



HAL
open science

Convergent evolution in silico of biochemical log-response

Mathieu Hemery, Paul François

► **To cite this version:**

Mathieu Hemery, Paul François. Convergent evolution in silico of biochemical log-response. *Journal of Chemical Physics*, 2019. hal-02389284

HAL Id: hal-02389284

<https://inria.hal.science/hal-02389284>

Submitted on 2 Dec 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Convergent evolution in silico of biochemical log-response

Mathieu Hemery^{†,‡} and Paul François^{*,‡}

[†]*EPI Lifeware, INRIA Saclay, Palaiseau, France*

[‡]*Rutherford Physics Building, 3600 rue University, H3A2T8 Montreal, Québec, Canada*

E-mail: paulf@physics.mcgill.ca;mathieu.hemery@polytechnique.org

Abstract

Numerous biological systems are known to harbour a form of logarithmic behaviour, from Weber's law to bacterial chemotaxis. Working on a logarithmic scale allows the organism to respond appropriately to large variations in a given input at a modest cost in terms of metabolism. Here we use a genetic algorithm to evolve biochemical networks displaying a direct logarithmic response. Interestingly, a quasi-perfect log-response implemented by the same simple core network evolves in a convergent way across our different replications. The best network is able to fit a logarithm over 4 order of magnitude with an accuracy of the order of 1%. At the heart of this network, we show that a logarithmic approximation may be implemented with one single non-linear interaction, that can be interpreted either as a phosphorylation or as a ligand induced multimerization and provide an analytical explanation of the effect. Biological log-response might thus be easier to implement than usually assumed.

15 Introduction

16 Multiple cellular pathways have evolved to work over several orders of magnitude. A spec-
17 tacular example can be found in immune recognition, where cells can simultaneously be
18 insensitive to tens of thousands of non cognate ligands while being sensitive to minute concen-
19 trations of agonists with only slightly higher affinity¹. In this case, specialized mechanisms,
20 implicating proofreading cascades and multiple biochemical feedbacks have been described²,
21 implementing a so-called adaptive proofreading strategy^{3,4}. Similarly, chemotactic response
22 in *E. Coli* has been shown to be logarithmic, and there too, specific biochemical adaptation
23 pathways are responsible⁵.

24 Log-normal distributed concentrations are also the norm rather than the exception in
25 cellular biology⁶ and can be explained by a variety of simple mechanisms such as exponen-
26 tial decay or growth with a normally distributed rate or the accumulation of multiplicative
27 noises⁷. But if cells have to deal with enormous phenotypic variability, it seems reasonable
28 to assume that they are routinely able to take logarithms of concentrations to filter it out.
29 This is paradoxical since most of biochemical reactions rather work on a linear scale. Classi-
30 cal examples are Michealis-Menten/Hill (or more generally sigmoidal) kinetics, where rates
31 quickly saturate above a well defined threshold⁸.

32 This motivates a special study of general mechanisms underlying “logarithmic-response”
33 over multiple orders of magnitude. As said above, previous works have already identified pos-
34 sible models and mechanisms to explain logarithmic sensing but to our knowledge few studies
35 try to develop a direct biochemical computation of a logarithmic function. A general rule in
36 such models is that one or several feedbacks are required to allow for logarithmic sensing. As
37 an example, a synthetic biology approach combined positive feedback on low-copy-number-
38 plasmid with a shunt component on high-copy-number-plasmid to provide sensitivity over
39 4 orders of magnitudes⁹. Another proposal relies on an allosteric system with sigmoidal
40 responses, that modulates its saturation threshold with the help of a negative feedback to
41 extend its range¹⁰. It is neither clear if such mechanisms are ubiquitous, nor if simpler

42 mechanisms exist.

43 In this work, we turn to machine-learning based approach to design biochemical networks
44 showing a response proportional to the logarithm of a given input. Such approaches have
45 produced very good results to recover and predict models underlying physical or biologi-
46 cal systems^{11,12}. A key advantage of those is their interpretability, circumventing the well
47 known "black box" problem currently observed in machine learning¹³. We use φ -evo an
48 evolutionary algorithm tailored for such biological problems¹⁴. We find a simple and generic
49 mechanism generating logarithmic response, where the Input concentration catalyzes a reac-
50 tion where rates are polynomial in the substrate concentration. This allows for log sensitivity
51 over several order of magnitudes. This could be easily implemented for instance with ligand
52 induced multimerization/oligomerization of receptors¹⁵, a process that does not require any
53 feedback. Positive feedforward and negative feedback interactions can then be added to
54 improve the logarithmic behavior over 4 orders of magnitudes. Our work reveals that there
55 might be simple mechanism without feedback underlying log-sensing in a generic way, which
56 therefore might be more frequent than expected.

57 **Theoretical Methods**

58 We use the φ -evo software to evolve biochemical networks able to implement a logarithm¹⁴.
59 In short φ -evo is a Python/C-based software simulating classical Darwinian evolution on a
60 population of gene networks. Gene networks are encoded with the help of a pre-defined bio-
61 chemical grammar. Possible interactions include protein protein interactions, transcriptional
62 regulations, catalytic interactions (possibly with strong non-linear behaviours) and both pas-
63 sive and enzymatic degradations. All those interactions are included in the grammar for the
64 simulations presented here. From a given gene network, φ -evo generates a set of ordinary
65 differential equations that are numerically integrated to determine the behavior of this in-
66 dividual. Those simulated gene networks then go through a cycle of selection, growth and

67 mutation, briefly detailed below. For an extended presentation of the biochemical grammar,
68 modelization and algorithmic details, we refer the reader to the software description.¹⁴

69 φ -evo requires the definition of a fitness function for Darwinian selection. We use the
70 simplest possible objective function to promote the evolution of logarithmic behaviour as
71 depicted in figure 1. Each network contains an “Input” and an “Output” biochemical species.
72 Then, the concentration of the Input is fixed to a random constant value I (the distribution
73 of which is exponential between 1 and 10^4) for an interval of duration τ and 0 otherwise (the
74 distribution of τ is the sum of a constant and an exponential term making it both broad
75 and never too short). φ -evo then integrates the differential equations corresponding to the
76 biochemical network, and computes as an output O of the network the maximum of the
77 concentration of the “Output” species as a function of time¹.

