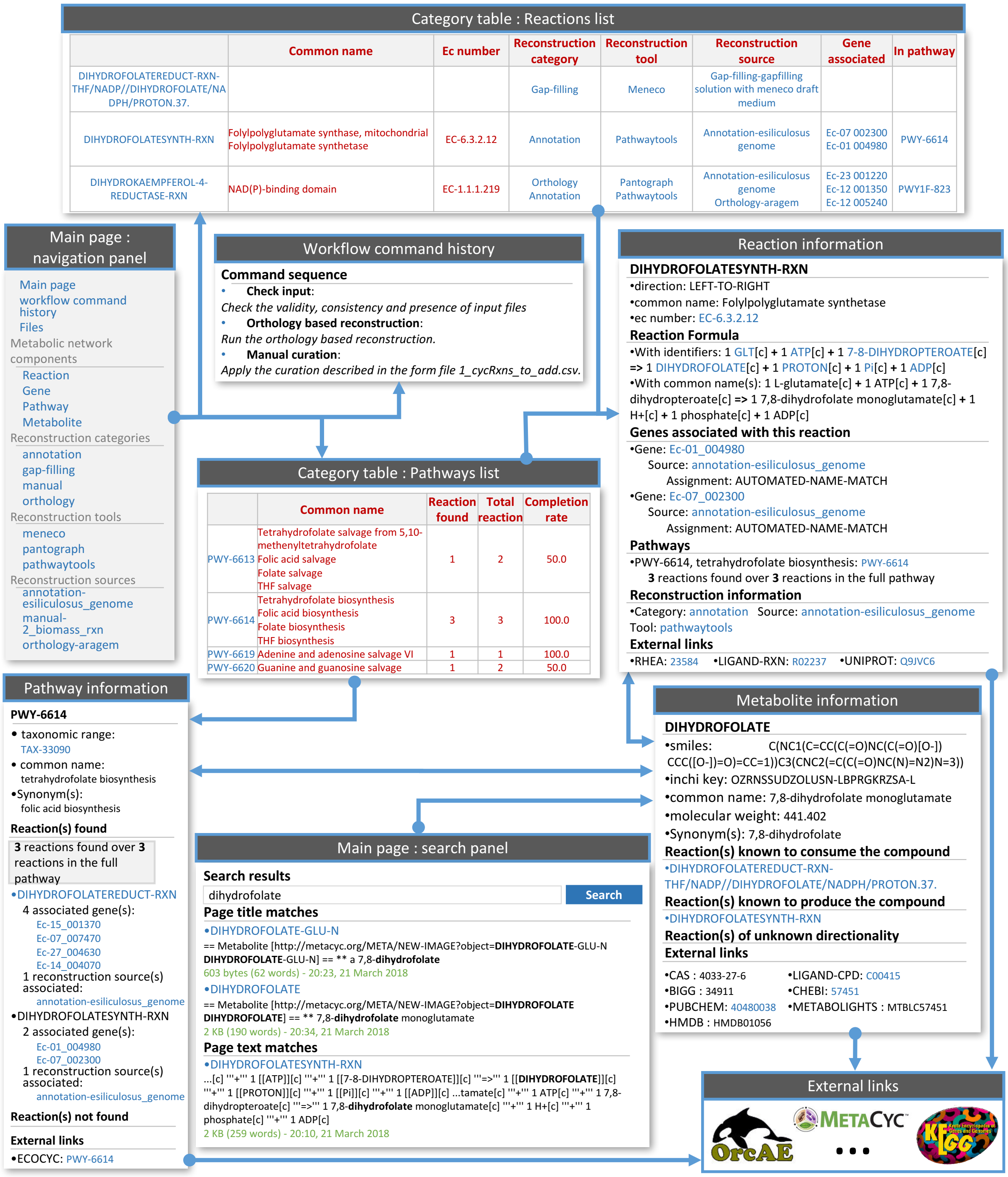
Graph-based and constraint-based analysis tools are pre-installed in the workspace. Among them are the MeNeTools that can be used between reconstruction steps to analyze the functionality of the model. Finally, the users can install other packages by themselves in the workspace and use them to suggest modifications to make to the model.

7.2.2 Providing a user view for traceability in GSM reconstruction

Wiki browsing of GSMs AuReMe efficiently tracks metadata all along the reconstruction of GSMs; yet a key question to address is how to make available this metadata to the user at several steps of the reconstruction. The solution we chose is to enable the creation a local wiki that gathers such information and can be browsed by users. Figure 7.1 provides screen captures of such a wiki for the brown alga *Ectocarpus siliculosus*. Each metabolite, pathway, gene and reaction of the model has a page that displays the object metadata as described in the database. Additional information specific to each object is also available. For instance, tools which added reactions to the model are indicated on reactions pages. Pathways of the model can be classified with their completion rates (number of pathway reactions in the model / number of pathway reactions in the full pathway, according to the database). The wiki aims at helping the user understand the GSM and the reconstruction processes.

Creation and future directions The wiki is created from a PADMet file. It is built locally by default although it is also possible to publish it online as this is the case for EctoGEM, the GSM of *Ectocarpus siliculosus* (`http://gem-aureme.irisa.fr/ectogem`). The creation of a wiki relies also on a Docker image (`https://hub.docker.com/r/dyliss/wiki-img/`).

Ongoing developments based on wikis include its application to communities of species. The purpose is to browse the complementarity of organisms' metabolism.



Category table : Reactions list

	Common name	Ec number	Reconstruction category	Reconstruction tool	Reconstruction source	Gene associated	In pathway
DIHYDROFOLATEREDUCT-RXN-THF/NADP//DIHYDROFOLATE/NADPH/PROTON.37.			Gap-filling	Meneco	Gap-filling-gapfilling solution with meneco draft medium		
DIHYDROFOLATESYNTH-RXN	Folylpolyglutamate synthase, mitochondrial Folylpolyglutamate synthetase	EC-6.3.2.12	Annotation	Pathwaytools	Annotation-esiliculosus genome	Ec-07 002300 Ec-01 004980	PWY-6614
DIHYDROKAEMPFEROL-4-REDUCTASE-RXN	NAD(P)-binding domain	EC-1.1.1.219	Orthology Annotation	Pantograph Pathwaytools	Annotation-esiliculosus genome Orthology-aragem	Ec-23 001220 Ec-12 001350 Ec-12 005240	PWY1F-823

Main page : navigation panel

Main page
workflow command history
Files
Metabolic network components
Reaction
Gene
Pathway
Metabolite
Reconstruction categories
annotation
gap-filling
manual
orthology
Reconstruction tools
meneco
pantograph
pathwaytools
Reconstruction sources
annotation-esiliculosus_genome
manual-2_biomass_rxn
orthology-aragem

Workflow command history

Command sequence

- **Check input**:

Check the validity, consistency and presence of input files

- **Orthology based reconstruction**:

Run the orthology based reconstruction.

- **Manual curation**:

Apply the curation described in the form file 1_cycRxns_to_add.csv.

Reaction information

DIHYDROFOLATESYNTH-RXN

•direction: LEFT-TO-RIGHT

•common name: Folylpolyglutamate synthetase

•ec number: EC-6.3.2.12

Reaction Formula

•With identifiers: 1 GLT[c] + 1 ATP[c] + 1 7-8-DIHYDROPTEROATE[c] => 1 DIHYDROFOLATE[c] + 1 PROTON[c] + 1 Pi[c] + 1 ADP[c]

•With common name(s): 1 L-glutamate[c] + 1 ATP[c] + 1 7,8-dihydropteroate[c] => 1 7,8-dihydrofolate monoglutamate[c] + 1 H+[c] + 1 phosphate[c] + 1 ADP[c]

Genes associated with this reaction

•Gene: Ec-01_004980
Source: annotation-esiliculosus_genome
Assignment: AUTOMATED-NAME-MATCH

•Gene: Ec-07_002300
Source: annotation-esiliculosus_genome
Assignment: AUTOMATED-NAME-MATCH

Pathways

•PWY-6614, tetrahydrofolate biosynthesis: PWY-6614
3 reactions found over **3** reactions in the full pathway

Reconstruction information

•Category: annotation Source: annotation-esiliculosus_genome
Tool: pathwaytools

External links

•RHEA: 23584 •LIGAND-RXN: R02237 •UNIPROT: Q9JVC6

Category table : Pathways list

	Common name	Reaction found	Total reaction	Completion rate
PWY-6613	Tetrahydrofolate salvage from 5,10-methenyltetrahydrofolate Folic acid salvage Folate salvage THF salvage	1	2	50.0
PWY-6614	Tetrahydrofolate biosynthesis Folic acid biosynthesis Folate biosynthesis THF biosynthesis	3	3	100.0
PWY-6619	Adenine and adenosine salvage VI	1	1	100.0
PWY-6620	Guanine and guanosine salvage	1	2	50.0

Pathway information

PWY-6614

- taxonomic range:
TAX-33090
- common name:
tetrahydrofolate biosynthesis

•Synonym(s):
folic acid biosynthesis

Reaction(s) found

3 reactions found over **3** reactions in the full pathway

•DIHYDROFOLATEREDUCT-RXN
4 associated gene(s):
Ec-15_001370
Ec-07_007470
Ec-27_004630
Ec-14_004070
1 reconstruction source(s) associated:
annotation-esiliculosus_genome

•DIHYDROFOLATESYNTH-RXN
2 associated gene(s):
Ec-01_004980
Ec-07_002300
1 reconstruction source(s) associated:
annotation-esiliculosus_genome

Reaction(s) not found

External links

•ECOCYC: PWY-6614

Metabolite information

DIHYDROFOLATE

•smiles: C(NC1(C=CC(C(=O)NC(C(=O)[O-])CCC([O-])=O)=CC=1))C3(CNC2(=C(C(=O)NC(N)=N2)N=3))

•inchi key: OZRNSSUDZOLUSN-LBPRGKRZSA-L

•common name: 7,8-dihydrofolate monoglutamate

•molecular weight: 441.402

•Synonym(s): 7,8-dihydrofolate

Reaction(s) known to consume the compound

•DIHYDROFOLATEREDUCT-RXN-THF/NADP//DIHYDROFOLATE/NADPH/PROTON.37.

Reaction(s) known to produce the compound

•DIHYDROFOLATESYNTH-RXN

Reaction(s) of unknown directionality

External links

•CAS : 4033-27-6	•LIGAND-CPD: C00415
•BIGG : 34911	•CHEBI: 57451
•PUBCHEM: 40480038	•METABOLIGHTS : MTBLC57451
•HMDB : HMDB01056	

Main page : search panel

Search results

dihydrofolate

Page title matches

•DIHYDROFOLATE-GLU-N
== Metabolite [http://metacyc.org/META/NEW-IMAGE?object=**DIHYDROFOLATE**-GLU-N **DIHYDROFOLATE**-GLU-N] == ** a 7,8-**dihydrofolate**
603 bytes (62 words) - 20:23, 21 March 2018

•DIHYDROFOLATE
== Metabolite [http://metacyc.org/META/NEW-IMAGE?object=**DIHYDROFOLATE** **DIHYDROFOLATE**] == ** 7,8-**dihydrofolate** monoglutamate
2 KB (190 words) - 20:34, 21 March 2018

Page text matches

•DIHYDROFOLATESYNTH-RXN
...[c] '''+''' 1 [[ATP]][c] '''+''' 1 [[7-8-DIHYDROPTEROATE]][c] '''=>''' 1 [[**DIHYDROFOLATE**]][c] '''+''' 1 [[PROTON]][c] '''+''' 1 [[Pi]][c] '''+''' 1 [[ADP]][c] ...tamate[c] '''+''' 1 ATP[c] '''+''' 1 7,8-dihydropteroate[c] '''=>''' 1 7,8-**dihydrofolate** monoglutamate[c] '''+''' 1 H+[c] '''+''' 1 phosphate[c] '''+''' 1 ADP[c]
2 KB (259 words) - 20:10, 21 March 2018

External links

Figure 7.1: ***Screen captures of several pages of the local wiki and the interactions between them***

A local wiki-based export of the GSM facilitates user-interface exploration and traceability of the reconstruction procedure. Several screenshots of a wiki are displayed, arrows represent the link between pages. Notably, reactions can be sorted and explored according to reconstruction categories, tools and sources. The navigation panel enables exploring and comparing the contributions of each tool used in the "à la carte" GSM reconstruction pipeline. Pathways can be sorted based on their completion rate.

Conclusion

Provide **user-friendly and useful software for biologists and computational biologists** is an important objective of our work. My contributions to this objective are twofold. **I first developed several tools for studying and refining metabolic networks under the graph-based or hybrid semantics of activation**. The MeNeTools analyze the topology and functionality of metabolic networks starting from nutrients. Fluto performs hybrid gap-filling. Miscoto enables to select minimal communities within microbiotas, under two levels of modeling. All these tools are based on Answer Set Programming and were packaged in Python in order to make these dependencies as much transparent to the user as possible. This is a requisite point to ensure the distribution and easy use of the associated software, especially if the target audience is not familiar to computer science. In the same direction, ensuring that inputs to the software are generic or widespread formats has to be taken into account during the development of software, this is why SBML is chosen here.

I introduced the AuReMe local workspace in the second part of this chapter. It is dedicated to **host the reconstruction or refinements of metabolic models to ensure traceability and reproducibility of the modifications** that are done. It used Docker to prevent the possibly troublesome installing of bioinformatics tools. It can be used to perform a *de novo* reconstruction or to pursue a reconstruction that was already started with one of the main platforms. We showed that heterogeneous software is used for reconstructing quality GSMs and that it is a threat to reproducibility of models in the sense that, in many of them, no information indicates which tool or method led to the addition of reactions, nor the targets and reasons of manual curation. These are issues that we fix with AuReMe. **Metadata that contains this information is stored all along the reconstruction**. Together with the biological metadata that contains notably links to databases of knowledge, all the information is easily retrieved by users at any time of the reconstruction through the creation of local wikis. Browsing them enable users to track the elements of metabolic models and to deeper explore pathways of interest.

Conclusion

The objective of this thesis was to propose combinatorial and optimization methods adapted to non-model organisms for the refinement of metabolic networks (GSMs) and the selection of communities in microbiotas. Having in mind that such work is meant to explain biological processes, we applied our methods on realistic data and eventually performed first experimental testing of our hypotheses and predictions.

The entry point of this thesis was that several **semantics to model the functionality and producibility in metabolic networks** have been developed for the last two decades. We focus on the general notion of activation from available metabolites called seeds. We elaborated on this general definition with existing producibility semantics to come up with the complementary graph-based [Ebenhöh et al., 2004] and constraint-based [Orth et al., 2010] activations of metabolism. These definitions of activation are a common thread along this thesis, notably the graph-based one that better supports incomplete data and knowledge as faced when studying non-model organisms, for which data is sparse [Russell et al., 2017]. We used them in various methods dedicated to reconstruct and analyze the metabolism of individual organisms, or collectively study hosts and their microbiotas.

Combinatorial and hybrid gap-filling

The first part of this thesis concerns **advances in metabolic gap-filling for non-model organisms**. This step consists in completing metabolic networks during their reconstruction in order to make them able to meet a defined objective, in most cases the production of biomass, in a chosen functionality semantics. In **Chapter 2** (Flexibility and accuracy of graph-based gap-filling), we first **validated a graph-based gap-filling method based on Answer Set Programming** (ASP), Meneco [Prigent et al., 2017], by comparing its performances to the ones of constraint-based gap-filling techniques [Satish Kumar et al., 2007, Vitkin and Shlomi, 2012, Thiele et al., 2014]. The ability of Meneco to both sample the space of solutions and propose small sets of reactions is valuable. Therefore, the parsimonious characteristic of Meneco appeared to be an asset in practice as we promote gap-filling to suggest reactions that will be validated by experts and associated to genes *a posteriori*.

A lesson of our benchmarking study is that every gap-filling algorithm varies in its results and that choosing a universal method is not necessarily appropriate. An illustration is that all major platforms for GSM reconstruction own their proper algorithm for this purpose. Automatic gap-filling has drawbacks and completions have to be curated for ensuring a good quality GSM [Faria et al., 2018]. Meneco is well-suited for GSMs with a reasonable degradation rate: it shows satisfying results for restoring constraint-based activation of the objective function and its solutions are small enough to be manually curated. More generally, its flexibility in the objective to be optimized is an asset for targeted gap-filling.

Therefore, the main contributions of Chapter 2 are in the field of computational biology. We learned *that combinatorial problems can propose relevant reactions for a flexible refinement of metabolic networks with respect to constraint-based methods. Solutions provided by Meneco are well-suited for reasonably degraded GSMs and curation is facilitated thanks to its parsimonious criterion.*

An observation from Chapter 2 is that for highly degraded models, Meneco's solutions are less compliant with FBA. We thus proposed to **extend the graph-based gap-filling with linear programming constraint ensuring the satisfiability of FBA constraints** in **Chapter 3** (Hybrid gap-filling reconciles graph-based and constraint-based formalisms). Technically, the association of ASP and linear constraints is made feasible with a propagator that enables to test the linear constraints satisfiability of models raised by ASP with a LP solver. Applied to gap-filling, this ensures that the completions are compliant with both the graph-based and constraint-based activation of reactions. This hybrid gap-filling has been implemented in Fluto. Its ability to restore functionality according to both criteria was verified using the same *E. coli* benchmark as for Meneco above. This demonstrates that both semantics can be reconciled through the notion of reaction activation and that graph-based semantics can be an asset to facilitate the solving of constraint-based problems. A limit so far in this work is the relatively small size of the database used for completing the models. It has the size of a GSM and not of a full database of metabolic knowledge that is five to ten times bigger. This concern will be addressed in the perspectives of the thesis. In the meantime, Fluto is usable as is, for unblocking precise functions of interest using a small database, possibly another GSM, as presented in Chapter 4.

To sum up, the main contributions of Chapter 3 are in the field of computer science. A Linear Programming constraint propagator was applied to the gap-filling problem and enables to combine graph-based and constraint-based activations to refine GSMs.

Applications

The two gap-filling methods were applied to real case studies, on seaweeds (Chapter 4). First we showed the impact of gap-filling and the interest of the graph-based one for the **GSM reconstruction of *Ectocarpus siliculosus* (EctoGEM) [Aite et al., 2018] following the re-annotation of its genome**. We observed that using complementary methods which take the most out of available data enables to restore functionality of the model with respect to target and biomass production and also to the topological completion of metabolic pathways. We then showed that once the GSM is reconstructed, it can serve as a basis for other metabolic networks. The GSM of *Chondrus crispus* was incapable of producing alanine under the constraint-based semantics. We used **Fluto to gap-fill the red alga model** with EctoGEM as a database of reactions. This unblocked the production of alanine and demonstrated the practical interest of using hybrid gap-filling for specific objectives.

