Studying the evolution of bacterial micro-organisms by modeling and numerical simulation approaches

Charles Rocabert

N°d'ordre NNT : 2017LYSEI106

THESE de DOCTORAT DE L'UNIVERSITE DE LYON
opérée au sein de
L'Institut National des Sciences Appliquées de Lyon

Ecole Doctorale N° EDA-512
Informatique et Mathématiques de Lyon (InfoMaths)

Spécialité/ discipline de doctorat : Informatique

Soutenue publiquement le 17/11/2017, par :
Charles Rocabert

Étude de l'évolution des micro-organismes bactériens par des approches de modélisation et de simulation informatique

---

Devant le jury composé de :

Samuel Bernard	Chargé de recherche HDR, CNRS	Examinateur
Guillaume Beslon	Professeur, INSA de Lyon	Directeur de thèse
Bahram Houchmandzadeh	Directeur de recherche, CNRS	Rapporteur
Carole Knibbe	Maître de conférences, INSA de Lyon	Directrice de thèse
Jean-Baptiste Mouret	Directeur de recherche, INRIA	Président du jury
Olivier Tenaillon	Directeur de recherche, INSERM	Rapporteur
Karine Van Doninck	Professeur, Université de Namur	Examinatrice

Département FEDORA – INSA Lyon - Ecoles Doctorales – Quinquennal 2016-2020

SIGLE	ECOLE DOCTORALE	NOM ET COORDONNEES DU RESPONSABLE
CHIMIE	**CHIMIE DE LYON** **http://www.edchimie-lyon.fr** Sec : Renée EL MELHEM Bat Blaise Pascal 3e etage secretariat@edchimie-lyon.fr Insa : R. GOURDON	M. Stéphane DANIELE Institut de Recherches sur la Catalyse et l'Environnement de Lyon IRCELYON-UMR 5256 Équipe CDFA 2 avenue Albert Einstein 69626 Villeurbanne cedex directeur@edchimie-lyon.fr
E.E.A.	**ELECTRONIQUE, ELECTROTECHNIQUE, AUTOMATIQUE** **http://edeea.ec-lyon.fr** Sec : M.C. HAVGOUDOUKIAN Ecole-Doctorale.eea@ec-lyon.fr	M. Gérard SCORLETTI Ecole Centrale de Lyon 36 avenue Guy de Collongue 69134 ECULLY Tél : 04.72.18 60.97 Fax : 04 78 43 37 17 Gerard.scorletti@ec-lyon.fr
E2M2	**EVOLUTION, ECOSYSTEME, MICROBIOLOGIE, MODELISATION** **http://e2m2.universite-lyon.fr** Sec : Sylvie ROBERJOT Bât Atrium - UCB Lyon 1 04.72.44.83.62 Insa : H. CHARLES secretariat.e2m2@univ-lyon1.fr	M. Fabrice CORDEY CNRS UMR 5276 Lab. de géologie de Lyon Université Claude Bernard Lyon 1 Bât Géode 2 rue Raphaël Dubois 69622 VILLEURBANNE Cédex Tél : 06.07.53.89.13 cordey@univ-lyon1.fr
EDISS	**INTERDISCIPLINAIRE SCIENCES-SANTE** **http://www.ediss-lyon.fr** Sec : Sylvie ROBERJOT Bât Atrium - UCB Lyon 1 04.72.44.83.62 Insa : M. LAGARDE secretariat.ediss@univ-lyon1.fr	Mme Emmanuelle CANET-SOULAS INSERM U1060, CarMeN lab, Univ. Lyon 1 Bâtiment IMBL 11 avenue Jean Capelle INSA de Lyon 696621 Villeurbanne Tél : 04.72.68.49.09 Fax :04 72 68 49 16 Emmanuelle.canet@univ-lyon1.fr
INFOMATHS	**INFORMATIQUE ET MATHEMATIQUES** **http://edinfomaths.universite-lyon.fr** Sec :Renée EL MELHEM Bat Blaise Pascal, 3e étage Tél : 04.72. 43. 80. 46 Fax : 04.72.43.16.87 infomaths@univ-lyon1.fr	M. Luca ZAMBONI Bâtiment Braconnier 43 Boulevard du 11 novembre 1918 69622 VILLEURBANNE Cedex Tél :04 26 23 45 52 zamboni@maths.univ-lyon1.fr
Matériaux	**MATERIAUX DE LYON** **http://ed34.universite-lyon.fr** Sec : Marion COMBE Tél:04-72-43-71-70 –Fax : 87.12 Bat. Direction ed.materiaux@insa-lyon.fr	M. Jean-Yves BUFFIERE INSA de Lyon MATEIS Bâtiment Saint Exupéry 7 avenue Jean Capelle 69621 VILLEURBANNE Cedex Tél : 04.72.43 71.70 Fax 04 72 43 85 28 Ed.materiaux@insa-lyon.fr
MEGA	**MECANIQUE,ENERGETIQUE,GENIE CIVIL,ACOUSTIQUE** **http://edmega.universite-lyon.fr/** Sec : Marion COMBE Tél:04-72-43-71-70 –Fax : 87.12 Bat. Direction mega@insa-lyon.fr	M. Philippe BOISSE INSA de Lyon Laboratoire LAMCOS Bâtiment Jacquard 25 bis avenue Jean Capelle 69621 VILLEURBANNE Cedex Tél : 04.72 .43.71.70 Fax : 04 72 43 72 37 Philippe.boisse@insa-lyon.fr
ScSo	**ScSo*** **http://ed483.univ-lyon2.fr/** Sec : Viviane POLSINELLI Brigitte DUBOIS Insa : J.Y. TOUSSAINT Tél : 04 78 69 72 76 viviane.polsinelli@univ-lyon2.fr	M. Christian MONTES Université Lyon 2 86 rue Pasteur 69365 LYON Cedex 07 Christian.montes@univ-lyon2.fr

*ScSo : Histoire, Géographie, Aménagement, Urbanisme, Archéologie, Science politique, Sociologie, Anthropologie

Remerciements

Il y a neuf ans, je reprenais mes études à Lyon avec l'intention de faire de la recherche en biologie évolutive. Le monde universitaire et académique m'était alors complètement étranger. Aujourd'hui encore, chaque nouveau pas dans cet univers est une découverte. Ce sont toutes les rencontres, scientifiques et humaines, qui ont fait de ces neuf années les plus passionnantes et les plus riches de ma vie. Je prends donc le temps ici d'évoquer les personnes qui ont compté pour moi, sur une période plus large que mes seules années de thèse.

Tout d'abord, je remercie chaleureusement Jean-Baptiste Mouret et Karine Van Doninck d'avoir accepté de participer à mon jury de thèse, ainsi que Bahram Houchmandzadeh et Olivier Tenaillon pour leur relecture attentionnée de ce manuscrit et leur retour positif.

Alors que j'étais en quête d'un stage de master 1, j'envoyai un mail laconique ("*Je suis très intéressé par les travaux de votre équipe (COMBINING). Serait-il possible de vous rencontrer afin de discuter d'un éventuel stage ?*") à un certain G_ _ll_ _ _ (barbu à l'époque) qui m'invita immédiatement à en discuter autour d'un café. Lorsque je lui présentai mon projet : étudier l'évolution de la stochasticité d'expression des gènes avec ævol, G_ _ll_ _ _ accepta sans hésitation (c'était avant qu'il ne découvre que je vouais un culte à K_p_ _ _). Cela fait maintenant six ans que je traîne dans les couloirs de Beagle, cela m'a laissé le temps de comprendre pourquoi cette équipe est simplement la plus humaine et la plus soudée que je connaisse. À l'évidence, c'est grâce à toi Guillaume. Ton respect pour les goûts et les intérêts de chacun·e, le fait que tu privilégies toujours le bien-être des membres de l'équipe, sont la source de l'ambiance chaleureuse et unique qui règne à Beagle. En quatre années de thèse, j'ai apprécié la disponibilité et l'ouverture d'esprit dont tu fais preuve, malgré un emploi du temps chargé. Tu m'as toujours fait confiance et laissé libre de mes choix, tout en étant là pour m'éclairer, me motiver et surtout, me faire prendre du recul. Tu t'en doutes certainement, ton approche et ta philosophie de la modélisation ont eu une influence immense sur moi.

Comment remercier Guillaume sans remercier Carole, leur complémentarité est évidente. Carole, cette thèse n'aurait jamais aboutie sans ta contribution décisive. Tu as la capacité de t'investir pleinement dans un travail de recherche, en ne laissant rien au hasard, et tu m'as transmis la pratique rigoureuse et objective de la science. Mais ce portrait serait bien réducteur, car tu t'es toujours souciée de ma situation, et tu as toujours été là pour me soutenir et me conseiller.

En début de thèse, alors que les lignées Beagle et Dracula s'étaient un peu plus mêlées, Carole a demandé à Samuel de relire un obscur manuscrit rempli d'équations. Avec Sam, nous avons construit un vrai travail collaboratif, au point que je ne puisse dire de qui proviennent certaines idées de notre modèle. Sam, je n'oublierai jamais la confiance sereine que tu as eu en notre projet dès le début – ainsi que notre tentative d'entraînement au semi-marathon. J'espère que nous continuerons à travailler ensemble.

Carole, Guillaume et Samuel, je revendiquerai toujours votre héritage. Pour tout cela, et plus encore, merci.

Ah, l'équipe Beagle, et son esprit d'indépendance – voire parfois de résistance – qui y règne et que j'affectionne tant. Mettons à part les blagues douteuses de Huge Belly, qui en a traumatisé plus d'un·e. Je pense au thésard·e·s bien-sûr : Priscilla, Vincent, Ilya, Sergio, Alexandre, Audrey, Marie, Alvaro, Jules, Marine, Yoram, avec qui j'ai passé tant de bons moments. J'ai tout de même une pensée particulière pour Jonathan et Vincent, avec qui nous avons partagé nombre de grenades lacrimogènes. Hugues, qui a toujours été là pour m'écouter et me conseiller, a été l'un des organisateurs de l'école la plus passionnante à laquelle j'ai assisté[1]. Ton humour me manquera, Hugues ; que tu le veuilles ou non, tu es un des piliers de l'équipe Beagle. Merci à David – qui m'a tant appris en informatique et en développement logiciel –, Éric, Christophe, Hédi, Maurizio, Jaap, Gaëlle et Nicolas, qui a eu le malheur de m'avoir comme enseignant. Je remercie aussi Caroline, qui a dû réparer bien des bourdes administratives de ma part.

Dracula n'est pas en reste. Je pense notamment à Fabien, qui m'a souvent écouté, aidé et encouragé. Je remercie également Olivier, qui connaît mon intérêt pour le rôle du hasard dans le fonctionnement cellulaire, et qui m'a accueilli deux fois en stage.

Beagle, Dracula, vous me manquerez.

Le monde scientifique est petit, il y a un peu du LIRIS dans Beagle, et un peu du LEHNA dans le LIRIS. À ce croisement où la bière bordelaise coule, naquit MoRIS. Jérôme Gippet, mon ami de fac de toujours, et Serge Fenet, forgeron à ses heures perdues (ou bouilleur de cru ?), je vous dois cette belle aventure. Un jour, Jérôme m'a parlé d'un modèle de propagation d'espèce invasive qu'il avait en tête. Quelques semaines plus tard était né MoRIS. D'un simple jouet, ce modèle est devenu le fruit d'un véritable travail collaboratif, avec financement, conférences et publications à l'appui. Jérôme, Serge, je n'oublierai jamais nos séances de travail au milieu des Alpes et notre petit voyage à Bordeaux. MoRIS n'est qu'au début d'une longue vie. Je remercie également Benjamin Galliot, pour son expertise en interface graphique, ainsi que Jean-Paul Léna, du LEHNA (ce n'est certainement pas un hasard) et Bernard Kauffman (du LEHNA aussi), pour nos discussions autour de votre infâme café.

Un peu plus proche des Alpes, je me tourne maintenant vers le monde des boîtes de Petri et des micro-pipettes, pour remercier chaleureusement Dominique Schneider, Jessika Consuegra, et Otmane Lamrabet, ainsi que tous les membres de l'équipe grenobloise pour

[1] http://ecoleporquerolles.inria.fr/index.html

les vins-fromages du vendredi midi. Je remercie également tous les membres du projet EvoEvo, que j'ai eu la chance de côtoyer tout au long de cette thèse.

À propos de rencontres, beaucoup de choses se sont tramées dans la résidence étudiante Puvis de Chavannes. Je pense tout particulièrement à mon ami, Pierre Charrier, avec qui j'ai eu des discussions scientifiques passionnées et partagé tant de lectures. Nous avons découvert l'informatique ensemble, sur notre temps libre, alors que nous étudiions la biologie. Cet apprentissage autodidacte a été déterminant pour nous deux. Entre autres, Pierre avait dans sa chambre de 9 m^2 une antenne wifi DIY en boîte de conserve, une imposante collection d'insectes morts, et un équipement de culture hydroponique dans un placard – pour des tomates. Je pense aussi à mon autre ami d'infortune étudiante, Wam Human. Nous avons beaucoup de souvenirs en commun, entre randonnées, séances de grimpe, joggings à 6 heures du matin en décembre, ainsi que toutes nos soirées à partager l'unique cuisine de notre étage avec Arnaud, Jacquou, Pierre et tant d'autres ; tout cela sous le regard bienveillant de Jean-Luc, le gardien de nuit de la résidence.

Que serais-je devenu politiquement si Ivo Vassilev et Stan Chabert n'avaient pas été là ? Votre esprit militant a influé sur toutes nos activités, que ce soit en manif, en amphi, ou en soirée. Je n'oublierai jamais les AGs de lutte à Lyon 1, les interventions de Stan, et la fameuse réunion du POI. C'est aussi grâce à vous deux – et à vos colocs respectives – que nous avons pu nous réunir si souvent pendant les années de licence. Merci aussi à Perrine, pour les virées à Limas, ainsi qu'à Maxence, Magali, Julie, Clément, Mélissa et tous les autres pour les bons souvenirs et les soirées au petit parc.

Avant de me tourner vers ma famille, il y a trois autres personnes que j'aimerais citer. Annabelle, même si nous nous sommes perdus de vue depuis, c'est toi qui m'a fait découvrir le travail de Jean-Jacques Kupiec, et qui a suscité ma passion pour ce sujet en biologie. C'est aussi sous ton impulsion que nous avons fait notre premier stage ensemble, dans l'équipe d'Olivier Gandrillon. Je te remercie vraiment pour cela. Je pense aussi à Vincent Lacroix, qui représente pour moi ce que devrait être l'enseignement universitaire : un travail d'émancipation collective, où tout le monde est tiré vers le haut. Une dernière personne que je veux remercier est Isabelle Illouz, qui s'est battue chaque année pour que je puisse renouveler ma bourse étudiante et sans qui je n'aurais jamais pu poursuivre mes études.

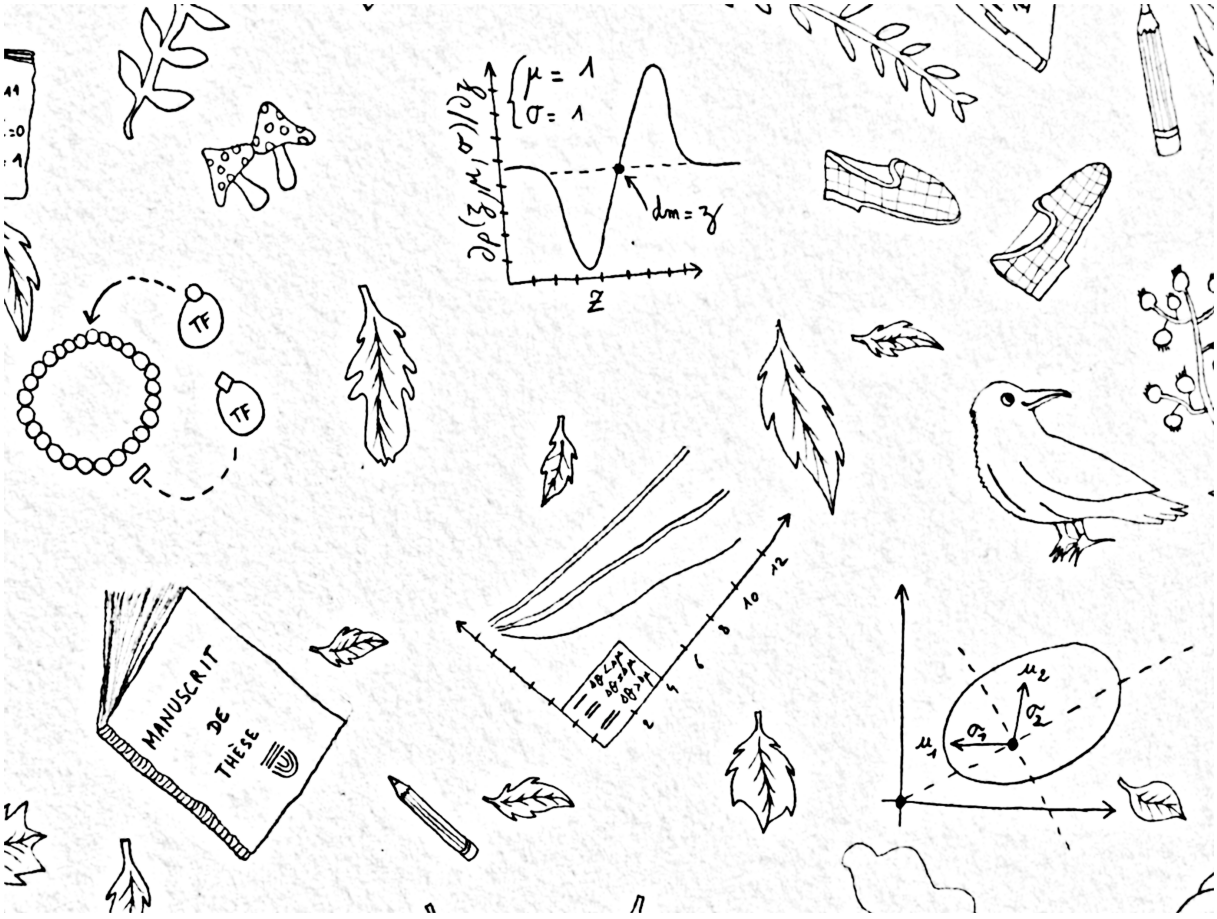
Un grand merci à Françoise, la maman de Manuèla, pour avoir toujours été là pour nous épauler, et nous avoir permis de respirer l'air frais des monts d'Ardèche toutes ces années. Merci aussi à Alain pour les belles échappées dans les-dits monts, grâce à son quad.

Je me tourne maintenant vers mes deux soeurs, Jeanne et Louise. Je vous dois la reprise de mes études. Vous m'avez révélé l'émancipation qu'apportent les études universitaires. L'élément déclencheur fût un voyage au Royaume-Uni, mais votre influence oeuvrait depuis longtemps déjà. Je ne vous remercierai jamais assez pour cela.

Maman et Papa, merci de m'avoir toujours laissé libre de mes choix, et de m'avoir toujours soutenu, quelque fût la direction que je pris. Papa, avec tes piles de Science & Vie, et

Maman avec ton intense activité syndicale, vous m'avez appris à avoir un esprit critique. Je sais que vous êtes fiers de moi pour cette thèse, mais au bout du compte, tout cela est de votre fait.

Manuèla, mon niabs, nous nous sommes rencontrés dans une résidence étudiante, et depuis nous partageons tout. Je ne peux imaginer la vie sans toi. Durant ces longues années de thèse, tu as été mon échappatoire, la plupart de mes souvenirs vont vers toi. Tu m'as insufflé ton courage pour traverser les épreuves, et tu m'as toujours tiré vers le haut. Ce manuscrit même porte ton empreinte, car il n'y a pas une figure, ou une mise en page qui ne viennent un peu de toi. Nous avons pleins de projets ensemble ; à tes côtés, je n'aurai jamais peur d'avancer.



Par Manuèla, octobre 2017

Résumé

(en français)

Variation et sélection sont au coeur de l'évolution Darwinienne. Cependant, ces deux mécanismes dépendent de processus eux-même façonnés par l'évolution. Chez les micro-organismes, qui font face à des environnements souvent variables, ces propriétés adaptatives sont particulièrement bien exploitées, comme le démontrent de nombreuses expériences en laboratoire. Chez ses organismes, l'évolution semble donc avoir optimisé sa propre capacité à évoluer, un processus que nous nommons évolution de l'évolution (Evo-Evo). La notion d'évolution de l'évolution englobe de nombreux concepts théoriques, tels que la variabilité, l'évolvabilité, la robustesse ou encore la capacité de l'évolution à innover (open-endedness). Ces propriétés évolutives des micro-organismes, et plus généralement de tous les organismes vivants, sont soupçonnées d'agir à tous les niveaux d'organisation biologique, en interaction ou en conflit, avec des conséquences souvent complexes et contre-intuitives. Ainsi, comprendre l'évolution de l'évolution implique l'étude de la trajectoire évolutive de micro-organismes – réels ou virtuels – et ce à différents niveaux d'organisation (génome, interactome, population, ...).

L'objectif de ce travail de thèse a été de développer et d'étudier des modèles mathématiques et numériques afin de lever le voile sur certains aspects de l'évolution de l'évolution. Ce travail multidisciplinaire, car impliquant des collaborations avec des biologistes expérimentateur·rice·s, des bio-informaticien·ne·s et des mathématicien·ne·s, s'est divisé en deux parties distinctes, mais complémentaires par leurs approches : **(i)** l'extension d'un modèle historique en génétique des populations – le modèle géométrique de Fisher – afin d'étudier l'évolution du bruit phénotypique en sélection directionnelle, et **(ii)** le développement d'un modèle d'évolution *in silico* multi-échelles permettant une étude plus approfondie de l'évolution de l'évolution. Dans un premier temps, grâce à une version étendue du modèle de Fisher, nous avons montré qu'un bruit corrélé sur différents caractères phénotypiques évolue sous sélection directionnelle vers une forme bien particulière permettant de compenser en grande partie le coût de la "complexité phénotypique", qui limite habituellement et fortement les chances de fixer des mutations favorables lorsque le nombre de caractères sous sélection est grand. Ces résultats prometteurs démontrent l'importance et l'avantage sélectif du bruit phénotypique en sélection directionnelle, et devrait susciter de nouveaux travaux de recherche, en collaboration avec des biologistes

expérimentateur·rice·s. Dans un deuxième temps, grâce au développement d'un modèle d'évolution expérimentale *in silico* multi-échelles, nous avons pu reproduire virtuellement des approches expérimentales *in vivo* – notamment l'expérience d'évolution à long terme (LTEE) – et ainsi contribué à comprendre les phénomènes de construction de niche et d'émergence d'un cross-feeding stable, prémisses à la diversification et à la spéciation bactérienne. Ce modèle d'évolution *in silico* nous a également permis de nous interroger sur l'émergence et l'évolution de la régulation génétique, comme solution au maintien de l'économie énergétique de la cellule. Nos résultats, encore préliminaires, confirment l'idée que le rôle de la régulation génétique n'est pas d'ajuster le métabolisme aux conditions environnementales, mais plutôt d'assurer l'économie énergétique interne et la survie de la cellule, indépendamment des variations environnementales. Ces premiers résultats suggèrent également que la structuration des génomes bactériens est fortement influencée par les contraintes énergétiques internes.

Cette thèse a été financée par le projet européen EvoEvo (FP7-ICT-610427), grâce à la commission européenne.

(in english)

Variation and selection are the two core processes of Darwinian Evolution. Yet, both are directly regulated by many processes that are themselves products of evolution. Microorganisms efficiently exploit this ability to dynamically adapt to new conditions. Thus, evolution seems to have optimized its own ability to evolve, as a primary means to react to environmental changes. We call this process evolution of evolution (EvoEvo). EvoEvo covers several aspects of evolution, encompassing major concepts such variability, evolvability, robustness, and open-endedness. Those phenomena are known to affect all levels of organization in bacterial populations. Indeed, understanding EvoEvo requires to study organisms experiencing evolution, and to decipher the evolutive interactions between all the components of the biological system of interest (genomes, biochemical networks, populations, ...). The objective of this thesis was to develop and exploit mathematical and numerical models to tackle different aspects of EvoEvo, in order to produce new knowledge on this topic, in collaboration with partners from diverse fields, including experimental biology, bioinformatics, mathematics and also theoretical and applied informatics. To this aim, we followed two complementary approaches: **(i)** a population genetics approach to study the evolution of phenotypic noise in directional selection, by extending Fisher's geometric model of adaptation, and **(ii)** a digital genetics approach to study multi-level evolution. This work was funded by the EvoEvo project, under the European Commission (FP7-ICT-610427).

Liste des publications personnelles

Articles parus

- **Rocabert, C.**, Knibbe, C., Consuegra, J., Schneider, D., & Beslon, G. (2017). Beware batch culture: Seasonality and niche construction predicted to favor bacterial adaptive diversification. *PLoS Computational Biology*, 13(3), e1005459.

- **Rocabert, C.**, Knibbe, C., & Beslon, G. (2015, July). Towards a Integrated Evolutionary Model to Study Evolution of Evolution. In Proceedings of *the First EvoEvo Workshop (Satellite workshop of ECAL 2015)* (York, UK).

Articles soumis ou en préparation

- **Rocabert, C.**, Bernard, S., Beslon, G., & Knibbe, C. (2017). Phenotypic Noise and the Cost of Complexity. *Evolution*, under review.

- Gippet M.W., J., **Rocabert, C.**, Fenet, S., & Kaufmann, B. MoRIS: Model of Routes of Invasive Spread. Human-mediated dispersal, road network and invasion parameters. In preparation.

Résumés des conférences

- **Rocabert, C.**, Knibbe C., Consuegra J., Schneider D., & Beslon G. (2017, September) Environmental seasonality drives digital populations towards stable cross-feeding. In Proceedings of *ECAL 2017 conference. 14^{th} European Conference on Artificial Life* (Lyon, France). Communication orale

- **Rocabert, C.**, Knibbe, C., Consuegra, J., Schneider, D., & Beslon, G. (2016, September). In Silico Experimental Evolution Highlights the Influence of Environmental Seasonality on Bacterial Diversification. In Proceedings of *the Second Evo-Evo Workshop. Satellite workshop of CCS2016* (Amsterdam, Netherlands). Communication orale

- Gippet, J., Fenet, S., Dumet, A., Kaufmann, B. & **Rocabert, C.** (2016, August) MoRIS: Model of Routes of Invasive Spread. Human-mediated dispersal, road network and invasion parameters. In Proceedings of *IENE 2016 conference. 5th International Conference on Ecology and Transportation: Integrating Transport Infrastructures with Living Landscapes* (Lyon, France). Communication orale par J. Gippet.

- **Rocabert, C.**, Knibbe, C., Consuegra, J., Schneider, D. & Beslon, G. (2016, April). In Silico Experimental Evolution Highlights the Influence of Environmental Seasonality on Bacterial Diversification. In *EvoAct: Evolution in action with living and artificial organisms conference* (Autrans, France). Communication orale

- **Rocabert, C.**, Knibbe C., Consuegra J., Schneider D., & Beslon G. (2016, March) Environmental Driving of Bacterial Diversification in In Silico Experimental Evolution. In *Evolutionary systems biology: from model organisms to human disease workshop*, (Cambridge, UK). Poster

- Gippet, J., **Rocabert, C.**, Fenet, S., Dumet, A., & Kaufmann, B. (2015, July). Modeling and evaluating human-mediated dispersal mechanisms at landscape scale: a study of road network and invasion parameters for Lasius neglectus ants invasive species. In Proceedings of *World Conference on Natural Resource Modeling* (Bordeaux, France). Communication orale par J. Gippet.

Contents

List of Figures

List of Tables

Foreword

I precisely remember the first book of theoretical biology I read. At this time, I was an aircraft mechanic, apparently far from the academic world. With *Chance and Necessity*, Jacques Monod made a remarkable demonstration of the central dogma of molecular biology, introduced 12 years ago by Francis Crick (Crick, 1958). As a novice, I have been impressed by the clarity and the rigor of this philosophical essay, stamped with a certain arrogance to have explained everything about life on Earth.

> L'ultima ratio de toutes les structures téléonomiques des êtres vivants est donc enfermée dans les séquences de radicaux des fibres polypeptidiques, "embryons" de ces démons de Maxwell biologiques que sont les protéines globulaires. En un sens, très réel, c'est à ce niveau d'organisation chimique que gît, s'il y en a un, le secret de la vie. Et saurait-on non seulement décrire ces séquences, mais énoncer la loi d'assemblage à laquelle elles obéissent, on pourrait dire que le secret de la vie est percé, l'ultima ratio découverte (Monod, 1970).

This book sowed the seeds of my fascination for evolutionary biology. In the following year, I went back to school with this kind of convictions: living organisms own a "genetic program", encoding for their phenotype. This program is regularly altered by purely random mutations, thereby providing the fuel for evolution. I found comfortable consistency with this dogma in the first lessons I followed at university. The apotheosis has probably been reached on reading *The Selfish Gene* by Richard Dawkins (Dawkins, 1976).

During my first year of college, I met astonishing people[1] that pushed me to take a more measured look at theoretical biology. In particular, I had the opportunity to read the work of Jean-Jacques Kupiec (Kupiec and Sonigo, 2000; Kupiec, 2008), a fervent partisan of nominalism in science, that definitively influenced my scientific itinerary. While J-J. Kupiec's narrative is tinged with a touch of bitterness, his epistemology has the merit to promote thought and criticism. I will always remember the little joke I heard at the time: "*People believing that stochastic gene expression is important are often to the left, people believing that a genetic program exists are often to the right*"[2].

[1] See the acknowledgments.

[2] A classical debate between essentialist and nominalist views in sum.

Armed with this very small, but necessary epistemological knowledge, I finally read *On the Origin of Species* by Charles Darwin. I was not surprised to discover that his reasoning was almost at the opposite from Jacques Monod's ones. While the latter stated that the three properties distinguishing living organisms from the rest of the universe were teleonomy, autonomous morphogenesis and reproductive invariance, Charles Darwin original theory defined individual differences and natural selection as the main properties of life.

> No one supposes that all the individuals of the same species are cast in the very same mould. These individual differences are highly important for us, as they afford materials for natural selection to accumulate, in the same manner as man can accumulate in any given direction individual differences in his domesticated productions (Darwin, 1859).

Indeed, a form of essentialism flowed back in biology in the 1960's, mainly following Erwin Schrödinger's book *What is Life?* (Schrödinger, 1944), and the discover of DNA structure (Avery et al., 1944; Watson and Crick, 1953). According to E. Schrödinger, most physical properties on a large scale (*e.g.* diffusion process or kinetic theory of gases) emerge from chaos at a low scale: this property is known as the principle of order-from-disorder. On the contrary, living matter "*is likely to involve 'other laws of physics' hitherto unknown, which however, once they have been revealed, will form just as integral a part of science as the former*" (Schrödinger, 1944). E. Schrödinger was believing in a principle of order-from-order, the living matter escaping from thermodynamics laws. This point of view, while probably never accepted as is by convinced darwinists[1], still influences nowadays scientific works.

> Pour la biologie moléculaire, l'organisme est donc toujours une machine déterministe, mais la vieille horloge de Descartes a été remplacée par un ordinateur (Kupiec, 2008).

As a consequence, a paradoxical representation of life pervaded theoretical biology for decades: while a small "Copernican revolution" was accomplished in evolutionary biology with C. Darwin, a strong essentialism was hidden in the wood, particularly in molecular biology.

One example is the strong belief that genes are the fundamental units of natural selection. Initially, genes have been defined as inheritable units predetermining[2] phenotypic traits (Johannsen, 1911) (W. Johannsen also introduced the notions of genotype and phenotype). As discussed right above, the discover of DNA and the mechanisms of gene expression reinforced this view and led to a "genocentric" view of evolution, with a clear

[1] In *Chance and Necessity*, J. Monod largely argued against the principle of order-from-order in life. However, all the paradox of his interpretation of life resurged when he used the term "Maxwell's demons" to design enzymes.

[2] One could ask why a causation was introduced, while just a correlation was observed.

separation between the genotype, undergoing mutations and inherited, and the phenotype (Rivoire and Leibler, 2014). This net separation is often criticized as a resurgence of an Aristotelian interpretation of life[1], against the evident nominalism of C. Darwin. *The selfish gene* by Dawkins (1976) is often cited as a culmination of genetic reductionism in biology.

Another consequence of essentialism in biology is the belief that mutations purely occur at random. As stated by Monod (1970), the purpose—or teleonomy—of any living organism is to transmit its intact genotype—by reproductive invariance—to its offspring. Of course, as explained by J. Monod himself, teleonomy is not teleology: the apparent finalism of life is a consequence of natural selection. But the damage is done: in the face of teleonomy and reproductive invariance, the way mutations occur is necessarily out of the scope of selection. For decades, any statement that organisms could partly control the way mutations occur (*e.g.*, the evolution of mutation rates, globally or locally on the genome), was accused of finalism—which is indeed quite paradoxical. In the last decades, this dogmatic view has been undermined by experimental an theoretical highlights, leading to the idea that evolution could shape its own fate. This phenomenon is usually known as the evolution of evolvability.

A last example I would give is the classical interpretation of organismal development as a genetic program, deterministically expressing the phenotype from the genotype: "*chaque œuf contient donc, dans les chromosomes reçus de ses parents, tout son propre avenir, les étapes de son développement, la forme et les propriétés de l'être qui en émergera. L'organisme devient ainsi la réalisation d'un programme prescrit par l'hérédité*" (Jacob, 1970)[2]. An essential role is attributed to the concept of protein stereospecificity, conferring to the cell the ability to run deterministic and logical tasks based on signaling and regulation pathways. As stated by Kupiec (2008): "*bien que cela soulève de nombreux problèmes, ce programme génétique a été conçu par analogie avec un programme informatique*". However, biochemical reactions do not escape thermodynamics laws, and the low number of molecules involved in many biological processes implies that "*chance is at the heart of the cell*" (Gandrillon et al., 2012). Nowadays, the stochastic nature of cellular functioning is widely documented and accepted, but its consequences on the evolution of biological organisms are still largely unknown.

Of course, there is no sense to criticize *a posteriori* the work of an entire scientific community, without considering the extraordinary progresses of biology during the XX$^{\text{th}}$ century. This is why my thesis work will be mostly based on the rationale that the tools and models of theoretical evolutionary biology proven to be efficient can be revisited, extended or modified to ask new scientific questions.

The starting point of my thesis work is the process of evolution of evolution, or EvoEvo, as coined by Hindré et al. (2012), that encompasses the evolution of variability, evolvability,

[1]The model (the essence) of an organism is encoded in its genotype, out of the scope of natural selection (in the world of ideas), its phenotype being an imperfect instantiation of the model (in the sensible world).

[2]These words could remind the preformationist views developed during the XVII$^{\text{th}}$ century.

robustness and open-endeness.

There is no scientific theory without modeling (Servedio et al., 2014). I explored two different—but not opposite—modeling approaches to study EvoEvo: mathematical modeling and multi-scale individual-based simulations. On this particular point, I have been strongly influenced all along my stay in the INRIA-Beagle team by the complementary approach of Carole Knibbe and Guillaume Beslon, well-resumed by the quadruplet "*Create, Play, Experiment, Discover*", used as a slogan for the 14th European Conference on Artificial Life.

As explained in chapter I, there is a deep sense to tackle EvoEvo by these two modeling sides. The first, and most obvious reason is that analytical demonstrations are needed to sit a theory and convince a scientific community. The second reason is more specific to EvoEvo: the evolution of an organism implies the interaction of many biological organization levels, with different temporalities and scales. To properly understand how evolution shapes such a complex system, we need to simulate it with a multi-scale model catching, at least partially, this complexity (Lavelle et al., 2008). As stated by S. L. Peck: "*The world is complex and we need all the tools that we can muster to understand it*" (Peck, 2004).

This manuscript is structured as following: in a first chapter, I will introduce the concept of evolution of evolution, and the modeling approaches used to decipher some of its aspects. Then, in part A, I will present a mathematical modeling study on the evolution of phenotypic noise. This model extends Fisher's geometric model (Fisher, 1930). In part B, I will introduce EVO^{2}SIM, a multi-scale model of *in silico* experimental evolution (Hindré et al., 2012) dedicated to the study of evolution of evolution (chapter III). Two results obtained with this model will be presented, one on niche construction and the evolution of stable cross-feeding (chapter IV), another on the evolution of regulation when organisms undergo strong energy trade-offs (chapter V).

Some of these results have been published, or submitted to scientific journals. Others come from documents produced in the course of the FP7 EvoEvo project I were involved in, or are preliminary. I will indicate it when it will be the case. This manuscript also contains appendices; I will refer to them in the main text when necessary.

Some rain forests in the Amazon region occur on white-sand soils. In these locations, the physical environment consists of clean white sand, air, falling water, and sunlight. Embedded within this relatively simple physical context, we find one of the most complex ecosystems on earth, containing hundreds of thousands of species. These species do not represent hundreds of thousands of adaptations to the physical environment. Most of adaptations of these species are to the other living organisms. The forest creates its own environment. (Ray, 1993)

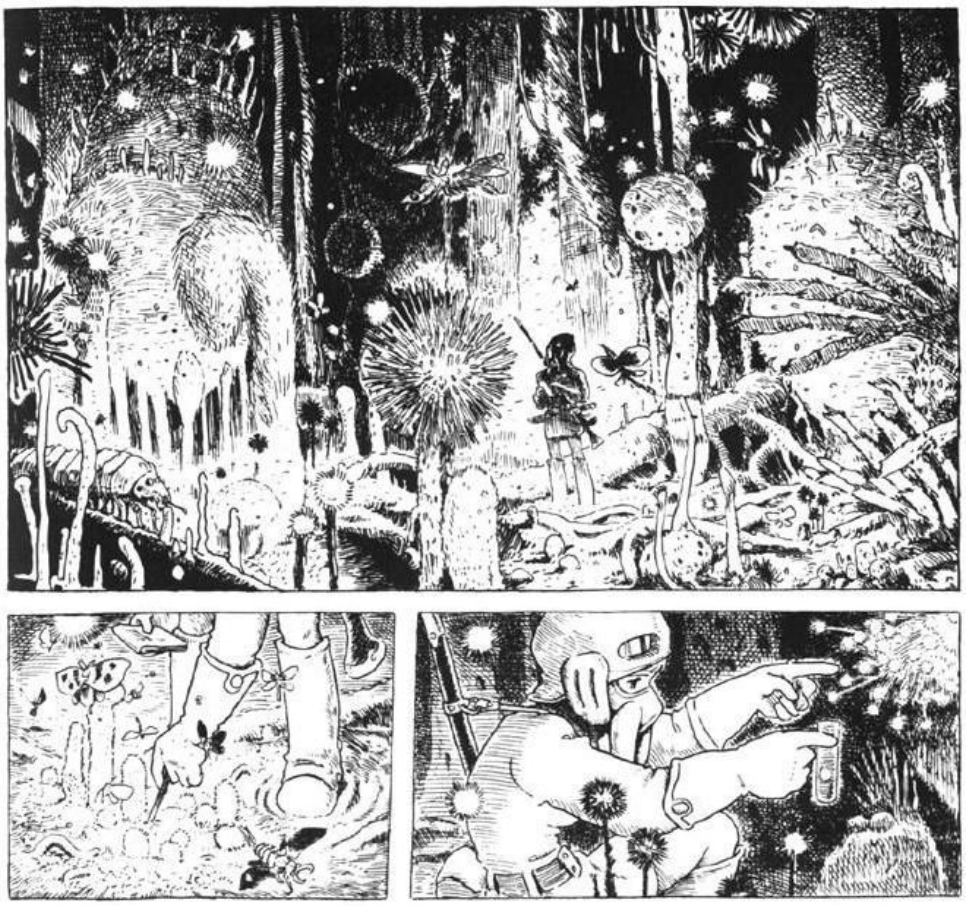


Figure 1 – An example of complex ecosystem. From *Nausicaä*, by Hayao Miyazaki.

Chapter I

Introduction

I.1 The limits of the modern synthesis

Variation and selection are at the core of evolution (Darwin, 1859). In theory, these two mechanisms are sufficient to engage a process of Darwinian evolution, where differences in reproduction and survival rates—summarized by the concept of **fitness**—lead to the "survival of the fittest" (Spencer, 1864). During the XX$^{\text{th}}$ century, the modern synthesis has been developed to rise this mechanism as the central paradigm of biology, merging C. Darwin's and G. Mendel's theories (Huxley, 1942) (Fig. I.1). Variation and selection are also exploited in other fields such as evolutionary optimization algorithms. However, while the modern synthesis mostly focused on molecular evolution, at the level of the **genotype** (by attributing a fitness to an allele segregating in the population for example), selection actually plays on the **phenotype** of an organism (Lande, 1976). Despite the attempt of quantitative genetics to link phenotypic variability with genetic mutations, the relationship between the genotype and the phenotype, known as the **genotype-to-phenotype map** (Alberch, 1991), is far from being understood, and classical models of evolution are unable to explain the most integrated properties of living organisms, *e.g.*, phenotypic innovations or major transitions (Smith and Szathmary, 1997). Three main reasons could be invoked to explain the apparent failure of modern synthesis to model the most complex evolutionary outcomes:

• **Biotic systems process information.** In the early 1970's, Paulien Hogeweg and Ben Hesper coined the term "bioinformatics" to design the study of "informatic processes in biotic systems" (Hogeweg and Hesper, 1978; Hogeweg, 1978). Even if the term has later been distorted to refer to biological data analysis, an important idea was released: according to P. Hogeweg, "*it seemed to us that one of the defining properties of life was information processing in its various forms, e.g., information accumulation during evolution, information transmission from DNA to intra and intercellular processes, and the interpretation of such information at multiple levels*" (Hogeweg, 2011). Indeed, an essen-

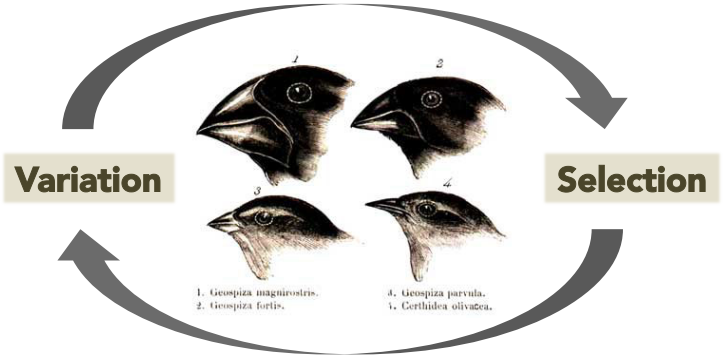


Figure I.1 – Variation and selection at the heart of Darwinian evolution. As symbolically represented by the Darwin finches, whose beaks are adapted to various sizes and shapes of seeds, variation and selection are at the heart of Darwinian evolution, a theoretical basis to the modern synthesis.

tial property of living species seems to be their ability to accumulate information from past environments, on the long-term. By "learning" about past environmental features, species can "react" to new environments, by enhancing their **evolvability** (*e.g.*, by increasing mutation rates, or favoring mutations in a specific region of their genome, ...), or their **robustness** (*e.g.*, by evolving DNA repair mechanisms, or by buffering genetic variations through the regulation network, ...). During the last decades, this property of evolution received a lot of interest, and is often referred as **evolution of evolvability**[1].

• **Evolution acts at multiple levels.** In 2003, Paulien Hogeweg and Nabuto Takeuchi noticed that: "*although there has been much discussion on what is the appropriate level on which Darwinian selection operates, we now know that in many cases the interesting features arise through the occurrence of multiple levels of selection which act in concordance and/or in conflict*" (Hogeweg and Takeuchi, 2003). While the definition of "level" is the source of a classical debate in biology[2], one would hardly disagree that life takes place on multiple physical and time scales. A living organism is composed of one or more cells, each containing a cytoplasm with numerous and complex structures, DNA, RNA, proteins and so on. Each organism interacts with its environment and with other organisms. Populations of organisms modify their environment, creating new selective pressures, and various species interact together, directly or indirectly. Life on earth thus scales from molecules to entire ecosystems, all of these structures interacting and evolving in concert. Here, a question rises about how to model such a complex system. Nobody pretends to be able to forecast the weather by only simulating a set of gas molecules under brownian motion. With such a low-scale modeling, important properties of weather, such as gravity and Coriolis forces, temperature gradients or day/night cycles will never emerge from the model. This is quite similar in evolutionary biology: a gene-centered model will not be able to produce the most integrated properties of evolution, simply because there is no

[1]The concept of information accumulation and "evolution learning" is also a concern of the **extended evolutionary synthesis**, an attempt to extend the modern synthesis (Laland et al., 2015; Watson and Szathmáry, 2016).

[2]See Banzhaf et al. (2016) for a discussion on the notion of "biological level".

support in the model for it (Banzhaf et al., 2016). To do so, **multi-level models** are needed.

• **Interesting properties of evolution emerge with second-order selection.** According to Tenaillon et al. (2001), the Darwinian view of evolution needs a refinement to explain its "complex dynamic aspects". More than just a selection for better adaptation to a specific environment, **second-order selection**, or **indirect selection** (Kirschner and Gerhart, 1998; Reisinger and Miikkulainen, 2006), acts on the regulation of the processes of adaptation to any new environment (Pennisi, 1998). Some survival strategies could not evolve without second-order selection, such as evolution of mutation rates and mutators (Denamur and Matic, 2006), or evolution of bet-hedging (Beaumont et al., 2009) for example. Second-order selection is also responsible for the emergence of important processes discussed just above, such as information accumulation in biotic systems.

Finally, as shown in Figure I.2, long-term evolution and second-order selection led to the emergence of many mechanisms observed in living systems, at all the biological organization levels. These mechanisms directly control the variability of organisms, and are themselves under selection. Hence, we can expect that living organisms, more than being adapted to their environment, **are adapted to evolve**. Hindré et al. (2012) coined the term **evolution of evolution** (EvoEvo) to refer to this evolutionary process.