78 For each network, the corresponding set of ODE is then integrates $N = 100$ times with
79 randomized choices for both I and τ and those data are collected to compute the network
80 fitness:

$$\varphi = \frac{\sigma(\alpha(I, \tau))}{\langle \alpha(I, \tau) \rangle^2} = \frac{\langle \alpha^2(I, \tau) \rangle}{\langle \alpha(I, \tau) \rangle^2} - 1, \text{ with: } \alpha(I, \tau) = \frac{O(I, \tau)}{\log(I)}. \quad (1)$$

81 Here σ indicates the variance and $\langle \dots \rangle$ the mean. During our simulation, the average
82 is taken over the N integrations with different random inputs. Intuitively, if the network
83 takes a perfect logarithm then $\alpha(I, \tau)$ should be a constant independent of I and τ , so that
84 each independent input choices should give the same constant α and the standard deviation
85 defined in equation 1 should be 0. We thus aim to minimize σ . The division in the fitness
86 definition is such that it does not impose a predefined value for the constant in front of
87 the logarithm definition, thus any function of the form $\alpha \log(I)$ will be selected. However,
88 the higher the α the lower the noise and this creates an indirect selection for the highest
89 possible constant which is the desired behaviour (see figure 1). σ can be interpreted as the

¹those choices can be relaxed, for instance results were similar when the "output" was the integral of the output concentration or its final concentration instead of its maximum. This property is actually a direct consequence of our desired independance of the final output in the time of the output τ as discussed later.

90 quality of the logarithm taken as a linear function of $\log I$ (lower σ being higher quality).
91 Note that while the value of τ appears in the computation of O , it is not present in the
92 evaluation of the fitness φ , hence the network should be able to get rid of the time for which
93 the input is present and respond only to the very intensity of the input signal. Figure 1
94 visually illustrates examples of fitness computation.

95 φ -evo uses an elite strategy to implement the selection. At each generation of the al-
96 gorithm, it selects half of the network population based on their value of φ . Some small
97 random component is added to φ to ensure mixing of the networks populations. Once half
98 of the networks are selected, we duplicate them and the duplicated networks go through a
99 round of mutations. Mutations randomly add, remove or change biochemical interactions
100 to the networks, based on the predefined biochemical grammar. In Supplement we give ini-
101 tialization files, fitness definition and mutation rates allowing to reproduce results presented
102 here. For more details on φ -evo, check Ref. 14.

103 Results

104 **Convergent evolution towards a simple biochemical implementation** 105 **of the logarithm**

106 A typical evolutionary simulation including a simulated evolutionary pathway (with value
107 of φ as a function of the number of generations) and intermediate network structures is
108 presented in Fig. 2.

109 The evolution reaches a fitness lower than 10^{-2} in less than 150 generations and stabilize
110 around 10^{-3} . This means that for such network, the average relative deviation of the Output
111 from the logarithm of the Input (up to a multiplicative constant) is less than 5%.

The simplest network topology achieving such a performance appears at generation 134,