Finally, we **derived the gap-filling theory to find complementarity between the GSM of *E. siliculosus* and its symbiotic non cultivable bacterium** *Candidatus* Phaeomarinobacter ectocarpi [Dittami et al., 2014a, Prigent et al., 2017]. Starting from both GSMs, the contents of algal growth medium and a large set of transcriptomics-based metabolic targets, Meneco was used to compute the metabolites whose algal producibility could depend on the bacterium. It was followed by a thorough curation of predictions, performed by biologists to remove false positive interactions. Eventually, nineteen compounds were classified as resulting from putative algal-bacterial interactions. This demonstrates that the search for interactions between organisms is a problem that can be addressed with combinatorial methods such as gap-filling.

These results also show that reconstructing GSMs while considering the isolated organism can be reductionist and lead to the addition of reactions that are possibly catalyzed externally by symbionts. This has to be taken into account during reconstruction processes. Therefore, studying communities using GSMs that do not possess reactions without genetic support ensures that less putative interactions will be missed. An evident cornerstone is that it can also lead to falsely assume that some functions rely on cooperation to be catalyzed. Yet, I advocate that no method can do the whole process (from GSM reconstruction to interaction prediction) without curation so far. In this direction, either the curation has to occur at the gap-filling step to ensure the reactions added to the model have a genetic support and exist in the species, or automatic gap-filling is ignored and the curation will have to occur later to discriminate the putative cooperation-based functions. In the case of non-model organisms in large microbiotas, it is not realistic to curate every bacterial GSM, so the second choice appears to be more tractable in practice.

To sum up, the main contributions of Chapter 4 are biological. The GSM of Ectocarpus siliculosus was enhanced, it was used to unblock a pathway of interest in Chondrus crispus' one. Finally, some compounds of interest were identified (histidine, beta-alanine, agmatine) as being the result of cooperation processes between E. siliculosus and its associated bacterium.

Selection of communities

We pursued the work on communities with the second results part of the thesis: **Scalability and combinatorics of community selection** with a first chapter of results (Chapter 5: Formalism and modeling of the community selection problem). We re-used gap-filling theory and graph-based semantics to **formalize a community selection problem solved with ASP**. We addressed the problem of selecting minimal communities of symbionts in a large microbiota to meet a metabolic objective with the following constraints. The community is expected to be **minimal in size and in terms of needed cooperation** ie metabolic exchanges. For this purpose, we proposed a two step process that enables to first remove a large part of the microbiota by only retrieving symbionts that belong to minimal-size communities. This problem is easy to solve provided a simplification of the formalism is accepted: by ignoring the nature and number of exchanges and considering a meta-organism or "mixed-bag". It is very similar to a gap-filling problem in which there are several databases, each corresponding to one symbiont. The objective of such gap-filling problem would be to minimize the number of databases in which reactions will be picked. ASP solving strategies and graph-based semantics enable to compute all communities or directly the union of all of them, that is to say the set of symbionts that appear in at least one community.

In a second step, a compartmentalized formalism addresses the exchanges that might be needed in the community and they can also be minimized. This can be done by computing minimal exchanges on every community obtained at the first step or by directly selecting the minimal-size minimal-exchanges communities starting with the union of symbionts previously calculated. This can be viewed as a second gap-filling like problem in which reactions are picked and not databases (organisms) as in the previous step. These tools were implemented in the Miscoto Python package. Let us emphasize the fact that the optimizations to minimize size and exchanges are not commutative. The optimizations are ruled by cardinality, hence the first step can possibly eliminate communities that would ensure the functions with fewer exchanges but more bacteria. This is a challenge to be addressed with the wide range of existing optimization heuristics in ASP and it is addressed in the perspectives of the thesis. [Julien-Laferrière et al., 2016] used ASP for synthetic consortia design with a similar concept. They chose bacteria among a small set of species by minimizing the transports and the exogenous reactions. They did not select communities based on a size criterion. This tool could also be applied in the second step of the workflow, but the combinatorics of transport reactions is so massive that it cannot be applied in a large scale microbiota without pre-filtering the species.

We tested our Miscoto workflow on the **Human Microbiome Project** data by selecting minimal communities for pairs of seed and target metabolites within more than two thousands bacterial GSMs. We showed that the functional redundancy of microbiotas is reflected by the high number of minimal communities for each function. Yet the benefit of combining both optimizations significantly reduces the number of bacteria and communities to be eventually curated by experts. And more importantly, this process enables to capture the whole combinatorics of communities and prevents missing information. Indeed we advocate for a community selection process that gives the final word to biologists who can *a posteriori* filtrate the solutions by applying extra criteria that are intrinsic to experimentation: growth incompatibilities between several strains, difficulty to work with others etc. With this benchmark, we refined the work performed by [Eng and Borenstein, 2016] who tested the existence of metabolic pathways for 10,000 random pairs of product and substrate and who did not find one for more than 97% of them. We showed that one pathway exists in 23% of them and testing their network-flow based algorithm on these pairs showed the same results, although only one solution was provided. This demonstrates the assets of using the graph-based semantics: the scalability concern of the size of the microbiota can be addressed, and the exhaustiveness of solutions can be obtained with an adequate ASP solver.

To sum up, the main contributions of Chapter 5 are in the fields of computer science and computational biology. A heuristic in two steps was proposed to address the problem of minimal community selection in large microbiotas. Its solving was performed with logic programming optimizations. On the computational biology side, a benchmark was set-up on the HMP dataset: the effect of the two-step approach was assessed.

Applications

We then extended and applied our work in **Chapter 6 Applications of community selection algorithms**. We studied once again the **gut microbiota**, but this time **by considering the human host Recon 2.2** [Swainston et al., 2016] **and 773 curated GSMs of the AGORA project** [Magnúsdóttir et al., 2016] **under strict nutritional conditions** that would mimic cell culture of enterocytes, intestinal absorptive cells. We computed all minimal-size communities that enable to the human GSM to produce the maximum number of cytosolic compounds. The association of the 89 bacteria that belong to the 381 communities was analyzed with clustering, graph and power graph analyses. We show that such analyses applied to the entirety of communities can discriminate the importance of bacteria and pinpoint equivalence groups of symbionts within the microbiota. We then computed the ability of each bacterium to unblock the human producibility of each target either with a minimal-size criterion only, or combined with the minimal-exchanges criterion. This enabled to better understand the clustering of bacteria through the target dependencies of each group. This work aims at demonstrating the possibilities offered by such workflows. It can provide work hypotheses for very flexible objectives (individual targets, whole screening of cytosolic metabolites, reactants of biomass reactions etc.) and rationales for exploring and analyzing the diversity of communities.

A similar work was performed in the project of **selecting bacterial communities to test and understand the metabolic dependencies of *Ectocarpus siliculosus* in its holobiont**. As *Ca.* P. ectocarpi, the bacterium with which the putative interactions were computed in Chapter 4, is not cultivable, we relied on the functional redundancy that exist in microbiota to select bacteria among cultivable ones to test dependencies. The communities predicted by Miscoto were tested experimentally with coworkers of Sorbonne Université. Preliminary results show improvements in growth with bacteria compared to the isolated alga. Metabolomics analyses demonstrates that 7 out of the 8 tested metabolites are found only in non-axenic cultures. Altogether this first collaboration of computational biologists and biologists on the prediction of communities for seaweed is promising and can form a basis for future experimentations. It also enables to shed light on the possible limits of the predictions performed. A first one is the absence of competition in the parameters taken into account for modeling. It will be discussed in the perspectives. A second one is the difficulty to control the bacterial communities in cultures. Antibiotics prior to experiments alter the microbiota but cannot reasonably delete any bacterium of the culture. Thus it leads to empty microbial niches that may not be filled by the bacteria added by experimenters but by others that were still present at that time. Indeed, metagenomics studies on some microbiota show that some genera or phyla are in larger abundance than others; bacteria are not even in their distribution in microbiotas [Arumugam et al., 2011, KleinJan et al., 2017]. Thus, taking into account the abundance of the bacteria in the native microbiota when selecting communities, although not ensuring the abundance will be similar if the communities are altered, might be interesting to decipher the most suitable communities.

To sum up, the main contributions of Chapter 6 are biological. Bacteria of interest were identified in the gut, as well as key metabolites to discriminate their roles. Regarding marine biology, communities expected to be valuable for Ectocarpus siliculosus were selected in silico and tested experimentally.

In a final **Chapter 7, Integrating heterogeneous bioinformatics software in traceable workflows**, I presented the software I developed or contributed to, during my PhD. Packaging the ASP-based tools is necessary for ensuring that they can easily be used in practice. In that sense, I presented the MeNeTools for graph-based analysis of metabolic networks, Fluto for hybrid gap-filling, and Miscoto for selecting communities. I discussed the use and integration of ASP solvers within packages and the importance of facilitating them for user-friendliness purposes. In parallel, a joint work on traceability and reproducibility in GSM reconstruction has been performed, leading to the development of AuReMe.

We showed in this thesis the **applicability of combinatorial techniques** based on gap-filling theories to study the metabolism of non-model organisms, and microbiotas. We established that methods relying on graph-based metabolic activation are adapted for screening and solution space sampling, through the efficiency of logic programming paradigms such as ASP. They are more scalable and resilient to imprecise data than other constraint-based techniques due to their graph-based semantics which make them particularly fitted to first line analyses and scale reduction. The results can then be refined with the quantitative semantics, or with expert curation. We additionally demonstrated that an association of combinatorial techniques with linear programming is applicable in practice to solve problems such as gap-filling. Finally, we proved that combinatorial methods can be used as prediction means for experimental testing.

Perpectives

The works of this thesis can be extended with several perspective proposals that differ into the domain they involve: computer science, computational biology and modeling, and biological applications. These perspectives can also be discriminated in short, medium or long-term based on the priority that should be given to them.

Computer science perspectives

Going further with the work of this thesis from the computer science aspect entails to propose improvements related to Answer Set Programming (ASP) whose constant development and solver enhancement can be used to the profit of optimization problems in biological modeling. A first point can be directed to **Fluto** through the **improvement of its computational performances to large databases**. Fluto is the hybrid graph-based/constraint-based gap-filling method developed during my thesis. It was shown to be a useful complement to Meneco, the graph-based gap-filling tool, to unblock the producibility of alanine in *Chondrus crispus'* GSM (Chapter 4). It was used with another GSM as a reaction database and proposed a reaction from it, to ensure the production of alanine is constraint-based activated (Flux Balance Analysis). The possibility to use Fluto as a stand-alone gap-filling method however is limited due to the computational time required to scale to large databases of reactions that can contain more than ten thousand items [Karp et al., 2017]. Meneco can support such scaling much easier that its hybrid counterpart. The switch from databases of the range of one thousand reactions to ten thousand reactions can be facilitated by applying heuristics in the code that filters the reactions that are not interesting with respect to the objective to optimize. This is done in Meneco with graph-based criteria and could possibly be extended to Fluto by eliminating more rapidly reactions that are not of interest in either semantics. This objective, although not pressing for Fluto so far if its usage remains with relatively small databases, could benefit to any other hybrid tool that could be developed in a near future.

The two following computer science perspectives of this thesis aim to benefit from the latest **optimization** conveniences in ASP for an application to our biology-related problems and their solving. First, dealing with **optimization priority** would be an interesting short term objective. So far, the optimizations performed in the community selection with Miscoto follow a priority rank based on cardinality; the highest priority being to maximize the number of producible targets, then to minimize the size of the community and eventually the number of exchanges. An example of limitation raised by such ranking is the following case. Miscoto would select a two-species community that requires 3 exchanges for meeting the objective rather than a three-species community requiring 2 exchanges (two bacteria provide one precursor to a third one). It is delicate to compare the quality of these two solutions based only on these numbers. A good solution would be to consider both and filtrate them a posteriori with additional criteria. A new optimization system to be set up here would therefore to use **pareto** optimization [Ehrgott, 2005], with some or all the optimizations used in Miscoto. Answer Set Programming can support pareto optimization [Brewka et al., 2015]. Pareto optimization has already been applied to the study of metabolic models and/or communities [Zakrzewski et al., 2012, Schuetz et al., 2012, Budinich et al., 2017]. Applying it to the selection of communities would possibly propose interesting results that are currently missed by current optimizations. Yet, the effect of the optimization change will need to be addressed in terms of complexity. Indeed, the two-step approach used so far in Miscoto's community selection was a good solution to get through the complexity raised by the huge combinatorics of the problem. The effect of pareto on solving performances will need to be established. Nevertheless, these optimization heuristics could also benefit the addition of additional criteria in microbiota study, notably the analysis of competition that will be discussed in the following section.

More generally, fluxes can help choosing between all minimal communities of bacteria selected by Miscoto. They could be integrated in the process of community selection as another short term objective related to optimization. Relying on hybrid activation could therefore be considered. This thesis has demonstrated the interest of graph-based semantics in a field that is still strongly dominated by the constraint-based one. I believe that combining both is of high interest and in that sense, hybrid semantics can be applied to many problems apart from gap-filling which has been done in this thesis. For instance, the selection of community and more generally **the analysis of microbiota could be extended to support the hybrid semantics**. So far, the prediction of communities with Miscoto implies the graph-based semantics of metabolic activation. We are aware the absence of constraint-based semantics can be seen as a major limitation and we advocated that flux-based modeling of the communities can be done after the selection we provide, to filtrate the solutions. We can imagine the application of hybrid graph-based/constraint-based semantics to the community selection problem, as both have been individually applied to the study of microbiota [Julien-Laferrière et al., 2016, Frioux et al., 2018a, Opatovsky et al., 2018, Eng and Borenstein, 2016, Zomorrodi et al., 2014]. In particular, flux constraints could be applied to the putative transports that would ensure the exchanges predicted by Miscoto are feasible. Indeed, since several sets of minimal exchanges often co-exist for a minimal community, decision between them could be contemplated with fluxes by testing their distribution into the exchange reactions selected by Miscoto.

Altogether, the future work related to the optimization problems relies on ASP technologies and improvements that have not been applied so far to biological applications. They could respond to problems that exist in the domain, particularly for the design of tailor-made optimizations that take better in account the complex biological reality.

Enhance the biological relevance of predictions

My research is strongly motivated by biological questions and I want the models I propose to guide hypotheses and experimentation so that the range of initial conjectures can be lowered or weighted. To that purpose, adequate data must be obtained and more importantly smartly integrated into models, which is a limit when organisms are not well studied (impossibility to growth them individually, Non-Model Organisms (NMOs) etc.). In the context of microbiota selection such as for *Ectocarpus siliculosus*, the pertinence of predictions could certainly be enhanced. A key aspect of computational biology according to me is to build a bridge between hypotheses and experimentation. Being able to validate the hypotheses raised by modeling is crucial for better understanding physiology, thus providing hypotheses of quality is what I believe to be the main duty of a computational biologist.

The joint work on the prediction of communities for *Ectocarpus siliculosus* and its *in-vitro* testing (Chapter 6) is an opportunity that not all modelers have. It puts in perspective the differences between what is taken into account for modeling and what really happens biologically. Results are encouraging with this experiment: added value of bacteria is shown over the effect on algal growth, and in metabolomics. Yet metabolomics results are not completely in accordance with the predictions, and sequencing shows that controlling the diversity of communities and bacteria is very difficult. Experimentation therefore shows the fallibility of models and can pinpoint their weaknesses. In the case of *Ectocarpus siliculosus*, the probable main weakness that can emerge from experimentation is the ignorance of putative competition events. From the experimental point of view, proving interactions within communities of non-model organisms is difficult to set-up; and in particular the precise identification of exchanges. I believe the models could be enhanced by integrating the observations made after these first experiments. Comparing what was really observed with respect to what was predicted is crucial to enhance the model, so are the discussions with biologists to obtain hypotheses. For instance, in our experiments, some bacteria or some metabolites were not retrieved in the cultures that there were expected to be found in. Thus adapting the inputs to the model based on what was observed could give hints for explanation. In any way, being able to work even closer in collaboration with experimenters is a strong asset. The context of the experiments and other constraints that are unique to these settings (growth difficulty for certain strains, presumed incompatibilities for others etc.) should be added easily to the existing constraints of the model, all of this requiring flexibility. This is in my opinion an important objective of the future work in microbiota selection. Regarding the application, experiments and predictions need to be carried on for *Ectocarpus*. Among other organisms of interest, diatoms and their dependencies toward the microbiota are also currently under study.

The biology of community selection is also tightly related to the **transports of metabolites between species**. A cornerstone in interaction modeling is indeed the accuracy of exchanges prediction. The optimizations presented in Miscoto (Chapter 5) do not take into account the existence of transports in the model. An additional optimization about existing transports has been implemented ever since. Yet it does not raise different results to the application on gut microbiota. No transport reactions are annotated in the models from the Human Microbiome Project. More generally, the characterized transport reactions in non-model organisms GSMs is very sparse. There are initiatives to predict transports from genomic data or associate genes to transports identified with literature or experimental evidence [Elbourne et al., 2017, Sung et al., 2017, Dias et al., 2017]. Some limitations reside in the genericity of identified transports. A question that can be asked is whether a transport reaction should be derived for all metabolites related to the concerned compound classes with the risk of leading to false positives. The improvement of selection methods is thus tightly pertained to the improvement of transport identification in GSMs for non-model organisms.

Finally, an important biological question still remains in the **definition of objectives for the community**. So far in Miscoto, the objective is described as a set of target metabolites, whose producibility, for symbionts or the host, has to be ensured through community cooperation. Generally, the optimization of biomass production is the main criterion applied as an objective for the reconstruction of individual GSMs, and a combination of these biomass is optimized for community modeling [Zomorrodi and Maranas, 2012, Zomorrodi et al., 2014]. The composition of this reaction and its associated stoichiometry are not trivial to be deciphered for non-model organisms. This composition derives in most cases from the composition of biomass in a few well-studied organisms [Xavier et al., 2017]. In community modeling, it thus can be of interest to set-up other objectives in addition to, or in place of, the biomass maximization. Here, with Miscoto, we chose flexibility with a target list that does not depend on reactions. An example that has been set-up by others is to combine energetic yields (ATP) with biomass [Schuetz et al., 2007], which could be tested for *Ectocarpus*. Adding some flexibility into the objective that are optimized in community selection and comparing the results deserves to be carried out and it is consistent with the computer science perspective presented above about the work on solving heuristics and optimizations.

Computational biology and modeling perspectives

From the computational biology or bioinformatics outlook, my future work aims to improve and extend the modeling of communities on two aspects: take into accounts competition events for the selection of communities, and work on the effect of data incompleteness with respect to the resiliency of the predictions for non-model organisms. The absence of experimental studies on some non-model organisms for which only genetic sequences are known is a drawback that has already been discussed in this thesis. It leads to models that are evidently not as trustworthy as those for well-studied species, and for which the limitations of good models apply even more. For instance, there are two main limitations of GSMs built on genome annotation. The presence of the gene in the genome does not imply i) its expression physiologically, ii) nor that the enzyme, derived from the gene, catalyzes the exact reaction as the one added to the GSM. Genome annotation for NMOs can be troublesome and thus lead to lower-quality GSMs. Having such limitations in mind is important to favour the constant enhancement of modeling, notably to better support the gaps in GSMs. In this thesis, it was chosen to use non gap-filled GSMs for the computation of community selection in order to prevent functions, performed through cooperation, to be falsely added into the individual GSMs. This has the opposite shortcoming that is to predict too much functions than what is likely to result from cooperation, and consequently necessitates to filter them a posteriori. These gaps thus alter the observed functionality of the metabolisms. A solution is to compensate them by modulating the seeds such that their effect is reduced.

