Geometric Simplification for Radiative Thermal Simulation of satellites

Vincent Vadez

UNIVERSITÉ CÔTE D'AZUR

ÉCOLE DOCTORALE
SCIENCES ET TECHNOLOGIES DE L'INFORMATION ET DE LA COMMUNICATION

THÈSE DE DOCTORAT

Simplification Géométrique pour la Simulation Thermique Radiative de satellites

Vincent VADEZ

Inria Sophia Antipolis – Méditerranée

Présentée en vue de l'obtention du grade de docteur en informatique d'Université Côte d'Azur
Dirigée par : Pierre Alliez
Soutenue le : 10 juin 2022

Devant le jury, composé de :

Sébastien Valette	Chargé de recherche	CNRS
Martin Raynaud	Ingénieur de recherche	Thales Alenia Space
Pierre Alliez	Directeur de recherche	Inria Sophia Antipolis
Laurent Busé	Directeur de recherche	Inria Sophia Antipolis
Jean-Baptiste Caillau	Professeur	Université Côte d'Azur
Harrie Rooijackers	Docteur	ATG Europe

DOREA Cifre ANRT ASSOCIATION NATIONALE RECHERCHE TECHNOLOGIE Inria

Simplification Géométrique pour la Simulation Thermique Radiative de satellites

Geometric model reduction for radiative thermal simulation of satellites

Jury:

Rapporteurs
Sébastien Valette, Chargé de recherche, CNRS
Martin Raynaud, Ingénieur de recherche, Thales Alenia Space

Examinateurs
Pierre Alliez, Directeur de recherche, Inria Sophia Antipolis
Laurent Busé, Président du jury, Directeur de recherche, Inria Sophia Antipolis
Jean-Baptiste Caillau, Professeur, Université Côte d'Azur
Harrie Rooijackers, Docteur, ATG Europe détaché auprès de l'Agence Spatiale Européenne

Invités
Samuel Hornus, Chargé de recherche, Inria Nancy
François Brunetti, Dirigeant, Dorea Technology

Abstract

The life cycle of a satellite includes the launch phase, the positioning on the desired orbit, different maneuvers (e.g. deployment of solar panels and safety position), and finally placing the satellite on the junk orbit. The satellite gravitates in a hostile environment, exposed to thermal variations of very large amplitude, alternating sun exposure and eclipse phases. The survival of the satellite depends on the temperature of its components, the variation of which must be monitored within safety intervals. In this context, the thermal simulation of the satellite for its design is crucial to anticipate the reality of its operation. Radiative thermal simulation is essential for anticipating the generation of energy from solar and albedo radiation, and for regulating temperatures of on-board equipments. Ideal operation consists in providing appropriate cooling for components exposed to radiation, and conversely, heating of unexposed components. As an order of magnitude, the external temperature ranges from -150 to +150 degrees Celsius, and the internal electronic equipment has for instance a safe range between -50 and +50 degrees Celsius, with a safety margin of 10 degrees. In the eclipse phase where the radiation is significantly lower, heating is provided by the energy accumulated during the exposed phase, combined with heat pipes for thermal regulation.

In this thesis, the objective is to advance the knowledge on radiative thermal simulation calculation methods for satellites. To this end, two approaches are considered. The first approach consists in establishing a reference calculation of a quantity governing radiative thermal simulation: view factors. Being subject to time constraints, this method is based on a hierarchical data structure enabling progressive computation of view factors, in order to offer a satisfactory tradeoff between time dedicated to computations and desired accuracy. For the sake of accuracy, a prediction step is added to guarantee a better convergence towards the reference value.

The second approach, also motivated by time constraints, aims at reducing the geometric model of a mechanical part or a spacecraft while being faithful to the numerical simulation. In order to render the decimation physics-informed, a preprocessing step relying on a sensitivity analysis is carried out. To better preserve the physical simulation, the geometric cost of a simplification operator is coupled to a factor deduced from the simulation deviation between the reference model and the reduced model.

Keywords: satellites, radiative simulation, view factors, geometric reduction.

Résumé

Le cycle de vie d'un satellite comprend la phase de lancement, la mise en orbite, le fonctionnement avec déploiement des panneaux solaires (manoeuvres, positions de sécurité), et la mise en orbite de fin de vie. Le satellite gravite dans un environnement hostile, exposé à des variations thermiques de très grandes amplitudes, avec des alternances de phases d'éclipses et éclairées. La survie du satellite est liée à la température de ses composants, dont la variation doit être contrôlée dans des intervalles de sécurité. Dans ce contexte, la simulation thermique du satellite pour sa conception est cruciale pour anticiper le réel de son fonctionnement. Les échanges radiatifs sont indispensables au satellite à la fois pour la génération d'énergie à partir de rayonnements solaires et de l'albédo, et pour la régulation thermique des équipements. Un bon fonctionnement consiste à assurer une réfrigération des composants exposés aux rayonnements, et à l'inverse un réchauffement des composants non exposés. Pour donner un ordre de grandeur, les amplitudes externes vont de -150 à +150 degrés Celsius, et les équipements électroniques internes ont une amplitude de sécurité entre -50 et +50, avec une marge de sécurité de 10 degrés. En phase d'éclipse où le rayonnement est bien plus faible, le réchauffement est assuré par l'énergie accumulée en phase exposée, combinée à des caloducs.

Dans cette thèse, l'objectif est d'avancer les connaissances sur les méthodes de calcul de la simulation thermique radiative des satellites. Pour ce faire, deux approches sont considérées. La première approche consiste en l'établissement d'un calcul de référence pour des grandeurs gouvernant la simulation thermique radiative: les facteurs de vue. Etant soumise à des contraintes temporelles, cette méthode repose sur une structure de données hiérarchique permettant un calcul progressif des facteurs de vue afin d'optimiser le compromis entre temps dédié aux calculs et précision souhaitée. Par soucis de précision, une étape de prédiction est ajoutée pour garantir une meilleure convergence vers la valeur de référence.

La seconde approche, également motivée par des contraintes de temps de calculs, a pour objectif la réduction du modèle géométrique d'une pièce mécanique ou d'un engin spatial tout en étant fidèle à la simulation numérique. Afin de rendre la décimation informée de la physique, un prétraitement réalisant une analyse de sensibilité est effectué. Pour accentuer la préservation de la simulation physique, le coût géométrique d'un opérateur de simplification est couplé à un facteur déduit des écarts de simulation entre modèle de référence et modèle réduit.

Mots-clés: satellites, simulation radiative, facteurs de vue, réduction géométrique.

Acknowledgments

This project would not have been possible without the support of many people. I would like to deeply thank my PhD adviser, Pierre Alliez, for his patience and his guidance with the slow learner I am, and for his implication during the full duration of the PhD, in particular during this rough pandemic period. Many thanks also to my company adviser François Brunetti, CEO of Dorea, whose insight and knowledge into the subject matter steered me through this research, and who financed my PhD thesis. I will forever be grateful.

I wish to sincerely thank Sébastien Valette and Martin Raynaud, who agreed to review this thesis.

I would like to extend my deepest thanks to Laurent Busé and Radhia Gaddouri for agreeing to be part of my Personal Monitoring Committee, I really appreciated your valuable inputs and insights.

I give my most sincere thanks to Harrie Rooijackers, for his useful reviews of my presentations at the European Space Thermal Engineering Workshop and for helping me to relieve stress for my first presentation at the European Space Research and Technology Centre.

I am greatly thankful to my colleagues from Dorea for their positiveness and for finding some joy in the everyday. Shout out to *Le Bureau de l'ambiance*, Marine and Adrien, for making my stay at the company delightful but mostly for enduring my terrible jokes.

And also people from Inria, in particular the Titane team, with whom I had thrilling discussions about various research topics, notably Cédric for helping me regarding the mesh decimation part.

I take this opportunity to thank professors from Polytech Nice Sophia, in particular Jean-Baptiste Caillau, Cédric Boulbe and Didier Auroux for their teaching and for trusting me to be a teaching assistant at my former engineering school.

For the boys, Arnaud, Bertrand, Lois and Thibault, whom I have known for a long time and consider them as my best friends.

Of course my family members, who always supported me and kept faith in me even during difficult times.

Last but not least, Mathilde, your continuous support and your affection helped me to stay on course for all this time, I shall not forget.

À mes grands-parents, Pierre et Renée.

Même la nuit la plus sombre prendra fin et le soleil se lèvera.
—Victor Hugo, *Les Misérables*

Contents

List of Figures

List of Algorithms

List of Acronyms

BVD: Bounded centroidal Voronoi Diagram
CAD: Computer Aided Design
CNES: French National Centre for Space Studies
ESA: European Space Agency
GMM: Geometrical Mathematical Model
MCRT: Monte Carlo Ray Tracing
QEM: Quadric Error Metric
TMM: Thermal Mathematical Model
VF: View Factor

Chapter 1

Introduction

Contents

1.1 Context and Motivations

Space thermal analysis is an essential and mandatory step for any spacecraft's longevity. Telecommunication satellites have a lifespan of roughly 15 to 17 years, while scientific satellites usually operate between 6 to 7 years (mainly due to the low orbit consequences), and eventually reach the graveyard orbit or the spacecraft cemetery in the Pacific Ocean. Optimizing the lifespan of spacecrafts is an essential duty in order to maximize the return on investment [SHN02]. The main threats arise from small debris, meteoroids, extreme temperatures and atmospheric drag. All of these must be considered during the design phase to maximize the spacecrafts longevity [GDL14]. The complete life cycle of a satellite is depicted Fig. 1.1, from its design to its disposal.

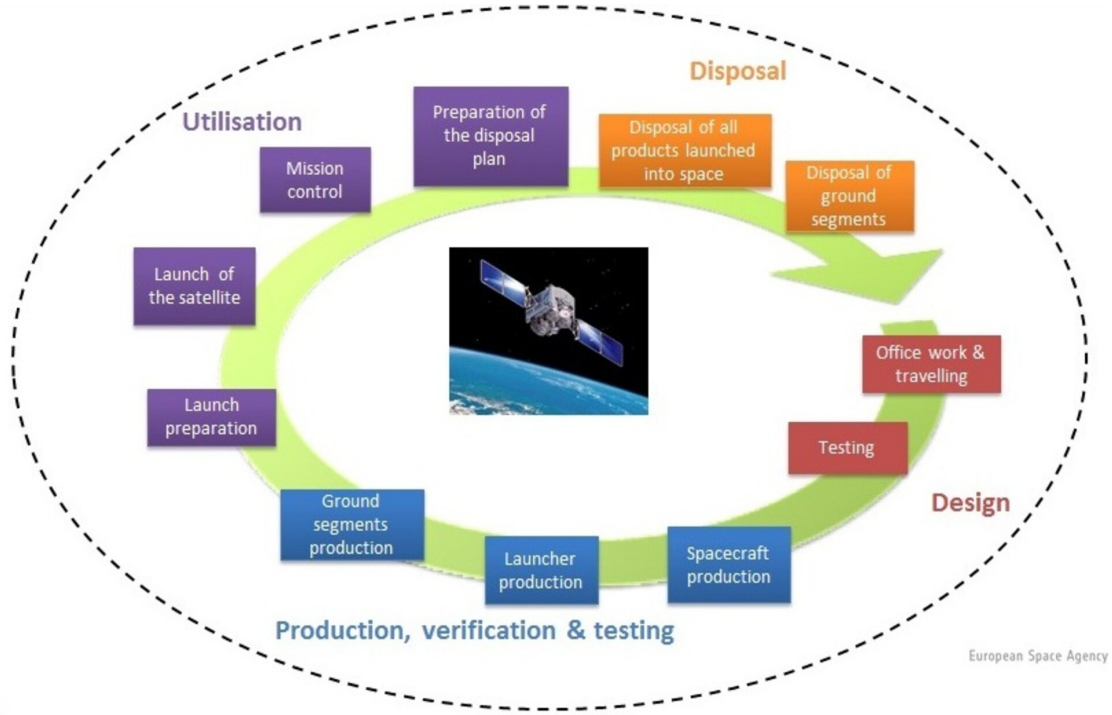


Figure 1.1: Life cycle of a satellite, from esa.int.

As a matter of fact, in order to ensure that the satellite is in good health on its orbit, temperatures of on-board elements are measured and compared to the simulations. On-board equipments operate within specific temperature ranges (see Fig. 1.2, and going beyond these ranges might deteriorate the device or even worse, destroy it, making it useless until the "death" of the spacecraft, hence justifying the importance of space thermal analysis. To regulate the temperature of sensitive components, several devices exist (see Fig. 1.3).

The mathematical model used for the thermal simulation defines the thermal exchanges between the components of the satellite, which is most often simplified by a

Component:	Temp. Margins [°C]:	Min. Limit [°C:]	Min./Max. Temp. [°C:]	Max. Limit [°C:]
Front Reflector ring 1	145.5 / 13.8	-50	95.5 / 386.2	400
FrontReflector ring 2	132.9 / 7.3	-50	82.9 / 392.7	400
Front Reflector ring 3	92.9 / 60.5	-50	42.9 / 339.5	400
Front Reflector ring 4	65.9 / 84.2	-50	15.9 / 315.8	400
Rear Reflector ring 1	145.5 / 13.8	-50	95.5 / 386.2	400
Rear Reflector ring 2	132.9 / 7.3	-50	82.9 / 392.7	400
Rear Reflector ring 3	92.9 / 60.6	-50	42.9 / 339.4	400
Rear Reflector ring 4	65.9 / 84.3	-50	15.9 / 315.7	400
Front Sub Reflector ring	131.7 / 17	-50	81.7 / 383	400
Rear Sub Reflector ring	131.7 / 17	-50	81.7 / 383	400
Cone ring 1	33.3 / 130	-50	-16.7 / 270	400
Cone ring 2	74.1 / 173.1	-50	24.1 / 226.9	400
Cone ring 3	105.8 / 219.5	-50	55.8 / 180.5	400
Cylinder ring 1	133.3 / 90.9	-50	83.3 / 309.1	400
Cylinder ring 2	112.4 / 142	-50	62.4 / 258	400
Cylinder ring 3	80.6 / 178.9	-50	30.6 / 221.1	400

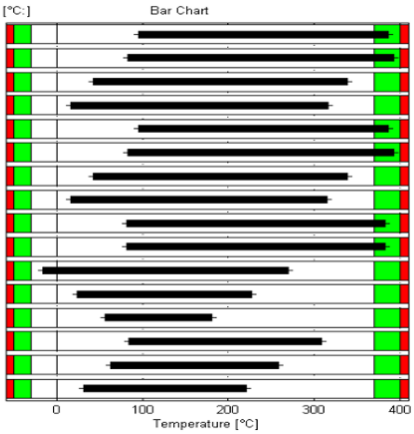
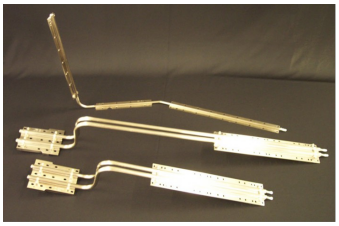
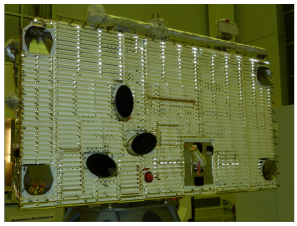


Figure 1.2: Temperature margins for various equipments on board of a satellite.





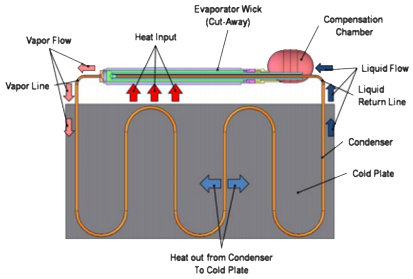


Figure 1.3: Cooling instruments on-board of a satellite. Left: heat pipe (heat-transfer internal devicethat combines the principles of boththermal conductivityandphase transitionto effectively transfer heat between two solidinterfaces). Middle: radiator (external device directly connected to space by radiation). Right: loop heat pipe (closed loop internal device mechanically pumped or operating by capillarity).

decomposition into atomic elements referred to as thermal nodes. It is common to use from 2,000 up to 200,000 nodes for a satellite. The exchanges between these nodes include a conductive component, and a radiative component. The conductive component is determined by the conductive links between nodes. The radiative component, on the other hand, is more complex since it requires the calculation of view factors between all pairs of nodes having mutual visibility. The quadratic complexity of the view factors induces a very high memory consumption and computation times, incompatible with a real-time simulation. Real time is currently only possible by reducing the complexity from 200,000 to a few thousand nodes.

For space applications, a specific radiation-tolerant processor is utilized on-board of the satellite to favor real-time simulation [VPP+06]. The thermal simulator must be able to respond to events coming from the flight simulator. Regarding temperatures, the information is sent as telemetry data at a refresh rate of 32 seconds. Then, to be real-time compliant, the complete thermal simulation has to be performed within this

32 seconds lap time. For instance, if computation times devoted to thermal simulation are greater than 32 seconds, a lag appears between the simulated and the real behavior of the satellite. This phenomenon is in particular noticeable at the end of the eclipse phase (where the satellite goes from being in the shadow to being directly exposed to solar radiation), where external temperatures of the satellite are expected to increase but keep decreasing instead due to the lag induced by computation times. To avoid this worst case scenario, the thermal simulator is configured to be multiprocess, with one process dedicated to temperature calculations and one process dedicated to radiative components computations (including view factors).

The growing needs for real-time simulation and digital twins [SCD20, PLK+21], justified by better control of the satellite to react effectively to unforeseen events, require an additional step of simplification of the radiative model, called "model reduction" in the literature. There are three families of model reduction methods. The first is to manually generate a faceted geometric abstraction through numerical expertise and interactive geometric modeling tools. This is a trial-and-error process, without direct and real time feedback from the digital simulation. The second consists in simplifying the geometry of the CAD model of the satellite, by automatic methods for mesh generation and simplification, most often utilizing planar facets. The third is to generate a radiative coupling matrix modeling the view factors, and to simplify this matrix with tools originating from linear algebra.

Regarding the first manual method resorting to numerical expertise, its main advantage lies into the feedback from the thermal engineer, resulting in reliable models. However, such a manual method is very time consuming in engineering time to simplify the entire model. In addition, it requires a strong expertise to identify the groupings of facets and faces risks related to uncertainties in the choice of the weights used.

The advantage of the second method is that its calculation is fast and automatic, but it is difficult to estimate the numerical approximation error which can only be measured after reduction and simulation.

The third method is very interesting from a geometric point of view because it preserves the shapes, is fast and straightforward. However, reduction percentages are still too low (50% reduction), far from the target facet elements budget of a few thousands.

Among the three methods discussed above, only the first, proceeding by manual approximation of the geometry by the expert, approaches the expected result. This approach is relevant by respecting the facet budget and the expert's feedback.

1.2 Problem Statement

In the following we first describe the considered thermal simulation applied to a satellite in space, then focus on the radiative computations part coupled to modern industrial problems, motivating the research interests of such a topic.

1.2.1 Heat Transfer

Heat transfer can proceed by conduction, convection and radiation. Thermal conduction is induced by a temperature gradient between two entities physically in contact. Fourier's law formulates the heat flux density $\mathbf{q}$ as the product of thermal conductivity k ($W.m^{-1}.K^{-1}$)) by the temperature gradient $-\nabla\mathbf{T}$ ($K.m^{-1}$), which can be written as:

$$\mathbf{q} = -k\nabla\mathbf{T} \quad (W.m^{-2}). \tag{1.1}$$

The rate of conductive heat transfer can be expressed as:

$$\dot{Q}_{1\to2} = -kA\frac{\mathbf{T}_2 - \mathbf{T}_1}{d} \quad (W), \tag{1.2}$$

where A denotes the cross-sectional surface area (m^2) and d (m) denotes the distance between the ends (such as material thickness).

Radiative heat transfer between two surfaces (noted 1 and 2) is the radiation leaving the first surface for the other minus the one arriving from the second surface. The Stefan-Boltzmann law yields:

$$\dot{Q}_{1\to2} = \sigma A_1 F_{1\to2}(\mathbf{T}_1^4 - \mathbf{T}_2^4) \quad (W), \tag{1.3}$$

where σ denotes the Stefan-Boltzmann constant ($W.m^{-2}.K^{-4}$), A_1 denotes the area of the first surface (m^2) and $F_{1\to2}$ denotes the view factor from 1 to 2 (unitless).

Heat transfer by convection in presence of fluid or gas is given by:

$$\dot{Q} = hA(\mathbf{T} - \mathbf{T}_f)^\alpha \quad (W), \tag{1.4}$$

where h denotes the heat transfer coefficient ($W.m^{-2}.K^{-1}$), $\mathbf{T}$ denotes the surface temperature of the object, T_f denotes the fluid (or gas) temperature and α denotes an exponent for scaling. In space, convection effects can be ignored since there is no gas nor fluid outside of the satellite.

The common approach to solve the above equations relies on tessellating the model via finite element methods (see Fig. 1.4), but these are compute-intensive in particular for complex models with curved parts. An alternative and faster method consists in subdividing the model into volume entities called thermal nodes.

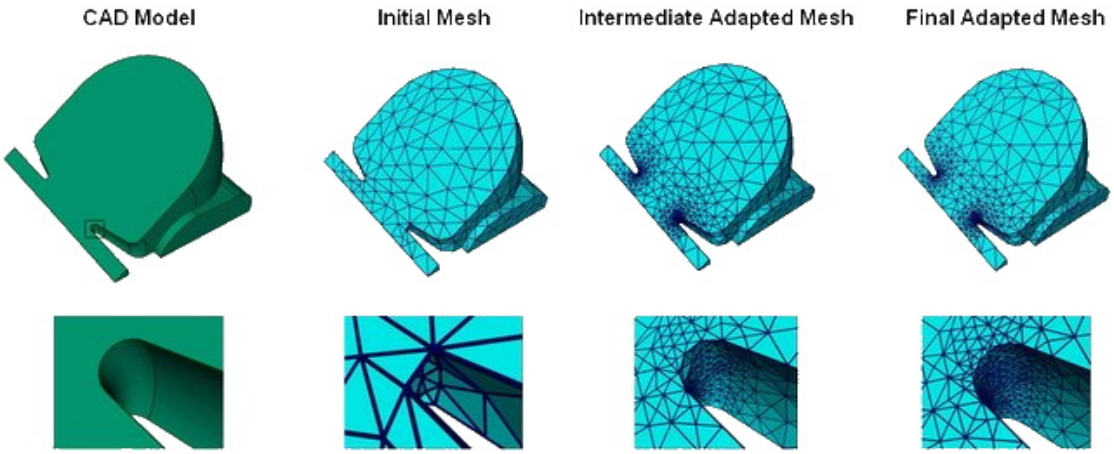


Figure 1.4: Tessellation of a CAD model for finite-element analysis, taken from Videla et al [VA17].

Using previous equations, temperatures can be computed at steady state (when variables are constant over time) or for transient state (when variables evolve until the system reaches a steady state). For space thermal analysis, it involves conductive and radiative exchanges between nodes, as well as powers received by the faces of the model, which can be internal or external (Earth, albedo and solar fluxes). This yields the following equation [Nas11]:

$$mC\frac{dT}{dt} = P(t) + \boldsymbol{C_{ond}}T + \sigma \boldsymbol{R_{ad}}T^4, \tag{1.5}$$

where denotes m the mass of the system, C denotes the heat capacity, P denotes the powers, $\boldsymbol{C_{ond}}$ denotes the matrix of conductive couplings and $\boldsymbol{R_{ad}}$ denotes the matrix of radiative couplings.

In steady state, it is assumed that $\frac{dT}{dt} = 0$ and $P(t)$ is constant. The resulting equation can thus be solved via Newton's iterative method. In transient state, the solution is found with implicit or explicit iterative methods for numerically solving a system of n nonlinear differential equations (such as Runge-Kutta) [KU77, VM21].

1.2.2 Space Thermal Analysis

The powers dissipated in the various pieces of equipment on a satellite are relatively low (around a few watts) while the experiments carried out in the laboratory dissipate much more energy and do not pose any problem of abnormal heating. However, the difficulty for on-board experiments resides in the environmental conditions prevailing in space. It is advisable to take into account from the start of the project the specific constraints posed by thermal control and thus avoid discovering problems during an advanced development phase which would entail modifications and delays (often significant and costly). In orbit, the spacecraft is subjected to a highly and rapidly variable environment over

time [BFFP70]. Hence, the three types of fluxes received vary according to the Earth-Satellite distance and the relative Earth-Satellite-Sun position, see Fig. 1.5. The fluxes also vary depending on the attitude of the satellite.

The solar flux varies in intensity depending on the season (variation of the Satellite-Sun distance). It cancels out when the satellite passes through the Earth's shadow cone (eclipse). The amount of solar energy absorbed varies depending on the attitude of the vehicle and the coating of the satellite.

The albedo flux (fraction of the solar flux reflected by the Earth) varies according to the rate of reflection from the Earth's surface and the Sun-Earth-Satellite angle. The absorbed fraction depends, as before, on the attitude and coating of the satellite.

The terrestrial flux varies according to the Earth-Satellite distance (the absorbed fraction always depends on the same parameters). In addition, the satellites are placed at altitudes such that the residual atmospheric pressure is very low so that there is no possibility of convection around or in the satellite. Other factors have an indirect impact on the thermal behavior of a spacecraft. On the one hand, the degassing of non-metallic products can condense on a cooler part of the vehicle and modify the absorption rates of external flows. On the other hand, the radiative and electromagnetic environment can influence the thermo-optical characteristics.

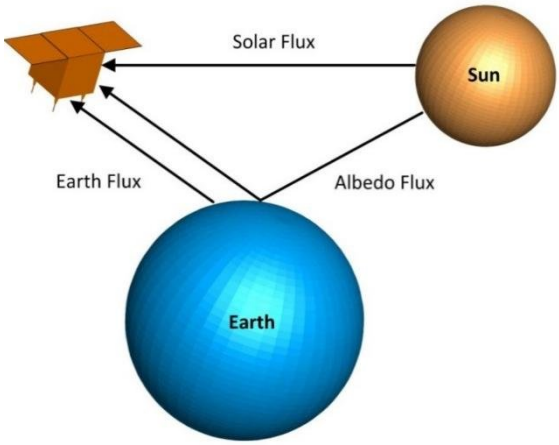


Figure 1.5: Schematic illustration of the types of external fluxes acting on a satellite. Taken from Uygur et al [Uyg18].

Heat dissipated in a satellite is radiated away from its outer surfaces to space. The external environment brings solar, terrestrial and albedo radiation to the satellite (which can be reduced using MLI, see Fig. 1.6). The calculation of these various external exchanges makes it possible to determine the average temperature levels of the satellite. The temperature differences inside a satellite depend on the internal heat exchanges. These differences take place by radiation between the different surfaces of the spacecraft, by conduction through the structure, heat pipes or fluid loop devices. As a result of the weight reduction of the structures to reduce the masses, the exchanges by conduc-

tion are limited and have only a local importance. In the first generation of satellites, where dissipation was weakly concentrated, internal heat transfer took place mainly by radiation. In the current generation, exchanges by conduction have an increased importance, due to the increase in the concentration of thermal loads. When this concentration is very high, heat pipes are utilized to distribute dissipation [Shu15].

Figure 1.6: Satellite before and after adding Multi-Layer Insulation (MLI) to protect it from extreme radiation. Taken from Fockersperger et al [FLK+10].

1.2.3 A Need for Mathematical Modeling

A thermal model is a representation of a physical system in which heat exchanges are involved. This representation can be conducted either by a mathematical model, or by an analog model, the latter resulting from the establishment of formal analogies between different physical phenomena. It is thus usual to establish an analog electrical model of certain thermal systems, see Fig. 1.7. It is assumed that a thermal model represents a mathematical representation of heat exchange phenomena that occur within a physical system. Setting up the model consists in choosing the state variables used to describe the exchange laws, then in writing the equations connecting these variables as well as those which characterize the environment of this system. Developed as part of nodal analysis, these equations are organized into systems, either algebraic in the case of stationary regimes, or differentials in the case of transitional regimes.

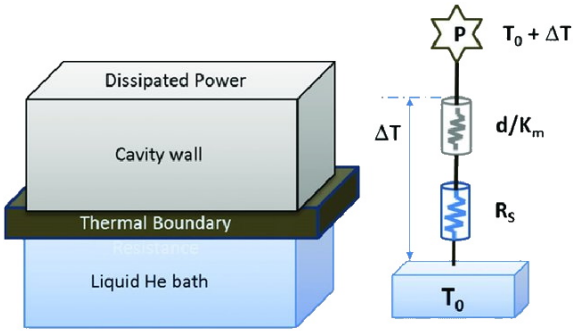


Figure 1.7: One-dimensional model electrical analog for the heat transfer in a superconducting cavity from Rossi et al [RSV14].

The objectives of thermal modeling are to discover, explain, and even predict certain aspects of the behavior of the real system studied by mainly calculating temperatures, exchanged flows and energy balances. The interest for thermal modeling lies for example during the design of a pre-project, where the realization of an experimental model could suffer from delivery times, manufacturing, or even present machining or assembly defects. The model is used to guide the choice of sizing and the nature of the materials, thus avoiding the accumulation of costly material errors. Modeling also plays a major role in the optimization phase of the final project. For reasons of cost, interpretation of experiments and safety, modeling seems to take precedence over the experimental field, but it would be a serious mistake to suppress any call for experience, using simulation only. In practice, models and experiments are compared for the purpose of validating the model by comparing the predictions of the initial model and the information provided by the experiment and possibly correcting certain parameters of the model. The model is then refined. Subsequently, the simulation should allow rapid and inexpensive predictions, consisting for example in providing, using the validated model, the behavior of the system in multiple configurations, which, for reasons of cost or time, cannot be treated experimentally. Modeling is therefore not an end in itself but rather a means to establish a bridge between experimental observation and speculations of a more abstract nature.

A model can only partially represent the real system: Its description, correct for a given environment, can deteriorate when the latter evolves from the stationary to the transient thermal regime. It may also provide only a limited number of elements of answers. An over simplified model moves away from reality, but conversely, an excess of realism sometimes makes it cumbersome, as difficult to grasp as the system it translates. While it is true to say that a model is neither true nor false, the best model is the one leading to the most accurate predictions obtained from the associated experience. Without denying the validity of mathematical analysis, however, it is common to observe its inability to provide solutions that are easy to exploit. It suffices, for example, to deviate from common Cartesian, cylindrical or spherical geometries. In any case, the practitioners do not seek for a temperature field in the form of a continuous function, but

rather for values at a specified number of discrete points. The finite differences [SS05] or finite elements methods certainly have a role to play for thermal analysis [RG10], but they do not seem to currently offer a range of possibilities as rich as that of nodal analysis. To summarize the first aforementioned methods: The finite elements are well suited to the problems dealing with non-standard geometry but the nodal method takes over when the system to be studied becomes complex and faster time responses are required. As for the finite differences method, while it is similar to the nodal method in specific configurations such as conduction in Cartesian geometry, it overshadows the latter for all that concerns the fine digitization of convective phenomena. The nodal method is thus, among the aforementioned discretization methods, the only one leading to general, well-established application software, and therefore suitable for the daily work of thermal engineers.

1.2.4 Thermal Modeling Principles

Before constructing the thermal model, it is mandatory to construct the geometrical mathematical model (GMM), which consists in the design of the geometry, the definition of the thermo-optical properties of the faces as well as the attitude and orbital parameters of the satellite (see Fig. 1.8) [Gil02, AK18]. The thermal expert then proceeds to nodal breakdown, leading to the development of the thermal mathematical model (TMM).

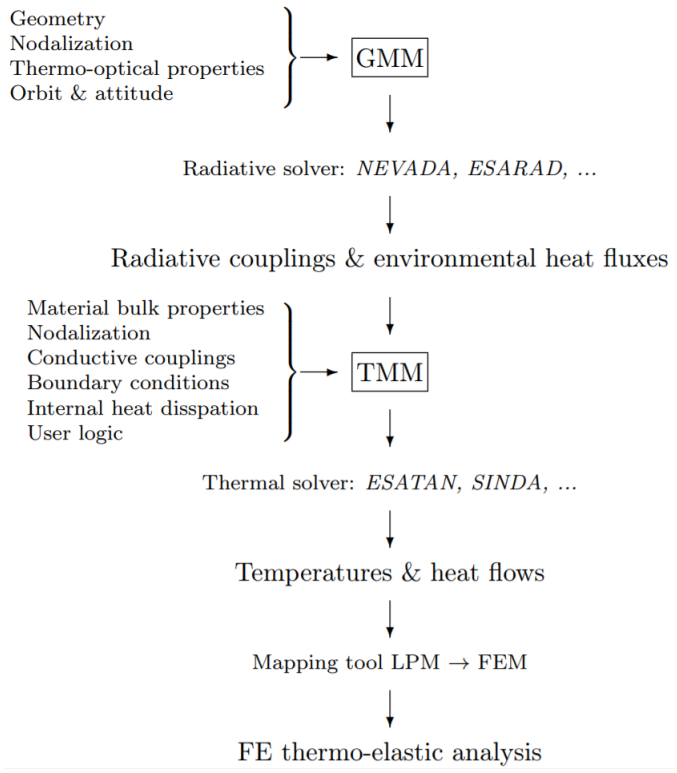


Figure 1.8: Thermal model construction steps as described in Jacques's Ph.D. thesis [Jac16].

The objective of thermal modeling is to determine the set of radiative and conductive couplings between all the elements of the satellite. A spatial model consists of a radiative part and a conductive part. The two are assembled in the overall thermal model, see Fig. 1.9. For modeling purposes, the satellite is broken down into a certain number of isothermal nodes which can be made of equipments dissipating energy or having significant dimensions, as well as structural elements, possibly with one or more items of equipment attached to it (the latter being small and not dissipating thermal power).