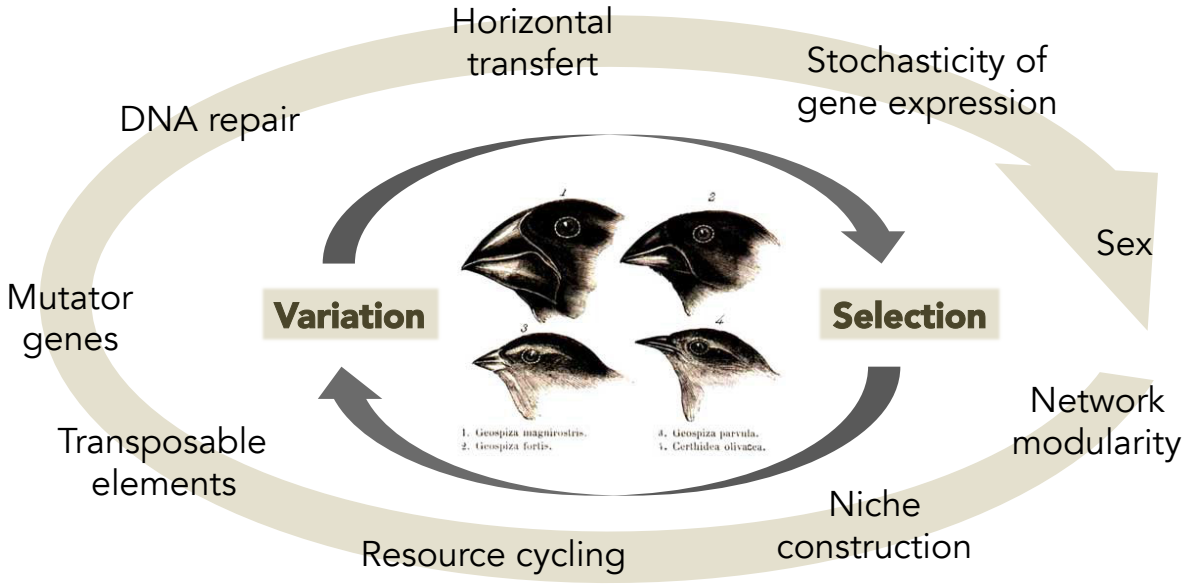


Figure I.2 – Long-term evolution leads to evolution of evolution. On the long-term, living organisms have evolved different mechanisms (DNA repair, horizontal transfer, sex, and so on) that control their own variability. However, these mechanisms are themselves under selection, implying that the basic mechanisms of evolution are therefore themselves evolving, a process called evolution of evolution, or EvoEvo.

I.2 What is evolution of evolution?

By evolving, living organisms permanently adapt to rarely stable and sometimes unpredictable environments. Moreover, organisms constantly modify their environment, by interacting with it and by evolving, thus generating complex and challenging conditions. While higher eukaryotes have evolved complex sensori-motor systems to plastically adapt to environmental variations, micro-organisms (that represent more than an half of the biomass on earth) do not have such complex sensoring abilities. However, they are surprisingly efficient to adapt to their environment by simply ... evolving. Many experimental studies demonstrated that bacteria and viruses are able to adapt to new environments in only a few tens of generations (Rainey and Travisano, 1998; Zhang, Q.,Lambert, G., Liao, D., Kim, H., Robin, K., Tung, C.-k., Pourmand, N., 2011). Hence, micro-organisms are excellent candidates to study evolution of evolution (Hindré et al., 2012).

EvoEvo encompasses the evolution of four essential evolutionary properties: **variability**, **evolvability**, **robustness**, and **open-endedness**. While the notions of evolvability and robustness pervaded theoretical evolutionary biology during the last decades, the concept of open-endedness is more familiar to computational scientists. However, it is strongly related to **phenotypic innovations** and **major transitions** (Smith and Szathmary, 1997), two important concepts in evolutionary biology.

I.2.1 Variability

Variability is the ability to generate new phenotypes. It is a necessary condition for any evolutionary process to take place. A lot of biological mechanisms have been identified that produce and/or control variability:

(1) Genetic variability. For historical reasons, genetic variability has been widely studied during the XX$^{\text{th}}$ century, and a variety of mutational events altering genomes have been identified (point mutations, small insertions and deletions, large rearrangements, horizontal transfers, gene amplifications, ...). Many mechanisms are known to modify the rate at which these mutation events occur in the genome, locally or globally, as reviewed in Ryall et al. (2012). For example, when **contingency loci**—localized on small portions of the genome—are mutated, mutation rates are locally increased. Another example is **mutations in DNA repair or maintenance genes**, that can lead to hyper-mutator strains, which have constitutively elevated mutation rates. In some conditions, these strains can be positively selected and favor adaptation (Tenaillon et al., 1999; Denamur and Matic, 2006). As a last example, **transient changes in the expression level of DNA repair and maintenance genes** allow for rapid mutation rates increase in case of environmental stress (Foster, 2007);

(2) Phenotypic plasticity. According to Stearns (1989), phenotypic plasticity refers

to phenotypic variability in response to the environment. Micro-organisms own genetic regulation networks, able to sense their environment. Evolution can shape these regulation networks such that one genotype can produce many phenotypes as a function of an environmental signal (the reaction norm). When one genotype produces several discrete phenotypes depending on the environmental signal, we speak about **polyphenism**. When one genotype produces a single phenotype, whatever the environmental variations, we speak about **environmental canalization**, one source of evolutionary robustness (Waddington, 1942);

(3) Transgenerational epigenetic inheritance. According to Veening et al. (2008), epigenetic inheritance refers to any transmission of a cellular state from one generation to another without genome modification. A classical example of this mechanism is DNA methylation or acetylation (Avery, 2006). For example, the agouti yellow mouse phenotype is due to the unmethylation of the retrotransposon gene *Avy*, inserted upstream of *agouti* gene. Agouti yellow mices have yellow coat and suffer obesity. It appeared that unmethylated sequences are transmissible from one generation to the other via the gametes, without modification of the genotype.

(4) Phenotypic stochasticity. Finally, the stochastic fluctuations of the phenotype (or phenotypic noise) are an important source of variability (Symmons and Raj, 2016). Phenotypic noise is mainly due to the inherent stochastic nature of biochemical reactions inner the cell, because of the low number of implicated molecules and thermodynamic fluctuations. An example is **stochastic gene expression** (SGE). SGE has an important role in genetic regulation and the emergence of interesting phenotypic properties such as stochastic switching (Acar et al., 2008; Tsuru et al., 2011). Stochastic fluctuations are of primary importance in some survival strategies, such as bet-hedging (Veening et al., 2008), as reviewed in details by De Jong et al. (2011). The evolution of phenotypic noise will be studied in part A of this manuscript.

All these mechanisms being themselves under selection, we can expect that variability—and thus evolution—can evolve.

I.2.2 Evolvability and robustness

The question of the evolution of evolvability and its relationship with the evolution of robustness has received important contributions in the last years. However, the question is still open. While the term evolvability has been used in different ways (Wagner, 2013), it is usually defined as **the ability to increase the proportion of beneficial mutations**, while robustness is defined as **the ability to withstand mutations without losing fitness**. Both mechanisms has been shown to evolve, mainly in numerical simulations (see *e.g.* Bedau and Packard 2003; Elena and Sanjuán 2008; Crombach and Hogeweg 2008; Beslon et al. 2010b). Demonstrating evolution of evolvability or robustness experimentally is much more difficult since it necessitates to perform experimental evolution experiments, which are long and costly (see *e.g.* Elena and Lenski 2003).

At first sight, evolvability and robustness seem to be antagonistic. An evolvable organism should not be robust, and a robust organism should not be evolvable. However, the remarkable ability of Darwinian evolution to generate sophisticated emergent properties is demonstrated here. Indeed, evolvability has an important role in **innovation**: a biological system is evolvable if it can acquire novel functions through genetic change that increase fitness (Wagner, 2005). However, and counter-intuitively, robustness and **neutral mutations** also play a key role in the innovation process, because they allow to explore the phenotypic space while the fitness of the organism remains constant. By exploring the neutral landscape of an organism, neutral mutations promote future innovation.

This mechanism has nicely been represented by Wagner (2008) (Fig. I.3). Let's consider a network of all possible genotypes of an organism, each node being a genotype, linked to accessible other genotypes by single mutations. A fitness is attributed to each genotype. Some mutations are neutral, meaning that they link two genotypes with the same fitness, negative (if they decrease the fitness), or positive. A positive mutation can also be an innovation (the acquisition of a novel function with a beneficial fitness value, as discussed later in this introduction). For a robust organism, many mutations are neutral, such that evolution consists to travel in the neutral genotype network. Robust organisms can thus explore vast regions of the genotype network with no consequence on their fitness, and access new genotypes not accessible otherwise. As metaphorically stated by A. Wagner:

> Perhaps the most compact way to express this problem is with an analogy from politics: evolving populations need to be both "conservative" and "progressive" at the same time (Wagner, 2012).

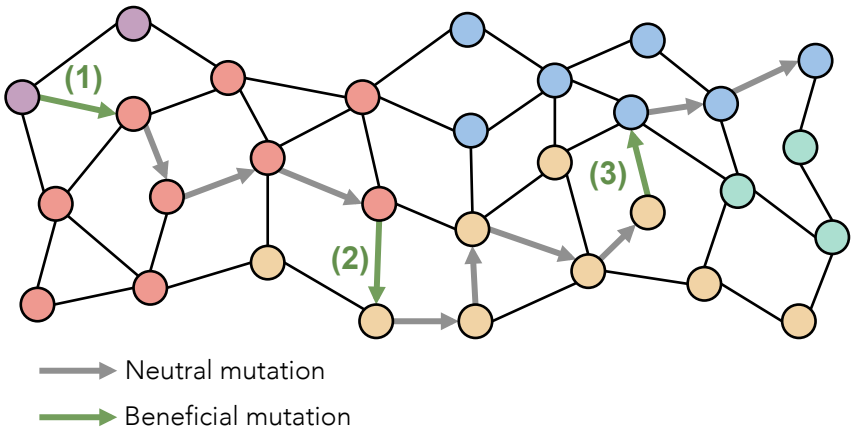


Figure I.3 – Evolution on the genotype network for a robust organism. Each genotype is represented by a node, colored according to its fitness. Single mutations linking genotypes together generate a network, explored by evolving populations of organisms. The successive fixed mutations are represented by a path on this network. Beneficial mutations are represented by green arrows, while neutral ones are represented by grey arrows. Here, the evolutive path is composed by long successive travelling sessions on the neutral network, punctuated with three beneficial mutations (inspired from Wagner 2008)

I.2.3 Open-endedness

The notion of open-endedness has been extensively discussed in Banzhaf et al. (2016). Often defined as the ability to continuously produce novelty and/or complexity, open-endedness is a quite fuzzy notion. Considered as an important modeling challenge in the field of artificial life, the term is almost unknown in theoretical biology. Indeed, this recent concept still needs to be properly defined. According to Banzhaf et al. (2016), open-endedness is essentially a **modeling concept**, and can refer to the capacity of a model to generate "novelty". Banzhaf et al. (2016) identified three types of novelties, depending on model capabilities: **variation**, **innovation** and **emergence**:

(1) Variation. A variation is defined here as a change to the values of a variable, or an instance in the model. This means that variations are simply the exploration of the predefined space of the model ("*novelty in the model*", Banzhaf et al. 2016). This definition could correspond to the notion of variability presented above;

(2) Innovation. An innovation is a change to the model itself. Hence, an innovation modifies the space in which variation can operate ("*novelty that changes the model*", Banzhaf et al. 2016). This definition could correspond to the notion of innovation discussed above;

(3) Emergence. Emergence is a change to the "meta-model". Indeed, a model is the instantiation of a conceptual model, defining types of objects and their relationships ("*novelty that changes the meta-model*", Banzhaf et al. 2016). This idea is exemplified by Andrews et al. (2011): collective behavior (*e.g.* collective bird fly) is often modeled by a class of individual-based models known as flocking or boids models (Reynolds, 1987). It is first needed to define the behavior of the boids (to define the agents), and then to collect individual positions (to collect data) in order to detect flockness (to measure the data). Then, a meta-model of a flocking model is the association of the concepts of agent, data and measure (Andrews et al., 2011). The notion of emergence directly refers to **major transitions**. According to Smith and Szathmary (1997), a major transition occurs when "*entities that were capable of independent replication before the transition can only replicate as parts of a large unit after it*". The comprehension of this property of living organisms is an important challenge for evolutionary biologists.

We see that the concepts of variability, evolvability, robustness and open-endedness are intertwined in a complex way. Their evolution also require interdependencies between many mechanisms and properties, at multiple levels of biological organization.

I.3 Capturing the whole spectrum of EvoEvo, or the necessity to build multi-level models

I.3.1 Modeling choices and the experimental method

Building a model is a tough task, since modeling choices depend on the scientific question, but also on the kind of desired output (consciously or not) and maybe on some intuition. On this point, the modeling work presented in this manuscript has been largely influenced by the approach of the INRIA-Beagle team, in particular the works of Knibbe (2007) and Beslon (2008) on the modeling of complex biological systems. A model is always false, and implies unavoidable assumptions, simplifications and shortcuts (Banzhaf et al., 2016). But more than that, when a model is correctly used to produce a new hypothesis or theory, this hypothesis or theory should acquire its own existence, independently from the model (Grimm, 1999). In this sense, the model is useful to generate new ideas, but should then disappear in their shadow (Beslon, 2008).

According to Servedio et al. (2014), in evolutionary research, as in many other fields, some models are conceived to test the logic of verbal explanations of a theory, in the same way that empirical data is used to test scientific hypotheses. To build such a **proof-of-concept model**, we should follow the four steps of the **experimental method** promoted by Claude Bernard: **(i)** First, observe the nature and build hypotheses. **(ii)** Then, pick assumptions and build a model. **(iii)** Third, analyze the model, and finally **(iv)** evaluate new hypotheses and propose new directions, closing the loop (Fig. I.4). Even if the reality of scientific modeling has been shown to be more complex (the four steps are often interconnected, and even self-connected, such that building a model consist in navigating between them, Chalmers 1990; Beslon 2008), we should stick to this "best practice" guideline as much as possible. The hardiest task (but also the most exciting) probably consists in picking the right assumptions and build the model.

There is no well-defined guideline to pick the modeling assumptions, and to adjust the complexity of the model. However, depending on the scientific question, the model must at least represent the objects of interest, and their interactions. Regarding the study of EvoEvo, two important theoretical objects summarize the relationship between an evolving organism and its environment: **the genotype-to-phenotype map**, and the **fitness landscape**:

(1) The genotype-to-phenotype map. The phenotype of an organism results from a complex and non-linear cascade of developmental, physiological and regulatory processes, summarized by the concept of genotype-to-phenotype map. According to the central dogma of molecular biology (Crick, 1958), the development of an organism reflects the flow of information from the genetic sequence to the phenotype. As such, the genotype-to-phenotype map is an object that represents all the functions of an organism (transcription, translation, regulation, protein folding, metabolism,

environmental sensing, and so on). Hence, the genotype-to-phenotype map is generally a very complex object, an important condition to the evolution of evolution, as discussed above.

(2) The fitness landscape. The fitness landscape is considered as one of the most important concepts in theoretical evolutionary biology. The fitness landscape projects the space of all possible genotypes, or phenotypes of a population of organisms in the space of fitness values, usually through a fitness function. Firstly used by Wright (1932), the fitness landscape is at the heart of historical models of evolution, such as **Fisher's geometric model** (Fisher, 1930) or **NK-fitness landscapes model** (Kauffman and Levin, 1987). The latter has been used to show how the complexity of a landscape influences the course of an evolutionary process (Correia and Fonseca, 2007). The former will be presented in detail in part A of this manuscript. Often represented by a smooth function (*e.g.*, a Gaussian-shaped function in Fisher's geometric model), the fitness landscape of living organisms is probably a much more complex, fluctuating and highly dimensional landscape (as discussed below).

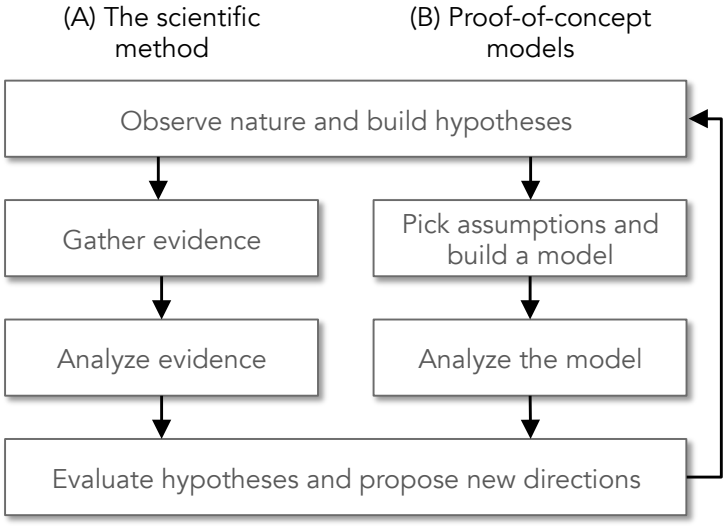


Figure I.4 – Proof-of-concept modeling and the scientific method. This flowchart shows the steps in the scientific process, with a parallel between the scientific method, as defined by C. Bernard, and proof-of-concept modeling methodology. (A) The main steps of the scientific method. (B) The steps of proof-of-concept modeling (inspired from Servedio et al. 2014).

I.3.2 The necessity of multi-level modeling of evolution

Computational models have been used to study evolution since the beginning of the 1990's (Adami, 2006). However, since then, most computational models used a partial representation of the genotype-to-phenotype map, generally in a fixed, predefined, fitness landscape. Yet, fitness is the result of the interaction of all the biological structures of an organism, including its interactions with the environment. Similarly, the variability/robustness/evolvability/open-endedness of the phenotype is the result of the

interaction of variability/robustness/evolvability/open-endedness of all the biological structures of an organism, including its interactions with its environment. Furthermore, these properties are co-dependent and they may interact in a cooperative or competitive way (*e.g.*, evolving chaperone proteins reduce phenotypic variability, thereby increasing robustness). Moreover, both the genotype-to-phenotype map and the fitness landscape are very likely to change during the course of evolution. That is why a computational model of EvoEvo must be multi-level, including the main organization levels of the genotype-to-phenotype map and the fitness landscape (genome, transcription network, metabolic network, phenotype, fitness, population, environment, and so on).

First, the genotype-to-phenotype map has long been considered as a one-directional and deterministic process, the genetic information flowing from the genotype to the phenotype. However, the development of living organisms is a non-deterministic process, depending on many molecular mechanisms that are fundamentally stochastic, as demonstrated by the stochastic nature of gene expression (Elowitz et al., 2002). Thus, one genotype can lead to several random phenotypes, and this stochastic variability can itself evolve through the genotype-to-phenotype map, as discussed in part A of this manuscript. Moreover, the information can flow back from the phenotype to the genotype (*e.g.*, thanks to RNA interference, genetic regulation, or environment influence). Altogether, these mechanisms make the genotype-to-phenotype map of an organism a very complex object to analyze, generating non-intuitive situations through evolution of evolution.

Second, the fitness landscape of living organisms is much more complex than suggested in early models of evolutionary biology. The fitness of an organism depends on its environment. However, organisms constantly interact with it (including other organisms), such that the fitness landscape is constantly fluctuating, triggering complex evolutionary outcomes, such that co-evolution, niche construction, resource cycling, and ultimately major transitions. Some authors use the concept of **fitness seascape** to render the effect of fluctuating fitness landscapes on evolution (Mustonen and Lässig, 2009).

As a whole, we see that the genotype-to-phenotype map and the fitness landscape form a complex system, which cannot be modeled statically as it is the case in classical mathematical representations (Fisher, 1930; Kauffman and Levin, 1987) (even the number of dimensions in the fitness landscape and in the genotype-to-phenotype map are evolvable). Moreover, both the genotype-to-phenotype map and the fitness landscape interact through evolution, a condition for evolution of evolution. For these reasons, a model of evolution of evolution should necessarily include complex, multi-layered and evolvable genotype-to-phenotype map and fitness landscape. Such a model will incorporate a large set of parameters and its study is likely to be very difficult. But as a compensation, it will give rise to new hypotheses and predictions, impossible to obtain with previous models.

I.3.3 But ...

Based on the previous arguments, one could argue that our modeling approach should be exclusively a computational multi-scale approach, in the hope to observe the most complex features of EvoEvo; it would be foolish to do so. **(i)** The first reason is that history of theoretical evolutionary biology demonstrated the importance of mathematical models to understand evolution. From mendelian genetics to population genetics, quantitative genetics, coalescence theory, and so on, mathematical models remain the most powerful—and convincing—scientific tools. **(ii)** The second reason is that when the intuition of an hypothesis or a theory is acquired by the exploitation of a computational model of evolution, the best practice would be to derivate the mathematical equations representing the phenomenon in a more abstract way, and provide a robust mathematical analysis, if possible. This is for example the case for ævol model (Knibbe et al., 2007a) (presented below): in ævol, a strong correlation between the genome size of bacterial-like digital organisms and their mutation rates has been identified. This observation has further been generalized with a more abstract mathematical model (Fischer et al., 2014). **(iii)** The third reason is more practical: if some properties of EvoEvo can be studied with mathematical models, there is no reason not to do it (Peck, 2004). The approach used in this manuscript mostly results from this last reason. We decided to have a complementary approach, anchored in the modeling practice of the INRIA-Beagle team, using both sustainable mathematical and complex multi-scaled and individual-based approaches, as exemplified in the next parts of this manuscript.

I.4 State of the art

We have seen above that the study of EvoEvo requires the use of a variety of models, including mathematical and multi-scaled individual-based approaches. In both cases, many models, with sometimes a long history behind them, already allowed to largely highlight the evolutive interactions between the genotype-to-phenotype map and the fitness landscape. Two modeling approaches will be presented below, and then be used as a basis to decipher some aspects of EvoEvo: **(i) Fisher's geometric model**, an historical mathematical model of the genetic theory of adaptation, and **(ii) digital genetics** formalism, that led to an experimental method in evolutionary modeling: **in silico experimental evolution**.

I.4.1 Fisher's geometric model of adaptation

Fisher's geometric model (FGM, Fisher 1930) has a long and interesting history (reviewed in Orr 2005; Tenaillon 2014), and received renewed interest in the last decades. According to Tenaillon (2014), a reason is that behind its apparent simplicity and limited number of parameters, FGM integrates a full model of mutations and epistatic interactions, with

surprising emergent properties.

In FGM, each phenotypic character of an organism is represented by an axis in a Cartesian coordinate system. R.A. Fisher used the term **phenotypic complexity** to refer to the dimensionality n of this space. Let's define the phenotype of an organism with n characters by a point $\boldsymbol{z} = (z_1, z_2, ..., z_n)^T$ (T being the matrix transposition operator). The fitness $W(\boldsymbol{z})$ of this organism is then determined by its distance to the fitness optimum $\boldsymbol{z_{opt}}$, such that:

$$W(\boldsymbol{z}) = exp\left[-(\boldsymbol{z} - \boldsymbol{z_{opt}})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{z} - \boldsymbol{z_{opt}})\right] \tag{I.1}$$

where $\boldsymbol{\Sigma}$ denotes a $n \times n$ positive-definite and symmetrical matrix defining the shape of the fitness landscape. For the sake of simplicity, an isotropic fitness landscape is usually assumed, meaning that fitness varies independently and in the same proportion for all characters. The origin of the coordinate system is also used as the fitness optimum ($\boldsymbol{z_{opt}} = \boldsymbol{0}$), such that the fitness function is reduced to a simple Gaussian-shaped function:

$$W(d) = \exp\left[-\frac{d^2}{2}\right] \tag{I.2}$$

with $d = \|\boldsymbol{z}\|$ the euclidean distance of the phenotype $\boldsymbol{z}$ from the fitness optimum. Mutations are represented by a random vector $\boldsymbol{r} = (r_1, r_2, ..., r_n)^T$ moving the ancestral phenotype $\boldsymbol{z}$ to its offspring $\boldsymbol{z'}$ such that $\boldsymbol{z'} = \boldsymbol{z} + \boldsymbol{r}$. The probability distribution of the mutants is often characterized by a multivariate normal distribution of the form:

$$p(\boldsymbol{r}) = \frac{1}{\sqrt{(2\pi)^n |\boldsymbol{\Sigma_r}|}} \exp\left[-\frac{1}{2} \boldsymbol{r}^T \boldsymbol{\Sigma_r}^{-1} \boldsymbol{r}\right] \tag{I.3}$$

with $\boldsymbol{\Sigma_r}$ a covariance matrix. Thus, a mutation can potentially modify every characters, a property known as the **universal pleiotropy assumption** (Wagner and Zhang, 2011). Usually, initial conditions are a maladapted clonal population of asexual organisms, sitting at a certain distance from the optimum $\boldsymbol{z_{opt}}$. Then, the work consists in studying the bout of adaptation towards the optimum. FGM implies some well-known assumptions, as described in details in Martin (2014): **(i)** the distribution of all random variables have finite mean and variance and satisfy Lindeberg's conditions (the central limit theorem can be applied), **(ii)** the fitness function is twice differentiable and admits at least one non-degenerate optimum, **(iii)** mutations have mild effects on the phenotype (mutational events remain "local"), **(iv)** each mutation potentially affect every characters (the universal pleiotropy assumption), and **(v)** the variety of mutants is very large, such that quantitative characters vary continuously (the infinite-alleles approximation).

For a given phenotype $\boldsymbol{z}$, Fisher (1930) demonstrated that the probability $P_a(x)$ that a random mutation of a given phenotypic size s is favorable is $1 - \Phi(x)$, where Φ is the cumulative distribution function of a standard normal random variable, and x is a standardized mutational size $x = s\sqrt{n}/(2d)$. n is the number of characters and $d = \|\boldsymbol{z}\|$ is the euclidean distance to the optimum. As shown in Figure I.5A, this probability quickly decreases with the mutational size.

R.A. Fisher also suggested that organisms may pay a cost for the complexity of their phenotype (the complexity being defined here as the number of characters n under selection),

because the probability to fix a beneficial mutation of a certain size literally vanishes when the number of characters increases (Orr, 2000). In consequence, only mutations with a very small size should segregate in a population. R.A. Fisher argued that his result was a demonstration of a **micro-mutationism** view of evolution, populations evolving smoothly by very little steps. However, R.A. Fisher omitted to consider that mutations occur in populations of finite size. As later demonstrated by M. Kimura with the neutral theory of evolution, new mutations appearing in a population have a significative chance to be lost at random, especially when their beneficial value is low. Thus, according to the cost of complexity and the effects of genetic drift, we should expect that only mutations of an intermediate size would segregate in an evolving population. Finally, as discussed by Orr (2005), evolution towards a fitness optimum cannot be reduced to the study of a single mutational event. When the entire boot of adaptation towards the fitness optimum is scrutinized in FGM, it appears that the size of fixed mutations depends on the distance from the optimum: very few large mutations are usually necessary to approach the fitness optimum, the remaining distance being filled with many small mutations, as shown in Figure I.5B.

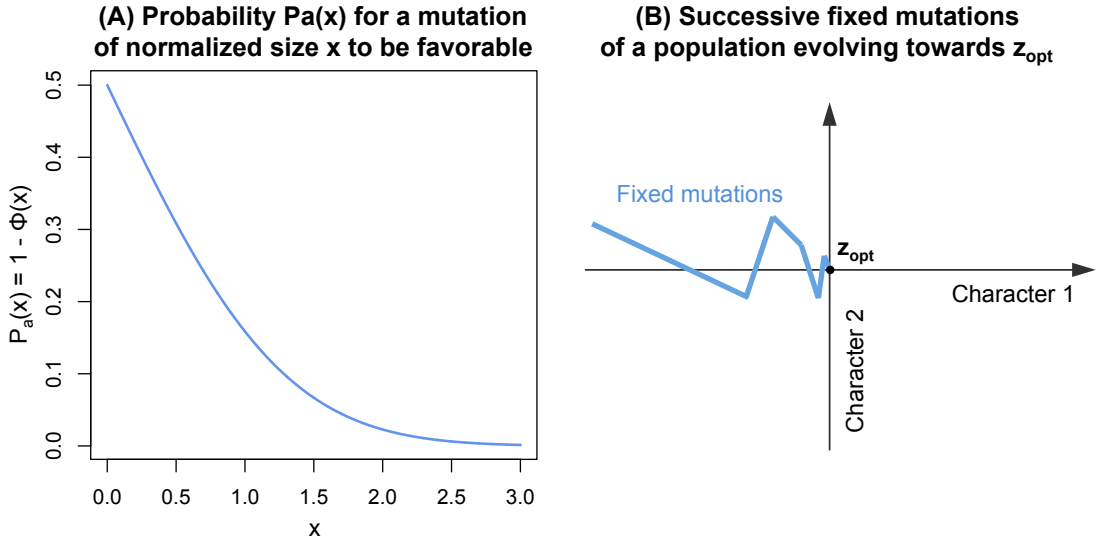


Figure I.5 – The beneficial value of a mutation in FGM depends on its size. (A) The probability $P_a(x)$ for a mutation of normalized size $x = s\sqrt{n}/(2d)$ (with n the number of characters, and $d = \|\boldsymbol{z}\|$ the euclidean distance from the fitness optimum) to be favorable is $P_a(x) = 1 - \phi(x)$ (blue curve). **(B)** When a population evolves towards the fitness optimum $\boldsymbol{z_{opt}}$, a few number of large mutations are usually sufficient to approach $\boldsymbol{z_{opt}}$. Then, a lot of small mutations are necessary to reduce this distance to zero, as shown in blue (inspired from Orr 2005).

I.4.2 In silico experimental evolution: a tool to study evolution

In silico experimental evolution (Hindré et al., 2012; Mozhayskiy and Tagkopoulos, 2013; Batut et al., 2013) is an approach based on the usage of individual-based models to evolve **digital organisms** in a computer, a field known as **digital genetics**. In digital genetics models (Adami, 2006), organisms are modeled by data-structures representing their

genotype. The kind of structure used depends on the studied level(s) of organization (numerical vectors, binary sequences, regulation network, ...) and the formalism used to develop the model (reviewed in Mozhayskiy and Tagkopoulos 2013; Hindré et al. 2012). As discussed in part B of this manuscript, the development of an *in silico* model of evolution needs some "ingredients". The minimum requirement is the **evolutionary engine** enabling the data-structures to reproduce, mutate and be selected depending on a fitness function. Many digital genetics models have been proposed in the literature, Avida being the best known (Wilke et al., 2001; Adami, 2006). However, only a few models are able to efficiently address questions related to evolution of evolution, in particular because most formalisms impose that the structure of organisms and the fitness landscape are fixed over time.

The increasing number of parameters and the computational and time resources needed to run multi-scale individual-based simulations forbid an exhaustive and rigorous exploration of the parameters and state spaces of the system, as it could be the case for Fisher's geometric model for example. Hence, an **experimental approach** is needed to study this kind of models. According to Peck (2004), complex simulation models should be explored with the same experimental and statistics tools used for real systems:

> Simulations are experimental systems. Their complexity can make them closer cousins in complexity to nature itself than to simple analytic models, but with a powerful advantage over the real world: the modeler has complete control of the system (Peck, 2004).

In evolutionary biology, the experimental method that consists in studying evolving organisms is **experimental evolution**. In experimental evolution, fast replicating microorganisms (*e.g.*, bacteria or viruses) are being evolved in controlled environments for thousands of generations (Philippe et al., 2007). It is then possible to recover precisely the evolutionary history of lab strains by reviving frozen samples (Elena and Lenski, 2003). However, despite its explanatory and statistical power, experimental evolution remains a long and costly process. **In silico experimental evolution** (ISEE) consists in mimicking this process with digital organisms (Hindré et al., 2012; Mozhayskiy and Tagkopoulos, 2013; Batut et al., 2013), as shown in Figure I.6: ancestral microbial (or digital) populations are evolved in controlled environments and regularly frozen (or saved in a backup), independent repetitions are made, and frozen populations can be revived (or reloaded in memory) to perform competition experiments and other analyses. ISEE approach while be exemplified in the part B of this manuscript.

Two formalisms have recently been used to develop computational models allowing for *in silico* experimental evolution. Knibbe et al. (2007a,b) used the **sequence-of-nucleotides** formalism to develop ævol software. With this model, the authors showed that indirect selection could select specific genetic and network structures depending on the mutational and selective pressures (Knibbe et al., 2007b; Beslon et al., 2010b,a). In parallel, Crombach and Hogeweg (2008) developed the **pearls-on-a-string** formalism and used it to show that, in time-varying environments, regulation networks, metabolic networks and

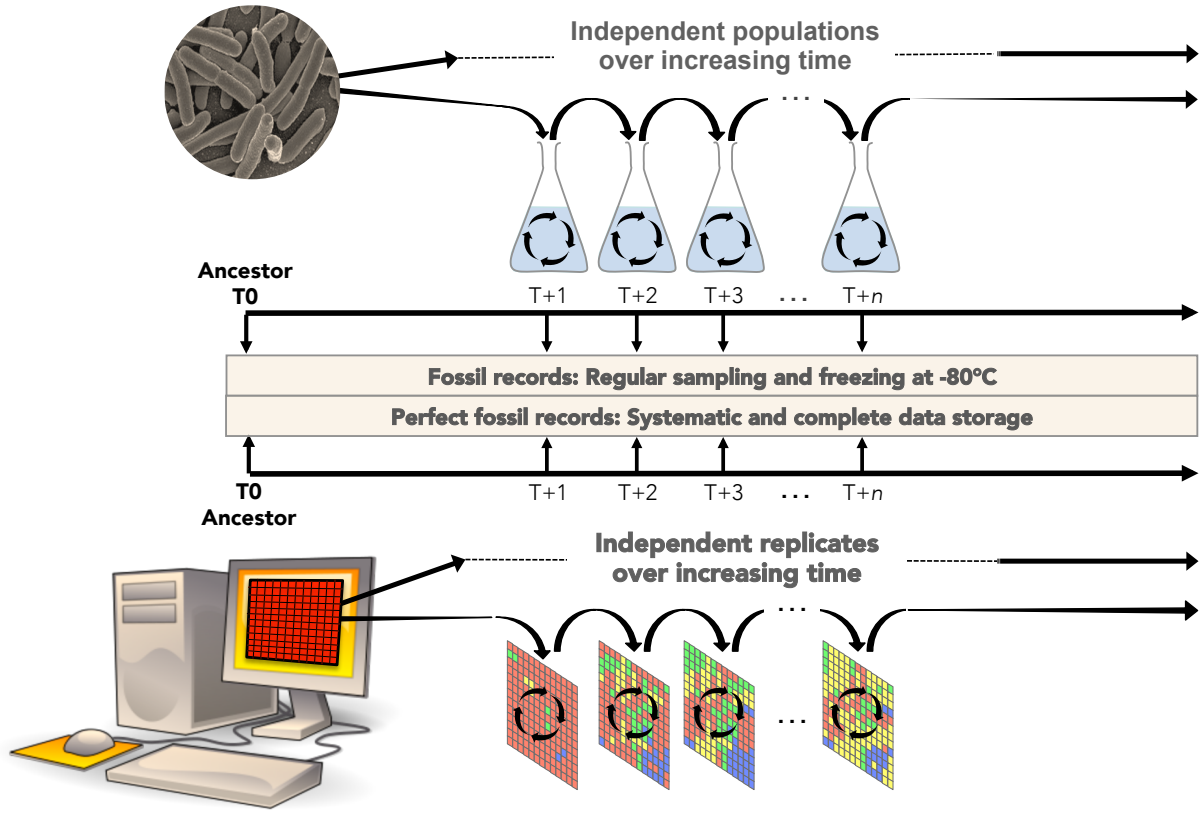


Figure I.6 – In vivo and in silico evolution experiments. Ancetral real micro-organisms (top) and digital organisms (bottom) are propagated in controlled environments, in a laboratory or in a computer respectively. Wet or digital populations are regularly frozen, or saved in backups, respectively and can be revived or reloaded at any time. Many replicate populations can be independently evolved from a common ancestor T0 (inspired from Hindré et al. 2012).

species networks can acquire structures that increase the evolvability of the organisms (Crombach and Hogeweg, 2007, 2008, 2009). These two formalisms are described in the next sections.

I.4.3 The sequence-of-nucleotides formalism

In the sequence-of-nucleotides formalism, the genome is a variable-length string of characters. Predefined signal sequences, analogous to promoters, terminators or start/stop codons, are used to detect genes. Therefore, mutational processes, such as point mutations, small insertions and deletions, or large rearrangements can be simulated in a realistic manner (Hindré et al., 2012). The sequence-of-nucleotides formalism has been successfully used to study *e.g.*, the evolution of non-coding DNA and the genes number (Knibbe et al., 2007b), the evolution of the size and topology of gene networks (Dwight Kuo et al., 2006; Beslon et al., 2010b), gene network interference (Mattiussi and Floreano, 2007; Marbach et al., 2009), the evolution of "public good" production (Frénoy et al.,

2013), and the reduction of genome size in some species (Batut et al., 2013).

Here, we will focus on the ævol software (Knibbe et al., 2007a), because we will get inspired from its genome and genetic regulation representations in the following. In ævol, each digital organism owns a circular, double-stranded chromosome (Fig. I.7a) that is actually a string of binary nucleotides, 0 being complementary of 1 and reciprocally. This chromosome contains coding sequences (genes) separated by non-coding regions. Each coding sequence is detected by a transcription-translation process and decoded into a "protein" able to contribute positively or negatively to a range of abstract quantitative characters (Fig. I.7a). The mechanisms of transcription and translation are modeled in detail (Fig. I.7b,c,e), depending on a genetic code (Fig. I.7d). The combination of all proteins yields the value of each abstract phenotypic character (Fig. I.7g). Adaptation is then measured by comparing the phenotypic values to an arbitrary set of target values. The most adapted individuals have higher chances of reproduction. When a chromosome is replicated, it can undergo point mutations, small insertions and small deletions, but also large chromosomic rearrangements: duplications, large deletions, inversions, and translocations. The various types of mutations can modify existing genes, but also create new genes, delete some existing genes, modify the length of the intergenic regions, modify gene order, and so on.

ævol model has been extended to include regulation of genetic expression, by adding a representation of cellular gene networks (Beslon et al., 2010b). This extended version of ævol, named R-ævol, is a model of prokaryotic regulation. To simulate the interactions between **transcription factors** and **promoters**, two **binding sites** are defined for each promoter. Located immediately before the promoter, the **enhancer site** increases the transcriptional activity when transcription factors bind to it. Directly following the promoter, **the operator site**, down-regulates the promoter's activity when a transcription factor binds to it. Each promoter i owns a basal expression level β_i, which depends on how close its sequence is to a consensus sequence. The transcriptional activity of this promoter depends on the combined activity of the enhancer site activity A_i and the operator site activity O_i, that read:

$$A_i(t) = \sum_j c_j(t) A_{ji} \tag{I.4}$$

and:

$$O_i(t) = \sum_j c_j(t) O_{ji} \tag{I.5}$$

with A_{ji} (resp. O_{ji}) the affinity of protein j for the enhancer site of the promoter i (resp. for the operator site) and $c_j(t)$ the concentration of protein j at time t.

The transcription rate $e_i(t)$ of the RNA sequence associated to the promoter i is then given by the following Hill-like function:

$$e_i(t) = \beta_i \left(\frac{\theta^n}{O_i(t)^n + \theta^n} \right) \left(1 + \left(\frac{1}{\beta_i} - 1 \right) \left(\frac{A_i(t)^n}{A_i(t)^n + \theta^n} \right) \right) \tag{I.6}$$

where n and θ are the two parameters defining the shape of the Hill-function. Finally, given the transcription rate, one can compute the protein concentration (for simplicity, it is

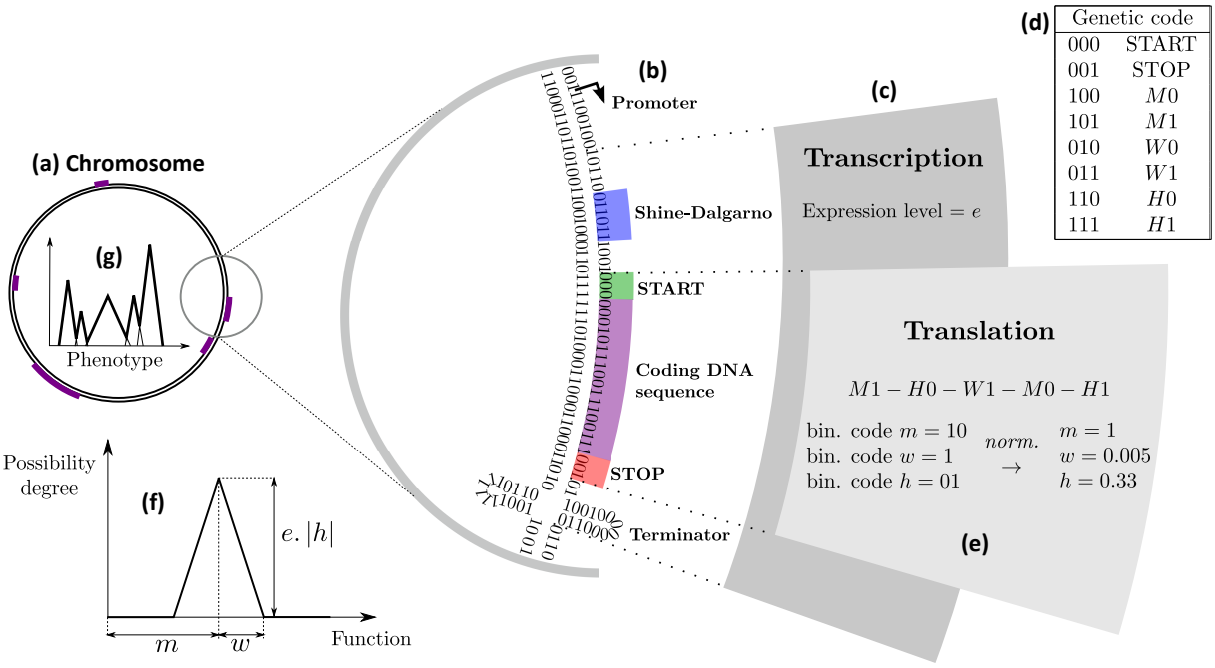


Figure I.7 – A description of ævol model. In the model, each organism owns a circular double-stranded binary chromosome **(a)** along which genes are delimited by predefined signal sequences **(b)**. Promoters and terminators mark the boundaries of RNAs **(c)** within which coding sequences are in turn identified between a Shine-Dalgarno-START signal and an in-frame STOP codon. Each coding sequence is then translated into a protein sequence using a predefined genetic code **(d)**. This protein sequence is decoded as three real parameters called m, w and h **(e)**. Proteins, phenotypes and environments are represented similarly through mathematical functions that associate a level to each abstract phenotypic character in [0, 1]. The contribution of a protein is a piecewise-linear function with a triangular shape, with position m, half-width w and height h **(f)**. All proteins encoded in the chromosome are then combined to compute the phenotype **(g)**, which is compared to the environmental target to compute the fitness of the individual (inspired from Knibbe and Parsons 2014).

assumed that the protein concentration is linearly proportional to the RNA concentration) through the following synthesis-degradation rule:

$$\begin{cases} c_i(0) &= \beta_i/\phi \\ \dfrac{dc_i}{dt} &= e_i(t) - \phi c_i(t) \end{cases} \tag{I.7}$$

where ϕ is a temporal scaling constant representing the protein degradation rate. Thus, when a gene is regulated, the concentration of its product is scaled up or down depending on its transcription rate.

I.4.4 The pearls-on-a-string formalism

In the pearls-on-a-string formalism, the genome is a variable-length string of "pearls" of different types: phenotype genes, transcription factor genes, repeats, retrotransposons, binding sites, and so on. Each pearl type can exist in a predefined number of variants. Mutational operators (point mutations, rearrangements) can modify the genes number, the order of the pearls and the regulation. The pearls-on-a-string formalism has been successfully used for the study of genome and network evolvability (Crombach and Hogeweg, 2007, 2008), resource processing in ecosystems (Crombach and Hogeweg, 2009), and sympatric speciation (ten Tusscher and Hogeweg, 2009).

Recently, Cuypers and Hogeweg (2012) developed a multi-scale model based on the pearls-on-a-string formalism: the Virtual Cell model. As shown on Figure I.8, in Virtual Cell model, digital organisms own circular genomes made of "pearls", encoding for five types of proteins. Organisms grow on an externally provided resource A (Fig. I.8a), by pumping it (Fig. I.8b) or by passive diffusion through the cell's membrane (Fig. I.8c). The pumps require the consumption of an energy carrier molecule X, enzymatically produced from A by a catabolic reaction (Fig. I.8d). Both A and X molecules are required to build end products via another enzymatic reaction (Fig. I.8e). Two other protein types are transcription factors that up-regulate or down-regulate the production of proteins depending on the effect of their ligands, A or X (Fig. I.8f). With their model, Cuypers and Hogeweg (2012) proposed that the complex genotype-to-phenotype map of digital organisms drives genome size dynamics, due to an emerging interplay between adaptation, neutrality, and evolvability, showing that genome expansion and streamlining are generic patterns of evolving systems. More recently, Cuypers et al. (2017) shown with the Virtual Cell model that depending on the frequency of environmental changes, digital organisms evolve different adaptive strategies: when the change frequency is low, evolution leads to phenotypic plasticity, while when the change is high, evolution leads to enhanced evolvability.

I.5 An attempt to merge sequence-of-nucleotides and pearls-on-a-string formalisms

I.5.1 A common formalism: the "bag of tuples"

The **sequence-of-nucleotides** and the **pearls-on-a-string** formalisms have a common property: while their genomic representation (the way information is stored in the genome) differ significantly, in both formalisms, a non-ordered set of tuples is extracted from the genomic data-structure: a **bag of tuples**.