Indeed, the functionality of organisms, either characterized with the graph-based or the constraint-based semantics, is highly dependent to the inputs to the system. These inputs are mostly represented by growth media but modeling issues and gaps in GSMs also lead to consider additional metabolites as available, notably cofactors [Julien-Laferrière et al., 2016, Eng and Borenstein, 2016, Greenblum et al., 2012, Cottret et al., 2010]. Some inputs (import reactions) can also be added to unblock flux distribution within metabolic models [Thiele et al., 2014]. When associating metabolic networks to model communities, these inputs are even more important as they combine with exchanges to modify the functionality and behaviour of other models. Selection of medium for communities is thus a field of interest. The work of [Klitgord and Segrè, 2010] enables to select growth media that would induce symbiosis between species. In the direction of identifying environments for co-growing organisms, [Zarecki et al., 2014] propose the MENTO algorithm. It takes molecular weights of compounds into account; they are minimized in the medium. The authors raise an interesting side effect of minimizing the number of medium components: it leads to the selection of complex metabolites that can be sources of many nutritional needs for organisms but are unrealistic to be seeds biologically. Pursuing the work on the effects and dependencies of inputs on metabolic behaviours is of interest to bypass the gaps in models and improve their resilience to incomplete data. Technically, we could consider computing an optimal medium for cooperation with existing techniques presented above; retrieving which compounds are actually used among the seeds with our current modeling; or observe seeds transports reactions that carry flux in the community for specific objectives [Rose and Mazat, 2018].

Finally, a major perspective of the thesis, that has been mentioned previously is to take into account competition for the selection of community. We discussed the absence of competition in the parameters taken into account by Miscoto. Such events as well as the possible degradation of compounds in the medium that would impair exchanges [Eng and Borenstein, 2016] have to be considered. Metrics already exist to assess competition potential between species. They are based on the similarity between their GSMs or on the calculation of their seeds ie what metabolites are needed to activate their reactions [Kreimer et al., 2012, Mendes-Soares et al., 2016]. Such methods can be associated to the optimizations for community selection, entailing to modify the current optimization heuristics. First the competition has to be modeled so that an objective can be derived from it (e.g. minimize a competition score within the selected community). Then the objective related to competition has to be combined with the existing ones (production of targets, minimization of community size and exchange number). So far the objectives were suitable for a cardinal ordering. It is however difficult to give a cardinal priority to competition with respect to the other objectives, thus the optimization heuristics (preferences, pareto etc.) that exist in ASP have to be studied for taking competition into account in community selection.

List of Figures

Acronyms

ASP Answer Set Programming.

CB Constraint-Based.

FBA Flux Balance Analysis.

FVA Flux Variability Analysis.

GB Graph-Based.

GSM Genome-Scale Model.

HMP Human Microbiome Project.

ILP Integer Linear Programming.

LP Linear Programming.

LPNMR Logic Programming and Nonmonotonic Reasoning.

MILP Mixed Integer Linear Programming.

NMO Non-Model Organism.

OTU Operational Taxonomic Unit.

TIC Thermodynamically Infeasible Cycles.

Bibliography

[Abubucker et al., 2012] Abubucker, S., Segata, N., Goll, J., Schubert, A. M., Izard, J., Cantarel, B. L., Rodriguez-Mueller, B., Zucker, J., Thiagarajan, M., Henrissat, B., White, O., Kelley, S. T., Meth??, B., Schloss, P. D., Gevers, D., Mitreva, M., and Huttenhower, C. (2012). Metabolic reconstruction for metagenomic data and its application to the human microbiome. *PLoS Computational Biology*, 8(6):e1002358.

[Aite et al., 2018] Aite, M., Chevallier, M., Frioux, C., Trottier, C., Got, J., Cortés, M. P., Mendoza, S. N., Carrier, G., Dameron, O., Guillaudeux, N., Latorre, M., Loira, N., Markov, G. V., Maass, A., and Siegel, A. (2018). Traceability, reproducibility and wiki-exploration for "à-la-carte" reconstructions of genome-scale metabolic models. *PLOS Computational Biology*, 14(5):e1006146.

[Allison and Martiny, 2008] Allison, S. D. and Martiny, J. B. H. (2008). Resistance, resilience, and redundancy in microbial communities. *Proceedings of the National Academy of Sciences of the United States of America*, 105 Suppl(Supplement_1):11512–9.

[Amin et al., 2015] Amin, S. A., Hmelo, L. R., Van Tol, H. M., Durham, B. P., Carlson, L. T., Heal, K. R., Morales, R. L., Berthiaume, C. T., Parker, M. S., Djunaedi, B., Ingalls, A. E., Parsek, M. R., Moran, M. A., and Armbrust, E. V. (2015). Interaction and signalling between a cosmopolitan phytoplankton and associated bacteria. *Nature*, 522(7554):98–101.

[Ankrah et al., 2017] Ankrah, N. Y., Luan, J., and Douglas, A. E. (2017). Cooperative metabolism in a three-partner insect-bacterial symbiosis revealed by metabolic modeling. *Journal of Bacteriology*, pages JB.00872–16.

[Ansótegui et al., 2013] Ansótegui, C., Bonet, M., and Levy, J. (2013). SAT-based MaxSAT algorithms. *Artificial Intelligence*, 196:77–105.

[Arkin et al., 2016] Arkin, A. P., Stevens, R. L., Cottingham, R. W., Maslov, S., Henry, C. S., Dehal, P., Ware, D., Perez, F., Harris, N. L., Canon, S., Sneddon, M. W., Henderson, M. L., Riehl, W. J., Gunter, D., Murphy-Olson, D., Chan, S., Kamimura, R. T., Brettin, T. S., Meyer, F., Chivian, D., Weston, D. J., Glass, E. M., Davison, B. H., Kumari, S., Allen, B. H., Baumohl, J., Best, A. A., Bowen, B., Brenner, S. E., Bun, C. C., Chandonia, J.-M., Chia, J.-M., Colasanti, R., Conrad, N., Davis, J. J., DeJongh, M., Devoid, S., Dietrich, E., Drake, M. M., Dubchak, I., Edirisinghe, J. N., Fang, G., Faria, J. P., Frybarger, P. M., Gerlach, W., Gerstein, M., Gurtowski, J., Haun, H. L., He, F., Jain, R., Joachimiak, M. P., Keegan, K. P., Kondo, S., Kumar, V., Land, M. L., Mills, M., Novichkov, P., Oh, T., Olsen, G. J., Olson, B., Parrello, B., Pasternak, S., Pearson, E., Poon, S. S., Price, G., Ramakrishnan, S., Ranjan, P., Ronald, P. C., Schatz, M. C., Seaver, S. M. D., Shukla, M., Sutormin, R. A., Syed, M. H., Thomason, J., Tintle, N. L., Wang, D., Xia, F., Yoo, H., and Yoo, S. (2016). The DOE Systems Biology Knowledgebase (KBase). *bioRxiv*.

[Arumugam et al., 2011] Arumugam, M., Raes, J., Pelletier, E., Le Paslier, D., Yamada, T., Mende, D. R., Fernandes, G. R., Tap, J., Bruls, T., Batto, J. M., Bertalan, M., Borruel, N., Casellas, F., Fernandez, L., Gautier, L., Hansen, T., Hattori, M., Hayashi, T., Kleerebezem, M., Kurokawa, K., Leclerc, M., Levenez, F., Manichanh, C., Nielsen, H. B., Nielsen, T., Pons, N., Poulain, J., Qin, J., Sicheritz-Ponten, T., Tims, S., Torrents, D., Ugarte, E., Zoetendal, E. G., Wang, J., Guarner, F., Pedersen, O., De Vos, W. M., Brunak, S., Doré, J., Weissenbach, J., Ehrlich, S. D., and Bork, P. (2011). Enterotypes of the human gut microbiome. *Nature*, 473(7346):174–180.

[Aziz et al., 2008] Aziz, R. K., Bartels, D., Best, A. A., DeJongh, M., Disz, T., Edwards, R. A., Formsma, K., Gerdes, S., Glass, E. M., Kubal, M., Meyer, F., Olsen, G. J., Olson, R., Osterman, A. L., Overbeek, R. A., McNeil, L. K., Paarmann, D., Paczian, T., Parrello, B., Pusch, G. D., Reich, C., Stevens, R., Vassieva, O., Vonstein, V., Wilke, A., Zagnitko, O., Meyer, F., Goesmann, A., McHardy, A., Bartels, D., Bekel, T., Clausen, J., Kalinowski, J., Linke, B., Rupp, O., Giegerich, R., Bryson, K., Loux, V., Bossy, R., Nicolas, P., Chaillou, S., van de Guchte, M., Penaud, S., Maguin, E., Hoebeke, M., Bessieres, P., Vallenet, D., Labarre, L., Rouy, Z., Barbe, V., Bocs, S., Cruveiller, S., Lajus, A., Pascal, G., Scarpelli, C., Medigue, C., Overbeek, R., Begley, T., Butler, R., Choudhuri, J., Chuang, H., Cohoon, M., de Crecy-Lagard, V., Diaz, N., Disz, T., Edwards, R., Tatusov, R., Natale, D., Garkavtsev, I., Tatusova, T., Shankavaram, U., Rao, B., Kiryutin, B., Galperin, M., Fedorova, N., Koonin, E., Schneider, M., Tognolli, M., Bairoch, A., Kanehisa, M., Goto, S., Haft, D., Loftus, B., Richardson, D., Yang, F., Eisen, J., Paulsen, I., White, O., Overbeek, R., Bartels, D., Vonstein, V., Meyer, F., Wu, C., Shivakumar, S., Lowe, T., Eddy, S., Delcher, A., Harmon, D., Kasif, S., White, O., Salzberg, S., DeJongh, M., Formsma, K., Boillot, P., Gould, J., Rycenga, M., Best, A., Becker, S., and Palsson, B. (2008). The RAST Server: Rapid Annotations using Subsystems Technology. *BMC Genomics*, 9(1):75.

[Bäckhed et al., 2005] Bäckhed, F., Ley, R. E., Sonnenburg, J. L., Peterson, D. A., and Gordon, J. I. (2005). Host-bacterial mutualism in the human intestine.

[Bais et al., 2006] Bais, H. P., Weir, T. L., Perry, L. G., Gilroy, S., and Vivanco, J. M. (2006). The role of root exudates in rhizosphere interactions with plants and other organisms. *Annual Review of Plant Biology*, 57(1):233–266.

[Becker et al., 2007] Becker, S. A., Feist, A. M., Mo, M. L., Hannum, G., Palsson, B. O., and Herrgard, M. J. (2007). Quantitative Prediction of Cellular Metabolism with Constraint-based Models: The COBRA Toolbox. *Nat Protoc*, 2(3):727–738.

[Benedict et al., 2014] Benedict, M. N., Mundy, M. B., Henry, C. S., Chia, N., and Price, N. D. (2014). Likelihood-Based Gene Annotations for Gap Filling and Quality Assessment in Genome-Scale Metabolic Models. *PLoS Computational Biology*, 10(10):e1003882.

[Berry and Widder, 2014] Berry, D. and Widder, S. (2014). Deciphering microbial interactions and detecting keystone species with co-occurrence networks. *Frontiers in Microbiology*, 5:219.

[Biedermann et al., 2014] Biedermann, A., Kriebel, K., Kreikemeyer, B., and Lang, H. (2014). Interactions of anaerobic bacteria with dental stem cells: An in vitro study. *PLoS ONE*, 9(11).

[Blackall et al., 2015] Blackall, L. L., Wilson, B., and van Oppen, M. J. H. (2015). Coral-the world’s most diverse symbiotic ecosystem. *Molecular Ecology*, 24(21):5330–5347.

[Blasche et al., 2017] Blasche, S., Kim, Y., Oliveira, A. P., and Patil, K. R. (2017). Model microbial communities for ecosystems biology. *Current Opinion in Systems Biology*, 6:51–57.

[Borenstein and Feldman, 2009] Borenstein, E. and Feldman, M. W. (2009). Topological signatures of species interactions in metabolic networks. *Journal of computational biology : a journal of computational molecular cell biology*, 16(2):191–200.

[Borenstein et al., 2008] Borenstein, E., Kupiec, M., Feldman, M. W., and Ruppin, E. (2008). Large-scale reconstruction and phylogenetic analysis of metabolic environments. *Proceedings of the National Academy of Sciences of the United States of America*, 105(38):14482–7.

[Bosch and McFall-Ngai, 2011] Bosch, T. C. and McFall-Ngai, M. J. (2011). Metaorganisms as the new frontier. *Zoology*, 114(4):185–190.

[Bourneuf and Nicolas, 2017] Bourneuf, L. and Nicolas, J. (2017). FCA in a Logical Programming Setting for Visualization-Oriented Graph Compression. In *ICFCA 2017: Formal Concept Analysis*, pages 89–105. Springer, Cham.

[Brewka et al., 2015] Brewka, G., Delgrande, J. P., Romero, J., and Schaub, T. (2015). asprin: Customizing Answer Set Preferences without a Headache. pages 1467–1474, Unknown, Unknown or Invalid Region. AAAI Press.

[Brinza et al., 2009] Brinza, L., Viñuelas, J., Cottret, L., Calevro, F., Rahbé, Y., Febvay, G., Duport, G., Colella, S., Rabatel, A., Gautier, C., Fayard, J.-M., Sagot, M.-F., and Charles, H. (2009). Systemic analysis of the symbiotic function of Buchnera aphidicola, the primary endosymbiont of the pea aphid Acyrthosiphon pisum. *Comptes Rendus Biologies*, 332(11):1034–1049.

[Budinich et al., 2017] Budinich, M., Bourdon, J., Larhlimi, A., and Eveillard, D. (2017). A multi-objective constraint-based approach for modeling genome-scale microbial ecosystems. *PLOS ONE*, 12(2):e0171744.

[Carr and Borenstein, 2012] Carr, R. and Borenstein, E. (2012). NetSeed: A network-based reverse-ecology tool for calculating the metabolic interface of an organism with its environment. *Bioinformatics*, 28(5):734–735.

[Caspi et al., 2014] Caspi, R., Altman, T., Billington, R., Dreher, K., Foerster, H., Fulcher, C. A., Holland, T. A., Keseler, I. M., Kothari, A., Kubo, A., Krummenacker, M., Latendresse, M., Mueller, L. A., Ong, Q., Paley, S., Subhraveti, P., Weaver, D. S., Weerasinghe, D., Zhang, P., and Karp, P. D. (2014). The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of Pathway/Genome Databases. *Nucleic acids research*, 42(Database issue):D459–71.

[Caspi et al., 2016] Caspi, R., Billington, R., Ferrer, L., Foerster, H., Fulcher, C. A., Keseler, I. M., Kothari, A., Krummenacker, M., Latendresse, M., Mueller, L. A., Ong, Q., Paley, S., Subhraveti, P., Weaver, D. S., and Karp, P. D. (2016). The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. *Nucleic acids research*, 44(D1):D471–80.

[Caspi et al., 2018] Caspi, R., Billington, R., Fulcher, C. A., Keseler, I. M., Kothari, A., Krummenacker, M., Latendresse, M., Midford, P. E., Ong, Q., Ong, W. K., Paley, S., Subhraveti, P., and Karp, P. D. (2018). The MetaCyc database of metabolic pathways and enzymes. *Nucleic Acids Research*, 46(D1):D633–D639.

[Cavaliere et al., 2017] Cavaliere, M., Feng, S., Soyer, O. S., and Jiménez, J. I. (2017). Cooperation in microbial communities and their biotechnological applications. *Environmental Microbiology*, 19(8):2949–2963.

[Chan et al., 2017] Chan, S. H., Cai, J., Wang, L., Simons-Senftle, M. N., and Maranas, C. D. (2017). Standardizing biomass reactions and ensuring complete mass balance in genome-scale metabolic models. *Bioinformatics*, 33(22):3603–3609.

[Chazalviel et al., 2017] Chazalviel, M., Frainay, C., Poupin, N., Vinson, F., Merlet, B.-J., Gloaguen, Y., Cottret, L., and Jourdan, F. (2017). MetExploreViz: web component for interactive metabolic network visualization. *Bioinformatics*, pages 0–0.

[Chitale et al., 2016] Chitale, M., Khan, I. K., and Kihara, D. (2016). Missing gene identification using functional coherence scores. *Scientific Reports*, 6(1):31725.

[Christian et al., 2009] Christian, N., May, P., Kempa, S., Handorf, T., and Ebenhöh, O. (2009). An integrative approach towards completing genome-scale metabolic networks. *Molecular BioSystems*, 5(12):1889–1903.

[Cock et al., 2010] Cock, J. M., Sterck, L., Rouzé, P., Scornet, D., Allen, A. E., Amoutzias, G., Anthouard, V., Artiguenave, F., Aury, J.-M., Badger, J. H., Beszteri, B., Billiau, K., Bonnet, E., Bothwell, J. H., Bowler, C., Boyen, C., Brownlee, C., Carrano, C. J., Charrier, B., Cho, G. Y., Coelho, S. M., Collén, J., Corre, E., Da Silva, C., Delage, L., Delaroque, N., Dittami, S. M., Doulbeau, S., Elias, M., Farnham, G., Gachon, C. M. M., Gschloessl, B., Heesch, S., Jabbari, K., Jubin, C., Kawai, H., Kimura, K., Kloareg, B., Küpper, F. C., Lang, D., Le Bail, A., Leblanc, C., Lerouge, P., Lohr, M., Lopez, P. J., Martens, C., Maumus, F., Michel, G., Miranda-Saavedra, D., Morales, J., Moreau, H., Motomura, T., Nagasato, C., Napoli, C. A., Nelson, D. R., Nyvall-Collén, P., Peters, A. F., Pommier, C., Potin, P., Poulain, J., Quesneville, H., Read, B., Rensing, S. A., Ritter, A., Rousvoal, S., Samanta, M., Samson, G., Schroeder, D. C., Ségurens, B., Strittmatter, M., Tonon, T., Tregear, J. W., Valentin, K., von Dassow, P., Yamagishi, T., Van de Peer, Y., and Wincker, P. (2010). The Ectocarpus genome and the independent evolution of multicellularity in brown algae. *Nature*, 465(7298):617–21.