For radiative modeling, each face is abstracted by a set of canonical geometric shapes (rectangles, triangles, cylinders, spheres) having a given heat capacity and being the site of a given heat dissipation, which can be a function of time and on which the assumption of uniform radiative flux is made. Each face is then characterized with respect to neighboring faces using geometric view factors, conductive coupling coefficients, physical and thermo-optical properties and an initial temperature. For conductive modeling, each face is assigned a shape which enables a straightforward calculation of the conductance. A thermal node is a grouping of thermal faces that respects the isothermal assumption. Several tools exist in order to help thermal engineers building and testing thermal models, such as ESATAN (ITP Aero), SYSTEMA (Airbus) and e-Therm (Thales Alenia Space), among others [RHS05].

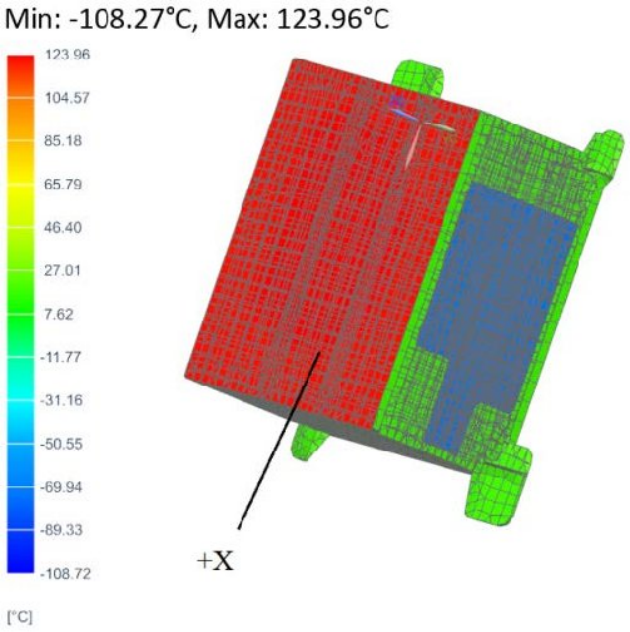
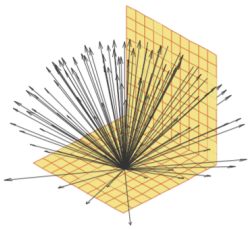


Figure 1.9: Thermal model of the "AIST" small satellite. Taken from Ivanushkin et al [ITS+19].

1.2.5 Radiative Computations Complexity

Radiosity problems and radiative thermal simulations [HS67] are governed by a geometric quantity referred to as the view factor, also referenced as shape factors, form factors, configuration factors or angle factors (see equation 1.3). Given two surfaces S and T, the view factor F_{S-T} measures the proportion of the radiation which leaves S and strikes T. More specifically, the view factor quantity measures the fraction of the heat flux radiated by an isothermal surface with isotropic emission received by another surface in a non-participatory environment (no emission, absorption or diffusion). This value depends only on the geometry of the environment. Such a proportion is lower in the presence of obstacles that reduce radiation or incur shadows on T. Considering a uniform hemispherical light transport or heat transfer model without obstacles, the exact value is determined by the following formula (Fig. 1.10):

$$F_{1\to 2} = \frac{1}{A_1}\int_{A_1}\int_{A_2} \frac{\cos\theta_1 \cos\theta_2}{\pi s^2}\,\mathrm{d}A_2\,\mathrm{d}A_1,$$



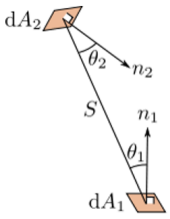


Figure 1.10: View factors formulation without obstacles. Left: integral formula. Middle: uniform hemispherical model. Right: associated configuration where two surface elements are mutually visible.

A_1 denotes the area of the issuing face 1, s^2 denotes the squared distance from 1 to 2, and θ_1 denotes the angle between the normal vector of 1 and the segment between 1 and 2. Quantity $F_{1\to 2}$, a real number between 0 and 1, indicates the percentage of the radiations leaving the issuing face 1 and striking the target face 2. It then depends on three components, (1) the distance, (2) the contours (and areas) and (3) the orientations between the two considered geometric elements.

For differential areas (obtained by taking the limit of a small and flat surface), the view factor can be expressed as:

$$dF_{1\to 2} = \frac{cos\theta_1 cos\theta_2}{\pi s^2} dA_2. \tag{1.6}$$

View factor properties:

$$\underline{Summation} : \sum_{j=1}^{n} F_{S_i\to S_j} = 1 \tag{1.7}$$

$$\underline{Superposition} : F_{1\to(2,3)} = F_{1\to 2} + F_{1\to 3} \tag{1.8}$$

$$\underline{Reciprocity} : A_1 F_{1\to 2} = A_2 F_{2\to 1} \tag{1.9}$$

$$\underline{Planar\ or\ convex\ surface} : F_{1\to 1} = 0 \tag{1.10}$$

$$\underline{Concave\ surface} : F_{1\to 1} \neq 0 \tag{1.11}$$

Considering a geometric model with n surface elements, and since each facet can potentially see the $n-1$ others (or even itself when concave, see 1.11), view factor computations come at a quadratic cost. Calculating the double integrands for every pair of faces is a tedious task. Furthermore, in presence of visibility obstacles between the two considered surface elements, direct calculations cannot be proceeded and quadrature methods must be utilized to approach the exact value, complicating the task even

more. Additionally, spacecraft geometrical models tend to be more and more complex, notably due to topological optimization and $3D$ printing for structure and thermal analysis. Common equipments which were represented by a few hundreds of faces now require hundred of thousands of faces (see Fig. 2.8), introducing additional substantial complexity to the view factor computations.

1.3 Goals and Challenges

Despite recent progress in the analysis of correlation between thermal model and reference data, the main limitations of the state of the art arise from the insufficient or indirect link between model reduction methods and thermal simulation. Mesh simplification methods are most often guided by purely geometric error metrics and do not guarantee fidelity with the numerical simulation.

The main scientific issue is explained by the difficulty in quantifying the relationships between geometric approximation operations (with a dual combinatorial and continuous component for the simplification of meshes) and their impact on simulation. Intuitively, it is unknown how to formulate a continuous derivation between approximation and simulation, on the one hand because of the discontinuous combinatorial component, and on the other hand because for complexity reasons, it is not feasible to run a complete simulation for each atomic operation of reduction of the model. In addition, the view factors involving a discretization of directions by ray tracing are non-derivable in the presence of obstructions.

1.3.1 Reference Calculation Case

The first objective is to design and develop a reference calculation case by relying on a new hierarchical geometric data structure for the efficient computation of view factors, in a progressive and interruptible way in order to best trade desired accuracy and spent computation times. Additionally, a prediction step is requested at the end of calculations to approach the exact view factor value even more. For the sake of accuracy, exact predicates and closed form formulae are favored. The challenge is to be able to handle the initial complexity of a model with hundreds of thousands or even millions of facets, where each occlusion part is itself represented by a facet with its own view factor.

1.3.2 Geometric Reduction Accurate to Simulation

The second objective consists in designing a geometric approximation method where each atomic simplification or optimization operation is guided by the preservation of the numerical simulation of the reference model, rather than the preservation of the geometry only. The method is applied to radiative thermal simulation and must be sufficiently modular to impact other potential numerical simulations highly related to the geometry of the considered system (such as e.g., conductive thermal simulation or structure deformation).

1.3.3 Outline

This thesis manuscript is organized as follows:

Chapter 2 covers the related work focusing on view factor computations, with and without visibility obstructions, as well as mathematical and geometrical model reduction techniques.

Chapter 3 describes our algorithmic contributions for computing progressive and interruptible view factors.

Chapter 4 describes our algorithmic contributions to geometrical model reduction driven by numerical simulation.

Chapter 5 provides concluding remarks and discusses potential perspectives for future work.

Chapter 2

State of the Art and Contributions

Contents

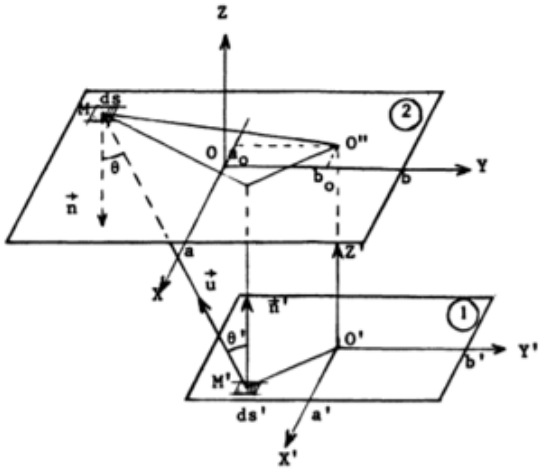
2.1 Computing View Factors

This section aims to summarize the major methods of view factor computations, from calculating the quadruple integrals in full visibility scenarios to performing quadrature in presence of visibility obstacles.

2.1.1 View Factors without Visibility Obstacles

While view factors can be computed in closed form for certain classes of canonical surfaces without obstacles, general free-form surfaces require calculating complex integrals with quadrature schemes. Detecting and assimilating obstacles into these integrals adds further numerical hurdles, and time-varying objects such as articulated parts of satellites require new computations for each configuration.

Closed form expressions for view factors are no recent problematics, beginning during the 18^{th} century with the work of Lambert. Until recently, the quadruple integrals were computed by hand for the most common configurations in order to have an exact and rapid evaluation, easy to implement for small computers. For instance for parallel rectangular planar faces, Ritoux [Rit82] in 1982 proposed the following demonstration:



Proof. **M'M** = **r**, **OO"**=$\rho_0(a_0, b_0, 0)$
O'M'=$\rho'_0(X', Y', 0)$, $r^2 = (X - X' - a_0)^2 + (Y - Y' - b_0)^2 + d^2$
OM=$\rho(X, Y, 0)$, $cos(\theta) = cos(\theta') = \frac{d}{r}$, **O'O"**=d
With:

$$-a' < X' < +a', \quad -a < X < +a$$
$$-b' < Y' < +b', \quad -b < Y < +b$$
$$F_{1\to 2} = \frac{1}{\pi s'} \int_s \int_{s'} \frac{cos\theta . \cos\theta' \mathrm{d}s\mathrm{d}s'}{r^2} = \frac{d^2}{\pi s'} \int_s \int_{s'} \frac{\mathrm{d}s\mathrm{d}s'}{r^4}$$

For the sake of readability, we introduce the reduced dimensions of the surfaces noted:

$$X = \frac{a}{d}, \quad Y = \frac{b}{d}, \quad X_0 = \frac{a_0}{d}$$
$$X' = \frac{a'}{d}, \quad Y' = \frac{b'}{d}, \quad Y_0 = \frac{b_0}{d}$$

The full calculation of the four integrals finally gives:

$$\begin{aligned} F_{1\to 2} = \frac{1}{8\pi X'Y'}(&F(X - X_0 + X', Y - Y_0 + Y') - F(X - X_0 + X', Y - Y_0 - Y') \\ &+F(X - X_0 + X', Y + Y_0 - Y') - F(X - X_0 + X', Y + Y_0 - Y') \\ &-F(X - X_0 - X', Y - Y_0 + Y') + F(X - X_0 - X', Y - Y_0 - Y') \\ &-F(X - X_0 - X', Y + Y_0 - Y') + F(X - X_0 - X', Y + Y_0 - Y') \\ &+F(X + X_0 + X', Y - Y_0 + Y') - F(X + X_0 + X', Y - Y_0 - Y') \\ &+F(X + X_0 + X', Y + Y_0 - Y') - F(X + X_0 + X', Y + Y_0 - Y') \\ &-F(X + X_0 - X', Y - Y_0 + Y') + F(X + X_0 - X', Y - Y_0 - Y') \\ &-F(X + X_0 - X', Y + Y_0 - Y') + F(X + X_0 - X', Y + Y_0 - Y')) \end{aligned}$$

With:

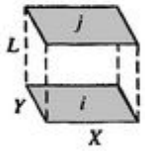
$$F(u,v) = u\sqrt{v^2+1}Arctg(\frac{u}{\sqrt{v^2+1}}) + v\sqrt{u^2+1}Arctg(\frac{v}{\sqrt{u^2+1}}) - \frac{1}{2}ln(u^2+v^2+1)$$

□

This complex but easy to program formula is of real importance since it contains, in fact, all cases to calculate the view factors between any parallel rectangular or square faces (centered or not). Thermal engineer handbooks contain demonstrations for multiple configurations, see Fig. 2.1.

Geometry | **Relation**

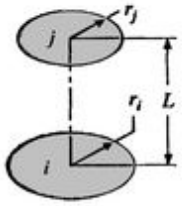
Aligned Parallel Rectangles (Figure 13.4)



$$\overline{X} = X/L,\ \overline{Y} = Y/L$$

$$F_{ij} = \frac{2}{\pi \overline{X}\,\overline{Y}}\left\{\ln\left[\frac{(1+\overline{X}^2)(1+\overline{Y}^2)}{1+\overline{X}^2+\overline{Y}^2}\right]^{1/2} + \overline{X}(1+\overline{Y}^2)^{1/2}\tan^{-1}\frac{\overline{X}}{(1+\overline{Y}^2)^{1/2}} + \overline{Y}(1+\overline{X}^2)^{1/2}\tan^{-1}\frac{\overline{Y}}{(1+\overline{X}^2)^{1/2}} - \overline{X}\tan^{-1}\overline{X} - \overline{Y}\tan^{-1}\overline{Y}\right\}$$

Coaxial Parallel Disks (Figure 13.5)

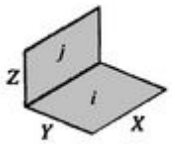


$$R_i = r_i/L,\ R_j = r_j/L$$

$$S = 1 + \frac{1+R_j^2}{R_i^2}$$

$$F_{ij} = \frac{1}{2}\left\{S - \left[S^2 - 4(r_j/r_i)^2\right]^{1/2}\right\}$$

Perpendicular Rectangles with a Common Edge (Figure 13.6)



$$H = Z/X,\ W = Y/X$$

$$F_{ij} = \frac{1}{\pi W}\left(W\tan^{-1}\frac{1}{W} + H\tan^{-1}\frac{1}{H} - (H^2+W^2)^{1/2}\tan^{-1}\frac{1}{(H^2+W^2)^{1/2}} + \frac{1}{4}\ln\left\{\frac{(1+W^2)(1+H^2)}{1+W^2+H^2}\left[\frac{W^2(1+W^2+H^2)}{(1+W^2)(W^2+H^2)}\right]^{W^2} \times \left[\frac{H^2(1+H^2+W^2)}{(1+H^2)(H^2+W^2)}\right]^{H^2}\right\}\right)$$

Figure 2.1: View Factors for common Three-Dimensional Geometries, from Fundamentals of Heat and Mass Transfer [BIDL11]. These configurations are rather common in aerospace and nuclear industries.

These formulae provide substantial accelerations and a better accuracy compared to approaches using nomograms [BP61] or catalogs [HM11] for view factor evaluations, although such closed forms must be computed for each configuration and complex objects are not always decomposable into canonical geometric primitives. A geometric analog, called the Nusselt Analog [Nus28], is useful for computing differential view factors, see Fig. 2.2. The idea being to construct a unit half sphere centered on the considered differential area and to approximate the view factor by projections. Notably, the projected

element's area on the unit sphere is equal to the solid angle covered by the element, amounting to the quantity $\frac{cos\theta_j}{s^2}$. The $cos\theta_i$ component derives from the projection onto the base, and lastly π being the area of the unit circle, it naturally appears in the denominator. A similar version developed by Cohen et al, named hemi-cube, relied on projections onto a half cube instead of a half sphere [CGD85].

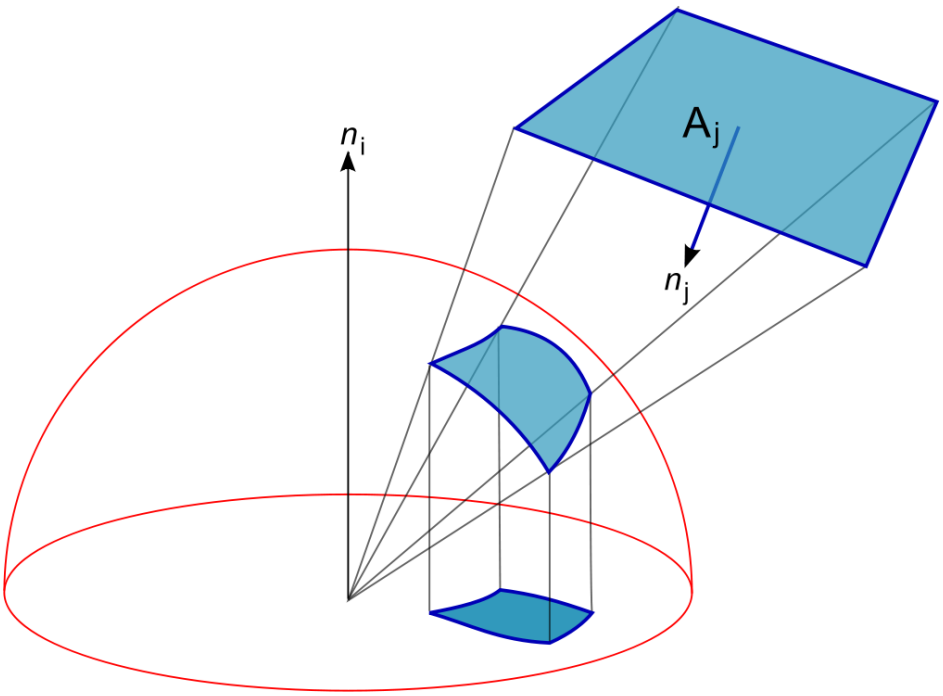


Figure 2.2: Nusselt analog. The view factor between a differential element dA_i and the element A_j can be obtained by projecting the element A_j onto the surface of a unit hemisphere, and then projecting that in turn onto a unit circle around the point of interest in the plane of A_i. The view factor is then equal to the differential area dA_i times the proportion of the unit circle covered by this projection. Credits to Jheald.

Baum et al. [BRW89] developed a technique to correctly compute view factors based on analytically-determined values for multiple configurations to prevent numerical errors of the hemi-cube algorithm. In 1993, Schröder and Hanrahan [SH93] proposed a closed form solution of the view factors between any pair of convex polygons. The formula was established by applying Stokes' theorem [Her09] to 1.10, giving an expression of the view factor with two contour integrals, see Fig. 2.3. For polygons the contour integrals become a sum of double line integrals over all pairwise combinations of edges. This

result is extremely valuable for meshed models.

$$\int_0^{c_2}\int_0^{c_0} \ln f(s,t)\,ds\,dt = \left\{\left[(s+\frac{c_3}{2})\mathrm{G}(f(.,t))(s)+\frac{c_1}{2}\mathrm{H}(f(.,t))(s)\right]\Big|_{s=0}^{s=c_0}\right\}\Bigg|_{t=c_2} - 2c_0c_2 + c_{14}c_{15}\left\{\left[\pi(2k(s)+1)\mathrm{M}(t) - i\left(\mathrm{L}(-c_{17}(s))(t)+\mathrm{L}(-c_{18}(s))(t)-\mathrm{L}(c_{17}(s))(t)-\mathrm{L}(c_{18}(s))(t)\right)\right]\Big|_{s=0}^{s=c_0}\right\}\Bigg|_{t=\sqrt{\frac{c_{13}}{c_{13}}}}^{t=\sqrt{\frac{c_{13}+c_2}{c_{13}+c_2}}}$$

$$\begin{aligned}
\mathrm{L}(b)(y) &:= \int^y t^2(1-t^2)^{-3}\ln(b+t)\,dt &&= \frac{1}{16}\left[\frac{-b\ln(y-1)}{(b+1)^2}-\frac{b\ln(1+y)}{(b-1)^2}+\left(\frac{2(b+y)(1+by)[(b-y)^2+(by-1)^2]}{(b^2-1)^2(y^2-1)^2}+\ln\frac{(1-y)(1-b)}{(1+y)(1+b)}\right)\ln(b+y)\right.\\
&&&\left.+\frac{2(b-y)}{(b^2-1)(y^2-1)}+\mathrm{Li}_2\left(\frac{1-y}{1+b}\right)-\mathrm{Li}_2\left(\frac{1+y}{1-b}\right)\right]\\
\mathrm{M}(y) &:= \int^y t^2(1-t^2)^{-3}\,dt &&= \frac{1}{16}\left[4y(y^2-1)^{-2}+2y(y^2-1)^{-1}+\ln\frac{y-1}{y+1}\right]\\
\mathrm{G}(q)(y) &:= \int^y \ln q(t)\,dt &&= \frac{q'(y)}{2a}\ln q(y)-2y+\frac{d}{a}\tan^{-1}\frac{q'(y)}{d}\\
\mathrm{H}(q)(y) &:= \int^y t\ln q(t)\,dt &&= \left(\frac{y^2}{2}+\frac{c}{2a}-\frac{b^2}{4a^2}\right)\ln q(y)-\frac{y(ay-b)}{2a}-\frac{bd}{2a^2}\tan^{-1}\frac{q'(y)}{d}
\end{aligned}$$

Table 1: Four auxiliary integrals needed in the solution. Notice that $\mathrm{L}(b)(y)$ uses the dilogarithm [1], $\mathrm{Li}_2(z)=\sum_1^\infty \frac{z^k}{k^2}$, $\frac{d}{dz}\mathrm{Li}_2(z)=-\frac{\ln(1-z)}{z}$. In G and H the argument q is an arbitrary quadratic polynomial $q(t)=at^2+bt+c$ and $d=\sqrt{4ac-b^2}$.

$$\begin{aligned}
c_0 &= \|E_j\| & c_{13} &= \frac{c_{11}-\sqrt{c_{11}^2-4c_{10}c_{12}}}{2c_{10}}\\
c_1 &= -2\vec{d_i}\cdot\vec{d_j} & c_{14} &= \frac{\sqrt{c_{11}^2-4c_{10}c_{12}}}{c_{10}}\\
c_2 &= \|E_i\| & c_{15} &= \sqrt{c_{10}}c_{14}\\
c_3 &= -2\vec{d_j}\cdot(\vec{p_i}-\vec{p_j}) & c_{16}(s) &= c_1c_{13}-c_3-2s\\
c_4 &= 2\vec{d_i}\cdot(\vec{p_i}-\vec{p_j}) & c_{17}(s) &= \frac{-c_{15}+\sqrt{c_{15}^2-4|c_{16}(s)|^2}}{2ic_{16}(s)}\\
c_5 &= \|\vec{p_i}-\vec{p_j}\|^2 & c_{18}(s) &= \frac{-c_{15}-\sqrt{c_{15}^2-4|c_{16}(s)|^2}}{2ic_{16}(s)}\\
c_{10} &= 4-c_1^2\\
c_{11} &= 4c_4-2c_1c_3\\
c_{12} &= 4c_5-c_3^2
\end{aligned}$$

Table 2: All expressions for two edges E_{ij} with parameterization $\vec{x}_i(t)=\vec{p}_i+t\vec{d_i}$ and $\vec{x}_j(s)=\vec{p}_j+s\vec{d_j}$ ($\|\vec{d}_{i,j}\|=1$).

Figure 2.3: Schröder and Hanrahan view factor formula for the double integrand between any convex polygons [SH93].

Finite-element based procedures to obtain view factors estimations have also been developed, in particular between a non-uniform emitter field and the receiving surface [MI20].

2.1.2 View Factors with Visibility Obstacles

Unfortunately computing view factors in presence of visibility obstructions between the two considered geometric elements requires other techniques (Fig. 2.4 depicts the different visibility scenarios occurring in 2D and 3D). Closed form formulae alone do not lend to the exact and fast evaluation of *obstructed* view factors between complex objects with partial vision of each others. More general situations require both numerical approaches to approximate geometric view factors and detecting obstacles. Numerical approximation of integrals via adaptive quadratures is an art in itself, and detecting obstacles between two surface elements, commonly performed via ray-shooting, is already a compute-intensive operation for complex objects.

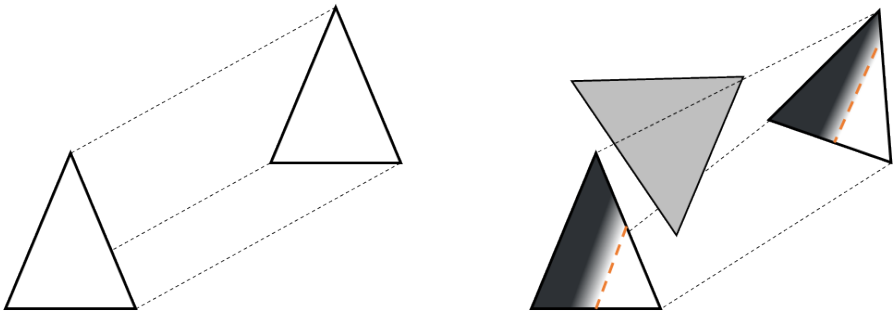


Figure 2.4: Different visibilities. Left: full visibility. Right: partial visibility (with umbra, penumbra and full visibility respectively).

The modus operandi in presence of visibility obstructions consists in applying a quadrature to approximate the exact value of the view factor. The most common implementation relies on ray shooting, also known as Monte Carlo Ray Tracing methods (MCRT) [VLB06, MS11, WXB12]. Monte Carlo techniques rely on repeated random sampling to obtain numerical results, these methods are powerful in particular when the use of other approaches is laborious. View factor can be approximated by the quantity $F_{1\to 2} = \frac{N_b}{N_{total}}$ with N_b the number of rays from face 1 to face 2 intersecting nothing except the two considered faces and N_{total} the total number of thrown rays in random directions and starting positions from face 1. Another MCRT technique samples with points both considered faces and evaluates the view factor integrand at N random quadruples $(p_{i1}, p_{i2}, p_{i3}, p_{i4})$ for $i = 1..N$, where coordinate pairs (e.g. p_{i1}, p_{i2}) refer to a point on one of the surfaces, and the other pair (p_{i3}, p_{i4}) to a point on the other surface. The view factor $F_{1\to 2}$ from surface A_1 to surface A_2 is then approximated by:

$$F_{1\to 2} = \frac{A_2}{N} \sum_{i=1}^{N} \frac{cos\theta_1 cos\theta_2}{\pi s^2} \tag{2.1}$$

where each argument in the sum is evaluated for each ray connecting face 1 and face 2. The main benefit from MCRT methods lies in giving view factors estimations between any geometric shapes (polygons, curve elements, geometric primitives), which comes at

a quadratic algorithm complexity depending on the number of rays thrown. The major drawback of MCRT algorithms being the need to shoot a significant number of rays in order to have a decent approximation of the real value of the view factor, see Fig. 2.5.

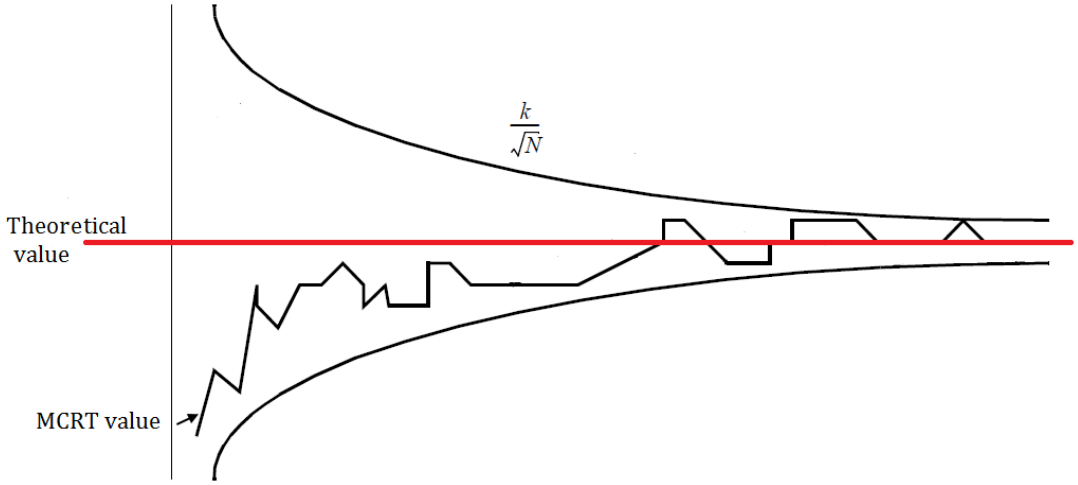


Figure 2.5: View factor estimation with Monte Carlo Ray Tracing. The evolution of the estimated value lies within a band whose width is inversely proportional to the square root of the number of rays fired, multiplied by a constant.

MCRT methods optimization is a research subject by itself. For instance Jacques et al. [JMK13] proposed a quasi-MCRT method with improved convergence, based on the notion of isocells for point sampling (see Fig. 2.6). The idea is to proceed a more uniform sampling by partitioning faces into sub-domains of similar areas and add a sampling point into each of them, resulting in better convergence rates compared to random sampling. Beckers et al. [BB16] developed a similar algorithm relying on stratified sampled Monte-Carlo method by defining cells between two latitudes and two longitudes.

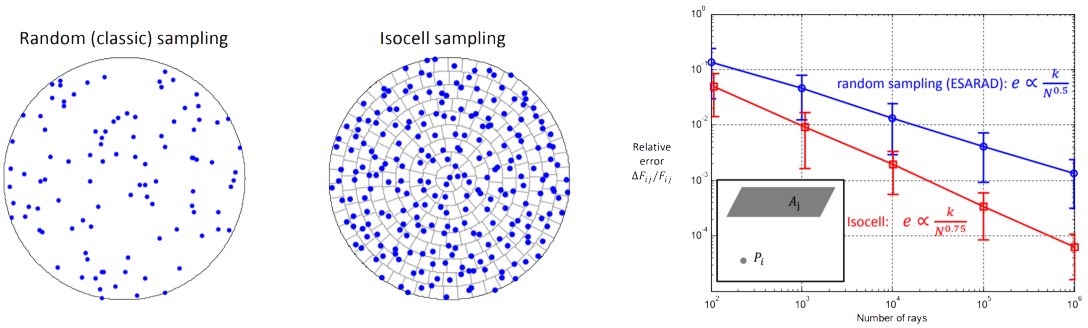


Figure 2.6: Jacques et al. isocell sampling [Jac16]. Left: Comparison between random and isocell samplings. Right: Convergence of the two methods.

Another quadrature approach consists in partitioning the considered faces into sub-polygons, named polygon-based quadrature. Compared to previous detailed point-based quadratures, polygon-based quadrature methods benefit from closed form expression when sub-polygons fully see each others. Also, the view factor estimation obtained through polygon-based quadratures only increases with the number of sub-polygons as a monotonic function, whereas the approximated value given by point-based quadrature varies widely (see Fig. 2.5). Hanrahan et al. proposed first a hierarchical data-structure for progressive radiosity [HSA91], using view factors as an estimator guiding the refinement process. Regarding radiative thermal simulation, Le Bohec suggested two other estimators, being the illumination and the exchanged net flux density [LB17]. Walton contributed both point-based and polygon-based adaptive quadratures and made them available through the View3D software [Wal02], notably detailing a method based on projecting the shadow of the obstruction onto the plane of the second surface. Different point-based and polygon-based quadratures are implemented and compared on multiple common scenarios, see Fig. 2.7.

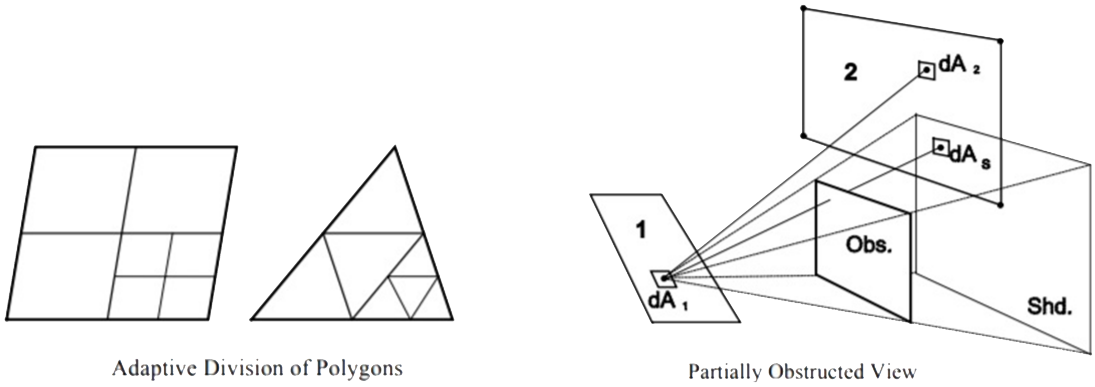


Figure 2.7: Walton et al. polygon-based quadratures [Wal02]. Left: recursive divisions of polygons. Right: projection of obstacle's shadow onto the target face.

View factor computations being a purely geometric and visibility problematic, GPU solutions naturally come to mind. D'Azevedo et al. [DAHSW14] proposed a method to compute view factors on GPU-based parallel computers. Kramer et al. detailed the point-to-area view factor computations using an OpenGL-based hemicube method [KGP+14]. Richmond et al. detailed a robust computational framework to compute view factors through a multiple GPU-accelerated ray tracing scheme [RHSB21], handling the shadow effect thanks to the Möller-Trumbore ray-triangle intersection algorithm [MT97]. A review of the methods for evaluation of radiation view factor between two concentric finite cylinders with different length has been proposed by Gupta et al [GBS+17]. Arias-Rosales et al. performed a comparison of view factor modeling frameworks with stochastic rays simulations [ARL20].

The recent advances in topological optimization and additive manufacturing yield an increasing number of complex shapes with improved tradeoff between weight and

structural strength (see Fig. 2.8). The purpose is to the reduce manufacturing costs (if the material is expensive such as gold, platinum or titanium) and the launching costs (the heavier a satellite, the higher the cost for launching phase). Such shapes, common in modern design of satellites where weight reduction is crucial, challenge all subsequent radiative thermal simulations as the objects contain many more free-form surface elements and obstacles.