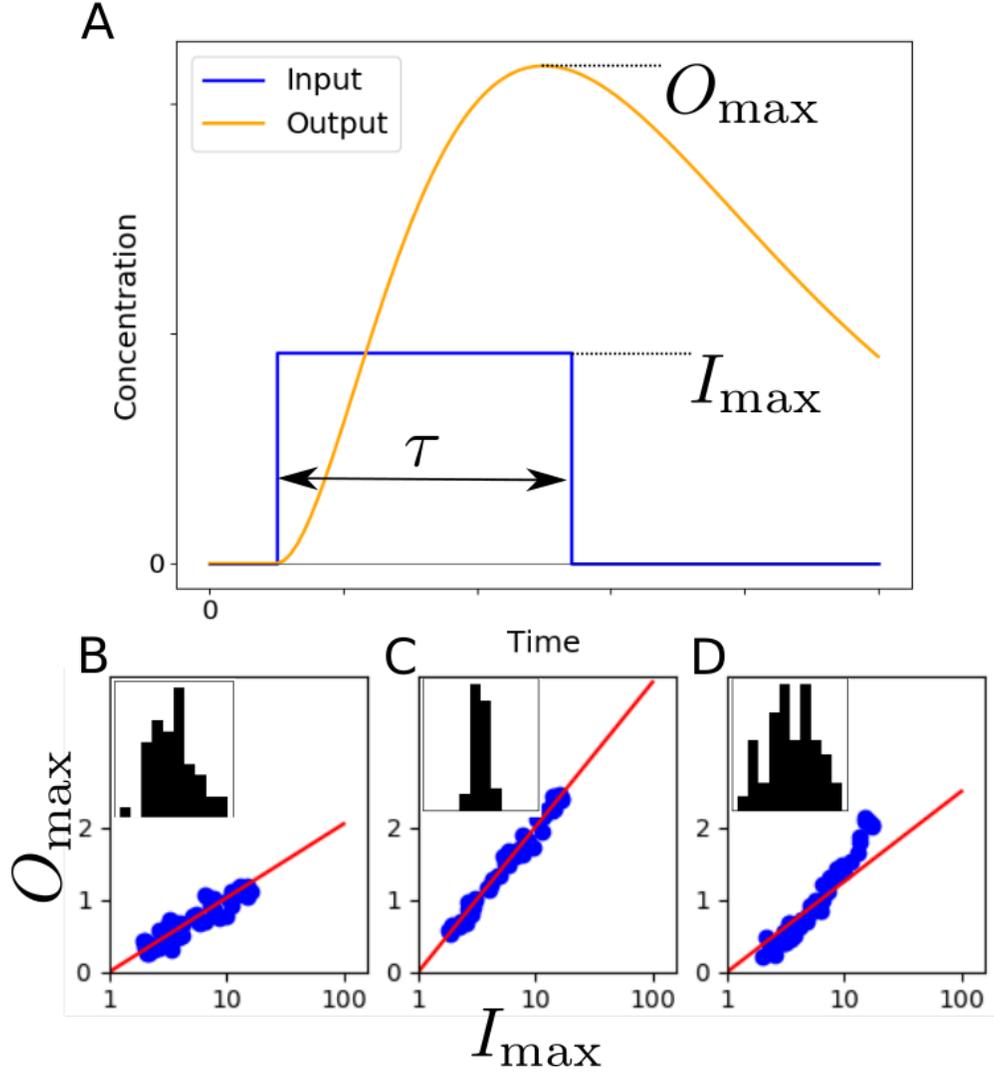


Figure 1: To compute the fitness, each network is evaluated under different randomized inputs (panel A), and the results are then gathered in a single score (panels BCD). **A.** Sketch of an input (in blue) and the response of the network in orange, the input parameters (τ and I_{\max}) are drawn from two distributions and the response is computed numerically through integration of the ODE corresponding to the network. We then retrieve I_{\max} and O_{\max} to compute the fitness score φ . **B-C-D.** Sketches of the fitness computation, blue dots represent the responses for different input on the I_{\max} , O_{\max} plane. The red line gives the ideal linear response and the inset represent the rescaled distribution the standard deviation of which we have defined as our fitness score. (note that the scale are the same for every plot). The final score of this three cases is: $\varphi_B = 3.10^{-2}$, $\varphi_C = 6.10^{-3}$ and $\varphi_D = 7.10^{-2}$. Note that this definition also naturally favors large response range as a high mean will flatten the noise and lower the variance as seen on panel C.

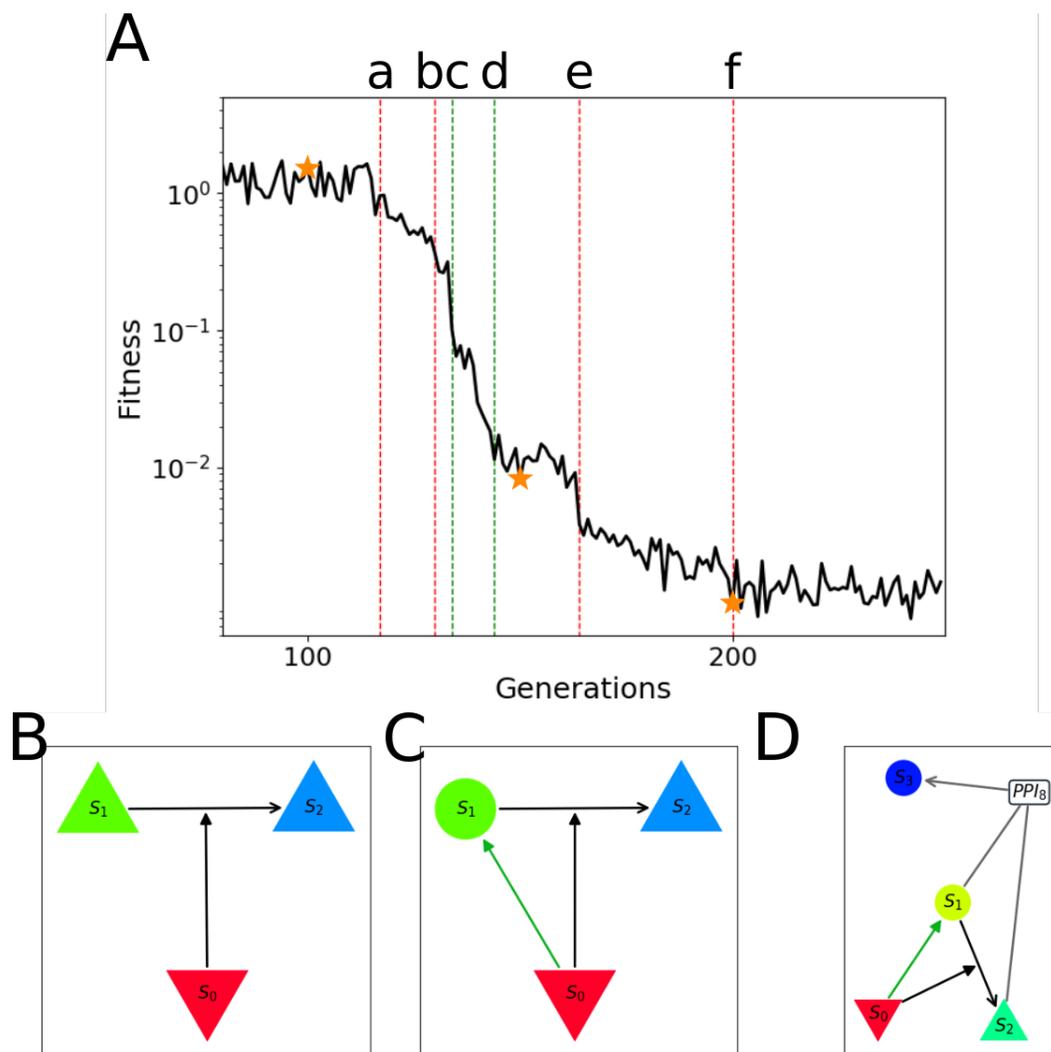


Figure 2: **A.** Excerpt of the fitness as a function of time (in generation). Vertical red lines correspond to changes of network topologies while green lines correspond to changes of major parameters. Orange stars correspond to the networks of panels B-C-D. See main text for details. **B - D.** Representation of the best network in the current population after generations 100 (B), 150 (C) and 200(D). Boxed coloured letters correspond to biochemical species, Input is the red triangle S_0 , Output is the Species S_2 . Arrows indicate possible biochemical interactions. In B, only a phosphorylation is present (red arrow), in C the input activates the transcription of the first species (green arrow). In D, the phosphorylated and unphosphorylated species form a complex, this interaction is an *ad hoc* form of repression that is very current in networks produced with φ -evo.

and is displayed in figure 2 C. Differential equations corresponding to this network are:

$$\frac{dS}{dt} = k_a \frac{I^{n_a}}{I^{n_a} + t_a^{n_a}} - k_p I \frac{S^{n_p}}{S^{n_p} + t_p^{n_p}} + k_d O - \delta_S S, \quad (2)$$

$$\frac{dO}{dt} = k_p I \frac{S^{n_p}}{S^{n_p} + t_p^{n_p}} - k_d O - \delta_O O. \quad (3)$$

112 This network work in the following way: the Input first activates the expression of a gene
 113 encoding a protein S (term $\frac{I^{n_a}}{I^{n_a} + t_a^{n_a}}$), then S is non-linearly catalyzed by I (term $I \frac{S^{n_p}}{S^{n_p} + t_p^{n_p}}$)
 114 to produce the output, here in the example of a phosphorylation. Note that the input acts
 115 both as a transcriptional activator and a catalyst (kinase).