[Collén et al., 2013] Collén, J., Porcel, B., Carré, W., Ball, S. G., Chaparro, C., Tonon, T., Barbeyron, T., Michel, G., Noel, B., Valentin, K., Elias, M., Artiguenave, F., Arun, A., Aury, J.-M., Barbosa-Neto, J. F., Bothwell, J. H., Bouget, F.-Y., Brillet, L., Cabello-Hurtado, F., Capella-Gutiérrez, S., Charrier, B., Cladière, L., Cock, J. M., Coelho, S. M., Colleoni, C., Czjzek, M., Da Silva, C., Delage, L., Denoeud, F., Deschamps, P., Dittami, S. M., Gabaldón, T., Gachon, C. M. M., Groisillier, A., Hervé, C., Jabbari, K., Katinka, M., Kloareg, B., Kowalczyk, N., Labadie, K., Leblanc, C., Lopez, P. J., McLachlan, D. H., Meslet-Cladiere, L., Moustafa, A., Nehr, Z., Nyvall Collén, P., Panaud, O., Partensky, F., Poulain, J., Rensing, S. A., Rousvoal, S., Samson, G., Symeonidi, A., Weissenbach, J., Zambounis, A., Wincker, P., and Boyen, C. (2013). Genome structure and metabolic features in the red seaweed Chondrus crispus shed light on evolution of the Archaeplastida. *Proceedings of the National Academy of Sciences of the United States of America*, 110(13):5247–52.

[Collet et al., 2013] Collet, G., Eveillard, D., Gebser, M., Prigent, S., Schaub, T., Siegel, A., and Thiele, S. (2013). Extending the Metabolic Network of Ectocarpus Siliculosus Using Answer Set Programming. In *LPNMR 2013: Logic Programming and Nonmonotonic Reasoning*, pages 245–256. Springer.

[Comte et al., 2013] Comte, J., Fauteux, L., and Del Giorgio, P. A. (2013). Links between metabolic plasticity and functional redundancy in freshwater bacterioplankton communities. *Frontiers in Microbiology*, 4(MAY):112.

[Cormier et al., 2017] Cormier, A., Avia, K., Sterck, L., Derrien, T., Wucher, V., Andres, G., Monsoor, M., Godfroy, O., Lipinska, A., Perrineau, M. M., Van De Peer, Y., Hitte, C., Corre, E., Coelho, S. M., and Cock, J. M. (2017). Re-annotation, improved large-scale assembly and establishment of a catalogue of noncoding loci for the genome of the model brown alga Ectocarpus. *New Phytologist*, 214(1):219–232.

[Cottret et al., 2018] Cottret, L., Clément, F., Maxime, C., Flo eal, C., Yoann, G., Etienne, C., Benjamin, M., Ephanie, H. S., Jean-Charles, P., Nathalie, P., Florence, V., and Fabien, J. (2018). MetExplore: collaborative edition and exploration of metabolic networks. *Nucleic Acids Research*.

[Cottret and Jourdan, 2010] Cottret, L. and Jourdan, F. (2010). Graph methods for the investigation of metabolic networks in parasitology.

[Cottret et al., 2010] Cottret, L., Milreu, P. V., Acuña, V., Marchetti-Spaccamela, A., Stougie, L., Charles, H., and Sagot, M.-F. (2010). Graph-Based Analysis of the Metabolic Exchanges between Two Co-Resident Intracellular Symbionts, Baumannia cicadellinicola and Sulcia muelleri, with Their Insect Host, Homalodisca coagulata. *PLoS Computational Biology*, 6(9):e1000904.

[Coyte et al., 2015] Coyte, K. Z., Schluter, J., and Foster, K. R. (2015). The ecology of the microbiome: Networks, competition, and stability. *Science (New York, N.Y.)*, 350(6261):663–6.

[Dantzig, 1963] Dantzig, G. (1963). *Linear Programming and Extensions*. Princeton University Press.

[Dantzig and Orden, 1955] Dantzig, G. and Orden, A. (1955). The generalized simplex method for minimizing a linear form under linear inequality restraints. *Pacific Journal of Mathematics*, 5(2):187–195.

[Desouki et al., 2015] Desouki, A. A., Jarre, F., Gelius-Dietrich, G., and Lercher, M. J. (2015). CycleFreeFlux: efficient removal of thermodynamically infeasible loops from flux distributions. *Bioinformatics*, 31(13):2159–2165.

[Devoid et al., 2013] Devoid, S., Overbeek, R., DeJongh, M., Vonstein, V., Best, A. A., and Henry, C. (2013). Automated genome annotation and metabolic model reconstruction in the SEED and Model SEED. *Methods in molecular biology (Clifton, N.J.)*, 985:17–45.

[Dias et al., 2017] Dias, O., Gomes, D., Vilaça, P., Cardoso, J., Rocha, M., Ferreira, E. C., and Rocha, I. (2017). Genome-Wide Semi-Automated Annotation of Transporter Systems. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 14(2):443–456.

[Dittami et al., 2014a] Dittami, S. M., Barbeyron, T., Boyen, C., Cambefort, J., Collet, G., Delage, L., Gobet, A., Groisillier, A., Leblanc, C., Michel, G., Scornet, D., Siegel, A., Tapia, J. E., and Tonon, T. (2014a). Genome and metabolic network of "Candidatus Phaeomarinobacter ectocarpi", a new candidate genus of Alphaproteobacteria frequently associated with brown algae. *Frontiers in Genetics*, 5:241.

[Dittami et al., 2016] Dittami, S. M., Duboscq-Bidot, L., Perennou, M., Gobet, A., Corre, E., Boyen, C., and Tonon, T. (2016). Host–microbe interactions as a driver of acclimation to salinity gradients in brown algal cultures. *The ISME Journal*, 10(1):51–63.

[Dittami et al., 2014b] Dittami, S. M., Eveillard, D., and Tonon, T. (2014b). A metabolic approach to study algal-bacterial interactions in changing environments. *Molecular Ecology*, 23(7):1656–1660.

[Douglas, 1992] Douglas, A. E. (1992). Requirement of pea aphids (Acyrthosiphon pisum) for their symbiotic bacteria. *Entomologia Experimentalis et Applicata*, 65(2):195–198.

[Duarte et al., 2007] Duarte, N. C., Becker, S. A., Jamshidi, N., Thiele, I., Mo, M. L., Vo, T. D., Srivas, R., and Palsson, B. Ø. (2007). Global reconstruction of the human metabolic network based on genomic and bibliomic data. *Proceedings of the National Academy of Sciences of the United States of America*, 104(6):1777–82.

[Ebenhöh et al., 2004] Ebenhöh, O., Handorf, T., and Heinrich, R. (2004). Structural analysis of expanding metabolic networks. *Genome informatics. International Conference on Genome Informatics*, 15(1):35–45.

[Ebenhöh et al., 2006] Ebenhöh, O., Handorf, T., and Kahn, D. (2006). Evolutionary changes of metabolic networks and their biosynthetic capacities. *Systems biology*, 153(5):354–8.

[Ebrahim et al., 2013] Ebrahim, A., Lerman, J. A., Palsson, B. O., and Hyduke, D. R. (2013). COBRApy: COnstraints-Based Reconstruction and Analysis for Python. *BMC systems biology*, 7:74.

[Edwards and Palsson, 1999] Edwards, J. S. and Palsson, B. O. (1999). Systems properties of the Haemophilus influenzae Rd metabolic genotype. *Journal of Biological Chemistry*, 274(25):17410–17416.

[Egan et al., 2013] Egan, S., Harder, T., Burke, C., Steinberg, P., Kjelleberg, S., and Thomas, T. (2013). The seaweed holobiont: Understanding seaweed-bacteria interactions. *FEMS Microbiology Reviews*, 37(3):462–476.

[Ehrgott, 2005] Ehrgott, M. (2005). *Multicriteria optimization*. Lecture notes in economics and mathematical systems. Springer.

[Elbourne et al., 2017] Elbourne, L. D. H., Tetu, S. G., Hassan, K. A., and Paulsen, I. T. (2017). TransportDB 2.0: a database for exploring membrane transporters in sequenced genomes from all domains of life. *Nucleic Acids Research*, 45(D1):D320–D324.

[Ellegren, 2014] Ellegren, H. (2014). Genome sequencing and population genomics in non-model organisms. *Trends in Ecology and Evolution*, 29(1):51–63.

[Emms and Kelly, 2015] Emms, D. M. and Kelly, S. (2015). OrthoFinder: solving fundamental biases in whole genome comparisons dramatically improves orthogroup inference accuracy. *Genome Biology*, 16(1):157.

[Eng and Borenstein, 2016] Eng, A. and Borenstein, E. (2016). An algorithm for designing minimal microbial communities with desired metabolic capacities. *Bioinformatics*, 32(13):2008–2016.

[Engel et al., 2016] Engel, P., Kwong, W. K., McFrederick, Q., Anderson, K. E., Barribeau, S. M., Chandler, J. A., Cornman, R. S., Dainat, J., de Miranda, J. R., Doublet, V., Emery, O., Evans, J. D., Farinelli, L., Flenniken, M. L., Granberg, F., Grasis, J. A., Gauthier, L., Hayer, J., Koch, H., Kocher, S., Martinson, V. G., Moran, N., Munoz-Torres, M., Newton, I., Paxton, R. J., Powell, E., Sadd, B. M., Schmid-Hempel, P., Schmid-Hempel, R., Song, S. J., Schwarz, R. S., VanEngelsdorp, D., and Dainat, B. (2016). The Bee Microbiome: Impact on Bee Health and Model for Evolution and Ecology of Host-Microbe Interactions. *mBio*, 7(2):e02164–15.

[Faria et al., 2016] Faria, J. P., Khazaei, T., Edirisinghe, J. N., Weisenhorn, P., Seaver, S. M. D., Conrad, N., Harris, N., DeJongh, M., and Henry, C. S. (2016). Constructing and Analyzing Metabolic Flux Models of Microbial Communities. In *Hydrocarbon and Lipid Microbiology Protocols*, pages 247–273. Springer.

[Faria et al., 2018] Faria, J. P., Rocha, M., Rocha, I., and Henry, C. S. (2018). Methods for automated genome-scale metabolic model reconstruction. *Biochemical Society transactions*, 46(4):931–936.

[Faust and Raes, 2012] Faust, K. and Raes, J. (2012). Microbial interactions: from networks to models. *Nature Reviews Microbiology*, 10(8):538–550.

[Feist et al., 2007] Feist, A. M., Henry, C. S., Reed, J. L., Krummenacker, M., Joyce, A. R., Karp, P. D., Broadbelt, L. J., Hatzimanikatis, V., and Palsson, B. (2007). A genome-scale metabolic reconstruction for Escherichia coli K-12 MG1655 that accounts for 1260 ORFs and thermodynamic information. *Molecular Systems Biology*, 3:121.

[Feist and Palsson, 2010] Feist, A. M. and Palsson, B. O. (2010). The biomass objective function. *Current Opinion in Microbiology*, 13(3):344–349.

[Fiehn, 2001] Fiehn, O. (2001). Combining Genomics, Metabolome Analysis, and Biochemical Modelling to Understand Metabolic Networks. *Comparative and Functional Genomics*, 2(3):155–168.

[Freilich et al., 2011] Freilich, S., Zarecki, R., Eilam, O., Shtifman Segal, E., Henry, C. S., Kupiec, M., Gophna, U., Sharan, R., and Ruppin, E. (2011). Competitive and cooperative metabolic interactions in bacterial communities. *Nature Communications*, 2.

[Friedman and Alm, 2012] Friedman, J. and Alm, E. J. (2012). Inferring Correlation Networks from Genomic Survey Data. *PLoS Computational Biology*, 8(9):e1002687.

[Frioux et al., 2017] Frioux, C., Schaub, T., Schellhorn, S., Siegel, A., and Wanko, P. (2017). *Hybrid Metabolic Network Completion*, pages 308–321. Springer International Publishing, Cham.

[Gauthier et al., 2015] Gauthier, J.-P., Outreman, Y., Mieuzet, L., and Simon, J.-C. (2015). Bacterial communities associated with host-adapted populations of pea aphids revealed by deep sequencing of 16S ribosomal DNA. *PloS one*, 10(3):e0120664.

[Gebser et al., 2016a] Gebser, M., Kaminski, R., Kaufmann, B., Ostrowski, M., Schaub, T., and Wanko, P. (2016a). Theory solving made easy with clingo 5. In Carro, M. and King, A., editors, *Technical Communications of the Thirty-second International Conference on Logic Programming (ICLP'16)*, volume 52, pages 2:1–2:15. Open Access Series in Informatics (OASIcs).

[Gebser et al., 2016b] Gebser, M., Kaminski, R., Kaufmann, B., Ostrowski, M., Schaub, T., and Wanko, P. (2016b). Theory Solving made easy with Clingo5. In *http://software.imdea.org/Conferences/ICLP2016/proceedings.html*, New-York City.

[Gebser et al., 2012] Gebser, M., Kaminski, R., Kaufmann, B., and Schaub, T. (2012). Answer Set Solving in Practice. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 6(3):1–238.

[Gelfond and Lifschitz, 1991] Gelfond, M. and Lifschitz, V. (1991). Classical negation in logic programs and disjunctive databases. *New Generation Computing*, 9:365–385.

[Gibson et al., 2004] Gibson, G. R., Probert, H. M., Loo, J. V., Rastall, R. A., and Roberfroid, M. B. (2004). Dietary modulation of the human colonic microbiota: updating the concept of prebiotics. *Nutrition Research Reviews*, 17(02):259.

[Glick, 1995] Glick, B. R. (1995). The enhancement of plant growth by free-living bacteria. *Canadian Journal of Microbiology*, 41(2):109–117.

[Goecke et al., 2010] Goecke, F., Labes, A., Wiese, J., and Imhoff, J. F. (2010). Review chemical interactions between Marine macroalgae and bacteria. *Marine Ecology Progress Series*, 409:267–300.

[Goldford and Segrè, 2018] Goldford, J. E. and Segrè, D. (2018). Modern views of ancient metabolic networks. *Current Opinion in Systems Biology*, 8:117–124.

[Gottstein et al., 2016] Gottstein, W., Olivier, B. G., Bruggeman, F. J., and Teusink, B. (2016). Constraint-based stoichiometric modelling from single organisms to microbial communities.

[Granger et al., 2016] Granger, B. R., Chang, Y. C., Wang, Y., DeLisi, C., Segrè, D., and Hu, Z. (2016). Visualization of Metabolic Interaction Networks in Microbial Communities Using VisANT 5.0. *PLoS Computational Biology*, 12(4):e1004875.

[Greenblum et al., 2012] Greenblum, S., Turnbaugh, P. J., and Borenstein, E. (2012). Metagenomic systems biology of the human gut microbiome reveals topological shifts associated with obesity and inflammatory bowel disease. *Proceedings of the National Academy of Sciences of the United States of America*, 109(2):594–9.

[Griesemer et al., 2018] Griesemer, M., Kimbrel, J. A., Zhou, C. E., Navid, A., and D'haeseleer, P. (2018). Combining multiple functional annotation tools increases coverage of metabolic annotation. *bioRxiv*, page 160887.

[Ha et al., 2014] Ha, C. W. Y., Lam, Y. Y., and Holmes, A. J. (2014). Mechanistic links between gut microbial community dynamics, microbial functions and metabolic health. *World journal of gastroenterology*, 20(44):16498–517.

[Handorf et al., 2005] Handorf, T., Ebenhöh, O., and Heinrich, R. (2005). Expanding metabolic networks: Scopes of compounds, robustness, and evolution. *Journal of Molecular Evolution*, 61(4):498–512.

[Hanemaaijer et al., 2017] Hanemaaijer, M., Olivier, B. G., Röling, W. F. M., Bruggeman, F. J., Teusink, B., and Hettich, R. (2017). Model-based quantification of metabolic interactions from dynamic microbial-community data. *PLOS ONE*, 12(3):e0173183.

[Heavner and Price, 2015] Heavner, B. D. and Price, N. D. (2015). Transparency in metabolic network reconstruction enables scalable biological discovery. *Current opinion in biotechnology*, 34.

[Heinken et al., 2016] Heinken, A., Ravcheev, D. A., and Thiele, I. (2016). Systems biology of bacteria-host interactions. In *The Human Microbiota and Chronic Disease: Dysbiosis as a Cause of Human Pathology*, pages 113–137. John Wiley & Sons, Inc., Hoboken, NJ, USA.

[Heinken et al., 2013] Heinken, A., Sahoo, S., Fleming, R. M. T., and Thiele, I. (2013). Systems-level characterization of a host-microbe metabolic symbiosis in the mammalian gut. *Gut microbes*, 4(1):28–40.

[Heinken and Thiele, 2015] Heinken, A. and Thiele, I. (2015). Systems biology of host-microbe metabolomics. *Wiley interdisciplinary reviews. Systems biology and medicine*, 7(4):195–219.

[Henry et al., 2016] Henry, C. S., Bernstein, H. C., Weisenhorn, P., Taylor, R. C., Lee, J. Y., Zucker, J., and Song, H. S. (2016). Microbial Community Metabolic Modeling: A Community Data-Driven Network Reconstruction. *Journal of Cellular Physiology*, 231(11):2339–2345.

[Henry et al., 2010] Henry, C. S., DeJongh, M., Best, A. A., Frybarger, P. M., Linsay, B., and Stevens, R. L. (2010). High-throughput generation, optimization and analysis of genome-scale metabolic models. *Nature biotechnology*, 28(9):977–982.

[Hood et al., 2008] Hood, L., Rowen, L., Galas, D. J., and Aitchison, J. D. (2008). Systems biology at the Institute for Systems Biology. *Briefings in Functional Genomics and Proteomics*, 7(4):239–248.

[Hornung et al., 2018] Hornung, B., Martins dos Santos, V. A., Smidt, H., and Schaap, P. J. (2018). Studying microbial functionality within the gut ecosystem by systems biology.

[Hucka et al., 2003] Hucka, M., Finney, A., Sauro, H. M., Bolouri, H., Doyle, J. C., Kitano, H., Arkin, A. P., Bornstein, B. J., Bray, D., Cornish-Bowden, A., Cuellar, A. A., Dronov, S., Gilles, E. D., Ginkel, M., Gor, V., Goryanin, I. I., Hedley, W. J., Hodgman, T. C., Hofmeyr, J. H., Hunter, P. J., Juty, N. S., Kasberger, J. L., Kremling, A., Kummer, U., Le Nov??re, N., Loew, L. M., Lucio, D., Mendes, P., Minch, E., Mjolsness, E. D., Nakayama, Y., Nelson, M. R., Nielsen, P. F., Sakurada, T., Schaff, J. C., Shapiro, B. E., Shimizu, T. S., Spence, H. D., Stelling, J., Takahashi, K., Tomita, M., Wagner, J., and Wang, J. (2003). The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models. *Bioinformatics*, 19(4):524–531.

[Human Microbiome Project, 2012] Human Microbiome Project, C. (2012). A framework for human microbiome research. *Nature*, 486(7402):215–221.