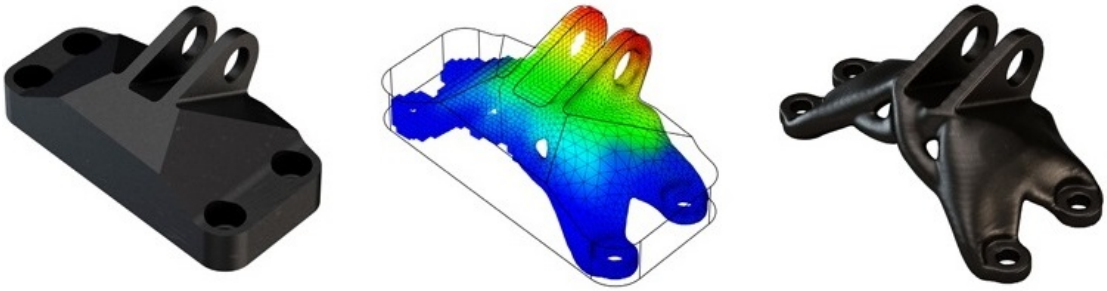


Figure 2.8: Topological optimization applied to a support structure. Left: original model. Middle: topological optimization process. Right: output model. Considerably more surface elements are required compared to the original model.

2.2 Overview of Model Reduction Approaches

2.2.1 Mathematical Model Reduction

The common thread of approaches for faster numerical simulations proceed by mathematical model reduction. The latter consists in reducing the degrees of freedom of a partial differential system of equations, without changing its behavior significantly. The objective is to approximate the original system as accurately as possible, while reducing the system order as much as possible, the key principle of these methods lies in a change of basis. While these methods are very effective, they often assume steady-state simulations. Model reduction is currently mostly performed manually, which consumes considerable time, effort and costs. This is particularly true for computing radiations as the radiative term is strongly nonlinear. When dealing with mathematical order reduction for linear systems, sub-structuring techniques come to mind such as Guyan [Guy65] or Craig-Bampton [CB68]. Guyan approach consists in ignoring the inertial terms of the equilibrium equations and expressing the unloaded degrees of freedom in terms of the loaded degrees of freedom. In other words, considering a stiffness matrix $\mathbf{K}$, a force vector $\mathbf{f}$ and displacement vector $\mathbf{d}$, the static equilibrium equation can be written as:

$$\boldsymbol{K}.\boldsymbol{d} = \boldsymbol{f} \tag{2.2}$$

Now considering the different master/slave degrees of freedom of the system, the expression can be rewritten as:

$$\begin{pmatrix} \boldsymbol{K_{mm}} & \boldsymbol{K_{ms}} \\ \boldsymbol{K_{sm}} & \boldsymbol{K_{ss}} \end{pmatrix} \begin{pmatrix} \boldsymbol{d_m} \\ \boldsymbol{d_s} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f_m} \\ \boldsymbol{0} \end{pmatrix} \tag{2.3}$$

The second line of the above system can be expressed as:

$$\boldsymbol{K_{sm}}.\boldsymbol{d_m} + \boldsymbol{K_{ss}}.\boldsymbol{d_s} = 0 \tag{2.4}$$

Which gives:

$$\boldsymbol{d_s} = -\boldsymbol{K_{ss}^{-1}}.\boldsymbol{K_{sm}}.\boldsymbol{d_m} \tag{2.5}$$

The final reduced system is obtained by substituting the previous result into the first line of the original system, resulting in:

$$(\boldsymbol{K_{mm}} - \boldsymbol{K_{ms}}\boldsymbol{K_{ss}^{-1}}\boldsymbol{K_{sm}})\boldsymbol{d_m} = \boldsymbol{f} \tag{2.6}$$

In the same spirit the Craig-Bampton method proceeds a distinction between boundary and interior degrees of freedom.

Gaume et al. [GJQ19] as well as [GLB+21] proposed, respectively, a modal based reduction approach for radiative and conductive model order reduction problematics, commonly used in mechanical structures analysis as part of the finite element method. The modal approaches seek to remove the modes which contributions to the considered system can be neglected [Var95]. Another mathematical model reduction named Proper Orthogonal Decomposition (POD) is commonly used in the industry, notably for fluid dynamics where the Navier-Stokes equations are replaced by simpler systems to solve [BHL93]. The main use of this method is to decompose a physical field depending on the different variables that influence its physical behaviors. The process relies on an orthogonal decomposition along with the principal components of the field [KGVB05]. It can then be assimilated with the Principal Component Analysis (PCA) [Hot33] method or the Singular Value Decomposition (SVD) [EY36] one.

Mathematical order reduction dedicated to space thermal analysis is a research topic by itself. A first model reduction comes from the nodal breakdown, see Fig. 2.9. The transition from the real satellite to a thermal model is ensured by a decomposition into elements called nodes, thus achieving a discretization of the physical equations [SA85] [BCGP+96]. This decomposition is easily justified when the delimited elements have a low thermal gradient, that is to say when their internal thermal resistance is low (isothermal node). It is then possible to characterize an element by a single temperature representing the average value of the temperatures at the different points of the element, so that the overall radiated power is conserved. Thus, the principle consists in establishing a set of isothermal nodes linked together by conductive, radiative or convective couplings.

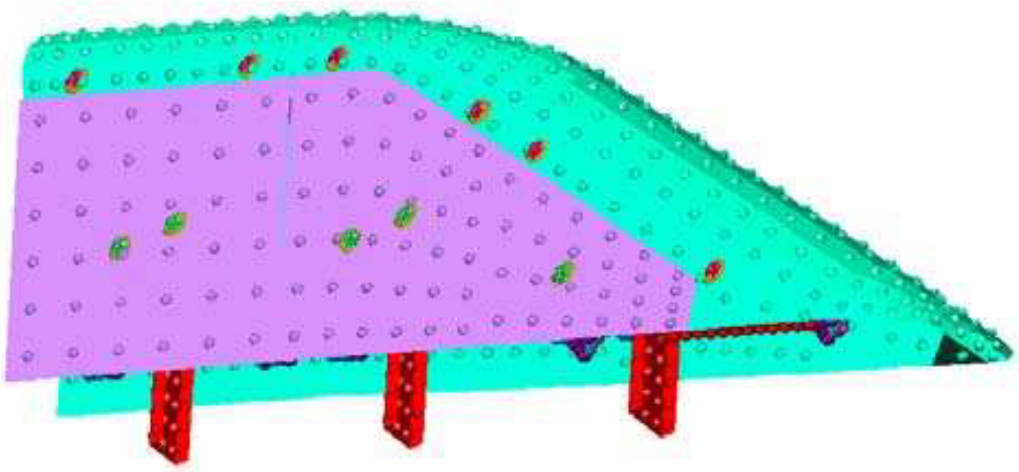


Figure 2.9: Thermal model nodal breakdown applied to a winglet profile. Fossati et al. [FGdpL06]

Basset et al. [BHCF16] developed a thermal model reduction tool (TMRT) for spacecrafts, valuable when a reduced model is required by launcher authority for coupled analysis. Usually, when the spacecraft model is integrated into the launcher model, the maximum allowed budget of thermal nodes is often exceeded, hence a reduction of the mathematical spacecraft model has to be proceeded. The main concept being to define average nodes as a grouping of thermal nodes, as well as kept nodes and eliminated nodes that can be involved or not in thermal equations, resulting in a mathematical model order reduction.

Jacques et al. [Jac16] introduced the concept of super nodes for thermal model reduction, see Fig. 2.10. A super node is defined as a new mathematical node whose temperature equals the average temperature of a group of nodes of the underlying detailed finite element model. Weights of the super nodes are given by the capacitance associated to the node divided by the total capacitance of the cluster. The conductive reduced model is then obtained through sub-structuring method.

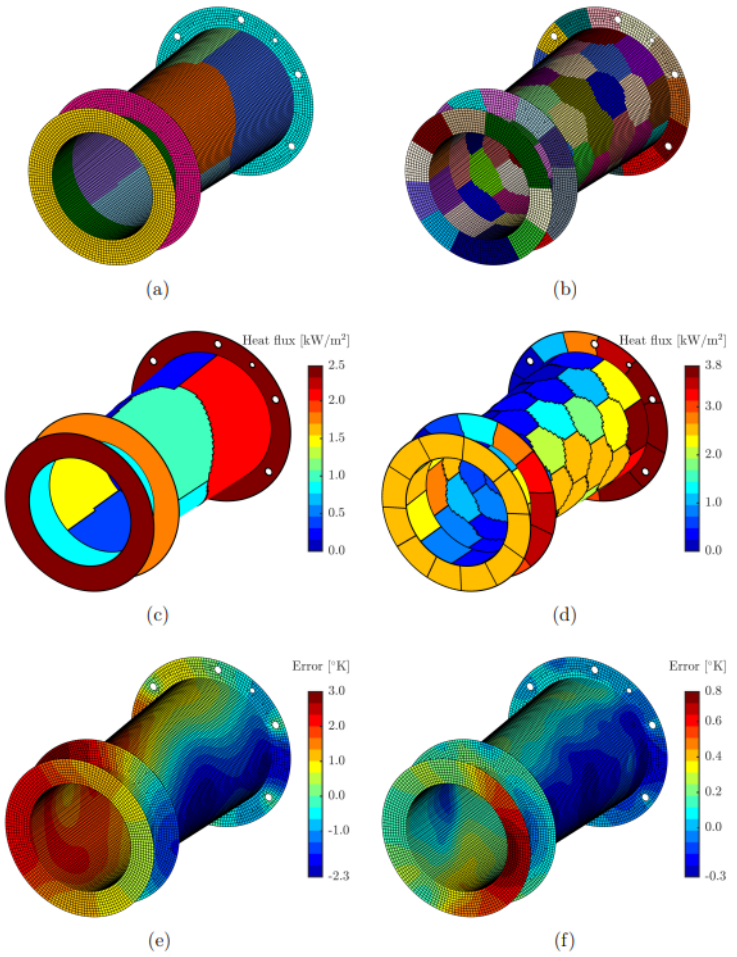


Figure 2.10: Jacques et al. super nodes [Jac16]. Left: 10 super nodes. Right: 100 super nodes. Top (a-b): region growing clustering results. Centre (c-d): average solar heat flux absorbed by the super face. Bottom (e-f): recovered detailed temperature error.

2.2.2 Geometric Model Reduction

Geometric order reduction methods have been mainly developed in the geometric modeling community [KCS98], notably to tackle radiosity problematics such as real-time rendering of complex polygonal 3D scenes. The main objective of these approaches is to preserve a specific feature (such as planarity, normal orientations or volume) while reducing the number of mesh elements by vertex removal (see Fig. 2.11), edge collapse (see Fig. 2.12), or face suppression (see Fig. 2.13). The edge collapse operation replaces an edge with a single vertex and removes 2 triangular faces. The coordinates of the newly obtained vertex are computed according to a chosen geometric error metric.

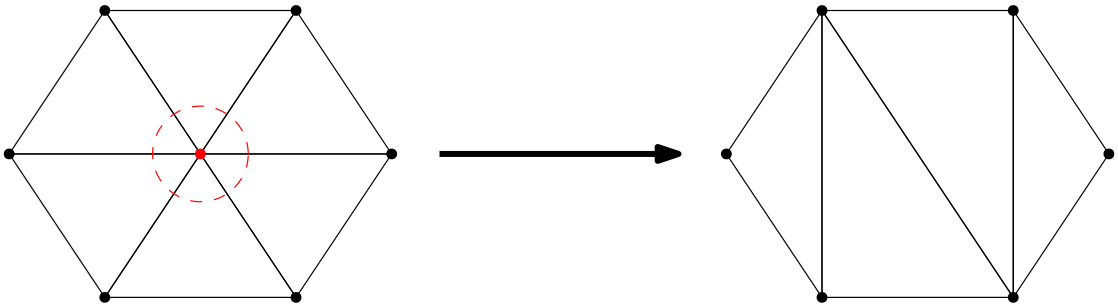


Figure 2.11: Vertex removal operator. Number of vertices and triangles reduced by 1 and 2, respectively. Neighborhood elements are updated.

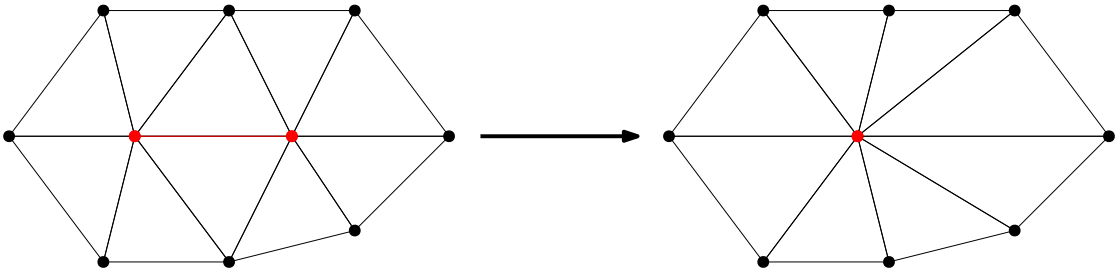


Figure 2.12: Edge collapse operator. Number of vertices and triangles reduced by 1 and 2, respectively. Neighborhood elements are updated. The half-edge collapse operator would place the new vertex at the position of one of the vertices belonging to the deleted edge.

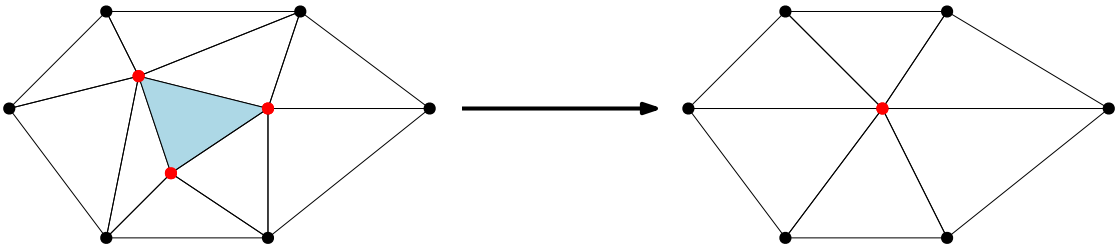


Figure 2.13: Face suppression operator. Number of triangles reduced by 4. Neighborhood elements are updated. This approach removes more geometric elements compared to the previous ones, offering less freedom over the decimation possibilities.

Each of the previous atomic operations are coupled with a specific error metric, that will guide the decimation process with respect to a desired feature to preserve. The position of a new element after applying an operator to the mesh as well as the order in which the operators are applied are determined by the error metric.

First designed error metrics were accurate to specific distances, for instance the distance to an average plane with the work of Schroeder et al. [SZL92]. Then Klein

Algorithm 1 Generic Mesh Decimation with Edge Collapse

```
1: INPUT: a triangular mesh T and a stop criterion stop (e.g.: number of operators or maximum error)
2: Declaration of priority queue pqueue
3: For each edge e ∈ T do
4:     Simulate collapse of e
5:     if e collapse possible then
6:         Compute error and new vertex position v*
7:         Push (e, error, v*) in pqueue
8: while pqueue not empty and !stop do
9:     operator ← pqueue top first element
10:    operator pop (removed from pqueue)
11:    Do collapse operator
12:    Update pqueue with modified edges and edges to erase
   return Reduced mesh T
```

et al. proposed an error metric accurate to the Hausdorff distance [KLS96], whereas Ronfard et al. relied on the maximum distance to supporting planes [RR96].

Such methods are either memory-aware or memory-unaware. An example of the latter is the Lindstrom-Turk error metric [LT98], developed to preserve the volume of the original model. The vertex position and cost computations after an edge collapse require only the face connectivity and the the vertex locations in the intermediate model. Specifically, in order to conserve the volume of a mesh, the volumes of the tetrahedrons formed by the newly obtained vertex and each face belonging to the set of faces in the one-ring neighborhood of the collapsed edge are minimized (see Fig. 2.14 and Fig. 2.15).

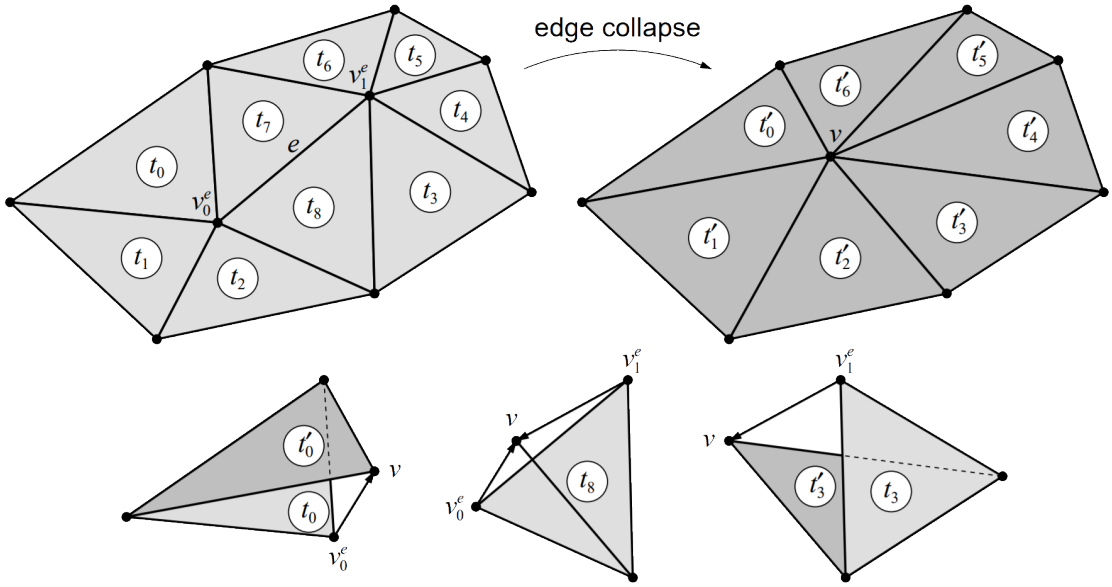


Figure 2.14: Edge collapse operator with tetrahedrons associated to the newly obtained vertex as described by Lindstrom et al [LT98]. Edge e is collapsed and replaced by vertex v. Tetrahedrons volumes associated to triangles t_0, t_3 and t_8 are depicted.

Figure 2.15: Surface mesh simplification from the CGAL library [The20] using Lindstrom-Turk placement and cost strategy.

For the memory-aware approaches, the most common one is the Quadric Error Metric (QEM) developed by M. Garland and P. Heckbert [GH97]. The key idea is to associate a set of planes for each vertex of the mesh and to define the error of the vertex with respect to this set as the sum of squared distances to its planes, see Fig. 2.16. Since a plane can be defined by the equation $ax + by + cz + d = 0$ with $a^2 + b^2 + c^2 = 1$, the following matrix can be computed:

$$\boldsymbol{Q_{plane}} = \begin{pmatrix} a^2 & ab & ac & ad \\ ab & b^2 & bc & bd \\ ac & bc & c^2 & cd \\ ad & bd & cd & d^2 \end{pmatrix} \tag{2.7}$$

The distance D from a vertex $v = [v_x, v_y, v_z, 1]^t$ to a plane is then given by:

$$D^2(v) = v^t \boldsymbol{Q_{plane}} v \tag{2.8}$$

Then the quadric error associated to a vertex results in the summation of squared distances with all planes to the vertex, resulting in:

$$E(v) = \sum_{planes\ of\ v} D^2(v) = v^t \Big(\sum_{planes\ of\ v} \boldsymbol{Q_{plane}} \Big) v \tag{2.9}$$

For each collapse operation, the quadric are summed to keep the memory of the original mesh during the decimation process.

An important degree of freedom of mesh decimation algorithms is the position of the new vertex. For Quadric Error Metric, the error function being quadratic (see 2.9), v^* is obtained by solving:

$$\frac{\partial E(v)}{\partial x} = \frac{\partial E(v)}{\partial y} = \frac{\partial E(v)}{\partial z} = 0 \tag{2.10}$$

Which can also be written as:

$$\underbrace{\begin{pmatrix} q_{11} & q_{12} & q_{13} & q_{14} \\ q_{21} & q_{22} & q_{23} & q_{24} \\ q_{31} & q_{32} & q_{33} & q_{34} \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathrm{A}} \begin{pmatrix} v_x \\ v_y \\ v_z \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \tag{2.11}$$

If the matrix is invertible, the optimal position for v^* is:

$$v^* = A^{-1} b \tag{2.12}$$

However if A is not invertible, a singular value decomposition is performed [Lin00], yielding a numerically robust solution.

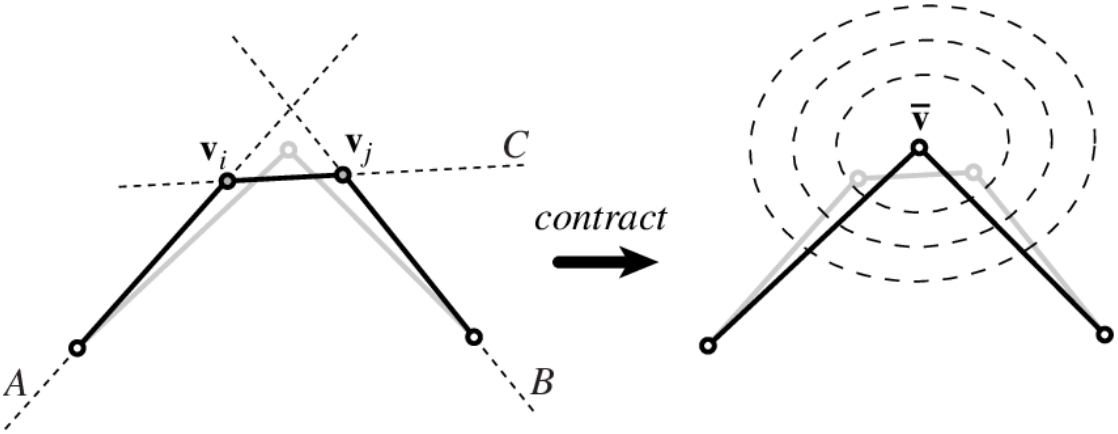


Figure 2.16: Edge collapse operator with Garland and Heckbert Quadric Error Metric [Gar99]. The dashed elliptical curves around the new vertex represent isocontours for various errors.

The constructed quadrics have an important geometric interpretation that can be rendered and visualized. For a given quadric Q, a level surface ϵ is the set of all points whose error with respect to Q is ϵ. The representation of the said level surface is an ellipsoid in the common scenario, but can be either cylinders or parallel planes for degenerative cases (see Fig. 2.17).

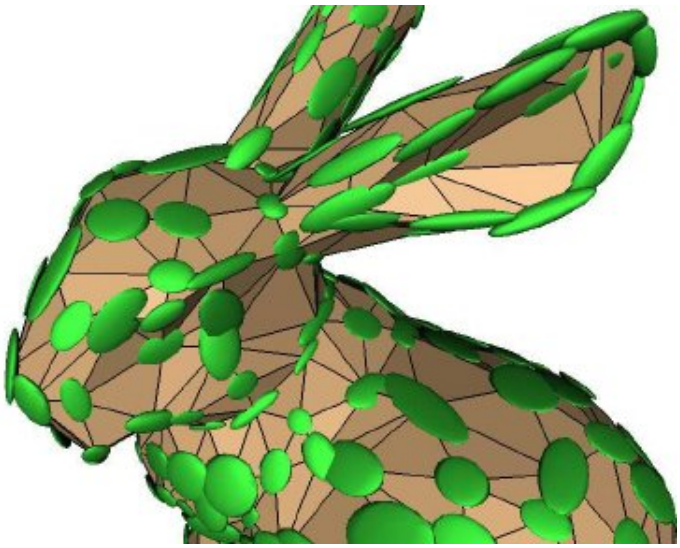


Figure 2.17: Isosurfaces of each quadric represented around corresponding vertices, from Garland et al [GH00].

A recent improved version of the QEM has been proposed by Trettner and Kobbelt [TK20], called probabilistic quadrics. The input mesh is treated as genuinely uncer-

tain information and the optimal point minimizes the expected squared error. Also an extension of the quadric error metric for mesh decimation with appearance attributes has been proposed by H. Hoppe [Hop99] (see Fig. 2.18), notably used for progressive compression of textured meshes in 5D ((x, y, z) and (u, v) coordinates) [PAH+19].

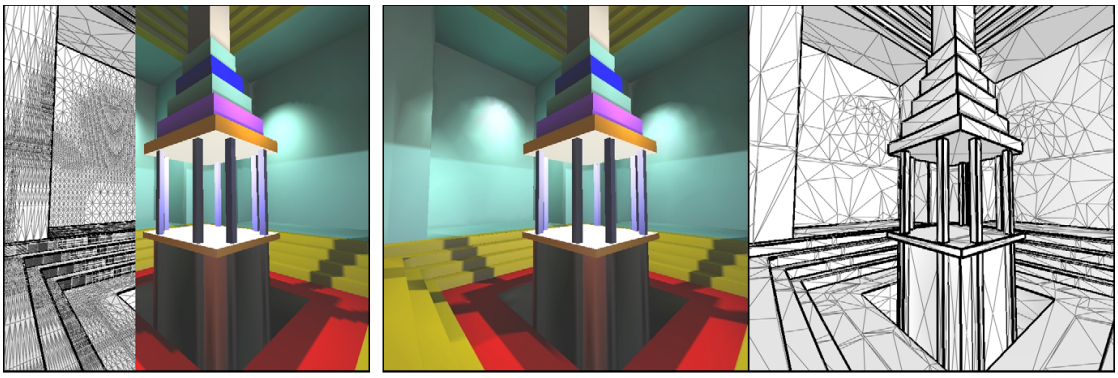


Figure 2.18: Mesh simplification with color attributes from H. Hoppe [Hop99]. Left: original mesh with 298 468 faces. Right: reduced mesh with 5000 faces.

Cohen-Steiner et al. introduced the notion of geometric proxies as best-fit geometric surrogates to effectively circumvent topological issues of a mesh with the Variational Shape Approximation (VSA) algorithm [CSAD04]. Digne et al. proposed an algorithm to reconstruct a surface from a set of points through surface simplification and an error metric involving optimal transport [DCSA+13]. Recently, a structure-aware mesh decimation has been suggested by Salinas et al. [SLA15] taking as input a set of planar proxies to approximate correctly the planar parts of a mesh for high decimations.

Moreover, crucial features of a mesh can be preserved through user-guided decimation. For instance by weighting original quadrics of vertices and adding constraint quadrics [KG03] or by directly weighting the cost of an edge collapse [PS03]. Furthermore, a user-assisted simplification method for triangle meshes preserving boundaries has been proposed by Gonzalez et al [GGC+09].

Recent works have been published to tackle the problematic of multiple meshes constituting a whole CAD model, for instance Ghazanfarpour et al. [Gha19] proposed an approach for the joint decimation of multiple triangular meshes by combining Quadric Error Metric with a proximity-aware penalty function in order to increase the error of edge collapses modifying the geometry in proximity areas of multiple meshes (see Fig. 2.19).

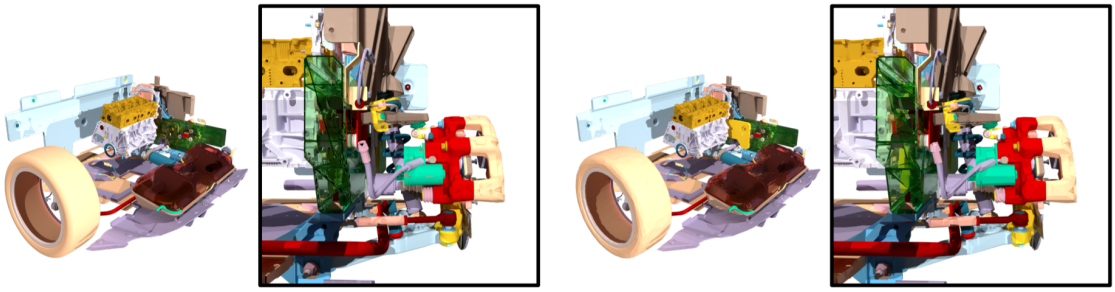


Figure 2.19: Multiple meshes decimation from Ghazanfarpour et al [GMH+20]. Left: 425 meshes and 3M faces. Right: 150k faces.

The main drawback of these methods is that they are physics unaware, so that the reduced geometric model can deteriorate important mathematical or physical properties. Recent physics-informed contributions have been made regarding mesh generation and reconstruction accurate to finite element simulations [Fre04] [WH19] and mesh decimation applied to unstructured finite elements for real-time Hurricane storm surge forecasting [BHM20]. This motivates the need for a geometric model reduction method tailored to numerical simulations.

2.3 Contributions

In this work, two lines of research regarding geometric aspects for radiative thermal simulation have been considered and investigated.

2.3.1 Progressive Geometric View Factors

First, an approach to compute geometric view factors is detailed. Justified by the need of real time simulation, our approach is made progressive, and supplemented by a prediction step to close the gap between the approximated value obtained through quadrature in presence of visibility obstructions and the reference value. Also, in the interests of accuracy, exact predicates and closed form formulations are favored. The method considers triangular surface meshes, stored as a forest data structure. The two possible scenarios are (1) if two faces fully see each others, the closed form formulad from Schroder et al. [SH93] is applied. Otherwise, (2) a polygon-based quadrature is initiated. The method is compared with standard ray-shooting algorithms, which can be found in common thermal industrial software.

2.3.2 Geometric Model Reduction driven by Numerical Simulation

The second focus of this work is put on the geometric model reduction faithful to numerical simulation results. Since radiative thermal computations are governed by the geometry (hence view factors) of the spacecraft/mechanical part, the key idea is to reduce the geometrical model while being accurate to the reference calculation case. State

of the art algorithms for mesh decimation are considered [GH97, LT98] and applied on mechanical part and spacecraft meshes partitioned into clusters of faces with isothermal properties (thermal nodes). A sensitivity analysis is applied to highlight clusters sensitive to reduction in order to guide the decimation process. Furthermore, a factor called importance for each cluster is computed based on the distortion between reference and reduced cases. This quantity is then coupled with the geometric cost of a collapse operator to make the reduction physics informed.

2.3.3 Publications, Presentations and Others

Conference Proceedings

- **V. Vadez**, F. Brunetti, *"Reduced Geometric Model of Sentinel-3A for Radiative Thermal Simulation"*, submitted to International Conference on Environmental Sytems **(ICES)**, 2022
- **V. Vadez**, F. Brunetti, P. Alliez, *"Geometric Model Reduction Accurate to Numerical Simulation Accuracy"*, International Conference on Environmental Sytems **(ICES)**, remote, July, 2021
- **V. Vadez**, F. Brunetti, P. Alliez, *"Progressive Geometric View Factors for Radiative Thermal Simulation"*, International Conference on Environmental Sytems **(ICES)**, Lisbon, Portugal, July, 2020

Workshop Presentations

- **V. Vadez**, F. Brunetti, *"TMM Reduction using Machine Learning Techniques"*, European Space Thermal Engineering Workshop **(ESTEW)**, remote, October, 2021
- **V. Vadez**, F. Brunetti, P. Alliez, *"Physics Driven Geometrical Model Reduction (GMM)"*, European Space Thermal Engineering Workshop **(ESTEW)**, remote, October, 2020
- **V. Vadez**, F. Brunetti, P. Alliez, *"First steps of numerical simulation using Artificial Intelligence"*, European Space Thermal Engineering Workshop **(ESTEW)**, Amsterdam, Netherlands, October, 2019

Teaching

- Numerical Analysis Teaching assistant at Polytech Nice Sophia Engineering's School, 2018-2022
- Model Reduction at Polytech Nice Sophia Engineering's School, 2020-2022

Miscellaneous

The results of the thesis, in particular the reduction of the geometry of the Sentinel-3A satellite, allowed DOREA to have the support of the French National Centre for Space Studies (CNES) for the development of a real-time thermal digital twin conditioned on a sufficiently small model and faithful to the detailed model. The project is funded by ESA & CNES (250K€) in collaboration with Thales Alenia Space, signed in April 2022.

Chapter 3

Computing Progressive View Factors

Contents

3.1 Background

This chapter focuses on the problem of computing view factors between the planar facet elements of an object approximated by a surface triangular mesh. More specifically, a *progressive* and *interruptible* approach which trades accuracy for time is described, mainly motivated by industrial applications where time is critical, such as a satellite thermal digital twin.

In particular, a hybrid geometric & numerical method that performs an adaptive polygon-based quadrature is detailed. A hierarchical axis-aligned bounding box (AABB) tree data structure is used to accelerate obstacles detection, and a hierarchical forest data structure stores the input triangles that are recursively subdivided while the accuracy improves over time. More specifically, pairs of faces with either full or null mutual visibility are exactly detected by intersecting their 3D convex hull with the primitives of the AABB tree. These two cases favor closed-form solutions and early terminations. Pairs of faces with partial mutual visibility are adaptively partitioned, notably using a SVM-based approach (support vector machine) such that the shadows cast by the obstacles (penumbra or umbra) are later best approximated by summing closed form solutions. This adaptive subdivision approach is coupled with a *prediction* step relying on point-based quadratures and segments shooting. This method is validated for closed form solutions and the convergence rates are compared to common state-of-the art numerical methods.

The method is summarized as follows. It takes as input a surface triangular mesh with N facets. Each facet is oriented so that heat is emitted under the uniform hemispherical model in the half-space according to the facet normal. In the particular case where a facet is transferring heat on both sides, for instance facets of the solar generators of a spacecraft, it is split into two facets with opposite orientations. The output is generated as a $N \times N$ matrix M, progressively refined over time, where each entry $OF_{i \to j}$ denotes the approximated obstructed view factor between facet i and facet j. The matrix is sparse when the input geometric model has many full obstructions between facet pairs. The main objective is to best trade accuracy for time, while refining M.