116 Improvement of the fitness over the course of the evolution corresponds to major changes
 117 in the topology or in the parameters of the system. For the example displayed on figure 2A,
 118 at generation 117 (a), the network selects as a new output the catalyzed version of S . At
 119 generation 130 (b), S gets activated by the Input, so that the topology of the network of
 120 figure 2 C is now present. Generations 134 and 144 (c and d) correspond to increase in
 121 production rate of S and in catalytic rate by the Input, which allows for subsequent major
 122 improvement of the fitness, plateauing at fitness 10^{-2} . At generations 162 and 200 (e and
 123 f), changes in network topologies allow for small improvement of the fitness. For instance,
 124 at generation 200 a protein-protein interaction appears between S and O , allowing for an
 125 effective repression of S by O that slightly improves the fitness.

126 The motif displayed in figure 2C, that allow the simulation to reach low fitness region,
 127 appears repeatedly in our simulations. For instance, in a typical run of 20 consecutive
 128 simulations with exact same mutation parameters, 10 ended with a sufficient fitness to be
 129 considered as successful (of the order of 10^{-2} or lower). Surprisingly, each of these simula-
 130 tions display the same core topology at the heart of the network of interactions corresponding
 131 to Fig 2 C. Thus our simulations display strong convergent evolution. Variations on the nu-
 132 merical value of the parameters, other species and interactions around the functional core
 133 can be observed but do not impinge the main function of the core network. Interestingly,

134 not only the core final network is conserved, but the evolutionary path is too, with the same
135 interactions appearing in the same order (for an example compare Fig 2 with Fig. S 1 in
136 the Supplement). Over many simulations, the most common addition to the basic motif of
137 Fig 2 D is a form or another of negative control of the output on its own gene (that is the
138 species S), see e.g. Fig. 3 A.

139 Importantly, this particular network evolves because there is no restriction in φ -evo on
140 the number/nature of reactions a species can be involved in. Here, the input acts both as a
141 transcription factor and a kinase. This combination may not be possible in nature in some
142 cases, but we checked that the logarithm function is preserved if we separate the kinase and
143 transcriptional activities of the Input into an extra kinase activated transcriptionally by the
144 Input (see Supplement for an explicit example).

145 **Logarithm phenomenology and mechanism**

146 A sketch of the best topology we obtained is displayed in Fig. 3 A, which is very similar to
147 the network of Fig 2 C with an additional negative control of S by the Output².

148 Fig. 3 details the behaviour of the two species composing the network as a function of the
149 Input, compared to a logarithmic behaviour. We clearly see that such a simple network can
150 implement a precise logarithm of the Input over 4 orders of magnitude, compatible with what
151 is typically observed in processes such as immune recognition. Furthermore, this logarithmic
152 behavior works at steady state values of both S and O , keeping I constant. However it is
153 far from clear how such behaviour emerges from the interactions, requiring a more detailed
154 analysis to disentangle the influence of various biochemical interactions.

155 No simple analytic forms can be derived from the network equations, due to multiple non-
156 linearities. To proceed, we successively remove parts of the networks, and further optimize
157 parameters using our fitness function to identify which interactions are most crucial at steady

²The precise form of this control is subject to variability between the different simulations, it may be a transcriptional regulation, a protein protein interaction that form a dimer with low dissociation or high degradation or a direct enzymatic degradation, in all these cases however, the negative influence is manifest.