[Ideker et al., 2001] Ideker, T., Galitski, T., and Hood, L. (2001). A new approach to decoding life : Systems Biology. *Annual Review of Genomics and Human Genetics*, 2(1):343–372.

[Janhunen et al., 2017] Janhunen, T., Kaminski, R., Ostrowski, M., Schellhorn, S., Wanko, P., and Schaub, T. (2017). Clingo goes linear constraints over reals and integers. In *Theory and Practice of Logic Programming*, pages 872–888.

[Johns et al., 2016] Johns, N. I., Blazejewski, T., Gomes, A. L., and Wang, H. H. (2016). Principles for designing synthetic microbial communities. *Current Opinion in Microbiology*, 31:146–153.

[Joyce and Palsson, 2006] Joyce, A. R. and Palsson, B. Ø. (2006). The model organism as a system: integrating 'omics' data sets. *Nature Reviews Molecular Cell Biology*, 7(3):198–210.

[Julien-Laferrière et al., 2016] Julien-Laferrière, A., Bulteau, L., Parrot, D., Marchetti-Spaccamela, A., Stougie, L., Vinga, S., Mary, A., and Sagot, M.-F. (2016). A Combinatorial Algorithm for Microbial Consortia Synthetic Design. *Scientific Reports*, 6:29182.

[Kaminski et al., 2017] Kaminski, R., Schaub, T., and Wanko, P. (2017). A tutorial on hybrid answer set solving with clingo. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10370 LNCS, pages 167–203.

[Kanehisa et al., 2016] Kanehisa, M., Sato, Y., Kawashima, M., Furumichi, M., and Tanabe, M. (2016). KEGG as a reference resource for gene and protein annotation. *Nucleic acids research*, 44(D1):D457–62.

[Karp et al., 2017] Karp, P. D., Billington, R., Caspi, R., Fulcher, C. A., Latendresse, M., Kothari, A., Keseler, I. M., Krummenacker, M., Midford, P. E., Ong, Q., Ong, W. K., Paley, S. M., and Subhraveti, P. (2017). The BioCyc collection of microbial genomes and metabolic pathways. *Briefings in Bioinformatics*, 28(12):1–6.

[Karp et al., 2016] Karp, P. D., Latendresse, M., Paley, S. M., Krummenacker, M., Ong, Q. D., Billington, R., Kothari, A., Weaver, D., Lee, T. T., Subhraveti, P., Spaulding, A., Fulcher, C., Keseler, I. M., and Caspi, R. (2016). Pathway tools version 19.0 update: Software for pathway/genome informatics and systems biology. *Briefings in Bioinformatics*, 17(5):877–890.

[Karp et al., 2002] Karp, P. D., Paley, S., and Romero, P. (2002). The Pathway Tools software. *Bioinformatics*, 18(Suppl 1):S225–S232.

[Khandelwal et al., 2013] Khandelwal, R. A., Olivier, B. G., Röling, W. F. M., Teusink, B., and Bruggeman, F. J. (2013). Community flux balance analysis for microbial consortia at balanced growth. *PloS one*, 8(5):e64567.

[Kim and Lun, 2014] Kim, M. K. and Lun, D. S. (2014). Methods for integration of transcriptomic data in genome-scale metabolic models. *Computational and Structural Biotechnology Journal*, 11(18):59–65.

[Kim et al., 2012] Kim, T. Y., Sohn, S. B., Kim, Y. B., Kim, W. J., and Lee, S. Y. (2012). Recent advances in reconstruction and applications of genome-scale metabolic models. *Current opinion in biotechnology*, 23(4):617–23.

[King et al., 2016] King, Z. A., Lu, J., Dräger, A., Miller, P., Federowicz, S., Lerman, J. A., Ebrahim, A., Palsson, B. O., and Lewis, N. E. (2016). BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. *Nucleic Acids Research*, 44(D1):D515–22.

[Kitano, 2002a] Kitano, H. (2002a). Computational systems biology. *Nature*, 420(6912):206–210.

[Kitano, 2002b] Kitano, H. (2002b). Systems biology: A brief overview. *Science*, 295(5560):1662–1664.

[KleinJan et al., 2017] KleinJan, H., Jeanthon, C., Boyen, C., and Dittami, S. M. (2017). Exploring the Cultivable Ectocarpus Microbiome. *Frontiers in Microbiology*, 8:2456.

[Klitgord and Segrè, 2010] Klitgord, N. and Segrè, D. (2010). Environments that induce synthetic microbial ecosystems. *PLoS Computational Biology*, 6(11):e1001002.

[Koch et al., 2016] Koch, S., Benndorf, D., Fronk, K., Reichl, U., and Klamt, S. (2016). Predicting compositions of microbial communities from stoichiometric models with applications for the biogas process. *Biotechnology for Biofuels*, 9(1):17.

[Kreimer et al., 2012] Kreimer, A., Doron-Faigenboim, A., Borenstein, E., and Freilich, S. (2012). NetCmpt: A network-based tool for calculating the metabolic competition between bacterial species. *Bioinformatics*, 28(16):2195–2197.

[Kruse and Ebenhöh, 2008] Kruse, K. and Ebenhöh, O. (2008). Comparing flux balance analysis to network expansion: producibility, sustainability and the scope of compounds. *Genome Informatics*, 20:91–101.

[Laniau et al., 2017] Laniau, J., Frioux, C., Nicolas, J., Baroukh, C., Cortes, M.-P. M.-P., Got, J., Trottier, C., Eveillard, D., and Siegel, A. (2017). Combining graph and flux-based structures to decipher phenotypic essential metabolites within metabolic networks. *PeerJ*, 5(10):e3860.

[Latendresse, 2014] Latendresse, M. (2014). Efficiently gap-filling reaction networks. *BMC bioinformatics*, 15:225.

[Leonelli and Ankeny, 2013] Leonelli, S. and Ankeny, R. A. (2013). What makes a model organism? *Endeavour*, 37(4):209–212.

[Letunic and Bork, 2016] Letunic, I. and Bork, P. (2016). Interactive tree of life (iTOL) v3: an online tool for the display and annotation of phylogenetic and other trees. *Nucleic acids research*, 44(W1):W242–5.

[Levy and Borenstein, 2013] Levy, R. and Borenstein, E. (2013). Metabolic modeling of species interaction in the human microbiome elucidates community-level assembly rules. *Proceedings of the National Academy of Sciences of the United States of America*, 110(31):12804–9.

[Levy et al., 2015] Levy, R., Carr, R., Kreimer, A., Freilich, S., and Borenstein, E. (2015). NetCooperate: a network-based tool for inferring host-microbe and microbe-microbe cooperation. *BMC bioinformatics*, 16(1):164.

[Li et al., 2016] Li, C., Lim, K. M. K., Chng, K. R., and Nagarajan, N. (2016). Predicting microbial interactions through computational approaches. *Methods*, 102:12–19.

[Loira et al., 2015] Loira, N., Zhukova, A., and Sherman, D. J. (2015). Pantograph: A template-based method for genome-scale metabolic model reconstruction. *Journal of Bioinformatics and Computational Biology*, 13(02):1550006.

[Lozupone et al., 2012] Lozupone, C. A., Stombaugh, J. I., Gordon, J. I., Jansson, J. K., and Knight, R. (2012). Diversity, stability and resilience of the human gut microbiota. *Nature*, 489(7415):220–230.

[Magnúsdóttir et al., 2016] Magnúsdóttir, S., Heinken, A., Kutt, L., Ravcheev, D. A., Bauer, E., Noronha, A., Greenhalgh, K., Jäger, C., Baginska, J., Wilmes, P., Fleming, R. M. T., and Thiele, I. (2016). Generation of genome-scale metabolic reconstructions for 773 members of the human gut microbiota. *Nature Biotechnology*, 35(1):81–89.

[Magnúsdóttir and Thiele, 2018] Magnúsdóttir, S. and Thiele, I. (2018). Modeling metabolism of the human gut microbiome. *Current Opinion in Biotechnology*, 51:90–96.

[Mahadevan and Schilling, 2003] Mahadevan, R. and Schilling, C. (2003). The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic Engineering*, 5(4):264–276.

[Manor et al., 2014] Manor, O., Levy, R., and Borenstein, E. (2014). Mapping the inner workings of the microbiome: genomic- and metagenomic-based study of metabolism and metabolic interactions in the human microbiome. *Cell metabolism*, 20(5):742–52.

[Maranas and Zomorrodi, 2016] Maranas, C. D. and Zomorrodi, A. R. (2016). *Optimization methods in metabolic networks*. Wiley.

[Marchesi and Ravel, 2015] Marchesi, J. R. and Ravel, J. (2015). The vocabulary of microbiome research: a proposal. *Microbiome*, 3(1):31.

[Mendes-Soares and Chia, 2017] Mendes-Soares, H. and Chia, N. (2017). Community metabolic modeling approaches to understanding the gut microbiome: Bridging biochemistry and ecology.

[Mendes-Soares et al., 2016] Mendes-Soares, H., Mundy, M., Soares, L. M., and Chia, N. (2016). MMinte: an application for predicting metabolic interactions among the microbial species in a community. *BMC bioinformatics*, 17(1):343.

[Monk et al., 2017] Monk, J. M., Lloyd, C. J., Brunk, E., Mih, N., Sastry, A., King, Z., Takeuchi, R., Nomura, W., Zhang, Z., Mori, H., Feist, A. M., and Palsson, B. O. (2017). iML1515, a knowledgebase that computes Escherichia coli traits.

[Moran, 2006] Moran, N. A. (2006). Symbiosis. *Current Biology*, 16(20):866–871.

[Moretti et al., 2016] Moretti, S., Martin, O., Van Du Tran, T., Bridge, A., Morgat, A., and Pagni, M. (2016). MetaNetX/MNXref – reconciliation of metabolites and biochemical reactions to bring together genome-scale metabolic networks. *Nucleic Acids Research*, 44(D1):D523–D526.

[Morterol, 2016] Morterol, M. (2016). *Méthodes avancées de raisonnement en logique propositionnelle : application aux réseaux métaboliques*. PhD thesis.

[Moya and Ferrer, 2016] Moya, A. and Ferrer, M. (2016). Functional Redundancy-Induced Stability of Gut Microbiota Subjected to Disturbance. *Trends in Microbiology*, 24(5):402–413.

[Oberhardt et al., 2009] Oberhardt, M. A., Palsson, B. Ø., and Papin, J. A. (2009). Applications of genome-scale metabolic reconstructions. *Molecular systems biology*, 5:320.

[O'Brien et al., 2015] O'Brien, E. J., Monk, J. M., and Palsson, B. O. (2015). Using Genome-scale Models to Predict Biological Capabilities. *Cell*, 161(5):971–87.

[Ofaim et al., 2017] Ofaim, S., Ofek-Lalzar, M., Sela, N., Jinag, J., Kashi, Y., Minz, D., and Freilich, S. (2017). Analysis of Microbial Functions in the Rhizosphere Using a Metabolic-Network Based Framework for Metagenomics Interpretation. *Frontiers in Microbiology*, 8:1606.

[Oksman-Caldentey and Saito, 2005] Oksman-Caldentey, K.-M. and Saito, K. (2005). Integrating genomics and metabolomics for engineering plant metabolic pathways. *Current Opinion in Biotechnology*, 16(2):174–179.

[Opatovsky et al., 2018] Opatovsky, I., Santos-Garcia, D., Ruan, Z., Lahav, T., Ofaim, S., Mouton, L., Barbe, V., Jiang, J., Zchori-Fein, E., and Freilich, S. (2018). Modeling trophic dependencies and exchanges among insects' bacterial symbionts in a host-simulated environment. *BMC Genomics*, 19(1):402.

[Orth et al., 2011] Orth, J. D., Conrad, T. M., Na, J., Lerman, J. a., Nam, H., Feist, A. M., and Palsson, B. Ø. (2011). A comprehensive genome-scale reconstruction of Escherichia coli metabolism–2011. *Molecular systems biology*, 7(535):535.

[Orth and Palsson, 2010] Orth, J. D. and Palsson, B. (2010). Systematizing the generation of missing metabolic knowledge. *Biotechnology and Bioengineering*, 107(3):403–412.

[Orth et al., 2010] Orth, J. D., Thiele, I., and Palsson, B. Ø. (2010). What is Flux Balance Analysis ? *Nature biotechnology*, 28(3):245–248.

[Pan and Reed, 2018] Pan, S. and Reed, J. L. (2018). Advances in gap-filling genome-scale metabolic models and model-driven experiments lead to novel metabolic discoveries. *Current Opinion in Biotechnology*, 51:103–108.

[Peres et al., 2014] Peres, S., Morterol, M., and Simon, L. (2014). SAT-Based Metabolics Pathways Analysis without Compilation. In Mendes, P., Dada, J. O., and Smallbone, K., editors, *Computational Methods in Systems Biology*, pages 20–31, Cham. Springer International Publishing.

[Pharkya et al., 2004] Pharkya, P., Burgard, A. P., and Maranas, C. D. (2004). OptStrain: A computational framework for redesign of microbial production systems. *Genome Research*, 14(11):2367–2376.

[Plata et al., 2012] Plata, G., Fuhrer, T., Hsiao, T.-L., Sauer, U., and Vitkup, D. (2012). Global probabilistic annotation of metabolic networks enables enzyme discovery. *Nature Chemical Biology*, 8(10):848–854.

[Prigent et al., 2014] Prigent, S., Collet, G., Dittami, S. M., Delage, L., De Corny, F. E., Dameron, O., Eveillard, D., Thiele, S., Cambefort, J., Boyen, C., Siegel, A., and Tonon, T. (2014). The genome-scale metabolic network of Ectocarpus siliculosus (EctoGEM): A resource to study brown algal physiology and beyond. *Plant Journal*, 80(2):367–381.

[Prigent et al., 2017] Prigent, S., Frioux, C., Dittami, S. M., Thiele, S., Larhlimi, A., Collet, G., Gutknecht, F., Got, J., Eveillard, D., Bourdon, J., Plewniak, F., Tonon, T., and Siegel, A. (2017). Meneco, a Topology-Based Gap-Filling Tool Applicable to Degraded Genome-Wide Metabolic Networks. *PLOS Computational Biology*, 13(1):e1005276.

[Provasoli and Pintner, 1980] Provasoli, L. and Pintner, I. J. (1980). Bacteria induced polymorphism in an axenic laboratory strain of Ulva lactuca (Chlorophyceae). *Journal of Phycology*, 16(2):196–201.

[Raymond and Segrè, 2006] Raymond, J. and Segrè, D. (2006). The effect of oxygen on biochemical networks and the evolution of complex life. *Science (New York, N.Y.)*, 311(5768):1764–7.

[Reed et al., 2006] Reed, J. L., Patel, T. R., Chen, K. H., Joyce, A. R., Applebee, M. K., Herring, C. D., Bui, O. T., Knight, E. M., Fong, S. S., and Palsson, B. O. (2006). Systems approach to refining genome annotation. *Proc Natl Acad Sci U S A*, 103(46):17480–17484.

[Reed et al., 2003] Reed, J. L., Vo, T. D., Schilling, C. H., and Palsson, B. O. (2003). An expanded genome-scale model of Escherichia coli K-12 (iJR904 GSM/GPR). *Genome biology*, 4(9):R54.

[Rose and Mazat, 2018] Rose, T. and Mazat, J.-P. (2018). FluxVisualizer, a Software to Visualize Fluxes through Metabolic Networks. *Processes*, 6(5):39.

[Royer et al., 2008] Royer, L., Reimann, M., Andreopoulos, B., and Schroeder, M. (2008). Unraveling Protein Networks with Power Graph Analysis. *PLoS Comput Biol*, 4(7):e1000108.

[Russell et al., 2017] Russell, J. J., Theriot, J. A., Sood, P., Marshall, W. F., Landweber, L. F., Fritz-Laylin, L., Polka, J. K., Oliferenko, S., Gerbich, T., Gladfelter, A., Umen, J., Bezanilla, M., Lancaster, M. A., He, S., Gibson, M. C., Goldstein, B., Tanaka, E. M., Hu, C. K., and Brunet, A. (2017). Non-model model organisms. *BMC Biology*, 15(1):55.

[Satish Kumar et al., 2007] Satish Kumar, V., Dasika, M. S., and Maranas, C. D. (2007). Optimization based automated curation of metabolic reconstructions. *BMC bioinformatics*, 8:212.

[Schaub and Thiele, 2009a] Schaub, T. and Thiele, S. (2009a). Metabolic network expansion with answer set programming. In *International Conference on Logic Programming (ICLP)*, pages 312–326. Springer Berlin Heidelberg.

[Schaub and Thiele, 2009b] Schaub, T. and Thiele, S. (2009b). Metabolic network expansion with ASP. In Hill, P. and Warren, D., editors, *Proceedings of the Twenty-fifth International Conference on Logic Programming (ICLP'09)*, volume 5649 of *Lecture Notes in Computer Science*, pages 312–326. Springer-Verlag.

[Schellenberger et al., 2011a] Schellenberger, J., Lewis, N. E., and Palsson, B. Ø. (2011a). Elimination of thermodynamically infeasible loops in steady-state metabolic models. *Biophysical journal*, 100(3):544–53.

[Schellenberger et al., 2011b] Schellenberger, J., Que, R., Fleming, R. M. T., Thiele, I., Orth, J. D., Feist, A. M., Zielinski, D. C., Bordbar, A., Lewis, N. E., Rahmanian, S., Kang, J., Hyduke, D. R., Palsson, B. Ø. O., Holder, A., Hucka, M., Hyduke, D. R., Jamshidi, N., Lee, S. Y., Novère, N. L., Lerman, J. A., Lewis, N. E., Ma, D., Mahadevan, R., Maranas, C., Nagarajan, H., Navid, A., Nielsen, J., Nielsen, L. K., Nogales, J., Noronha, A., Pal, C., Palsson, B. Ø. O., Papin, J. A., Patil, K. R., Price, N. D., Reed, J. L., Saunders, M., Senger, R. S., Sonnenschein, N., Sun, Y., and Thiele, I. (2011b). Quantitative prediction of cellular metabolism with constraint-based models: the COBRA Toolbox v2.0. *Nature protocols*, 6(9):1290–307.

[Schuetz et al., 2007] Schuetz, R., Kuepfer, L., and Sauer, U. (2007). Systematic evaluation of objective functions for predicting intracellular fluxes in Escherichia coli. *Molecular systems biology*, 3:119.

[Schuetz et al., 2012] Schuetz, R., Zamboni, N., Zampieri, M., Heinemann, M., and Sauer, U. (2012). Multidimensional optimality of microbial metabolism. *Science*, 336(6081):601–4.

[Sekirov et al., 2010] Sekirov, I., Russell, S. L., Antunes, L. C. M., and Finlay, B. B. (2010). Gut Microbiota in Health and Disease. *Physiological Reviews*, 90(3):859–904.