A tuple is an ordered list $(x_1, x_2, ..., x_n) : T_1 \times T_2 \times ... \times T_n$ with T_i the "product type" of x_i (*e.g.*, $\mathbb{R}$, $\mathbb{N}$,...). In both sequence-of-nucleotides and pearls-on-a-string formalisms, the

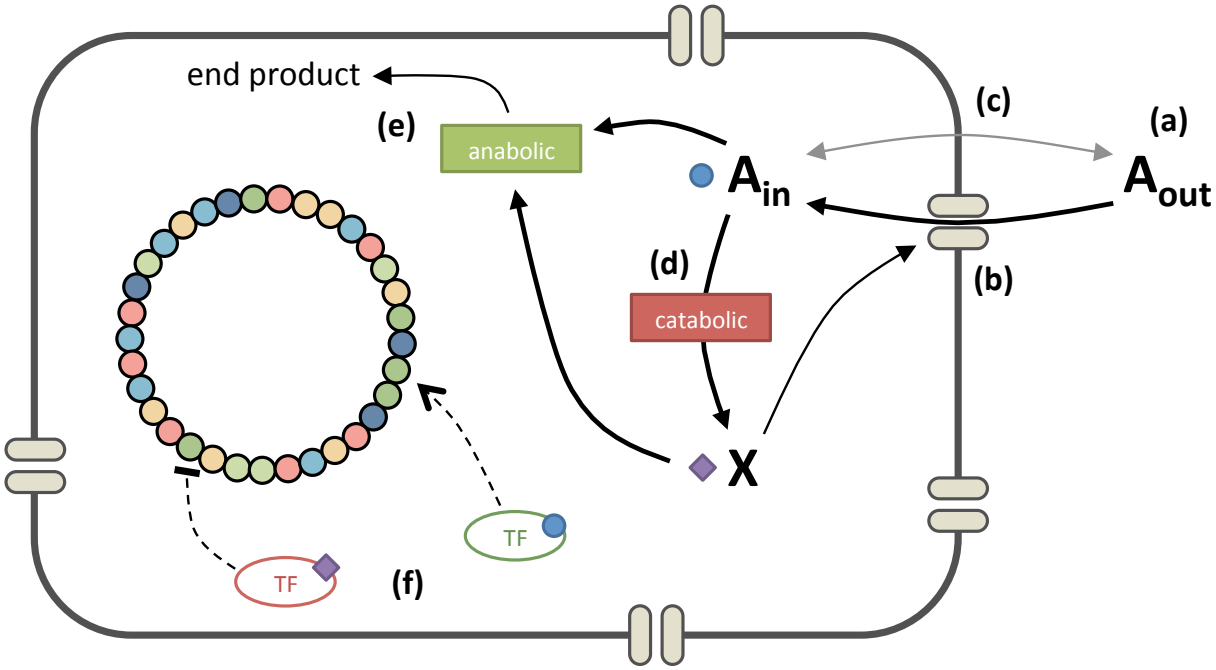


Figure I.8 – A description of the Virtual Cell model. In Virtual Cell model, digital organisms own circular genomes made of "pearls", encoding for five types of proteins. Organisms grow on an externally provided resource A **(a)**, by pumping it **(b)** or by passive diffusion through the cell's membrane **(c)**. The pumps require the consumption of an energy carrier molecule X, enzymatically produced from A by a catabolic reaction **(d)**. Both A and X molecules are required to build end products via another enzymatic reaction **(e)**. Two other protein types are transcription factors that up-regulate or down-regulate the production of proteins depending on the effect of their ligands, A or X **(f)** (inspired from Cuypers and Hogeweg 2012).

genotype-to-phenotype map is based on the extraction of an unordered set of tuples from the genotype. This set of tuples is then used to build the higher organism level in another specified space. For example, ævol uses a complex and non-linear artificial genetic code to extract a set of triplets $(m, w, h) \in \mathbb{R}^3$ from a circular and double-stranded binary sequence. In pearls-on-a-string models, the genome is an unordered list of tuples. Depending on the complexity of projection operators, the evolution on the genome structure and the genotype-to-phenotype map will not be the same. In both models, the order of the tuples does not impair the fitness, but, since the tuples are encoded locally in the genome (in coding regions, or in pearls), the modification of their order on the sequence can potentially affect long-term evolution, as demonstrated in Knibbe et al. (2007a,b).

I.5.2 Bags of tuples and artificial chemistries

When developing a new individual-based model of evolution, one important task is to define an **artificial chemistry** for this model: how to represent the various bio-molecules (DNA, RNA, proteins, metabolites, and so on) and their interactions? Artificial chemistry (AChem) is an entire field of research (Dittrich et al., 2001; Banzhaf and Yamamoto, 2015),

which is not directly in the scope of this manuscript. However, it is important to define here the most basic steps necessary to develop an AChem:

An AChem can be defined as a triplet (S, R, A), where S is the set of all possible molecules, R is a set of reaction rules representing the interactions among the molecules, and A is an algorithm describing the reaction vessel or domain and how the rules are applied to the molecules inside the vessel (Dittrich et al., 2001). The set of molecules $S = \{s_1, s_2, ..., s_n\}$ can potentially be infinite. A reaction rule $r \in R$ is a chemical equation:

$$s_1 + s_2 + ... + s_i \rightarrow s'_1 + s'_2 + ... + s'_j \tag{I.8}$$

With the reactants (or the substrates) on the left side, and the products on the right side. i is the order of the reaction. The set of reaction rules R can be defined explicitly (all possible reactions r are defined and are in finite number), or implicitly. In this example, stoichiometry is 1 for all reactants, but there is no constraint on this point. The algorithm A is applied on an instance of S, that is, a collection P of molecules. The set of chemical equations R can be solved with stochastic or deterministic methods, possibly adding spatial rules.

From this simple definition, two ways to define an AChem in the bag-of-tuples formalism are possible in first instance (Fig. I.9):

(1) Each tuple codes for a reaction rule. In this case, each organism i owns a specific set of reactions rules R_i, somehow carrying its own artificial chemistry. For instance, a tuple $(x_1, x_2, ..., x_n)$ could define the chemical equation of order $n/2$:

$$s_1 + s_2 + ... + s_{\frac{n}{2}} \rightarrow s_{\frac{n}{2}+1} + s_{\frac{n}{2}+2} + ... + s_n \tag{I.9}$$

With $x_i \equiv s_i$ (Fig. I.9.1).

(2) Each tuple codes for a chemical species, being potentially a reactant for a subset of reactions in R. In this case, R is defined once for the whole system, a reaction occurring only if all the reactants are present. For instance, let's consider the set of reaction rules R containing the reaction:

$$s_i + s_j \leftrightarrow s_i.s_j \tag{I.10}$$

And the reaction:

$$s_i.s_j \rightarrow s_k + s_j \tag{I.11}$$

With $s_i, s_j, s_k \in S$, and "." symbolizing a chemical bound. Thus, a singleton $x_j \equiv s_j$ (a tuple of length 1) catalyzes the enzymatic reaction:

$$s_i + s_j \leftrightarrow s_i.s_j \rightarrow s_k + s_j \tag{I.12}$$

To describe more precisely the reaction, it is also possible to replace the singleton x_i by a pair (x_j, c_j), with $c_j = [s_j]$. With this AChem, a tuple could encode a useless compound, if no other reactant is present (Fig. I.9.2).

The bag-of-tuples formalism thus provides a general framework to encode an artificial chemistry with a genetic sequence. As shown in part B, we chose the modeling scheme **(1)** to define the artificial chemistry in our multi-scale model of evolution (Fig. I.9.1).

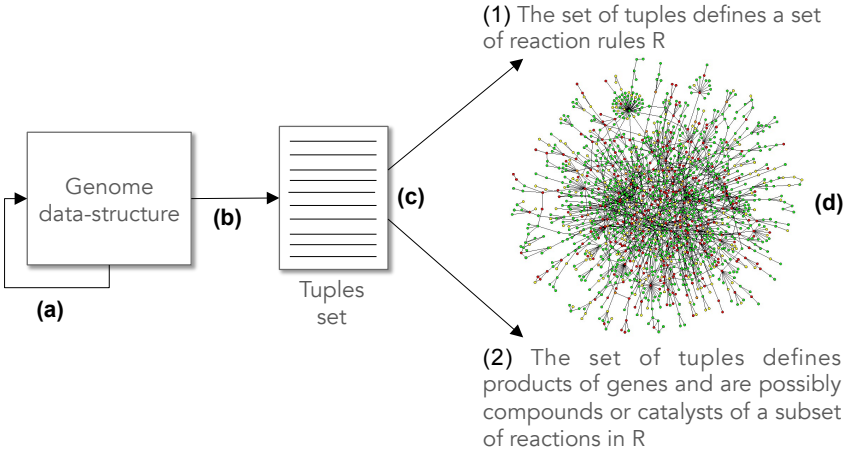


Figure I.9 – A general framework for the bag-of-tuples formalism. (a) At each replication, the genome data-structure undergoes mutations (point mutations, large rearrangements, recombinations, horizontal transfers). **(b)** A mapping, often complex and non-linear, gives a non-ordered set of tuples (the bag of tuples). **(c)** Depending on modeling choices, the set of tuples defines: (1) an independent set of reactions rules R in each organism, or (2) chemical products (proteins, catalysts, metabolites, ...) involved or not in a subset of reactions belonging to R. **(d)** The set of reaction rules defines the interactome of the organism (the biochemical network including all organism's reactions). On modeling purpose, this biochemical network can be splitted into several sub-networks (genetic regulation network, metabolic network, ...).

I.6 Outline

To summarize, we have seen that the modeling approach needed to model and study EvoEvo is multi-faceted, but can efficiently use well-defined modeling formalisms available in the literature.

On the one hand, it is essential to extend previous mathematical models used in theoretical evolutionary biology to deal with some aspects of EvoEvo. The advantage of this approach is to provide robust predictions, often accompanied with analytical solutions and mathematical proofs. In the next part of this manuscript, this approach will be exemplified with an extended version of Fisher's geometric model accounting for the evolution of phenotypic noise (part A). With this model, we made promising predictions on the evolution of phenotypic noise in the face of phenotypic complexity.

On the other hand, we have seen that some of the most salient properties of EvoEvo emerge from multi-level evolution. The usage of a multi-scaled individual-based model of evolution, including a complex and evolvable genotype-to-phenotype map is required to tackle this complexity. Two formalisms have been independently developed with the ultimate goal to deal with some of the EvoEvo aspects: the sequence-of-nucleotides formalism and the pearls-on-a-string formalism. Moreover, a methodology has been specifically developed to study *in silico* models of evolution: *in silico* experimental evolution, that provides the same experimental tools than wet experimental evolution. In the second part of this manuscript, a multi-scale model of evolution merging these different formalisms

will be presented. This model allowed us to study the evolution of niche construction and stable cross-feeding, and also the evolution of genetic regulation networks.

Part A

An extended version of Fisher's geometric model to study phenotypic noise

Chapter II

Phenotypic noise and the cost of complexity

The results presented in this chapter have been submitted to the Evolution journal.

My Umi said shine your light on the world
Shine your light for the world to see
(Mos Def – Umi Says, Black on Both Sides)

Abstract

Experimental studies demonstrate the existence of phenotypic diversity despite constant genotype and environment, and suggest that the intensity of this phenotypic noise could be evolvable. Theoretical models based on a single phenotypic character predict that during an adaptation event, phenotypic noise should be positively selected under directional selection, and then be reduced when the selection becomes stabilizing. However, it is unclear whether the (temporary) selective advantage of phenotypic noise would hold for more realistic, multidimensional phenotypes. Indeed, Fisher's geometric model of adaptation predicts a cost of complexity, where beneficial mutations become increasingly harder to fix as the number of phenotypic characters increases. Here, we extend Fisher's geometric model by adding an evolvable phenotypic noise. We show that the cost of complexity makes noise useless under directional selection, except if noise correlations between characters are evolvable. In this case, noise evolves to a specific configuration, with elevated noise towards the fitness optimum, and minimized noise in all other directions. Such an aligned noise speeds up adaptation and largely compensates for the cost of complexity. By analyzing published phenomic data of 37 yeast strains, we confirm the plausibility of intra-strain noise correlations between phenotypic characters.

II.1 Introduction

The phenotype of an organism results from a complex and non-linear cascade of developmental, physiological and regulatory processes, formalized by the concept of *genotype-to-phenotype map* (Alberch, 1991). An increasing number of experimental studies demonstrated that the genotype-to-phenotype map is not a deterministic process and can generate phenotypic diversity (Symmons and Raj, 2016), not explained by environmental interactions. Indeed, isogenic populations having the same genotype and grown in the same environment sometimes exhibit a random distribution of phenotypes, a phenomenon known as *phenotypic noise* (Yvert et al., 2013). This observed phenotypic stochasticity is mainly due to the propagation of stochastic molecular events (Elowitz et al., 2002; Jo et al., 2005; Raser and O'Shea, 2005; Bahar et al., 2006; Dar et al., 2014) through the genotype-to-phenotype map.

Recent experimental results have suggested that phenotypic noise can be tuned. Yvert et al. (2013) used single-cells phenomics (Ohya et al., 2015) on different natural strains of yeast to measure hundreds of phenotypic characters. They showed that phenotypic noise is strongly dependent on the strain background and largely character-specific (specific

strains showing elevated noise for subset of characters) but also global (a few strains displaying elevated noise for many unrelated characters). Shen et al. (2012) demonstrated the importance of "variance-controlling genes" controlling phenotypic variance in *Arabidopsis thaliana*. Boukhibar and Barkoulas (2016) reviewed experimental results also demonstrating the existence of "variance-amplifier loci", "phenotypic capacitors", or "master regulators" controlling phenotypic noise on multiple characters (Levy and Siegal, 2008; Lempe et al., 2013).

Theoretical and experimental results also tend to demonstrate that phenotypic noise has an impact on fitness. On the one hand, phenotypic noise appears to be deleterious for organisms facing a stable environment. Recently, Keren et al. (2016) demonstrated with an experimental study on *Saccharomyces cerevisiae* that phenotypic characters sensitive to variation (*i.e.*, having a sharp fitness function) exhibit low noise, in opposition to robust characters, that exhibit higher noise levels. Mineta et al. (2015) used a mathematical model from population genetics to demonstrate that elevated phenotypic noise reduces effective population size and enhances genetic drift, when the population is at the fitness optimum. On the other hand, many experimental and theoretical studies demonstrated the positive role of phenotypic noise in several *evolutionary stable strategies* (ESS) such as bet-hedging (Kussell and Leibler, 2005; Acar et al., 2008; Beaumont et al., 2009; Tsuru et al., 2011), stable mix and altruistic strategies (De Jong et al., 2011), or bacterial persistence (Balaban et al., 2004). Phenotypic noise could also be exploited by organisms in stress responses (Chalancon et al., 2012). For example, Charlebois et al. (2014) showed with a mathematical model that regulatory network motifs can enhance drug resistance by transiently increasing isogenic cell-to-cell variability. Holland et al. (2014) measured variability on fitness-dependent phenotypic characters in natural populations of yeasts. They showed that populations living in a polluted environment develop high heterogeneity as a survival strategy against adverse conditions in the wild. Experimental studies on yeast also revealed that expression noise of essential genes (Newman et al., 2006) (or "dosage-sensitive" genes, Fraser et al. 2004) is minimized to prevent harmful variations (Lehner, 2008; Wang and Zhang, 2011). Moreover, "stress-related" genes (*e.g.*, drug-resistance genes) often present high levels of expression noise (Fraser and Kærn, 2009; Zhuravel et al., 2010; Charlebois et al., 2011, 2014; Charlebois, 2015). This phenomenon was demonstrated in laboratory experiments on *Escherichia coli* (Ito et al., 2009), and on yeast (Liu et al., 2015).

Together, these studies support "*the possibility that, if noise is adaptive, microevolution may tune it in the wild. This tuning may happen on specific traits or by varying the degree of global phenotypic buffering*" (Yvert et al., 2013). Phenotypic noise thus appears as a complex and evolvable phenotypic character, exploited in many survival strategies. However, while phenotypic noise has been demonstrated to be exploited by evolution in a variety of ESS, the possible role of an evolvable phenotypic noise in directional selection, when a population must adapt to a new environment, is poorly known. A few studies examined the simple case of a single gene undergoing stochasticity of gene expression, and brought important insights. Zhang et al. (2009) suggested with a mathematical model of a single gene that elevated expression noise facilitates evolution in directional selection, because it enhances the probability to fix beneficial mutations. According to the

authors, this facilitated evolution is only possible if the fitness function is convex (there is no advantage for elevated noise in the case of a linear or a concave fitness function). A recent experimental study on *Saccharomyces cerevisiae* (Keren et al., 2016) suggests that fitness landscapes related to gene expression present a variety of curvatures, among which convex, concave or linear forms. The prediction of Zhang et al. (2009) has also been corroborated by a recent experimental study by Bódi et al. (2017), that showed that phenotypic heterogeneity due to the stochasticity of gene expression enhances the adaptive value of beneficial mutations on a specific gene of *Saccharomyces cerevisiae* in directional selection.

As stated by Eldar and Elowitz (2010), "*based on these general results, one might expect increased phenotypic noise during periods of adaptation to new environments, followed by reduction in noise when selection becomes stabilizing*". It is tempting to generalize results from single gene models to the level of an entire and complex phenotype undergoing phenotypic noise. However, this generalization is far from being straightforward. Indeed, the phenotype is the result of a complex and non-linear process involving the expression of tens on thousands of genes. Fisher's geometric model of adaptation (FGM) has been specifically conceived to study the adaptation of complex phenotypes to a new environment. Using it, Fisher (1930) suggested that organisms may pay a cost to the "complexity" of their phenotype, beneficial mutations becoming increasingly harder to fix when the number of phenotypic characters under selection increases. As demonstrated by Orr (2000), the cost of complexity is a robust result of FGM, little affected by organismal modularity (Welch and Waxman, 2003). In 2006, Martin and Lenormand (2006) compared the distributions of fitness effects of mutations across several species (from *Escherichia coli* to fruit flies). Their results suggest that there may be a cost to phenotypic complexity, even if it is weaker than predicted by theoretical studies. Could the evolution of phenotypic noise and its adaptive value be impacted by the complexity of the phenotype? And if yes, what are the consequences on the predictions made by Zhang et al. (2009) and Eldar and Elowitz (2010) for a population evolving a single phenotypic character?

To address these questions, we extended FGM with a model of evolvable phenotypic noise. Based on an analysis of yeast phenomic data, we allowed for evolvable correlations between the noise levels on the various characters. We studied how phenotypic noise would evolve when a population of asexual organisms is placed under directional selection, and must adapt to a novel environment. With this model, named σFGM, we showed that phenotypic noise is indeed beneficial for organisms evolving a single phenotypic character in directional selection (Zhang et al., 2009; Eldar and Elowitz, 2010). However, this benefit is quickly impaired when the number of characters increases. Nonetheless, this cost of complexity on the phenotypic noise can be compensated if noise correlations between characters are allowed to evolve. In this case, phenotypic noise evolves towards a flattened, one-dimensional configuration in the phenotypic space, with elevated noise in the direction of the fitness optimum, and minimized noise in all other directions. When phenotypic noise evolves this pattern, it strongly facilitates evolution towards the fitness optimum by producing very fit organisms and by increasing the probability to fix beneficial mutations. In these conditions, the convergence time towards the fitness optimum is even faster than for organisms having no phenotypic noise (as in canonical FGM),

thereby demonstrating that an evolvable phenotypic noise can significantly compensate for the cost of complexity, as defined by Fisher (1930). Thus, our results suggest that such a non-isotropic and correlated phenotypic noise could be exploited by evolution, and call for further experiments to assess the functional nature of phenotypic noise.

II.2 Methods

Analysis of phenomic data in various strains of yeast

In order to guess what would be the general shape of the phenotypic noise in real organisms, we analyzed phenomic data provided by Yvert et al. (2013). The authors monitored 125 phenotypic characters on isogenic populations of 37 strains of yeast, in order to characterize phenotypic diversity at a single-cell resolution. We used raw datasets provided by the authors (freely available at http://sunlight.k.u-tokyo.ac.jp/wild37noise/index.html) to study intra-strain isogenic phenotypic noise.

The purpose of the analysis was to determine whether intra-strain variability presents correlations between characters *once inter-strain correlations between characters have been removed*.

We first processed inter-strain variability. The idea was to find a phenotypic space in which there is as little character-specific variability and correlation as possible. Here we had 37 isogenic strains, hence 37 genotypes. We defined the "phenotype" of a strain/genotype as the vector of mean trait values, computed over all cells from this strain/genotype. We then defined the "centered phenotype" of a strain/genotype by removing the grand mean of each character. The singular value decomposition of the 37×125 matrix of centered strain phenotypes gave us a set of orthonormal linear combinations of characters. By construction, when the centered strain phenotypes are expressed according to these new characters, they lose all their pairwise correlations, implying that the variance-covariance matrix is diagonal for those new characters. Moreover, we normalized the variance of each new strain phenotype to 1, such that the 37 new strain phenotypes were isotropically distributed.

This new base is the closest analogy we could think of to the phenotypic space in the classical version of Fisher's geometric model. Fisher's phenotypic space is orthogonal and normalized, and mutations on the genotype cause phenotypic traits to vary independently and with the same amplitude, according to an isotropic mutational distribution.

The second step was to project intra-strain single-cell data in Fisher's space, and to compute the possible remaining correlations of intra-strain phenotypic variability in this space.

The whole analysis is presented in more details in Appendix II.5.2.

II.2.1 Evolving phenotypic noise in Fisher's geometric model

In Fisher's geometric model, the phenotype of an organism is represented as a point $\boldsymbol{z} = (z_1, z_2, ..., z_n)^T \in \mathbb{R}^n$ (T being the matrix transposition operator) where n is the number of phenotypic characters under selection. Fisher (1930) used the term "phenotypic complexity" to refer to the dimensionality n. The absolute fitness of this organism is determined by its Euclidean distance $d(\boldsymbol{z}, \boldsymbol{z_{opt}}) = \|\boldsymbol{z} - \boldsymbol{z_{opt}}\|_2$ from the fitness optimum $\boldsymbol{z_{opt}}$, the absolute fitness function $W(\boldsymbol{z})$ commonly being a simple Gaussian-shaped function reading:

$$W(\boldsymbol{z}) = \exp\left[-\frac{d(\boldsymbol{z}, \boldsymbol{z_{opt}})^2}{2}\right]. \tag{II.1}$$

Usually, $\boldsymbol{z_{opt}}$ lies at the origin of the euclidean space ($\boldsymbol{z_{opt}} = \boldsymbol{0}$). In FGM, mutations are modeled as a random perturbation of the ancestral phenotype $\boldsymbol{z}$. It is usually assumed that a mutation can affect multiple trait values (an hypothesis known as the "universal pleiotropy assumption", Paaby and Rockman 2013). The mutated phenotype $\boldsymbol{z'} \in \mathbb{R}^n$ is a random vector drawn from a n-dimensional multivariate normal distribution centered at $\boldsymbol{z}$, with a $n \times n$ covariance matrix $\boldsymbol{C_z}$,

$$\boldsymbol{z'} \sim \mathcal{N}_n(\boldsymbol{z}, \boldsymbol{C_z}). \tag{II.2}$$

The distribution of $\boldsymbol{z'}$ is often assumed to be isotropic around $\boldsymbol{z}$, such that mutations have no preferential direction and can affect all characters similarly; in that case $\boldsymbol{C_z}$ can be written as $\sigma_z^2 \boldsymbol{I_n}$, with $\boldsymbol{I_n}$ the $n \times n$ identity matrix, and σ_z the standard deviation of the mutation sizes along each axis. Usually, initial conditions are a maladapted clonal population of asexual organisms, sitting at a certain distance of the optimum $\boldsymbol{z_{opt}}$ ($d(\boldsymbol{z}, \boldsymbol{z_{opt}}) \gg 0$). Then, the work consists in studying the bout of adaptation towards the optimum.

Fisher's geometric model implies some well-known assumptions helping mathematical resolution of the equations (Martin, 2014): **(i)** the distribution of all random variables have finite mean and variance and satisfy Lindeberg's conditions (the central limit theorem can be applied), **(ii)** the fitness function is twice differentiable and admits at least one non-degenerate optimum, **(iii)** mutations have mild effects on the phenotype (mutational events remain "local"), **(iv)** each mutation potentially affect all trait values (an assumption known as the universal pleiotropy assumption), and **(v)** the variety of mutants is very large, such that trait values vary continuously (an assumption known as the "infinite-alleles" approximation).

We now present an extended version of Fisher's geometric model accounting for an evolvable phenotypic noise. We called this extended model σFGM. In canonical FGM, the phenotype $\boldsymbol{z}$ of each organism is deterministic and fixed by mutations. Decades ago, Russell Lande paved the way to the usage of mathematical models of population genetics to study the impact on fitness of stochastic events (Lande, 1976) and correlations between characters (Lande and Arnold, 1983). In line with this work, we represented the phenotype of each organism by a random variable. We assumed that the phenotype $\boldsymbol{z}$ follows a normal multivariate distribution $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. The vector $\boldsymbol{\mu} \in \mathbb{R}^n$ is the mean phenotype

of the organism, and the $n \times n$ matrix $\boldsymbol{\Sigma}$ is the covariance matrix of the phenotypic noise around its mean $\boldsymbol{\mu}$. Thus, the phenotype of a given organism can be characterized by a deterministic component $\boldsymbol{\mu}$ (equivalent to $\boldsymbol{z}$ in canonical FGM) and a stochastic component characterized by the covariance matrix $\boldsymbol{\Sigma}$ representing the phenotypic noise.

The fitness of an organism is given by its realized phenotype $\boldsymbol{z}$, the fitness function $W(\boldsymbol{z})$ being defined in Equation II.1. However, $W(\boldsymbol{z})$ is now a random variable, so it is useful to look at the expected fitness $\overline{W}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ of an organism with parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$,

$$\overline{W}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \int_{\mathbb{R}^n} W(\boldsymbol{z}) p(\boldsymbol{z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) d\boldsymbol{z}, \tag{II.3}$$

where $p(\boldsymbol{z}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the density function of the law $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$,

$$p(\boldsymbol{z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{\sqrt{(2\pi)^n \det \boldsymbol{\Sigma}}} \exp\left[-\frac{1}{2}(\boldsymbol{z} - \boldsymbol{\mu})^T \boldsymbol{\Sigma}^{-1} (\boldsymbol{z} - \boldsymbol{\mu})\right]. \tag{II.4}$$

Compared to canonical FGM, a distinction is made between the realized phenotype $\boldsymbol{z}$ and the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ characterizing the distribution of $\boldsymbol{z}$. Here, $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ undergo mutations and are the inherited properties of the phenotype, while the realized phenotype $\boldsymbol{z}$ is not inherited. Thus, we purposely place ourselves in a worst-case scenario for the evolution of noise, meaning a scenario where a high noise level is not trivially expected to be selected (Charlebois et al., 2011).

Our model allows for correlated phenotypic noise through the covariance matrix $\boldsymbol{\Sigma}$. This choice is justified both by mathematical and experimental considerations: let us define a phenotypic space where mutations of the mean phenotype $\boldsymbol{\mu}$ are orthogonalized and normalized, *i.e.*, mutations on $\boldsymbol{\mu}$ are isotropically distributed in this space. As in canonical FGM (Eq. II.2), the mutation probability distribution is isotropic. If we make the reasonable hypothesis that molecular mechanisms controlling the mean phenotype and its variance are not the same (see *e.g.*, Viñuelas et al. 2012), there is no reason to suppose that phenotypic noise, nor its mutations, are also orthogonal in this phenotypic space, *i.e*, that the phenotypic noise is isotropic. Consequently, it is necessary to consider a correlated phenotypic noise. This modeling choice is supported by a recent experimental work by Cressler et al. (2017) on *Daphnia pulicaria*, a species of freshwater zooplankton. By growing genetic variants in a wide range of food quality environments, and by measuring three important life-history characters (growth, reproduction and longevity), they showed that there is no significant genetic correlations between characters, while there are significant non-genetic correlations. We also performed additional analyses on recently published single-cell data measuring hundreds of phenotypic characters from different species of yeast (Yvert et al., 2013). Our results suggest that phenotypic noise is indeed correlated in a space where mean phenotypic character variations between strains are uncorrelated (see Appendix II.5.2 and first part of the Results section).

We now describe in more details our modeling of the genotype-to-phenotype map in σFGM. The $n \times n$ covariance matrix $\boldsymbol{\Sigma}$ is a real, symmetric and positive-definite matrix. As such, it admits an eigenvalue decomposition

$$\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T. \tag{II.5}$$

The matrix $\boldsymbol{D}$ is a diagonal matrix containing the n positive eigenvalues $\boldsymbol{\sigma}^2 = (\sigma_1^2, \sigma_2^2, ..., \sigma_n^2)^T$ of $\boldsymbol{\Sigma}$, with $\boldsymbol{D} = \text{diag}(\boldsymbol{\sigma}^2)$. The matrix $\boldsymbol{U}^T$ is a real orthogonal matrix decomposed as a product of rotations ($\boldsymbol{U}$ can be chosen as not to have any reflections, see Anderson et al. 1987).

We thus made the following geometrical interpretation: $\boldsymbol{\Sigma}$ defines an hyper-ellipse in $\mathbb{R}^n$ with semi-axis orientations are given by the column vectors of $\boldsymbol{U}$, and the semi-axis lengths by the square roots of the eigenvalues. Geometrically its makes sense to express mutations in the phenotypic noise by mutations in the lengths and in the orientations of the semi-axes of the hyper-ellipse. Therefore, we define $\boldsymbol{\Sigma}$ by a vector of n lengths $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, ..., \sigma_n)^T$ and a vector of $n(n-1)/2$ rotation angles $\boldsymbol{\theta} = (\theta_1, \theta_2, ..., \theta_{n(n-1)/2})^T$.

The matrix $\boldsymbol{U}$ is built by applying successive rotations (in that specific order),

$$\boldsymbol{U} = \prod_{i=1}^{n-1} \prod_{j=i+1}^{n} \boldsymbol{G}_{ij}(\theta_{n(i-1)+(i-1)(i-2)/2+j-i}) \tag{II.6}$$

with $\boldsymbol{G}_{ij}(\theta)$ the Givens matrix associated to the rotation between axes i and j, with an angle θ.

Metzger et al. (2015) suggested with an experimental study that the expression noise of TDH3 gene may evolve faster than its mean expression. To test this hypothesis, we decided to mutate $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\theta}$ independently in σFGM. Indeed, in this model the mean phenotype $\boldsymbol{\mu}$ results from the genotype-to-phenotype map and is defined in an abstract way, such that there is no particular reason to consider it correlated with the phenotypic noise, as it is the case for gene expression for example (Ozbudak et al., 2002). Similarly to Equation II.2, the mutated mean phenotype $\boldsymbol{\mu}'$ follows a multivariate normal distribution $\boldsymbol{\mu}' \sim \mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{C_\mu})$, the mutated phenotypic noise amplitudes vector is $\boldsymbol{\sigma}' \sim \mathcal{N}_n(\boldsymbol{\sigma}, \boldsymbol{C_\sigma})$, and the mutated phenotypic noise orientations vector is $\boldsymbol{\theta}' \sim \mathcal{N}_{n(n-1)/2}(\boldsymbol{\theta}, \boldsymbol{C_\theta})$. $\boldsymbol{C_\mu}$, $\boldsymbol{C_\sigma}$ and $\boldsymbol{C_\theta}$ are three constant covariance matrices of sizes $n \times n$ for $\boldsymbol{C_\mu}$ and $\boldsymbol{C_\sigma}$, and $n(n-1)/2 \times n(n-1)/2$ for $\boldsymbol{C_\theta}$.

In summary, σFGM includes three classes of variables: $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\theta}$. The mean phenotype of each organism is represented by a vector $\boldsymbol{\mu}$. The phenotypic noise of each organism is modeled by a multivariate normal law $\mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, $\boldsymbol{\Sigma}$ being decomposed in its semi-axis sizes $\boldsymbol{\sigma}$, and its rotation angles $\boldsymbol{\theta}$. σFGM also includes three constant mutational parameters $\boldsymbol{C_\mu}$, $\boldsymbol{C_\sigma}$ and $\boldsymbol{C_\theta}$, and a fitness function $W(\boldsymbol{z})$ being defined here as a simple Gaussian-shaped function (Eq. II.1). Canonical Fisher's geometric model is a particular case of σFGM, when the noise vanishes and there is no mutation on noise amplitudes (*i.e.*, $\boldsymbol{\sigma} \to \boldsymbol{0}$ and $\boldsymbol{C_\sigma} = \boldsymbol{0}$; see details in Appendix II.5.4).

While σFGM admits a relatively low number of parameters, it generates complex and non-intuitive outcomes. Besides the analytical resolution of equations in the simplest cases (see below), we had to develop numerical tools to solve the equations in the most complex situations, especially for elevated phenotypic complexity. Our numerical approaches are presented below. All the mathematical variables used in this manuscript are listed in Table II.1.

Table II.1 – List of mathematical variables.

Variable notation	Type	Description
n	$\mathbb{N}$	Dimension of phenotypic space
z	$\mathbb{R}^n$	Phenotype
W	$\mathbb{R}^n \to \mathbb{R}$	Fitness function
d	$\mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$	Euclidean distance to optimal phenotype
$\boldsymbol{z_{opt}}$	$\mathbb{R}^n$	Optimal phenotype
$\boldsymbol{C_z}$	$\mathbb{R}^{n\times n}$	Covariance matrix of mutations in FGM
$\boldsymbol{\mu}$	$\mathbb{R}^n$	Mean phenotype in σFGM
$\boldsymbol{\Sigma}$	$\mathbb{R}^{n\times n}$	Covariance matrix of phenotypic noise in σFGM
$\overline{W}$	$\mathbb{R}^n \times \mathbb{R}^{n\times n} \to \mathbb{R}$	Expected fitness given $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$ in σFGM
$\boldsymbol{D}$	$\mathbb{R}^{n\times n}$	Diagonal matrix of eigenvalues of $\boldsymbol{\Sigma}$
$\boldsymbol{U}$	$\mathbb{R}^{n\times n}$	Real orthogonal matrix of the eigenvectors of $\boldsymbol{\Sigma}$
$\boldsymbol{\sigma}^2$	$\mathbb{R}^n$	Eigenvalues of $\boldsymbol{\Sigma}$
$\boldsymbol{\theta}$	$\mathbb{R}^{n(n-1)/2}$	Pairwise plane rotations to generate $\boldsymbol{U}$
$\boldsymbol{C_\mu}$	$\mathbb{R}^{n\times n}$	Covariance matrix of mutations of $\boldsymbol{\mu}$ in σFGM
$\boldsymbol{C_\sigma}$	$\mathbb{R}^{n\times n}$	Covariance matrix of mutations of $\boldsymbol{\sigma}$ in σFGM
$\boldsymbol{C_\theta}$	$\mathbb{R}^{n(n-1)/2\times n(n-1)/2}$	Covariance matrix of mutations of $\boldsymbol{\theta}$ in σFGM
s_μ	$\mathbb{R}$	$\boldsymbol{C_\mu} = s_\mu^2 I_n$
s_σ	$\mathbb{R}$	$\boldsymbol{C_\sigma} = s_\sigma^2 I_n$
s_θ	$\mathbb{R}$	$\boldsymbol{C_\theta} = s_\theta^2 I_{n(n-1)/2}$

II.2.2 A numerical implementation of σFGM

As shown in the Results section below, it is possible to perform an analytical resolution of σFGM equations in simple conditions, *e.g.* when organisms evolve a single phenotypic character, or when the phenotypic noise is isotropic. However, for the most complex scenarios, a numerical approach is needed to solve the evolutionary trajectories through time.

To this aim, we consider a Markov process where random events are the appearance of new mutants in the population. This stochastic process is known in population dynamics as the *stochastic branching process* of Galton-Watson (Watson and Galton, 1875), with jumps corresponding to mutated offspring. This branching process proceeds as follows. We consider a finite population with $N(t)$ organisms at time t, where each organism is characterized by a unique triplet $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$. From this triplet, a realized phenotype $\boldsymbol{z}$ is drawn from the multivariate normal distribution $\mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. This realized phenotype remains constant for the lifetime of the organism. Time evolves continuously.

(0) At the beginning of a simulation, a isogenic population of N_0 organisms having the same triplet $(\boldsymbol{\mu}_0, \boldsymbol{\sigma}_0, \boldsymbol{\theta}_0)$ is generated;

(1) During a time interval Δt, the probability for an organism to produce an offspring during the interval $t + \Delta t$ is $p_{birth} = W(\boldsymbol{z}) \times \Delta t + \mathcal{O}(\Delta t^2)$. $\mathcal{O}(\Delta t^2)$ is due to the possible occurrence of other events in the interval Δt (branchings, or deaths) that introduce a small error in the probability estimation. If Δt is small enough, this

error is negligible. To avoid to many events to occur during any time interval, Δt is rescaled such that the best fitness W_{max} in the population at time t is always equal to 0.1 ($\Delta t = 0.1/W_{max}$);

(2) At birth, an organism acquires the mutated triplet $(\boldsymbol{\mu}', \boldsymbol{\sigma}', \boldsymbol{\theta}')$ derived from its parent's one $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$;

(3) To keep the population size constant around a target size N_{eq}, we introduced a death process. During a time interval Δt, every organism as the same probability p_{death} to die, proportional to the population size N:

$$p_{death} = max\left(0,\ \frac{N - N_{eq}}{N}\right). \tag{II.7}$$

For the sake of simplicity, we made an additional assumption on the mutational process. We considered that mutations are isotropic for $\boldsymbol{\mu}$, $\boldsymbol{\sigma}$ and $\boldsymbol{\theta}$. Each mean trait value μ_i independently mutates through a normal distribution $\mathcal{N}(0, s_\mu^2)$. Each semi-axis size σ_i independently mutates through a normal distribution $\mathcal{N}(0, s_\sigma^2)$. Finally, each rotation angle θ_i independently mutates through a normal distribution $\mathcal{N}(0, s_\theta^2)$. Thus, by varying the relative values of s_μ, s_σ and s_θ, it is possible test the hypothesis of Metzger et al. (2015) (see above) at the level of the phenotype. The stochastic branching process is simulated with a time-adaptive tau-leaping algorithm (Gillespie, 2007). An example of the temporal dynamics is represented on Figure II.5.1, for $n = 10$ dimensions. The code of the numerical solver is freely available in Script II.5.7, and is distributed under the open source GNU General Public License. Details on the numerical solver are given in Appendix II.5.3.

II.3 Results

II.3.1 Phenomic data on 37 strains of yeast reveals correlated phenotypic noise

Using raw datasets provided by Yvert et al. (2013), we determined whether intra-strain variability presents correlations between characters *once inter-strain correlations between characters have been removed*. We found that intra-strain phenotypic noise is indeed correlated in many ways, for all the 37 strains.

For example on Figure II.1a, we show what would be an uncorrelated phenotypic noise for each strain for the two first principal components (PC1 and PC2) of the Fisher's space (the shape of the phenotypic noise of each strain is symbolized by an ellipse representing the standard deviation of the associated bivariate normal law, rescaled by a factor 0.002). On Figure II.1b, the real observed phenotypic noise is represented, showing noise correlations for all the strains. The most variable combinations of phenotypic characters (following

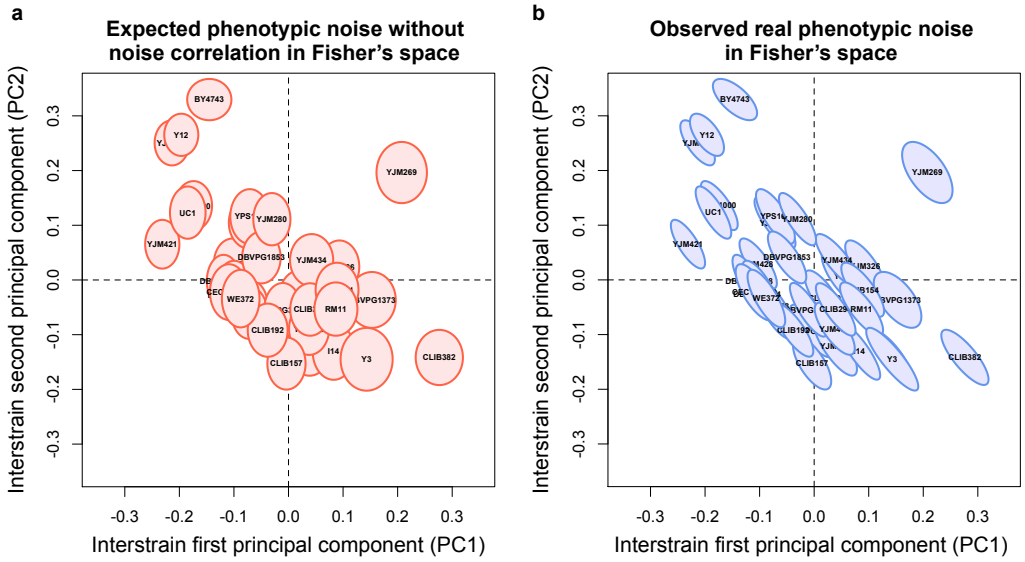


Figure II.1 – Yeast intra-strain phenotypic noise is correlated in Fisher's space. A singular value decomposition (SVD) is performed on the mean trait values of each of the 37 yeast strains. This space is similar to the phenotypic space used in Fisher's geometric model, where phenotypic characters mutate independently and with the same amplitude (*i.e.* mean phenotype mutations are isotropic in this space). For this reason, we called this space "Fisher's space". We then projected single-cell data of each strain in this space. We identified the two Fisher's space axes showing most elevated noise correlation in mean, for all strains: they correspond to the first two components of Fisher's space (PC1 and PC2). **a,** Expected phenotypic noise for each strain without noise correlation between Fisher's space axes (each axis representing a linear combination of phenotypic characters). The shape of the phenotypic noise of each strain is symbolized by an ellipse representing the standard deviation of the associated bivariate normal law. Each ellipse is tagged with the corresponding strain name. The size of the ellipses are rescaled by a factor 0.002 to better distinguish them. The coordinates of the center of each ellipse correspond to the real position of the corresponding strain in the Fisher's space (from real data). **b,** Real observed phenotypic noise is represented, showing noise correlation between PC1 and PC2 axes, for all the strains.

PC1 and PC2 axes) between strains are also those exhibiting the most correlated intra-strain phenotypic noise. Thus, if one assume that phenotypic differences across strains are adaptive, our result suggests that the phenotypic characters most exposed to directional selection are also the ones with the most correlated phenotypic noise between characters.

Our analysis is described in details in Appendix II.5.2, in Data II.5.5 in Data II.5.6, and in Script II.5.8.

II.3.2 Analytical and numerical study of σFGM

Our analytical and numerical approach followed three steps, as presented in Figure II.2. **(1)** We first studied σFGM in the case of organisms evolving a single phenotypic character, in order to evaluate previous statements (Zhang et al., 2009; Eldar and Elowitz, 2010; Bódi

et al., 2017) (Fig. II.2a). **(2)** We then studied σFGM for more complex phenotypes when the phenotypic noise is isotropic, to test whether the fitness benefit of phenotypic noise for a single character is maintained for higher phenotypic complexity (Fig. II.2b). **(3)** Finally, we studied analytically and numerically the most general case in σFGM, where noise amplitudes on each character, as well as noise correlations between characters are evolvable, as suggested by our analysis of the phenomic data provided by Yvert et al. (2013) and by the experimental study by Cressler et al. (2017) (Fig. II.2c).

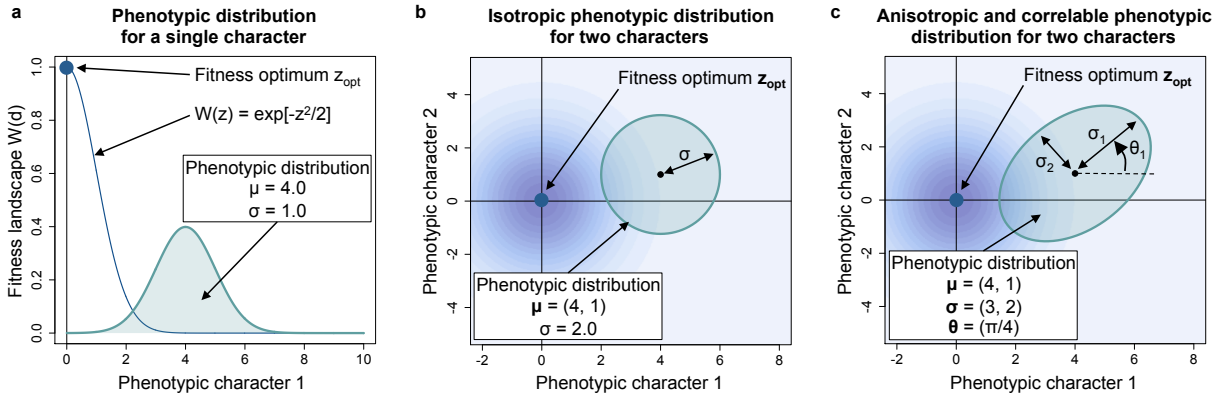


Figure II.2 – Three successive approaches to model phenotypic noise in Fisher's geometric model. a, Example of an evolvable phenotypic noise on a single phenotypic character ($n = 1$). $\mu = 4$ and $\sigma = 1$ (black box). The phenotypic distribution follows $z \sim \mathcal{N}(\mu, \sigma^2)$ (blue area). The one-dimensional fitness landscape $W(z) = \exp\left[-z^2/2\right]$ is represented in dark blue (dark blue dot: fitness optimum at $z_{opt} = 0$). **b,** Example of an evolvable isotropic phenotypic noise on two phenotypic characters ($n = 2$). $\boldsymbol{\mu} = (4, 1)^T$ and $\sigma = 2$ (black box). The standard deviation of the bivariate and isotropic phenotypic distribution is represented by the disc colored in blue. The fitness landscape $W(\boldsymbol{z})$ is represented by a gradient of blue (dark blue dot: fitness optimum at $\boldsymbol{z_{opt}} = (0, 0)^T$). **c,** Example of an evolvable anisotropic and correlated phenotypic noise on two phenotypic characters ($n = 2$). $\boldsymbol{\mu} = (4, 1)^T$, $\boldsymbol{\sigma} = (3, 2)^T$ and $\boldsymbol{\theta} = (\pi/4)$ (black box). The standard deviation of the bivariate phenotypic distribution is represented by the blue ellipse colored in blue. The fitness landscape $W(\boldsymbol{z})$ is represented by a gradient of blue (dark blue dot: fitness optimum $\boldsymbol{z_{opt}} = (0, 0)^T$).

II.3.2.1 Elevated phenotypic noise is beneficial in directional selection for a single phenotypic character.