3.2 Progressive View Factors

As discussed in the state-of-the-art section about view factor computations, those can be exactly computed in closed form when the two considered geometric elements fully see each other. Otherwise, view factors can be approximated either by a point-based quadrature or a polygon-based quadrature in presence of visibility obstacles between the emitter and receiver elements. In the following, a polygon-based quadrature is favored primarily for its capability to compute view factors in closed form in the absence of obstacles [SH93]. The key idea is to maximize closed forms during the quadrature process in order to rely on exact results for the view factors approximation. In addition, the numerical integration process is made recursive to offer progressiveness and interruptibility. The three degrees of freedom of this progressive quadrature method are (1) the way to

split the triangles into smaller convex polygon elements, (2) the order in which we apply the splitting operators, (3) and the prediction method used when splitting operators are stopped.

3.2.1 Polygon-based Quadrature

Pre-processing steps on the mesh are required before computing the view factors. First, an index i is assigned to each facet of the input mesh. Then, an iteration over each pair of facets (f_i, f_j) is proceeded, with $i < j$, to check whether facets i and j are mutually (partially or entirely) in their respective visible half-space, see Fig. 3.1. Two facets (f_i, f_j) are considered seeing each other when at least one point of facet f_j is on the positive side of the plane associated to facet f_i and vice versa. If it is not the case, both $OF_{i\to j}$ and $OF_{j\to i}$ are set to 0, otherwise the numerical integration can go on for (f_i, f_j). This procedure increases the sparsity of the view factor matrix.

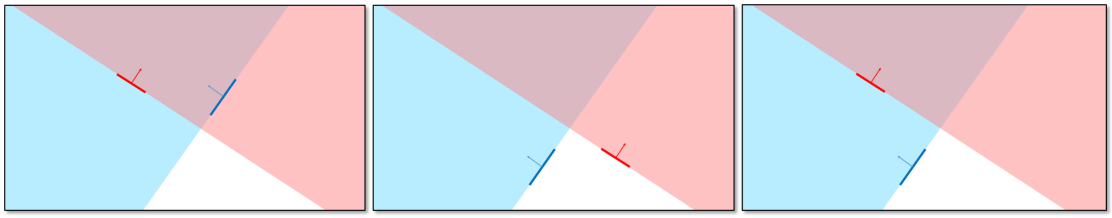


Figure 3.1: Half-space visibility between two faces. Left: each facet is contained in the half-space of the other facet, hence they see each other. Middle: none of the facet are contained in the half-space of the other facet, hence they do not see each other. Right: one facet sees the other facet, but the converse is not true, hence they do not see each other.

Algorithm 2 Progressive view factor algorithm

```
1: INPUT: a triangular mesh T composed of N oriented facets
2: Let i,j two oriented facets or fragments belonging to T
3: procedure FillVFMatrix(T, VF_matrix)
4:     if (half_space_reciprocity(i,j)) then [Algorithm 3]
5:         oc_type = compute_oc(i, j) [Algorithm 4]
6:         if (oc_type == no_occlusion) then
7:             Use closed form solution to increment VF_matrix
8:         else if (oc_type == full occlusion) then
9:             Nothing to do
10:        else (oc_type == partial occlusion)
11:            Iterative subdivision
```

For each pair of original facets (f_i, f_j), the key idea of the numerical integration method consists in summing closed-form view factors between convex polygons that

Algorithm 3 Halfspace Reciprocity algorithm

1: INPUT: *2 oriented facets (i,j) belonging to T*
2: **procedure** HALF_SPACE_RECIPROCITY(i,j)
3: *A1 = Facet(i)->halfedge()->vertex()->point()*
4: *B1 = Facet(i)->halfedge()->next()->vertex()->point()*
5: *C1 = Facet(i)->halfedge()->next()->next()->vertex()->point()*
6: *A2 = Facet(j)->halfedge()->vertex()->point()*
7: *B2 = Facet(j)->halfedge()->next()->vertex()->point()*
8: *C2 = Facet(j)->halfedge()->next()->next()->vertex()->point()*
9: $Plane_1 = Plane(A1, B1, C1)$ *//determines plane equation using 3 points*
10: $Plane_2 = Plane(A2, B2, C2)$
11: *//checks if at least one point from facet j is on positive side of* $Plane_1$
12: *bool* $i\_sees\_j = A2 \in Plane_1^+ \| \ B2 \in Plane_1^+ \| \ C2 \in Plane_1^+$
13: **if** $(i\_sees\_j == false)$ **then**
14: *return false*
15: *bool* $j\_sees\_i = A1 \in Plane2^+ \| \ B1 \in Plane2^+ \| \ C1 \in Plane2^+$
16: **if** $(j\_sees\_i == false)$ **then**
17: *return false*
18: *return true*

Algorithm 4 Determination of occlusion type algorithm

1: INPUT: *2 oriented facets (i,j) belonging to T*
2: **procedure** COMPUTE_OC(i,j)
3: *Compute convex hull CH between i and j*
4: **if** *(no other facet in CH)* **then**
5: *oc_type = no_occlusion*
6: **else if** *(union of facets in CH totally hide i from j)* **then**
7: *oc_type = total_occlusion*
8: **else**
9: *oc_type = partial_occlusion*
10: *return oc_type*

correspond to the entirety or fragments of facets f_i and f_j. This process is made possible thanks to view factor properties detailed next.

In the reminder, we denote by f_i^k the k^{th} fragment of the facet f_i and f_j^l the l^{th} fragment of the facet f_j. When two fragments f_i^k and f_j^l are fully visible, 100% of their closed form view factor contribute to $OF_{i\to j}$ and $OF_{j\to i}$ thanks to the reciprocity property 1.9. When the visibility between two fragments is partial, both f_i^k and f_j^l are subject to further recursions.

3.2.1.1 Forest Data Structure

Since the detailed method increments values in the global view factor matrix M with closed forms only, this recursive process can be implemented through a hierarchical *forest* data structure (see Fig. 3.2) in order to store the facet fragments. Such a hierarchical approach is similar to the work of Hanrahan et al [HSA91]. In particular, the forest can be described as a list of trees, where each tree is initialized by a facet of the original input mesh (i.e., not a fragment). The facet fragments f_i^k of root facet f_i, obtained through recursive splitting, are stored as child nodes of the corresponding split node. A *leaf* node of a tree has no child nodes. Each split operator, occurring on a leaf node of a tree when the two considered fragments partially see each other, generates new facet fragments (partitioning f_i^k) stored as children of the split node. Furthermore, each leaf node possesses a list of visible nodes which corresponds to all facets or facet fragments seen from the considered facet node. The main steps of our method are depicted on the following flowchart, see Fig. 3.3.

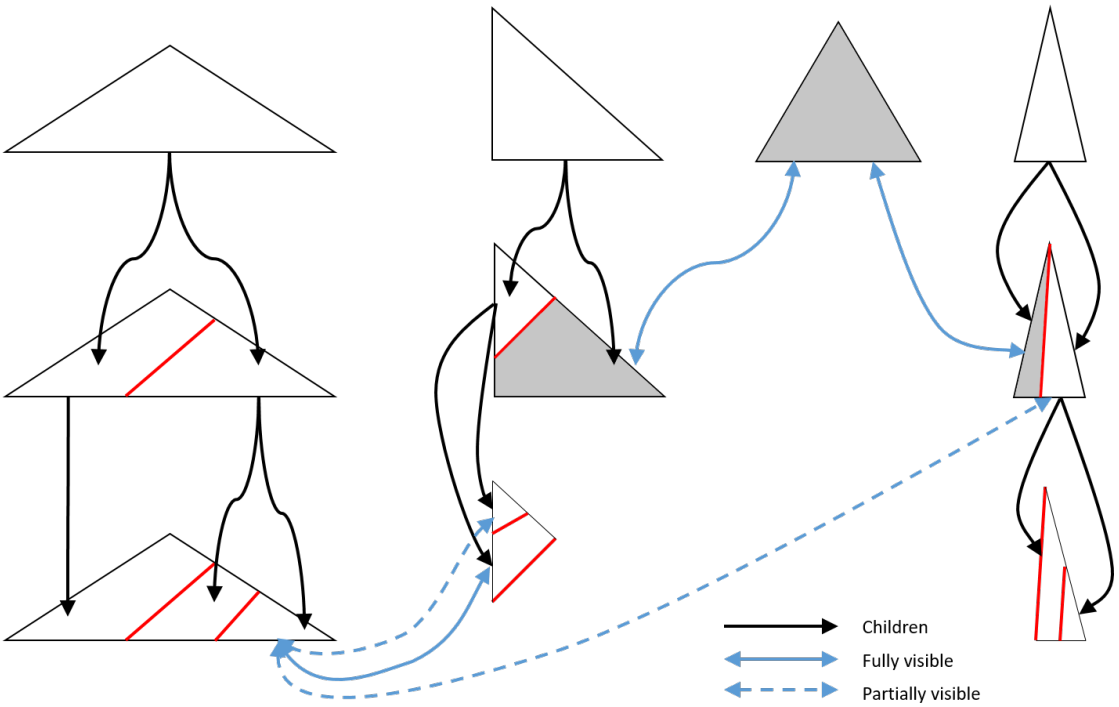


Figure 3.2: Forest Data Structure detailed with every possible occlusion scenario. Top: root nodes corresponding to triangle facets of the input mesh. Middle: child nodes of root nodes. Splitting operators are depicted in red. Blue filled-in lines represent the list of fully visible nodes, and dashed blue lines depict the partially visible nodes. Only a subset of blue arrows are shown for the sake of clarity. Closed forms are used between pairs of fully visible nodes (represented by grey polygons). Bottom: child nodes of middle row nodes. The recursive splitting operators occur only for partially visible nodes.

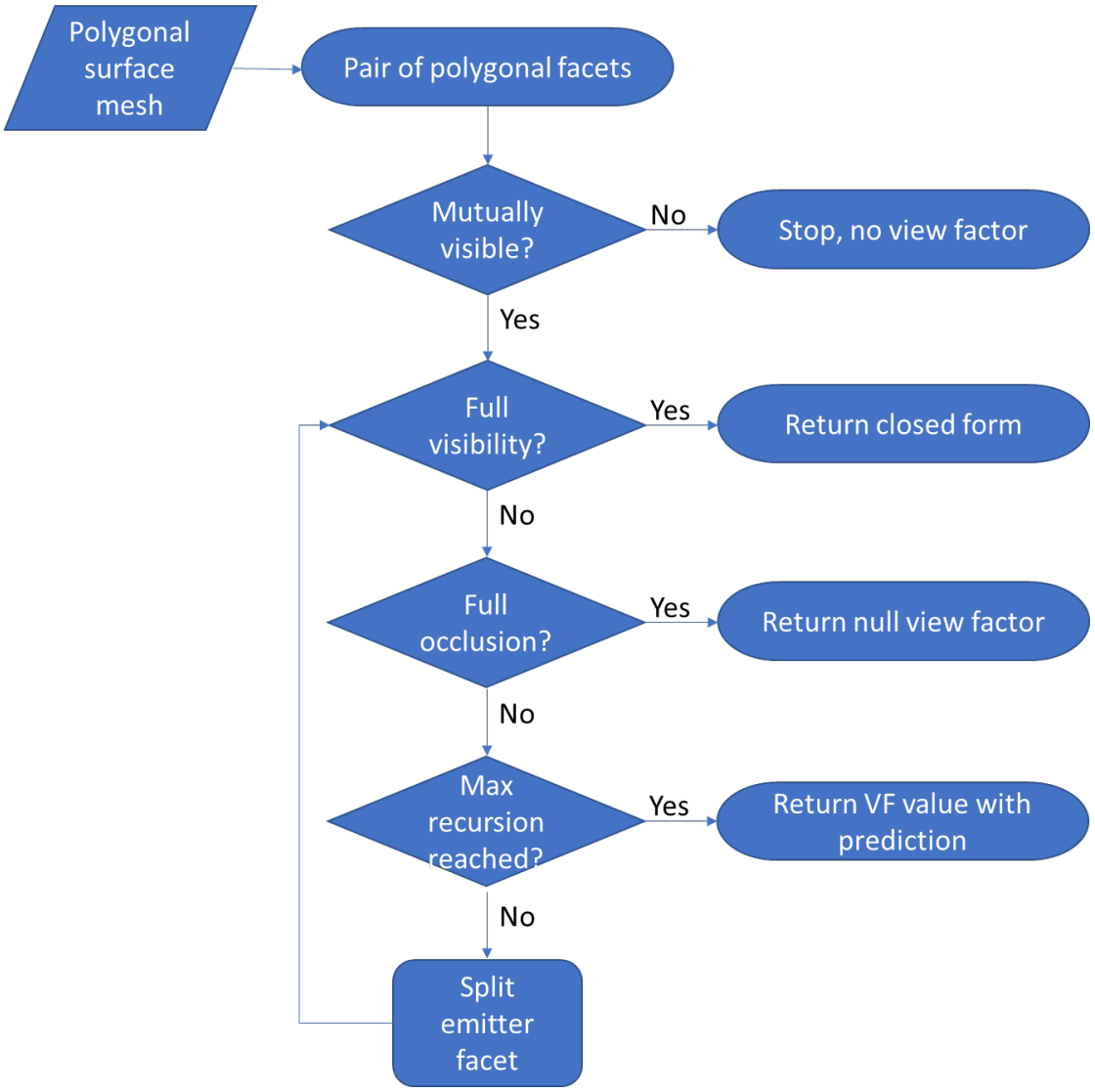


Figure 3.3: Flowchart summarizing the main steps of our approach to compute view factors.

3.2.1.2 Handling Intersections

Detecting when an obstacle partially occludes the visibility between pairs of convex polygons is rather straightforward. However, detecting full visibility or full occlusion is more compute-intensive as requires complex geometric operations. The common segment-shooting approach benefits from an ease of implementation but only provides approximate answers. Wallace et al. proposed a ray tracing algorithm for progressive radiosity problem, in which ray tracing is used to perform the numerical integration of the view factor equation [WEH89]. Shaft culling techniques coupled with ray-tracing for comput-

ing radiosity was investigated by Haines and Wallace [HW94]. Durand et al. introduced extended projection operators of occluders onto a set of projection planes to create extended occlusion maps [DDTP00]. An effective method to compute occlusion formed by a set of occluders has been proposed by Schaufler et al., also referred to as occluder fusion [SDDS00]. Zeeb et al. contributed a space subdivision technique with the notion of voxel [Zee02], initially applied jointly with octrees as explained by Glassner et al [Gla84], notably useful for volume approximations of 3D models [SMS+16] and repair of polygonal models [NT03]. The approach consists in checking the intersections between faces included in the current voxel and the considered ray, then move to the adjacent voxel, and so on. Jacques et al. developed a uniform sampling based on a new method referred to as isocell, in order to improve the view factor computations [JMK13], see Fig. 2.6. Apostu et al. deal with occlusion issues using exact from-polygon visibility [AMDL12]. To address this problem with previous forest data structure, the 3D convex hull of the polygon pair is computed exactly using an exact arithmetic kernel from the CGAL library [The20]. A 3D convex hull represents the smallest convex 3D polyhedron containing the two considered facet fragments, see Fig. 3.4.

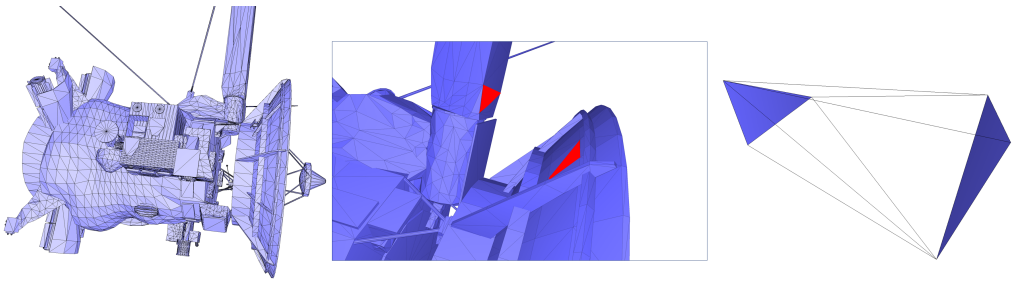


Figure 3.4: 3D convex hull on the Cassini 3D model, courtesy of nasa3d.arc.nasa.gov. Left: input surface triangle mesh. Middle: closeup on a pair of facets depicted in red color. Right: 3D convex hull of the considered facet pair.

We then compute the intersection of the convex hull with the original input triangle facets (subtracting the two original facets containing the fragments). This process being compute-intensive when performed in brute force against all input facets of a complex model, we utilize an accelerating hierarchical data structure in the form of a static axis-aligned bounding box (AABB) [ATC20], see Fig. 3.5.

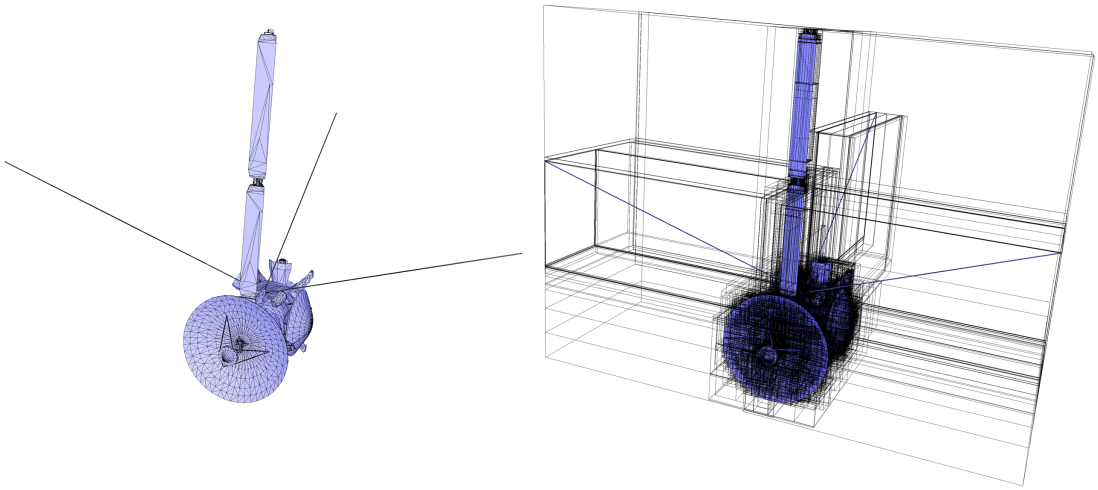


Figure 3.5: AABB tree of the Cassini 3D model. Left: input surface triangle mesh. Right: input mesh with its corresponding AABB tree.

The convex hull is partitioned into tetrahedra, and the AABB tree is requested to compute intersections with these tetrahedra. If none of the given tetrahedra are intersected, then the two corresponding polygon fragments are labeled as fully visible. Testing full occlusion is achieved by checking whether the convex hull is disconnected by its intersection with the input facets. More specifically, full occlusion is established when the source and target facet fragments are disconnected by the union of input facets intersected by the convex hull.

3.2.1.3 Optimizing Intersection Detection through Probing

As discussed above, 3D convex hulls are relevant for the full visibility and full occlusion scenarios as they lend to early terminations of the recursive splitting process. However, this method is inadequate and overly expensive for pairs of facet fragments with partial mutual visibility. To tackle this problem, a prior filtering phase based on line-probing is added. Similar to ray-shooting methods, line segments are thrown according to a user-specified number, then intersections between these line segments and the input facets are performed. In particular, each node contained in the tree is point-sampled uniformly, then potential intersections between random line segments connecting two samples on each fragment of the pair and the AABB tree are computed, resulting in three possible scenarios. If none of the line segments intersect the AABB tree, an exact full visibility test is done. If all of them intersect the AABB tree, an exact full occlusion test is processed. If not, the visibility is considered partial and a recursive splitting is performed, expanding the tree size of the current original facet.

Also, the way of sampling the fragments has an important impact on the effective filtering and quality prediction. Notably, it was observed by Jacques et al. that a uniform sampling yields better result than a random one for view factor computations.

Algorithm 5 Line-probing algorithm for early terminations.

```
1: INPUT: 2 oriented facets (i,j) and n line-segments
2: procedure PROBING(i,j,n)
3:     Uniformly point-sample facets i and j
4:     Construct n random line-segments between i and j samples
5:     if (no line-segment intersects AABB-tree) then
6:         try full visibility
7:     else if (all line-segments intersect AABB-tree) then
8:         try full occlusion
9:     else
10:        partial visibility, do recursive split
```

Thus, to improve prediction and occlusion types detection, a bounded centroidal Voronoi diagram via Lloyd relaxation [TAD10] (see Fig. 3.6) is computed to generate uniform point samples on each facet fragment of the nodes, see Fig. 3.7.

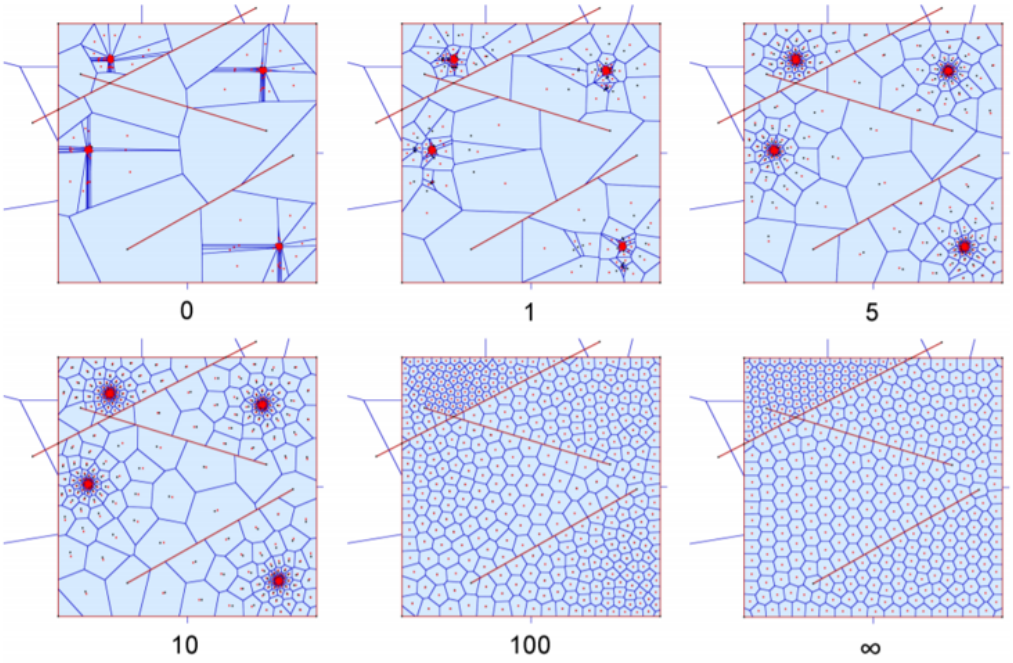


Figure 3.6: 2D centroidal Voronoi tessellation with geometric constraints from Tournois et al [TAD10]. The number of Lloyd iteration is depicted.

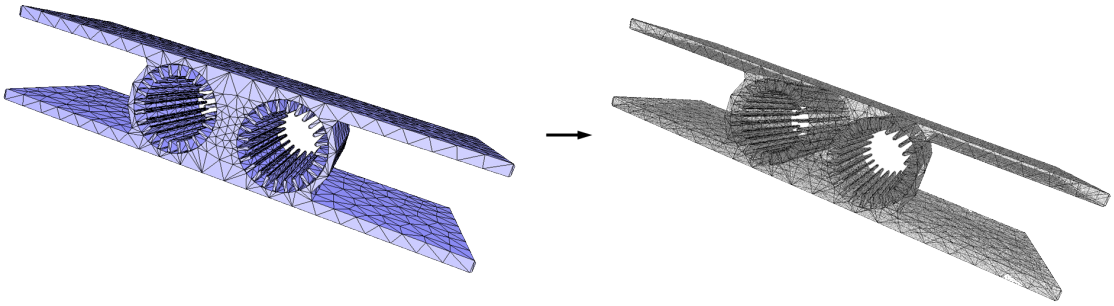


Figure 3.7: Sampling of a slice of a heat pipe on board of a spacecraft. Left: input model (9K triangles). Right: sampled faces (20M samples in total). Samples are involved in visibility detection and prediction steps.

3.2.2 Adaptive Splitting

To partition convex polygons, a common practice consists in performing longest-edge or all-edges bisection, as performed by Walton et al. in the View3D software [Wal02]. The longest-edge split tends to produce isotropic fragments while the midpoint approach better preserves the shape of the original triangles, see Fig. 3.8. For partially visible situations (penumbra areas), isotropy and shape-preserving features are both not particularly well suited to best approximate the view factors. Ideally, a method that splits the input triangles into polygons whose shapes adapt to the local penumbra, e.g. along the minimum curvature direction in the elliptic situation, would be relevant.

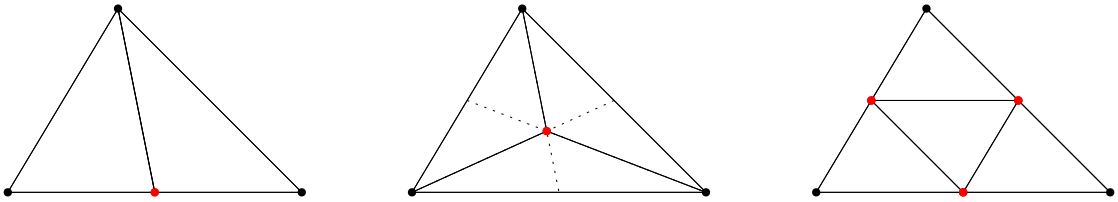


Figure 3.8: Different splitting strategies. Left: longest-edge bisection. Middle: bisectors intersection. Right: midpoints split.

Assuming the source facet (or fragment) emits light (or heat) under the hemispherical model, the obstacles cast a shadow on the target, whose shading (transitions from white to black, see Fig. 3.10, left) depends notably on the distance between the source and the target elements. In the following, a dense point-sampling of the source is considered, and line-probes are randomly selected between the source and target. Points associated to probes intersecting the obstacles are depicted in red, and blue otherwise. The objective is to find the splitting line that best separates (or classifies) the blue samples from the red samples (see Fig. 3.11). Machine learning, notably support-vector machines (SVM), come in handy to solve this problem [CL11] that involves solving a linear program, see Fig. 3.9. In particular, the signed distance function from the samples to the splitting line is maximized. If the confidence in the best separating line reveals to be low, which

means there is no privileged splitting direction, the longest edge bisection method is adopted instead.

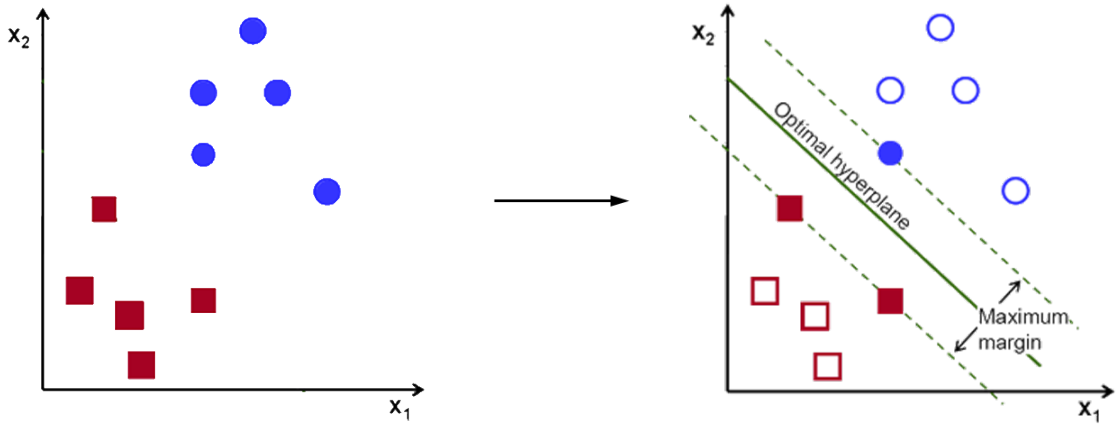


Figure 3.9: SVM classifier [BGV96]. The algorithm finds the hyperplane that maximizes the margins between samples from two classes. Left: original dataset. Right: computation of the hyperplane.

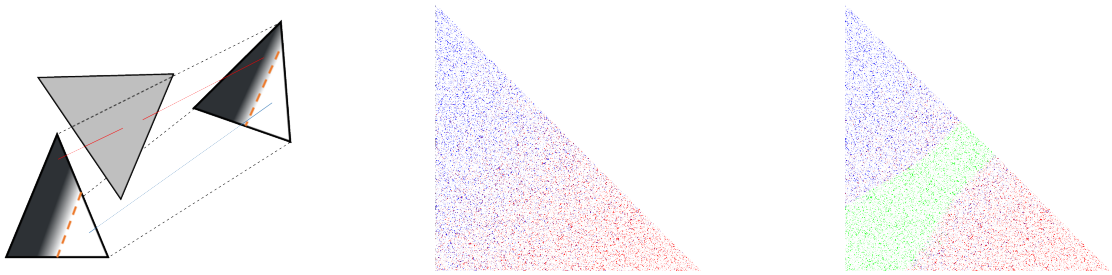


Figure 3.10: Adaptive Splitting. Left: Obstacle configuration, if a probe intersects the obstacle, associated points are colored in red, otherwise in blue. Middle: Colored points rendering for 10^6 samples. Right: Green points depict the penumbra, where every point has almost as many blue points as red points for neighbors (here with a 10% tolerance).

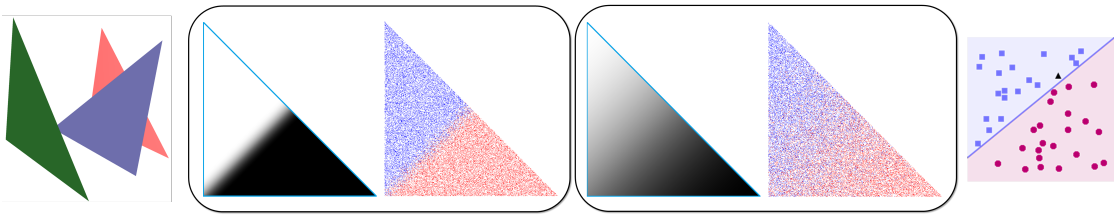


Figure 3.11: Adaptive splitting operator for partial visibility. Left: two canonical triangles and one obstacle in-between. Middle left: the obstacle is close to the source facet, the optimal split between full visibility and shadow is perceptible. Middle Right: the obstacle is equidistant from the source and target facets, the splitting operator becomes less trivial. Right: optimal splitting line.

3.2.3 Prediction

The above progressive numerical integration algorithm only sums up closed forms for pairs of facet fragments that are fully visible. Also, each partially visible case lends to recursive splitting into smaller fragments. Then, the exact geometric view factors are systematically underestimated during the recursion process. A feasible approach to diminish this underestimation is to add a prediction of the visibility ratio for all pairs of partially visible fragments at the final allowed recursion step (maximum recursion depth), and multiply this ratio by the closed form formula as if there was no visibility obstructions.

The prediction alone gives a decent approximation of the view factor, and reduces significantly the number of rays compared to Monte Carlo ray tracing methods, see Fig. 3.12, Fig. 3.13 and Fig. 3.14.

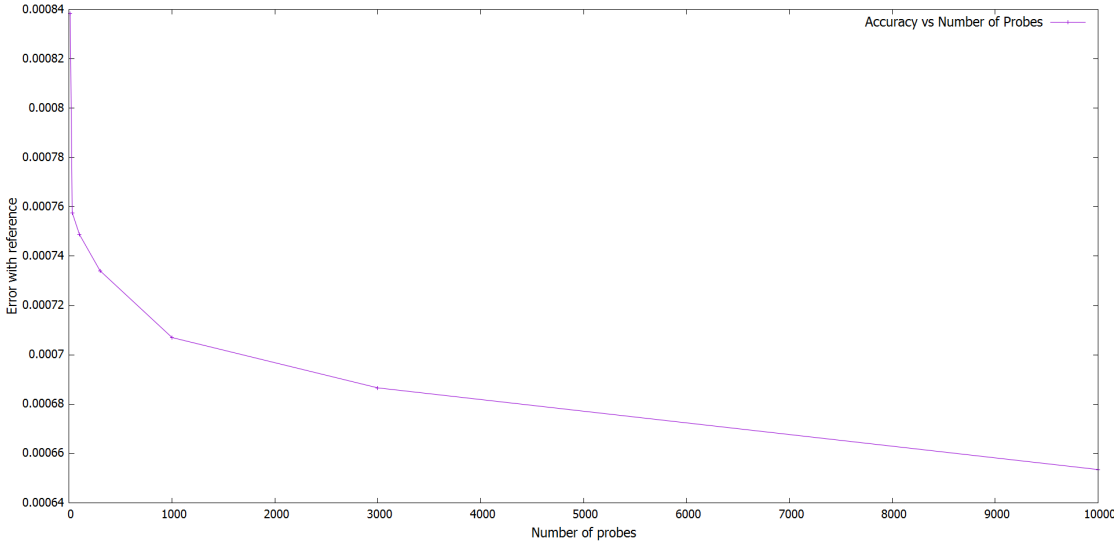


Figure 3.12: Average error against number of probes.