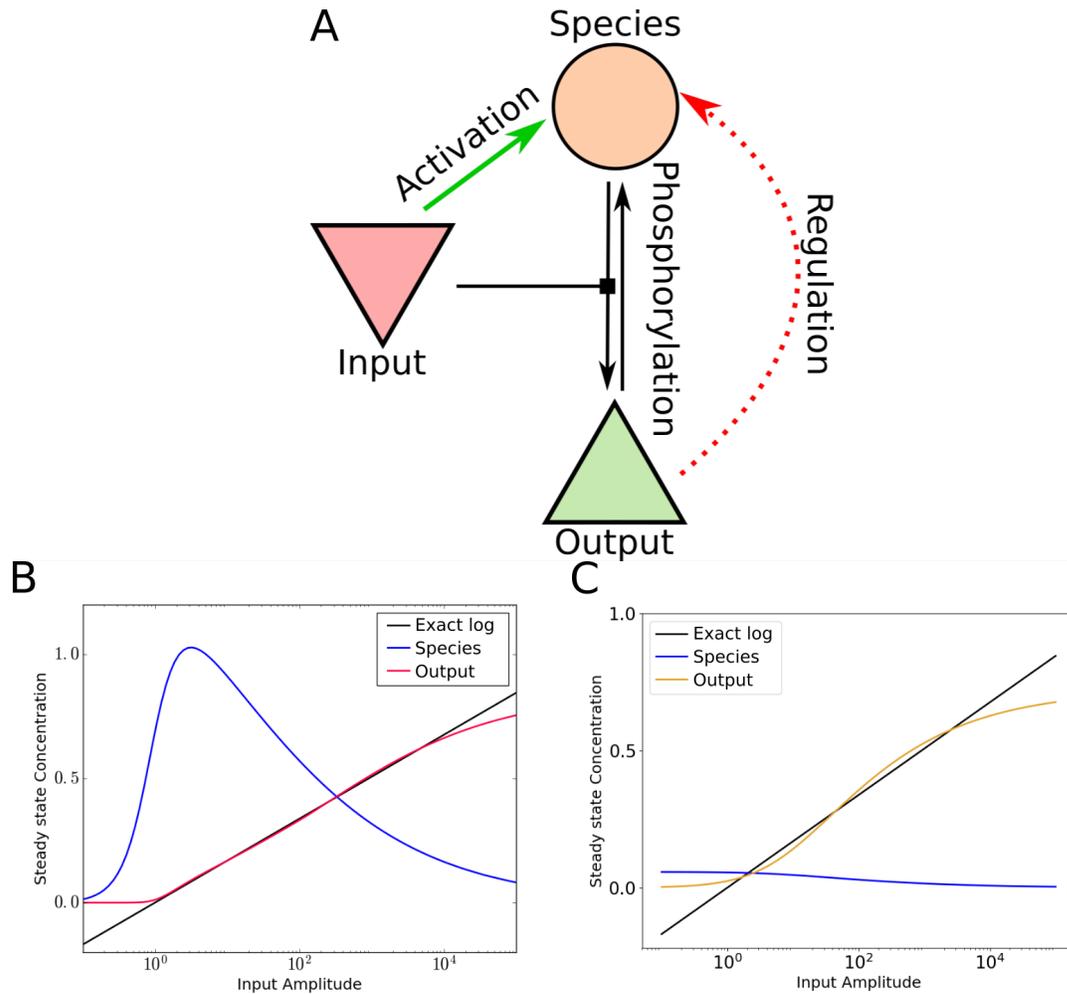


Figure 3: **A.** Convergent network structure evolved to produce a logarithmic response. Typically, the input activates or produces a species that should then be phosphorylated to produce the output. To achieve an even better fitness, the evolved networks then may harbor a form or another of regulation (that is regulation, complexation or enzymatic degradation) of the output on its non-phosphorylated form. **B.** Steady state concentration of the species A and B as a function of the input concentration (in log scale), the black line indicates the logarithmic function on which we fit our output concentration over the $[10^0, 10^4]$ range. Note the complex behavior of the precursor species the non-monotonicity of which is crucial to perform the desired function. The mean error between the output and the desired logarithm is of the order of 1% on the range of interest. **C.** Same as **B** but for the case where both activation and regulation has been deleted. Note the weak range of variation of S and the deviation of O from the desired target. Still, the approximation of a logarithm is roughly correct, the mean error from the logarithm being of the order of 10%.

158 state.

159 As expected from the example of Figure 2, the negative regulation of species S by the
160 Output is not crucial, and the network behaviour can be almost logarithmic without it.
161 Further removing the activation within the networks and optimizing parameters yields a
162 behaviour that, without being logarithmic strictly speaking, still presents strong similarity
163 with a logarithm upon 4 orders of magnitude as depicted in Figure 3 C.

164 This behaviour can be analytically understood by considering the following limits of
165 equations 2 and 3. On the one hand, at steady state for 3, assuming we are far from
166 saturation for the catalytic interaction, we clearly have

$$O \propto IS^{n_p} \quad (4)$$

167 On the other hand, substituting steady state equation 3 in 2 and assuming constant
168 production, we have

$$\delta_S S + \delta_O O = k_a \quad (5)$$

169 which is an effective conservation law stemming from the fact that S is transformed into O .

Now let us remark that while I varies over 4 order of magnitudes, S and O vary only on a limited range due to their logarithmic nature. From equation 4 we can write $S = \left(\frac{O}{I}\right)^{\frac{1}{n_p}}$ and thus injecting this solution in equation 5:

$$\delta_O O = k_a - \delta_S \left(\frac{O}{I}\right)^{\frac{1}{n_p}} \quad (6)$$

$$\delta_O O = k_a - \delta_S \exp\left(\frac{1}{n_p}(\log O - \log I)\right) \quad (7)$$

$$\delta_O O = k_a - \delta_S - \frac{\delta_S}{n_p} \log O + \frac{\delta_S}{n_p} \log I \quad (8)$$

$$O = \frac{k_a - \delta_S}{\delta_O} + \frac{\delta_S}{\delta_O n_p} \log I \quad (9)$$

170 where we Taylor expand the exponential up to first order in $\frac{1}{n_p}$ and use the fact that $\log O$
171 is negligible in the last expression since, as we noticed, O is small by construction while I
172 can take large values.

173 Equation 9 promises us a perfect logarithm (up to first order) if $k_a = \delta_S$ so that the first
174 term in the r.h.s. cancel. Moreover we also see that increasing n_p should be counterbalance
175 by either lowering δ_O or increasing δ_S .

176 To verify this prediction, we numerically solve the equation 6 with the choice of coefficients
177 described above and check that it effectively gives a logarithmic approximation when $n_p \rightarrow$
178 $+\infty$ as shown in supplement.

179 **Parameter dependency and range constraints**

180 While we have identified in the previous section the core mechanism for log-sensing, it
181 is worth examining more carefully what range of parameters give consistent logarithmic
182 behaviour.

183 To further clarify parameter dependency, we turn back to the full network displayed in
184 Fig. 3 A, and study its behaviour when parameters are randomly chosen. For 5000 parameter
185 sets, we compute the fitness φ and the range of output defined as the difference between the
186 maximum and the minimum value of the Output when the Input varies from 1 to 10^4 and
187 present the results in the Fig 4. As we hope to find networks that harbour a low fitness but
188 a high range, the better networks will be found in the bottom right corner (corresponding to
189 maximum range, minimum fitness). The optimized network of Fig. 3 A is indicated with a
190 red dot, and clearly displays parameter optimization with respect to networks with random
191 parameters. Still it is interesting to see that random sets of parameters may reach interesting
192 regions of the functional plane.