We first studied σFGM in the simple case of the evolution of a single phenotypic character. The phenotypic noise is then reduced to an univariate normal law $\mathcal{N}(\mu, \sigma^2)$, with μ the single mean trait value and σ the standard deviation of the phenotypic noise on this character (Fig. II.2a). To understand what would be the selective pressures on μ and σ in the phenotypic space, we analytically studied the sub-population fitness $\overline{W}(\mu, \sigma)$, under the hypothesis of an infinite population (Eq. II.3). The mathematical details of our analytical and numerical approaches are presented in Appendix II.5.3 and Appendix II.5.4. Figure II.3 shows that, if evolvable, phenotypic noise should increase during directional selection, and then decrease when selection becomes stabilizing, as predicted in Eldar and Elowitz (2010). When the population is far from the fitness optimum z_{opt}, it is

beneficial to increase the phenotypic noise (Fig. II.3 green area). On the contrary, when the population is near the fitness optimum, or when noise amplitude is too high, it is better to decrease phenotypic noise, as shown in previous studies (Mineta et al., 2015; Keren et al., 2016) (Fig. II.3 red area).

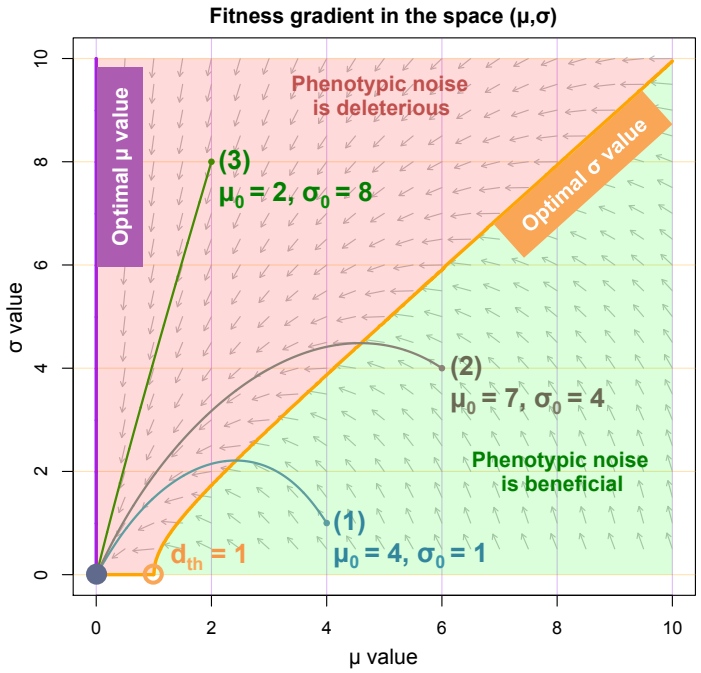


Figure II.3 – Variations of the sub-population fitness $\overline{W}(\mu, \sigma)$ depending on μ and σ values. Following the purple curve, the optimal μ value given a σ value. Following the orange curve, the optimal σ value given a μ value. Green area: it is beneficial to increase the phenotypic noise. Red area: it is beneficial to reduce the phenotypic noise. Three trajectories following the fitness gradient are represented **(1)** in blue (initial values: $\mu_0 = 4, \sigma_0 = 1$), **(2)** in brown (initial values: $\mu_0 = 6, \sigma_0 = 4$) and **(3)** in green (initial values: $\mu_0 = 2, \sigma_0 = 8$). Black dot: fitness optimum z_{opt}. Orange circle: inflection point $d_{th} = 1$ of the fitness landscape $W(z)$. Grey arrows indicate the direction of the fitness gradient, but not its amplitude.

Thus, depending on the euclidean distance from the fitness optimum, there exists an optimal value of σ giving the highest fitness value, as shown by the orange curve on Figure II.3. At the critical distance $d_{th} = 1$ (Fig. II.3 orange circle, Appendix II.5.4), which corresponds to the inflection point of the fitness function $W(z)$ (Zhang et al., 2009), phenotypic noise is always deleterious and must be minimized by organisms. However, reducing the euclidean distance from the fitness optimum is always beneficial, whatever the value of σ, as shown by the purple curve on Figure II.3. As exemplified by trajectories **(1)**, **(2)** and **(3)** on Figure II.3, a population adapting to the new fitness optimum (Fig. II.3 dark blue dot) will increase or decrease its phenotypic noise depending on initial conditions. The prediction of Eldar and Elowitz (2010) corresponds to trajectory **(1)**, *e.g.*, when a population anciently in stabilizing selection (with reduced phenotypic noise) must adapt to a new environment. Trajectories **(2)** and **(3)** could correspond *e.g.*, to a single stress-related gene, with elevated phenotypic noise at the moment of the environmental shift.

Then, in the case of a population evolving a single phenotypic character, our results confirm the claim that phenotypic noise is beneficial in directional selection, when the population is far from the fitness optimum, and that phenotypic noise is deleterious in stabilizing selection, when the population reaches the fitness optimum (Eldar and Elowitz, 2010). As previously demonstrated by Zhang et al. (2009), a condition to the positive selection of phenotypic noise is the existence of a convex fitness landscape. In a recent experimental study on *Saccharomyces cerevisiae*, Keren et al. (2016) suggest that it could be the case for many traits.

II.3.2.2 There is a cost of complexity on isotropic phenotypic noise in directional selection

When organisms evolve a single phenotypic character, σFGM is in accordance with previous results (Zhang et al., 2009; Eldar and Elowitz, 2010; Bódi et al., 2017). However, the fitness effect of phenotypic noise is unclear when the phenotypic complexity increases, since the evolution of phenotypic noise can potentially be impeded by a cost of complexity, as defined by Fisher (1930). To address this question, we first generalized the one-dimensional case by increasing the number of phenotypic characters, but keeping an isotropic phenotypic noise (similar to a "global" phenotypic noise affecting the whole phenotype, Yvert et al. 2013). An isotropic noise is applied to the mean phenotype $\boldsymbol{\mu}$ of an organism, by independently varying each trait value μ_i with the same amplitude σ (Fig. II.2b). In σFGM, this scenario corresponds to constrain the evolution of the covariance matrix $\boldsymbol{\Sigma}$ such that $\boldsymbol{\Sigma} = \sigma^2 \boldsymbol{I_n}$, and to remove noise correlations ($\boldsymbol{\theta} = \mathbf{0}$). We repeated the analysis made for a single phenotypic character, but ranging from a single to 50 characters. The details of our analytical and numerical approaches are presented in Appendix II.5.3 and Appendix II.5.4.

The results, presented in Figure II.4, show that the advantage of an isotropic phenotypic noise is quickly impeded when the phenotypic complexity increases. First, the euclidean distance d_{th} below which phenotypic noise is deleterious increases with the phenotypic complexity. An analytical resolution shows that d_{th} increases as the square root of the number of phenotypic characters ($d_{th} = \sqrt{n}$, Fig. II.4a, and Appendix II.5.4). Second, the fitness gain brought by an optimal phenotypic noise (Fig. II.4a orange and grey curves) quickly vanishes with phenotypic complexity. Indeed, when organisms have more than one phenotypic characters ($n > 1$), the beneficial value of phenotypic noise becomes rapidly negligible (Fig. II.4b). The maximal fitness gain when phenotypic noise is optimal also rapidly falls down, with *e.g.*, a maximal fitness gain for two characters ($n = 2$) representing only $\sim 36\%$ of the maximal gain for a single character (Fig. II.4b black dots).

These results show that predictions based on the evolution of a single character cannot be generalized as is at the level of the phenotype, when the phenotypic noise is isotropic, as it undergoes a cost of complexity.

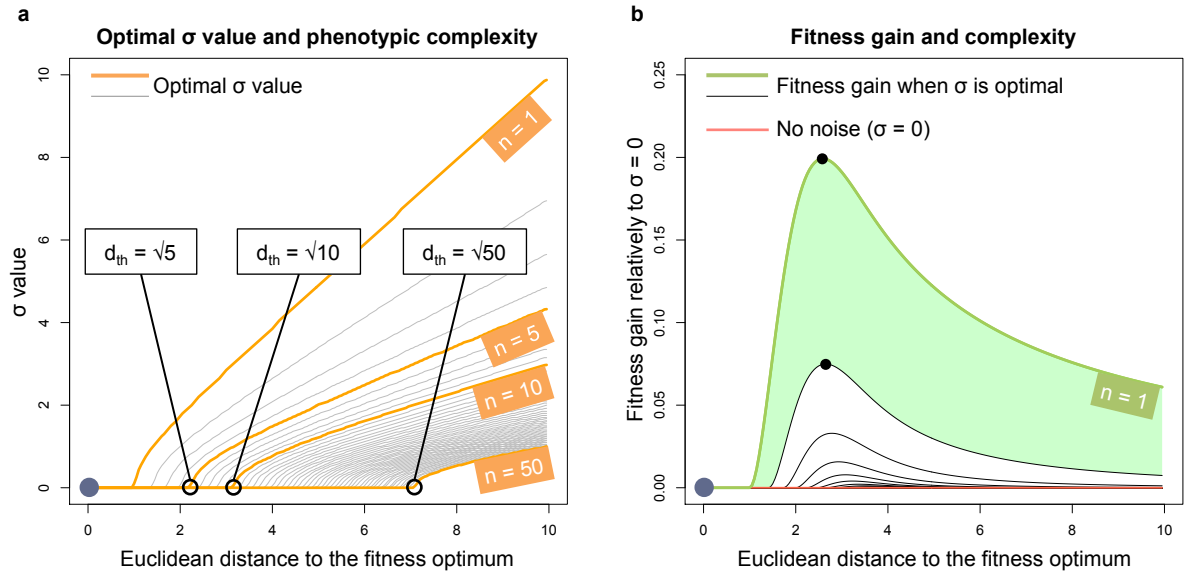


Figure II.4 – Effects of phenotypic complexity on isotropic phenotypic noise fitness gain. a, Variation of the optimal σ value, depending on the euclidean distance. x axis: the euclidean distance from the fitness optimum is varied from 0 to 10, for all phenotypic complexities ($d = \|\boldsymbol{\mu}\|$). y axis: the amplitude σ of the phenotypic noise is varied from 0 to 10. Grey curves: optimal σ value for each phenotypic complexity. Phenotype complexities $n = 1$, $n = 5$, $n = 10$ and $n = 50$ are highlighted in orange. Black circle: the critical euclidean distance d_{th} below which phenotypic noise is always deleterious is equal to $\sqrt{n}$ (exemplified here for $n = 5$, $n = 10$ and $n = 50$). **b,** Fitness gain when σ is optimal, compared to canonical FGM scenario with no phenotypic noise ($\sigma = 0$). Dark blue dot: fitness optimum. Black curves: fitness gain when isotropic noise is optimal. Phenotypic complexity $n = 1$ is highlighted in green. The green area indicates the difference of fitness gain between $n = 1$ and $n = 2$. Red line: no fitness gain (no phenotypic noise scenario). Black dots: maximal fitness gains when $n = 1$ and $n = 2$. The maximal fitness gain of isotropic noise when $n = 2$ represents $\sim 36\%$ of the maximal gain for a single character.

II.3.2.3 Anisotropic and correlated phenotypic noise is beneficial when aligned with the fitness optimum

Finally, we studied the most general case in σFGM, as described in Methods. Noise amplitude on each character, as well as noise correlations between characters are evolvable (Fig. II.2c). As a first step, we analyzed the model in a static situation, with no mutational process (as for previous results), in order to guess what would be the selective pressures on the phenotypic noise.

We show mathematically (Appendix II.5.4) that when the population is far from the fitness optimum, the best configuration (*i.e.*, the one that gives the best fitness advantage) is a flattened, one-dimensional phenotypic noise, with elevated noise in the direction of the fitness optimum and no noise in all other directions. Any other form of phenotypic noise (isotropic or not perfectly aligned with the fitness optimum) gives a lower sub-population fitness $\overline{W}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. This does not mean that a badly aligned phenotypic noise is deleterious for organismal fitness, compared to an organism with no phenotypic noise for example.

As shown in Figure II.4, even an isotropic noise slightly increases the fitness when the population is far from the fitness optimum. However, as also shown in Figure II.4, the best fitness gain is obtained when the phenotypic noise is one-dimensional. Similarly, the best phenotypic noise configuration in n dimensions consists in a dimensionality reduction to fight the cost of complexity on phenotypic noise. A population evolving such a phenotypic noise will recover the benefit of a single character scenario, phenotypic noise conferring a strong fitness advantage to organisms in directional selection (Figs. II.3 and II.4). The mathematical demonstration of this result is provided in Appendix II.5.4.

II.3.2.4 Evolvable anisotropic and correlated phenotypic noise compensates for the cost of complexity in directional selection

To test our mathematical prediction on the evolution of an anisotropic and correlated phenotypic noise (see above), we used a numerical scheme to compute the evolutionary trajectory of an initially maladapted population towards the fitness optimum. To do so, we estimated the evolution of the population distribution $n(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ through time by simulating the stochastic branching process in finite population associated to the model, as described in Methods.

In directional selection and with a complex phenotype, if organisms are allowed to evolve a correlated phenotypic noise, will they do so? And if yes, what will be the consequence on the evolution speed depending on phenotypic complexity? To address these questions, we measured the convergence time to the fitness optimum of an evolving population in four different scenarios:

(1) The mutation sizes s_σ and s_θ of the phenotypic noise parameters $\boldsymbol{\sigma}$ and $\boldsymbol{\theta}$ are lower than for s_μ ($s_\mu = 0.01, s_\sigma = s_\theta = 0.001$);

(2) The mutation sizes s_σ and s_θ are equal to the mutation size s_μ ($s_\mu = s_\sigma = s_\theta = 0.01$);

(3) The mutation sizes s_σ and s_θ are higher than s_μ ($s_\mu = 0.01, s_\sigma = s_\theta = 0.1$).

(4) Organisms have no phenotypic noise, as in canonical FGM ($s_\mu = 0.01, s_\sigma = s_\theta = 0.0$, and $\boldsymbol{\sigma} = \mathbf{0}, \boldsymbol{\theta} = \mathbf{0}$)

The simulations were computed for a phenotypic complexity ranging from $n = 1$ to $n = 10$. 100 repetitions have been computed per parameter set. The population was considered to have converged towards the optimum when the mean fitness of the population was higher than 0.9. All populations were initialized with a very low level of phenotypic noise ($\boldsymbol{\sigma} \sim \mathbf{0}$), and no rotation of the covariance matrix $\boldsymbol{\Sigma}$ ($\boldsymbol{\theta} = \mathbf{0}$). The initial euclidean distance was $d_{init} = 4.0$ for all the simulations (beyond the critical distance $d_{th} = \sqrt{n}$, see above). To do so, the μ_i values of the initial mean phenotype $\boldsymbol{\mu}$ were set to $\mu_i = d_{init}/\sqrt{n}$. To keep the mutation sizes constant whatever the phenotypic complexity, s_μ, s_σ and s_θ were also normalized by $\sqrt{n}$.

To facilitate the analysis of numerical outputs, we used three different measures:

(1) The *maximal eigenvalue* indicates the amount of phenotypic noise on the distribution $\mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ (Eq. II.4), and is equal to $\max(\boldsymbol{\sigma}^2)$;

(2) The *maximal eigenvalue contribution* indicates the contribution of the maximum eigenvalue to the total amount of noise. It is obtained by computing:

$$\frac{\max(\boldsymbol{\sigma}^2)}{\sum\limits_{i \in n} \sigma_i^2}; \tag{II.8}$$

(3) The *maximal eigenvector correlation* is the correlation (dot product) between the eigenvector associated to the maximal eigenvalue, and the direction of the optimum. If the correlation ≈ 1, the principal axis of the phenotypic distribution is aligned towards the optimum. If the correlation ≈ 0, the principal axis is orthogonal to the direction of the optimum (by symmetry, we take the absolute value of the dot product).

For each numerical simulation, both the mean and the variance of these measures have been computed along the 100 repetitions, at each time-step. As shown in Figure II.5, an evolvable phenotypic noise speeds up evolution, whatever the phenotypic complexity (Fig. II.5a). However, this gain depends on the mutation size of phenotypic noise parameters (s_σ and s_θ) relative to the mutation size of the mean trait values (s_μ). If s_σ and s_θ are lower or equal to s_μ, the fitness gain is low (Fig. II.5a). If s_σ and s_θ are higher than s_μ, the evolution speed gain is significant, with a convergence time much lower than for canonical FGM scenario (with no phenotypic noise), whatever the phenotypic complexity. The analysis of the three measures indicates that a cost of complexity exists on the evolution of the phenotypic noise, such that if $s_\sigma, s_\theta \leq s_\mu$, the phenotypic noise does not have the time to evolve towards a one-dimensional shape (with elevated noise in the direction of the fitness optimum, and no noise in all other directions). For each measure (the maximum eigenvalue, its contribution and its dot product), the maximum value reached during a simulation is plotted against the phenotypic complexity, for each of the four scenarios (Figs. II.5b,c,d). This maximum value represents the efficacy of evolution in shaping the phenotypic noise in directional selection, knowing that noise increases when the population is far from the fitness optimum, and then decreases when the population reaches the fitness optimum. A trade-off seems to exist between the convergence time of the mean phenotype $\boldsymbol{\mu}$ and the time needed for evolution to shape the phenotypic noise. On Figure II.5b, we see that the maximum eigenvalue does not reach the optimal value ≈ 9.0 when $s_\sigma, s_\theta \leq s_\mu$. However, when $s_\sigma, s_\theta > s_\mu$, the maximum eigenvalue reaches the optimal value whatever the phenotypic complexity. Indeed, in this situation organisms have time to evolve the most beneficial, flattened form of phenotypic noise. On Figures II.5c and II.5d, the maximum eigenvalue contribution and the maximum eigenvalue dot product are strongly lessened with phenotypic complexity when $s_\sigma, s_\theta \leq s_\mu$, while they are almost equal to 1 whatever the phenotypic complexity when $s_\sigma, s_\theta > s_\mu$. In the latter case, the phenotypic noise evolves towards a near perfect flattened form, as predicted previously, hence strongly speeding up adaptation.

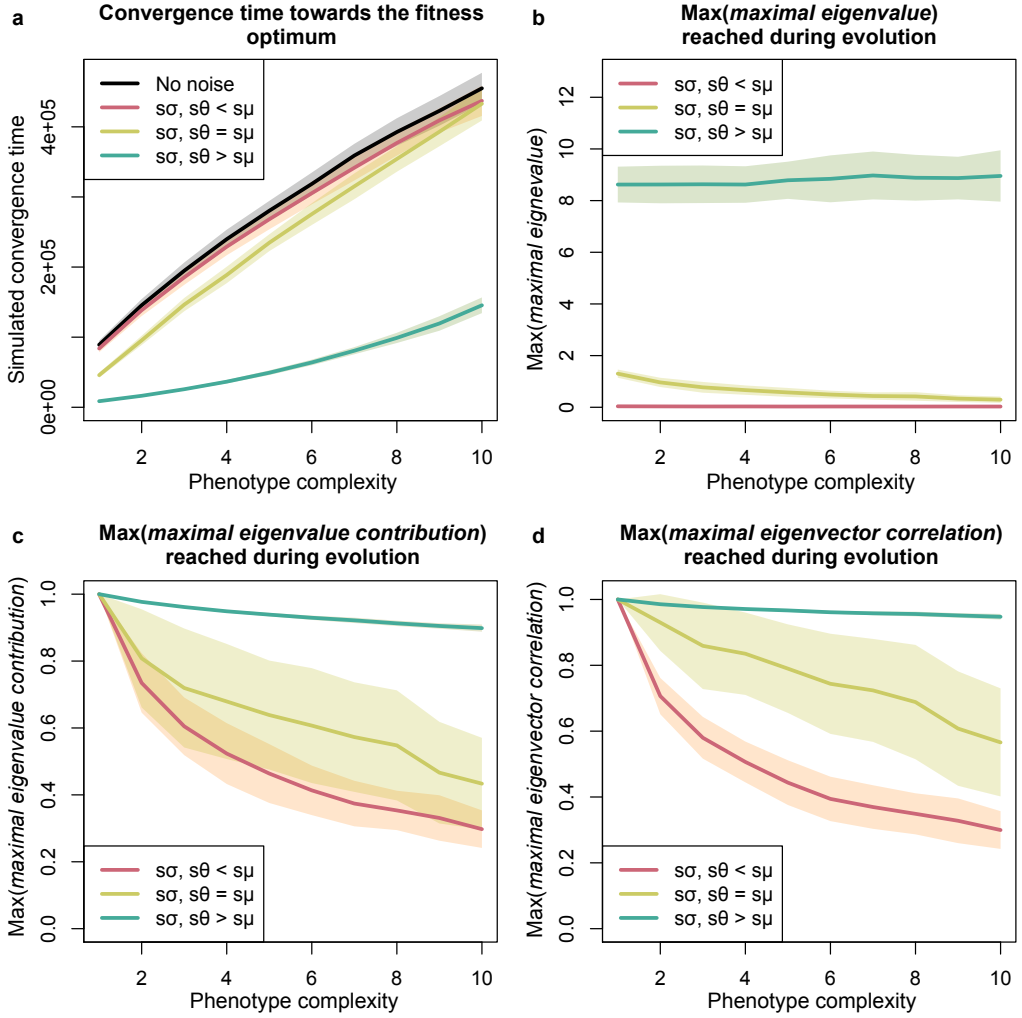


Figure II.5 – An evolvable anisotropic and correlated phenotypic noise speeds up evolution. a, Convergence time towards the fitness optimum. Four scenarios are evaluated. Black curve: canonical FGM scenario, with no phenotypic noise. Colored curves: evolvable phenotypic noise scenarios. x-axis: phenotypic complexity, from $n = 1$ to $n = 10$. y-axis: Convergence time. **b, c, d,** Maximum values of **b,** the *maximum eigenvalue* ($max(\boldsymbol{\sigma})$), **c,** the *maximum eigenvalue contribution* (Eq. II.8), and **d,** the *maximum eigenvalue dot product* (with the direction of the fitness optimum) reached by the population during its evolution towards the fitness optimum, respectively. Four scenarios are evaluated (legends): three with evolvable phenotypic noise, one with no phenotypic noise (canonical FGM scenario). For each scenario, the mean (colored lines) and the standard deviation along the 100 repetitions (shaded areas) are represented. In all cases, the scenario where $s_\sigma, s_\theta > s_\mu$ is characterized by the evolution of a near perfectly flattened and one-dimensional phenotypic noise, fully aligned with the fitness optimum, as predicted mathematically.

II.4 Discussion

The fitness benefit of elevated phenotypic noise in directional selection was foreseen by Eldar and Elowitz (2010), who stated that "*one might expect increased phenotypic noise during periods of adaptation to new environments, followed by reduction in noise when*

selection becomes stabilizing". Based on a mathematical model, Zhang et al. (2009) also stated that elevated noise would increase the probability to fix beneficial mutations in directional selection, provided that the local fitness landscape is convex. This prediction has been corroborated in a recent experimental study on the evolution of a single gene in *Saccharomyces cerevisiae* (Bódi et al., 2017). However, these results are based on the evolution of a single phenotypic character, while real phenotypes are much more complex, with multiple characters under selection. We extended Fisher's geometric model (Fisher, 1930) to account for evolvable phenotypic noise, in order to address the question of how phenotypic noise would evolve in directional selection, when organisms own complex phenotypes. This model, named σFGM, allows for evolvable phenotypic noise amplitudes on each character, but also for evolvable noise correlations between characters, as justified by our analysis of the phenomic data of Yvert et al. (2013), and an experimental study by Cressler et al. (2017).

First, we studied analytically the case where organisms own a single phenotypic character, and must evolve towards a novel environment. Doing so, we confirmed previous results (Zhang et al., 2009; Eldar and Elowitz, 2010; Bódi et al., 2017): elevated phenotypic noise is beneficial under directional selection, when the population is far from the fitness optimum and experiences a convex fitness landscape. When the population is near the fitness optimum, the phenotypic noise is deleterious and must be minimized, confirming previous statements that phenotypic noise is deleterious under stabilizing selection (Mineta et al., 2015; Keren et al., 2016).

In 1930, Fisher (1930) hypothesized that organisms evolving towards a fitness optimum experience a cost of complexity, beneficial mutations becoming increasingly harder to fix when the number of phenotypic characters under selection increases. Here, we demonstrated that this cost of complexity also hinders the benefit of an elevated phenotypic noise in directional selection, when noise is isotropic. In this case, when the number of phenotypic characters is higher than one, the beneficial fitness effect of phenotypic noise quickly vanishes. Moreover, the critical distance from the fitness optimum below which the phenotypic noise must be minimized increases as the square root of the number of characters, suggesting that for a constant distance from the fitness optimum, more complex organisms are expected to be less noisy.

Recent studies suggested that phenotypic noise could be considered as a complex phenotypic character, possibly tuned by the genotype-to-phenotype map (Yvert et al., 2013; Boukhibar and Barkoulas, 2016), and correlated (Cressler et al., 2017). Here, we demonstrated that under directional selection on a convex fitness landscape, the best possible configuration for the phenotypic noise is to evolve towards a flattened, one-dimensional configuration, with elevated noise in the direction of the fitness optimum, and no noise in all other directions. In this case, the evolving population recovers the beneficial value of an elevated phenotypic noise, as for the single phenotypic character scenario. We also demonstrated that in this specific configuration, phenotypic noise increases the probability to fix beneficial mutations and accelerates evolution, whatever the phenotypic complexity, thereby partly compensating for the cost of complexity. To be fully exploitable by evolution, the properties of the phenotypic noise must evolve at a higher speed than the

mean phenotype. However, it is not required for the noisy phenotype to be inherited in our model, suggesting that when it is the case (Charlebois et al., 2011), this constraint could be relaxed.

Our findings are in accordance with recent experimental results. First, Cressler et al. (2017) demonstrated the existence of correlated phenotypic noise on *Daphnia pulicaria* (a freshwater zooplankton). By measuring three integrated phenotypic characters at the individual level (body growth, number of eggs and longevity) on different populations of genetic variants, they showed that there are no significant genetic correlations between characters, while there is strong evidence for positive non-genetic correlations between characters. Moreover, they showed that increasing phenotypic noise enhances growth rate when non-genetic correlations between characters are positive, in agreement with our prediction on the evolution of phenotypic noise. Second, our analysis of single-cell yeast data provided by Yvert et al. (2013) revealed that the phenotypic characters showing the strongest noise correlations are also the most variables between strains, suggesting that a correlated phenotypic noise evolved on the phenotypic characters most exposed to directional selection (Appendix II.5.2). Finally, an experimental study by Metzger et al. (2015) suggested that the expression noise of TDH3 gene may evolve faster than its mean expression, suggesting that it could be the case for phenotypic noise in general, in agreement with our findings.

As a whole, our results show that such non-isotropic phenotypic noise could be exploited by evolution, and suggest further experiments to assess the functional nature of phenotypic noise. In particular, phenotypic noise has been demonstrated to have a role in drug resistance (Singh et al., 2010; Charlebois et al., 2014; Charlebois, 2015), cancer cells proliferation (Gascoigne and Taylor, 2008; Cohen et al., 2008; Huang, 2012; Pisco et al., 2013) as well as in the process of decision-making, seen as an adaptation to a environmental change (Richard et al., 2016). It could be interesting to initiate new experiments letting biological populations adapt to a novel environment, and acquire phenotypic noise data at the individual level (Ohya et al., 2015). The long-term evolution experiment (LTEE, Elena and Lenski 2003), where populations of *Escherichia coli* are evolved in a minimum glucose medium since more than 66,000 generations, and regularly frozen to keep track of evolution, would be a good candidate to initiate such an experiment. Moreover, our predictions on the evolution of phenotypic noise in directional selection could be used to predict the future direction of evolution, and to localize the fitness optimum in the phenotypic space. Indeed, tracking the evolution of phenotypic noise experimentally could help biologists understand what are the selective pressures at work on organisms, and to anticipate the next evolution steps.

By extending Fisher's geometric model with evolvable phenotypic noise, we offered general predictions on what would be the evolution of phenotypic noise in directional selection, and its consequences on the fate of asexual populations experiencing directional selection. Our demonstrations rely on the assumption that the phenotypic noise and the fitness landscape are Gaussian-shaped, as it is historically the case in Fisher's geometric model. If the phenotypic noise is not Gaussian, other particular cases could appear, even if locally there is always a benefit to be noisy in directions where the fitness is convex. By relaxing

our hypotheses, other interesting questions could be tackled, for example the case where the fitness landscape is multimodal or not static, the case of a degenerated noise (*e.g.*, where some directions in the phenotypic space are forbidden), or whether the case of a multiplicative noise. By deciphering the conditions in which phenotypic noise evolves towards specific patterns, our results may contribute to the growing field of predictive biology.

II.5 Supporting Information

II.5.1 Figure S1. An example of the temporal dynamics in σFGM.

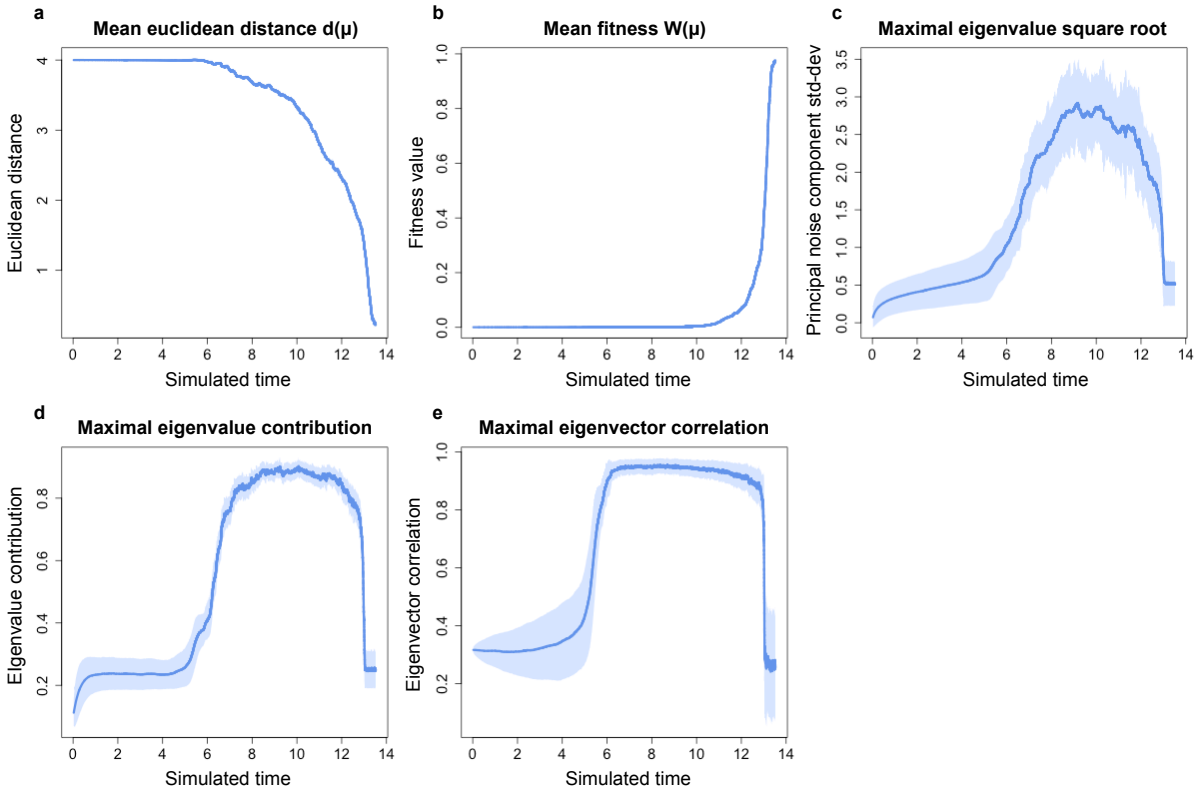


Figure II.6 – An example of the temporal dynamics in σFGM. A simulation in $n = 10$ dimensions is initialized with 10,000 organisms having the same triplet $(\boldsymbol{\mu}_0, \boldsymbol{\sigma}_0, \boldsymbol{\theta}_0)$, with $\boldsymbol{\mu}_0 = \{4.0/\sqrt{n}\}^T$, $\boldsymbol{\sigma}_0 = \mathbf{0}$, and $\boldsymbol{\theta}_0 = \mathbf{0}$. The initial population is then localized on the hyper-sphere of radius 4.0, none of the phenotypic characters being aligned with the fitness optimum z_{opt}, and thus requiring to be adapted. $s_\mu = 0.01$, and $s_\sigma = s_\theta = 0.1$. The simulation stopped when the population mean of the mean fitness $\langle W(\boldsymbol{\mu})\rangle$ reached 0.9. **a,** The population mean of the mean euclidean distance $\langle d(\boldsymbol{\mu})\rangle$. **b,** The population mean of the mean fitness $\langle W(\boldsymbol{\mu})\rangle$. **c,** The population mean of the maximal eigenvalue. **d,** The population mean of the maximal eigenvalue contribution. **e,** The population mean of the maximal eigenvector correlation with the direction of the fitness optimum. The standard deviation of each variable is represented by a shaded blue area.

II.5.2 Appendix S1. Various wild-types of yeast exhibit correlated phenotypic noise.

In this appendix, we present the results of our analysis on the experimental single-cell data provided by Yvert et al. (2013). We assume that the reader is aware of the basic definitions and equations provided in the main manuscript.

Yvert et al. (2013) used automated image analysis to describe yeast phenotypic diversity at a single-cell resolution (known as phenomics, Ohya et al. 2015). They monitored $n = 125$ phenotypic characters on isogenic populations of $m = 37$ different strains of yeast, living in natural or laboratory conditions. For each strain, they measured 5 replicates of approximately 200 cells each ($\sim$1000 cells per strain). They demonstrated that phenotypic noise significantly differs between strains, supporting "*the possibility that, if noise is adaptive, microevolution may tune it in the wild*" (Yvert et al., 2013).

We used the raw datasets published by the authors to measure intra-strain (*i.e.* isogenic) noise correlations between characters. The goal of our study is to test the existence of correlated phenotypic noise in natural strains of yeast. The datasets provided by Yvert et al. (2013) are structured as following: for each strain of yeast, a set of files is provided in a dedicated folder (a dataset per replicate). For each replicate, the list of single-cell measures is dispatched in three different files in `xls` format. Each cell is identified by a unique tag per image and the tag of the captured image. Each time a measure failed (on one character, or on the entire cell), the corresponding element (or line) in the table was filled with value -1. Several characters are redundant (for example the volume and the size of the nucleus), and strongly correlated. Moreover, each trait value is provided with specific units (*e.g.*, number of pixels, volume or angle units), such that some normalization is necessary. The code associated to this analysis is freely available in Script II.5.8. One can run again the whole analysis by following instructions provided in the README file. The raw dataset is freely provided by Yvert et al. (2013).

The purpose of the analysis is to determine whether intra-strain variability presents correlations between characters *once inter-strain correlations between characters have been removed*.

Let us first process inter-strain variability. The idea is to find a phenotypic space in which there is as little character-specific variability and correlation as possible. Here we have 37 isogenic strains, hence 37 genotypes. We define the "phenotype" of a strain/genotype as the vector of mean trait values, computed over all cells from this strain/genotype. We then define the "centered phenotype" of a strain/genotype by removing the grand mean of each character. The singular value decomposition of the 37x125 matrix of centered strain phenotypes will give us a set of orthonormal linear combinations of characters. By construction, when the centered strain phenotypes are expressed according to these new characters, they lose all their pairwise correlations, implying that the variance-covariance matrix is diagonal for those new characters. Moreover, we normalize the variance of each

new strain phenotype to one[1], such that the 37 new strain phenotypes are isotropically distributed.

This new base is the closest analogy we could think of to the phenotypic space in the classical version of Fisher's geometric model. Fisher's phenotypic space is orthogonal and normalized, and mutations on the genotype cause phenotypic traits to vary independently and with the same amplitude, according to an isotropic mutational distribution.

The second step is to project intra-strain single-cell data in Fisher's space, and to analyze the possible remaining correlations of intra-strain phenotypic variability in this space.

Figure II.7 shows the detailed steps of our analysis, as described below. First, we converted each `xls` file into `csv` format, and we merged the three files of each replicate to obtain a single dataset $\boldsymbol{M_{0,s,r}}$ ($s \in \{1, ..., 37\}$, $r \in \{1, ..., 5\}$) per replicate, and we removed useless information (such as cell identifiers, coordinates on the image, and so on) (Fig. II.7.1). Then we merged the 5 replicates of each strain (Fig. II.7.2) to compute the matrix $\boldsymbol{M_0}$ of the mean phenotypic characters per strain (Fig. II.7.3). Each column $\boldsymbol{M_{0}}_j$ of $\boldsymbol{M_0}$ was centered and normalized to obtain the matrix $\boldsymbol{M}$ (Fig. II.7.4):

$$\boldsymbol{M}_j = \frac{\boldsymbol{M_{0}}_j - \text{mean}(\boldsymbol{M_{0}}_j)}{\text{stdev}(\boldsymbol{M_{0}}_j)}. \tag{II.9}$$

We also standardized each replicate to obtain 37×5 matrices $M_{s,r}, s \in \{1, ..., 37\}, r \in \{1, ..., 5\}$ (Fig. II.7.6). For each column $\boldsymbol{M_{0,s,r}}_j$ of $\boldsymbol{M_{0,s,r}}$:

$$\boldsymbol{M_{s,r}}_j = \frac{\boldsymbol{M_{0,s,r}}_j - \text{mean}(\boldsymbol{M_{0,s,r}}_j)}{\text{stdev}(\boldsymbol{M_{0,s,r}}_j)}. \tag{II.10}$$

To find Fisher's space, we computed a SVD from $\boldsymbol{M}$ (see details below, and Fig. II.7.5). For each standardized replicate dataset $\boldsymbol{M_{s,r}}$, many trait values are missing, and are replaced by -1 values, making impossible some mathematical operations. For this reason, a next step was to estimate the missing values: we used a simple conservative method, as described below (Fig. II.7.7). Finally, each replicate dataset was projected in Fisher's space (Fig. II.7.8). The inter-replicate variability was evaluated to ensure that experimental variability is low enough (Fig. II.7.9), and intra-strain phenotypic noise correlations were analyzed (Fig. II.7.10).

We describe below the steps requiring details.

Estimation of missing values

To estimate missing values, we first computed the cell-to-cell Pearson correlation matrix $\boldsymbol{C}$ associated to each replicate, based on available data. Knowing there are $m' \sim 200$

[1] Which is why our analysis is not exactly a PCA: We drop the singular values that are usually left in the PCA.

cells and $n = 125$ phenotypic characters in each replicate, we defined a cell by a vector $\boldsymbol{X_i} \in \mathbb{R}^{125}$, with $i \in \{1, ..., m'\}$.

Each missing value $x_{i,j}$ of $\boldsymbol{X_i}$ (with $j \in \{1, ..., n\}$) was recovered by computing:

$$x_{i,j} = \bar{\boldsymbol{X}}_{\boldsymbol{i}} + \frac{\sum_{k=1}^{m'} (x_{k,j} - \bar{\boldsymbol{X}}_{\boldsymbol{k}}) c_{i,k}}{\sum_{k=1}^{m'} |c_{i,k}|} \tag{II.11}$$

with $c_{i,k}$ element of $\boldsymbol{C}$. This simple method is conservative, meaning that noise amplitudes tend to be reduced through this estimation method. Moreover, we removed all cells X_i that contained only -1 values.

Singular value decomposition

Let us consider the matrix $\boldsymbol{M}$ of dimension $m \times n$ that contains the standardized mean phenotypic trait values of each strain, where m is the number of strains ($m = 37$), and n is the number of characters ($n = 125$). $\boldsymbol{M}$ can be decomposed into a $m \times m$ unitary matrix $\boldsymbol{U}$, a $m \times n$ positive and diagonal matrix $\boldsymbol{\Sigma}$, and a $n \times n$ unitary matrix $\boldsymbol{V}$ such that:

$$\boldsymbol{M} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^* \tag{II.12}$$

with $\boldsymbol{V}^*$ being the conjugate transpose of $\boldsymbol{V}$.

The diagonal entries of $\boldsymbol{\Sigma}$ are the singular values $\boldsymbol{\sigma} \in \mathbb{R}^m$ of $\boldsymbol{M}$. $\boldsymbol{V}$ contains the right-singular vectors of $\boldsymbol{M}$, defining the base where m linear combinations of phenotypic characters are separated and orthonormal. These linear combinations give 37 new phenotypic characters, whose means vary independently and have been normalized to the same amplitude. In the following, we will call the space generated by the base $\boldsymbol{V}$ the "Fisher's space".

As shown on Figure II.8, looking at the vector $\boldsymbol{\sigma}$ reveals that only the first 8 singular values carry significant variability between mean phenotypic characters (a method to keep significant singular values consists in removing all values after the shoulder point in Fig. II.8). We thus truncated matrices $\boldsymbol{V}$ and $\boldsymbol{\Sigma}$ to only keep the most significant singular values and singular vectors. To this aim, we defined the matrix $\boldsymbol{V_{cut}}$ of size $n \times 8$ containing the 8 first singular vectors of $\boldsymbol{V}$, and the matrix $\boldsymbol{\Sigma_{cut}}$ of size 8×8 containing the first 8 singular values $\boldsymbol{\sigma_{cut}}$ such that $\boldsymbol{\Sigma_{cut}} = \mathrm{diag}(\boldsymbol{\sigma_{cut}})$.

Intra-strain data projection in Fisher's space

Since the singular value decomposition has been computed on $\boldsymbol{M}$, and the most significant singular values and vectors have been isolated, we can use the base $\boldsymbol{V_{cut}}$ and the diagonal matrix $\boldsymbol{\Sigma_{cut}}$ to project each replicate data in Fisher's space. Let us define the $m' \times n$ matrix $\boldsymbol{M_{s,r}}$ which contains the single-cell data of the replicate $r \in \{1, ..., 5\}$ of the strain $s \in \{1, ..., 37\}$. m' is the number of cells ($m' \sim 200$), and n is the number of characters ($n = 125$). The projection of $\boldsymbol{M_{s,r}}$ in Fisher's space is computed as following:

$$\boldsymbol{M'_{s,r}} = \boldsymbol{M_{s,r}} \boldsymbol{V_{cut}} \boldsymbol{\Sigma_{cut}^{-1}} \tag{II.13}$$

with $\boldsymbol{\Sigma_{cut}^{-1}} = \mathrm{diag}(1/\boldsymbol{\sigma_{cut}})$ a diagonal matrix where the diagonal entries are the reciprocal of the first 8 singular values. $\boldsymbol{M'_{s,r}}$ represents the single-cell data of the replicate r of the strain s, projected in Fisher's space.

Results

Inter-replicate variability does not impair phenotypic noise analysis.

A first step in our analysis of intra-strain phenotypic noise is to check the absence of significant experimental variability between replicates. To this aim, we compared the structure of each replicate. For each replicate dataset $\boldsymbol{M_{s,r}}$ (with $r \in \{1, ..., 5\}$ and $s \in \{1, ..., 37\}$), we computed the vectors $\boldsymbol{\mu_{s,r}}$ and $\boldsymbol{\sigma_{s,r}}$ containing respectively the means and the standard deviations by character of $\boldsymbol{M_{s,r}}$. As shown in Figures II.9 and II.10, replicates do not vary significantly from each other (each plot represents a strain, with one color per replicate). We also computed and plotted the correlation matrix of each replicate (5 matrices per strain) to check that experimental variability does not affect noise correlation structure. Noise correlations appeared to not strongly vary between replicates of each strain. The 37 figures corresponding to the 5 correlation matrices of each strain are provided in Data II.5.5.

As a conclusion, we didn't notice impairing experimental variability between replicates. For this reason, we decided to merge replicates in a single dataset to facilitate further analyses.

Phenotypic noise correlation matrices for each strain.

As described previously, experimental variability between replicates is low enough to allow us to merge replicates in a single dataset $\boldsymbol{M'_i}$, with $i \in \{1, ..., 37\}$. First, in order to identify

possible phenotypic noise correlations in Fisher's space, we computed the correlation matrix of $\boldsymbol{M_i'}$ $\forall$ i, and performed a Pearson correlation test on each off-diagonal pair of variables, with $\alpha = 0.05$. A Bonferroni correction of $k = 28$ ($k = 8 * 7/2$) was also applied on each test. Then, we focused on the phenotypic characters exhibiting elevated noise correlations, as shown below.

Correlation matrices demonstrated that all the natural strains of yeast studied in Yvert et al. (2013) exhibit correlated phenotypic noise in Fisher's space (defined before as the space where inter-strain mean phenotypic characters are uncorrelated and of the same amplitude). For each strain, we found significant noise correlations, despite the Bonferroni correction ($k = 28$). For each correlation matrix, we generated a figure showing the correlations and the results of the Pearson correlation test. For each pair of characters, the strength of the correlation is symbolized by the size of the corresponding circle. A blue color indicates a positive correlation, and a red color a negative correlation. When the Pearson correlation test is negative, the corresponding circle is marked with a cross. The 37 figures corresponding to the correlation matrix of each strain are provided in II.5.6.

Phenotypic characters with the highest noise correlation are also the most variable between strains.

For each strain, we also identified the two axes of the phenotypic space showing the highest phenotypic noise correlation. As shown in Figure II.11, in a majority of strains, these two axes correspond to the first two axes of Fisher's space. These axes correspond to the most variable inter-strain mean phenotypic characters. On Figure II.12a, we show what would be an uncorrelated phenotypic noise for each strain for the two first principal components (PC1 and PC2) of the Fisher's space (the shape of the phenotypic noise of each strain is symbolized by an ellipse representing the standard deviation of the associated bivariate normal law, rescaled by a factor 0.002). On Figure II.12b, the real observed phenotypic noise is represented, showing noise correlations for all the strains.

One must remember that PC1 and PC2 axes are a combination of phenotypic characters. The most variable combinations of phenotypic characters between strains are also the ones exhibiting the most correlated intra-strain phenotypic noise. Thus, if one assume that phenotypic differences across strains are adaptive, this result suggests that the phenotypic characters most exposed to directional selection are also the ones with the most correlated phenotypic noise between characters.