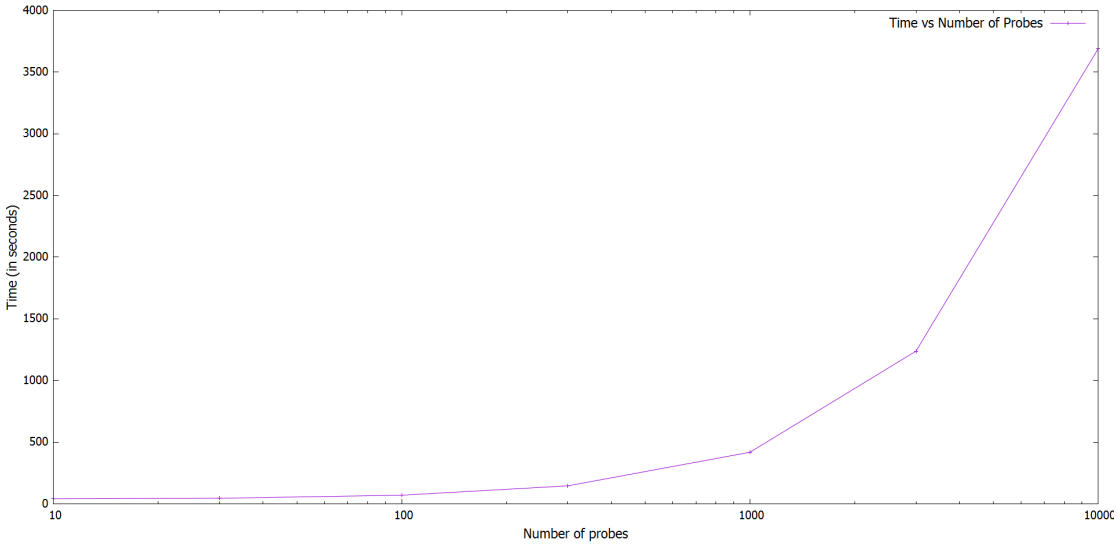


Figure 3.13: Time against number of probes.

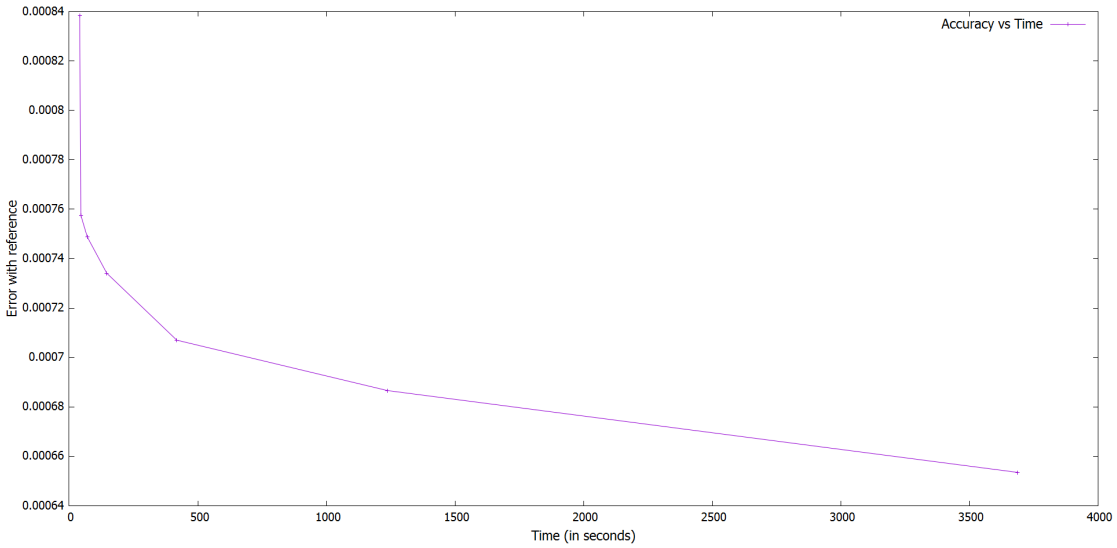


Figure 3.14: Average error against time.

For the purpose of filtering out the full occlusion and full visibility tests explained above, the prediction method detailed next leverages the line probes shooting operation. More specifically, we define the ratio r as the ratio between the number of line-segments not intersecting the obstacles over the total number of line-segments shot. Although a naive linear predictor $y = x$ (meaning r is directly picked up as visibility ratio) yields satisfying results, it should be noted that $r = 0$, respectively $r = 1$, does not always imply full occlusion, respectively full visibility. As a reminder, the full visibility or full

occlusion scenario would have been detected previously by the exact test. Alternately, an improved predictor of the visibility ratio is elaborated. Instead of a linear predictor, a degree-5 polynomial fitting regression is proceeded on ground-truth data generated offline on a series of complex models, see Fig. 3.15. In more details, ground-truth data correspond to visibility ratios derived from accurate estimates of the real geometric view factors, for all possible ratios of probes intersected over probes shot.

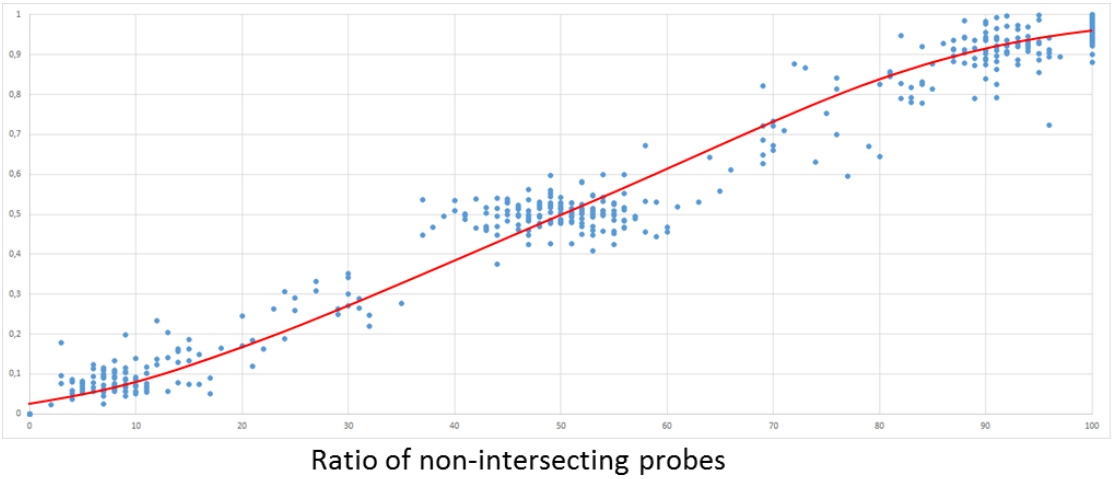


Figure 3.15: Predictor of the visibility ratio via degree-5 polynomial curve fitting.

3.3 Experiments

The method is implemented in C++ using the following libraries: The Computational Geometry Algorithms Library (CGAL) [The20], eigen [GJ+10], libsvm [CL11] and openMP [DM98].

3.3.1 Validation

First and foremost, for the sake of numerical validation, the implementation of the closed form formulation from Schröder and Hanrahan [SH93] had to be validated with several canonical configurations calculated by Ritoux [Rit82]. Then, the point-based and polygon-based quadratures have been tested by verifying proper convergence to the closed form in full visibility scenarios, and next with View3D [Wal02] in the presence of visibility obstacles. Initially, these validations have been performed on simple configurations (for instance Fig. 3.11 left), mechanical parts (see Fig. 3.16) then satellite models. For View3D, desired accuracy is tuned with either the convergence stopping criterion or the maximum recursion depth. Nevertheless, the software does not offer progressiveness as the method explained in this chapter. Also, the convergence criterion is recommended to not be lower than $1.0e^{-6}$ since many of the intermediate calculations are accurate to

single (32-bit) precision only. In our approach, most geometric operations performed rely on the accurate constructions and exact predicates offered by the CGAL library [The20].

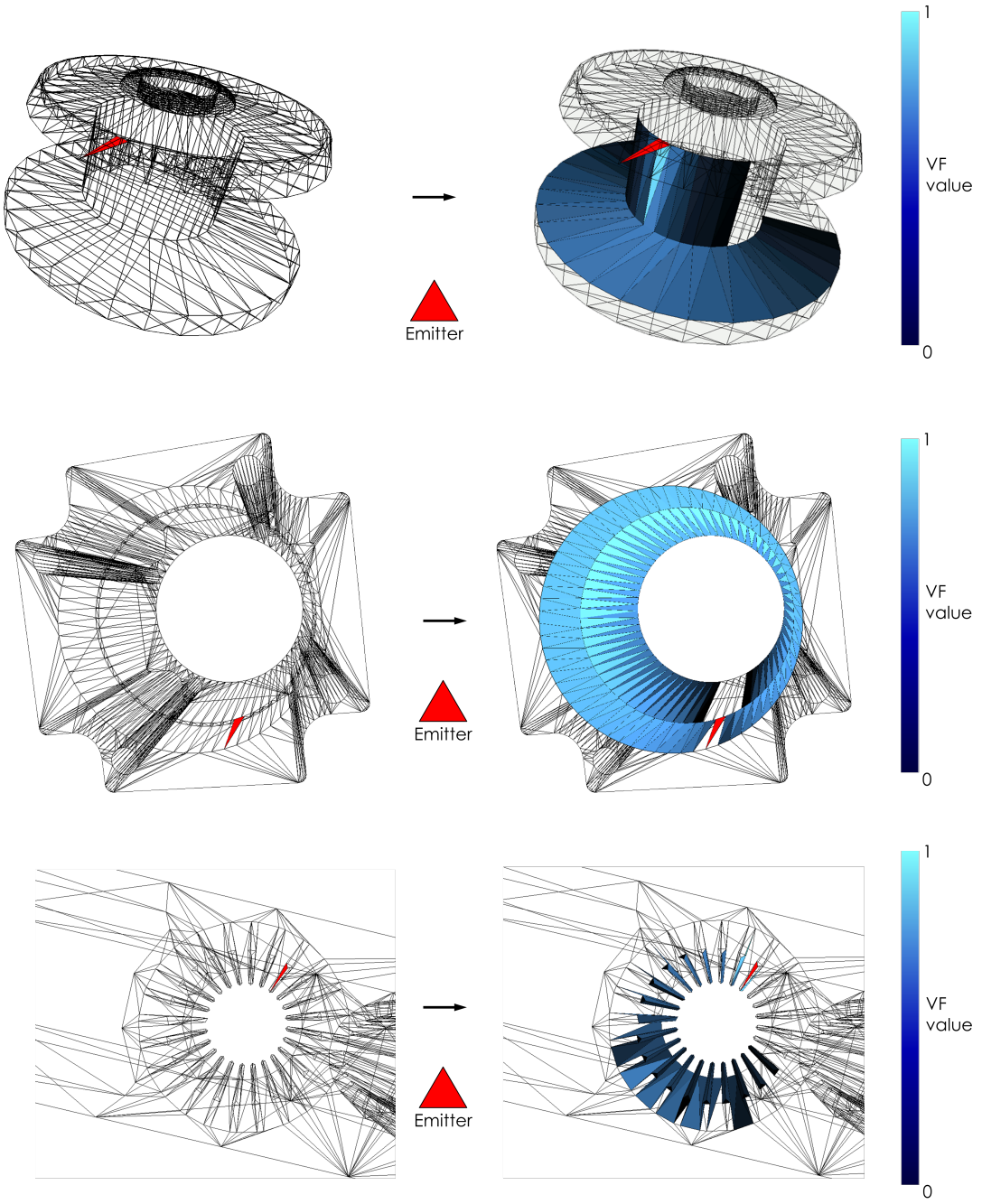


Figure 3.16: View factor computations applied to mechanical parts. Left: input models. Right: view factors rendering for a given emitter face (depicted in red).

3.3.2 Convergence Rates

Initially, a sampling obtained through bounded centroidal Voronoi diagram is compared to a random uniform sampling as shown in Fig. 3.17. Accuracy towards the real view factor value is reached with a smaller point density with the bounded centroidal Voronoi diagram compared to the random uniform sampling.

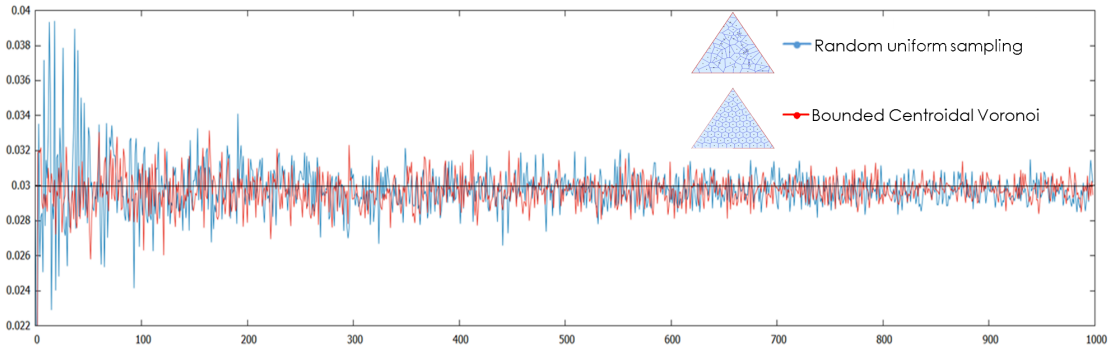


Figure 3.17: Convergence rates. We plot the accuracy against the number of quadrature points, for the random uniform sampling case (depicted in blue) and the bounded centroidal Voronoi diagram (depicted in red). No prediction is used for this case.

Then, the numerical results obtained with our method are compared with a state-of-the-art software dedicated to view factors computations, View3D [Geo], in order to validate the consistency of our results (see Fig. 3.18).

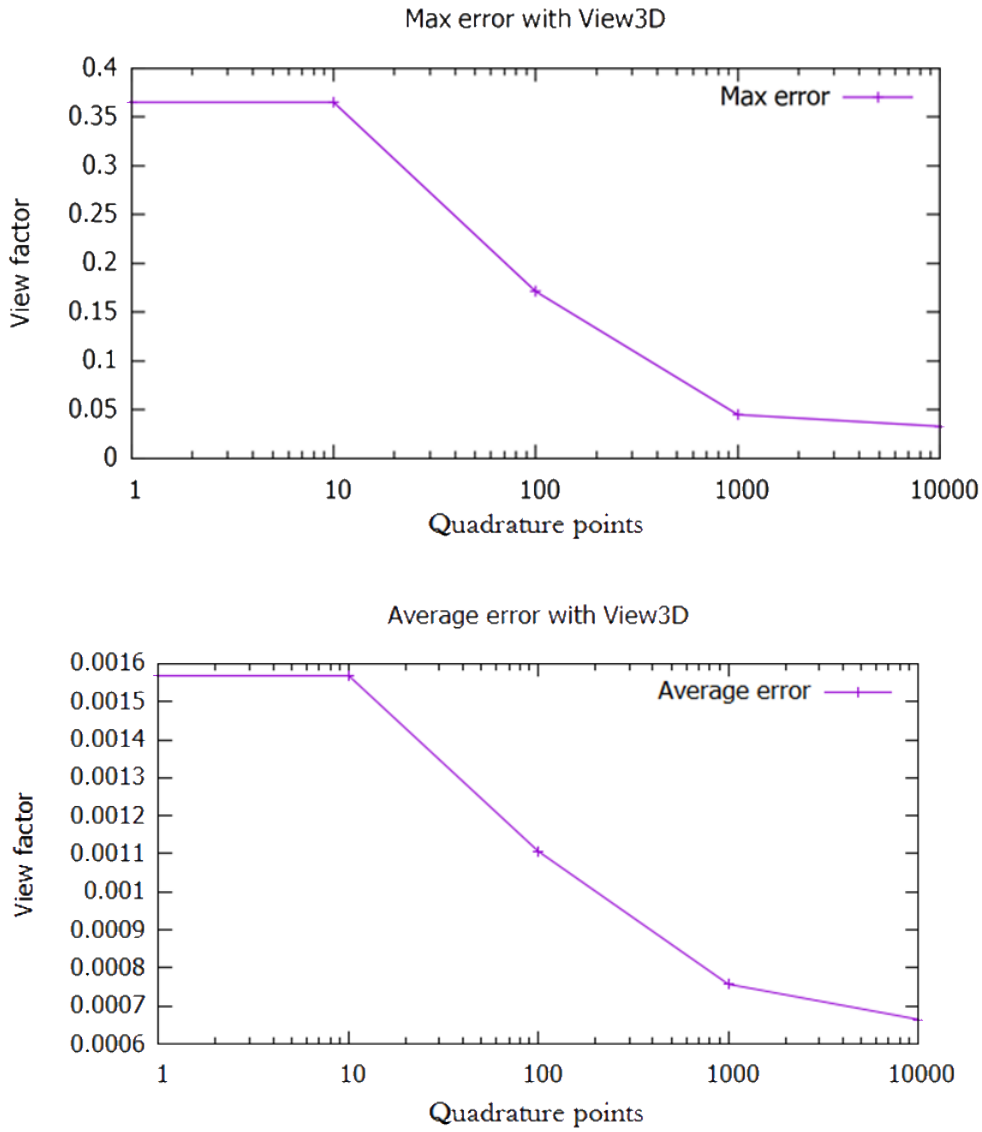


Figure 3.18: Validation with View3D. Both average and max errors of geometric view factors between all pairs of facets that are fully or partially visible are computed. Top: Max error. Bottom: Average error.

A simple polygon-based quadrature requires a substantial amount of quadrature elements to approximate the reference view factor value compared to a point-based quadrature (see Fig. 3.20), and the latter also requires a significant number of point samples to reach the reference value, even when the point sample is carefully optimized using a bounded centroidal Voronoi diagram (referred to as BVD).

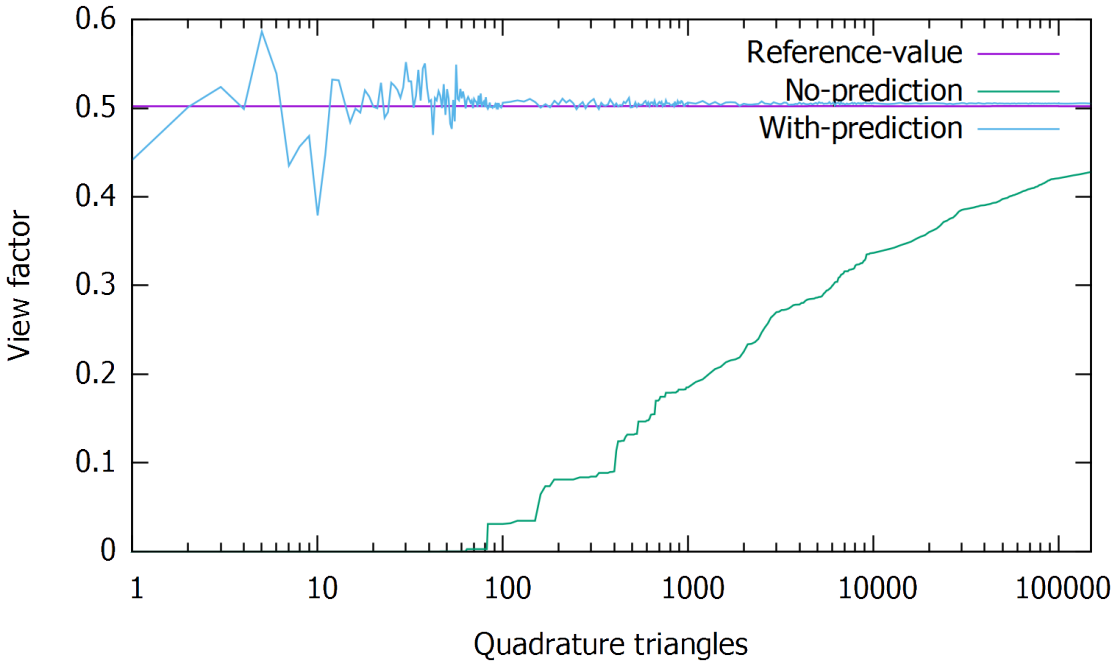


Figure 3.19: Convergence rate with and without prediction, i.e. accuracy of view factors against number of quadrature polygons, in log scale.

Yet, adding the prediction step offers a significant improvement, as highlighted by the closeup shown by Fig. 3.20 and by Fig. 3.19.

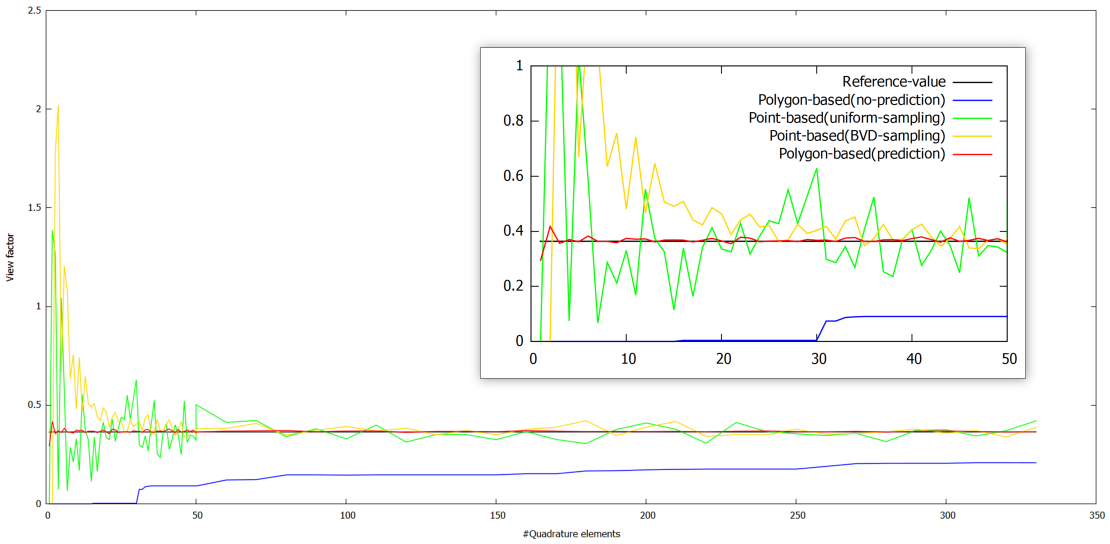


Figure 3.20: Convergence rates of the different quadrature methods. Plot of the accuracy of view factors, with respect to reference values, against the number of quadrature elements (#points or #polygon fragments elements per area unit). On the right, the closeup depicts the results between 0 and 50 quadrature elements.

3.4 Conclusion and Future Work

We proposed a progressive method for computing geometric view factors between facets of surface triangle meshes. The numerical integration approach relies on a recursive polygon-based quadrature, coupled with closed form formulae for fully visible configurations, adaptive facet splitting, line probing and prediction for partially visible configurations.

For the sake of numerical precisions and exactness, early termination tests such as detection of full visibility or full occlusions are performed. Convergence rates are improved thanks to the prediction step. However, a limitation of this approach is that it only considers linear elements as input. Currently, curved models require a preliminary meshing step (required by finite element methods), inevitably introducing an approximation error. Also, limited progressiveness for low accuracy is another limitation. The output matrix of view factors obtained through our method still requires a quadratic number of operations in the number of facets of the input mesh. An approach generating a coarser view factor output matrix in shorter time would be interesting, for instance relying on recursive facet clustering coupled with far-field approximation.

For future work, the method explained in this chapter can be adapted to curved, higher-order elements such as NURBS or Bézier surfaces. This implies devising closed-form formulae for these curve elements and approximation algorithms [FAB+18]. About the prediction step of the visibility ratio, another promising direction would consist of (1) exploiting the information collected during the descent in the tree during recursion,

and (2) leveraging modern machine learning methods for computing descriptors of the obstacles intersected by the probes. In particular, thanks to forest data structure, the nodes of the tree could record the probe ratios for their partially visible nodes, such that it is possible to perform a regression over the current probe ratio and also on its parents in the tree. Furthermore, the AABB tree not only computes intersection tests, but can return the list of intersected facets as well. Several geometric descriptors can be computed for these facets, such as the areas, aspect ratios, distance between source and target fragments, projected areas onto the bisector plane between source and target amongst others. Statistics over these quantities can be computed (average, variance, standard deviation) and can be given to a random forest regression algorithm [Wal19]. Nonetheless, calculating the descriptors might be compute-intensive, and may thus impact the convergence rates. Another stimulating approach is the recent work on view factors estimations via deep neural networks [TR21].

Chapter 4

Reducing Geometric Models

Contents

4.1 Background

This chapter focuses on the problem of reducing the geometric model of a physical object which undergoes radiative thermal simulation, while preserving a desired physical property. The approach detailed in this chapter aims at conceiving a reduced model (at the end of the model construction process) favoring real time simulation, devoted to flight operation phases. To pursue this objective, the contributed method relies upon two major elements: (1) a geometric reduction process (oblivious to physics) and (2) a simulation black box, seen as a physics oracle. The latter is requested every n^{th} step of reduction, which yields changes in the physical property of interest (here a temperature difference, also referred to as distortion). The considered case study is radiative thermal simulation at steady-state, neglecting specular behavior, performed over a complete orbit at a given day of the year. At first, an ingenuous approach is explored, which consists of performing the reduction of the geometric model until the maximum allowed temperature difference between the original and reduced model is reached (or if a given minimum number of mesh elements - vertices, edges or facets - is met). Then, an improved reduction process is detailed. The key idea of the latter is to perform a sensitivity analysis for each thermal node, so as to achieve a simulation-driven geometric reduction process. The method is first evaluated on a simple satellite geometric model with no topological defects, before evaluating it on a real-world use case as part as a project from Dorea: the Sentinel-3A satellite originating from ESA Copernicus program.

4.2 Driving Geometric Model Reduction by Numerical Simulation

4.2.1 Geometric Model Reduction

4.2.1.1 Mesh Data Structure

During the conception phase, satellites, equipments and mechanical parts are modeled via computer-aided design (CAD) tools. Hence, the resulting 3D models are represented by Bézier or NURBS surfaces. These models are then tessellated into triangles [NM95] (or quadrangles) to enable numerical simulation with finite elements. The triangulation process rarely generates oriented 2-manifold meshes as output, see Fig. 4.1.

In addition, these meshes are often non-conforming, meaning that edges and faces do not match exactly, see Fig. 4.2, left. Most of the time, a mesh repairing pre-processing is applied to the triangulated surface mesh before applying decimation algorithms [BK05, CAK12, ACK13]. To tackle this non-conforming issue, the input mesh is first triangulated and then made conforming via recursive edge bisection and facet triangulation, see Fig. 4.2, right.

In order to deal with non-manifold meshes, we implemented a mesh data structure similar to the *AIF data structure* [SG03b]. *AIF* stands for Adjacency and Incidence Framework, developed by Silva et al., and provides a data structure for manifold and non-manifold polygonal meshes. It is especially relevant for implementing mesh simplification

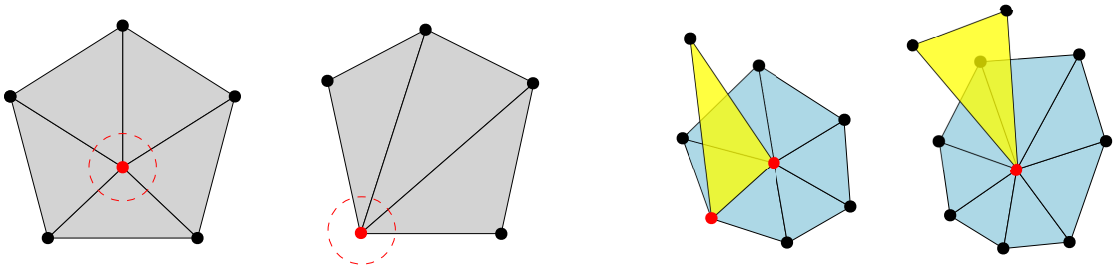


Figure 4.1: Manifold and non-manifold meshes. The mathematical definition of a 2-manifold is a topological space with the property that each point has a neighborhood homeomorphic to the Euclidean space of dimension 2. In a discrete setting, a mesh is considered manifold if each edge is incident to only one or two faces, and if the faces incident to a vertex form a single closed or open fan. Left: Manifold meshes. Right: Non-manifold meshes.

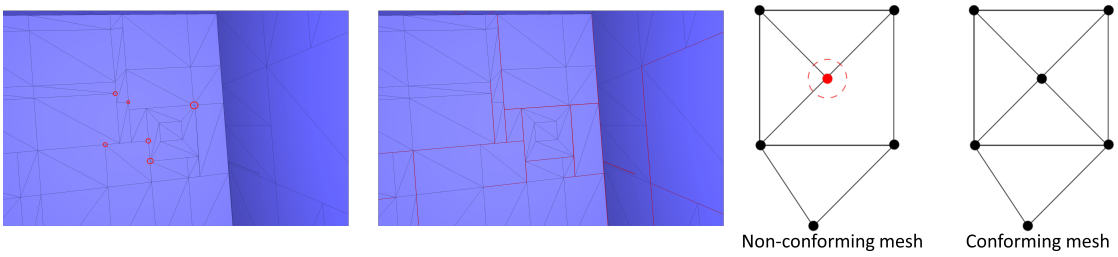


Figure 4.2: Non-conforming mesh and conforming process. Left: non-conforming area (highlighted in red). Middle: Holes between faces of a mechanical part (highlighted in red). Right: Non-conforming then conforming mesh after recursive edge bisection and conforming facet triangulation.

and refinement algorithms. Adjacency and incidence information between vertices, edges and faces of a mesh are retrieved via a minimal number of accesses. Each vertex contains the list of incident edges, each edge stores the associated incident faces and each face holds the corresponding frontier edges (see Fig. 4.3). The AIF data structure also deals with orientable facets, making it even more relevant for radiosity applications.

For simplification we considered two decimation-based approaches based on the edge collapse operator (see Fig. 4.4, Fig. 4.5 and Fig. 4.6). First, The Quadric Error Metric (QEM) from Garland and Heckbert [GH97] defines the error as the sum of squared distances to a set of supporting planes of the input mesh facets, represented by one quadric per vertex. This approach is memory-aware in the sense that each edge collapse operator sums up the quadrics associated to the two vertices of the said edge. Second, the volume-preserving approach from Lindstrom and Turk [LT98] aims at locally preserving the volume swept by each edge collapse operator, in a memoryless fashion.

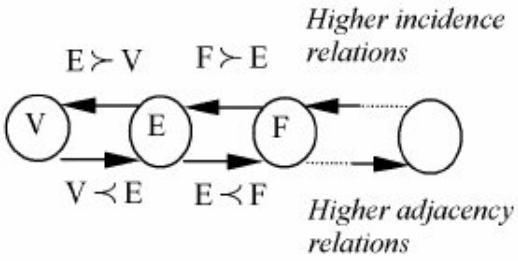


Figure 4.3: AIF diagram from Silva et al [SG03a]. $a \prec b$ means a is adjacent to b, and equivalently $b \succ a$ stands for b is incident on a.

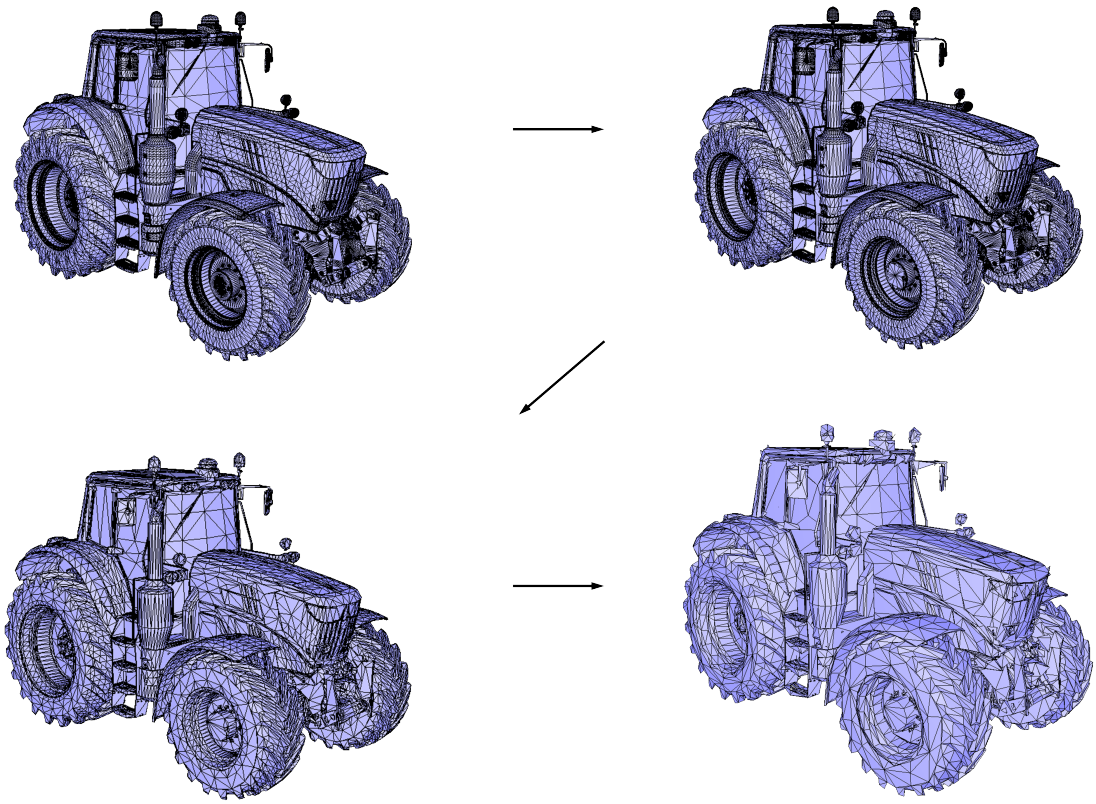


Figure 4.4: Geometric reduction applied to a mesh. From left to right: input mesh, 25%, 50%, and 75% reduction.