[Shannon et al., 2003] Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., Amin, N., Schwikowski, B. B., and Ideker, T. (2003). Cytoscape: A software Environment for integrated models of biomolecular interaction networks. *Genome Research*, 13(11):2498–2504.

[Simons et al., 2002] Simons, P., Niemelä, I., and Soininen, T. (2002). Extending and implementing the stable model semantics. *Artificial Intelligence*, 138(1-2):181–234.

[Steffensen et al., 2016] Steffensen, J. L., Dufault-Thompson, K., and Zhang, Y. (2016). PSAMM: A Portable System for the Analysis of Metabolic Models. *PLoS Computational Biology*, 12(2):e1004732.

[Steinway et al., 2015] Steinway, S. N., Biggs, M. B., Loughran, T. P., Papin, J. A., and Albert, R. (2015). Inference of Network Dynamics and Metabolic Interactions in the Gut Microbiome. *PLOS Computational Biology*, 11(6):e1004338.

[Stolyar et al., 2007] Stolyar, S., Van Dien, S., Hillesland, K. L., Pinel, N., Lie, T. J., Leigh, J. A., and Stahl, D. A. (2007). Metabolic modeling of a mutualistic microbial community. *Molecular systems biology*, 3(1):92.

[Succurro and Ebenhöh, 2018] Succurro, A. and Ebenhöh, O. (2018). Review and perspective on mathematical modeling of microbial ecosystems. *Biochemical Society Transactions*, 46(2):403–412.

[Sung et al., 2017] Sung, J., Kim, S., Cabatbat, J. J. T., Jang, S., Jin, Y. S., Jung, G. Y., Chia, N., and Kim, P. J. (2017). Global metabolic interaction network of the human gut microbiota for context-specific community-scale analysis. *Nature Communications*, 8(1).

[Swainston et al., 2016] Swainston, N., Smallbone, K., Hefzi, H., Dobson, P. D., Brewer, J., Hanscho, M., Zielinski, D. C., Ang, K. S., Gardiner, N. J., Gutierrez, J. M., Kyriakopoulos, S., Lakshmanan, M., Li, S., Liu, J. K., Martínez, V. S., Orellana, C. A., Quek, L. E., Thomas, A., Zanghellini, J., Borth, N., Lee, D. Y., Nielsen, L. K., Kell, D. B., Lewis, N. E., and Mendes, P. (2016). Recon 2.2: from reconstruction to model of human metabolism. *Metabolomics*, 12(7).

[Tapia et al., 2016] Tapia, J. E., González, B., Goulitquer, S., Potin, P., and Correa, J. A. (2016). Microbiota Influences Morphology and Reproduction of the Brown Alga Ectocarpus sp. *Frontiers in Microbiology*, 7(FEB):197.

[Thacker and Freeman, 2012] Thacker, R. W. and Freeman, C. J. (2012). Sponge-Microbe Symbioses. Recent Advances and New Directions. *Advances in Marine Biology*, 62:57–111.

[Thiele et al., 2013a] Thiele, I., Heinken, A., and Fleming, R. M. (2013a). A systems biology approach to studying the role of microbes in human health. *Current Opinion in Biotechnology*, 24(1):4–12.

[Thiele and Palsson, 2010] Thiele, I. and Palsson, B. Ø. (2010). A protocol for generating a high-quality genome-scale metabolic reconstruction. *Nature protocols*, 5(1):93–121.

[Thiele et al., 2018] Thiele, I., Sahoo, S., Heinken, A., Heirendt, L., Aurich, M. K., Noronha, A., and Fleming, R. M. T. (2018). When metabolism meets physiology: Harvey and Harvetta. *bioRxiv*, page 255885.

[Thiele et al., 2013b] Thiele, I., Swainston, N., Fleming, R. M., Hoppe, A., Sahoo, S., Aurich, M. K., Haraldsdottir, H., Mo, M. L., Rolfsson, O., Stobbe, M. D., Thorleifsson, S. G., Agren, R., Bölling, C., Bordel, S., Chavali, A. K., Dobson, P., Dunn, W. B., Endler, L., Hala, D., Hucka, M., Hull, D., Jameson, D., Jamshidi, N., Jonsson, J. J., Juty, N., Keating, S., Nookaew, I., Le Novère, N., Malys, N., Mazein, A., Papin, J. A., Price, N. D., Selkov, E., Sigurdsson, M. I., Simeonidis, E., Sonnenschein, N., Smallbone, K., Sorokin, A., Van Beek, J. H., Weichart, D., Goryanin, I., Nielsen, J., Westerhoff, H. V., Kell, D. B., Mendes, P., and Palsson, B. O. (2013b). A community-driven global reconstruction of human metabolism. *Nature Biotechnology*, 31(5):419–425.

[Thiele et al., 2014] Thiele, I., Vlassis, N., and Fleming, R. M. T. (2014). fastGapFill: efficient gap filling in metabolic networks. *Bioinformatics (Oxford, England)*, 30(17):2529–2531.

[Tremaroli and Bäckhed, 2012] Tremaroli, V. and Bäckhed, F. (2012). Functional interactions between the gut microbiota and host metabolism. *Nature*, 489(7415):242–249.

[van der Ark et al., 2017] van der Ark, K. C. H., van Heck, R. G. A., Martins Dos Santos, V. A. P., Belzer, C., and de Vos, W. M. (2017). More than just a gut feeling: constraint-based genome-scale metabolic models for predicting functions of human intestinal microbes. *Microbiome*, 5(1):78.

[Van Der Heijden et al., 2008] Van Der Heijden, M. G., Bardgett, R. D., and Van Straalen, N. M. (2008). The unseen majority: Soil microbes as drivers of plant diversity and productivity in terrestrial ecosystems.

[Venter et al., 2001] Venter, J. C., Adams, M. D., Myers, E. W., Li, P. W., Mural, R. J., Sutton, G. G., Smith, H. O., Yandell, M., Evans, C. A., Holt, R. A., Gocayne, J. D., Amanatides, P., Ballew, R. M., Huson, D. H., Wortman, J. R., Zhang, Q., Kodira, C. D., Zheng, X. H., Chen, L., Skupski, M., Subramanian, G., Thomas, P. D., Zhang, J., Gabor Miklos, G. L., Nelson, C., Broder, S., Clark, A. G., Nadeau, J., McKusick, V. A., Zinder, N., Levine, A. J., Roberts, R. J., Simon, M., Slayman, C., Hunkapiller, M., Bolanos, R., Delcher, A., Dew, I., Fasulo, D., Flanigan, M., Florea, L., Halpern, A., Hannenhalli, S., Kravitz, S., Levy, S., Mobarry, C., Reinert, K., Remington, K., Abu-Threideh, J., Beasley, E., Biddick, K., Bonazzi, V., Brandon, R., Cargill, M., Chandramouliswaran, I., Charlab, R., Chaturvedi, K., Deng, Z., Francesco, V. D., Dunn, P., Eilbeck, K., Evangelista, C., Gabrielian, A. E., Gan, W., Ge, W., Gong, F., Gu, Z., Guan, P., Heiman, T. J., Higgins, M. E., Ji, R.-R., Ke, Z., Ketchum, K. A., Lai, Z., Lei, Y., Li, Z., Li, J., Liang, Y., Lin, X., Lu, F., Merkulov, G. V., Milshina, N., Moore, H. M., Naik, A. K., Narayan, V. A., Neelam, B., Nusskern, D., Rusch, D. B., Salzberg, S., Shao, W., Shue, B., Sun, J., Wang, Z. Y., Wang, A., Wang, X., Wang, J., Wei, M.-H., Wides, R., Xiao, C., Yan, C., Yao, A., Ye, J., Zhan, M., Zhang, W., Zhang, H., Zhao, Q., Zheng, L., Zhong, F., Zhong, W., Zhu, S. C., Zhao, S., Gilbert, D., Baumhueter, S., Spier, G., Carter, C., Cravchik, A., Woodage, T., Ali, F., An, H., Awe, A., Baldwin, D., Baden, H., Barnstead, M., Barrow, I., Beeson, K., Busam, D., Carver, A., Center, A., Cheng, M. L., Curry, L., Danaher, S., Davenport, L., Desilets, R., Dietz, S., Dodson, K., Doup, L., Ferriera, S., Garg, N., Gluecksmann, A., Hart, B., Haynes, J., Haynes, C., Heiner, C., Hladun, S., Hostin, D., Houck, J., Howland, T., Ibegwam, C., Johnson, J., Kalush, F., Kline, L., Koduru, S., Love, A., Mann, F., May, D., McCawley, S., McIntosh, T., McMullen, I., Moy, M., Moy, L., Murphy, B., Nelson, K., Pfannkoch, C., Pratts, E., Puri, V., Qureshi, H., Reardon, M., Rodriguez, R., Rogers, Y.-H., Romblad, D., Ruhfel, B., Scott, R., Sitter, C., Smallwood, M., Stewart, E., Strong, R., Suh, E., Thomas, R., Tint, N. N., Tse, S., Vech, C., Wang, G., Wetter, J., Williams, S., Williams, M., Windsor, S., Winn-Deen, E., Wolfe, K., Zaveri, J., Zaveri, K., Abril, J. F., Guigó, R., Campbell, M. J., Sjolander, K. V., Karlak, B., Kejariwal, A., Mi, H., Lazareva, B., Hatton, T., Narechania, A., Diemer, K., Muruganujan, A., Guo, N., Sato, S., Bafna, V., Istrail, S., Lippert, R., Schwartz, R., Walenz, B., Yooseph, S., Allen, D., Basu, A., Baxendale, J., Blick, L., Caminha, M., Carnes-Stine, J., Caulk, P., Chiang, Y.-H., Coyne, M., Dahlke, C., Mays, A. D., Dombroski, M., Donnelly, M., Ely, D., Esparham, S., Fosler, C., Gire, H., Glanowski, S., Glasser, K., Glodek, A., Gorokhov, M., Graham, K., Gropman, B., Harris, M., Heil, J., Henderson, S., Hoover, J., Jennings, D., Jordan, C., Jordan, J., Kasha, J., Kagan, L., Kraft, C., Levitsky, A., Lewis, M., Liu, X., Lopez, J., Ma, D., Majoros, W., McDaniel, J., Murphy, S., Newman, M., Nguyen, T., Nguyen, N., Nodell, M., Pan, S., Peck, J., Peterson, M., Rowe, W., Sanders, R., Scott, J., Simpson, M., Smith, T., Sprague, A., Stockwell, T., Turner, R., Venter, E., Wang, M., Wen, M., Wu, D., Wu, M., Xia, A., Zandieh, A., and Zhu, X. (2001). The Sequence of the Human Genome. *Science*, 291(5507):1304–1351.

[Vijayakumar et al., 2017] Vijayakumar, S., Conway, M., Lió, P., and Angione, C. (2017). Seeing the wood for the trees: a forest of methods for optimization and omic-network integration in metabolic modelling. *Briefings in Bioinformatics*.

[Vitkin and Shlomi, 2012] Vitkin, E. and Shlomi, T. (2012). MIRAGE: a functional genomics-based approach for metabolic network model reconstruction and its application to cyanobacteria networks. *Genome biology*, 13(11):R111.

[Webster, 2014] Webster, N. S. (2014). Cooperation, communication, and co-evolution: Grand challenges in microbial symbiosis research. *Frontiers in Microbiology*, 5(APR):164.

[Werren et al., 2008] Werren, J. H., Baldo, L., and Clark, M. E. (2008). Wolbachia: Master manipulators of invertebrate biology.

[Widder et al., 2016] Widder, S., Allen, R. J., Pfeiffer, T., Curtis, T. P., Wiuf, C., Sloan, W. T., Cordero, O. X., Brown, S. P., Momeni, B., Shou, W., Kettle, H., Flint, H. J., Haas, A. F., Laroche, B., Kreft, J.-U., Rainey, P. B., Freilich, S., Schuster, S., Milferstedt, K., Van Der Meer, J. R., and Allen, R. (2016). Challenges in microbial ecology: building predictive understanding of community function and dynamics. *The ISME Journal*, 10(10):2557–2568.

[Xavier et al., 2017] Xavier, J. C., Patil, K. R., and Rocha, I. (2017). Integration of Biomass Formulations of Genome-Scale Metabolic Models with Experimental Data Reveals Universally Essential Cofactors in Prokaryotes. *Metabolic engineering*, 39:200–208.

[Xu et al., 2017] Xu, N., Ye, C., Chen, X., Liu, J., and Liu, L. (2017). Genome-scale metabolic modelling common cofactors metabolism in microorganisms. *Journal of Biotechnology*, 251:1–13.

[Xu and Zhao, 2018] Xu, Y. and Zhao, F. (2018). Single-cell metagenomics: challenges and applications. *Protein & Cell*, 9(5):501–510.

[Ye et al., 2015] Ye, N., Zhang, X., Miao, M., Fan, X., Zheng, Y., Xu, D., Wang, J., Zhou, L., Wang, D., Gao, Y., Wang, Y., Shi, W., Ji, P., Li, D., Guan, Z., Shao, C., Zhuang, Z., Gao, Z., Qi, J., and Zhao, F. (2015). Saccharina genomes provide novel insight into kelp biology. *Nature communications*, 6:6986.

[Yizhak et al., 2010] Yizhak, K., Benyamini, T., Liebermeister, W., Ruppin, E., and Shlomi, T. (2010). Integrating quantitative proteomics and metabolomics with a genome-scale metabolic network model. *Bioinformatics*, 26(12):i255–i260.

[Zakrzewski et al., 2012] Zakrzewski, P., Medema, M. H., Gevorgyan, A., Kierzek, A. M., Breitling, R., and Takano, E. (2012). MultiMetEval: comparative and multi-objective analysis of genome-scale metabolic models. *PloS One*, 7(12):e51511.

[Zarecki et al., 2014] Zarecki, R., Oberhardt, M. A., Reshef, L., Gophna, U., and Ruppin, E. (2014). A novel nutritional predictor links microbial fastidiousness with lowered ubiquity, growth rate, and cooperativeness. *PLoS computational biology*, 10(7):e1003726.

[Zelezniak et al., 2015] Zelezniak, A., Andrejev, S., Ponomarova, O., Mende, D. R., Bork, P., and Patil, K. R. (2015). Metabolic dependencies drive species co-occurrence in diverse microbial communities. *Proceedings of the National Academy of Sciences of the United States of America*, 112(20):6449–6454.

[Zhuang et al., 2011] Zhuang, K., Izallalen, M., Mouser, P., Richter, H., Risso, C., Mahadevan, R., and Lovley, D. R. (2011). Genome-scale dynamic modeling of the competition between Rhodoferax and Geobacter in anoxic subsurface environments. *The ISME Journal*, 5(2):305–316.

[Zomorrodi et al., 2014] Zomorrodi, A. R., Islam, M. M., and Maranas, C. D. (2014). d-OptCom: Dynamic Multi-level and Multi-objective Metabolic Modeling of Microbial Communities. *ACS Synthetic Biology*, 3(4):247–257.

[Zomorrodi and Maranas, 2012] Zomorrodi, A. R. and Maranas, C. D. (2012). OptCom: A multi-level optimization framework for the metabolic modeling and analysis of microbial communities. *PLoS Computational Biology*, 8(2):e1002363.

[Zuñiga et al., 2017] Zuñiga, C., Zaramela, L., and Zengler, K. (2017). Elucidation of complexity and prediction of interactions in microbial communities. *Microbial Biotechnology*, 10(6):1500–1522.

List of personal publications

[Aite et al., 2018] Aite, M., Chevallier, M., Frioux, C., Trottier, C., Got, J., Cortés, M. P., Mendoza, S. N., Carrier, G., Dameron, O., Guillaudeux, N., Latorre, M., Loira, N., Markov, G. V., Maass, A., and Siegel, A. (2018). Traceability, reproducibility and wiki-exploration for "à-la-carte" reconstructions of genome-scale metabolic models. *PLOS Computational Biology*, 14(5):e1006146.

[Dittami et al., 2018] Dittami, S. M., Corre, E., Brillet-Gueguen, L., Pontoizeau, N., Aite, M., Avia, K., Caron, C., Cho, C. H., Collen, J., Cormier, A., Delage, L., Doubleau, S., Frioux, C., Gobet, A., Gonzalez-Navarrete, I., Groisillier, A., Herve, C., Jollivet, D., KleinJan, H., Leblanc, C., Lipinska, A. P., Liu, X., Marie, D., Markov, G. V., Minoche, A. E., Monsoor, M., Pericard, P., Perrineau, M.-M., Peters, A. F., Siegel, A., Simeon, A., Trottier, C., Yoon, H. S., Himmelbauer, H., Boyen, C., and Tonon, T. (2018). The genome of Ectocarpus subulatus highlights unique mechanisms for stress tolerance in brown algae. *bioRxiv*, page 307165.

[Frioux et al., 2018a] Frioux, C., Fremy, E., Trottier, C., and Siegel, A. (2018a). Scalable and exhaustive screening of metabolic functions carried out by microbial consortia. *Bioinformatics*, 34(17):i934–i943.

[Frioux et al., 2017] Frioux, C., Schaub, T., Schellhorn, S., Siegel, A., and Wanko, P. (2017). Hybrid metabolic network completion. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 10377 LNAI, pages 308–321. Springer, Cham.

[Frioux et al., 2018b] Frioux, C., Schaub, T., Schellhorn, S., Siegel, A., and Wanko, P. (2018b). Hybrid Metabolic Network Completion. *Theory and Practice of Logic Programming - in press*. Preprint available on: https://arxiv.org/abs/1808.04149.

[Laniau et al., 2017] Laniau, J., Frioux, C., Nicolas, J., Baroukh, C., Cortes, M.-P. M.-P., Got, J., Trottier, C., Eveillard, D., and Siegel, A. (2017). Combining graph and flux-based structures to decipher phenotypic essential metabolites within metabolic networks. *PeerJ*, 5(10):e3860.

[Prigent et al., 2017] Prigent, S., Frioux, C., Dittami, S. M., Thiele, S., Larhlimi, A., Collet, G., Gutknecht, F., Got, J., Eveillard, D., Bourdon, J., Plewniak, F., Tonon, T., and Siegel, A. (2017). Meneco, a Topology-Based Gap-Filling Tool Applicable to Degraded Genome-Wide Metabolic Networks. *PLOS Computational Biology*, 13(1):e1005276.

Appendices

Appendix A

Validation of putative interactions between *E. siliculosus* and *Ca.* P. ectocarpi

The table is extracted from [Prigent et al., 2017]. For every metabolite that becomes producible for *Ectocarpus siliculosus* only through the cooperation with *Candidatus* Phaeomarinobacter ectocarpi, the essential reactions of the bacterium for the metabolic function are computed and manually studied (Simon Dittami, Station Biologique de Roscoff). This leads to classify the targets on the likeliness of the algal-bacterial cooperation involved for their producibility.