193 We define two subsets of interest. Blue dots identify a sub population of networks clus-
194 tering on a bottom-right front. Parameter sets on this front achieve small values of the ratio
195 between fitness and range. Small fitnesses lower than 10^{-2} with ranges of order 1 are still

196 reached on this front, even with parameters chosen randomly. We define a second subset of
 197 parameters with an orange window of networks with both small fitness and high range, from
 198 our 5000, 260 networks reach this threshold thus indicating that $5.2 \pm 0.6\%$ of the parameter
 199 space produced an acceptable logarithmic approximation.

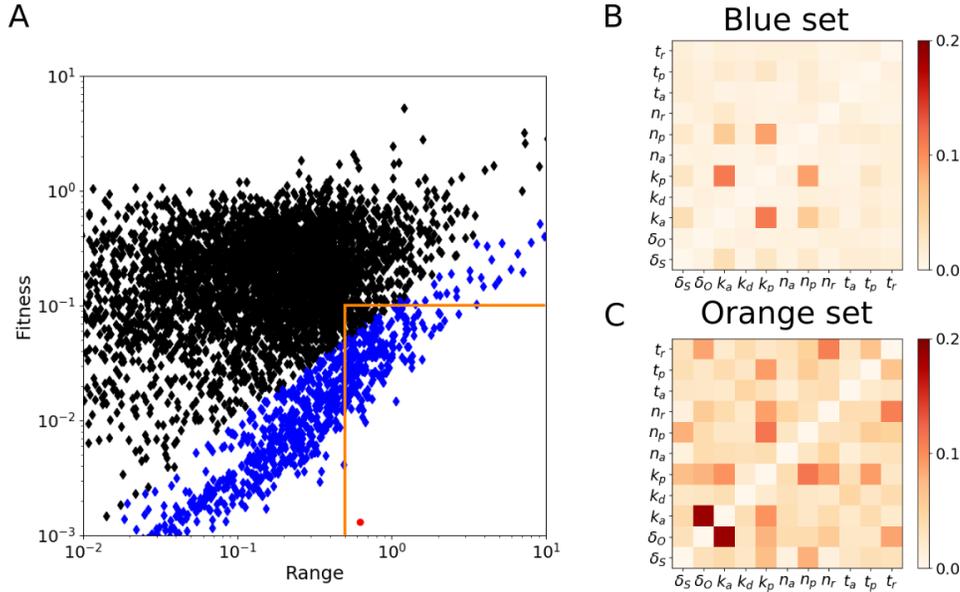


Figure 4: **A.** Fitness and range of response (defined as the difference between maximum and minimum Output values when the Input is varied) for 5000 random set of parameters (see details in Supplement for ranges of variation). Better networks are in the lower right corner. The red dot correspond to the optimized set of parameters for network of Fig. 3 A. We focus on two subsets of parameters. Blue dots are network close to minimal fitness over range ratio, corresponding to a linear front (~ 870 parameter sets), The orange window are parameter sets that have both good fitness and range (~ 260 parameter sets). **B-C.** Mutual information between parameters for the two parameter sets defined in Panel A. Warmer colours define stronger correlations, see the main text for the full detail of these correlations. The second set being smaller is also more noisy.

200 We now consider those two subsets of parameters to determine how parameters are con-
 201 strained. Distribution of each parameter inside a set does not reveal any particular pattern
 202 (see figure 10 in supplement), which steered us to look for second order correlation.

203 We compute the mutual information score for all couple of parameters for those two
 204 subsets, Fig 4 B-C (see Supplement for more details). High mutual information between
 205 two parameters means that their distributions are intertwined, thus indicating correlation

206 between their values from set to set and revealing constraints on the biochemical interactions.

207 Most couples of parameters have low mutual information, indicating lack of any strong
208 constraints between them. However, we identify three strong constraints for the network to
209 perform well. The blue set presents two constraints. A first one is seen on the activation rates:
210 $k_p + k_a < 1$ (Fig in Supplement), that constraints the sum of the production rate of S with
211 the maximum catalytic rate. Indeed, if those rates are too high, the system will be blocked
212 in the saturated regime and cannot take advantage on the non-linearity of the interactions,
213 this is thus consistent with our analytical study above.

214 Another less intuitive constraint is observed on the phosphorylation parameters k_p, n_p
215 see Fig. 5 : there is basically almost no selected set in the region above the first diagonal
216 $n_p = k_p$ (in rescaled units). To get a better understanding of the origin of this constraint, we
217 took the network of Fig. 3 C, ran several simulations with increasing values of n_p , and k_p ,
218 and study the behaviour of S . Increases of n_p clearly moves the behaviour of S towards the
219 right, with saturation appearing for small Input concentration. Conversely, increases of k_p
220 moves the behaviour of S towards the left with less saturation for small Input concentration.
221 Thus to get a linear behaviour of S over the range of concentration considered, from a
222 given set of optimized parameters, an increase of n_p can be essentially compensated with an
223 increase of k_p . However it is also clear that increases of n_p moves the curve more significantly
224 than increases of k_p , so that we expect there is a limit for which increases k_p can no longer
225 compensate n_p , explaining why the region where n_p is too big w.r. to k_p is forbidden.

226 The orange set allow us to identify one last strong constraint: $k_a > d_o$ asking that
227 the degradation rate should be smaller than the production. This constraint is easy to
228 understand as it ensures that the concentration of output may be high and thus allows for
229 a large range in the output concentration.