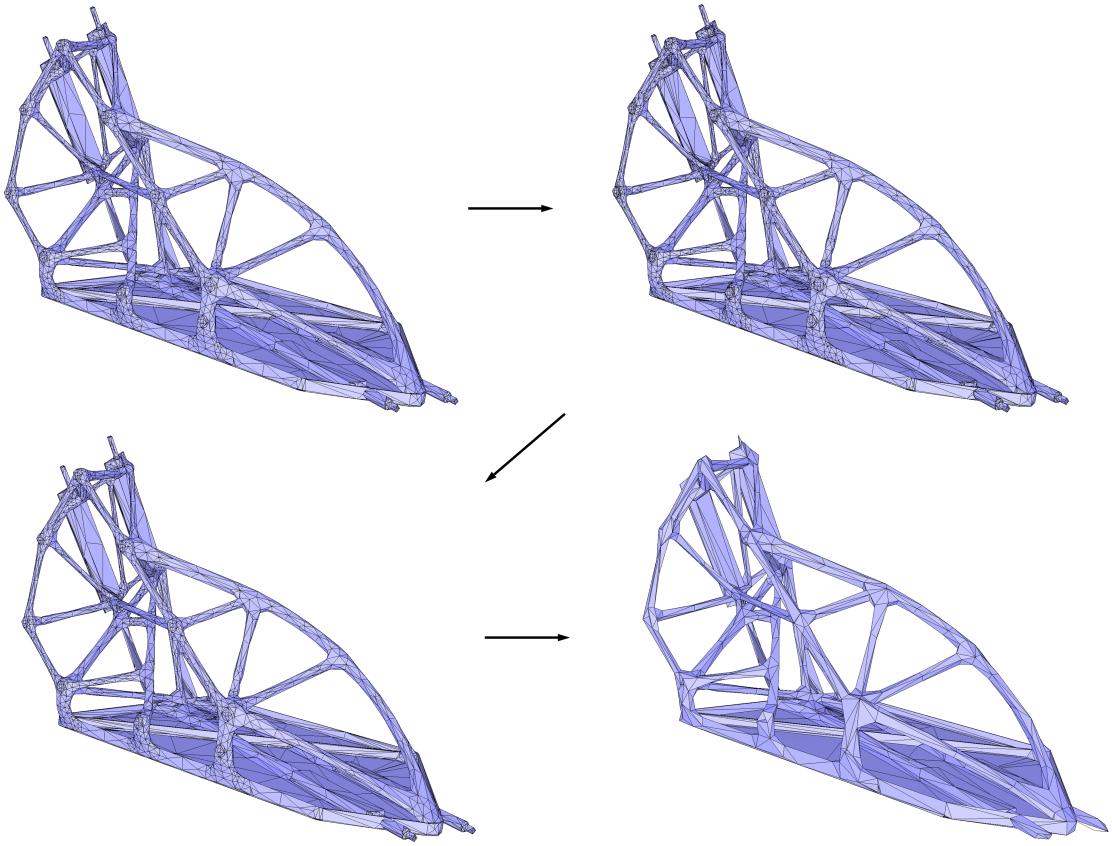


Figure 4.5: Geometric reduction applied to a free-form mesh. From left to right: input mesh, 25%, 50% and 75% reduction.

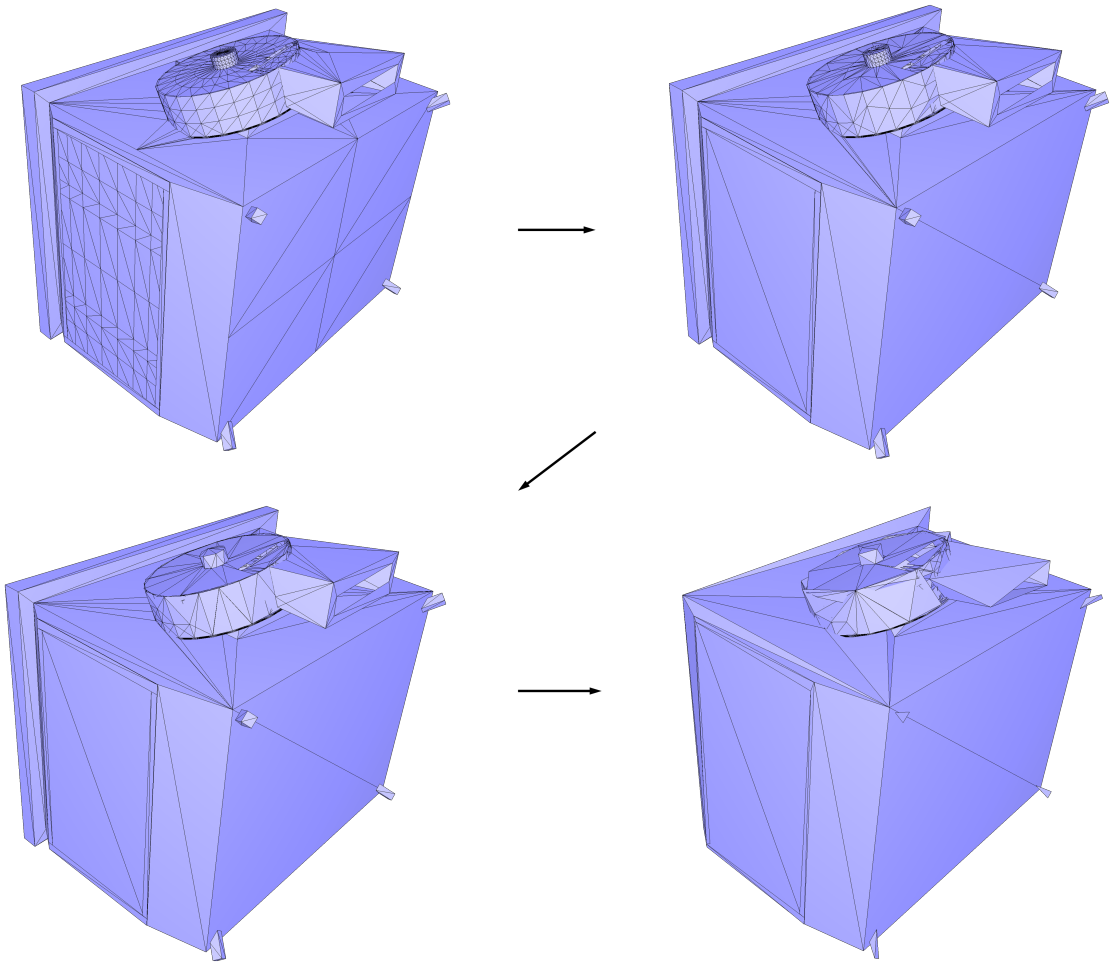


Figure 4.6: Mesh decimation applied to a satellite part, where planarity is preserved. From left to right: input mesh, 25%, 50% and 75% reduction.

4.2.1.2 Extracting the External Geometry

Radiative thermal simulation of spacecrafts is only carried out on the external geometry. Hence, to avoid superfluous computations, the external geometry of the input model is first extracted. State-of-the-art solutions to this problem rely upon ray shooting and intersection operations to retrieve the visible geometry from a given viewpoint for a $2D$ or $3D$ model [ZY13, CW15]. Radiative heat transfer can also be directly computed through GPU implementations [SP19]. Inspired by previous work, we contribute an algorithm which bears similarity with GPU-based methods (leveraging the Z-buffer), see Fig. 4.7. More specifically, multiple camera positions located outside of the geometric model and oriented towards it are set in the $3D$ space. Then, renderings of the external faces that are oriented towards the camera are generated, see Fig. 4.8. Hence, when both camera and faces mutually see each other (i.e., the scalar product of their normals is positive),

the visible faces are colored in different levels of gray depending on their indices (each face is then colored with a specific level of gray determined by its corresponding index), while the others are not rendered. By looping over all defined camera viewpoints, the colored faces are collected to recover only the full external geometry.

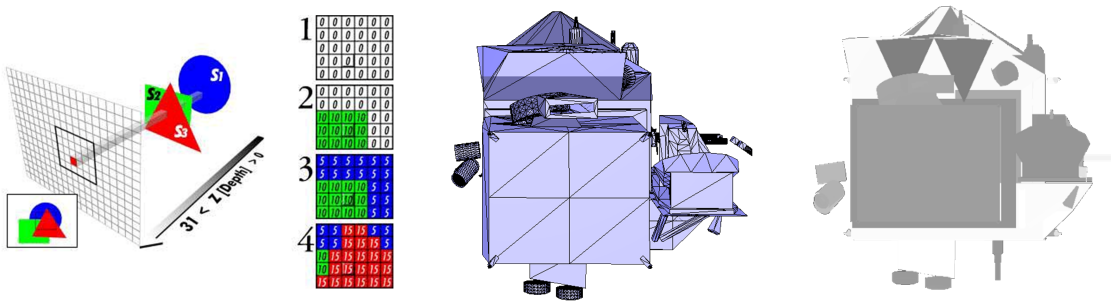


Figure 4.7: Extracting the external geometry of a satellite. Left: z-buffer principle. Middle: mesh view. Right: GPU-rendered view.

The external geometry extraction process detailed above can be extended to the task of reducing the computations time for received solar powers of a spacecraft. The complexity of such computations lies into the definition of the projection, which depends on the size of the considered model. It is observed that the number of faces has minor impact on the complexity of the calculations, explained by the number of pixels per area unit dedicated to recognizing the external faces. Furthermore, it is important to ignore faces smaller than a pixel to avoid introducing holes or obtaining missing parts of for external geometry.

Figure 4.8: External geometry rendering along different viewpoints. Detected faces are colored depending on their corresponding indices.

For a model with $100,000$ triangles measuring 6m x 3m, the previous method enables performing the calculation in 5 seconds for each viewpoint. The considered spacecraft

model is subject to rotations along the x, y and z axes. For this model, the full computation elapses 36 minutes and returns $40,000$ external triangles in the end, yielding a 60% reduction in the number of faces.

For this case study, less than 50 rotations along a given axis are sufficient to gather the visible faces from this axis, as illustrated by Fig. 4.9.

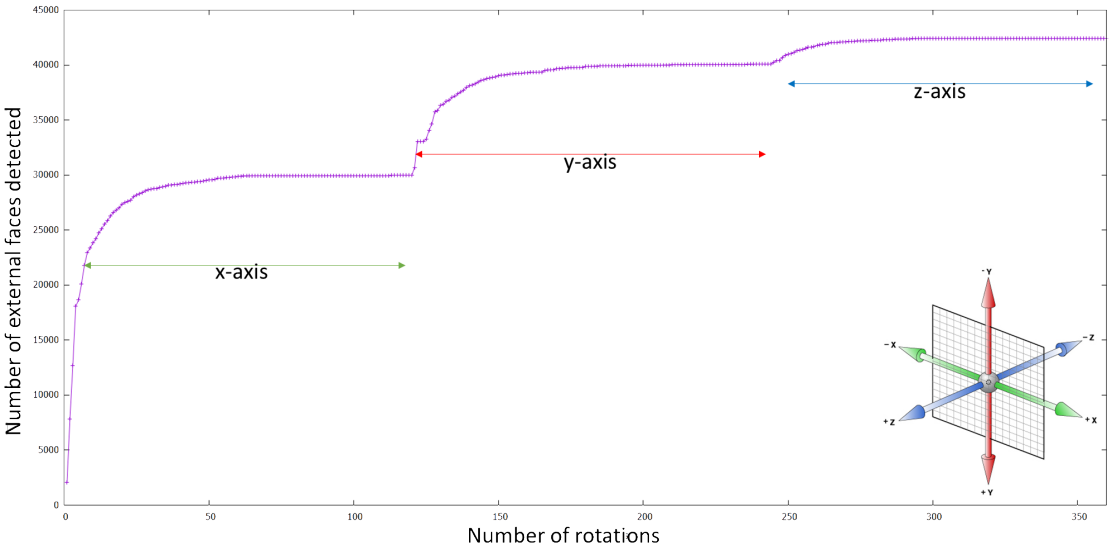


Figure 4.9: Number of detected faces depending on the number of rotations applied. The changes of axis are clearly visible.

Another application of this algorithm consists in computing incident powers emitted from a radiative heat source. By definition, the incident power received by a face is the percentage of illumination received from an energy emitter (such as the sun flux or the albedo flux reflected from the Earth). The latter depends on three components, being (1) the angle between the considered face and the source, (2) the visibility occlusions formed by the faces between the source and the considered face, as well as (3) the direct reflection of the illumination coming from the other faces (potentially not directly illuminated).

The number of pixels is sufficient to compute the illumination ratio, by dividing the number of pixels captured onto the projected screen by the area of the face, which can be written as: $\frac{pixel\ size * number\ of\ pixels}{area\ unit}$. However, computing the reflections illumination contributions requires calculating the view factors, thus representing the ratio of illumination of the faces not directly illuminated.

4.2.1.3 Preserving boundaries

Depending on the considered numerical simulation, the original geometric model can be split into sub-models, also referred to as clusters, to perform the computations. As for radiative thermal simulation, the geometric model is partitioned into thermal nodes

(grouping of faces for which temperatures are considered similar) through nodal breakdown. To preserve this constraint, an approach consisting in reducing each sub-model apart from each others is studied. All the reduced sub-models are then reassembled to yield the global reduced model. However, it is mandatory to preserve connectivity between thermal nodes. As a matter of fact, creating holes in the satellite outer envelope during the reduction process must be avoided since unwanted internal radiative exchanges coming from the outside would appear. For this study, two reduction approaches have been implemented, either (1) by reducing each sub-model separately and reassembling them, or (2) by directly reducing the full model by flagging faces with their corresponding thermal node ID. The first approach guarantees the same decimation ratio for every thermal node, at the risk of deteriorating the connectivity between different thermal nodes. The second approach preserves the connectivity between thermal nodes, see Fig. 4.11. However, and for important reduction ratios, thermal nodes might disappear during the reduction process, see Fig. 4.10.

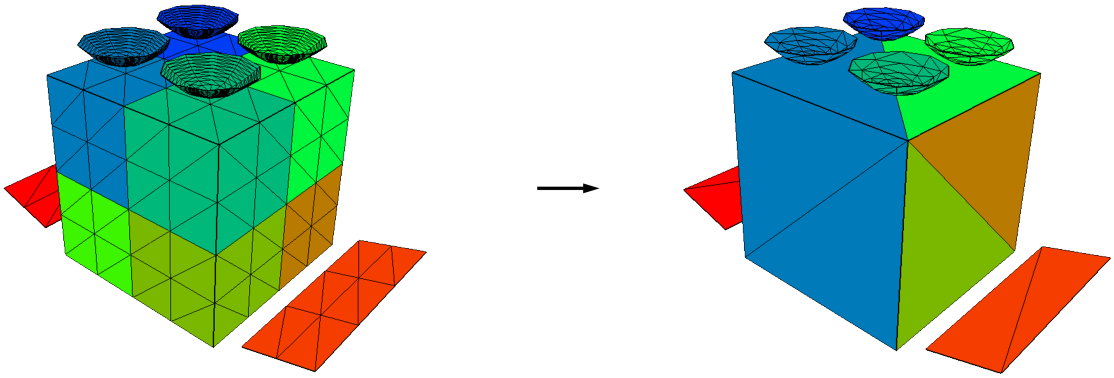


Figure 4.10: Global geometric reduction without clusters information. The number of clusters is not preserved during the reduction process. Left: Original model, 1000 faces, 10 clusters. Right: Reduced model, 150 faces, 8 clusters.

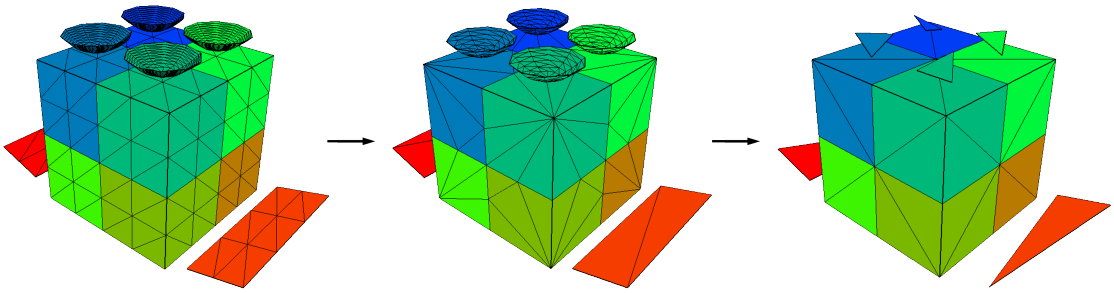


Figure 4.11: Global geometric reduction preserving edges belonging to at least two different clusters. Left: Original model, 1000 faces. Middle: Half reduction, 500 faces. Right: Maximum reduction, 55 faces

4.2.2 Physics-Unaware Reduction

In this section, a physics-unaware approach relying on reducing uniformly each thermal node independently is detailed. The considered use case for this approach, depicted by Fig. 4.12, is a simple geometric model of a satellite with 1K facets. The performed simulation is based on steady-state node-to-node radiative couplings, and no conductive simulation is proceeded. This simulation thus represents a pure radiative scenario. However, since preserving the topology and surfaces of the thermal nodes is possible, conduction effects remain valid after reduction. Regarding material properties, the latter are considered uniform for all facets.

The algorithm proceeds in 3 main stages. (1) First, if required by the considered numerical simulation, the input model is partitioned into clusters of faces, corresponding to thermal nodes in our setting. This first step is referred to as clustering, the output being files containing thermal nodes information (geometric elements and material properties), resulting in one file per thermal node. Note that clustering mainly depends on the position of a spacecraft on its orbit and its topology. In particular, when opposite sides of the spacecraft can be considered independent, view surfaces can be used to improve the clustering, provided that there is no shading. (2) The second step, referred to as geometric reduction, consists in reducing every thermal node according to a global reduction ratio defined by the user. However, if a thermal node is reduced to a single facet, a reduction deficit is applied to the next thermal node to reach the globally defined reduction ratio. (3) The third and last step relies on reassembling the reduced geometries of the thermal nodes into the full approximated model. Then, numerical simulation is performed and the numerical differences between input model and reduced models are returned for every thermal node. These three stages are repeated until the maximal allowed distortion tolerance is reached, or until the geometric model can no longer be reduced.

We utilize the volume-preserving error metric [LT98]. This error metric is relevant for radiative thermal simulation since the latter is governed by a purely geometric quantity: the view factors. View factors depend on three components: (1) the areas of the faces, (2) their orientations and (3) their distances. As the volume can be expressed as the product of an area by a distance, two factors are already met. This approach results in reaching the maximum allowed distortion quite rapidly and approximating thermal nodes too heavily while others could still be reduced, see Fig. 4.12. This suggests exploring an alternative approach informed by the numerical simulation.

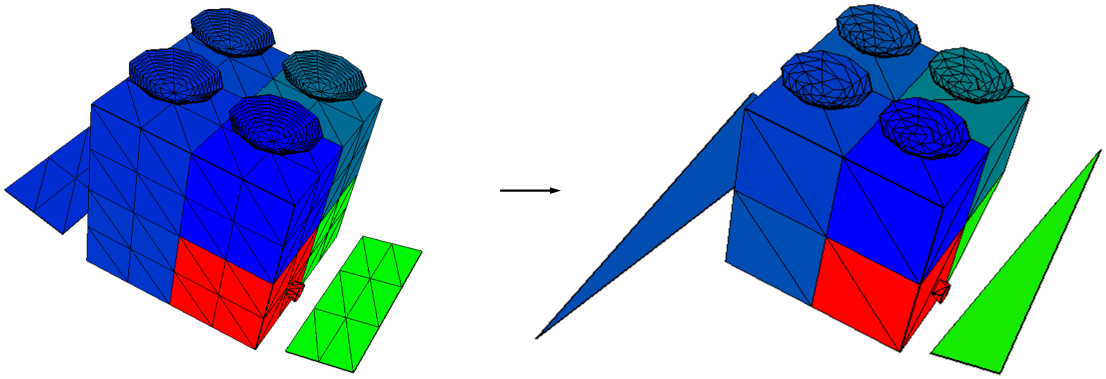


Figure 4.12: Physics-unaware mesh reduction. The maximum distortion tolerance (0.5 degrees) is reached when solar panels are heavily deteriorated, while the antennas can still be reduced (from 1006 to 370 facets).

4.2.3 Physics-Informed Reduction

In this section, a geometric model reduction informed by the numerical simulation results is detailed. First, sensitivity analysis is performed as a pre-processing step to highlight thermal nodes' (see Fig. 4.13) sensitiveness with respect to reduction. These results are then utilized to render the decimation process physics-informed. Furthermore, the initial geometric cost of an edge collapse operator is multiplied with a factor referred to as *importance*, directly computed from the returned simulated errors. We then utilize the Quadric Error Metric mesh decimation method [GH97] to weight the quadric matrices by the importance factor and benefit from the memory-aware feature provided by such method.

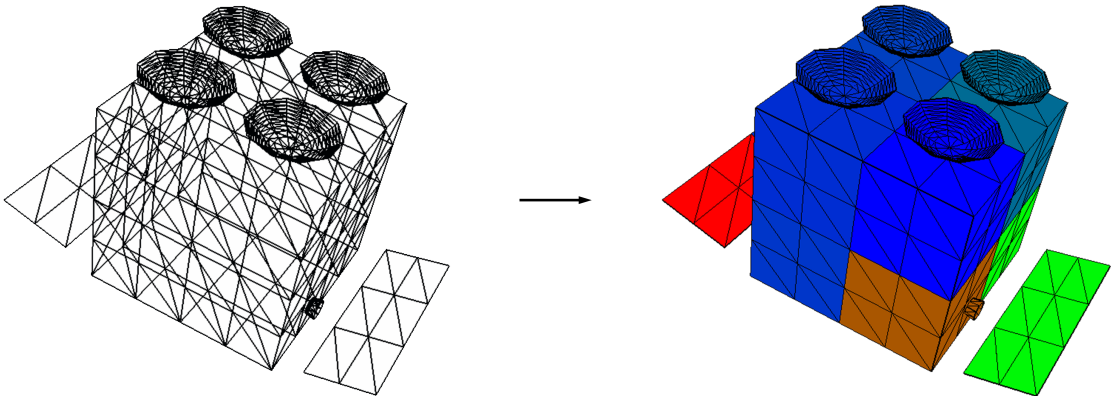


Figure 4.13: Nodal breakdown (i.e., clustering) applied to radiative thermal simulation. Left: geometric input model. Right: nodal breakdown and simulation output. The thermal nodes are depicted.

4.2.3.1 Sensitivity Analysis

In order to achieve a physics-informed geometric model reduction, the proposed approach performs a sensitivity analysis before undertaking the reduction algorithm. The aim of sensitivity analysis is to highlight the input variables of a model which mostly contribute to a quantity of interest [IL14], allowing notably a better understanding of the studied model and helping in designing an accurate reduced model. These techniques are commonly used in fluid mechanics [Duv15] and in shape optimization regarding CAD input parameters [Leb17]. Regarding the heat transfer community, sensitivity analysis is commonly applied to compute radiative heat transfer on specific structures [MV87, Ust02, MB19]. In this chapter, the considered sensitivity analysis is defined by a mesh reduction ratio step applied to each thermal node of the model (see Fig. 4.14), then the numerical simulation is performed and temperature distortions for the whole model are returned, the other thermal nodes being unchanged. This reduction process is resumed until a maximum temperature difference is reached. Temperature differences against reduction ratio are then plotted, enabling to identify which parts of the satellite are sensitive to mesh reduction (see Fig. 4.15).

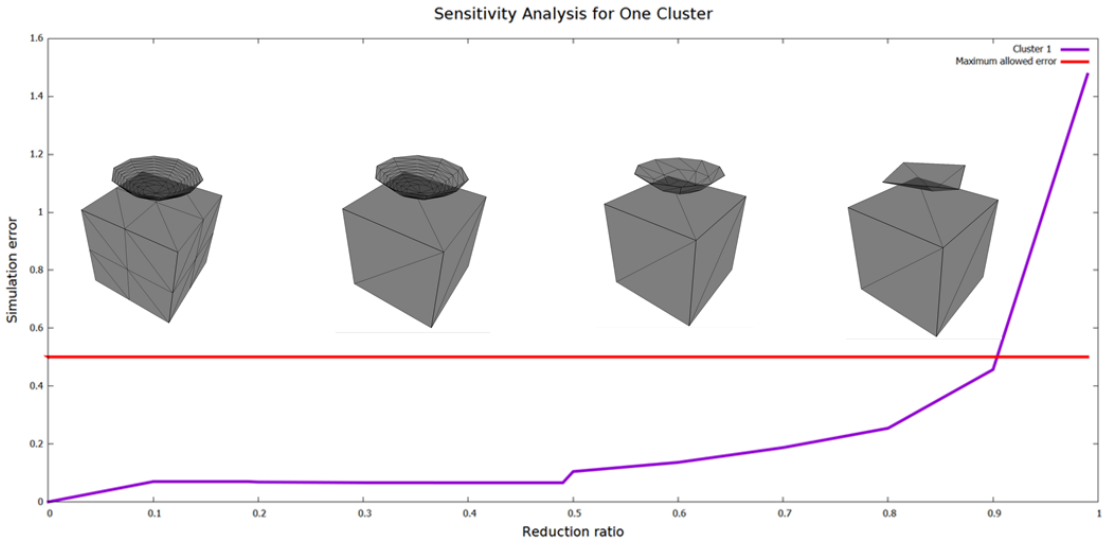


Figure 4.14: Sensitivity analysis for one cluster of the considered model (reduced geometric outputs are depicted).

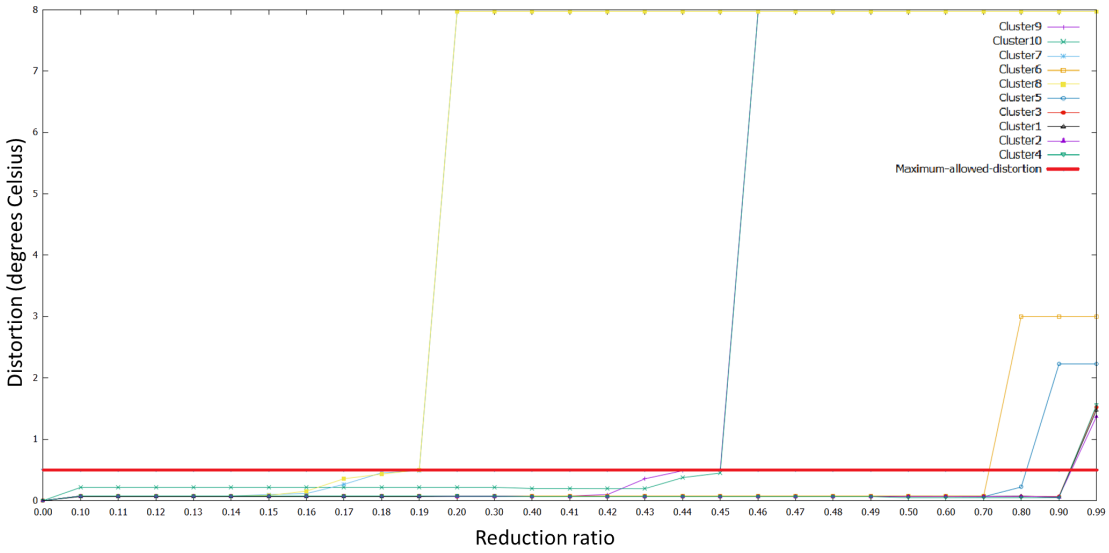


Figure 4.15: Sensitivity analysis for each thermal node of the considered model (thermal nodes are sorted in the caption by size from smallest to largest). Some thermal nodes quickly reach the maximum temperature difference while others can be further reduced without reaching it. The large difference jumps appear when a thermal node's shape is heavily deteriorated, no longer preserving the original view surfaces of the thermal node before decimation.

4.2.3.2 Prediction

For the purpose of prediction, the sensitive analysis curves obtained previously are now requested by choosing a maximum distortion target, then the feasible mesh reduction ratio for each cluster is deduced. It can be noticed that the prediction is faithful, except for large differences where the temperature error is underestimated, see Fig. 4.16. This phenomenon is explained by the fact that the sensitivity analysis is performed on each thermal node in isolation, not taking into account the potential coupling between thermal nodes from the radiative simulation point of view.

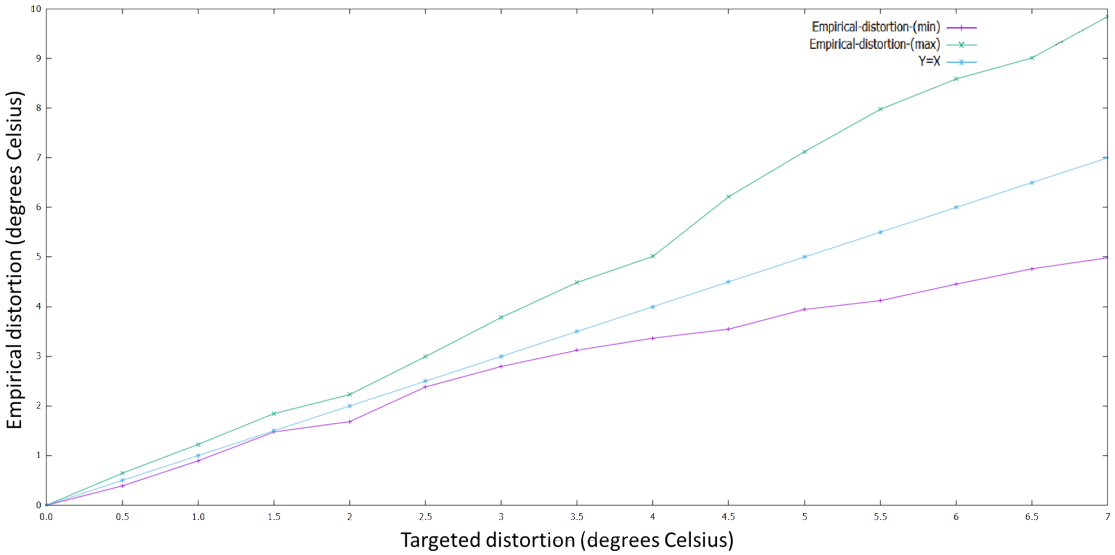


Figure 4.16: Real temperature difference against target temperature difference (in degrees Celsius). Two plots are shown: the empirical difference right before exceeding the targeted difference (below the Y=X straight line) and right after exceeding it (above the Y=X straight line).

The results of such an approach are depicted Fig. 4.17 and Fig. 4.18, where thermal simulation is conducted on a model reduced with prediction. The maximum temperature difference compared to the original model is 0.0449 Celsius degrees with temperatures ranging from -51.24 to $+53.61$ Celsius degrees.

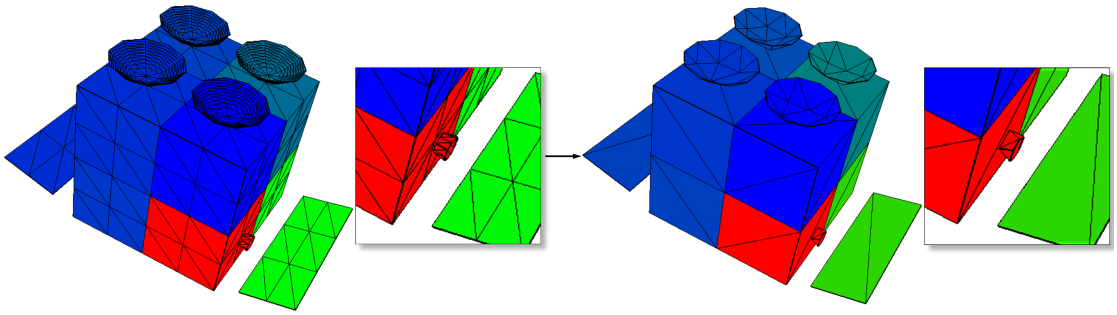


Figure 4.17: Thermal simulation applied to a reduced model with prediction (from 1006 to 136 facets, i.e. a 90% reduction ratio).

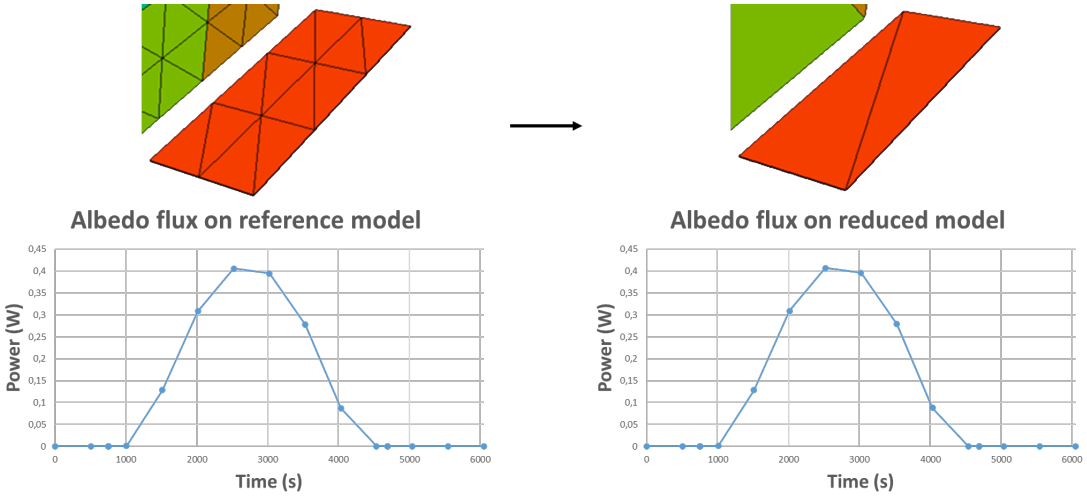


Figure 4.18: Albedo fluxes before and after reduction for the thermal node represented by one of the solar panel. Reduction from 12 faces to 2 faces, with a maximum error of 0.00114455W.

4.2.3.3 Importance Deduced from Numerical Simulation

An improvement of the physics-aware geometric reduction involves modifying the cost in the priority queue containing the edges of the mesh. The idea is to process a full reduction with simulation step and collect in the end the distortions for every cluster. A factor referred to as *importance* is then computed by linearizing the collected distortions on an interval, e.g. $[0, 2]$. If the returned distortion for a cluster is lower than half of the desired distortion, then the importance factor is set between 0 and 1 strictly, otherwise between 1 and 2, see Fig. 4.19. A new mesh decimation is then proceeded by multiplying each initial quadric of the mesh with the newly obtained importance factor, thus providing a simulation-derived information to the geometric reduction algorithm.