Local compound ID	Metacyc ID	Poducible by Ectogem ?	Producible by holobiont network ?	No. Essential reactions	Local ID of essential reactions	Metacyc Ids of essential reactions	Relevance of prediction / result of manual curation	Classification of errors	Conclusion
META23523	AGMATHINE	no	yes	1	META55371	ARGDECARBOX-RXN	Possibly: *E. siliculosus* encodes two genes involved in agmathine degradation (synthesis of polyamines), but cannot produce agmathine.	_	possible interaction
META18647	4-FUMARYL-ACETOACETATE	no	yes	1	META55928	MALEYLACETOACETATE-ISOMERASE-RXN	Possibly: The *E. siliculosus* genome encodes the complete tyrosine degradation pathway except for one step, EC 5.2.1.2. The *Ca.* P. ectocarpi genome comprises this reaction.	_	possible interaction
META21813	HISTIDINAL	no	yes	1	META50465	HISTIDPHOS-RXN	Possibly: EC 3.1.3.15 is also missing in other *Ectocarpus* strains and in *Saccharina,* but a corresponding enzyme was found in diatoms. Brown algae may rely on external histidine or histidinol, posssibly from bacteria.	_	possible interaction
META21814	HISTIDINOL	no	yes	1					
META22289	UROCANATE	no	yes	1					
META23618	HIS	no	yes	1					
META29176	AMINO-OH-HYDROXYMETHYL-DIHYDROPTERIDINE	no	yes	0*	_	_	Possible: These compounds are intermediates in tetrafolate biosynthesis. The last step in this pathway (EC 1.5.1.3) is encoded in the algal genome, but EC 6.3.2.12 is missing and may be provided by the bacterium.	_	possible interaction
META29204	DIHYDROPTERIN-CH2OH-PP	no	yes	0*					
META29211	7-8-DIHYDROPTEROATE	no	yes	0*					
META20387	CPD-597 = N-carbamoylputrescine	no	yes	0	_	_	Possibly: This compound can be produced from agmatine via the activity of an agmatine deaminase (EC 3.5.3.12, Esi0055_0036) and further converted to putrescine (EC3.5.1.53, Esi0030_0077). However, as mentioned above, an arginine decarboxylase necessary to synthesize agmatine from arginine is missing in the algal genome. The bacterium possesses a corresponding enzyme (Phect1139).	_	possible interaction
META23545	SPERMIDINE	no	yes	0	_	_	Possibly: The *E. siliculosus* genome encodes several good candidate genes for spermidine synthesis from putrescine (EC 2.5.1.16, Esi0000_0445), but putrescine synthesis in *E. siliculosus* probably relies on an external source of Agmatine (see above)	_	possible interaction
META20108	CPD-313 = propane-1,3-diamine	no	yes	0	_	_	Possibly: All compounds are related to beta-alanine synthesis. Beta-alanine is required for Vitamin B5 production in the alga and may be provided by the bacterium.	_	possible interaction
META20395	CPD-6082	no	yes	0					
META23942	B-ALANINE	no	yes	0					
META26478	CPD-330 = L-galactono-1,4-lactone	no	yes	0	_	_	Possibly: The *E. siliculosus* genome comprises several genes potentially involved in ascorbate sysnthesis via the L-galactose pathway (PWY-882). However, a few essential reactions are missing, notably EC 2.7.7.69, EC 1.1.1.316, and EC 2.7.7.13. *Ca.* Phaeomarinobacter is capable of producing ascorbate via the ascorbate biosynthesis pathway IV (PWY3DJ).	_	possible interaction
META30310	ASCORBATE	no	yes	0					
META29497	CPD-318 = monodehydroascorbate radical	no	yes	0					
META21893	L-DEHYDRO-ASCORBATE	no	yes	0					
META20061	CPD-237 = indole-3-acetamide	no	yes	0	_	_	Possibly: please refer to Figure 4 of the following publication of further information: http://journal.frontiersin.org/article/10.3389/fgene.2014.00241/abstract	_	possible interaction
META19429	CPD-12763 = 5-aminopentanal	no	yes	1	META50177	LYSDECARBOX-RXN	Unlikely: This compound was added to the list of targets because RXN-11784 was associated with Esi0076_0061, but the specificity of the enzyme is unknown. There is no evidence that brown algae produce biogenic amines.	_	insufficient information
META23533	Cadaverine		yes	1					
META22254	THIAMINE-P	no	yes	0*	_	_	Unlikely: Genes invoved in thiamine biosynthesis are present in the alga and the pathway is predicted. Only a cofactor required by the Thiamin-phosphate-phosphorylase, 2-(2-carboxy-4-methylthiazol-5-yl)ethyl phosphate, is not available in the alga. So far no eukaryotic and only one bacterial enzyme synthesizing this compound have been characterized. This makes a reliable identification of an algal gene based on sequence homology difficult.	_	insufficient information
META22617	CPD-9245 = palmitoleate	no	yes	10	META49353 META49615 META49359 META49976 META49402 META49454 META49305 META59828 META49619 META49613	RXN-10657 RXN-9655 RXN0-2145 RXN-10661 RXN-10660 RXN-9550 RXN0-2141 5.3.3.14-RXN RXN0-2144 RXN-10656	None: Synthesis of these fatty acids occurs via the following pathways in the alga: PWY-5156 (not predicted; synthesis ofpalmitoyl-CoA), followed by PWY-5366/PWY-6282 (predicted/partially predicted; synthesis of palmitoleate), and PWY-5973 (partially predicted, synthesis of cis-vaccenate). Missing predictions are essentially due to the fact that not all reactions have been annotated with EC numbers in the algal genome.	Missing algal reaction due to poor annotation	probably no interaction
META22641	CPD-9245 = cis-vaccenate	no	yes	12	as above + : META49820 META49300	as above + : RXN-9555 RXN-9557			
META24216	C1 = UDP-N-acetyl-α-D-muramoyl-L-alanyl-γ-D-glutamyl-meso-2,6-diaminopimeloyl-D-alanyl-D-alanine	no	yes	7	META59827 META56279 META52004 META60262 META50594 META55563 META54150	UDP-NACMURALGLDAPAALIG-RXN UDP-NACMURALGLDAPLIG-RXN UDP-NACMURALA-GLU-LIG-RXN DALADALALIG-RXN UDPNACETYLMURAMATEDEHYDROG-RXN UDP-NACMUR-ALA-LIG-RXN UDPNACETYLGLUCOSAMENOLPYRTRANS-RXN	None: These compounds are substrates for / intermediates in peptidoglycan biosynthesis. Peptidoglycans are components of bacterial cell walls but not expected in algae. They were added as targets because *E. siliculosus* expresses a gene annotated as correspondig to the PHOSNACMURPENTATRANS-RXN, which may consume/produce these compounds.However, this is most likely due to a human error during annotation.	False targets due to poor annnotation	probably no interaction
META24389	C5 = undecaprenyldiphospho-N-acetylmuramoyl-L-alanyl-γ-D-glutamyl-meso-2,6-diaminopimeloyl-D-alanyl-D-alanine	no	yes	7					
META28526	CPD-9646 = di-trans,octa-cis-undecaprenyl phosphate	no	yes	0					
META22459	PALMITATE	no	yes	3	META49615 META49818 META49612	RXN-9655 RXN-9533 RXN-9537	None: All three reactions are carried out by the fatty acid synthase complex. This enzyme is probably present in *Ectocarpus* : Esi0520_0008 is a good candiate.	Missing algal reaction due to poor annotation	probably no interaction
META22460	STEARIC_ACID	no	yes	3					
META22853	DEHYDROSPHINGANINE	no	yes	3					
META22859	CPD-13612 = sphinganine	no	yes	3					
META22860	CPD-13613 = L-threo-sphinganine	no	yes	3					
META26830	PALMITYL-COA	no	yes	3					
META26885	CPD0-2117 = a long-chain trans-2,3-dehydroacyl-CoA	no	yes	3					
META18690	5-AMINO-LEVULINATE	no	yes	1	META53378	5-AMINOLEVULINIC-ACID-SYNTHASE-RXN	None: the 5-AMINOLEVULINIC-ACID-SYNTHASE-RXN is necessary to synthesize tetrapyrole from glycine, but there is no indication for the presence of the corresponding enzyme in the genome. There is an alternate pathway for production from glutamate (PWY-5188) which appears complete in *E. siliculosus* . This pathway was not automatically predicted becausea GLT -tRNA was not annotated in the genome , but it is the most probable way of Tetrapyrol biosynthesis in *E. siliculosus* .	Missing algal reaction due to poor annotation	probably no interaction
META18966	COPROPORPHYRINOGEN_III	no	yes	1					
META21833	HYDROXYMETHYLBILANE	no	yes	1					
META22102	PORPHOBILINOGEN	no	yes	1					
META28671	MG-PROTOPORPHYRIN	no	yes	1					
META28679	PROTOPORPHYRINOGEN	no	yes	1					
META28680	PROTOPORPHYRIN_IX	no	yes	1					
META28681	UROPORPHYRINOGEN-III	no	yes	1					
META28673	COPROPORPHYRINOGEN_I	no	yes	2	META53378	5-AMINOLEVULINIC-ACID-SYNTHASE-RXN			
META28675	CPD-11444 = a porphyrin	no	yes	2	META55357	RXN-14396			
META23249	CPD-10330 = α-D-ribofuranose = ribose-1-phosphate	no	yes	1	META54065	RXN-14904	This reaction most likely occurs spontaneously but had not been included in the *E. siliculosus* metabolic network.	Other	probably no interaction
META26743	ISOVALERYL-COA	no	yes	1	META62215	2KETO-4METHYL-PENTANOATE-DEHYDROG-RXN	None: A good candidate gene (Esi0000_0413) for the missing reaction was found in *E. siliculosus,* but was had not been annotated accordingly.	Missing algal reaction due to poor annotation	probably no interaction
META28546	OCTAPRENYL-DIPHOSPHATE = precursor of the compound below	no	yes	1	META51463	RXN-8992 = octaprenyl/solanesyl diphosphate synthase (EC 2.5.1.90).	None: The bacterial enzyme EC 2.5.1.90 is necessary for the synthesis of Ubiquinone 8, but *E. siliculosus* possesses all genes necessary for Ubiquinone 9 synthesis (Esi0020_0133 and Esi0166_0045). These compounds were added as targets because of the association of the expressed gene Esi0552_0015 with reaction EC 2.5.1.39, but the specifictiy of the enzme is difficult to predict based on sequence information.	False targets due to poor annnotation	probably no interaction
META28521	3-OCTAPRENYL-4-HYDROXYBENZOATE	no	yes	1					
META18855	AMINO-RIBOSYLAMINO-1H-3H-PYR-DIONE	no	yes	1	META59243	RIBOPHOSPHAT-RXN	None: Phosphatases catalyzing this reaction are poorly characterized in eukaryotes and were therefore not detected in the algal metabolic network. However, several phosphatases of unknown specificity are present, and there is no compelling reason to assume that the alga relies on bacteria for FAD synthesis.	Missing algal reaction due to poor annotation	probably no interaction
META22336	CPD-12175 = (S)-3-hydroxy-isobutanoate	no	yes	1					
META26189	CH3-MALONATE-S-ALD	no	yes	1					
META26748	METHACRYLYL-COA	no	yes	1					
META26754	CPD-12173 = (S)-3-hydroxy-isobutanoyl-CoA	no	yes	1					
META29163	RIBOFLAVIN	no	yes	1					
META29164	META29164	no	yes	1					
META29165	FMN	no	yes	1					
META29166	CPD-316 = reduced riboflavin	no	yes	1					
META29167	FADH2	no	yes	1					
META29168	FMNH2	no	yes	1					
META29171	DIMETHYL-D-RIBITYL-LUMAZINE	no	yes	1					
META24487	CPD-15318 = α-D-ribose 5-phosphate	no	yes	0*	_	_	None: This compound may be produced by the alga via the activity of a ribokinase (2.7.1.15, Esi0018_0080) associated with the RIBOKIN-RXN. The product of this reaction is a compound class comprising both alpha- and beta-D-ribose 5-phosphate.	Other	probably no interaction
META26753	TRANS-3-METHYL-GLUTACONYL-COA	no	yes	0*	_	_	None: A coorsponding reaction to synthesize this compound is correctly predicted in the algal network, but lacks HCO3 as a cofactor. HCO3 is naturally present in seawater, but was not included in the list of seeds provided to meneco.	Other	probably no interaction
META18633	3OH-4P-OH-ALPHA-KETOBUTYRATE	no	yes	0*	_	_	None: There is an alternative pathway for Pyridoxal-5'-phosphate biosynthesis involving only a single reaction in the alga (PWY-6466), The corresponding reaction is catalyzed by a single enzyme EC 4.3.3.6, which is present in the *E. siliculosus* genome, but had not been annotated at the time of network reconstruction (Esi0185_0038).	Missing algal reaction due to poor annotation	probably no interaction
META18670	4-PHOSPHONOOXY-THREONINE	no	yes	0*					
META30330	PYRIDOXINE-5P	no	yes	0*					
META30326	PYRIDOXAL_PHOSPHATE	no	yes	0*					
META18930	CARBON-MONOXIDE	no	yes	0*	_	_	None: This is a byproduct of the HEME-OXYGENASE-DECYCLIZING-RXN, thought to be encoded by Esi0140_0061 in the algal genome. However, there is little evidence for the actual occurance of this reaction in the alga (enzyme specificity difficult to determine based on sequence homology).	False targets due to poor annnotation	probably no interaction
META20191	CPD-385 = 1,2-benzoquinone	no	yes	0*			None: Both compounds were added as targets based on the presence of the CATECHOL-OXIDASE-RXN (EC 1.10.3.1) in the algal network. The associated algal genes, however, correspond mainly to Tyrosinase kinases (EC 1.14.18.1) or	False targets due to poor	probably no

META23014	CATECHOL	no	yes	0*	_	_	unknown genes. "Tyrosinase kinase" is also considered a synonym for "Catechol oxidase" (EC 1.10.3.1) leading to a false association with this reaction by pathway tools.	to poor annnotation	interaction
META22399	CPD-110 = salicylate	no	yes	0*	_	_	None: This target was added because it may be produced via the action of a Carboxylesterase present and expressed in the *E. siliculosus* genome. However there is no evidence of salicylate in the alga, and the specificity of the carboxylesterase cannot be determined pureley based on sequence homology.	False targets due to poor annnotation	probably no interaction
META22473	BUTYRIC_ACID	no	yes	0*	_	_	None: BUTYRIC_ACID was added to the list of targets based on the automatic annotation of the expressed gene Esi0062_0097 as putative triacylglycerol lipase. However, this annotation is not justified due to low similarity with charaterized enzymes and as charateristic domains are missing. Butyryl-COA was included as target based on a reaction added by Pathway Tools Gapfilling, K-Hexaonyl-COA, based on the presence of an expressed enzyme who's specificity is difficult to determine based on sequence data (Esi0320_0011) and OH-HEXOANYL-COA based on the presence of an enzyme with a broad range of substrates (EC 1.1.1.35, Esi0063_0042).	False targets due to poor annnotation	probably no interaction
META26845	BUTYRYL-COA	no	yes	0*					
META26776	K-HEXANOYL-COA	no	yes	0*					
META26810	OH-HEXANOYL-COA	no	yes	0*					
META22719	OROTATE	no	yes	0*	_	_	None: This compound may be produced by the alga alone via the activity of EC 3.5.2.3. (Esi0000_0145), which has not been annotated with the corresponding EC number.	Missing algal reaction due to poor annotation	probably no interaction
META23105	PROPANOL	no	yes	0*	_	_	None: This compound was added to the list of targets based on the presence of an expressed alcohol deshydrogenase (EC 1.1.1.1) in the algal genome. The exact function of these enzymes, however is unknown.	False targets due to poor annnotation	probably no interaction
META23316	CPD-665 = propanal	no	yes	0*					
META23317	CPD-7000 = isobutanal	no	yes	0	_	_	None: Isobutanal was added as a target because RXN-7657 was predicted to be present and expressed in the alga. However, this prediction is made purely based on the presence of an alcohol dehydrogenase domain in two otherwise uncharacterized proteins.	False targets due to poor annnotation	probably no interaction
META21865	ISOBUTANOL	no	yes	0					
META22231	SUCC-S-ALD	no	yes	0	_	_	None: There is little evidence supporting the presence of this compound among the targets: the specificity of the expressed genes leading to its inclusion (4-HYDROXY-2-KETOPIMELATE-LYSIS-RXN, SUCCSEMIALDDEHYDROG-RXN) can currently not be relyably determined based on homology with charaterized sequences.	False targets due to poor annnotation	probably no interaction
META23539	CPD-14378 = dehydrospermidine	no	yes	0	_	_	None: There is very little support for the oxidation of spermidine to deshydrospermidine in *E. siliculosus* (the associated gene is poorly annotated). Thus there is no compelling reason to keep dehydrospermidine as a target.	False targets due to poor annnotation	probably no interaction
META23661	O-SUCCINYL-L-HOMOSERINE	no	yes	0	_	_	None: This compound was added to the list of targets because it is used in three reactions supported by expressed genes in the alga. In all three cases genes were associated with the reaction based on sequence homology, but are more closely realted to enzymes with other functions. We therefore currently have no evidence that o-succinyl-l-homoserine occurs in *Ectocarpus* and should be kept as a target.	False targets due to poor annnotation	probably no interaction
META23887	CPD0-2189 = 4-hydroxy-L-threonine	no	yes	0	_	_	None: This compound was added to the algal network based on the presence of an expressed gene (Esi0427_0005) associated with reaction RXN-14125 (4-phospho-hydroxy-L-threonine synthesis). This association was made automatically based on sequence homology. The enzyme corrsponding to Esi0427_0005, however, more likely constitutes a Threonine synthase.	False targets due to poor annnotation	probably no interaction

* No "essential" reactions have been identified to poduce these metabolites. At least two alternative reactions are available to produce this compound.