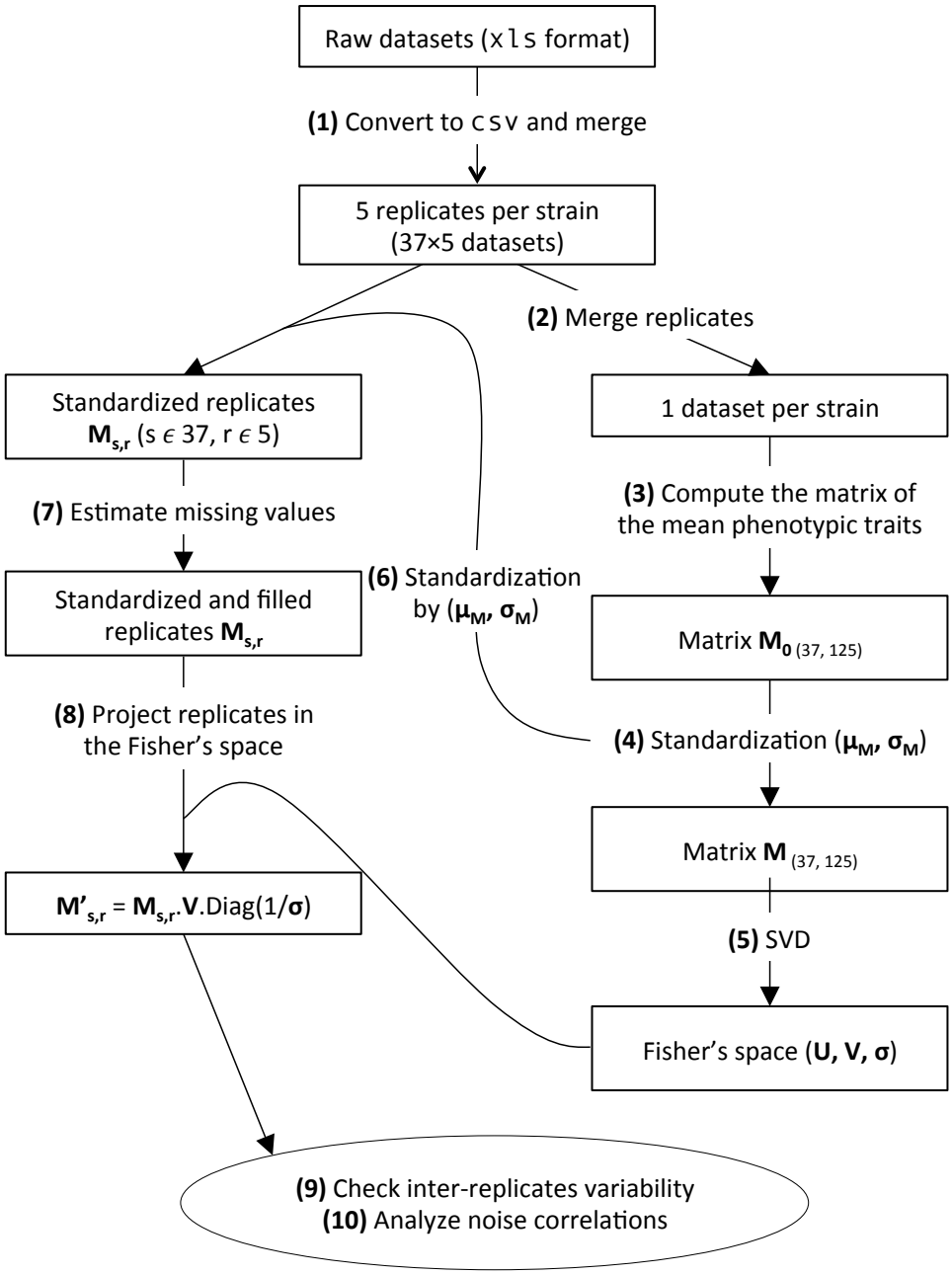


Figure II.7 – Step-by-step protocol used to analyze single-cell data. **(1)** Each `xls` file is converted into `csv` format, the three files related to each replicate being merged to obtain a single dataset $\boldsymbol{M_{0,s,r}}$ ($s \in \{1, ..., m = 37\}$, $r \in \{1, ..., 5\}$) per replicate. **(2)** The 5 replicates of each strain are merged to obtain a single dataset per strain. **(3)** The matrix $\boldsymbol{M_0}$ of the mean phenotypic trait values per strain is computed. **(4,6)** Datasets are standardized according to the mean vector $\boldsymbol{\mu_M} \in \mathbb{R}^{125}$ and the standard deviation vector $\boldsymbol{\sigma_M} \in \mathbb{R}^{125}$ of $\boldsymbol{M_0}$. **(5)** A singular values decomposition (SVD) is computed from standardized inter-strain dataset $\boldsymbol{M}$ (see above for the details of the SVD). **(7)** Replicate missing values are estimated (see above). **(8)** Each replicate dataset is projected into Fisher's space. **(9)** Inter-replicate variability is evaluated to ensure that experimental variability is low enough. **(10)** Intra-strain phenotypic noise correlations are analyzed.

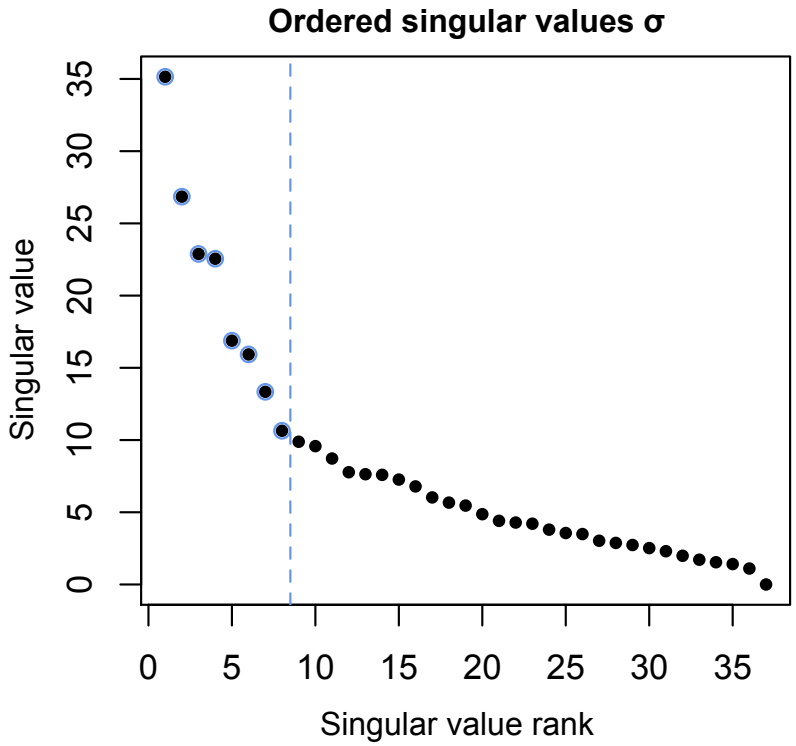


Figure II.8 – Ordered singular values contained in σ. A simple empirical method to keep only significant variations is to isolate all the singular values located before the shoulder point in the ordered plot (as shown by a blue dashed line). Here, we kept the first 8 singular values.

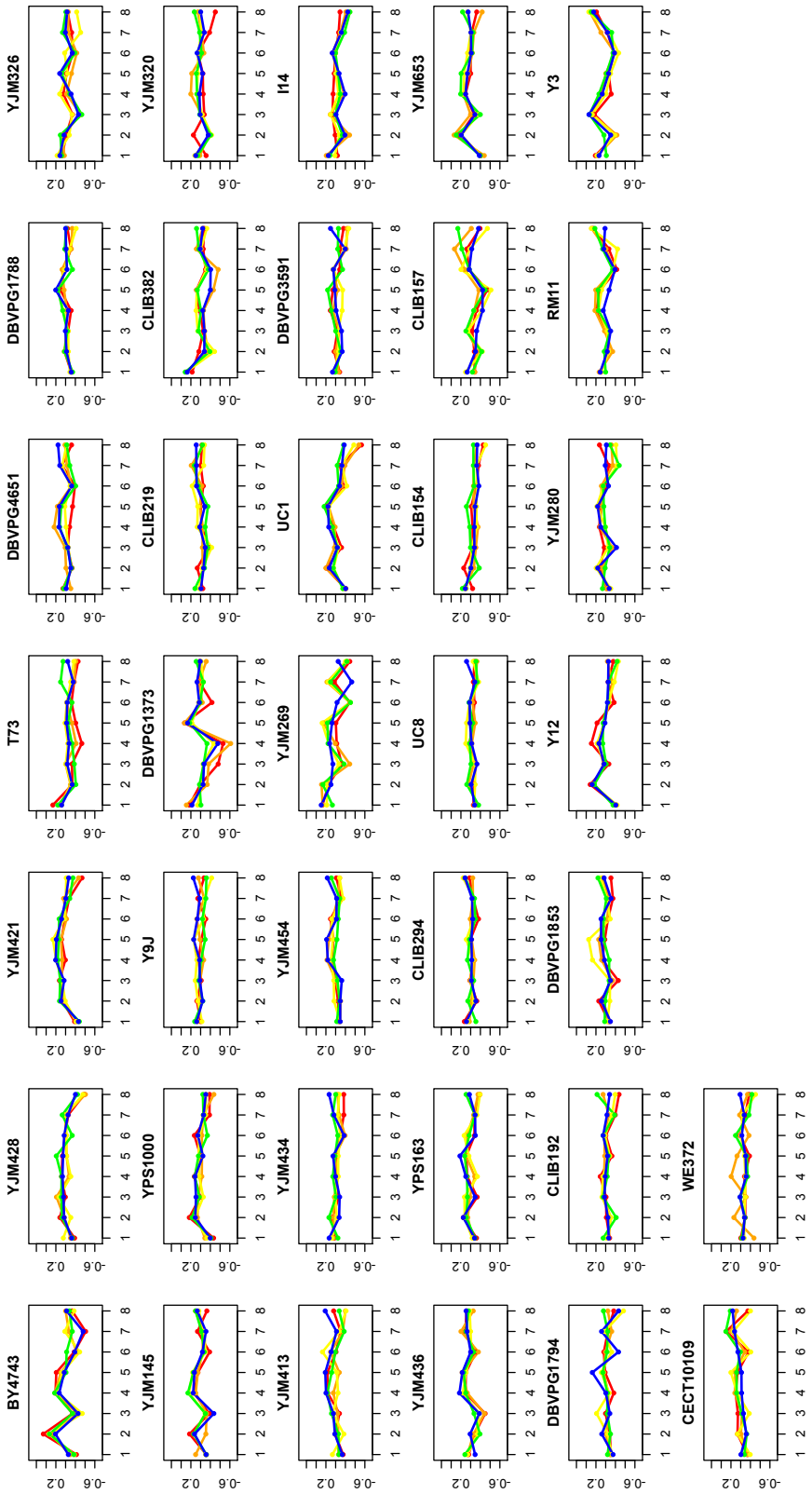


Figure II.9 – Mean phenotypic trait values per replicate per strain. For each replicate of each strain, the mean phenotypic trait values are plotted (one color per replicate on each plot, one plot per strain).

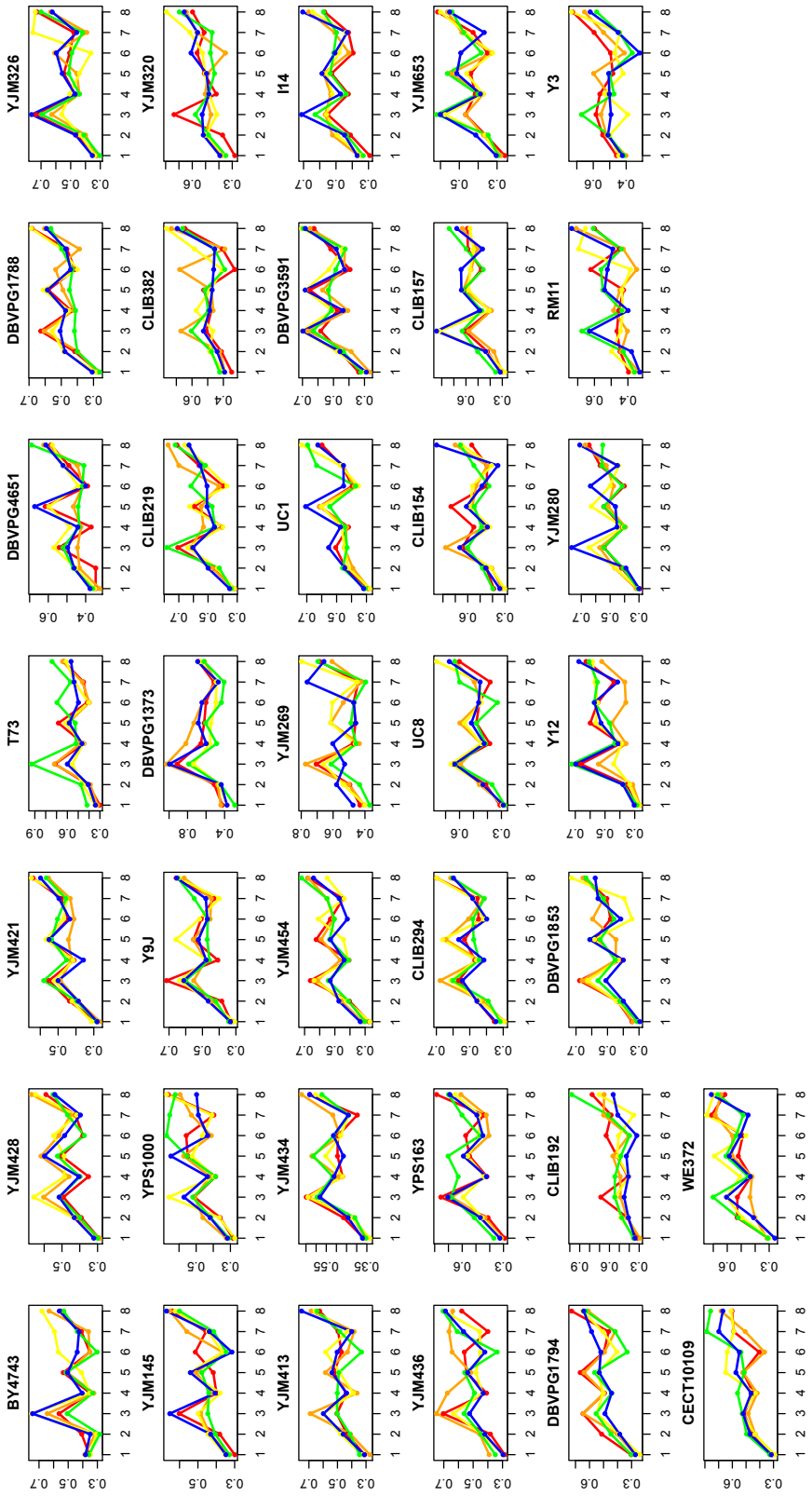


Figure II.10 – Standard deviation of each phenotypic character per replicate per strain. For each replicate of each strain, the standard deviation of each phenotypic character is plotted (one color per replicate on each plot, one plot per strain).

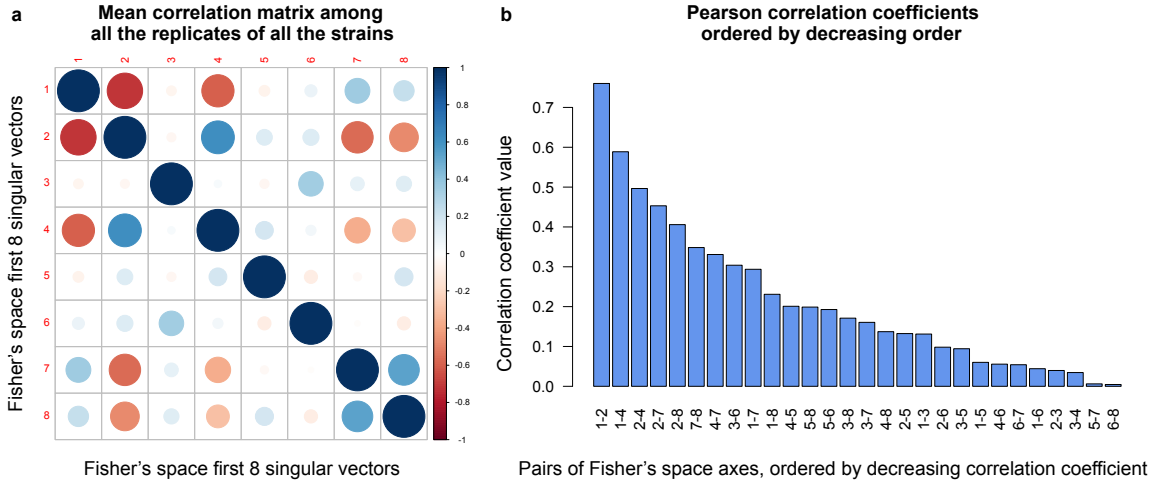


Figure II.11 – Mean phenotypic noise correlations in the Fisher's space. **a,** The mean correlation matrix across all the replicates of all the strains has been computed and plotted here. For each pair of characters, the strength of the correlation is symbolized by the size of the corresponding circle. A blue color indicates a positive correlation, and a red color a negative correlation. **b,** All off-diagonal pairwise correlations between the first 8 axes of Fisher's space are sorted by decreasing order. The most correlated axes in mean are axes 1 and 2 (called PC1 and PC2).

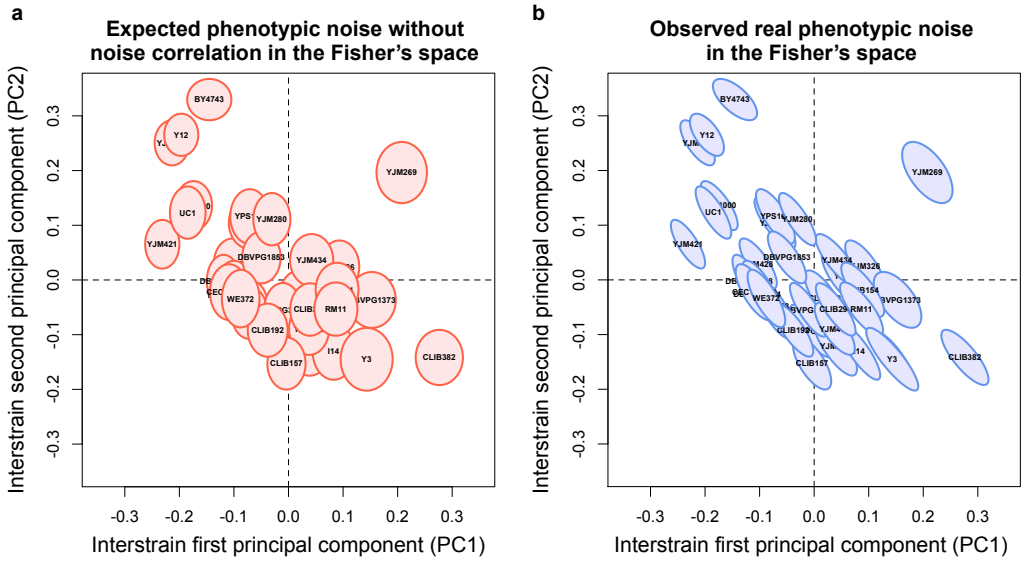


Figure II.12 – Yeast intra-strain phenotypic noise is correlated in the Fisher's space. A singular value decomposition (SVD) was performed on the mean trait values of each of the 37 yeast strains. This space is similar to the phenotypic space used in Fisher's geometric model, where phenotypic characters mutate independently and with the same amplitude (*i.e.* mean phenotype mutations are isotropic in this space). For this reason, we called this space "Fisher's space". We then projected single-cell data of each strain in this space. We identified the two axes showing the highest noise correlation in mean, for all strains: they correspond to the two first components of Fisher's space (PC1 and PC2). **a,** Expected phenotypic noise for each strain without noise correlation between Fisher's space axes (each axis representing a linear combination of phenotypic characters). The shape of the phenotypic noise of each strain is symbolized by an ellipse representing the standard deviation of the associated bivariate normal law. Each ellipse is tagged with the corresponding strain name. The size of the ellipses are rescaled by a factor 0.002 to better distinguish them. The coordinates of the center of each ellipse correspond to the real position of the corresponding strain in Fisher's space (from real data). **b,** Real observed phenotypic noise is represented, showing noise correlation between PC1 and PC2 axes, for all the strains.

II.5.3 Appendix S2. A numerical solver for σFGM.

In this appendix, we present in more details the numerical solver of σFGM. We assume that the reader is aware of the basic definitions and equations provided in the main manuscript.

To estimate the evolution of the population distribution $n(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ through time, we simulated the stochastic branching process associated to σFGM equations (as discussed in Methods). Once initial conditions are defined (Table II.2), the evolutionary trajectory of $n(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ is simulated through time using Algorithm 1, which is similar to a time-adaptive tau-leaping algorithm (Gillespie, 2007).

Parameters of the numerical solver

Table II.2 – List of parameters of the numerical solver for σFGM.

Variable	Symbol	Domain
Number of particles	N	$[1, +\infty]$
Number of phenotypic characters (or dimensions)	n	$[1, +\infty]$
Initial mean phenotype vector	$\boldsymbol{\mu_0}$	$\mathbb{R}^n$
Initial $\boldsymbol{\Sigma}$ eigenvalues vector	$\boldsymbol{\sigma_0}$	$\mathbb{R}^n$
Initial $\boldsymbol{\Sigma}$ rotation angles vector	$\boldsymbol{\theta_0}$	$\mathbb{R}^{n(n-1)/2}$
$\boldsymbol{\mu}$ values mutation size standard deviation	s_μ	≥ 0
$\boldsymbol{\sigma}$ values mutation size standard deviation	s_σ	≥ 0
$\boldsymbol{\theta}$ values mutation size standard deviation	s_θ	≥ 0

These parameters must be set to initialize a stochastic branching process simulation.

Code availability

The code of the numerical solver and parameter exploration scripts is freely available in Script II.5.7, and is distributed under the open source GNU General Public License.

Main algorithm of the numerical solver

Data: Set initial conditions (Table II.2); Set N particles with the same initial parameters $\boldsymbol{\mu_0}$, $\boldsymbol{\sigma_0}$ and $\boldsymbol{\theta_0}$.
Result: Evolution through time of the population distribution $n(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$.

$t = 0$;
$N_t = N$;
while *Stop criteria not reached* **do**
  $W_{max} = \max(W_i)$, for $i \in [0, N]$;
  $dt = 0.1/W_{max}$;
  for $i = 1$ ... N **do**
    if *uniform_ draw(0,1)* $< W_i \times dt$ **then**
      i' = Duplicate(i);
      Mutate(i');
      $\boldsymbol{z_i}$ = multivariate_normal_draw($\boldsymbol{\mu_i}, \boldsymbol{\Sigma_i}$);
      $W_i = W(\boldsymbol{z_i})$;
      $N_t = N_t + 1$;
    end
  end
  $p_{death} = \max(0, (N_t - N)/N_t)$;
  for $i = 1$ ... N **do**
    if *uniform_ draw(0,1)* $< p_{death}$ **then**
      Kill(i);
      $N_t = N_t - 1$;
    end
  end
  $t = t + dt$;
  Compute_moments();
  Compute_statistics();
end

Algorithm 1: Main algorithm of the numerical solver of σFGM. This algorithm simulates the stochastic branching process associated to the equations of σFGM. In this algorithm, similar to a tau-leaping algorithm, the timestep dt is not fixed and depends on the best organism's fitness W_{max} at time t. This method is used to avoid long periods with no branching events (usually when population fitness is very low). Thus, the time scale is rescaled to set the proliferation rate of the best particle at 0.1: at each simulation time-step, $dt = 0.1/W_{max}$. The population size N_t is also regulated by recomputing the death probability p_{death} at each time-step such that $p_{death} = \max(0, (N_t - N)/N_t)$. Finally, at each time-step, the two first moments of $n(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ are computed to extract the evolutionary trajectory, as well as the maximal eigenvalue, the maximal eigenvalue contribution and the maximal eigenvector correlation.

Parametric exploration for a single phenotypic character.

We performed a parametric exploration in the space (μ, σ) at a high resolution. More precisely, we computed the integrals in $\partial\overline{W}(\mu,\sigma)/\partial\sigma$ and $\partial\overline{W}(\mu,\sigma)/\partial\mu$ using the numerical method of Gauss-Kronrod adaptive integration on infinite intervals (QAGI) provided by the Gnu Scientific Library. We explored μ and σ between 0 and 10, with a step of 0.01, thus representing the computation of 10^6 points. We only explored $\mu \geq 0$ since the model is symmetric for $\mu < 0$ and $\mu > 0$.

We used the data to numerically find the ridge $\partial\overline{W}(\mu,\sigma)/\partial\sigma = 0$, and the $\overline{W}(\mu,\sigma)$ gradient in the space (μ, σ).

Parametric exploration for an isotropic noise in n dimensions

We also performed a parametric exploration in the space $(\boldsymbol{\mu}, \sigma)$ at a high resolution. We computed the integral $\partial\overline{W}(\boldsymbol{\mu},\sigma)/\partial\sigma$ using the numerical method of Gauss-Kronrod adaptive integration on infinite intervals (QAGI) provided by the Gnu Scientific Library.

We explored μ_1 (all other $\mu_i, i \in \{2, \ldots, n\}$ being equal to 0) and σ between 0 and 10, with a step of 0.05, from $n = 1$ to $n = 50$, thus representing the computation of 2.10^6 points. We only explored $\mu_1 \geq 0$ since the model is symmetric for $\mu_1 < 0$ and $\mu_1 > 0$.

We used the data to find numerically the ridge $\partial\overline{W}(\boldsymbol{\mu},\sigma)/\partial\sigma = 0$ for each dimension, and the $\overline{W}(\boldsymbol{\mu},\sigma)$ gradient in the space $(\boldsymbol{\mu}, \sigma)$.

II.5.4 Appendix S3. Analytical study of σFGM.

In this appendix, we present in details our analytical study of σFGM. We assume that the reader is aware of the basic definitions and equations provided in the main manuscript.

Analytical study for a single phenotypic character

In the single-character's version of σFGM ($n = 1$), each organism owns two evolvable parameters $\mu \in \mathbb{R}$ and $\sigma \in \mathbb{R}_+$, encoding for the phenotypic distribution $z \sim \mathcal{N}(\mu, \sigma^2)$. The probability density to express the phenotype z is:

$$p(z, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right]. \tag{II.14}$$

The fitness $W(z)$ of the expressed phenotype z reads:

$$W(z) = \exp\left[-z^2/2\right]. \tag{II.15}$$

However, it is much more informative to look at the expected fitness $\overline{W}(\mu, \sigma)$ of an organism (μ, σ), that reads:

$$\begin{aligned} \overline{W}(\mu, \sigma) &= \int_z p(z, \mu, \sigma) W(z) dz \\ &= \int_z \frac{1}{\sigma\sqrt{2\pi}} \exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right] \exp\left[-\frac{z^2}{2}\right] dz. \end{aligned} \tag{II.16}$$

We performed an analytical study of $\overline{W}(\mu, \sigma)$ in the space (μ, σ). We computed the partial derivatives $\partial\overline{W}(\mu, \sigma)/\partial\mu$ and $\partial\overline{W}(\mu, \sigma)/\partial\sigma$ in order to predict what would be the selective pressures on μ and σ, depending on the distance from the fitness optimum $z_{opt} = 0$.

By way of introduction, we know that for f continuously differentiable in x and t, according to the Leibniz's rule:

$$\frac{d}{dt} \int_{\mathbb{R}} f(x, t) dx = \int_{\mathbb{R}} \frac{\partial}{\partial t} f(x, t) dx. \tag{II.17}$$

Let us define the function $f(z, \mu, \sigma)$ such that:

$$f(z, \mu, \sigma) = p(z, \mu, \sigma) W(z) \tag{II.18}$$

then:

$$\overline{W}(\mu, \sigma) = \int_{\mathbb{R}} f(z, \mu, \sigma) dz. \tag{II.19}$$

Partial derivation on μ

According to Equations II.17 and II.19, we know that:

$$\frac{\partial \overline{W}(\mu,\sigma)}{\partial \mu} = \int_z \frac{\partial}{\partial \mu} f(z,\mu,\sigma) dz. \tag{II.20}$$

Let us compute $\partial f/\partial \mu$:

$$\begin{aligned}
\frac{\partial f}{\partial \mu} &= \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{z^2}{2}\right] \left(\exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right]\right)' \\
&= \frac{1}{\sigma\sqrt{2\pi}} \exp\left[-\frac{z^2}{2}\right] \frac{(z-\mu)}{\sigma^2} \exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right] \\
&= \frac{(z-\mu)}{\sigma^2} f(z,\mu,\sigma) \\
&= \frac{(z-\mu)}{\sigma^2} p(z,\mu,\sigma) W(z)
\end{aligned} \tag{II.21}$$

$p(z,\mu,\sigma)$ being a Gaussian density, we know that:

$$\frac{\partial p(z,\mu,\sigma)}{\partial z} = -\frac{(z-\mu)}{\sigma^2}\, p(z,\mu,\sigma). \tag{II.22}$$

Then, Equation II.21 can be rewritten as following:

$$\frac{\partial f(z,\mu,\sigma)}{\partial \mu} = -\frac{\partial p(z,\mu,\sigma)}{\partial z} W(z). \tag{II.23}$$

Here, the goal here is to determine the sign of $\partial \overline{W}(\mu,\sigma)/\partial \mu$, that reads:

$$\begin{aligned}
\frac{\partial \overline{W}(\mu,\sigma)}{\partial \mu} &= \int_z \frac{\partial f(z,\mu,\sigma)}{\partial \mu} dz \\
&= \int_z \underbrace{-\frac{\partial p(z,\mu,\sigma)}{\partial z}}_{\text{Anti-symmetric function}} \times \underbrace{W(z)}_{\text{Symmetric function}} dz.
\end{aligned} \tag{II.24}$$

To determine the sign of $\partial \overline{W}(\mu,\sigma)/\partial \mu$, we must consider the shape of the integrated function, being the product of:

(1) $-\partial p(z,\mu,\sigma)/\partial z$, an anti-symmetric function centered on μ, negative if $z < \mu$, and positive if $z > \mu$ (Fig. II.13a);

(2) $W(z)$, a strictly positive and symmetric function, centered on 0 (Fig. II.13b).

When $\mu = 0$, the product of the anti-symmetric and symmetric functions $\partial p(z,\mu,\sigma)/\partial z \times W(z)$ is an anti-symmetric function, its integral thus being equal to zero (Figs. II.13c and II.13d). Then, the sign of $\partial \overline{W}(\mu,\sigma)/\partial\mu$ depends on μ as following:

$$\begin{cases} \text{If } \mu < 0 \quad , \quad \partial f(z,\mu,\sigma)/\partial\mu > 0 \\ \text{If } \mu = 0 \quad , \quad \partial f(z,\mu,\sigma)/\partial\mu = 0 \\ \text{If } \mu > 0 \quad , \quad \partial f(z,\mu,\sigma)/\partial\mu < 0 \end{cases} \tag{II.25}$$

Hence, for any value of $\sigma > 0$, the selective pressures act to reduce μ towards $\mu = 0$, defining a ridge $\partial \overline{W}(\mu,\sigma)/\partial\mu = 0$ when $\mu = 0$. Any organism owning a value of $\mu \neq 0$ has a lower fitness than an organism with $\mu = 0$, for any given value of σ. This ridge is plotted in purple on Figure II.15.

As revealed in Equation II.21, two other conditions exist for $\partial W(\mu,\sigma)/\partial\mu = 0$:

$$\begin{cases} \text{If } \mu \to \pm\infty \quad , \quad \partial f(z,\mu,\sigma)/\partial\mu = 0 \\ \text{If } \sigma \to +\infty \quad , \quad \partial f(z,\mu,\sigma)/\partial\mu = 0 \end{cases} \tag{II.26}$$

Thus, organisms located very far from the fitness optimum (*i.e.*, $|\mu| \gg 0$), or organisms with a very dispersed phenotypic distribution (*i.e.*, $\sigma \gg 0$), do not experience selective pressures. However, according to Equation II.16, their mean fitness $\overline{W}(\mu,\sigma)$ is almost equal to zero in these conditions.

Another critical condition to clarify for the partial derivative $\partial \overline{W}(\mu,\sigma)/\partial\mu$ is when $\sigma \to 0$. Equation II.24 can help us to solve this special case. Indeed, normal distributions with parameters μ and σ converge towards a Dirac distribution $\delta(z-\mu)$ when $\sigma \to 0$. The Dirac distribution satisfies, for all continuous function $\varphi(z)$,

$$\int_z \delta(z-\mu)\varphi(z)dz = \varphi(\mu). \tag{II.27}$$

The first derivative of the Dirac distribution (in the sense of distributions) satisfies:

$$\int_{\mathbb{R}} \delta'(z-\mu)\varphi(z)dz = -\varphi'(\mu) \tag{II.28}$$

and the n-th derivative, denoted $\delta^{(n)}$, satisfies:

$$\int_z \delta^{(n)}(z-\mu)\varphi(z)dz = (-1)^n \varphi^{(n)}(\mu). \tag{II.29}$$

Thus, starting from Equation II.24, we can derive the following limiting equation when $\sigma \to 0$:

$$\begin{aligned}
\lim_{\sigma\to 0} \frac{\partial \overline{W}}{\partial \mu} &= \lim_{\sigma\to 0} \frac{\partial}{\partial \mu} \int_{\mathbb{R}} p(z,\mu,\sigma)W(z)dz \\
&= \frac{\partial}{\partial \mu} \lim_{\sigma\to 0} \int_{\mathbb{R}} p(z,\mu,\sigma)W(z)dz \\
&= \frac{\partial}{\partial \mu} \int_{\mathbb{R}} \delta(z-\mu)W(z)dz \\
&= \frac{\partial W(\mu)}{\partial \mu} \\
&= W'(\mu).
\end{aligned} \tag{II.30}$$

Thus, when $\sigma \to 0$, the system converges towards canonical FGM scenario, according that $\mu \equiv z$ in this case.

Partial derivation on σ

We first rewrite the equation $f(z,\mu,\sigma)$ to separate σ from other terms:

$$f(z,\mu,\sigma) = \frac{W(z)}{\sqrt{2\pi}} \times \frac{1}{\sigma} \exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right]. \tag{II.31}$$

Let us define the term $a = W(z)/\sqrt{2\pi}$ that does not depend on σ, such that:

$$f(z,\mu,\sigma) = a \times \frac{1}{\sigma} \exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right]. \tag{II.32}$$

The partial derivation of f with respect to σ is more technical than for μ. We describe it step by step below. Let us define the terms u and v such that:

$$\begin{cases} u &= \exp\left[\dfrac{-(z-\mu)^2}{2\sigma^2}\right] \\ v &= \sigma \end{cases} \tag{II.33}$$

Then:

$$\frac{\partial f(z,\mu,\sigma)}{\partial \sigma} = a \times \frac{(u'v - uv')}{v^2} \tag{II.34}$$

with:

$$\begin{cases} u' &= \dfrac{(z-\mu)^2}{\sigma^3}\exp\left[\dfrac{-(z-\mu)^2}{2\sigma^2}\right] \\ \\ v' &= 1 \end{cases} \tag{II.35}$$

Thus, the derivative $\partial f/\partial\sigma$ reads:

$$\begin{aligned} \frac{\partial f(z,\mu,\sigma)}{\partial\sigma} &= a \times \left(\frac{\dfrac{\sigma(z-\mu)^2}{\sigma^3}\exp\left[\dfrac{-(z-\mu)^2}{2\sigma^2}\right] - \exp\left[\dfrac{-(z-\mu)^2}{2\sigma^2}\right]}{\sigma^2} \right) \\ &= a \times \exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right] \times \left(\frac{\dfrac{(z-\mu)^2}{\sigma^2} - 1}{\sigma^2} \right) \\ &= a \times \exp\left[\frac{-(z-\mu)^2}{2\sigma^2}\right] \times \frac{(z-\mu)^2-\sigma^2}{\sigma^4} \\ &= \frac{(z-\mu)^2-\sigma^2}{\sigma^3}\, f(z,\mu,\sigma) \\ &= \frac{(z-\mu)^2-\sigma^2}{\sigma^3}\, p(z,\mu,\sigma)W(z) \end{aligned} \tag{II.36}$$

$p(z)$ being the density of a normal law, we know that:

$$\frac{\partial^2 p(z,\mu,\sigma)}{\partial z^2} = \frac{(z-\mu)^2-\sigma^2}{\sigma^4}\, p(z,\mu,\sigma). \tag{II.37}$$

From Equation II.36, we thus find:

$$\frac{\partial f(z,\mu,\sigma)}{\partial\sigma} = \underbrace{\frac{\partial^2 p(z,\mu,\sigma)}{\partial z^2}}_{\text{Symmetric function}} \times \underbrace{\sigma\, W(z)}_{\text{Symmetric function}} . \tag{II.38}$$

We were not able to compute the ridge $\partial\overline{W}(\mu,\sigma)/\partial\sigma = 0$. We used a numerical scheme to compute $\partial\overline{W}(\mu,\sigma)/\partial\sigma$ depending on μ and σ (see Appendix II.5.3).

However, it is possible to determine analytically some essential characteristics of the ridge $\partial\overline{W}(\mu,\sigma)/\partial\sigma = 0$ (Fig. II.15 orange curve):

(1) The sign of $\partial\overline{W}(\mu,\sigma)/\partial\sigma$ when $\sigma \to 0$;

(2) The position of the inflection point $\mu = d_{th}$, below which the phenotypic noise is always deleterious (Fig. II.15 orange circle).

(1) As previously (Eq. II.30), when $\sigma \to 0$, $\partial\overline{W}(\mu,\sigma)/\partial\sigma$ reads:

$$\begin{aligned} \lim_{x\to 0} \frac{\partial\overline{W}(\mu,\sigma)}{\partial\sigma} &= \int_z \delta''(z-\mu) \times \sigma\ W(z)dz \\ &= (-1)^2\sigma\ W''(\mu) \\ &= \sigma(\mu^2-1)\ W(\mu) \\ &= 0. \end{aligned} \tag{II.39}$$

Thus, when $\sigma \to 0$, the first order selective pressure on σ vanishes, for any values of μ. This means that to determine whether low noise levels are beneficial, one must look at the second derivative of $\overline{W}$ with respect to σ, and identify the inflection points that separate the regions where noise is beneficial or deleterious.

(2) We first compute $\partial^2 f/\partial\sigma^2$. From Equation II.36, we define the terms u and v such that:

$$\begin{cases} u &= \dfrac{(z-\mu)^2-\sigma^2}{\sigma^3} \\ v &= f(z,\mu,\sigma) \end{cases} \tag{II.40}$$

Thus, $\partial^2 f/\partial\sigma^2 = u'v + uv'$, with:

$$\begin{aligned} u' &= \frac{\sigma^2-3(z-\mu)^2}{\sigma^4} \\ v' &= \frac{(z-\mu)^2-\sigma^2}{\sigma^3} f(z,\mu,\sigma) \end{aligned} \tag{II.41}$$

The second derivative of f then reads:

$$\begin{aligned} \frac{\partial^2 f(z,\mu,\sigma)}{\partial\sigma^2} &= \frac{(z-\mu)^2-\sigma^2}{\sigma^3} f(z,\mu,\sigma) + \left(\frac{\sigma^2-3(z-\mu)^2}{\sigma^4}\right)^2 f(z,\mu,\sigma) \\ &= f(z,\mu,\sigma)\left(\frac{\sigma^4-3\sigma^2(z-\mu)^2+(z-\mu)^4-2(z-\mu)^2\sigma^2+\sigma^4}{\sigma^6}\right) \\ &= f(z,\mu,\sigma)\left(\frac{2\sigma^4-5(z-\mu)^2\sigma^2+(z-\mu)^4}{\sigma^6}\right). \end{aligned} \tag{II.42}$$

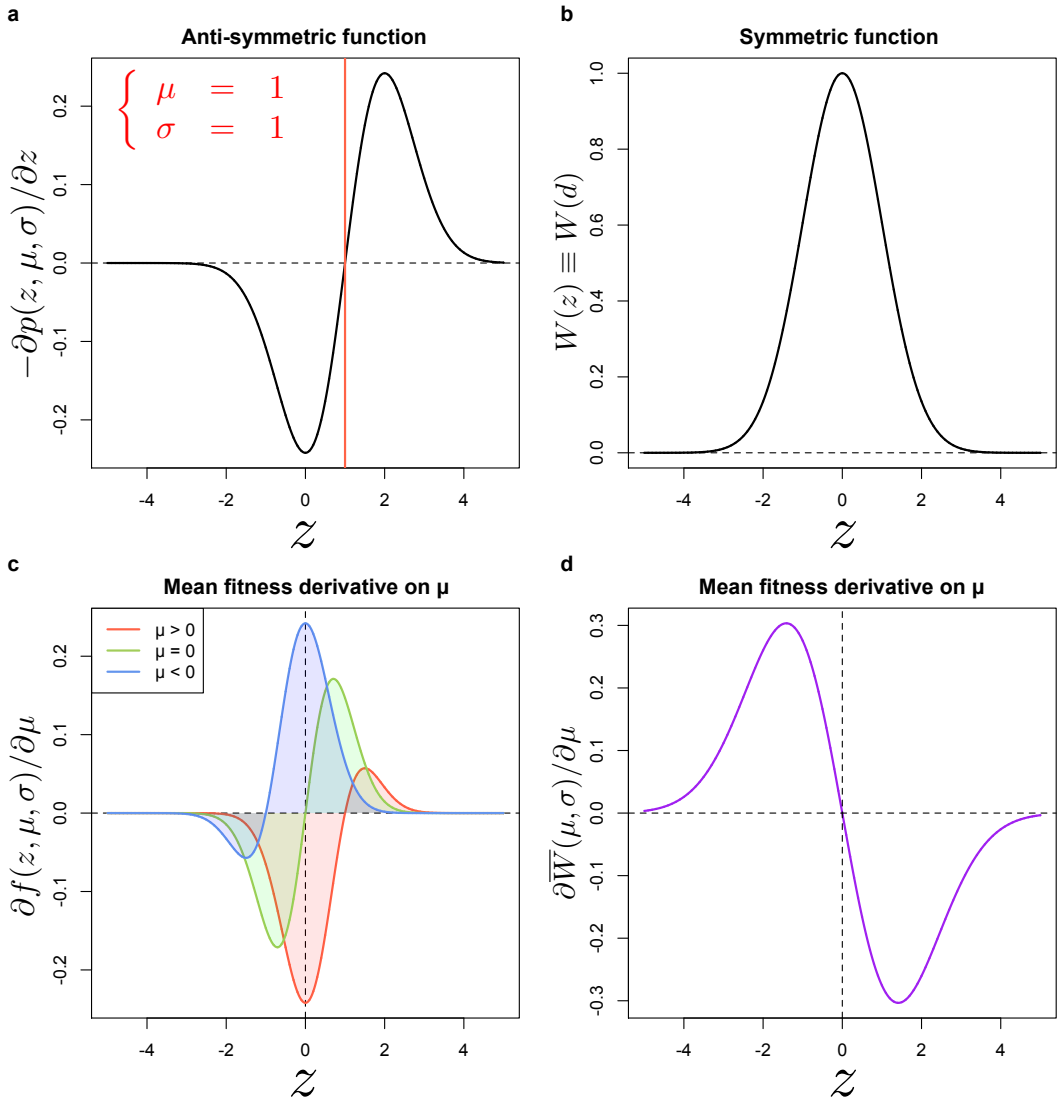


Figure II.13 – Behavior of $\partial \overline{W}(\mu,\sigma)/\partial \mu$. **a,** The normal distribution derivative $-\partial p(z,\mu,\sigma)/\partial z$ is an anti-symmetric function centered on $z = \mu$, such that the function is negative when $z < \mu$ and positive when $z > \mu$. Here, an example is given for $\mu = 1$ and $\sigma = 1$. The red vertical line represents the value of μ, which is an axis of symmetry of the function. **b,** The Gaussian-shaped fitness function $W(z)$ is a strictly positive, symmetric function centered on $z = 0$. **c,d,** The product of both functions, equal to $\partial f(z,\mu,\sigma)/\partial \mu$ is biased towards positive values if $\mu < 0$ (blue curve), towards negative values if $\mu > 0$ (red curve), or anti-symmetric if $\mu = 0$ (green curve). Thus the integral $\int_z \partial f(z,\mu,\sigma)/\partial \mu \; dz$ is respectively positive, negative, or zero, if μ is negative, positive or zero (as shown in panel **d**, purple curve).

As demonstrated previously, it is possible to rewrite this equation to extract derivatives

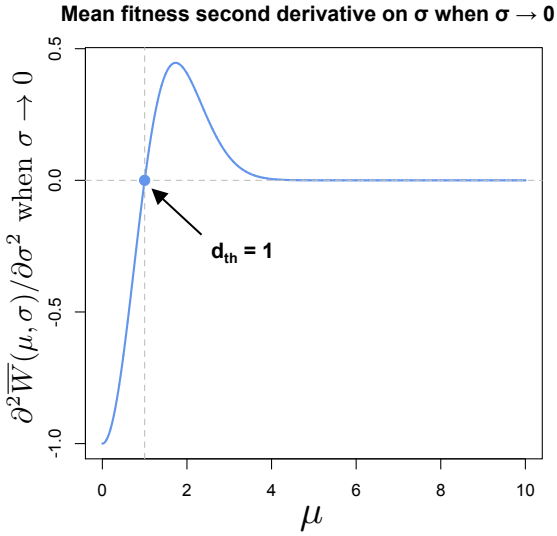


Figure II.14 – Variation of $\partial^2\overline{W}(\mu,\sigma)/\partial\sigma^2$ when $\sigma \to 0$. The variation of $\partial^2\overline{W}(\mu,\sigma)/\partial\sigma^2$ when $\sigma \to 0$ is represented here depending on μ. It is described by Equation II.44. Blue dot: inflection point of the ridge $\partial\overline{W}(\mu,\sigma)/\partial\sigma = 0$ (Fig. II.15 orange circle), for $\mu = d_{th} = 1$.