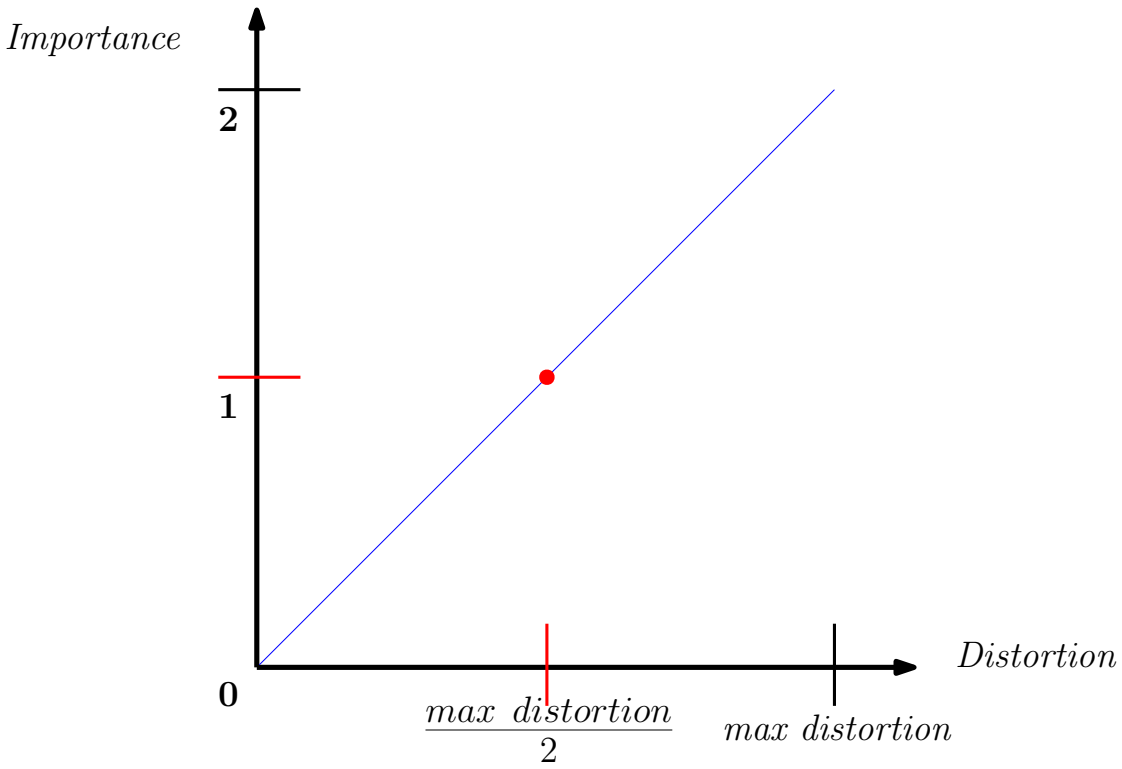
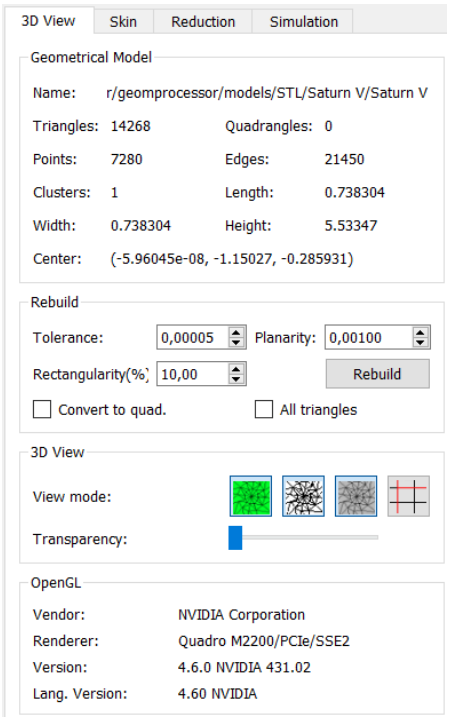


Figure 4.19: Importance depending on the returned distortion from numerical simulation.

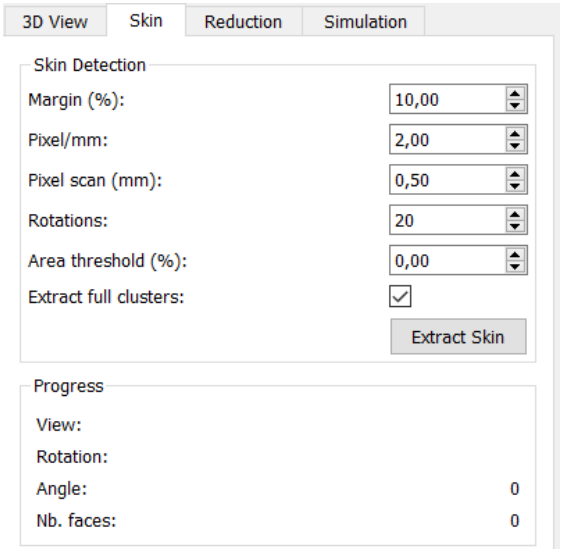
4.3 Industrial Use Case

4.3.1 Industrialization of the Reduction Tool

The early results of the reduction process detailed previously allowed DOREA to work on the development of a real-time thermal digital twin conditioned on a sufficiently small model and faithful to the detailed model, funded by ESA & CNES in collaboration with Thales Alenia Space. An industrialization of the initial prototype had to be performed for such an important task. The first step consisted in porting and cleaning the code into a standalone software to improve its maintainability. Then new essential features have been added such as import and export with file formats used by thermal engineers such as STEP-TAS (ESA), DAF files (Thales Alenia Space), as well as standard 3D model formats like .OBJ, .OFF, .STL, .GEO, .IGES and .STEP files. A graphic user interface has been designed to facilitate the use of the different tools and features, see Fig. 4.20a, Fig. 4.20b, Fig. 4.21a, and Fig. 4.21b (see appendixB for a global visualization).

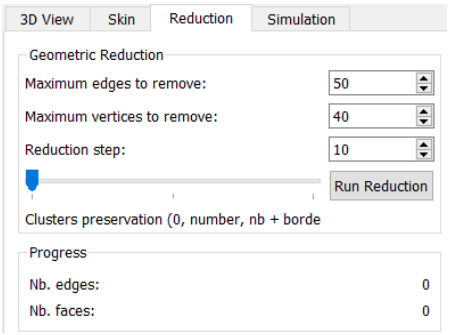


(a) 3D view tab displaying geometric attributes and a rebuild option for given tolerance, planarity and rectangularity inputs.

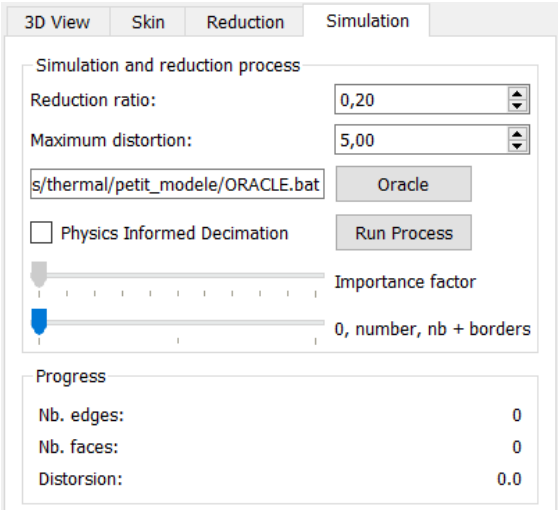


(b) Skin extraction tab used to retrieve the external geometry of a model.

Figure 4.20: First two tabs of the reduction software.



(a) Reduction tab performing a purely geometric model reduction (preserving clusters, borders or none of these).



(b) Simulation tab performing a reduction and simulation process coupled to a given numerical oracle.

Figure 4.21: Last two tabs of the reduction software.

4.3.2 Sentinel-3A use case

Initial experiments are first proceeded on a trivial satellite for implementation and debugging purposes. A real use case is then considered: the Sentinel-3 Earth observation's satellite.

The Sentinel satellites are a family of ESA's Earth observation satellites (from Sentinel-1 to Sentinel-6) and on-board instruments intended to replace the ENVISAT2 satellite, and constitute the space component of the Copernicus program [FFFP16] of the European Union, see Fig. 4.22.

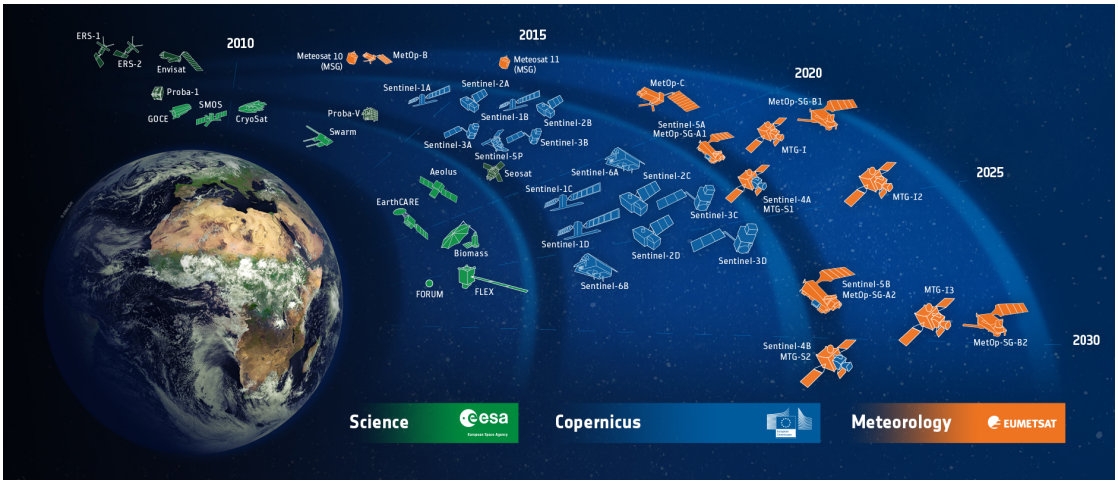


Figure 4.22: Past, operational and planned Earth observation satellites contributing to Copernicus program. The satellites are operated by the European Space Agency and EUMETSAT. Credits to ESA.

Notably, the Sentinel-3 (see Fig. 4.23 and Fig. 4.24) must in particular measure quantities characterizing the dynamic processes affecting the oceans: ocean wave height, wind speed, ocean temperature and color, reflection of the photosynthesis process.

In this context, as part of the rationalization of thermal simulators for operations project between Dorea/TAS and ESA/CNES for the development of a thermal digital twin, geometric reduced models accurate to numerical simulation of the Sentinel-3A satellite are explored.

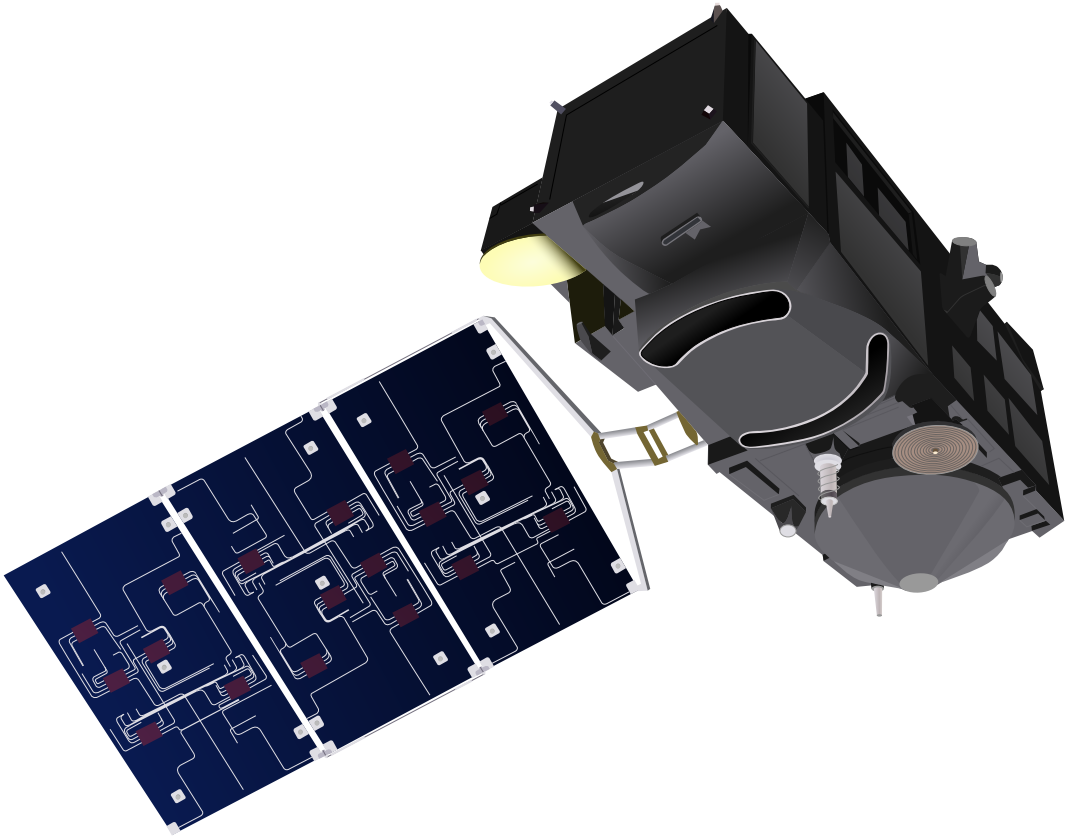


Figure 4.23: European Space Agency Sentinel-3 satellite. Credits to SkywalkerPL.

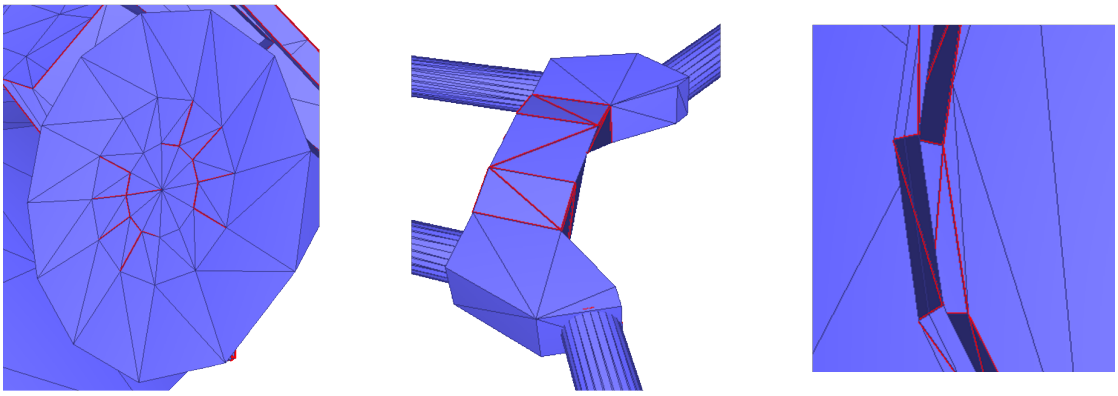


Figure 4.24: Connectivity and Orientation defects on Sentinel-3A tesselated geometric model. Holes between elements are depicted in red, orientations are depicted in light blue for the positive side and dark blue for negative side.

Since Sentinel-3A satellite has a low Earth orbit, solar flux as well as Earth and albedo fluxes have to be considered. A closeup of the reduced model can be seen Fig. 4.25 and its associated graph Fig. 4.26. Then two approaches can be adopted, either coupling the reduction process with the complete radiative simulation to obtain a reduced model accurate to solar, Earth and albedo fluxes, or running the simulations independently and obtain one reduced model for each of them.

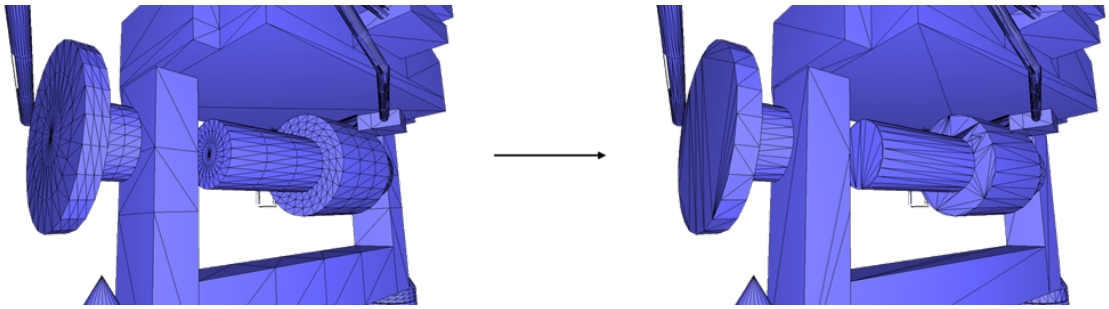


Figure 4.25: Before and after reduction of the solar panel of Sentinel-3A, closeup of one of the rotation axis, from 6k to 3k5 triangles.

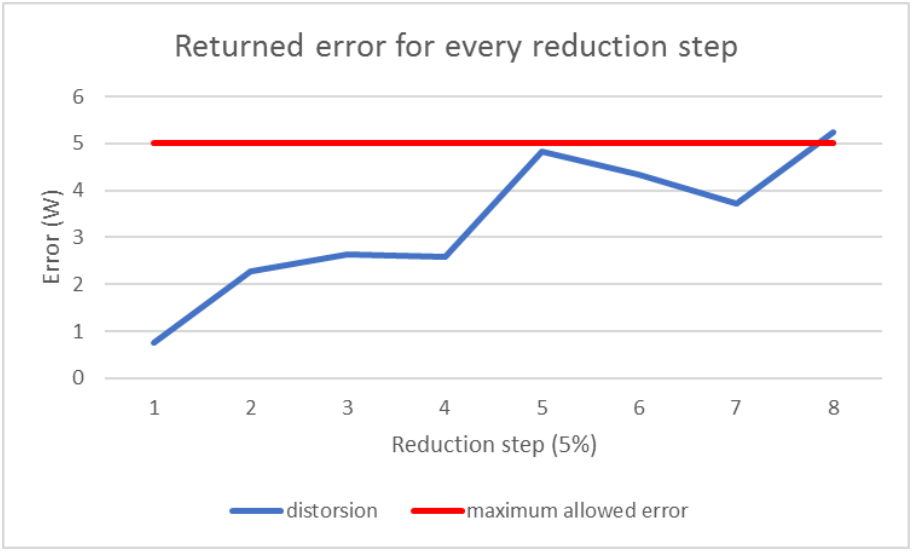


Figure 4.26: Maximum returned error in watts (W) for every decimation step. For this model, powers range from 0 to 140W and the maximum allowed error is reached for a thermal node whose original received power is 59.89W whereas the same node of the reduced model has 54.63W.

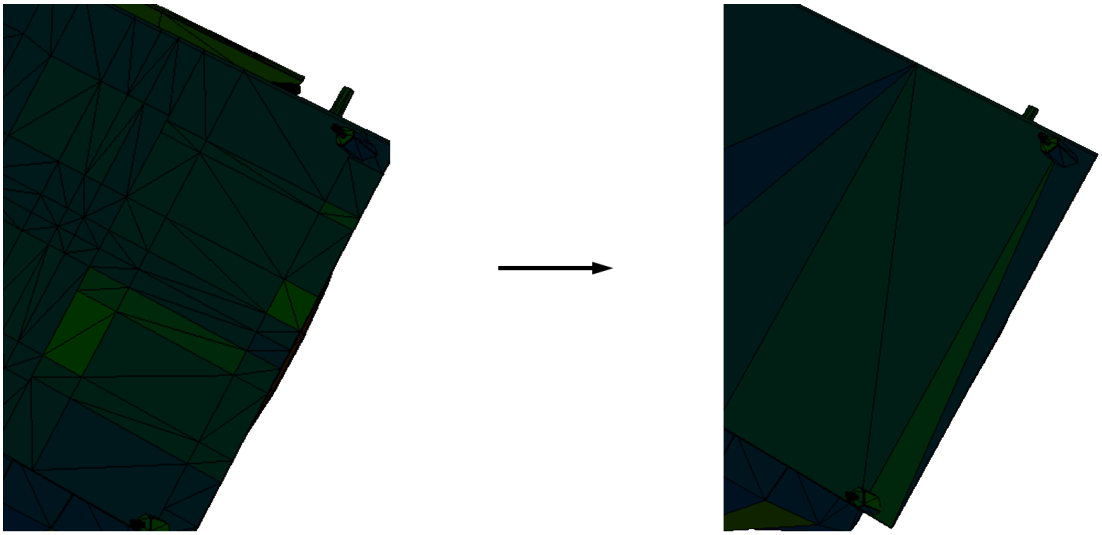


Figure 4.27: Closeup of a part hidden from direct fluxes before and after reduction with simulation without cluster constraints. Since the clusters depicted here (one color for each cluster) do not contribute to the considered simulation, they disappear during the reduction process.

The overall reduction process is summarized by Fig. 4.28.

Out of curiosity, mesh reductions were coupled with the solar and Earth & albedo components independently. As expected, the geometric reduced models accurate to numerical simulations were not the same. On one hand, for the solar simulation, clusters in the shadow were fully decimated without altering the nodal simulation. On the other hand, for the Earth & albedo simulation, clusters totally hidden from direct fluxes were almost fully decimated, see Fig. 4.27. In the end, two different reduced models are considered.

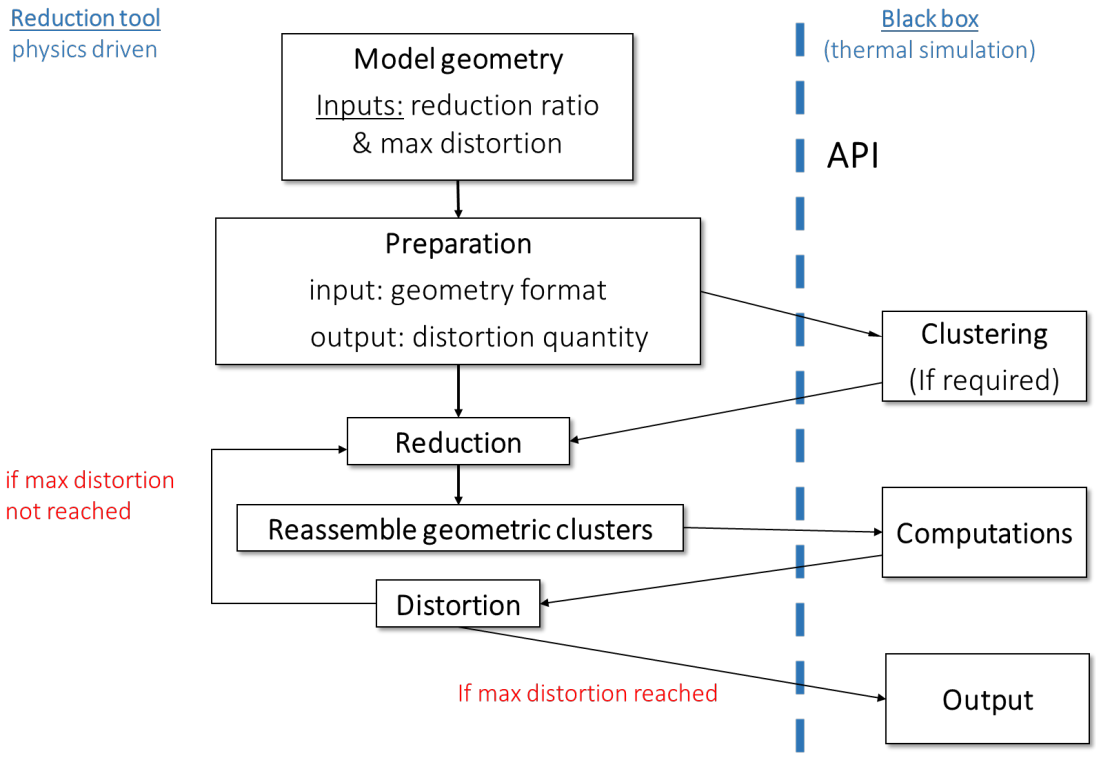


Figure 4.28: Summary of reduction process. Left: mesh reduction. Right: numerical simulation. The reduction and the simulation parts are connected with an API, converting the geometric format to a thermal one from left to right, and returning the maximum error value from right to left. The considered case for this thesis is the radiative thermal simulation, but another numerical simulation could be plugged to the geometric reduction instead.

4.4 Conclusion and Future Work

This chapter explored a mesh reduction process accurate to and driven by a numerical simulation. The approach relies on performing a sensitivity analysis as a pre-processing step for each thermal node, yielding predictions of feasible reduction ratios given temperature difference tolerances. These reduction ratios are then utilized to reduce the thermal nodes accordingly, which are reassembled afterward to perform the numerical simulation global reduced model. The reduction process is performed until the maximum allowed temperature difference is reached. Furthermore, we modify the priority queue by an importance factor obtained through the numerical simulation results. A specific focus has been put on being faithful to the reference numerical simulation values while greatly reducing the number of geometrical elements. However, several limitations arise from the proposed prediction approach. For instance, finding the globally optimal reduced model for a given maximum temperature difference cannot be guaranteed. Specificially,

multiple approximated models can be conceived depending on (1) the chosen error metric, (2) the considered simulation and (3) the geometric configuration of the satellite at a given time. As future work, another error metric designed to best preserve the numerical simulation can be explored. For instance, view factors between thermal nodes can be preserved during the reduction process by increasing the cost of a collapse operator for facets with important view factor values. Furthermore, supervised learning can be investigated to find the error metric that best preserves the numerical simulation. Recent advances based on fine-grained models such as molecular dynamics or agent-based models have been carried out with success for other simulated phenomena [KK21].

Chapter 5

Conclusion and Perspectives

Contents

5.1 Conclusion

With the growing needs of real-time simulation, particularly mandatory for the development of digital twins, interruptible computations and model order reduction techniques have to be considered. The main objective of the present thesis is to propose and validate new methods to tackle the real-time challenges induced by the radiative thermal simulation of satellites. To answer the considered problematic, the angle of attack is twofold. In chapter 3, the challenge of computing the quantity governing the radiative simulation, also known as view factors, is addressed. A novel method to compute view factors in a progressive and interruptible way is proposed. Basically, a hierarchical data-structure involving exact predicates is suggested to best trade desired accuracy with involved number of quadrature elements. Furthermore, when the time dedicated to view factor computations is up, a prediction step is added on top of it to approach the reference value even more. The accuracy of the method is compared with state of the art methods implemented in a software dedicated to view factor computations, View3D. The obtained accuracy depending on the number of quadrature elements are also shown. In chapter 4, a geometric model reduction connected to numerical simulation is proposed. The challenge being to reduce a geometrical model while being faithful to the radiative thermal simulation. To address this problematic, a quadric error metric algorithm has been implemented, then the reduction process is paired with an oracle that performs the numerical simulation and returns a maximum error, named distortion. A novel physics-aware geometric reduction is proceeded thanks to a sensitivity analysis performed as a pre-processing step. The meshed model is subdivided into grouping of faces with isothermal properties, named thermal nodes. A thermal node is picked, reduced according to a given ratio and re-added to the model, then the numerical simulation is requested for every decimation step applied to the cluster until reaching the maximum allowed error. The process is applied to every cluster, returning the information of which clusters are sensible or not to the reduction process. Also a modification of the cost operator in the decimation algorithm's priority queue obtained directly through numerical simulation is suggested. By extension, the decimation process accurate to numerical simulation results could be applied to other numerical simulations.

5.2 Perspectives

This section seeks to discuss further improvements to the methods presented in this thesis.

5.2.1 Computing View Factors

For the view factor computations, the method explained in this thesis require linear elements as input, obtained through tessellation of CAD models for finite elements techniques. Higher order elements, notably NURBS or Bézier surfaces, can be considered by extending the closed form formulas to these curve elements, removing the approximation

error induced by the triangulation step. Doing so will also save precious computation times since a single NURBS or Bézier surface is currently approximated by a substantial amount of triangles. Considering higher order elements thus reduces the number of requests to closed form formulae.

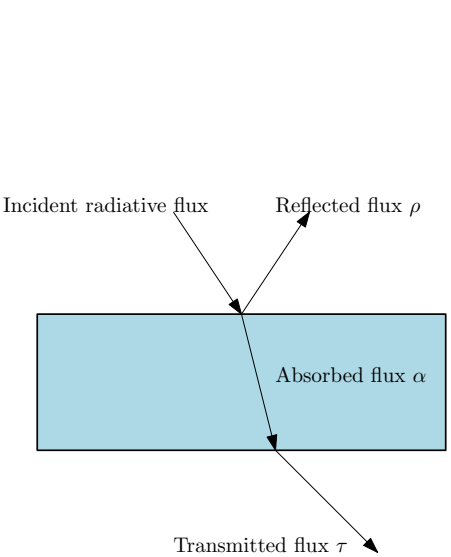
Another perspective consists in reducing the number of operations to compute the view factor matrix, currently coming at a quadratic cost in the number of faces. To tackle this problematic, facet clustering techniques can be considered to reduce the number of mesh elements. Since the complexity of view factors is quadratic, halving the number of faces theoretically reduces the number of operations by a factor 4, resulting in a substantial gain in computation times.

As future work, it would be interesting to improve the view factor predictor using machine learning algorithms for a more accurate visibility ratio by computing relevant descriptors. For instance, appropriate descriptors can be the ratio of visible line probes i.e. not intersecting any visibility obstacles, the mean of the angles between obstacles and line probes, the mean and standard deviation of the obstacle areas, the spread of the obstacles, the mean of obstacle areas over source and target areas, and the ratio of obstacles inside the convex hull amongst others. For that purpose, such data can be collected during the recursions in the tree of the data-structure for leaf nodes as well as their ancestors. These data can then be given to machine learning algorithms, separating relevant from meaningless descriptors.

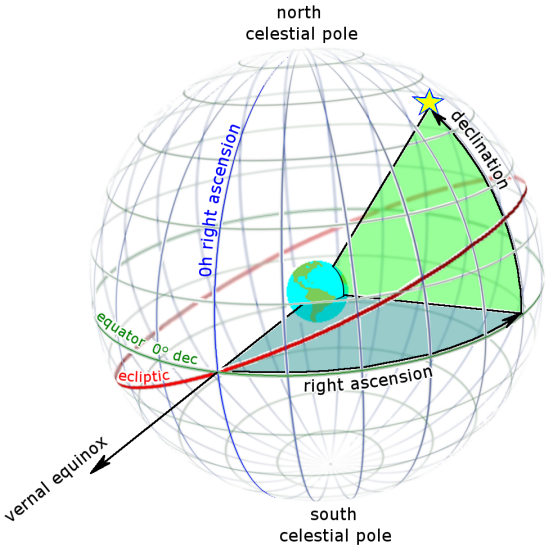
5.2.2 Driving Geometric Reduction by Numerical Simulation

For the geometric model reduction, a specific error metric directly accurate to view factors can be examined. For instance, pair of faces or even pair of face clusters with a mutual high view factor value should be preserved with a high decimation cost, whereas faces with a negligible view factor value could be decimated first. Such process requires a preprocessing step to compute view factors between faces or between face clusters (such as faces of thermal nodes), utilized to modify the geometric cost operator by a factor directly accurate to view factors.

Also, supervised-learning methods can be analyzed to determine an error metric that best preserves the numerical simulation. Regarding the radiative thermal simulation, several key features can be considered. The first and obvious one is the view factor (induced by the geometry of the model) as mentioned previously. Moreover, thermo-optical properties play an important role in radiative thermal simulation, being the absorptivity α (corresponds to the radiative flux absorbed by a surface element at a given temperature, related to the incident flux), reflexivity ρ (reflected radiative flux) and transmissivity τ (radiative flux passing through the material), see Fig. 5.1a. These quantities characterize the behavior of a body regarding the radiation it receives and are linked by the following formula: $\alpha + \rho + \tau = 1$. Furthermore, the solar declination and the right ascension angles are both involved in radiative computations, see Fig. 5.1b.



(a) Thermo-optical properties.



(b) Right ascension and declination, credits to Tfr000.

Figure 5.1: Quantities implied in radiative thermal simulation.

5.2.3 Geometric Model Reduction Limitations

Considering a geometric model partitioned into clusters of faces, the deepest achievable theoretical decimation would yield one face per cluster. However, even with such a heavy mesh reduction, real-time simulations can still be challenging with a high number of clusters (thermal nodes for radiative simulation). In this industry, a mathematical model reduction is manually proceeded by the thermal expert to reduce the number of thermal nodes, as explained by Basset et al. with TMRT tool [BHCF16], relying on trial and error processes. This method is mainly used when integrating a spacecraft model into a launcher model or for electronic equipment model integrations (tubes, transmitters, receivers amongst others). An interesting idea would be to automatize the reduction definition process, where the thermal nodes are gathered into average nodes, or unchanged (kept nodes), or deleted if their contribution is negligible (suppressed nodes). A first naive approach relying on machine learning techniques has been studied, consisting in clustering data in high dimension [UL16] (at least 4D) initially based on their position in the 3D space and their temperature. The idea is to learn and mimic technical expertise involved during the manual clustering with the help of the Python library scikit-learn [PVG+11]. Partitional, hierarchical and density based methods have been tested on a data set of a satellite where each datum consists of the position of the centroid of a thermal node and the corresponding temperature. In more details, the model consists of 1400 external thermal nodes, and considering a thermal mapping of nodes at steady state. The centers of the spheres represent the centers of gravity of

the clusters, and the radius of the spheres are a function of the thermal node areas, see Fig. 5.2. The considered error for a cluster is given by the maximum error between the average temperature of a cluster and the highest/lowest temperature of the nodes in the cluster.