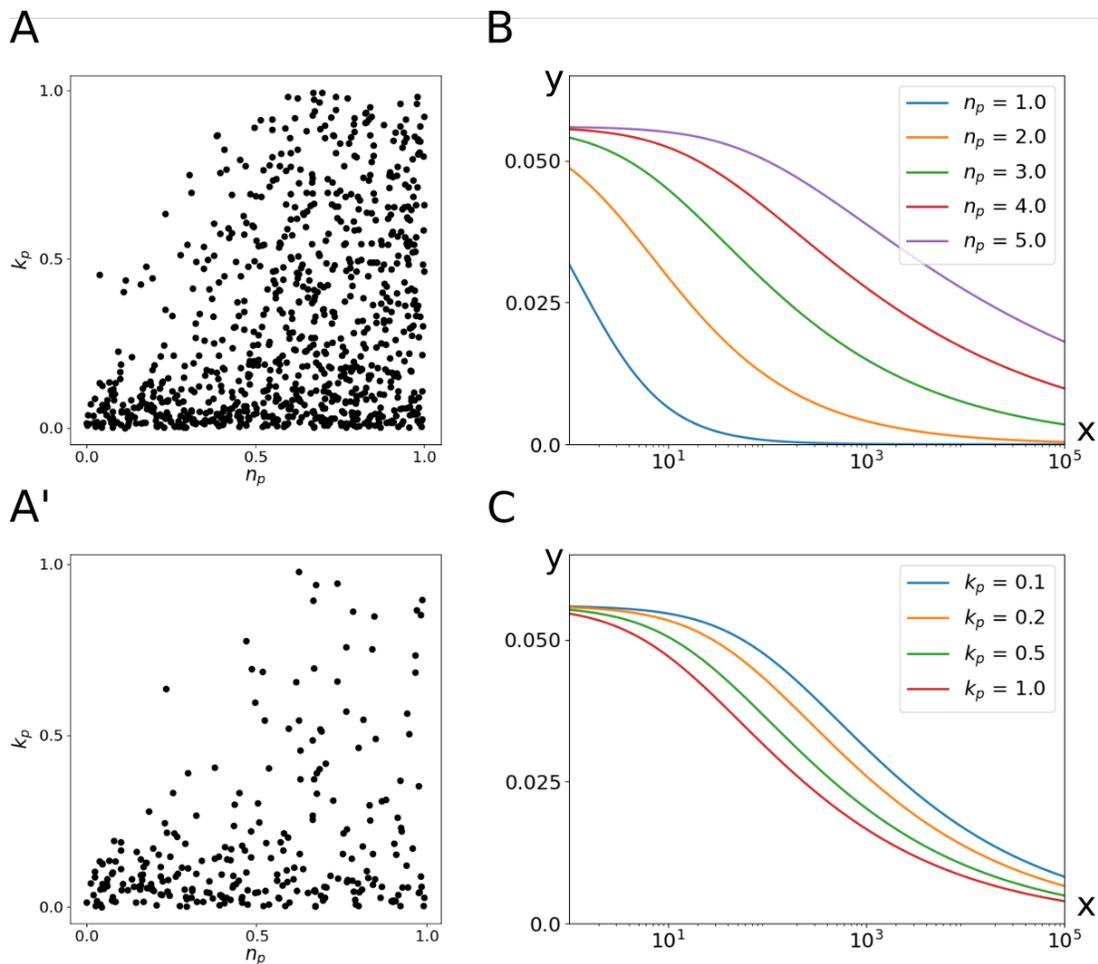


Figure 5: **A.** Relation between k_p and n_p for the network in the blue (A) and orange (A') set. The constraint is manifest although subtle. **B.** Influence of n_p on the solution of the network 3C, the hill coefficient of the phosphorylation thus allow to have a more and more linear response but shift the linear part on the high input regime. **C.** Influence of k_p on the solution of the network 3C, the rate of the phosphorylation thus correspond to a shift on the low input that allow to compensate for the shift due to the hill coefficient. Note however that this shift seems logarithmic in k_p (similar shift for our 4 curves that have a log spacing).

230 Discussion

231 We have used in silico evolution (the φ -evo algorithm) to design a biochemical log-sensor.
232 Results of our simulations show that an almost perfect logarithmic sensor over 4 orders of
233 magnitude can be obtained by combining feedback and feedforward interactions with a non
234 linear interaction catalyzed by the Input.

235 The key log-sensitivity comes from Eq. 6 relating I and O . It is clear here that such non-
236 linear catalysis could arguably constitute one of the simplest (if not the simplest) mechanism
237 implementing a logarithmic response. Such process is not unrealistic. A first possibility
238 is to have multiple phosphorylations as suggested in REF. Another easy way to get high
239 powers in Eq. 4 is to assume that biochemical species S have to form multimers before
240 being catalyzed in the active form O by I . Some Input can then trigger internalization of
241 receptors and subsequent signalling, effectively implementing log-sensing. Such processes are
242 common in biology. For instance in G protein-coupled receptors signalling, agonist triggers
243 oligomerizations of receptors¹⁵. Another context is immunology where cellular receptors are
244 well-known to form multimers at the surface of the cell¹⁶ and are phosphorylated to trigger
245 response¹⁷. Such networks (with internal feedback) could thus work directly in the log range
246 for Input concentration, ensuring broader response or non-trivial scaling laws¹⁸.

247 More generally, assuming a relation such as Eq. 4 holds irrespective of specific biochem-
248 istry, we can expect O and S to vary logarithmically with I as long as both O and S are
249 constrained to vary on a predefined range. Here, the latter constraint stems from the ef-
250 fective conservation law in equation 5, which imposes limits on both concentrations at the
251 same time. An alternative way to perform this would simply be that neither S and O are
252 degraded, nor S is produced, so that Eq. 4 would directly be mass conservation. We have
253 also identified parameter dependencies for this system to work correctly, key of which is
254 the necessity to play on the non linearity of the hill functions, that is avoid both high and
255 low saturation regimes. The constraint that n_p should be small enough with respect to k_p
256 also suggests that the non-linearity should not be too high. This is of biological relevance

257 in a case where n_p is the number of receptors forming multimers, which thus should stay
258 relatively low.

259 One can also refine the system to get a “perfect” log, by addition of other feedbacks that
260 are found by φ -evo. Of course, it is not entirely clear why a cell would need to take an exact
261 log, versus a response that would only be log-like. In our case, extra feedbacks can rather
262 be thought of as post-processing of a core signal sensitive over orders of magnitudes, and
263 as such can be likely performed otherwise downstream of the Output. This vision is very
264 different from previous proposals, where feedback need to be carefully engineered to ensure
265 log sensitivity^{9,10}. While it is not clear which solution is more amenable to be found by
266 biological evolutionary tinkering, solutions presented here are *de facto* evolvable, within the
267 range of allowed biochemical interactions and mutations in our simulations. The dual fact
268 that we observe very strong convergent evolution towards this solution and that receptor
269 multimerization is a possible common mechanism to implement it suggests that the solution
270 we propose might be easier to find by evolution, and therefore potentially ubiquitous.