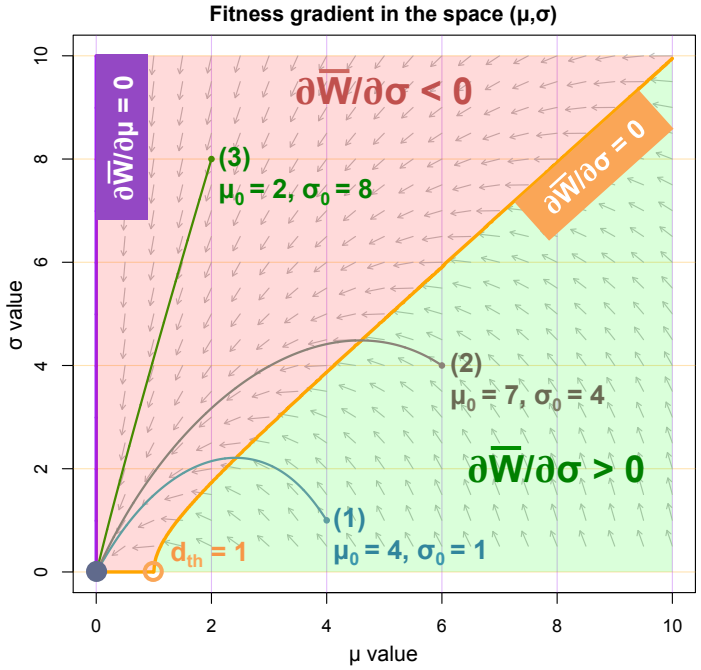


Figure II.15 – Variations of the mean fitness $\overline{W}(\mu,\sigma)$ in the space (μ,σ). In purple, the ridge $\partial\overline{W}/\partial\mu = 0$. In orange, the ridge $\partial\overline{W}/\partial\sigma = 0$. Green area: $\partial\overline{W}/\partial\sigma > 0$ (*i.e.*, it is beneficial to increase the phenotypic noise). Red area: $\partial\overline{W}/\partial\sigma < 0$ (*i.e.*, it is beneficial to reduce the phenotypic noise). Three trajectories following the fitness gradient are represented **(1)** in blue (initial values: $\mu_0 = 4, \sigma_0 = 1$), **(2)** in brown (initial values: $\mu_0 = 6, \sigma_0 = 4$) and **(3)** in green (initial values: $\mu_0 = 2, \sigma_0 = 8$). Black dot: fitness optimum z_{opt}. Orange circle: inflection point $d_{th} = 1$ of the fitness landscape $W(z)$. Grey arrows indicate the direction of the fitness gradient, but not its amplitude.

of normal law densities:

$$\begin{aligned}
\frac{\partial^2 f(z,\mu,\sigma)}{\partial\sigma^2} &= f(z,\mu,\sigma)\left(\frac{(z-\mu)^4-6(z-\mu)^2\sigma^2+3\sigma^4}{\sigma^6}+\frac{(z-\mu)^2\sigma^2-\sigma^4}{\sigma^6}\right)\\
&= \sigma^2\ f(z,\mu,\sigma)\left(\frac{(z-\mu)^4-6(z-\mu)^2\sigma^2+3\sigma^4}{\sigma^8}+\frac{1}{\sigma^2}\frac{(z-\mu)^2-\sigma^2}{\sigma^4}\right)\\
&= \sigma^2\ f(z,\mu,\sigma)\left(\frac{(z-\mu)^4-6(z-\mu)^2\sigma^2+3\sigma^4}{\sigma^8}+\frac{(z-\mu)^2-\sigma^2}{\sigma^6}\right)\\
&= \sigma^2\ W(z)\left(\frac{\partial^4 p(z,\mu,\sigma)}{\partial z^4}+\frac{1}{\sigma^2}\times\frac{\partial^2 p(z,\mu,\sigma)}{\partial z^2}\right)\\
&= \underbrace{\left(\sigma^2\ W(z)\frac{\partial^4 p(z,\mu,\sigma)}{\partial z^4}\right)}_{\to 0 \text{ when } \sigma\to 0}+\left(W(z)\frac{\partial^2 p(z,\mu,\sigma)}{\partial z^2}\right).
\end{aligned} \tag{II.43}$$

Consequently, when $\sigma \to 0$, the partial derivative $\partial^2\overline{W}(\mu,\sigma)/\partial\sigma^2$ reads:

$$\lim_{x\to 0}\ \frac{\partial^2\overline{W}(\mu,\sigma)}{\partial\sigma^2} = (\mu^2-1)\ W(\mu). \tag{II.44}$$

The only value of μ for which $\partial^2\overline{W}(\mu,\sigma)/\partial\sigma^2 = 0$ is $|\mu| = 1$. When $|\mu| < 1$, $\partial^2\overline{W}(\mu,\sigma)/\partial\sigma^2 < 0$, meaning that the selective pressure is towards a reduction of the phenotypic noise σ. When $|\mu| > 1$, $\partial^2\overline{W}(\mu,\sigma)/\partial\sigma^2 > 0$, meaning that the selective pressure is towards an increase of the phenotypic noise σ (Fig. II.14 blue curve, and Fig. II.15 orange curve and circle). Thus, $d_{th} = 1$.

Analytical and numerical studies of an isotropic noise on n phenotypic characters

As described in Results, an isotropic noise is applied to the mean phenotype $\boldsymbol{\mu}$, by independently varying each trait value μ_i with the same noise amplitude σ. The probability $p(\boldsymbol{z},\boldsymbol{\mu},\sigma)$ for an organism $(\boldsymbol{\mu},\sigma)$ to express the phenotype $\boldsymbol{z}$ is then:

$$p(\boldsymbol{z},\boldsymbol{\mu},\sigma) = \prod_{i\in n}\frac{1}{\sigma\sqrt{2\pi}}\exp\left[\frac{-(z_i-\mu_i)^2}{2\sigma^2}\right]. \tag{II.45}$$

As in the previous section, the goal is to compute the second derivative of $\overline{W}(\boldsymbol{\mu},\sigma)$ on σ, and find its inflection point to detect the critical euclidean distance d_{th} below which

phenotypic noise must be minimized. However, we now must compute it in n dimensions. Hopefully, two conditions allow us to strongly simplify the equations:

(1) Noise is isotropic, such that $\overline{W}(\boldsymbol{\mu}, \sigma)$ can be decomposed in a product of one-dimensional integrals;

(2) The mean phenotype of an organism $\boldsymbol{\mu}$ is taken away from the fitness optimum on a single axis, all other axes remaining at a distance zero of the fitness optimum. By rotational symmetry, we can generalize to any position $\boldsymbol{\mu}$ away from the fitness optimum.

$\overline{W}(\boldsymbol{\mu}, \sigma)$ reads:

$$\begin{aligned} \overline{W}(\boldsymbol{\mu}, \sigma) &= \int_{\mathbb{R}^n} p(\boldsymbol{z}, \boldsymbol{\mu}, \sigma) W(\boldsymbol{z}) d\boldsymbol{z} \\ &= \int_{\mathbb{R}} p(z_1, \mu_1, \sigma) W(z_1) dz_1 \times ... \times \int_{\mathbb{R}} p(z_n, \mu_n, \sigma) W(z_n) dz_n. \end{aligned} \tag{II.46}$$

If we only move the mean phenotypic trait value μ_1 away from the fitness optimum, all other mean trait values being equal to zero, $\overline{W}(\boldsymbol{\mu}, \sigma)$ then reads:

$$\overline{W}(\boldsymbol{\mu}, \sigma) = \int_{\mathbb{R}} p(z_1, \mu_1, \sigma) W(z_1) dz_1 \times \left(\int_{\mathbb{R}} p(z, 0, \sigma) W(z) dz \right)^{n-1}. \tag{II.47}$$

As demonstrated above, we know that for a single character and when $\sigma \to 0$, the successive derivatives of $\overline{W}(\mu, \sigma)$ read:

$$\begin{cases} \overline{W} &= W(\mu) \\ \overline{W}' &= 0 \\ \overline{W}'' &= (\mu^2 - 1) W(\mu) \end{cases} \tag{II.48}$$

Moreover, when $\mu = 0$, we find that:

$$\begin{cases} \overline{W}_0 &= 1 \\ \overline{W}_0' &= 0 \\ \overline{W}_0'' &= -1 \end{cases} \tag{II.49}$$

We then define the terms a, b and c such that:

$$\begin{cases} a &= \int_{\mathbb{R}} p(z_1, \mu_1, \sigma) W(z_1) dz_1 \\ b &= \int_{\mathbb{R}} p(z, 0, \sigma) W(z) dz \\ c &= b^{\,n-1} \end{cases} \tag{II.50}$$

We now compute the second derivative of $\overline{W}(\boldsymbol{\mu}, \sigma)$ on σ, according to a, b and c, this equation reads:

$$\frac{\partial^2 \overline{W}(\boldsymbol{\mu}, \sigma)}{\partial \sigma^2} = ac'' + 2a'c' + a''c \tag{II.51}$$

With:

$$\begin{cases} c' &= (n-1)b'b^{\,n-2} \\ c'' &= (n-1)(n-2)b'b^{\,n-3} + (n-1)b''b^{\,n-2} \end{cases} \tag{II.52}$$

Then, the complete equation of $\partial \overline{W}(\boldsymbol{\mu}, \sigma)/\partial \sigma$ reads:

$$\begin{aligned} \frac{\partial^2 \overline{W}(\boldsymbol{\mu}, \sigma)}{\partial \sigma^2} &= a\left((n-1)(n-2)b'b^{\,n-3} + (n-1)b''b^{\,n-2}\right) \\ &+ 2a'(n-1)b'b^{\,n-2} \\ &+ a''b^{\,n-1}. \end{aligned} \tag{II.53}$$

We now replace the terms a and b by the corresponding terms in Equations II.48 and II.49:

$$\begin{aligned} \frac{\partial^2 \overline{W}(\boldsymbol{\mu}, \sigma)}{\partial \sigma^2} &= \overline{W}\left((n-1)(n-2)\overline{W}_0'\overline{W}_0^{\,n-3} + (n-1)\overline{W}_0''\overline{W}_0^{\,n-2}\right) \\ &+ 2\overline{W}'(n-1)\overline{W}_0'\overline{W}_0^{\,n-2} \\ &+ \overline{W}''\overline{W}_0^{\,n-1}. \end{aligned} \tag{II.54}$$

Finally, the second derivative of $\overline{W}(\boldsymbol{\mu}, \sigma)$ on σ reads:

$$\frac{\partial^2 \overline{W}(\boldsymbol{\mu}, \sigma)}{\partial \sigma^2} = (\mu_1^2 - n)W(\mu_1). \tag{II.55}$$

The only value of μ_1 for which Equation II.55 is equal to zero is $\mu_1 = \pm\sqrt{n}$. Since this result is valid when the euclidean distance $d = \|\boldsymbol{\mu}\|$ is equal to $\sqrt{n}$, we can conclude

that the critical distance below which isotropic phenotypic noise must be minimized is $d_{th} = \sqrt{n}$.

Anisotropic and correlated phenotypic noise is beneficial when aligned with the fitness optimum

Let us consider the organism $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ in a n-dimensional phenotypic space, sitting at a certain distance of the fitness optimum $\boldsymbol{z_{opt}}$ (beyond $d_{th} = \sqrt{n}$) and evolving towards it. We describe the phenotypic noise of this organism by a multivariate normal distribution $\mathcal{N}_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$. This multivariate normal distribution can be represented by an hyper-ellipse in $\mathbb{R}^n$, as shown in Figure II.16 for two dimensions.

We now define the axes $\boldsymbol{v_1}, ..., \boldsymbol{v_n}$ (the origin of the new basis also being $\boldsymbol{\mu}$), with $\boldsymbol{v_1}$ aligned towards the fitness optimum $\boldsymbol{z_{opt}}$, all other axes $\boldsymbol{v_i}$ being orthogonal to $\boldsymbol{v_1}$ (Fig. II.16a). Along axis $\boldsymbol{v_1}$, the organism $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ experiences a convex fitness if $\|\boldsymbol{\mu}\| > 1$. Along all other axes $\boldsymbol{v_i}$, the organism experiences a concave fitness, sitting at a local optimum in all cases (Fig. II.16b). The basis associated to axes $\boldsymbol{v_1}, ..., \boldsymbol{v_n}$ is the orthonormal matrix $\boldsymbol{V}$ of size $n \times n$, where $\boldsymbol{v_1}$ is defined by the vector $\boldsymbol{\mu}$. By defining $\overline{\boldsymbol{\mu}} = \boldsymbol{\mu}/\|\boldsymbol{\mu}\|$, the matrix $\boldsymbol{V}$ reads:

$$\boldsymbol{V} = \left(\overline{\boldsymbol{\mu}} | ...\right) \tag{II.56}$$

The goal here is to find the phenotypic noise configuration that maximizes the expected fitness $\overline{W}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, knowing that:

$$\overline{W}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \int_{\mathbb{R}^n} p(\boldsymbol{z}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) W(\boldsymbol{z}) d\boldsymbol{z}. \tag{II.57}$$

We first make a variable change by defining $\boldsymbol{\epsilon}$ such that the realized phenotype $\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\epsilon}$. Thus:

$$\overline{W}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) = \int_{\mathbb{R}^n} p(\boldsymbol{\epsilon}, \boldsymbol{0}, \boldsymbol{\Sigma}) W(\boldsymbol{\mu} + \boldsymbol{\epsilon}) d\boldsymbol{\epsilon}. \tag{II.58}$$

According to our previous results, we know that if $\|\boldsymbol{\mu}\| > 1$, phenotypic noise is beneficial along $\boldsymbol{v_1}$ axis, where $W(\boldsymbol{z})$ is convex, and is deleterious along all other orthogonal axes of $\boldsymbol{V}$ (Fig. II.16b). Thus, for any covariance matrix $\boldsymbol{\Sigma}$ and for any $\boldsymbol{\epsilon} \sim \mathcal{N}_n(\boldsymbol{0}, \boldsymbol{\Sigma})$, the fitness $W(\boldsymbol{\mu} + \boldsymbol{\epsilon})$of the expressed phenotype $\boldsymbol{z} = \boldsymbol{\mu}, \boldsymbol{\epsilon}$ is always lower or equal to the fitness of its projection along axis $\boldsymbol{v_1}$ (*i.e.*, the distance towards the optimum will always be shorter or equal for the projection). Thus:

$$\int_{\mathbb{R}^n} p(\boldsymbol{\epsilon}, \boldsymbol{0}, \boldsymbol{\Sigma}) W(\boldsymbol{\mu} + \boldsymbol{\epsilon}) d\boldsymbol{\epsilon} \leq \int_{\mathbb{R}^n} p(\boldsymbol{\epsilon}, \boldsymbol{0}, \boldsymbol{\Sigma}) W(\boldsymbol{\mu} + \overline{\boldsymbol{\mu}}^T \boldsymbol{\epsilon} \overline{\boldsymbol{\mu}}) d\boldsymbol{\epsilon}. \tag{II.59}$$

We then express $\boldsymbol{\epsilon}$ in the basis $\boldsymbol{V}$ by making the following variable change: $\boldsymbol{s} = \boldsymbol{V}^T \boldsymbol{\epsilon}$.

Consequently, $\boldsymbol{\epsilon} = \boldsymbol{V}\boldsymbol{s}$, such that:

$$\begin{aligned} \overline{\boldsymbol{\mu}}^T \boldsymbol{\epsilon} \overline{\boldsymbol{\mu}} &= \overline{\boldsymbol{\mu}}^T \boldsymbol{V}\boldsymbol{s}\overline{\boldsymbol{\mu}} \\ &= s_1 \overline{\boldsymbol{\mu}}. \end{aligned} \tag{II.60}$$

We can rewrite the right term of the Equation II.59 as following:

$$\int_{\mathbb{R}^n} p(\boldsymbol{V}\boldsymbol{s}, \boldsymbol{0}, \boldsymbol{\Sigma}) \underbrace{W(\boldsymbol{\mu} + s_1\overline{\boldsymbol{\mu}})}_{\text{Only depends on } s_1} ds_1, ..., ds_n. \tag{II.61}$$

The term $W(\boldsymbol{\mu}+s_1\overline{\boldsymbol{\mu}})$ only depending on s_1, we can extract it from the integral by writing:

$$\int_{\mathbb{R}} W(\boldsymbol{\mu} + s_1\overline{\boldsymbol{\mu}}) \left[\int_{\mathbb{R}^{n-1}} p(\boldsymbol{V}\boldsymbol{s}, \boldsymbol{0}, \boldsymbol{\Sigma}) ds_2, ..., ds_n \right] ds_1. \tag{II.62}$$

The probability density function $p(\boldsymbol{V}\boldsymbol{s}, \boldsymbol{0}, \boldsymbol{\Sigma})$ from Equation II.62 is strictly equivalent to $p(\boldsymbol{s}, \boldsymbol{0}, \boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V})$ (demonstration not shown here). In this case, the inner integral of Equation II.62 reads:

$$\int_{\mathbb{R}^{n-1}} p(\boldsymbol{s}, \boldsymbol{0}, \boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}) ds_2, ..., ds_n. \tag{II.63}$$

This equation (Eq. II.63) describes the marginal density of s_1, following the univariate normal law:

$$s_1 \sim \mathcal{N}(0, \left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1}) \tag{II.64}$$

the subscript "$1,1$" denoting the coefficient of the first raw and first column.

Then, Equation II.63 reads:

$$\int_{\mathbb{R}} W(\boldsymbol{\mu} + s_1\overline{\boldsymbol{\mu}}) p(s_1, 0, \left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1}) ds_1. \tag{II.65}$$

Note that, by rotational symmetry, $W(\boldsymbol{\mu} + s_1\overline{\boldsymbol{\mu}}) = W((\|\boldsymbol{\mu}\| + s_1)\boldsymbol{e_1})$, because vectors $\boldsymbol{\mu}$ and $\overline{\boldsymbol{\mu}}$ are aligned. Thus, Equation II.65 is just the one dimensional expected fitness $\overline{W}(\|\boldsymbol{\mu}\|, \left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1})$. Previous results demonstrated that along axis $\boldsymbol{v_1}$, phenotypic noise properties can be described in one dimension, such that there is an optimal value $\left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1} = \sigma^2_{opt}$ that maximizes the expected fitness $\overline{W}(\|\boldsymbol{\mu}\|, \left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1})$. This leads to the inequality:

$$\overline{W}(\|\boldsymbol{\mu}\|, \left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1}) = \int_{\mathbb{R}} W(z) p(z, \|\boldsymbol{\mu}\|, \left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1}) dz \leq \overline{W}(\|\boldsymbol{\mu}\|, \sigma^2_{opt}). \tag{II.66}$$

By decomposing the covariance matrix $\boldsymbol{\Sigma}$ in its eigenvalues $\boldsymbol{\sigma}^2$, such that $\boldsymbol{\Sigma} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T$, we obtain:

$$\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V} = \boldsymbol{V}^T\boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T\boldsymbol{V}. \tag{II.67}$$

If the eigenvector $\boldsymbol{u_1}$ of $\boldsymbol{U}$ (Fig. II.16a) is aligned with the axis $\boldsymbol{v_1}$ of the basis $\boldsymbol{V}$, then the best phenotypic noise configuration is to have $\boldsymbol{\sigma}^2 = \{\sigma^2_{opt}, 0, ..., 0\}$ (with $\boldsymbol{D} = \mathrm{diag}(\boldsymbol{\sigma}^2)$).

As a whole, our analytical study led to two important inequalities described in Equations II.59 and II.66. For any organism $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ where $\|\boldsymbol{\mu}\| > 1$, the first inequality (Eq. II.59) demonstrates that it is always preferable to not have positive phenotypic noise that is not aligned with the fitness optimum (*i.e.*, along the axis $\boldsymbol{v_1}$). However, if none of the principal components $\boldsymbol{u_i}$ of the covariance matrix $\boldsymbol{\Sigma}$ are aligned with the fitness optimum, this optimal scenario is unreachable because there is always positive phenotypic noise orthogonal to the direction of the fitness optimum $\boldsymbol{u_i}$ (the inequality II.59 is strict). On the contrary, if one of the principal components $\boldsymbol{u_1}$ is aligned with the fitness optimum, it is possible to minimize orthogonal noise components (the inequality II.59 is not strict).

Moreover, in the case where $\boldsymbol{u_1}$ is aligned with the fitness optimum and if orthogonal noise components are set to zero (*i.e.*, phenotypic noise is one-dimensional), results presented above show that it exists an optimal noise amplitude σ^2_{opt} that maximizes the one dimensional expected fitness $\overline{W}(\|\boldsymbol{\mu}\|, \left[\boldsymbol{V}^T\boldsymbol{\Sigma}\boldsymbol{V}\right]_{1,1})$. This corresponds to setting all the eigenvalues of $\boldsymbol{\Sigma}$ to zero, except the one associated to $\boldsymbol{u_1}$, that is equal to σ^2_{opt}.

Concluding remarks

Our demonstrations mainly rely on the study of the local convexity of the fitness function. As such, any function admitting the same properties as $W(\boldsymbol{z}) = \exp\left[-\boldsymbol{z}^2/2\right]$ (*i.e.*, a positive function being concave at the optimum and admitting one convex inflection point) will give the same general results. Moreover, in our study, phenotypic noise and mutation distributions are Gaussian-shaped. As demonstrated above, this choice allowed us to obtain precise analytical results.

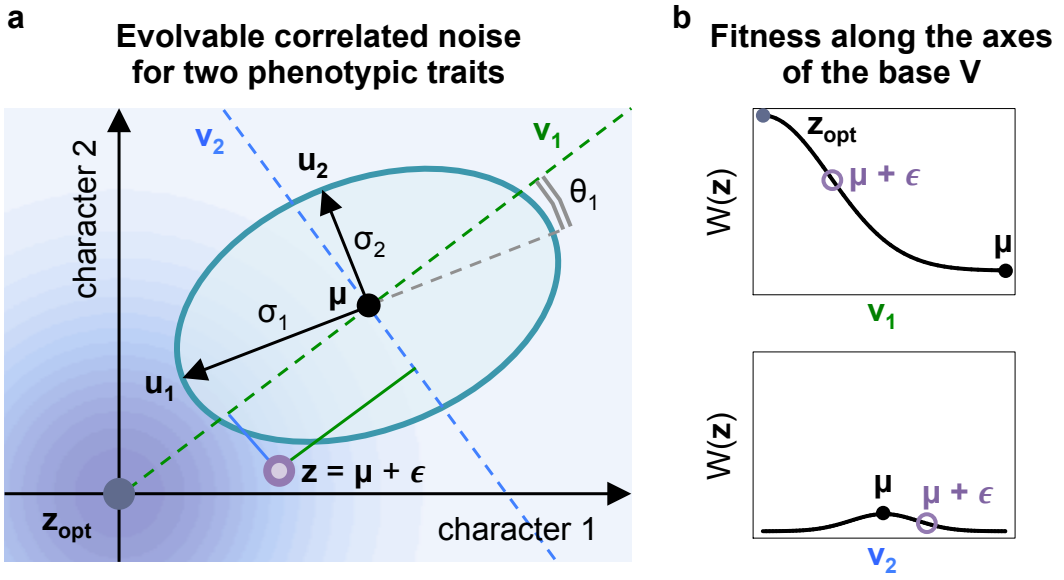


Figure II.16 – Anisotropic and correlated phenotypic noise for two traits. **a,** The phenotypic distribution of an organism $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{\theta})$ is defined by a multivariate normal distribution with mean $\boldsymbol{\mu}$ (black dot), noise amplitudes σ_1 and σ_2 (black arrows) along axes $\boldsymbol{u_1}$ and $\boldsymbol{u_2}$, and a parameter of correlation θ_1 (grey angle), defining a rotation of the basis $\boldsymbol{U} = (\boldsymbol{u_1}, \boldsymbol{u_2})$. A phenotype $\boldsymbol{z}$ (purple dot) is generated from the multivariate normal distribution by drawing a random vector $\boldsymbol{\epsilon} \sim \mathcal{N}_n(\mathbf{0}, \boldsymbol{\Sigma})$ (with $\boldsymbol{\Sigma}$ the covariance matrix built from $\boldsymbol{\sigma}$ and $\boldsymbol{\theta}$), such that $\boldsymbol{z} = \boldsymbol{\mu} + \boldsymbol{\epsilon}$. The contribution of $\boldsymbol{\epsilon}$ on each axis $\boldsymbol{v_1}$ (in green) and $\boldsymbol{v_2}$ (in blue) of the basis $\boldsymbol{V}$, where $\boldsymbol{v_1}$ is aligned with the fitness optimum $\boldsymbol{z_{opt}}$, is represented by the vector $\boldsymbol{s} = (s_1, s_2)^T$. The fitness landscape is represented by a gradient of blue centered on the fitness optimum $\boldsymbol{z_{opt}}$ (blue dot). **b,** Fitness along axes of the basis $\boldsymbol{V} = (\boldsymbol{v_1}, \boldsymbol{v_2})$. Along axis $\boldsymbol{v_1}$, aligned with the fitness optimum $\boldsymbol{z_{opt}}$, the organism experiences a convex fitness (if $\|\boldsymbol{\mu}\| > 1$). Along axis $\boldsymbol{v_2}$, orthogonal to $\boldsymbol{v_1}$, the organism a concave fitness, sitting on a local fitness optimum.

II.5.5 Data S1. Phenotypic noise correlations matrices of each replicate of the 37 strains of yeast in Fisher's space.

https://github.com/charlesrocabert/SigmaFGM/tree/master/phenomics_analysis/DataS1.zip

II.5.6 Data S2. Phenotypic noise correlations matrices of each of the 37 strains of yeast in Fisher's space, with Pearson correlation tests.

https://github.com/charlesrocabert/SigmaFGM/tree/master/phenomics_analysis/DataS2.zip

II.5.7 Script S1. A numerical solver for σFGM.

https://github.com/charlesrocabert/SigmaFGM

II.5.8 Script S2. Phenomics analysis of 37 strains of yeast.

https://github.com/charlesrocabert/SigmaFGM/tree/master/phenomics_analysis

Part B

An *in silico* experimental evolution approach to study Evolution of Evolution

Chapter III

Evo^{2}Sim, a multi-scale model dedicated to Evolution of Evolution

The development of Evo^{2}Sim is part of the European project EvoEvo (FP7-ICT-610427). The description of work of the project and the deliverables related to Evo^{2}Sim are freely available at www.evoevo.eu.

> Although there has been much discussion on what is the appropriate level on which Darwinian selection operates, we now know that in many cases the interesting features arise through the occurrence of multiple levels of selection which act in concordance and/or in conflict.
> (Hogeweg and Takeuchi, 2003)

III.1 Meet Evo²Sim

Figure III.1 – Evo²Sim logo.

Evo²Sim is a **multi-scale** and **individual-based** model of evolution, inspired from **pearls-on-a-string** (Crombach and Hogeweg, 2008) and **sequence-of-nucleotides** (Beslon et al., 2010b; Hindré et al., 2012). As discussed in introduction, developing complex representations of the genotype-to-phenotype map and fitness landscape has been a primary goal in the conception of this model. To do so, we used the **bag-of-tuples** formalism (also discussed in introduction) to develop an **artificial chemistry** allowing for multi-level evolution.

Two major objectives constrained the development of Evo²Sim: **(i)** integrate a maximum number of pertinent biological structures and levels (genome, genetic regulation, metabolic network, cell, population, ...) to enable deep exploration of EvoEvo, and **(ii)** maintain the model complexity low enough to enable its practical use. As such, a tough compromise had to be made between the degree of realism (the number of assumptions we want to pick to build the model, see Servedio et al. 2014), and what we want the model to tell us. This modeling problem is well-resumed by the concept of Medawar zone: as illustrated in Figure III.2, the Medawar zone is the area where the model complexity is most likely to produce fruitful results. Too simple models are unlikely to produce novel or significant results. Too complex models may not succeed at all or may be rejected by the research community at large (Loehle, 1990).

As described in the description of work of the EvoEvo project[1], a primary objective in the development of Evo²Sim was to merge the R-ævol model (Beslon et al., 2010b), which includes a complex representation of the genome and the genetic regulation network, with the pearls-on-a-string formalism, which is very flexible and allows for vast modeling possibilities at the level of the regulation and the metabolism (Crombach and Hogeweg, 2007, 2008, 2009). Six biological structures have been modeled in Evo²Sim. **(i)** The **genome** encodes two interlaced networks: **(ii)** the **genetic regulatory network**, that controls gene expression, and **(iii)** the **metabolic network**, that allows the cell to perform tasks

[1] Available at www.evoevo.eu

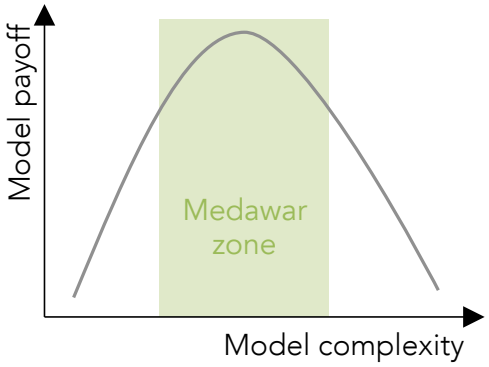


Figure III.2 – The Medawar zone. The Medawar zone is the area where the model complexity is most likely to produce fruitful results. Too simple models are unlikely to produce novel or significant results. Too complex models may not succeed at all or may be rejected by the research community at large (Loehle, 1990).

in interaction with the environment. **(iv)** Together, these three first levels form the fourth one: the **cell**. By uptaking, transforming and releasing metabolites (actively or at death), the cell grows, and produces material necessary to its division. **(v)** Living cells compose the **population**, and evolve in **(vi)** a two dimensional **environment**, in which free metabolites diffuse and degrade over time.

The fitness of each organism depends on the production of essential metabolites, built from available resources in the local environment. Doing so, organisms constantly modify their environment, thus perturbing selective pressures. Free metabolites can be depleted, new unseen free metabolites can appear in the environment, resources can cycle, and so on. The fitness landscape is then completely dependent on the interaction between the population and the environment, and is evolvable. In this sense, it could be more appropriate to use the term **fitness seascape** to render the fluctuating selective pressures (Mustonen and Lässig, 2009).

In the following section, the modeling choices for the artificial chemistry, the genotype-to-phenotype map and the fitness landscape will be described in more details.

III.2 The genome

III.2.1 Genome structure

In Evo^{2}Sim, the genome structure mimics bacterial genomes organization, with some simplifications. Following the pearls-on-a-string formalism, the resolution of genomic sequences is coarse-grained: no nucleotides representation here, a sequence is made of **genetic units**, somehow corresponding to small DNA sequences carrying specific functions. Thus, the genome is a circular, single-stranded sequence of genetic units, belonging

to five different types: **non-coding units** (**NC**), **promoter units** (**P**), **binding site units** (**BS**), **transcription factor coding units** (**TF**) and **enzyme coding units** (**E**). There is a unique, hard-coded, reading frame. Each genetic unit is an ordered list of attributes (a tuple), and has a specific role in the mapping. The interactions between the various objects of Evo^{2}Sim artificial chemistry are defined by integer values called "tags". For example, if a transcription factor tag matches to a binding site tag, the transcription factor is allowed to bind to it. Metabolites are also implicitly defined by tags $\in \mathbb{N}$. In this case, we refer to the metabolite by # (*e.g.* metabolite #10). The different genetic units and their attributes are described below, and summarized in Table III.1:

(1) Non-coding units (**NC**) have no particular function. They constitute the non-coding part of the genome, which has been demonstrated to have a strong influence on the long-term evolution of the genome structure (Knibbe et al., 2007a);

(2) Promoter units (**P**) contain a floating-point number $\beta \in [0.0, 1.0]$ representing the production rate of the protein(s) under its control. Indeed, transcription and translation are implicit processes in Evo^{2}Sim. β can be up or down-regulated by the regulation network;

(3) Binding site units (**BS**) participate to the regulation if they flank promoters upstream (enhancer site) or downstream (operator site), and if transcription factors bind to them (Fig. III.3). To this aim, they own a transcription factor tag $\mathrm{TF}_{\mathrm{tag}} \in \mathbb{Z}$ indicating which transcription factors can bind;

(4) Transcription factor coding units (**TF**) encode for transcription factors whose properties are defined by four attributes: the binding site tag $\mathrm{BS}_{\mathrm{tag}} \in \mathbb{Z}$ indicates on which binding site to bind. The co-enzyme tag $\mathrm{CoE}_{\mathrm{tag}} \in \mathbb{N}^*$ indicates which co-enzyme can bind to the transcription factor, and activate or inhibit it. The co-enzyme constant k_{CoE}, the free activity A_{free} and the bound activity A_{bound} define the effect of the co-enzyme on the transcription factor. Finally, the binding window W_{bind} controls the transcription factor binding affinity, allowing it to bind on a binding site with a certain degree of mismatch;

(5) Enzyme coding units (**E**) encode for enzymes, that catalyze metabolic reactions. Four attributes define the activity of an enzyme: the substrate tag $s \in \mathbb{N}^*$; the product tag $p \in \mathbb{N}^*$; and two constants $k_{cat} \in \mathbb{R}$ and $k_{cat}/k_m \in \mathbb{R}_+$. These attributes define the properties of the Michaelis-Menten equation ruling the metabolic reaction.

As for real bacteria, Evo^{2}Sim genomes own **functional regions** having some transcriptional activity, and **non-coding regions**. Functional regions must have the following pattern: a promoter (**P**), optionally flanked upstream or downstream by one or more binding sites (**BS**), followed by one or more contiguous coding units (**E** or **TF**). A promoter can thus control several coding units, like in **bacterial operons**. Upstream binding sites constitute the **enhancer site** of a promoter, that up-regulates its activity. Downstream binding sites constitute the **operator site** of a promoter, that down-regulates its activity. The first unit that is not a coding one interrupts the transcription and marks

the end of the functional region. Importantly, apart from non-coding units **(NC)**, any units that are not correctly ordered to form a functional region also compose the non-coding part of the genome. Figures III.3a and III.3b give some examples of combinations of genetic units forming functional or non-functional regions. The structure of a typical functional region is also presented in Figure III.3c.

Table III.1 – Presentation of the five types of genetic units. Each genetic unit is represented by a graphical symbol (that will be used in further figures), and is an ordered list of attributes (a tuple). **NC** units have no attributes. **P** units have one attribute (the basal expression level β). **BS** units also have one attribute (the transcription factor tag TF_{tag}). **TF** coding units own 5 attributes (the binding site tag BS_{tag}, the co-enzyme tag CoE_{tag}, the co-enzyme constant k_{CoE}, the free and bound activities A_{free} and A_{bound}, and the binding window W_{bind}). **E** coding units own 4 attributes (the substrate tag s, the product tag p, the k_{cat} constant, and the k_{cat}/k_m constant. The role of each genetic unit is detailed in the following sections.

Type of genetic unit	Attributes	Graphical symbol
Non coding unit (NC)	No attributes;	
Promoter unit (P)	Basal expression level β;	
Binding site unit (BS)	Transcription factor tag TF_{tag};	
Transcription factor coding unit (TF)	Binding site tag BS_{tag}; Co-enzyme tag CoE_{tag}; Co-enzyme constant k_{CoE}; Free activity A_{free}; Bound activity A_{bound}; Binding window W_{bind};	
Enzyme coding unit (E)	Substrate tag s; Product tag p; k_{cat} constant; k_{cat}/k_m constant;	

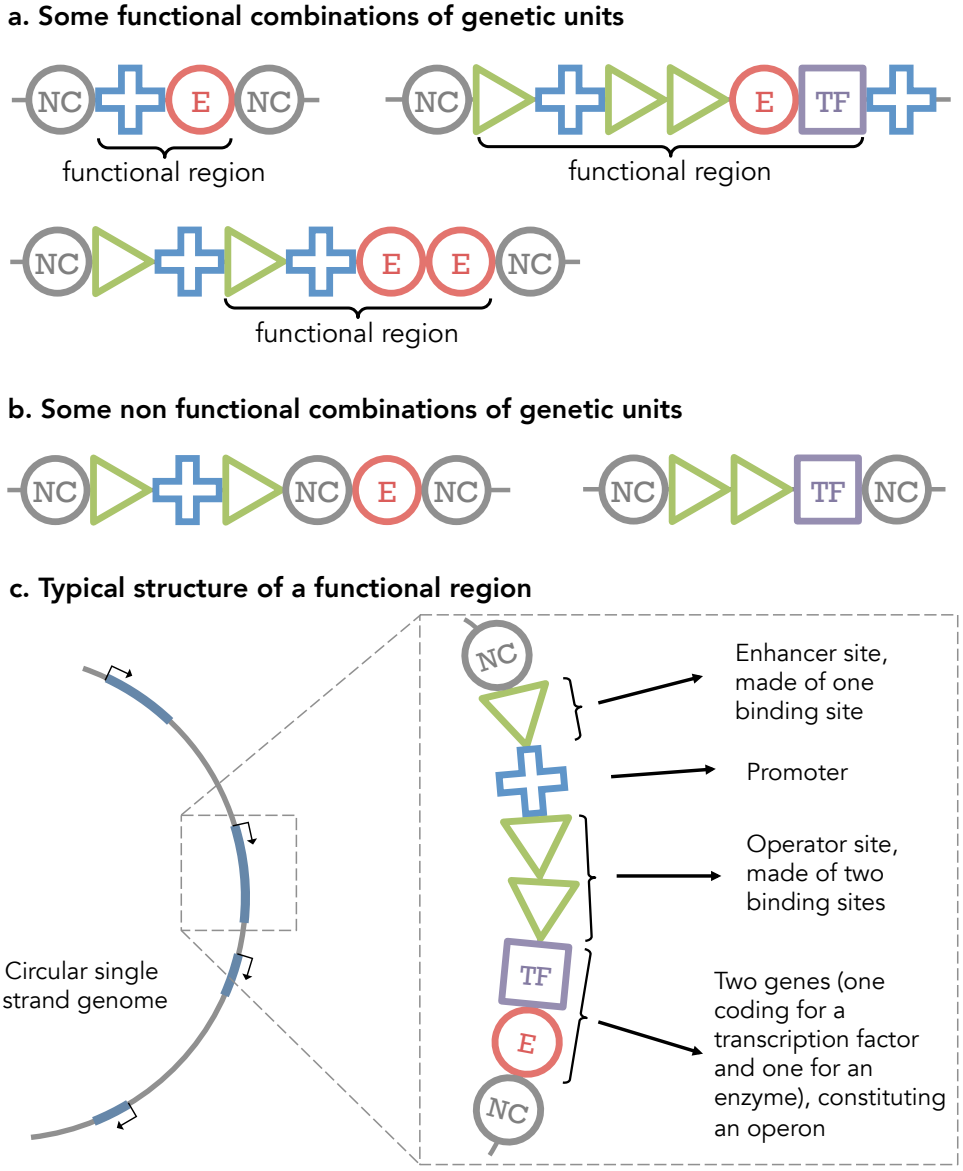


Figure III.3 – Some examples of arrangements of genetic units forming or non-functional functional regions. Grey circles: non-coding units (**NC**). Blue triangles: binding site units (**BS**). Orange crosses: promoter units (**P**). Purple squares: transcription factor coding units (**TF**). Magenta circles: enzyme coding units (**E**). The functional regions of a genome are those that have the following pattern: a promoter (**P**), optionally flanked upstream (enhancer site) or downstream (operator site) by one or more binding sites (**BS**), followed by one or more contiguous coding units (**E** or **TF**). **a.** Some functional combinations of genetic units. **b.** Some non functional combinations of genetic units. **c.** An example of the structure of a typical functional region. The genome is a circular single-strand genome with a single reading frame. A zoom is done in one functional region (magenta regions). The rest of the genome (in grey) is non-coding (non-coding units or any type of unit not correctly arranged to form a functional region).

III.2.2 Mutational operators

At each replication, the genome undergoes **point mutations** and **large rearrangements** (duplications, deletions, translocations and inversions). To account for the effects of these events on the coarse-grained genome, two additional types of mutation have been introduced in Evo^{2}Sim: **(i) transitions** : a genetic unit can transit from one type to any other at a certain rate, and **(ii) breakpoints**: during large rearrangements, genetic units located on sequence breakpoints are exposed to mutations.

(1) Point mutations. Point mutations modify the attributes of a genetic unit by adding random values to them. Each attribute (see Table III.1) owns a dedicated **mutation kernel** whose properties are predefined as model parameters (usually uniform or gaussian). Two types of attribute exist: integer values and floating-point values. Integer variables mutate by adding a random value from a uniform distribution. Floating-point variables mutate by adding a random value from a normal distribution. For example, the basal expression level β (a floating-point variable) mutation kernel is a normal law with a variance defined by the user at the beginning of the simulation. The substrate tag s (an integer value) mutation kernel is a uniform law with a range defined by the user at the beginning of the simulation. In summary, the parameters of eight mutation kernels have to be set by the user (see the Evo^{2}Sim user guide in Appendix A). Besides the parametrization of the mutation kernels, the **point mutation rate** must be set by the user. The point mutation rate is expressed per attribute per replication;

(2) Transitions. Genetic units can also undergo a type transition from any unit type to any other at a rate defined by the user. The transition rate is expressed per genetic unit per replication. All types of genetic units are actually implemented as a tuple containing all possible attributes, like (unit_type, $\beta, s, p, k_{cat}, k_{cat}/K_M$). The unit type tells us which parameters are functionally relevant and the others are free to mutate neutrally. Doing so, digital organisms can explore the neutral fitness landscape and potentially innovate if a non-coding unit is re-functionalized by a type transition (as it the case for pseudogenes);

(3) Duplications. Large duplications consist in duplicating a random sequence on the genome, and inserting the duplicate at a random location. To select the random sequence to duplicate, two random locations are uniformly drawn in the whole genome. The insertion point is also drawn uniformly (for example, it is possible to insert a duplicate in the duplicated sequence). A duplication implies one breakpoint (Fig. III.4a). The duplication rate is expressed per genetic unit per replication;

(3) Deletions. Large deletions consist in deleting a random sequence from the genome, and join the two extremities of the remaining sequence. To select the random sequence to delete, two random locations are uniformly drawn. A deletion implies two breakpoints (Fig. III.4b). The deletion rate is expressed per genetic unit per replication;

(4) Translocations. Large translocations consist in moving a random sequence from one genome location to another. To select the random sequence to move, two random locations are uniformly drawn. The insertion point is also drawn uniformly in the whole genome. A translocation implies three breakpoints (Fig. III.4c). The translocation rate is expressed per genetic unit per replication;

(5) Inversions. Large inversions consist in reverting a random sequence on the genome. To select the random sequence to revert, two random locations are uniformly drawn in the whole genome. An inversion implies two breakpoints (Fig. III.4d). The inversion rate is expressed per genetic unit per replication;

(6) Breakpoints. In real genomes, spontaneous rearrangement breakpoints have no reason to lie exactly between two functional regions and could thus break them. To model that with the coarse-grained genome representation, the content of the two genetic units that are adjacent to a rearrangement breakpoint is altered. Suppose for example that a deletion joins two genetic units, one containing the attributes (unit_type$_1$, β_1, s_1, p_1, k_{cat1}, $(k_{cat}/K_M)_1$) and the other the attributes (unit_type$_2$, β_2, s_2, p_2, k_{cat2}, $(k_{cat}/K_M)_2$). Then for each attribute, there is a probability for the value in unit 1 to be exchanged with the value in unit 2. Both units could for example exchange their values of s, thereby leading to (unit_type$_1$, β_1, s_2, p_1, k_{cat1}, $(k_{cat}/K_M)_1$) and (unit_type$_2$, β_2, s_1, p_2, k_{cat2}, $(k_{cat}/K_M)_2$). The breakpoint rate is expressed per breakpoint per replication, and must be set by the user.

III.3 The genetic regulatory network

When transcription factors are expressed, they can contribute to the genetic regulatory network by binding to functional enhancer or operator sites. At each time-step t and for each promoter i belonging to a functional region, four steps are necessary to compute the activity of the network:

(1) The activity $A_s(t)$ of each binding site s reads:

$$A_s(t) = \sum_j c_j(t) A_{js} \tag{III.1}$$

with $c_j(t)$ the concentration of the transcription factor j at time t and $A_{js} \in [0, 1]$ the affinity of this transcription factor for the binding site s. In the following, all the concentrations will be expressed in **arbitrary concentration units** (ACU). The affinity A_{js} depends on the distance between the transcription factor tag $\mathrm{TF}_{\mathrm{tag}}(j) \in \mathbb{Z}$ and the binding site tag $\mathrm{BS}_{\mathrm{tag}}(j) \in \mathbb{Z}$, and the binding window $W_{\mathrm{bind}}(j)$ of the transcription factor j. It reads:

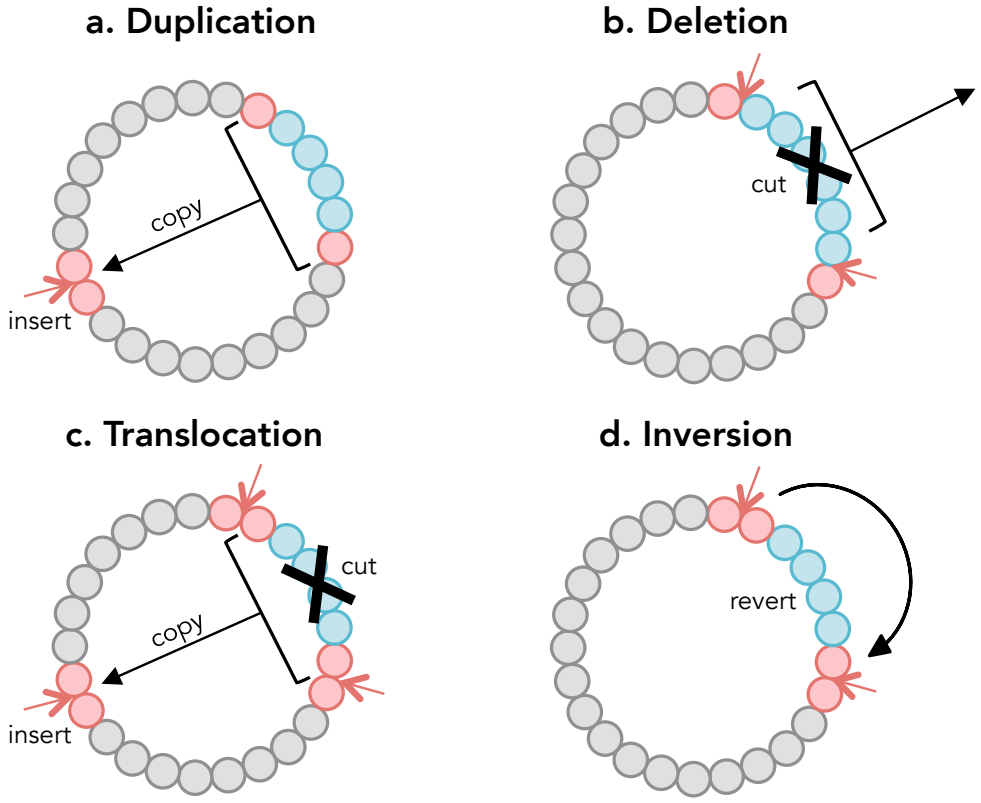


Figure III.4 – The four types of large rearrangements in Evo²Sim. At replication, genomes undergo four types of large rearrangements: **a.** duplications, **b.** deletions, **c.** translocations, and **d.** inversions. The genome sequence targeted by the rearrangement is colored in blue. Breakpoints are represented by red arrows. The genetic units undergoing mutations at breakpoints are colored in red.

$$A_{js} = \begin{cases} 1 - \dfrac{|\mathrm{TF}_{\mathrm{tag}}(j) - \mathrm{BS}_{\mathrm{tag}}(j)|}{W_{\mathrm{bind}}(j)} & \text{if} \quad |\mathrm{TF}_{\mathrm{tag}}(j) - \mathrm{BS}_{\mathrm{tag}}(j)| \leq W_{\mathrm{bind}}(j) \\ 0 & \text{else} \end{cases} \tag{III.2}$$

Figure III.5 shows the variation of the affinity when the distance between the transcription factor tag and the binding site tag varies.