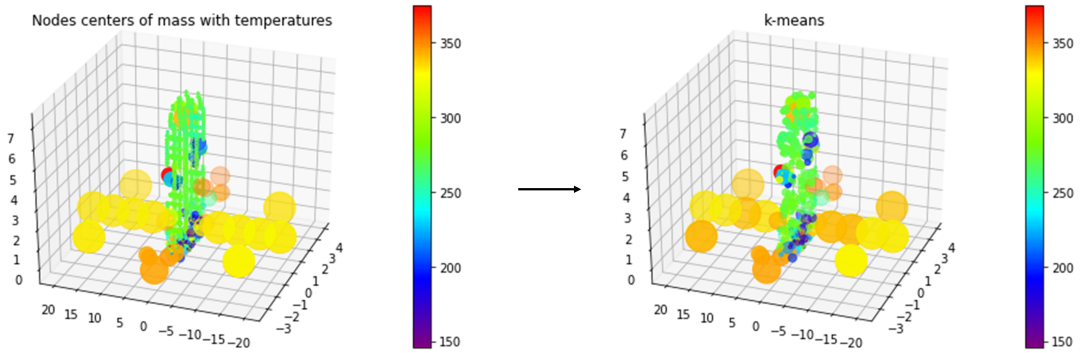


Figure 5.2: K-means clustering algorithm [Llo82] applied to the external part of a satellite (model from DOREA). Reduction from 1400 thermal nodes to 400 clusters with an error of 0.85K degree. K-means take as input the desired number of clusters.

This first approach can be improved by considering a thermal distance instead of an euclidean one, for instance by taking into account boundary conditions as well as conductive and radiative couplings.

Bibliography

[ACK13] Marco Attene, Marcel Campen, and Leif P. Kobbelt. Polygon mesh repairing: An application perspective. *ACM Comput. Surv.*, 45:15:1–15:33, 2013.

[AK18] Andreas Ampatzoglou and Vassilis Kostopoulos. Design, analysis, optimization, manufacturing, and testing of a 2u cubesat. *International Journal of Aerospace Engineering*, 2018:1–15, 2018.

[AMDL12] O. Apostu, F. Mora, Ghazanfarpour D., and Aveneau L. Analytic ambient occlusion using exact from-polygon visibility. *Computers and Graphics Volume 36*, 2012.

[ARL20] Andrs Arias-Rosales and Philip R. LeDuc. Comparing view factor modeling frameworks for the estimation of incident solar energy. *Applied Energy*, 277:115510, 2020.

[ATC20] P. Alliez, S. Tayeb, and Wormser C. 3D fast intersection and distance computation. In *CGAL User and Reference Manual*. CGAL Editorial Board, 5.0.1 edition, 2020.

[BB16] Benoit Beckers and Pierre Beckers. Fast and accurate view factor generation. In *First International Conference on Urban Physics*, 09 2016.

[BCGP+96] H. Boyer, J.P. Chabriat, B. Grondin-Perez, C. Tourrand, and J. Brau. Thermal building simulation and computer generation of nodal models. *Building and Environment*, 31(3):207–214, 1996.

[BFFP70] J. Bourrieau, C. Flbaz, J.F. Faugere, and A. Paillous. Space and spacecraft environment. *IFAC Proceedings Volumes*, 3(1):239–244, 1970. 3rd International IFAC Conference on Automatic Control in Space, Toulouse, France, March 2-6, 1970.

[BGV96] Bernhard Boser, Isabelle Guyon, and Vladimir Vapnik. A training algorithm for optimal margin classifier. *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, 5, 08 1996.

[BHCF16] T. Basset, P. Hugonnot, P. Connil, and M. Ferrier. Tmrt (thermal model reduction tool), presentation of the tool and application on satellite model reduction for launcher coupled analysis. *46th Int. Conf. Environ. Syst., American Institute of Aeronautics and Astronautics, Vienna, Austria*, 07 2016.

[BHL93] G Berkooz, P Holmes, and J L Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25(1):539–575, 1993.

[BHM20] Matthew V. Bilskie, Scott C. Hagen, and Stephen C. Medeiros. Unstructured finite element mesh decimation for real-time hurricane storm surge forecasting. *Coastal Engineering*, 156:103622, 2020.

[BIDL11] T.L. Bergman, F.P. Incropera, D.P. DeWitt, and A.S. Lavine. *Fundamentals of Heat and Mass Transfer*. Wiley, 2011.

[BK05] Stephan Bischoff and Leif P. Kobbelt. Structure preserving cad model repair. *Computer Graphics Forum*, 24, 2005.

[BP61] Albert J. Buschman and Claud M. Pittman. Configuration factors for exchange of radiant energy between axisymmetrical sections of cylinders, cones, and hemispheres and their bases. *NASA TN D-944*, 1961.

[BRW89] D. R. Baum, H. E. Rushmeier, and J. M. Winget. Improving Radiosity Solutions Through the Use of Analytically Determined Form-Factors. *Computer Graphics 23*, 1989.

[CAK12] Marcel Campen, Marco Attene, and Leif Kobbelt. A practical guide to polygon mesh repairing. *Eurographics Tutorials*, 01 2012.

[CB68] Roy R Craig and Mervyn C C Bampton. Coupling of Substructures for Dynamic Analyses. *AIAA Journal*, 6(7):1313–1319, 1968.

[CGD85] Michael Cohen, Michael Greenberg, and P.C. Donald. The hemi-cube: A radiosity solution for complex environments. *SIGGRAPH '85 conference proceedings. July, 1985. vol. 19 ; no. 3: pp. 31-39 : ill. (some col.). includes bibliography*, 19, 01 1985.

[CL11] C.C. Chang and C.J. Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2:27:1–27:27, 2011.

[CSAD04] David Cohen-Steiner, Pierre Alliez, and Mathieu Desbrun. Variational Shape Approximation. Research Report RR-5371, INRIA, November 2004.

[CW15] Danny Z. Chen and Haitao Wang. Visibility and ray shooting queries in polygonal domains. *Computational Geometry*, 48(2):31–41, 2015.

[DAHSW14] E. D´ Azevedo, Z. Hu, S.-Q. Su, and K. Wong. Solving a large scale radiosity problem on GPU-based parallel computers, 2014.

[DCSA+13] Julie Digne, David Cohen-Steiner, Pierre Alliez, Fernando De Goes, and Mathieu Desbrun. Feature-Preserving Surface Reconstruction and Simplification from Defect-Laden Point Sets. *Journal of Mathematical Imaging and Vision*, pages 1–14, January 2013.

[DDTP00] Frédo Durand, George Drettakis, Joëlle Thollot, and Claude Puech. Conservative visibility preprocessing using extended projections. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH 00, pages 239–248, USA, 2000. ACM Press/Addison-Wesley Publishing Co.

[DM98] L. Dagum and R. Menon. Openmp: an industry standard api for shared-memory programming. *IEEE Computational Science and Engineering*, 5(1):46–55, 1998.

[Duv15] Régis Duvigneau. A Sensitivity Equation Method for Unsteady Compressible Flows: Implementation and Verification. Research Report RR-8739, Inria, June 2015.

[EY36] C. Eckart and G. Young. The approximation of one matrix by another of lower rank. *Psychometrika*, 1:211–218, 1936.

[FAB+18] L. Feng, P. Alliez, L. Busé, H. Delingette, and M. Desbrun. Curved Optimal Delaunay Triangulation. *ACM Transactions on Graphics*, 37(4):16, 2018.

[FFFP16] Jaime Fernández, Carlos Fernández, Pierre Féménias, and Heike Peter. The copernicus sentinel-3 mission. In *ILRS workshop*, pages 1–4, 2016.

[FGdpL06] Franco Fossati, G. Gambacciani, Savino de palo, and S. Langlois. Design and manufacturing of a wing hybrid metal/ceramic hot structure. *5th European Workshop on Thermal Protection Systems and Hot Structures*, page 25, 07 2006.

[FLK+10] Stefan Föckersperger, Klaus Lattner, Clemens Kaiser, Silke Eckert, Swen Ritzmann, Robert Axmann, and Michael Turk. The on-orbit verification mission tet-1 project status of the small satellite mission & outlook for a one year mission operation phase. *ICSO 2010 Conference Proceedings, Rhodes, Greece, International Conference on Space Optics*, pages 4–8, 01 2010.

[Fre04] Pascal Frey. Generation and adaptation of computational surface meshes from discrete anatomical data. *International Journal for Numerical Methods in Engineering*, 60, 2004.

[Gar99] Michael Garland. *Quadric-based polygonal surface simplification.* PhD thesis, School of Computer Science Carnegie Mellon University, 1999.

[GBS+17] Manoj Kumar Gupta, Kuldip J Bumtariya, HA. Shukla, Pranav Patel, and Ziauddin Khan. Methods for evaluation of radiation view factor: A review. *Materials Today: Proceedings*, 4(2, Part A):1236–1243, 2017. 5th International Conference of Materials Processing and Characterization (ICMPC 2016).

[GDL14] Jesus Gonzalo, Diego Dominguez, and Deibi Lopez. On the challenge of a century lifespan satellite. *Progress in Aerospace Sciences*, 70:28–41, 2014.

[Geo] George Walton and John Pye. View3d.

[GGC+09] Carlos Gonzalez, Jesus Gumbau, Miguel Chover, Francisco Ramos, and Ricardo Quiros. User-assisted simplification method for triangle meshes preserving boundaries. *Computer-Aided Design*, 41:1095–1106, 12 2009.

[GH97] Michael Garland and Paul Heckbert. Surface simplification using quadric error metrics. *Proceedings of the ACM SIGGRAPH Conference on Computer Graphics*, 1997, 07 1997.

[GH00] Michael Garland and Paul Heckbert. Simplifying surfaces with color and texture using quadric error metrics. *IEEE Visualization 98*, 02 2000.

[Gha19] Anahid Ghazanfarpour. *Proximity-aware multiple meshes decimation using quadric error metric.* Theses, Université Paul Sabatier - Toulouse III, November 2019.

[Gil02] David G. Gilmore. *Spacecraft Thermal Control Handbook, Volume I: Fundamental Technologies.* American Institute of Aeronautics and Astronautics, 2002.

[GJ+10] G. Guennebaud, B. Jacob, et al. Eigen v3. http://eigen.tuxfamily.org, 2010.

[GJQ19] B. Gaume, F. Joly, and O. Quemener. Modal reduction for a problem of heat transfer with radiation in an enclosure. *International Journal of Heat and Mass Transfer*, 141:779 – 788, 2019.

[Gla84] Andrew S. Glassner. Space subdivision for fast ray tracing. *IEEE Computer Graphics and Applications*, 4(10):15–24, 1984.

[GLB+21] Manuel Girault, Yang Liu, Yann Billaud, Adel Benselama, Didier Saury, and Denis Lemonnier. Reduced Order Models for conduction and radiation inside semi-transparent media via the Modal Identification Method. *International Journal of Heat and Mass Transfer*, 168, April 2021.

[GMH+20] Anahid Ghazanfarpour, Nicolas Mellado, Chems E. Himeur, Loic Barthe, and Jean-Pierre Jessel. Proximity-aware multiple meshes decimation using quadric error metric. *Graphical Models*, 109:101062, 2020.

[Guy65] Robert J. Guyan. Reduction of stiffness and mass matrices. *AIAA Journal*, 3(2):380, February 1965.

[Her09] R.A. Herman. *A Treatise on Geometrical Optics*. Read Books, 2009.

[HM11] John Howell and M. Pinar Meng. Radiative transfer configuration factor catalog: A listing of relations for common geometries. *Journal of Quantitative Spectroscopy & Radiative Transfer - J QUANT SPECTROSC RADIAT*, 112:910–912, 03 2011.

[Hop99] Hugues Hoppe. New quadric metric for simplifying meshes with appearance attributes. *Proceedings Visualization '99 (Cat. No.99CB37067)*, pages 59–510, 1999.

[Hot33] H. Hotelling. Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology*, 24:498–520, 1933.

[HS67] H. C. Hottel and A. F. Sarofim. *Radiation transfer (Mc Graw-Hill)*. Mc Graw-Hill, 1967.

[HSA91] Pat Hanrahan, David Salzman, and Larry Aupperle. A rapid hierarchical radiosity algorithm. *SIGGRAPH Comput. Graph.*, 25(4):197–206, July 1991.

[HW94] Eric A. Haines and John R. Wallace. Shaft culling for efficient ray-cast radiosity. In P. Brunet and F. W. Jansen, editors, *Photorealistic Rendering in Computer Graphics*, pages 122–138, Berlin, Heidelberg, 1994. Springer Berlin Heidelberg.

[IL14] Bertrand Iooss and Paul Lemaitre. A review on global sensitivity analysis methods, 2014.

[ITS+19] Maksim Ivanushkin, Ivan Tkachenko, S Safronov, Ivan Kaurov, and Sergey Volgin. On the results of processing of the telemetry data received from the aist small satellite constellation. *Journal of Physics: Conference Series*, 1368:042062, 11 2019.

[Jac16] Lionel Jacques. *Space Thermal Analysis through Reduced Finite Element Modelling*. PhD thesis, Centre Spatial de Liege, 12 2016.

[JMK13] L. Jacques, L. Masset, and G. Kerschen. Ray Tracing Enhancement For Space Thermal Analysis: Isocell Method, 2013.

[KCS98] Leif P. Kobbelt, Swen Campagna, and Hans-Peter Seidel. A general framework for mesh decimation. In *Graphics Interface*, 1998.

[KG03] Youngihn Kho and Michael Garland. User-guided simplification. In *Proceedings of the 2003 Symposium on Interactive 3D Graphics*, I3D, pages 123–126, New York, NY, USA, 2003. Association for Computing Machinery.

[KGP+14] Stephan Kramer, Ralf Gritzki, Alf Perschk, Markus Rosler, and Clemens Felsmann. Fully parallel, opengl-based computation of obstructed area-to-area view factors. *Journal of Building Performance Simulation*, 8:1–16, 12 2014.

[KGVB05] Gaetan Kerschen, J.-C. Golinval, Alexander Vakakis, and Lawrence Bergman. The method of proper orthogonal decomposition for dynamical characterization and order reduction of mechanical systems: An overview. *Nonlinear Dynamics*, 41:147–169, 08 2005.

[KK21] Sebastian Kaltenbach and Phaedon-Stelios Koutsourelakis. Physics-aware, probabilistic model order reduction with guaranteed stability, 2021.

[KLS96] R. Klein, G. Liebich, and W. Strasser. Mesh reduction with error control. In *Proceedings of Seventh Annual IEEE Visualization '96*, pages 311–318, 1996.

[KU77] Ashok Kumar and T.E. Unny. Application of runge-kutta method for the solution of non-linear partial differential equations. *Applied Mathematical Modelling*, 1(4):199–204, 1977.

[LB17] Mickael Le Bohec. *Contribution du rayonnement au confort thermique et aux économies d'énergie dans l'habitat.* PhD thesis, ESMA, 2017. Thèse de doctorat dirigée par Lemonnier, Denis et Saury, Didier Energétique, thermique, combustion Chasseneuil-du-Poitou, Ecole nationale supérieure de mécanique et d'aérotechnique 2017.

[Leb17] Timothee Leblond. *Calcul de gradient sur des paramètres CAO pour l'optimisation de forme.* Theses, Université Paris-Saclay, March 2017.

[Lin00] Peter Lindstrom. Out-of-core simplification of large polygonal models. In *Siggraph 2000, Computer Graphics Proceedings*, 07 2000.

[Llo82] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[LT98] Peter Lindstrom and Greg Turk. Fast and memory efficient polygonal simplification. In *Proceedings of the Conference on Visualization '98*, VIS '98, pages 279–286, Washington, DC, USA, 1998. IEEE Computer Society Press.

[MB19] Joby Mackolil and Mahanthesh B. Sensitivity analysis of radiative heat transfer in casson and nano fluids under diffusion-thermo and heat absorption effects. *The European Physical Journal Plus*, 134, 12 2019.

[MI20] T. Muneer and Stoyanka Ivanova. Efficient routines for obtaining radiation view-factor for non-uniform horizons. *Energies*, 13:2551, 05 2020.

[MS11] M. Mirhosseini and A. Saboonchi. View factor calculation using the monte carlo method for a 3d strip element to circular cylinder. *International Communications in Heat and Mass Transfer*, 38(6):821–826, 2011.

[MT97] Tomas Moller and Ben Trumbore. Fast, minimum storage ray-triangle intersection. *Journal of Graphics Tools*, 2(1):21–28, 1997.

[MV87] M. P. Menguc and R. Viskanta. A sensitivity analysis for radiative heat transfer in a pulverized coal-fired furnace. *Combustion Science and Technology*, 51(1-3):51–74, 1987.

[Nas11] Nasa. *Thermal Network Modelling Handbook*. K& K Associates, 2011.

[NM95] Atul Narkhede and Dinesh Manocha. Vii.5 - fast polygon triangulation based on seidel's algorithm. In Alan W. Paeth, editor, *Graphics Gems V*, pages 394–397. Academic Press, Boston, 1995.

[NT03] F.S. Nooruddin and G. Turk. Simplification and repair of polygonal models using volumetric techniques. *IEEE Transactions on Visualization and Computer Graphics*, 9(2):191–205, 2003.

[Nus28] Wilhelm Nusselt. Graphische bestimmung des winkelverhaltnisses bei der wärmestrahlung. *Zeitschrift des Vereines Deutscher Ingenieure*, 72(20):673, 1928.

[PAH+19] Cédric Portaneri, Pierre Alliez, Michael Hemmer, Lukas Birklein, and Elmar Schoemer. Cost-driven framework for progressive compression of textured meshes. In *Proceedings of the 10th ACM Multimedia Systems Conference*, MMSys, pages 175–188, New York, NY, USA, 2019. Association for Computing Machinery.

[PLK+21] Erick Petersen, Jorge Lopez, Natalia Kushik, Claude Poletti, and Djamal Zeghlache. Satellite communication digital twin for evaluating novel solutions: Dynamic link emulation architecture, 2021.

[PS03] Erik Pojar and Dieter Schmalstieg. User-controlled creation of multiresolution meshes. In *Proceedings of the 2003 Symposium on Interactive 3D Graphics, SI3D 2003, Monterey, California, USA, April 28-30, 2003*, pages 127–130, 01 2003.

[PVG+11] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[RG10] J.N. Reddy and D.K. Gartling. *The Finite Element Method in Heat Transfer and Fluid Dynamics*. Applied and Computational Mechanics. CRC Press, 2010.

[RHS05] A. Robson, C. Heller, and H. Sdunnus. Space systems thermal analysis software - a user's view. In *International Conference On Environmental Systems*. SAE International, jul 2005.

[RHSB21] Katie Richmond, Asher Hancock, Shervin Sammak, and Matthew Barry. Numerically resolved radiation view factors via multi-gpu accelerated ray tracing. In *15th International Conference on Heat Transfer, Fluid Mechanics and Thermodynamics*, 07 2021.

[Rit82] G. Ritoux. Evaluation numérique des facteurs de forme, 1982.

[RR96] Rémi Ronfard and J. Rossignac. Full-range approximation of triangulated polyhedra. *Computer Graphics Forum*, 15(3):67–76, 1996.

[RSV14] Antonio Rossi, Sergey Stark, and Ruggero Vaglio. Evidence for thermal boundary resistance effects on superconducting radiofrequency cavity performances. *Superconductor Science and Technology*, 27:085004, 06 2014.

[SA85] J. Saulnier and A. Alexandre. La modélisation thermique par la méthode nodale: ses principes, ses succès et doses limites. *Revue Générale de Thermique*, 24:363–372, 1985.

[SCD20] Duansen Shangguan, Liping Chen, and Jianwan Ding. A digital twin-based approach for the fault diagnosis and health monitoring of a complex satellite system. *Symmetry*, 12(8), 2020.

[SDDS00] Gernot Schaufler, Julie Dorsey, Xavier Décoret, and François X. Sillion. Conservative Volumetric Visibility with Occluder Fusion. In Kurt Akeley, editor, *Computer Graphics Proceedings*, Annual Conference Series, pages 229–238, Nouvelle-Orléans, United States, 2000. ACM Press / ACM SIGGRAPH / Addison Wesley Longman. Annual Conf.

[SG03a] Frutuoso G. M. Silva and Abel J. P. Gomes. Adjacency and incidence framework: A data structure for efficient and fast management of multiresolution meshes. In *Proceedings of the 1st International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, GRAPHITE 03, pages 159–166, New York, NY, USA, 2003. Association for Computing Machinery.

[SG03b] Frutuoso GM Silva and Abel JP Gomes. Aif - a data structure for polygonal meshes. In *International Conference on Computational Science and Its Applications*, pages 478–487. Springer, 2003.

[SH93] P. Schroeder and P. Hanrahan. A Closed Form Expression for the Form Factor between Two Polygons, 1993.

[SHN02] Joseph H. Saleh, Daniel E. Hastings, and Dava J. Newman. Spacecraft design lifetime. *Journal of Spacecraft and Rockets*, 39(2):244–257, 2002.

[Shu15] Kedar Shukla. Heat pipe for aerospace applications - an overview. *Journal of Electronics Cooling and Thermal Control*, 05:1–14, 01 2015.

[SLA15] David Salinas, Florent Lafarge, and Pierre Alliez. Structure-Aware Mesh Decimation. *Computer Graphics Forum*, page 20, January 2015.

[SMS+16] M. Sindram, T. Machl, H. Steuer, M. Pültz, and T. H. Kolbe. Voluminator 2.0 - Speeding up the Approximation of the Volume of Defective 3d Building Models. *ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences*, III2:29–36, June 2016.

[SP19] S. Silvestri and R. Pecnik. A fast gpu monte carlo radiative heat transfer implementation for coupling with direct numerical simulation. *Journal of Computational Physics: X*, 3:100032, 2019.

[SS05] Stanko Vl. Shtrakov and Anton Stoilov. New approach for finite difference method for thermal analysis of passive solar systems. *CoRR*, abs/cs/0502059, 2005.

[SZL92] William J. Schroeder, Jonathan A. Zarge, and William E. Lorensen. Decimation of triangle meshes. In *Proceedings of the 19th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH, pages 65–70, New York, NY, USA, 1992. Association for Computing Machinery.

[TAD10] J. Tournois, P. Alliez, and O. Devillers. 2D Centroidal Voronoi Tessellations with Constraints. *Numerical Mathematics: Theory, Methods and Applications*, 3(2):212–222, 2010.

[The20] The CGAL Project. *CGAL User and Reference Manual*. CGAL Editorial Board, 5.0.1 edition, 2020.

[TK20] Philip Trettner and Leif Kobbelt. Fast and Robust QEF Minimization using Probabilistic Quadrics. *Computer Graphics Forum*, 2020.

[TR21] Josef Tausendschon and Stefan Radl. Deep neural network-based heat radiation modelling between particles and between walls and particles. *International Journal of Heat and Mass Transfer*, 177:121557, 2021.

[UL16] Alfred Ultsch and Jorn Lotsch. Machine-learned cluster identification in high-dimensional data. *Journal of Biomedical Informatics*, 66, 12 2016.

[Ust02] Eugene Ustinov. Sensitivity analysis of radiative heating and cooling rates in planetary atmospheres: General linearization and adjoint approaches. *European Geophysical Union*, 01 2002.

[Uyg18] Ahmet Bilge Uygur. A complete methodology for the computation of external heat fluxes for the transient thermal analysis of satellites. *Journal of Aeronautics and Space Technologies*, 11:17–27, 01 2018.

[VA17] Javier Videla and Elena Atroshchenko. *Fracture Modelling directly from Computer-Aided Design (CAD) by the Extended Isogeometric Finite Element Method (X-IGA FEM) with trimmed NURBS*. PhD thesis, Universidad de Chile, Facultad de Ciencias Fisicas y Matematicas, Departamento de Ingeneria Mecanica, 06 2017.

[Var95] A. Varga. Enhanced modal approach for model reduction. *Mathematical Modelling of Systems*, 1(2):91–105, 1995.

[VLB06] Mile Vujicic, Nicholas Lavery, and S.G.R. Brown. View factor calculation using the monte carlo method and numerical sensitivity. *Communications in Numerical Methods in Engineering*, 22:197 – 203, 05 2006.

[VM21] Veenit Kr. Verma and Sabyasachi Mondal. A brief review of numerical methods for heat and mass transfer of casson fluids. *Partial Differential Equations in Applied Mathematics*, 3:100034, 2021.

[VPP+06] A Viana, Oscar Polo, Pablo Parra, Oscar Poblacin, J. Ignacio Garca Tejedor, Sebastin Snchez Prieto, and D Meziat. Increasing the determinism in real-time operating systems for erc32 architecture. *Proceedings of the 5th WSEAS International Conference on Software Engineering, Parallel and Distributed Systems*, 02 2006.

[Wal02] G. Walton. Calculation of Obstructed View Factors by Adaptive Integration. Technical Report 6925, NIST, 2002.

[Wal19] S. Walk. Random Forest Template Library, 2019.

[WEH89] J. R. Wallace, K. A. Elmquist, and E. A. Haines. A Ray Tracing Algorithm for Progressive Radiosity. *Computer Graphics 23*, 1989.

[WH19] Jianren Wang and Yihui He. Physics-aware 3d mesh synthesis. In *2019 International Conference on 3D Vision (3DV)*, pages 502–512, 2019.

[WXB12] Trevor Walker, S.-C. Xue, and Geoff W. Barton. A robust monte carlo based ray-tracing approach for the calculation of view factors in arbitrary three-dimensional geometries. *Computational Thermal Sciences: An International Journal*, 4(5):425–442, 2012.

[Zee02] Charles Nelson Zeeb. *Performance and accuracy enhancements of radiative heat transfer modeling via Monte Carlo*. PhD thesis, Colorado State University, Fort Collins, Colorado, 2002.

[ZY13] Y. Zheng and K. Yamane. Ray-shooting algorithms for robotics. *IEEE Transactions on Automation Science and Engineering*, 10(4):862–874, 2013.

Appendices

Appendix A

Progressive View Factors Software

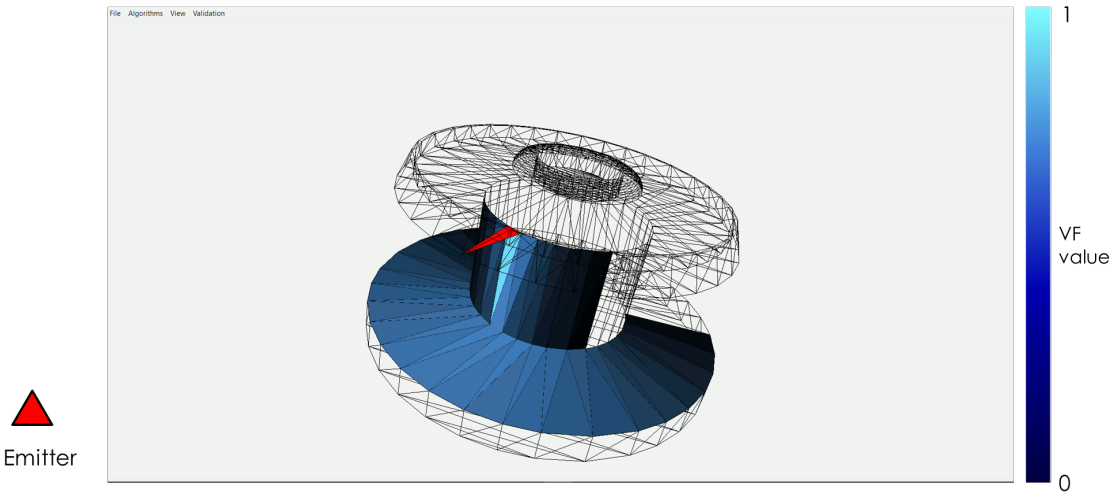


Figure A.1: Preview of the view factor software relying on our methods detailed in chapter 3.

Appendix B

Geometric Model Reduction Accurate to Numerical Simulation Software

This software is being developed by Dorea, relies on the reduction and simulation algorithms detailed in 4 and supports import and export of STEP-TAS (ESA), DAF (Thales Alenia Space), .OBJ, .OFF, .STL, .GEO, .IGES and .STEP files.

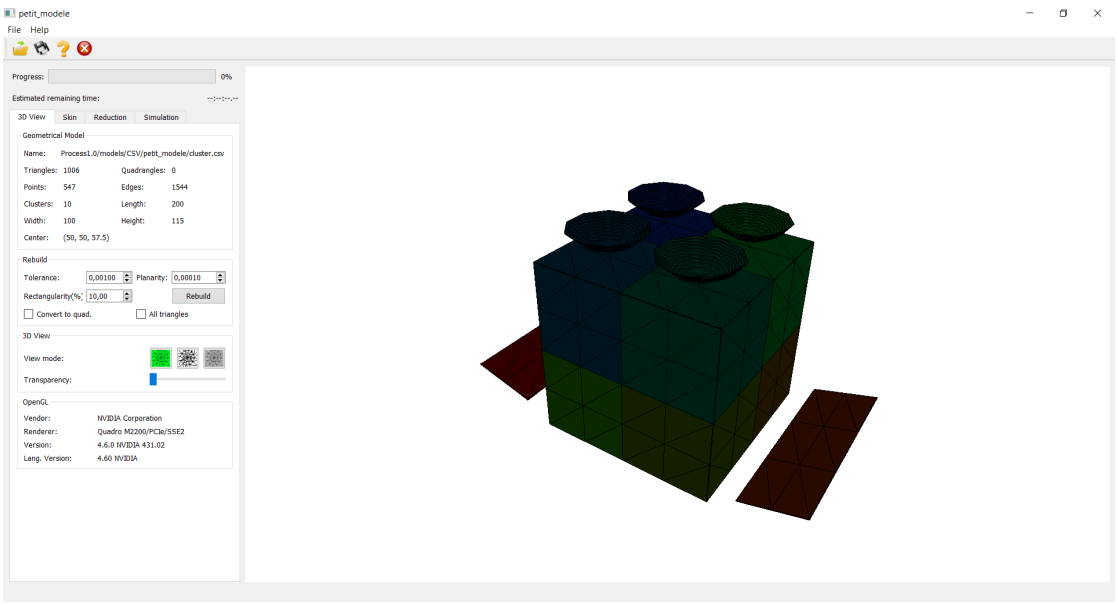


Figure B.1: Overview of the physics-informed reduction software detailed in chapter 4.

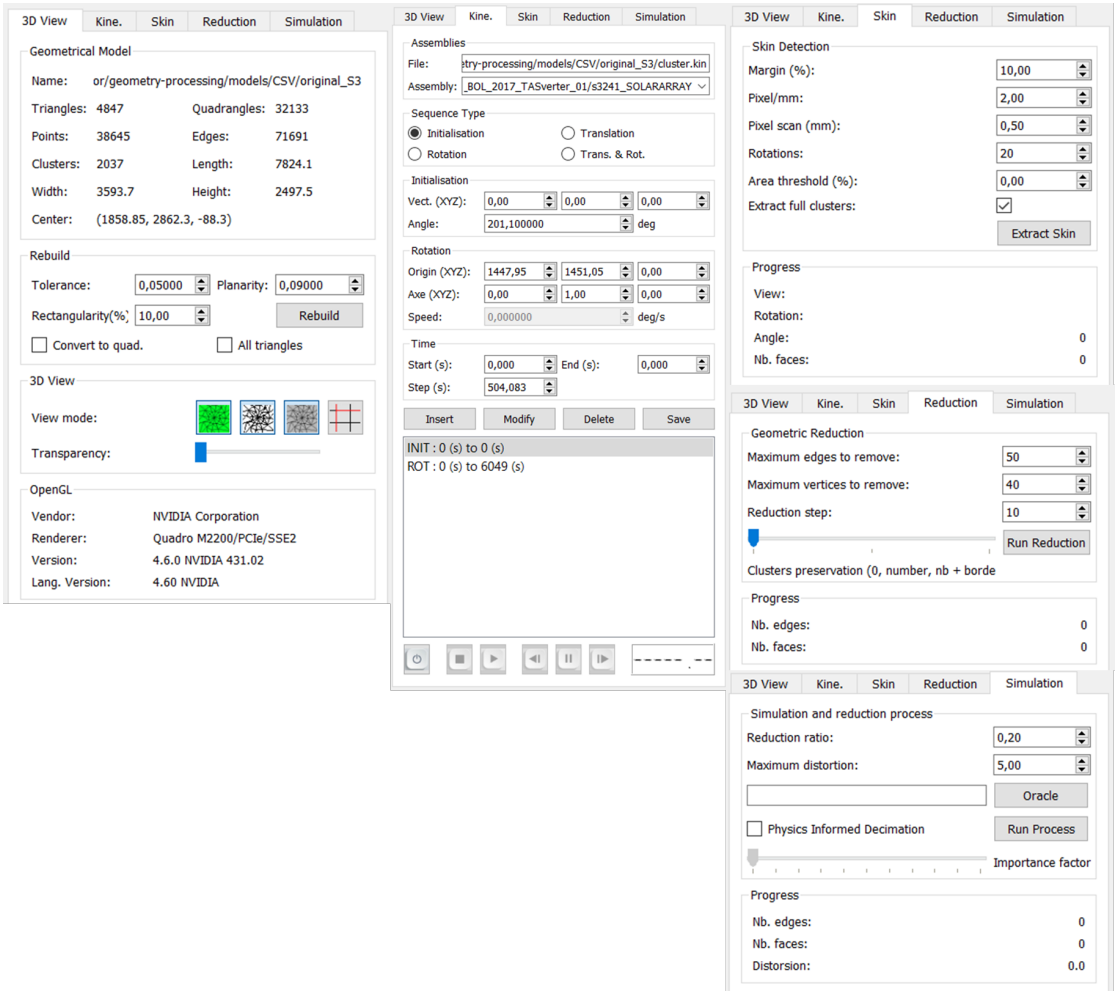


Figure B.2: Details of the different tabs of the Reduction Software. Different tabs are depicted: 3D View for mesh information and rebuilding model, Kine. to define kinematics, Skin to extract the external geometry, Reduction for mesh decimation and Simulation for reduction and simulation.