271 Supporting Information Description

- 272 • A short document gathering figures, details and explanations not important enough to
273 be embedded in the main text and refer in it as supplement.
- 274 • Initialization files of the main run described in the text (to be used with φ -evo software).

275 Note also that all our source code files for the optimization or simulation are written in
276 Python and may be obtained through mailing the authors.

277 Acknowledgement

278 References

- 279 (1) Feinerman, O.; Germain, R. N.; Altan-Bonnet, G. *Molecular Immunology* **2008**, *45*,
280 619–631.
- 281 (2) Altan-Bonnet, G.; Germain, R. N. *PLoS Biology* **2005**, *3*, e356.
- 282 (3) François, P.; Voisinne, G.; Siggia, E. D.; Altan-Bonnet, G.; Vergassola, M. *Proc Natl*
283 *Acad Sci U S A* **2013**, *110*, E888–97.
- 284 (4) Lalanne, J.-B.; François, P. *Physical Review Letters* **2013**, *110*, 218102.
- 285 (5) Kalinin, Y. V.; Jiang, L.; Tu, Y.; Wu, M. *Biophysj* **2009**, *96*, 2439–2448.
- 286 (6) Milo, R.; Phillips, R. *Cell Biology by the Numbers*; Garland Science, 2015.
- 287 (7) Koch, A. L. *Journal of Theoretical Biology* **1966**, *12*, 276–290.
- 288 (8) Frank, S. A. *Biology Direct* **2013**, *8*, 221.
- 289 (9) Daniel, R.; Rubens, J. R.; Sarpeshkar, R.; Lu, T. K. *Nature* **2013**, *497*, 619–623.
- 290 (10) Olsman, N.; Goentoro, L. *Proc Natl Acad Sci U S A* **2016**, *113*, E4423–30.
- 291 (11) Daniels, B. C.; Nemenman, I. *PLoS ONE* **2015**, *10*, e0119821.
- 292 (12) François, P. *Seminars in cell & developmental biology* **2014**, *35*, 90–97.
- 293 (13) Knight, W. There’s a big problem with AI: even its creators can’t ex-
294 plain how it works. [https://www.technologyreview.com/s/604087/
295 the-dark-secret-at-the-heart-of-ai/](https://www.technologyreview.com/s/604087/the-dark-secret-at-the-heart-of-ai/).
- 296 (14) Henry, A.; Hemery, M.; François, P. *PLoS computational biology* **2018**, *14*, e1006244.

- 297 (15) Li, Y.; Shivnaraine, R. V.; Huang, F.; Wells, J. W.; Gradinaru, C. C. *Biophysical*
298 *Journal* **2018**, *115*, 881–895.
- 299 (16) Taylor, M. J.; Husain, K.; Gartner, Z. J.; Mayor, S.; Vale, R. D. *Cell* **2017**,
- 300 (17) Kersh, E. N.; Shaw, A. S.; Allen, P. M. *Science* **1998**, *281*, 572–575.
- 301 (18) Tkach, K. E.; Barik, D.; Voisinne, G.; Malandro, N.; Hathorn, M. M.; Cotari, J. W.;
302 Vogel, R.; Merghoub, T.; Wolchok, J.; Krichevsky, O.; Altan-Bonnet, G. *eLife* **2014**,
303 *3*, e01944.

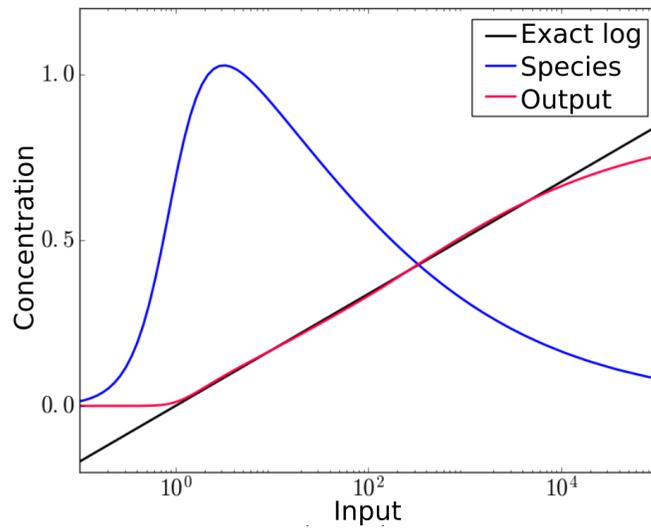
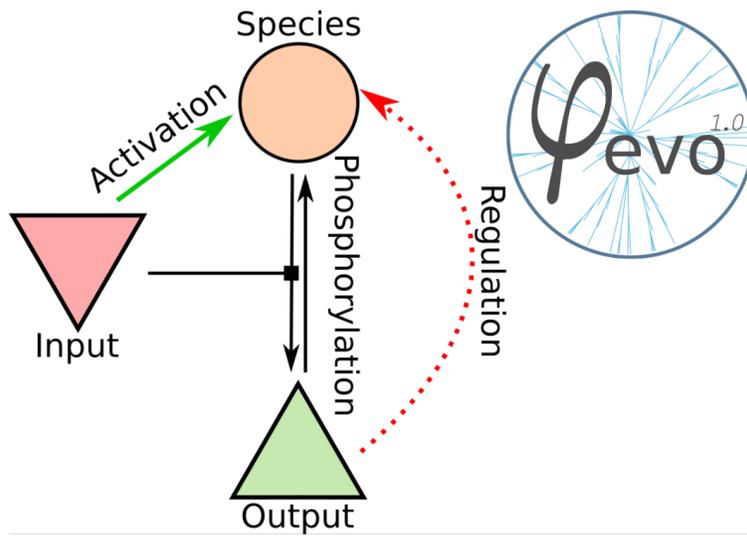


Figure 6: TOC figure