🟨	Insufficient information
🟥	Probably no interaction - false positive
🟩	Possible interaction

Appendix B

Taxonomy of the selected gut bacteria

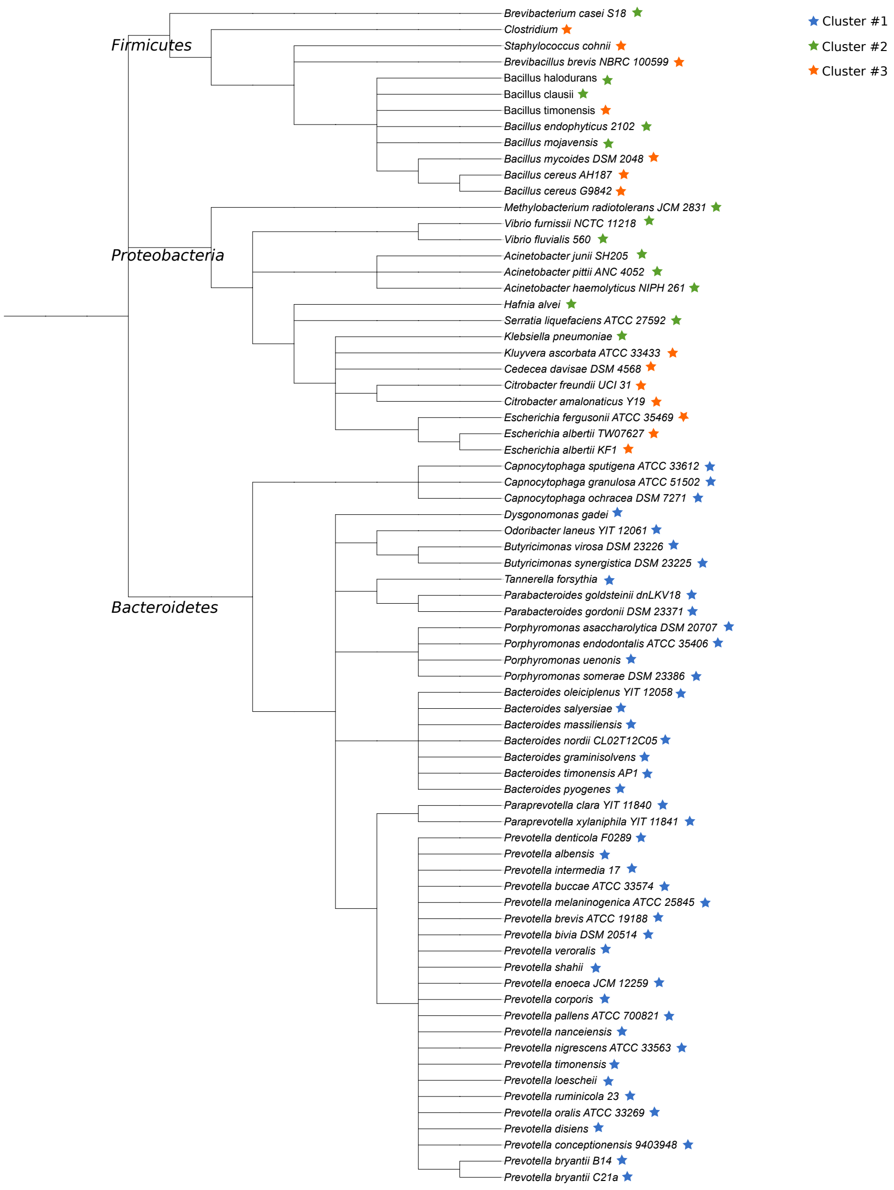

Cluster #1
Cluster #2
Cluster #3
Firmicutes
Proteobacteria
Bacteroidetes
Brevibacterium casei S18
Clostridium
Staphylococcus cohnii
Brevibacillus brevis NBRC 100599
Bacillus halodurans
Bacillus clausii
Bacillus timonensis
Bacillus endophyticus 2102
Bacillus mojavensis
Bacillus mycoides DSM 2048
Bacillus cereus AH187
Bacillus cereus G9842
Methylobacterium radiotolerans JCM 2831
Vibrio furnissii NCTC 11218
Vibrio fluvialis 560
Acinetobacter junii SH205
Acinetobacter pittii ANC 4052
Acinetobacter haemolyticus NIPH 261
Hafnia alvei
Serratia liquefaciens ATCC 27592
Klebsiella pneumoniae
Kluyvera ascorbata ATCC 33433
Cedecea davisae DSM 4568
Citrobacter freundii UCI 31
Citrobacter amalonaticus Y19
Escherichia fergusonii ATCC 35469
Escherichia albertii TW07627
Escherichia albertii KF1
Capnocytophaga sputigena ATCC 33612
Capnocytophaga granulosa ATCC 51502
Capnocytophaga ochracea DSM 7271
Dysgonomonas gadei
Odoribacter laneus YIT 12061
Butyricimonas virosa DSM 23226
Butyricimonas synergistica DSM 23225
Tannerella forsythia
Parabacteroides goldsteinii dnLKV18
Parabacteroides gordonii DSM 23371
Porphyromonas asaccharolytica DSM 20707
Porphyromonas endodontalis ATCC 35406
Porphyromonas uenonis
Porphyromonas somerae DSM 23386
Bacteroides oleiciplenus YIT 12058
Bacteroides salyersiae
Bacteroides massiliensis
Bacteroides nordii CL02T12C05
Bacteroides graminisolvens
Bacteroides timonensis AP1
Bacteroides pyogenes
Paraprevotella clara YIT 11840
Paraprevotella xylaniphila YIT 11841
Prevotella denticola F0289
Prevotella albensis
Prevotella intermedia 17
Prevotella buccae ATCC 33574
Prevotella melaninogenica ATCC 25845
Prevotella brevis ATCC 19188
Prevotella bivia DSM 20514
Prevotella veroralis
Prevotella shahii
Prevotella enoeca JCM 12259
Prevotella corporis
Prevotella pallens ATCC 700821
Prevotella nanceiensis
Prevotella nigrescens ATCC 33563
Prevotella timonensis
Prevotella loescheii
Prevotella ruminicola 23
Prevotella oralis ATCC 33269
Prevotella disiens
Prevotella conceptionensis 9403948
Prevotella bryantii B14
Prevotella bryantii C21a

Figure B.1: ***Taxonomic tree of the 89 gut bacteria***

The 381 minimal communities include a set of 89 bacteria. Their taxonomic tree is presented here, together with their clustering information.

Appendix C

MeneTools

MeNeTools is a Python package (https://pypi.org/project/MeneTools/). It is a toolbox that enables to investigate the properties of a GSM with respect to the graph-based definition of functionality. All MeNeTools rely on Answer Set Programming (ASP) for the analysis of the model topology. The link between Python and ASP is ensured by the use of the PyASP package (https://pypi.org/project/pyasp/).

The functioning of the MeNeTools will be explained using the following metabolic model, depicted in Figure C.1

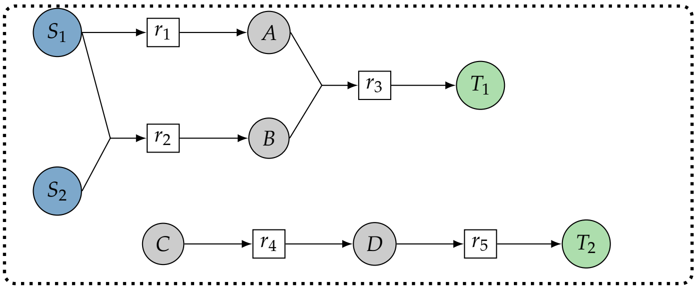


Figure C.1: *Toy example for* ***MeNeTools***

This small model contains 5 reactions and 8 metabolites among which are 2 seeds $\{S_1, S_2\}$ (blue) and 2 targets $\{T_1, T_2\}$ (green).

Menescope

Menescope yields the set of metabolites that belong to the scope of a set of seeds, following the definition 1.2. It takes as an input a GSM and seeds, both in SBML format.

```
scope(M) :- product(M,R), reaction(R), scope(M2) : reactant(M2,R).
scope(M) :- reactant(M,R), reaction(R), reversible(R), scope(M2) : product(M2,R).
scope(M) :- seed(M).

#show scope/1.
```

Listing 4: *ASP encoding of Menescope*

Applied to the example of Figure C.1 in Figure C.2, it yields a scope composed of metabolites $\{S_1, S_2, A, B, T_1\}$.

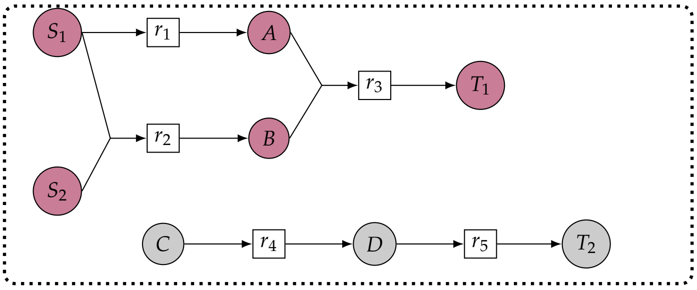


Figure C.2: ***Menescope toy example***

Pink metabolites are in the scope of the seeds.

Menecheck

Menecheck relies on the previous definition of Menescope to test whether elements of a set of targets are reachable (i.e. producible) or not. It gives to the user two lists, one gathering the unproducible targets, the other gathering the producible ones.

```
% what is producible by all reactions

scope(M) :- seed(M).

scope(M) :- product(M,R), reaction(R),
           dscope(M2) : reactant(M2,R).

scope(M) :- reactant(M,R), reaction(R), reversible(R),
           dscope(M2) : product(M2,R).


% unproducible targets do not belong to the scope

unproducible_target(M) :- target(M), not scope(M).

% unproducible targets belong to the scope

producible_target(M) :- target(M), scope(M).


#show unproducible_target/1.
#show producible_target/1.
```

Listing 5: *ASP encoding of Menecheck*

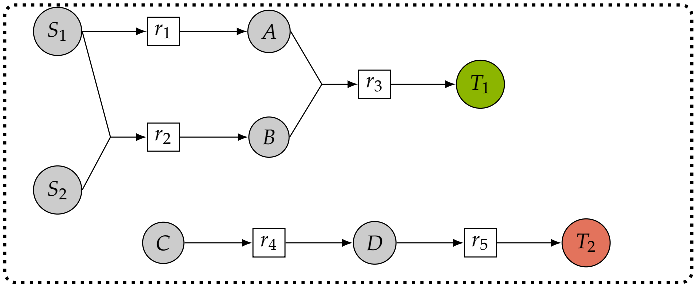


Figure C.3: ***Menecheck toy example***

Only target T_1 is in the scope of the seeds, hence it is producible (green metabolite), contrary to target T_2 that is unproducible (red metabolite).

Applied to the example of Figure C.1 in Figure C.3, only one target is producible, T_1, while the second one, T_2 is unproducible.

Menepath

Menepath aims at explaining the producibility of target compounds by giving a subset of the model that is sufficient to produce the target from the seeds. The objective is to isolate the reactions that are needed to produce a metabolite.

```
% what is producible in the network

  scope(M) :- seed(M).

  scope(M) :- product(M,R), dreaction(R), scope(M2) : reactant(M2,R).

  scope(M) :- reactant(M,R), dreaction(R), draft(N), reversible(R), scope(M2) : product(M2,R).

% predecessors of targets
% = reactant of a reaction that has at least one predecessor among its products

  predecessor(T,T) :- target(T).
  predecessor(M,T) :- reactant(M,R), dreaction(R), product(M1,R), predecessor(M1,T).
  predecessor(M,T) :- product(M,R), dreaction(R), reversible(R), reactant(M1,R), predecessor(M1,T).


% a reaction belongs to the subnetwork that gather production paths of target T
% if all its reactants are in the scope and if at least one of its products
% is a predecessor of target T
  prodpath(R,T) :- target(T), predecessor(M,T), product(M,R),
                   dreaction(R) : reactant(M1,R), scope(M1).


% reactions belonging to the selected production path are chosen in the prodpath

  {selected(R,T) : prodpath(R,T)}.

% scope of selected path

  pscope(M,T) :- seed(M), target(T).

  pscope(M,T) :- product(M,R), selected(R,T), target(T), pscope(M2,T) : reactant(M2,R).

  pscope(M,T) :- reactant(M,R), selected(R,T), target(T),reversible(R),
                 pscope(M2,T) : product(M2,R).

% pscope must include target

  :- target(T), scope(T), not pscope(T,T).


%minimize the size of the path
  #minimize { 1@1,R : selected(R,T)}.

  #show selected/2.
```

Listing 6: *ASP encoding of Menepath*

The design of the ASP model was made in collaboration with Julie Laniau, PhD. The minimize is optional. Applied to the example of Figure C.1 in Figure C.4, the production path for target T_1 (the only one that is producible), involves reactions $\{r_1, r_2, r_3\}$.

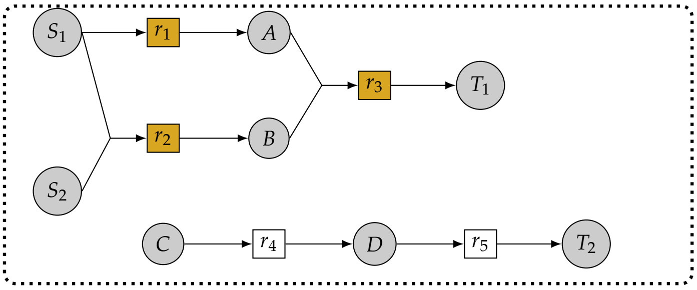


Figure C.4: ***Menepath toy example***

The path to target T_1 includes the reactions $\{r_1, r_2, r_3\}$ (yellow nodes). Target T_2 being unproducible, its path cannot be computed.

Menecof

Menecof proposes metabolites that, if made producible, would enable the producibility of targets. The objective is to pinpoint metabolites that could be set as objectives for gap-filling methods or be considered as initiation seeds for modeling.

There is a weighted version with weights derived from the occurrence of the metabolites in the MetaCyc database. One could also imagine putting the molecular weights into the ASP model.

Applied to the example of Figure C.1 in Figure C.5, the only unproducible target is T_2; a metabolite whose producibility would enable to produce the target is C.

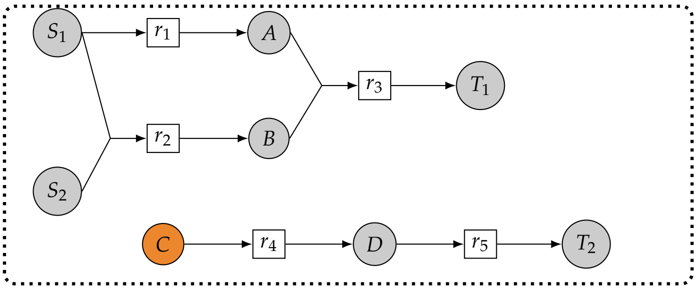


Figure C.5: ***Menecof toy example***

A minimal precursor of target T_2 that can enable its producibility is metabolite C. D is also a solution.

```
    % scope of the initial model
    dscope(M) :- product(M,R), reaction(R),
                dscope(M2) : reactant(M2,R).
    dscope(M) :- reactant(M,R), reaction(R), reversible(R),
                dscope(M2) : product(M2,R).
    dscope(M) :- seed(M).

    % scope of the model with any cofactor allowed
    allscope(M) :- product(M,R), reaction(R),
                allscope(M2) : reactant(M2,R).
    allscope(M) :- reactant(M,R), reaction(R), reversible(R),
                allscope(M2) : product(M2,R).
    allscope(M) :- seed(M).
    allscope(M) :- cofactor(M).

    % selected cofactors belong to the list of given cofactors
    {needed_cof(M) : cofactor(M)}.

     % scope of the model with the chosen cofactors
    xscope(M) :- product(M,R), reaction(R),
                xscope(M2) : reactant(M2,R).
    xscope(M) :- reactant(M,R), reaction(R), reversible(R),
                xscope(M2) : product(M2,R).
    xscope(M) :- seed(M).
    xscope(M) :- needed_cof(M).


    :- target(M), allscope(M), not xscope(M).

    still_unprod(M) :- target(M), not allscope(M).

    newly_prod(M) :- target(M), xscope(M), not dscope(M).

    % minimize the number of unproducible targets
        #minimize { 1@2,M : still_unprod(M) ; 0@3}.
    % minimize the number of cofactors which also are targets
        #minimize { 1@3, M : needed_cof(M,_), target(M)}.
    % minimize the number of cofactors
        #minimize { 1@1,R : needed_cof(R)}.


#show still_unprod/1.
#show needed_cof/1.
#show newly_prod/1.
```

Listing 7: *ASP encoding of Menecof*

Titre : Etude de la coopération hôte-microbiote par des problèmes d'optimisation basés sur la complétion de réseaux métaboliques

Mots clés : programmation logique – réseaux métaboliques – microbiote – ASP – complétion - modélisation

Résumé : La biologie des systèmes intègre données et connaissances par des méthodes bioinformatiques, afin de mieux appréhender la physiologie des organismes. Une problématique est l'applicabilité de ces techniques aux organismes non modèles, au centre de plus en plus d'études, grâce aux avancées de séquençage et à l'intérêt croissant de la recherche sur les microbiotes. Cette thèse s'intéresse à la modélisation du métabolisme par des réseaux, et de sa fonctionnalité par diverses sémantiques basées sur les graphes et les contraintes stoechiométriques. Une première partie présente des travaux sur la complétion de réseaux métaboliques pour les organismes non modèles. Une méthode basée sur les graphes est validée, et une seconde, hybride, est développée, en programmation par ensembles réponses (ASP). Ces complétions sont appliquées à des réseaux métaboliques d'algues en biologie marine, et étendues à la recherche de complémentarité métabolique entre *Ectocarpus siliculosus* et une bactérie symbiotique. En s'appuyant sur les méthodes de complétion, la seconde partie de la thèse vise à proposer et implémenter une sélection de communautés à l'échelle de grands microbiotes. Une approche en deux étapes permet de suggérer des symbiotes pour l'optimisation d'un objectif donné. Elle supporte la modélisation des échanges et couvre tout l'espace des solutions. Des applications sur le microbiote intestinal humain et la sélection de bactéries pour une algue brune sont présentées. Dans l'ensemble, cette thèse propose de modéliser, développer et appliquer des méthodes reposant sur des sémantiques de graphe pour élaborer des hypothèses sur le métabolisme des organismes.

Title: Investigating host-microbiota cooperation with gap-filling optimization problems

Keywords: logic programming – metabolic networks – microbiota – ASP – gap-filling – modeling

Abstract: Systems biology relies on computational biology to integrate knowledge and data, for a better understanding of organisms' physiology. Challenges reside in the applicability of methods and tools to non-model organisms, for instance in marine biology. Sequencing advances and the growing importance of elucidating microbiotas' roles, have led to an increased interest into these organisms. This thesis focuses on the modeling of the metabolism through networks, and of its functionality using graphs and constraints semantics. In particular, a first part presents work on gap-filling metabolic networks in the context of non-model organisms. A graph-based method is benchmarked and validated and a hybrid one is developed using Answer Set Programming (ASP) and linear programming. Such gap-filling is applied on algae and extended to decipher putative interactions between *Ectocarpus siliculosus* and a symbiotic bacterium. In this direction, the second part of the thesis aims at proposing formalisms and implementation of a tool for selecting and screening communities of interest within microbiotas. It enables to scale to large microbiotas and, with a two-step approach, to suggest symbionts that fit the desired objective. The modeling supports the computation of exchanges, and solving can cover the whole solution space. Applications are presented on the human gut microbiota and the selection of bacterial communities for a brown alga. Altogether, this thesis proposes modeling, software and biological applications using graph-based semantics to support the elaboration of hypotheses for elucidating the metabolism of organisms.