(2) From **(1)**, the activities of the enhancer site $E_i(t) > 0$ and of the operator site $O_i(t) > 0$ flanking the promoter i read:

$$\begin{cases} E_i(t) &= \displaystyle\sum_{s \,\in\, \mathrm{enhancer}_i} A_s(t) \\ O_i(t) &= \displaystyle\sum_{s \,\in\, \mathrm{operator}_i} A_s(t) \end{cases} \tag{III.3}$$

(3) Then, the expression rate $e_i(t)$ of the promoter i is given by the following Hill-like function:

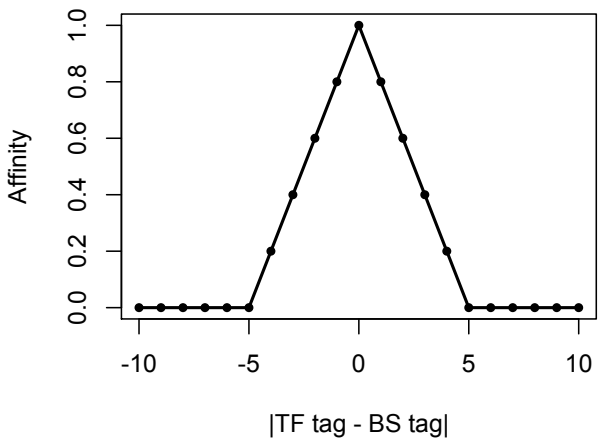


Figure III.5 – The affinity of a transcription factor for a binding site depends on the distance between their respective tags. On x-axis, the distance between the transcription factor tag and the binding site tag. On y-axis, the affinity computed thanks to Eq III.2. Here the binding window $W_{\text{bind}} = 5$.

$$e_i(t) = \beta_i . \left(\frac{\theta^n}{O_i(t)^n + \theta^n} \right) . \left(1 + \left(\frac{1}{\beta_i} - 1 \right) . \left(\frac{E_i(t)^n}{E_i(t)^n + \theta^n} \right) \right) \tag{III.4}$$

with $\beta_i \in [0, 1]$ the basal expression level of the promoter i, n and θ two constants shaping the Hill-like function (defined by the user).

(4) At each time-step t, coding units being controlled by the promoter i are expressed at a rate $e_i(t)$. Then, the concentration of each protein depending on the promoter i in the cytoplasm depends on the following synthesis-degradation rule:

$$\begin{cases} c_i(0) = \dfrac{\beta_i}{\phi} \\ \dfrac{\partial c_i}{\partial t} = e_i(t) - \phi . c_i(t) \end{cases} \tag{III.5}$$

with ϕ the protein degradation rate, set by the user before the beginning of any simulation.

III.4 The metabolic network

Enzyme coding units products can either be pumps, pumping metabolites from or to the growth medium, or enzymes performing catalytic transformations in the metabolic space.

Let us consider an enzyme in the cytoplasm that catalyzes one specific reaction $s \rightarrow p$ (with $s \in \mathbb{N}^*$ and $p \in \mathbb{N}^*$ being the substrate and the product of a Michaelis-Menten-like

reaction, respectively). The variation in concentrations $[s]$ and $[p]$ over time are then driven by Eq III.6:

$$\begin{cases} \dfrac{d[s]}{dt} = -\dfrac{k_{cat}[E][s]}{K_M + [s]} \\ \\ \dfrac{d[p]}{dt} = \dfrac{k_{cat}[E][s]}{K_M + [s]} \end{cases} \tag{III.6}$$

where K_M and k_{cat} are the kinetic attributes of the enzyme (K_M being deduced from k_{cat} and k_{cat}/K_M attributes).

Is $s = p$, enzymes are treated as pumps, for which $[s]$ and $[p]$ describe the internal and external concentrations of the same metabolite. If k_{cat} is positive (resp. negative), $[s]$ is the external (resp. internal) concentration of the metabolite and $[p]$ the internal (resp. external) concentration. The dynamics of metabolic concentrations $[s]$ and $[p]$ are thus also driven by Eq III.6 when the enzyme coding unit product is a pump.

III.5 Coupling the genetic regulatory network and the metabolic network

Bacteria are able to sense their environment by detecting the presence of a particular molecule or signal, and to give an appropriate answer by updating their gene expression profile. The archetype of this behaviour is the **lactose operon** (Jacob and Monod, 1961).

As shown in Figure III.6, this operon is composed of three genes (*lacZ*, *lacY* and *lacA*), controlled by one promoter flanked by an operator. Another gene, *lacI*, codes for a transcription factor which inhibates the operon when binding on the operator. *lacI* is constitutively expressed and its concentration in the cytoplasm is almost constant. However its conformation, hence its affinity for the operator is modified by lactose. In absence of lactose, *lacI* is active and down-regulates the operon. If lactose is present, it binds on *lacI* and inhibits it. In this case, the operon is expressed and the cell is able to degrade lactose.

This mechanism is integrated to Evo^{2}Sim: some metabolites can behave as co-enzymes, and repress or activate transcription factors activity. To this aim, each transcription factor own a co-enzyme tag $\text{CoE}_{\text{tag}} \in \mathbb{N}^*$, a co-enzyme constant k_{CoE}, a free activity A_{free} and a bound activity A_{bound}. A metabolite m can repress or activate a transcription factor in four ways:

(1) If $A_{\text{free}} = 1$ and $A_{\text{bound}} = 0$, m inhibits the transcription factor;

(2) If $A_{\text{free}} = 0$ and $A_{\text{bound}} = 1$, m activates the transcription factor;

(3) If $A_{\text{free}} = 1$ and $A_{\text{bound}} = 1$, the transcription factor is always activated;

(4) If $A_{\text{free}} = 0$ and $A_{\text{bound}} = 0$, the transcription factor is always repressed.

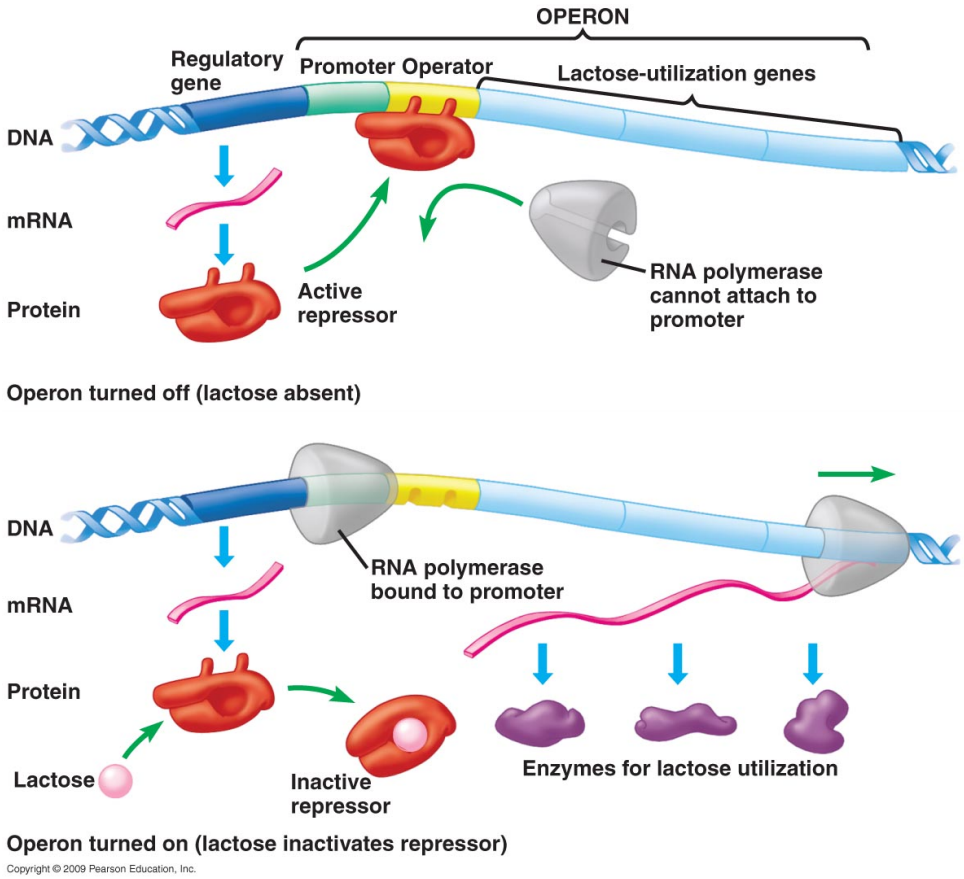
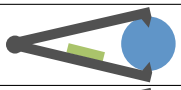
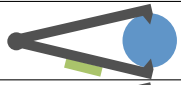
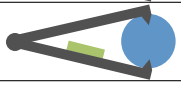


Figure III.6 – The lactose operon. The lactose operon is inactive in the absence of lactose (top) because a repressor blocks attachment of RNA polymerase to the promoter. With lactose present (bottom), the repressor is inactivated, and transcription of lactose-processing genes proceeds (from https://2fsfox.blogspot.fr/2013/05/the-lac-operon-continued-and-other.html).

Table III.2 resumes these different outcomes by using the following picture: let's consider a transcription factor as a structure with two arms, linked by a pivotal point. The active site of the transcription factor is located on one arm, its exposure depending on the equilibrium state (or conformation) of the structure. Two configurations are possible: one when the transcription factor is free, and another when a co-enzyme binds to it via the anchoring points located at arms end. The combination of free and bound activities then leads to four behaviors, as described in Equation III.7.

$$[\mathrm{TF}_+] = \begin{cases} [\mathrm{TF}] \times \dfrac{k_{\mathrm{CoE}}}{k_{\mathrm{CoE}} + [m]} & \text{if} \quad A_{free} = 1 \text{ and } A_{bound} = 0 \\ [\mathrm{TF}] \times \dfrac{[m]}{k_{\mathrm{CoE}} + [m]} & \text{if} \quad A_{free} = 0 \text{ and } A_{bound} = 1 \\ [\mathrm{TF}] & \text{if} \quad A_{free} = 1 \text{ and } A_{bound} = 1 \\ 0.0 & \text{if} \quad A_{free} = 0 \text{ and } A_{bound} = 0 \end{cases} \tag{III.7}$$

Table III.2 – The eight possible states of a transcription factor. The transcription factor is represented in dark grey. Its active site (the part allowing binding on a binding site) is represented in green. Depending on free and bound activities attributes, the co-enzyme (in blue) acts as an activator or a repressor. The active site is then free (or not) to bind on a binding site.

Free TF	Bound TF	Free activity	Bound activity
		*1*	*0*
		*0*	*1*
		*1*	*1*
		*0*	*0*

III.6 Optional feature: energy constraints

One of the most evident constraints living organisms must cope with in the real world are the laws of thermodynamics. Indeed, real organisms cannot violate the energy balance with their environment, or have negative entropy. One direct consequence is that global entropy cannot decrease, whatever the organism's activity. For example, catabolic reactions produce heat, that will propagate in the local environment of the organism (and possibly kill it). This energy is lost for the organism. In this sense, life could be seen as a fight against entropy (Alberts et al., 2013), as illustrated in Figure III.7. Billions years of evolution made cells very efficient engines to exploit the energy gained with catabolism. Energy carriers, like ATP molecules, allow cells to transfer the energy won by degrading food, or capturing photons, in useful but costly reactions (for example, producing—or actively degrading—a protein). This coupling between food process (catabolism), and production of useful macromolecules (anabolism) is at the heart of cell's metabolism.

We introduced energy constraints in Evo^{2}Sim by doing the distinction between two types

of reactions: reactions rewarding the cell in energy (catabolic reactions), and reactions consuming energy (anabolic reactions). Implicit energy carrier molecules allow us to compute the **energy balance** $\mathcal{E}$ of the cell at each time-step t. To this aim, we introduced a notion of reaction cost, each type of reaction owning a specific cost defined by the user before the beginning of a simulation. There are four energy costs:

(1) The **expression cost** $c_{\text{expr}} \geq 0$: the cell consumes energy when proteins are expressed;

(2) The **degradation cost** $c_{\text{degr}} \geq 0$: the cell consumes energy when proteins are degraded (symbolizing, *e.g.*, the functioning of the proteasome);

(3) The **enzymatic cost** $c_{\text{enz}} \geq 0$: depending on the type of metabolic reaction (see below), the cell consumes or produces energy when an enzymatic reaction is performed;

(4) The **pumping cost** $c_{\text{pump}} \geq 0$: the cell consumes energy when a metabolite is pumped in or out;

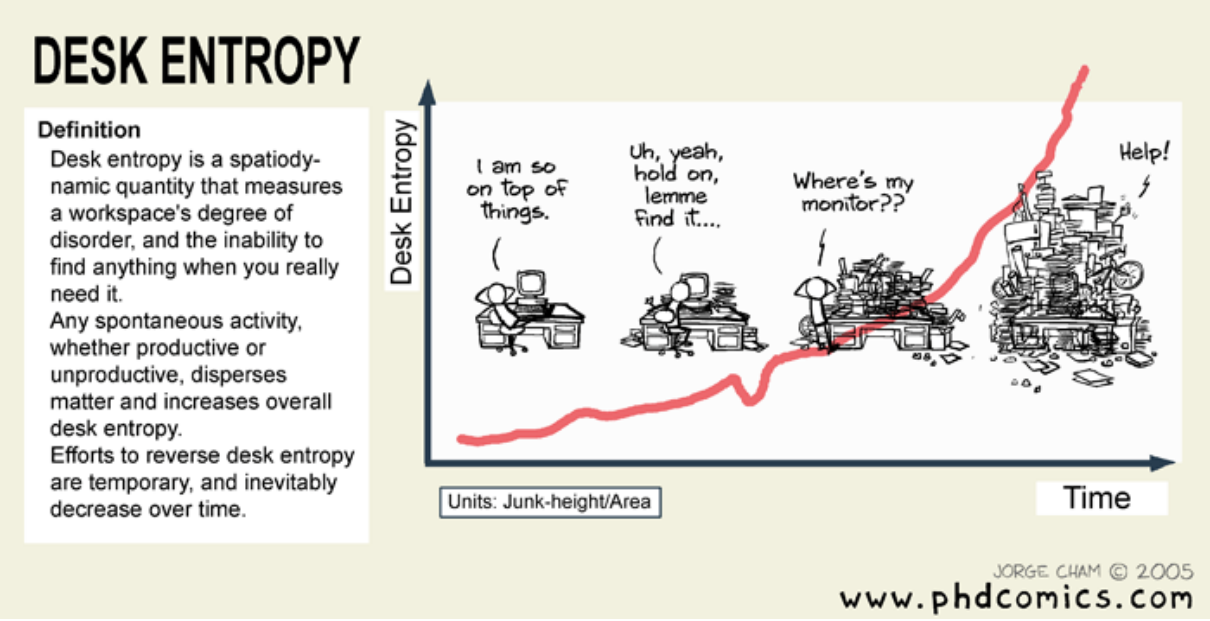


Figure III.7 – An impossible-to-win fight against entropy. An illustration of the unavoidable increase of entropy in a system (from www.phdcomics.com).

Enzymatic reactions consume or produce energy depending on the values of the substrate tag s, the product tag p and the enzymatic cost c_{enz}:

(1) if $s < p$, the reaction consumes energy at a rate $(p - s).c_{\text{enz}}$

(2) if $s > p$, the reaction produces energy at a rate $(s - p).c_{\text{enz}}$

It is then possible to describe the evolution of the energy balance $\mathcal{E}$ through time, as following:

$$\begin{aligned}
\frac{\partial \mathcal{E}}{\partial t} &= \sum_{i \in \text{ catabolic reactions}} \left(\frac{\partial [p_i]}{\partial t} \times (s_i - p_i) \times c_{\text{enz}} \right) \\
&- \sum_{i \in \text{ anabolic reactions}} \left(\frac{\partial [p_i]}{\partial t} \times (p_i - s_i) \times c_{\text{enz}} \right) \\
&- \sum_{i \in \text{ inflowing pumps}} \left(\frac{\partial [s_{in}]}{\partial t} \times (s_{in} - s_{out}) \times c_{\text{pump}} \right) \\
&- \sum_{i \in \text{ outflowing pumps}} \left(\frac{\partial [s_{out}]}{\partial t} \times (s_{out} - s_{in}) \times c_{\text{pump}} \right) \\
&- \sum_{i \in \text{ expressed genes}} \left(\frac{\partial [e_i]}{\partial t} \times c_{\text{expr}} \right) \\
&- \sum_{i \in \text{ degraded proteins}} ([c_i] \times \phi \times c_{\text{degr}})
\end{aligned} \tag{III.8}$$

For practical reasons, $\mathcal{E}$ is not solved as an ordinary differential equation. Indeed, incorporating energy in differential equations would have lead to intractable simulations. For this reason, the energy balance $\mathcal{E}$ is evaluated at the end of each simulation time-step t. The cell's score is impaired if the energy balance $\mathcal{E} \leq 0$ (the score function is described below).

III.7 The score function

The set of all metabolites contained in the cell's cytoplasm can be converted into a unique concentration vector $\boldsymbol{M} = \{m_1, m_2, ..., m_n\}$. In EVO^{2}SIM, $\boldsymbol{M}$ constitutes the "phenotype" determining the score S of the cell. It is then possible to define a score function $f : \mathbb{R}^n \rightarrow \mathbb{R}_+$ such that $S = f(\boldsymbol{M})$.

Some metabolites are essential to cell's growth, and some other are intermediate products or waste. In EVO^{2}SIM, essential metabolites are **prime numbers**: their production contributes to the growth rate by increasing the probability to produce offspring. However, producing metabolites above a predefined threshold leads to cell's toxicity, and impairs cell's score. Let's define the subset of $\boldsymbol{M}$ representing the essential metabolites $\boldsymbol{E} =$

$\{e_1, e_2, ..., e_m\} \subset \boldsymbol{M}$. Then, the score S of the cell is:

$$S = \begin{cases} \sum_i e_i & \text{if} \quad (\forall e \in \boldsymbol{E} \mid e < T_E) \cup (\forall m \in (\boldsymbol{M} \backslash \boldsymbol{E}) \mid m < T_M) \\ 0 & \text{else} \end{cases} \tag{III.9}$$

with T_M the toxicity threshold of non-essential metabolites ($\boldsymbol{M} \backslash \boldsymbol{E}$) and T_E the toxicity threshold of essential metabolites ($\boldsymbol{E}$).

If the cell's score is under a minimum score $S < S_{\min}$ defined by the user, then $S = 0$.

Importantly, the score and the fitness are different. The score represents the instantaneous performance of a cell (being computed at each time-step), while the fitness is usually defined as the combined effect of survival and reproduction, and can only be computed *a posteriori*, when the whole cell's history is known. We will never compute the fitness in Evo²Sim. Instead, we will analyze the lineage of the final population, that is supposed to have increased its mean fitness through time, in order to recover evolutionary events.

III.8 Population and selection

Organisms evolve on a two-dimensional toroidal grid (each location containing at most one organism), and compete for the external metabolites to produce offspring in empty locations. They interact with their local environment by pumping metabolites in and out and they release their metabolic content at death. At each time-step, organisms are evaluated and either killed, updated or replicated depending on their current state:

(1) If the organism does not die and cannot divide (*e.g.*, because there is no free space in its neighborhood), its metabolic network is updated, and its score is computed;

(2) Organisms can also die randomly with a probability following a Poisson law of parameter p_{death} expressed per organism per time-step. At death, the metabolic content of a cell is released into the local environment;

(3) For each empty grid location, all living organisms in the Moore neighborhood whose score is higher than a minimum score S_{min} compete. The organism having the best score in the neighborhood is allowed to divide if it did not replicate previously at the same time-step (such that any dividing cell generates at most two daughters per time-step).

III.9 The environment

The physical environment is described at the grid level: each grid location contains external metabolites, each with its concentration. These external metabolites diffuse with

a diffusion parameter D expressed in gridstep^2.time-step^{-1}, meaning that a fraction D of each metabolite present at one location will diffuse to each of the eight neighboring grid locations at each time-step. The discrete diffusion equation we are using is inspired from Frénoy et al. (2013). External metabolites are also degraded with a degradation rate D_g, meaning that a fraction D_g of each metabolite at each location will disappear at each time-step. We made the simplifying assumption that there are no enzymatic reaction in the environment, and that metabolite transformation only occurs inside the organisms.

At each time-step t, each grid location k of coordinates (x, y) is characterized by the individual occupying the location (possibly none), and the list of free metabolites, each metabolite i being at concentration $c_{i,k}(t)$. Given the parameters of the environment, the dynamics of a free metabolite i in a grid location k reads:

$$c_{i,k}(t+1) = c_{i,k}(t) - D_g.c_{i,k}(t) + \sum_{j \in \text{neighbors}} D.c_{i,j}(t) - 8.D.c_{i,k}(t) + I_i(t) \qquad \text{(III.10)}$$

With I_i the inflow rate of metabolite i in the environment.

In conclusion, EVO^{2}SIM allows for a precise parametrization of the environment. Apart from parameters described above, it is possible for the user to set a variety of behaviors (*e.g.* the periodicity of metabolites influx, the type of metabolites provided or their locations, ...). It is thus possible to mimic realistic experimental setups, such as chemostat or batch-culture, as we will discuss in the next chapter.

III.10 Trophic networks

Cells uptake various metabolites, provided externally or being by-products released by other cells. EVO^{2}SIM keeps trace of the metabolic activity of every cells and computes, at each time-step, a **trophic network** representing the relationships between cells. This feature is mandatory to study, for example, the evolution of cross-feeding in the population.

At each time-step t, a **trophic profile** is computed for each organism from its metabolic network activity. The trophic profile is a bit string summarizing the uptake, production, and release activity of an organism. The length of the bit string is defined by the largest metabolite tag present in the system at time t. For example, if an organism uptakes metabolite #4, produces #3 from #4 and releases #3, knowing that the largest metabolite tag in the system is #5, then its profile is |00010|00100|00100|. Organisms with identical trophic profiles are grouped together, and the trophic network is computed depending on profile relationships. For example, if organisms of a profile i uptake a metabolite produced by a profile j, then a directed link is created from i to j. Cooperating links are also computed: a cell cooperates with another cell if the former **actively** releases metabolites useful to the latter.

Trophic profiles are then classified in four **trophic levels**:

(1) Level 0 cells exclusively feed on exogenous metabolites, flowing in the environment;

(2) Level 1 cells feed on exogenous metabolites and on metabolites produced by other cells;

(3) Level 2 cells exclusively feed on metabolites produced by other cells;

(4) No level cells have no active uptaking functions.

Figure III.8 shows a simple example of trophic network computed on the fly during a simulation, and available in Evo²Sim HTML viewer (see Appendix A). Exogenously provided metabolites are symbolized by a black node (the `ENV` node), and other trophic profile nodes are colored depending on their level (purple for level 0, blue for level 1, green for level 2 and grey for no level). Trophic links are represented by solid arrows, cooperating links being represented by dashed arrows.

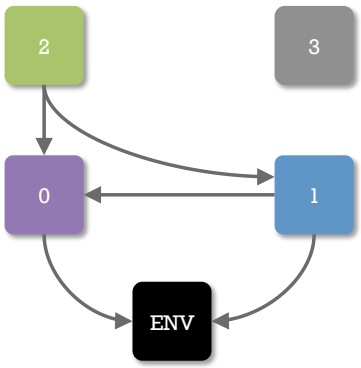


Figure III.8 – A basic example of trophic network. A basic example of a trophic network, as it is computed in Evo²Sim, is presented here. Exogenously provided metabolites are represented by a black node tagged `ENV`. Trophic profiles (*i.e.*, a group of cells having the exact same metabolic activity) exclusively feeding on exogenous metabolites belong to the level 0, and are represented by purple nodes. Trophic profiles feeding on exogenous metabolites and on by-products of other cells belong to level 1 ; they are represented by blue nodes. Trophic profiles exclusively feeding on by-products are represented by green nodes (level 2). Trophic profiles having no metabolic activity are represented by grey nodes. Here, one level 0 profile feeds on the environment, one level 1 profile feeds on the environment and on profile 0, one level 2 profile feeds on profiles 0 and 1, and one profile has no metabolic activity (no level).

III.11 Lineages and phylogeny

In Evo²Sim, phylogenetic relationships are exhaustively recorded during a simulation. Two trees are updated at each time-step: the **lineage tree**, that saves the lineage relationships of every living cells, and the **phylogenetic tree**, that saves the complete phylogeny of every living cells. Besides phylogenetic relationships, many informations about the genome structure, the phenotype, the mutations, the trophic profile, and so on,

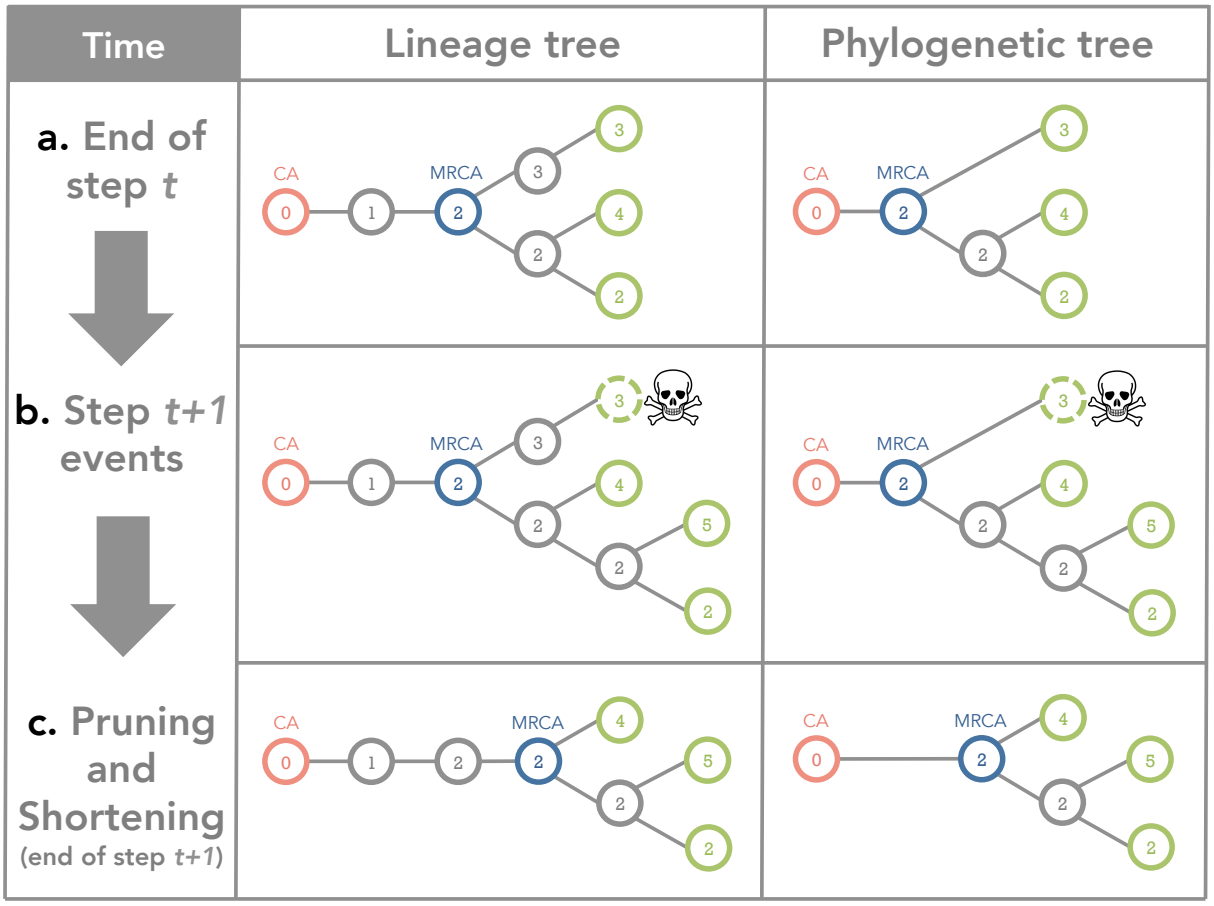


Figure III.9 – Live update of lineage and phylogenetic trees. At each time-step t, the population state is updated (divisions, deaths, cell updates, ...): **(i)** at each division, the two daughter cells are added to the trees as leaves, with their parent as a common ancestor, **(ii)** dead cells are removed from trees. Both lineage and phylogenetic trees are pruned (dead branches are removed), and the phylogenetic tree is shortened (intermediate nodes not being common ancestors are removed). **In this example**, we start at time t. The common ancestor of the whole population (**CA**, in red) is the **dead cell** labelled 0. The most recent common ancestor (**MRCA**, in blue) is the **alive cell** 2. Tree leaves are represented in green and all correspond to alive cells (first row). The population state is then updated to time $t+1$: the cell 3 dies, and the cell 2 divides in daughter cells 2 and 5 (the cell 2 is still tracked because it divided 4 times and didn't died yet). These events are added to both trees (second row). Then, **pruning** and **shortening** algorithms are applied: the lineage tree looses the branch $3-3$. The phylogenetic tree looses the leaf 3, and the oldest 2 node. The MRCA is now the node 2, linked to nodes 2 and 4.

are saved in every nodes of the trees. Thus, it is possible to precisely recover the evolution of a population, including fixed mutations. In particular, it is possible to determine if trophic groups are **monophyletic**, and thus can be considered as **ecotypes** (see the next chapter for a precise example).

Algorithmically speaking, the phylogenetic tracking deployed in EVO^{2}SIM is updated as follows: at each simulation time-step t, **(i)** new offspring are added to both trees (Fig. III.9b), **(ii)** both trees are pruned to remove dead branches (Fig. III.9c), and **(iii)** the

phylogenetic tree is shortened to remove intermediate nodes between common ancestors (Fig. III.9c). One node in the lineage or phylogenetic tree corresponds to one generation in the population. This means that when a cell produces offspring, new nodes are created for the two daughters, even if each cell is individually tracked for its entire life. For example, two or more contiguous nodes in a tree could correspond to a single cell that divided one or more times, as shown in Figure III.9 with cell #2. In each tree, the common ancestor (CA) of the whole population is tagged (red node on Fig. III.9), as well as the most recent common ancestor (MRCA, blue node on Fig. III.9).

III.12 General algorithm

The general algorithm behind Evo²Sim is a classical, asynchronous algorithm of *in silico* evolution (Hindré et al., 2012). At each time-step, each living cell is evaluated, and a decision is made between death, division, or simple update (as presented in Fig. III.10). The lineage and phylogenetic trees are updated on the fly, as well as the trophic network. In the same time, very complete statistics are computed (tracking hundred of variables), and a large amount of statistics (population means, best individual lineage, phylogeny, ...) are displayed on the fly, in the Evo²Sim HTML viewer (see Appendix A).

To solve the ordinary differential equations, We used the adaptive Runge-Kutta-Cash-Karp method (RKCK). At each simulation time-step t and for each alive cell, the state of the genetic regulatory network and the metabolic network are updated by solving the ODE system during t_{ODE} time-steps. This constant is set by the user before the beginning of the simulation (usually $t_{\text{ODE}} = 100$, meaning that each simulation time-step corresponds to 100 ODE time-steps). Altogether, if we consider a 32×32 environmental grid full of organisms, a time-step involves a thousand of ODE systems. The parameter values of each ODE system are potentially unique, as they are encoded in the organism's genome and thus result from the mutation process. Those ODE systems can also differ by their number of equations, which depends on the organism's genome.

Moreover, Evo²Sim admits parallel computing algorithms, and is designed for high performance computing (see the user guide in Appendix A).

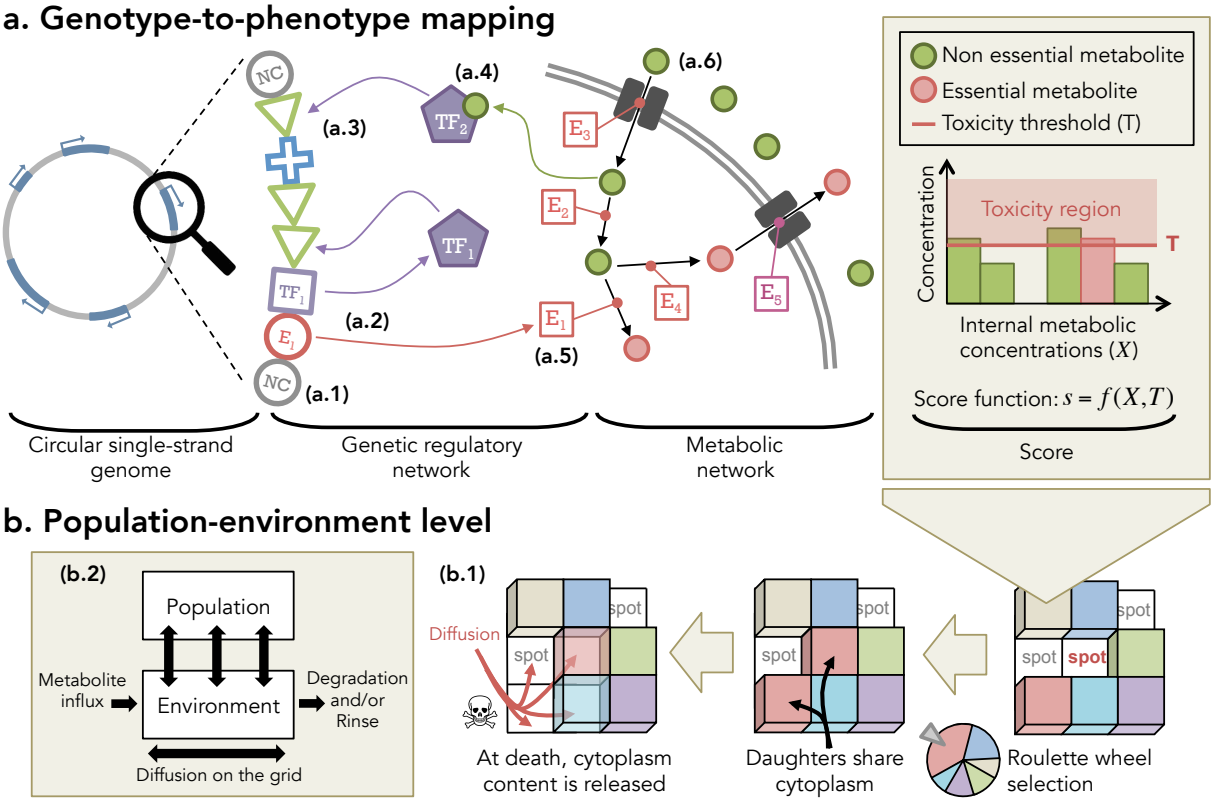


Figure III.10 – Global picture of Evo²Sim. a. Description of the genotype-to-phenotype mapping. Organisms own a coarse-grained genome made of units. This genome is a circular single-strand sequence, with a unique reading frame. Non coding **(NC)** units are not functional **(a.1)**. The arrangement of the units on the sequence defines functional regions, where a promoter (**P**, blue cross) controls the expression of enzyme coding units (**E**, red circles) or transcription factor coding units (**TF**, purple squares), thereby allowing for operons (here, one E and one TF). When coding units are expressed **(a.2)**, they contribute to the genetic regulatory network (for TFs) and the metabolic network (for Es). Depending on their attributes (see III.2 and III.3), transcription factors bind on binding sites. **(a.3)** If they bind on the enhancer sequence (binding sites flanking the promoter upstream), the promoter activity is up-regulated. If they bind on the operator sequence (binding sites flanking the promoter downstream), the promoter activity is down-regulated. **(a.4)** Metabolites can bind on a transcription factor as co-enzymes, and activate or inhibit it, depending on transcription factor attributes (see III.5). Enzymes perform metabolic reactions in the cytoplasm **(a.5)**, or pump metabolites in or out **(a.6)**. The score of an organism is computed from its "essential metabolites" (usually the score is the sum of essential metabolite concentrations). Lethal toxicity thresholds are applied to each metabolic concentration and forbid organisms to accumulate resources. **b. Description of the population and environment levels.** Organisms are placed on a 2D toroidal grid, and compete for resources and space. When an organism dies, it leaves its grid cell empty and organisms in the Moore neighborhood (if any) compete to divide in available space. The competition is based on scores, a minimal threshold being applied on scores to forbid worst organisms to divide. At division, daughters share cytoplasm content (enzymes and metabolites). At death, metabolites from the cytoplasm are released in the local environment, and diffuse on the grid **(b.1)**. On the largest scale, the population evolves on the environment by up-taking, transforming and releasing metabolites. Metabolites then diffuse and are degraded. This strong interaction between the population and the environment allows for the evolution of complex ecological situations, depending on environmental properties **(b.2)**.

III.13 Code availability

We developed Evo²Sim in C++, from scratch. Some scripts are written in Python and R, especially for the automatic generation of live statistics and figures. We also implemented a HTML viewer, including many informations (from best lineage evolution to phylogeny), useful to track evolution during a simulation. This viewer includes some Javascript. The code is hosted on Github in `charlesrocabert/Evo2Sim` repository. The Evo²Sim user manual is available in Appendix A. Some simulation examples are also available on the EvoEvo project website at http://evoevo.liris.cnrs.fr/evo2sim/.

III.14 What next?

In the two following chapters, we will present two results obtained with Evo²Sim. The first has been published in PLoS Computational Biology, and is about niche construction and the evolution of stable cross-feeding. This work does not consider genetic regulation. For this reason, a simplified version of Evo²Sim will be presented. The second result is preliminary and is about genetic regulation evolution when energy constraints are applied to digital organisms.

Chapter IV

Beware Batch Culture: Seasonality and Niche Construction Predicted to Favor Bacterial Adaptive Diversification

The results presented in this chapter have been published in PLoS Computational Biology.

No one wins. One side just loses more slowly.
(Roland "Prez" Pryzbylewski – No refugees, The Wire)

Abstract

Metabolic cross-feeding interactions between microbial strains are common in nature, and emerge during evolution experiments in the laboratory, even in homogeneous environments providing a single carbon source. In sympatry, when the environment is well-mixed, the reasons why emerging cross-feeding interactions may sometimes become stable and lead to monophyletic genotypic clusters occupying specific niches, named ecotypes, remain unclear. As an alternative to evolution experiments in the laboratory, we developed Evo^{2}Sim, a multi-scale model of *in silico* experimental evolution, equipped with the whole tool case of experimental setups, competition assays, phylogenetic analysis, and, most importantly, allowing for evolvable ecological interactions. Digital organisms with an evolvable genome structure encoding an evolvable metabolic network evolved for tens of thousands of generations in environments mimicking the dynamics of real controlled environments, including chemostat or batch culture providing a single limiting resource. We show here that the evolution of stable cross-feeding interactions requires seasonal batch conditions. In this case, adaptive diversification events result in two stably co-existing ecotypes, with one feeding on the primary resource and the other on by-products. We show that the regularity of serial transfers is essential for the maintenance of the polymorphism, as it allows for at least two stable seasons and thus two temporal niches. A first season is externally generated by the transfer into fresh medium, while a second one is internally generated by niche construction as the provided nutrient is replaced by secreted by-products derived from bacterial growth. In chemostat conditions, even if cross-feeding interactions emerge, they are not stable on the long-term because fitter mutants eventually invade the whole population. We also show that the long-term evolution of the two stable ecotypes leads to character displacement, at the level of the metabolic network but also of the genome structure. This difference of genome structure between both ecotypes impacts the stability of the cross-feeding interaction, when the population is propagated in chemostat conditions. This study shows the crucial role played by seasonality in temporal niche partitioning and in promoting cross-feeding subgroups into stable ecotypes, a premise to sympatric speciation.

IV.1 Introduction

Stable metabolic cross-feeding interactions between microbial strains are commonly observed in nature (Stams, 1994; Dejonghe et al., 2003; Costa et al., 2006; Katsuyama et al., 2009). For example, nitrification, an important step of the nitrogen cycle, is carried out in consecutive steps by several bacterial species maintaining cross-feeding interactions

(Costa et al., 2006). In laboratory experiments, microbial populations also demonstrated their ability to quickly establish metabolic cross-feeding interactions between morphotypes (Rainey and Travisano, 1998; Rainey and Rainey, 2003; Helling et al., 1987; Rosenzweig et al., 1994; Turner et al., 1996; Treves et al., 1998; Rozen and Lenski, 2000; Rozen et al., 2005, 2009).

An important question, at the crossroads between ecology and evolution, is the evolutionary stability of such cross-feeding polymorphisms, because they are often considered to be the first steps toward speciation. According to Cohan (2002), the species concept in bacteria should not rely on the named species of systematics but on the notion of *ecotype*, which itself relies on the ecological and evolutionary dynamics of the subpopulations. Two bacterial subpopulations may be considered as different ecotypes if they form monophyletic clusters, occupy different ecological niches and if periodic selection purges diversity in one subpopulation independently from the other (Cohan, 2002). A cross-feeding polymorphism therefore leads to adaptive diversification and ultimately to speciation when it is stable enough to resist the invasion of a mutant that would otherwise take over the whole population.

If the environment is spatially structured, the stabilization of new ecotypes that emerged after an adaptive diversification event is facilitated by the locality of environmental conditions and frequency-dependent interactions. This mechanism of allopatric (or micro-allopatric) divergence is well-known, since ecotypes can escape competitive exclusion in their local niches (Cohan, 2002). For example, *Pseudomonas fluorescens* populations have been shown to produce adaptive diversification events in spatially heterogeneous environments, but not in homogenized conditions (Rainey and Travisano, 1998; Rainey and Rainey, 2003).

Microbial populations can also exhibit adaptive diversification in sympatry, when the environment is homogeneous with a single carbon source. In this case, the stability of ecotypes is maintained by frequency-dependent interactions, often due to cross-feeding interactions, as observed in the Long-Term Evolution Experiment with *Escherichia coli* (LTEE) (Elena and Lenski, 2003). In this ongoing experiment, 12 populations are being independently propagated in a constant glucose-limited environment in batch culture since 1988. The experiment reached 66,000 generations at the time of this writing. Every day, 1% of the population is transferred into fresh medium such that each population experiences a daily cycle of feast and famine phases. In one of the 12 populations, a long-term polymorphism has been observed (Rozen and Lenski, 2000). Two ecotypes, named S and L (for Small and Large, related to their respective colony sizes on plate), evolved from a common ancestor before generation 6,500. The L ecotype grows efficiently on glucose, while the S ecotype mainly grows on acetate, a by-product secreted by L (Großkopf et al., 2016). Experiments showed that the interaction between S and L ecotypes relies on negative frequency-dependent selection, each ecotype having a selective advantage when rare. This balanced polymorphism is now stable for more than 55,000 generations (Rozen and Lenski, 2000). It was also shown that S and L ecotypes specialized in their own niches, the L ecotype increasing its ability to grow on glucose but not on acetate, and conversely for the S ecotype (Großkopf et al., 2016).

The evolutionary stability of this polymorphism may be explained by the temporal niche partitioning that arises from the periodic transfers into fresh medium (Spencer et al., 2007). A first season starts immediately after a transfer, when the environment contains mostly glucose. The L ecotype grows during this season, consumes glucose and secretes acetate, thereby generating a second season where the environment contains mostly acetate and supports the growth of the S ecotype.

Yet several experiments have shown that microbial populations can also evolve cross-feeding interactions in a chemostat in a few tens of generations. Those interactions appear to be stable over a few hundreds of generations (Helling et al., 1987; Rosenzweig et al., 1994; Treves et al., 1998). In chemostat, there is no obvious spatial or temporal niche partitioning and it is thus intriguing that the dynamics predicted by the competitive exclusion principle has not been observed so far. Indeed, one would expect a mutant to eventually appear, which would either completely degrade glucose or feed on both glucose and acetate, thereby outcompeting the specialized ecotypes. It has been proposed that energy constraints and flux optimization principles prevent competitive exclusion, thereby stabilizing the polymorphism (Pfeiffer and Bonhoeffer, 2004; Gerlee and Lundh, 2010b). However, experimental evolution in chemostat has generally been performed for only a few hundreds of generations (up to 1,900 generations in Helling et al. (1987)), precluding the possibility to confirm this statement on a longer term.

Thus, as a step to better understand how cross-feeding, niche construction and seasonality contribute to microbial diversification, we addressed here the following question: What makes emerging cross-feeding interactions stable in the long-term, in single carbon source batch culture or chemostat experiments?

While experimental evolution provides a very precise picture of evolution, it remains a long and costly process. An alternative approach consists in simulating evolution in a computer. *In Silico* Experimental Evolution (ISEE), where digital organisms are evolved for tens of thousands of generations, reproduces the environmental conditions of experimental evolution (Hindré et al., 2012). Like in the wet approach, it is possible to simulate several independent populations to understand the respective importance of general laws and historical contingencies. In addition, ISEE provides an exhaustive fossil record and, more importantly, allows for "impossible experiments" (O'Neill, 2003), like saving the fitness at full resolution for tens of thousands of generations, or changing any parameter (mutation rates, environment fluxes) at will.

We developed Evo^{2}Sim, a multi-scale computational model of *in silico* experimental evolution. Evo^{2}Sim allows us to address many questions raised by experimental evolution (Hindré et al., 2012). Typically, we can use it to investigate how evolution shapes the different organization levels of an organism (*e.g.*, genome size, complexity of the regulation network and metabolic network) and of an ecosystem (polymorphism, speciation) depending on global parameters such as environmental conditions or mutation rates. Here, we tested which environmental conditions can lead to stable adaptive diversification events, by reproducing the resource dynamics of experimental evolution setups like chemostat and batch culture.

Previous mathematical works have already studied the conditions of interspecific coexistence via resource partitioning (Stewart and Levin, 1973), and of cross-feeding interactions (Doebeli, 2002; Pfeiffer and Bonhoeffer, 2004; Gudelj et al., 2016), during one or more competition episodes. Stewart and Levin (1973) studied the conditions of coexistence of several ecotypes in batch culture and chemostat. However, they focused on a single episode of competition between preexisting strains without modeling a random mutational process. Moreover, the strains were not allowed to cross-feed on by-products of other strains. Rozen et al. (2009) and Ribeck and Lenski (2015) modeled analytically the cross-feeding interaction between S and L ecotypes in the LTEE, showing the existence of negative frequency-dependence in batch conditions. These models also did not include a mutational process. Gudelj et al. (2016) studied the short-term dynamics of two competitors in various environmental conditions including batch and chemostat, and showed that stable cross-feeding was possible, depending on initial competitors frequency and resource abundance. Again, this model did not include the mutational process. Other mathematical studies introduced a simplified evolutionary dynamics, by computing successions of competition episodes and introduction of fit mutants. For example, Pfeiffer and Bonhoeffer (2004) studied the conditions of emergence of stable cross-feeding in chemostat conditions, when a trade-off on ATP production is introduced on abstract metabolic pathways. Doebeli (2002) compared the conditions of emergence of cross-feeding polymorphism in chemostat and batch culture. The authors concluded that the evolution of cross-feeding is more likely in chemostat than in batch culture. However, this model forced a trade-off between consumption rates of glucose and acetate, forbidding the emergence of a generalist mutant. Two rates are evolvable but only the glucose consumption rate is mutable, as the acetate rate is deduced from the glucose rate. The rate at which acetate is secreted is constant (*i.e.*, it does not depend on glucose consumption, which could affect the generality of the conclusions). Thus, none of the previous models take into account a realistic random mutational process, and none of them explicitly models the genomic level. Indeed, it is difficult to include a competition process as well as realistic mutational dynamics in a single mathematical model. Another approach consists in simulating evolution with individual-based models.

Computational models of *in silico* experimental evolution have already been used to explore the evolution of cross-feeding interactions. Johnson and Wilke (2004) used the Avida software (Ofria and Wilke, 2004) to study the evolution of resource competition between two digital species coexisting via mutualistic cross-feeding in a closed environment, with only two possible metabolites. However, they did not test the influence of the environmental dynamics. Williams and Lenton (2010) used an individual-based evolutionary model to explore the stability of connected ecosystems undergoing cross-feeding and "evolutionary regime shifts". Yet, the genotype-to-phenotype mapping of their organisms was rather simple (fixed size arrays defining the affinity of the organism for each resource), thus not allowing to study the effects of ecological dynamics on genome and metabolic network structures. Crombach and Hogeweg (2009) and Boyle et al. (2012) studied the evolution of resource cycling and its stability. In the first model (Crombach and Hogeweg, 2009), the resource cycling was imposed by the system. In the second model (Boyle et al., 2012), the environment was strongly structured (patches of individuals with random migration events), such that it was not possible to study sympatric diversification. Chow (2004)

used Avida (Ofria and Wilke, 2004) to explore the relation between productivity and diversity in a digital ecosystem under mixed influx of nine pre-defined resources, while Gerlee and Lundh (2010b) explained the maintenance of cross-feeding interactions in a microbial population by energy and efficiency constraints on metabolic fluxes. To do so, they developed an individual-based model evolving simple binary strings, thereby precluding evolvable interactions between the different organization levels of an organism, and their possible effects on the ecological dynamics. Gerlee and Lundh (2010a) also related ecosystem productivity to energy-uptake efficiency, with the same type of individual-based model as in Gerlee and Lundh (2010b). Recently, Liu and Sumpter (2017) used an individual-based model evolving artificial ecosystems relying on a "number soup": In this model, each species perform one modular addition transforming specific numbers into others, immediately available for other species. With their model, authors showed that artificial ecosystems always self-organize to consume all the available resources. While stable cross-feeding, and reciprocal cross-feeding, are common evolutionary outcomes in their model, authors also show that whole population extinctions sometimes occur, even without external perturbations. Yet, the absence of complex and evolvable genotype-to-phenotype map in their model precludes the possibility to get insights into the influence of ecosystem evolution on the structure of the organisms. Finally, Großkopf et al. (2016) predicted the adaptive diversification event leading to S and L ecotypes in the LTEE, by mixing flux balance analysis (FBA) and *in silico* evolution in a single model. By modeling the evolution of reaction rates in the metabolic network of *Escherichia coli*, they demonstrated that the emergence of a stable cross-feeding similar to S and L interaction is highly probable in the LTEE conditions. However, in their model, digital organisms are highly constrained (there is no innovation, *e.g.* new by-products cannot appear in the evolutionary process). To the best of our knowledge, none of these individual-based models compared the evolution of stable cross-feeding in different experimental setups, such as batch culture or chemostat.

To sum up, we were not able to find in the literature models that combine: **(i)** an explicit mutational process along with the modeling of natural selection and drift, **(ii)** evolvability at all organization levels (genome structure, metabolic network, number of reactions, number of metabolites, reaction rates, ...), and **(iii)** a comparison between batch culture and chemostat.

Our results show that *stable* cross-feeding interactions are favored in batch culture, owing to the seasonality of the environment. In continuous culture, the absence of seasonality precludes niche construction and leads to competitive exclusion, even if the population is initially composed of two ecotypes maintaining frequency-dependent interactions. We also demonstrate that the long-term evolution of a stable cross-feeding interaction in batch culture leads to character displacement (Legac et al., 2012; Großkopf et al., 2016), at the level of the metabolic network but also of the genome structure. This difference of genome structure between the two ecotypes has an impact on the further stability of the cross-feeding interaction when the population is propagated into continuous culture.

IV.2 Model

EVO^{2}SIM is a multi-scale and individual-based computational model. Digital bacterial-like organisms own a coarse-grained genome that contains genomic units encoding a simplified metabolic network. The organisms evolve on a two-dimensional toroidal grid (the environment), uptaking, transforming and releasing metabolites, and dividing in the presence of empty spots or dying. Extracellular metabolites diffuse across the grid spots. In this model, metabolites are implicit molecules identified by a tag $\in \mathbb{N}^*$. The model is described in more details below, and summarized in Figure IV.1. The source code is written in C++. All the material necessary to replay experiments (software, parameter files, strain backups, ...) is freely available at http://www.evoevo.eu/adaptive-diversification-simulations/. The latest version of EVO^{2}SIM is available on Github in `charlesrocabert/Evo2Sim` repository.

IV.2.1 Genome structure

The genome is a circular single-stranded sequence of genomic units, inspired from Crombach and Hogeweg (2008) and Beslon et al. (2010b). Genomic units belong to three different types: non-coding units (NC), promoter units (P), and enzyme coding units (E). The order of the units in the genome determines the existence of functional regions, meaning that not all sequences of units are functional. The functional regions of a genome are those that have the following pattern: a promoter (P) followed by one or more enzyme coding units (E). A promoter can thus control several coding units, as bacterial operons. The first genomic unit that is not enzyme coding interrupts transcription and marks the end of the functional region.

Non-coding units (NC) have no particular function. They constitute the non-coding part of the genome. Promoter units (P) contain a floating-point number $\beta \in [0.0, 1.0]$ representing the production rate of the protein(s) depending on the promoter. All the parameters and their units are listed in Table IV.8.1. Enzyme coding units (E) contain two integers s and $p \in \mathbb{N}^*$, indicating the tag of the substrate and product respectively, two floating-point numbers $k_{cat} \in \pm[10^{-3}, 10^{-1}]$, and the ratio $k_{cat}/K_M \in [10^{-5}, 10^{-3}]$ describing the enzymatic kinetics (see the description of the metabolic network below). In the special case where $s = p$, the enzyme is considered as a pump, actively pumping in (or out) the metabolite s if k_{cat} is positive (or negative, respectively). Initial genomes of 50 genomic units are generated. These genomes contain ten P and ten E, all with random positions and attribute values.

Upon cell division, the parental genome is replicated with mutations in the two daughter cells. Each genomic unit can undergo point mutations, meaning here changes in the numbers it contains, like the values of s, p, k_{cat} and k_{cat}/K_M for an E. Each unit attribute mutates at a rate of 10^{-3} per attribute per replication. For the substrate/product tags, a mutation consists in randomly incrementing/decrementing s or p respectively. For k_{cat}

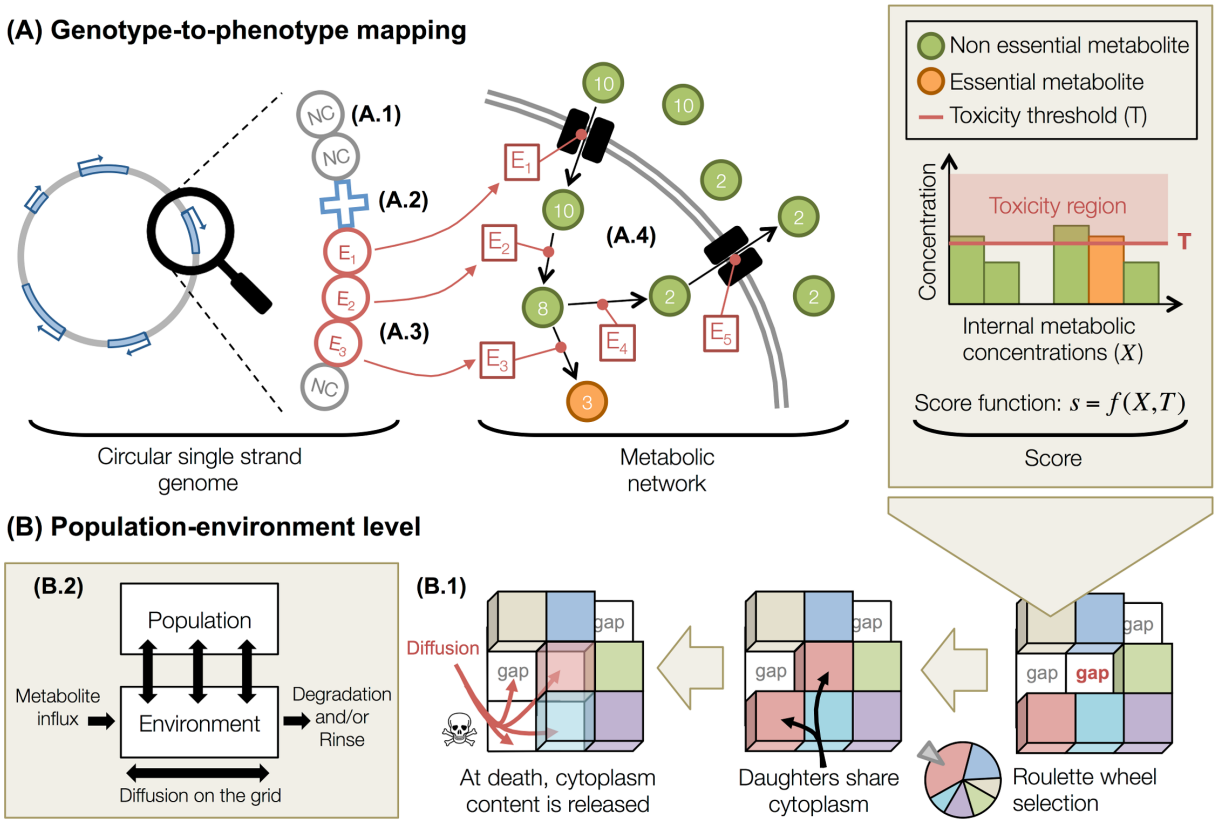


Figure IV.1 – Presentation of the model. The genotype-to-phenotype mapping, as well as the population and environment, are schematized here. **(A)** Description of the genotype-to-phenotype mapping. Organisms own a coarse-grained genome that contains genomic units. **(A.1)** Non-coding units (NC, grey circles) are not functional. The arrangement of the genomic units on the circular single strand defines functional regions, where a promoter (P, blue cross, **A.2**) controls the expression of all contiguous enzyme units (E, red circles), thereby allowing for operons. **(A.3)** When enzyme units are expressed, they contribute to the metabolic network. **(A.4)** Enzymes perform metabolic reactions in the cytoplasm, or pump metabolites in or out (see the description of the metabolic network below). The score of an organism is computed from its "essential metabolites" (see the description of the score function below). Lethal toxicity thresholds are applied to each metabolic concentration and preclude organisms to accumulate resources. **(B)** Description of the population and environment levels. Organisms are placed on a 2D toroidal grid, and compete for resources and space. **(B.1)** When an organism dies, it leaves its grid spot empty and organisms in the Moore neighborhood (if any) compete to divide in the available spot. The competition is based on scores, a minimal threshold being applied on scores to preclude worst organisms to divide. At division, daughters share cytoplasm content (enzymes and metabolites). At death, metabolites from the cytoplasm are released in the local environment and diffuse on the grid. **(B.2)** At the largest scale, the population evolves in the environment by uptaking, transforming and releasing metabolites. Metabolites then diffuse and are optionally degraded. This interaction between the population and its environment allows for the evolution of complex ecological situations.

or k_{cat}/K_M, a random number drawn from $\mathcal{N}(0, 0.1)$ is added to the decimal logarithm of the parameter. β mutates by adding a random number drawn from $\mathcal{N}(0, 0.1)$. A genomic unit can also undergo a type transition from any unit type to any other at a predefined rate, set here to 10^{-3} per genomic unit per replication. All types of genomic units are

actually implemented as a tuple containing all possible attributes, like (unit_type, β, s, p, k_{cat}, k_{cat}/K_M). The unit type tells us which parameters are functionally relevant and the others are free to mutate neutrally.

The genome can also undergo rearrangements affecting segments of any number of genomic units. There are four types of rearrangements : duplications, deletions, translocations and inversions. All rearrangement rates are set to 10^{-3} per genomic unit per replication, hence the number of rearrangements is related to the genome size thereby limiting genome expansion (Fischer et al., 2014). The breakpoints for each rearrangement are randomly drawn in the whole genome. In real genomes, spontaneous rearrangement breakpoints have no reason to lie exactly between two of our genomic units and could thus break our genomic units. To model that with our coarse-grained genome representation, we alter the content of the two genomic units that are adjacent to a rearrangement breakpoint. Suppose for example that a deletion joins two genomic units, one containing the attributes (unit_type$_1$, β_1, s_1, p_1, k_{cat1}, $(k_{cat}/K_M)_1$) and the other the attributes (unit_type$_2$, β_2, s_2, p_2, k_{cat2}, $(k_{cat}/K_M)_2$). Then for each attribute, there is a probability of 10^{-3} for the value in unit 1 to be exchanged with the value in unit 2. Both units could for example exchange their values of s, thereby leading to (unit_type$_1$, β_1, s_2, p_1, k_{cat1}, $(k_{cat}/K_M)_1$) and (unit_type$_2$, β_2, s_1, p_2, k_{cat2}, $(k_{cat}/K_M)_2$).

IV.2.2 Metabolic network

Gene products can either be pumps, pumping metabolites from or to the growth medium, or enzymes performing catalytic transformations in the metabolic space.

Let us consider an enzyme in the cytoplasm, that catalyzes one specific reaction $s \rightarrow p$, with $s \in \mathbb{N}^*$ and $p \in \mathbb{N}^*$ being the substrate and the product of a Michaelis-Menten-like reaction, respectively. The variation in concentrations $[E]$, $[s]$ and $[p]$ over time are then driven by Eq IV.1:

$$\begin{cases} \dfrac{d[E]}{dt} = \beta - \phi[E] \\ \dfrac{d[s]}{dt} = -\dfrac{k_{cat}[E][s]}{K_M + [s]} \\ \dfrac{d[p]}{dt} = \dfrac{k_{cat}[E][s]}{K_M + [s]} \end{cases} \tag{IV.1}$$

where β is the basal production rate specified in the promoter unit, ϕ is the enzyme degradation rate (set to 0.1 per centi-time-step for all enzymes here, with 1 centi-time-step = 0.01 time-steps), K_M and k_{cat} are the kinetic attributes of the enzyme (K_M being deduced from k_{cat} and k_{cat}/K_M attributes).

Pumps are treated here as special enzymes for which $[s]$ and $[p]$ describe the internal and external concentrations of the same metabolite. If k_{cat} is positive (resp. negative), $[s]$ is the external (resp. internal) concentration of the metabolite and $[p]$ the internal (resp. external) concentration. The dynamics of metabolic concentrations $[s]$ and $[p]$ are thus also driven by Eq IV.1 when the gene product is a pump.

Each organism has an ODE (Ordinary Differential Equation) system that keeps track of: **(i)** the concentrations of all metabolites inside the organism, *i.e.*, internal concentrations, **(ii)** the concentrations of all metabolites at the organism's location on the grid, *i.e.*, external concentrations, and **(iii)** the concentrations of all proteins (pumps and enzymes) in the cytoplasm. For a very simple organism whose genome merely encodes one pump importing metabolite #10 into the cell, and one enzyme converting #10 to #7, the ODE system would read:

$$\begin{cases} \dfrac{d[\text{Pump}]}{dt} = \beta^{\text{Pump}} - \phi[\text{Pump}] \\ \dfrac{d[\text{Enzyme}]}{dt} = \beta^{\text{Enzyme}} - \phi[\text{Enzyme}] \\ \dfrac{d[\#10_{\text{external}}]}{dt} = -\dfrac{k_{cat}^{\text{Pump}}[\text{Pump}][\#10_{\text{external}}]}{K_M^{\text{Pump}} + [\#10_{\text{external}}]} \\ \dfrac{d[\#10_{\text{internal}}]}{dt} = \dfrac{k_{cat}^{\text{Pump}}[\text{Pump}][\#10_{\text{external}}]}{K_M^{\text{Pump}} + [\#10_{\text{external}}]} - \dfrac{k_{cat}^{\text{Enzyme}}[\text{Enzyme}][\#10_{\text{internal}}]}{K_M^{\text{Enzyme}} + [\#10_{\text{internal}}]} \\ \dfrac{d[\#7_{\text{internal}}]}{dt} = \dfrac{k_{cat}^{\text{Enzyme}}[\text{Enzyme}][\#10_{\text{internal}}]}{K_M^{\text{Enzyme}} + [\#10_{\text{internal}}]} \end{cases} \tag{IV.2}$$

The number of equations in the ODE system generally differs across individuals within a population because it depends on the number of functional genes, and chromosomal rearrangements like duplications and deletions can alter gene number. In practice, the size of the ODE system goes from tens to thousands of equations depending on the individual. Similarly, the parameter values of the ODE system also vary across individuals, as they are encoded in the organism's genes and thus result from the mutation process.

Initially, in the individuals used to seed a run at time-step 0, each protein starts at its equilibrium concentration β/ϕ, and each metabolite starts with an internal concentration of 0.0 ACU (Arbitrary Concentration Unit). At time-step 0, for all grid spots, external concentrations are initialized to 0.0 ACU for all nutrients except for metabolite #10 (the exogenous carbon source). Between time-steps 0 and 1, the ODE system computes the dynamics of the metabolite and protein concentrations, using the adaptive Runge-Kutta-Cash-Karp method (RKCK), during 100 centi-time-steps. In organisms that possess a pump for metabolite #10, this metabolite will enter the cell. If the genome of this

organism also encodes an enzyme to transform #10 into #7 for example, then the internal concentrations will show an accumulation of #7. At time-step 1, each organism will either die, divide or just survive (see paragraph "Population and environment" below for details). If the organism merely survives without dividing, its current internal concentrations are used as initial conditions for the computation of the next 100 centi-time-steps (*i.e.*, for the transition from time-step 1 to 2). If the organism divides, each of the two daughter cells inherits half of each metabolite and each protein amounts. These will constitute the initial conditions for each cell's ODE system for the next 100 centi-time-steps. If the organism dies, its internal content is released into the environment, thereby increasing the local external concentrations. As the metabolites can diffuse across the grid, the metabolites produced by the dead cell, like metabolite #7, will become available to the neighboring cells, which will thus be able to feed on both #10 and #7, if they own the corresponding pumps. This process is repeated for each transition from time-step t to time-step $t+1$.

Thus, when *e.g.*, a 32×32 grid is full of organisms (see the description of the experimental protocol below), a time-step involves the computation of about a thousand different ODE systems, each of them containing from tens to thousands of equations depending on gene number.

IV.2.3 Score function

Some metabolites are essential for an organism's replication. Here, we arbitrarily define as essential the metabolites whose tag is a prime number. The score of an organism is then simply defined as the sum of its internal concentrations of essential metabolites. However, to prevent organisms from producing a single specific prime number in huge quantities, we also define lethal toxicity thresholds for both essential and non essential metabolites. Here these toxicity thresholds are set to 1.0 ACU for all metabolites.

IV.2.4 Population and environment

Organisms evolve on a two-dimensional toroidal grid, each spot containing at most one organism. The physical environment is described at the grid level: each grid spot contains external metabolites, each with its concentration. These external metabolites diffuse with a diffusion parameter $D = 0.1$ gridstep^2.time-step^{-1}, meaning that a fraction D of each metabolite present at one location will diffuse to each of the eight neighboring grid spots at each time-step. The discrete diffusion equation we are using is inspired from Frénoy et al. (2013). External metabolites are also degraded with a degradation rate D_g, meaning that a fraction D_g of each metabolite at each location will disappear at each time-step. We make the simplifying assumption that there are no enzymatic reaction in the environment, and thus that metabolite transformation only occurs inside the organisms. Organisms compete for the external metabolites to produce offspring in empty spots. They interact with their local environment by pumping metabolites in and

out and releasing their metabolic content at death. At each time-step, organisms are evaluated and either killed, updated or replicated depending on their current state:

1. If the organism does not die and cannot divide (*e.g.*, because there is no free space in its neighborhood), its metabolic network is updated, and its score is computed. If lethal toxicity thresholds are reached, the organism dies (see point 2);

2. Organisms can also die randomly with a probability following a Poisson law of parameter $p_{death} = 0.02$ per organism per time-step. At death, the metabolic content is released into the local environment;

3. For each empty grid spot, all living organisms in the Moore neighborhood whose score is higher than a minimum score of 10^{-3} ACU compete. The organism having the best score in the neighborhood is allowed to divide if it did not replicate previously at the same time-step (such that any dividing cell generates at most two daughters per time-step).

IV.3 Experimental protocol

In all our simulations, the environment provided one primary resource with tag $m_{exo} = \#10$. To initialize an evolutionary run, the entire grid was populated with individuals having random genomes (different for each individual). This initial population was allowed to evolve for 500 time-steps, at which point its viability is assessed. We repeated this procedure until a viable population was found, *i.e.*, with at least 500 viable individuals after the 500 time-steps. In this case, some organisms possess at least one pump to internalize m_{exo}, and (because m_{exo} is not a prime number, see the description of the score function above) one enzyme to transform m_{exo} into a prime number, thereby producing an "essential metabolite". Up to a few hundred trials were usually needed to find a viable population, which was then used to seed the evolutionary run. Each evolutionary run was seeded with a different viable population. These organisms grow on the primary resource and start to release by-products (mostly at death), hence modifying their environment. Populations evolved in two different environments:

1. The **periodic environment**, in which the resource dynamics of the LTEE (Elena and Lenski, 2003) was mimicked. The environment was periodically refreshed by removing all the external metabolites and introducing m_{exo} at concentration $f_{in} = 10.0$ ACU per grid spot. Internal metabolites were not affected by the refresh event. The refresh period was $\Delta t = 333$ time-steps. We call a "cycle" this time interval between two environmental resets. The value of Δt was calibrated to let the organisms live for approximately 7 generations per cycle, as in the LTEE. Within each cycle, the metabolites in the environment were conserved ($D_g = 0$ per time-step). Note that we mimicked the resource dynamics of the LTEE but not the 1% population

subsampling occurring during serial transfers, because it would have implied transferring populations of 10 individuals or fewer. Such a low population size would have implied dramatic genetic drift and impeded adaptive evolution (in the LTEE, where the population size before sampling is very large, the 1% subsampling still leaves the population large enough to keep genetic drift reasonably low). To simulate subsampling, a significantly larger grid would have been needed, making the whole campaign impossible to compute in a reasonable time.

2. The **continuous environment**, in which the resource dynamics of a chemostat environment was mimicked. The medium was constantly provided with a small influx of the primary resource. All the external metabolites were slowly degraded. Specifically, at each time-step, a concentration $\Delta f_{in} = 0.03$ ACU of m_{exo} was added in every grid spot, and external metabolites were degraded at rate $D_g = 0.003$ per time-step.

For each environment, 12 independent populations were propagated for 500,000 time-steps (approximately 50,000 generations). On the long-term, the quantity of resources available in the system was equivalent in both environments. The grid size is 32×32. Complementary experiments were also run in a randomized batch environment similar to the periodic environment except that the environment reset intervals followed a Poisson law of parameter $\Delta t = 333$ time-steps instead of the exact regular period of 333 time-steps. The simulation parameters common to all the simulations are described in Table IV.8.1.

IV.3.1 Cross-feeding interactions

In order to detect the potential cross-feeding interactions in the population, the metabolic activity of each individual was evaluated at each time-step. For each organism, a "trophic profile" was computed from its metabolic network activity. The trophic profile is a binary sequence summarizing the uptake, production and release activity of an organism. The length of the binary string was defined by the largest metabolite tag present in the system at time t. For example, if an organism uptakes metabolite #4, produces #3 from #4 and releases #3, knowing that the largest metabolite tag in the whole grid is #5, then its profile is |00010|00100|00100|. We classified organisms in two trophic groups depending on their trophic profiles:

1. "Group A" pumps in m_{exo}, and possibly other metabolites,
2. "Group B" pumps in group A by-products, and possibly other metabolites, but not the primary resource m_{exo}.

A trophic group is considered an ecotype if the organisms of the group form a monophyletic cluster (see below).

IV.3.2 Phylogenetic relationships

Phylogenetic relationships were exhaustively recorded during each simulation. Since organisms can only divide once per time-step, phylogenetic trees are binary trees. It was possible to recover the line of descent of any organism, and to compare the phylogenetic tree structure with the distribution of the trophic groups in the population. In particular, we can determine if groups A and B are monophyletic, and thus can be considered as ecotypes. To this aim, we computed a phylogenetic structure score (PS score) to identify the degree of monophyly of both groups. This phylogenetic structure score was defined as $PS = |f_1 - f_2|$, where f_1 and f_2 are the relative frequencies of group B in both subtrees rooted to the last common ancestor of the whole final population. A high PS value indicates a strong clustering of groups A and B in the phylogenetic tree, *i.e.*, that groups A and B are two different ecotypes. A low PS value indicates a random distribution or the absence of polymorphism.

IV.4 Sensitivity analysis

We tested variations of our parameters set (see Table IV.8.1), by changing the death probability p_{death}, the external metabolites diffusion rate, the mutation rates, the toxicity thresholds, the "migration rate" (a parameter controlling the fraction of exchanged pairs among all possible pairs of individuals), and the grid size. Details and results are described in Appendix IV.8.5.

IV.5 Results

First, the global evolutionary dynamics of the system can be analyzed by looking at main simulation statistics. The evolution of the mean score, the environmental richness (the number of different metabolites available in the environment), the number of trophic profiles, and the proportion of organisms of group A or B are represented in Figure IV.2. The score and the environmental richness were of the same order of magnitude in the continuous and the periodic environments, but they were more stable in the continuous environment. The number of trophic profiles showed no striking difference between the periodic and the continuous environment (Figs. IV.2A.3 and IV.2B.3), indicating that polymorphism was common in both situations. However, the dynamics of groups A and B were completely different. In the periodic environment, groups coexisted, even if they showed long-term frequency variations (Fig. IV.2A.4). In the continuous environment, group B quickly emerged too but also quickly disappeared in all cases (Fig. IV.2B.4). Thus, even if the diversity of trophic profiles was similar in both environments, all profiles belonged to group A in the continuous environment. Hence, there was no group exclusively specialized on by-products in the continuous environment, while they were common in

the periodic one.

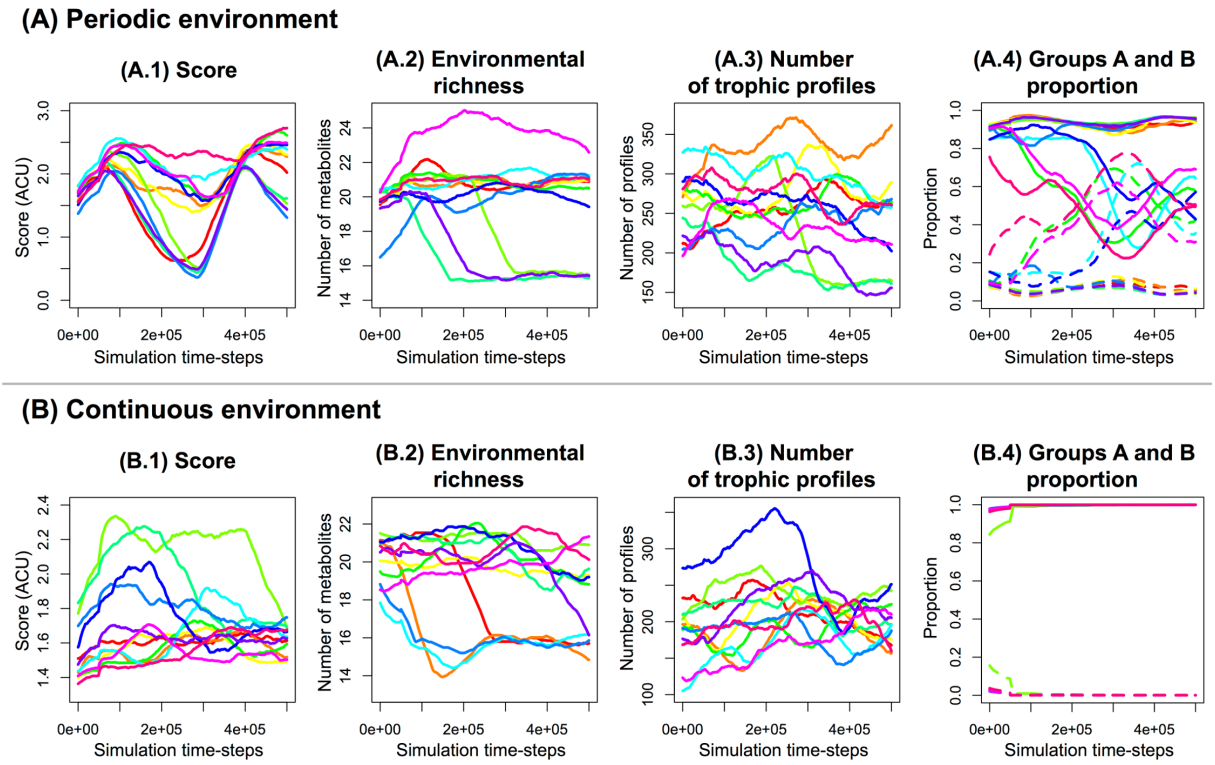


Figure IV.2 – Evolution of typical variables. Evolution of the mean score (**A.1** and **B.1**), the environmental richness (the number of different metabolites present in the environment, **A.2** and **B.2**), the number of trophic profiles (**A.3** and **B.3**), and the proportion of organisms of group A or B are represented (**A.4** and **B.4**). (**A**) Evolution in the periodic environment. (**B**) Evolution in the continuous environment. In **A.4** and **B.4**, group A is represented in solid lines and group B in dashed lines.

Impact of environmental dynamics on evolved genome and network organization

We compared the structure of both the genome and the metabolic network of final A organisms (after 500,000 time-steps) from the continuous and periodic environments of the main campaign (see above). We evaluated five variables: **(i)** the mean genome size, **(ii)** the mean amount of non-coding DNA, **(iii)** the mean number of enzyme coding units encoding the same metabolic reaction (the "metabolic redundancy"), **(iv)** the mean number of different essential metabolites pumped in (the "uptake diversity"), and **(v)** the mean number of different essential metabolites produced (the "production diversity"). For each measure, we performed a two-sample Wilcoxon test with a Bonferroni correction ($n = 5$).

As shown in Table IV.1, there was no significant variation in the amount of non-coding DNA and the uptake diversity. By contrast, genome size (resp. 227.47 and 346.24

units, p-value $< 0.001/5$) and metabolic redundancy (resp. 15.80 and 7.98 units, p-value $< 0.001/5$) were significantly lower in the periodic environment compared to the continuous environment. Moreover, the number of essential metabolites produced was significantly higher in the periodic environment than in the continuous environment (resp. 5.04 and 6.58 essential metabolites, p-value $< 0.001/5$). These differences are explained by selective pressures on the metabolic network. Indeed, organisms experienced a trade-off between maximizing their score (*i.e.*, maximizing the concentration of essential metabolites in their cytoplasm) and avoiding lethal toxicity thresholds. In the periodic environment, the external resource m_{exo} was introduced by bursts of 10.0 ACU at each serial transfer. Thus, to maximize the score without reaching toxicity thresholds, organisms must avoid specializing on a single essential metabolite and instead spread the toxicity by distributing metabolic fluxes in the production of several essential metabolites. In the continuous environment, the external resource was continuously provided at a lower concentration (0.03 ACU at each time-step). In this case, the selective pressure on toxicity was relaxed and the number of essential metabolites produced was significantly lower. Interestingly, metabolic fluxes were also adjusted by amplifying or deleting genes. Indeed, in the continuous environment, there were more copies of E (enzyme coding units) than in the periodic environment, while the production diversity was lower, meaning that those units were amplified in the continuous environment to maximize metabolic fluxes, thus increasing the genome size.

Table IV.1 – Comparison of the structure of the genome and metabolic network structure of final A organisms evolved in the continuous and periodic environments. Five variables were evaluated: **(i)** the mean genome size, **(ii)** the mean amount of non-coding DNA, **(iii)** the mean number of E encoding the same metabolic reaction (the "metabolic redundancy"), **(iv)** the mean number of different essential metabolites pumped in (the "uptake diversity"), and **(v)** the mean number of different essential metabolites produced (the "production diversity"). The standard deviation is also shown (mean ± sd.). For each measure, we performed a two-samples Wilcoxon test, with Bonferroni correction ($n = 5$).

Variable	Continuous env.	Periodic env.	Wilcoxon test	Units
Genome size	346.24 ± 12.98	227.47 ± 53.21	***	Genomic units
Non-coding DNA	5.69 ± 1.21	4.69 ± 1.54	-	Genomic units
Metabolic redundancy	15.80 ± 1.84	7.98 ± 2.08	***	Genomic units
Uptake diversity	3.48 ± 0.37	3.87 ± 1.41	-	Metabolites
Production diversity	5.04 ± 0.31	6.58 ± 0.93	***	Metabolites

Figure IV.3 shows an example of organisms A and B evolved in the periodic environment after 500,000 time-steps (repetition 10). The final best individual of groups A (Fig. IV.3A) and B (Fig. IV.3B) are represented including their genome (Figs. IV.3A.1 and IV.3B.1), metabolic network (Figs. IV.3A.2 and IV.3B.2) and internal metabolic concentrations (Figs. IV.3A.3 and IV.3B.3). The metabolic network of organism A was structured around m_{exo} (this metabolite being a hub in the network), even if the organism also fed on some by-products. Organism B's metabolic network was less complex, and indicates that the organism mostly grew on A-secreted products. Most parts of both genomes were coding enzymes, revealing large operons all along the genomes.

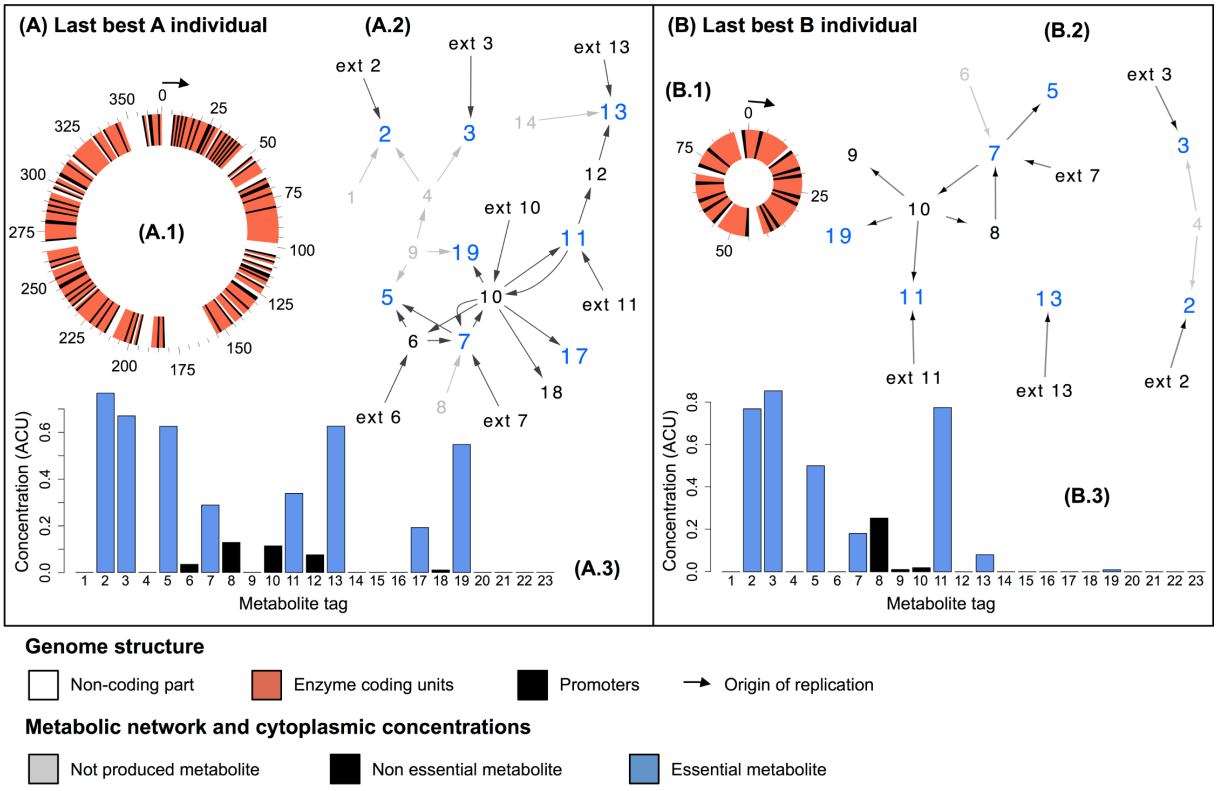


Figure IV.3 – Final best individuals of groups A and B, from repetition 10 of the periodic environment. **(A)** Final best organism A. **(B)** Final best organism B. **(A.1, B.1)** The circular single-stranded genome. Non-functional regions are white, promoters black, E red, revealing numerous operons all along the genomes. **(A.2, B.2)** The metabolic network. Non essential and essential metabolites are colored in black and blue, respectively. Non-functional parts of the metabolic network (where fluxes are null) are shown in grey. **(A.3, B.3)** The internal metabolic concentrations (non essential metabolite concentrations: black. Essential metabolite concentrations: blue).

Relationship between ecology and phylogeny

For each simulation, we analyzed the final phylogenetic tree and compared it to the distribution of groups A and B. All the phylogenetic trees are represented in Figure IV.8.2. Leaves are colored depending on their trophic group (group A in blue, group B in green). The structure of the trees was strongly related to the type of environment. In the periodic environment (Fig. IV.8.2), 5 phylogenetic trees among 12 (repetitions 1, 3, 7, 9 and 10) showed two well-separated clusters, each belonging to one ecological group. In these repetitions, two ecotypes evolved separately and remained stable on the long-term, showing that a stable cross-feeding interaction evolved. In the seven other cases, trees were less deep, had no well separated clusters, and no clear correlation between ecological groups and phylogenetic structure was observed. In the continuous environment (Fig. IV.8.2), trees were much shorter than in the periodic environment. Group A went to fixation in all repetitions. Then, while polymorphism and cross-feeding existed at a similar level in both periodic and continuous environments (Figs. IV.2A.3 and IV.2B.3), this polymorphism was not stable in the continuous environment.

Evolution of phylogenetic structure and trophic groups

To get more insight into the evolutionary dynamics, we computed the distribution of the Most Recent Common Ancestor (MRCA) age at each time-step for all the simulations. The MRCA age reflects the stability of the polymorphism in a population. As shown in Figure IV.4, distributions confirmed that the deepest trees evolved in the periodic environment, with a mean MRCA age of 71,004 time-steps, and a large distribution tail (some trees having almost the same depth as the total simulation time - 500,000 time-steps). By contrast, the mean MRCA age is only 13,524 time-steps in the continuous environment and 11,684 time-steps in the complementary experiment with randomized refresh. This result indicates that environmental variations must be regular to favor stable cross-feeding interactions. The evolution of MRCA age during simulations is also represented in Figure IV.8.3, for the three types of environment. This figure gives a better idea of the evolutionary dynamics of the phylogenetic trees. It shows that the MRCA age regularly collapsed in the random and continuous environments, but was still increasing for some simulations in the periodic environment.

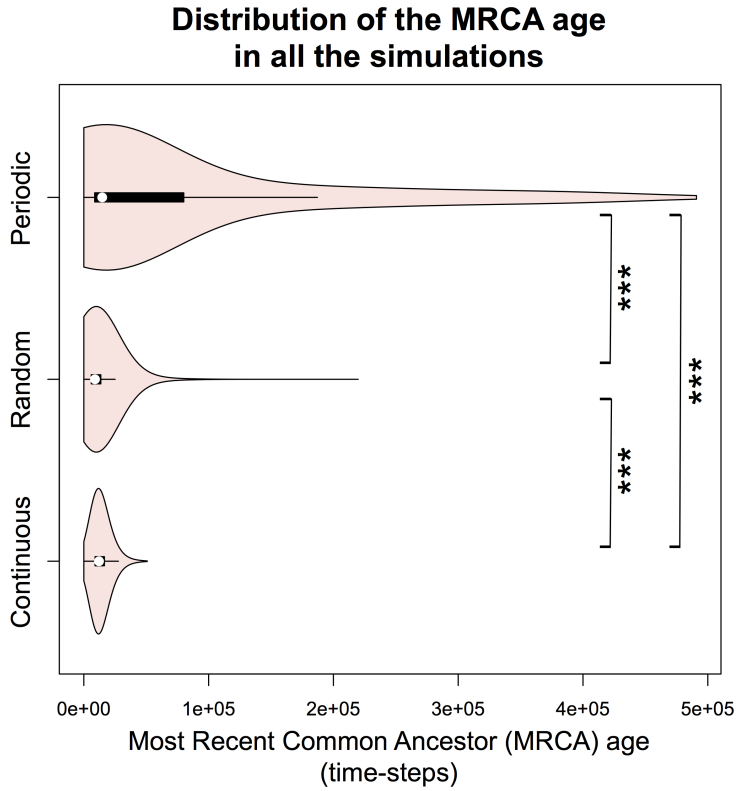


Figure IV.4 – Distribution of the Most Recent Common Ancestor age in all the simulations. For each environment, we computed the distribution across repetitions of the Most Recent Common Ancestor (MRCA) age, for each simulation time-step. All pairwise Student tests are significant, with Bonferroni correction (p-value $< 0.001/3$).

We then compared the phylogenetic structure with the distribution of groups A and B on tree leaves by computing the phylogenetic structure score PS (Fig. IV.5A). In Figure IV.5B, this PS score is plotted against the MRCA age every 1,000 time-steps for

all the repetitions (grey points). Points corresponding to the end of each simulation are colored in black. On each plot, three areas are identified: the purple area indicates long-diverged clades (MRCA age higher than 200,000 time-steps), the orange area indicates when clades correspond to ecotypes (PS score > 0.9), and the intersection of the previous two areas (inside dashed borders) indicates long-diverged monophyletic ecotypes. In the periodic environment (Fig. IV.5B.1), the deepest trees were also the most structured, with two well separated monophyletic ecotypes A and B. In the random environment (Fig. IV.5B.2), the situation was contrasted, with a large distribution of the PS score, ranging from monomorphic trees (A or B groups being fixed), to polymorphic trees. However, the MRCA age was very short compared to the periodic environment, revealing the instability of the phylogenetic structure. Note that the random environment is the only one where we observed a population extinction (1 out of 12). In the continuous environment (Fig. IV.5B.3), the population was mostly monomorphic (group A being fixed), with short MRCA ages.

To evaluate the robustness of these results to the variation of main simulation parameters, we performed a sensitivity analysis. The results are presented in details in Appendix IV.8.5. Even if some parameters were more sensitive than others (*e.g.*, the death probability and the toxicity thresholds, discussed in Appendix IV.8.5), this analysis revealed that our results are robust. In the continuous environment, no single simulation evolved a stable A/B cross-feeding in the whole analysis. Moreover, when the diffusion rate was infinite, or when the population was perfectly mixed (all locations being randomized at each time-step), almost all repetitions (80% in infinite diffusion conditions, 100% in well-mixed conditions) evolved a stable A/B cross-feeding in the periodic environment (see Appendix IV.8.5). This result is in agreement with previous studies showing that the spatial structure may affect polymorphism (Hauert and Doebeli, 2004; Gerlee and Lundh, 2012).

These results confirmed that the periodic environment strongly favored the evolution of stable cross-feeding interactions, in contrast to the random and continuous environments, in apparent contradiction with the results of wet experiments in chemostat, and we will discuss this point below.

Evolution of trophic profiles

We then recovered the proportion of trophic profiles over time (at every 1,000 time-steps) in all the simulations of the periodic and continuous environments (Figs. IV.6A and IV.6B, respectively). Trophic profiles belonging to groups A and B are colored in shades of blue and green, respectively. Those figures show that evolution in the model was ruled by periodic selection in a highly polymorphic population. This polymorphism was mainly due to competition for resources, with the organisms constantly competing for the primary resource but also the by-products available in the environment. However, in the periodic environment (Fig. IV.6A), trophic profiles from groups A and B coexisted over time, with periodic selection events occurring independently in both groups. This dynamics

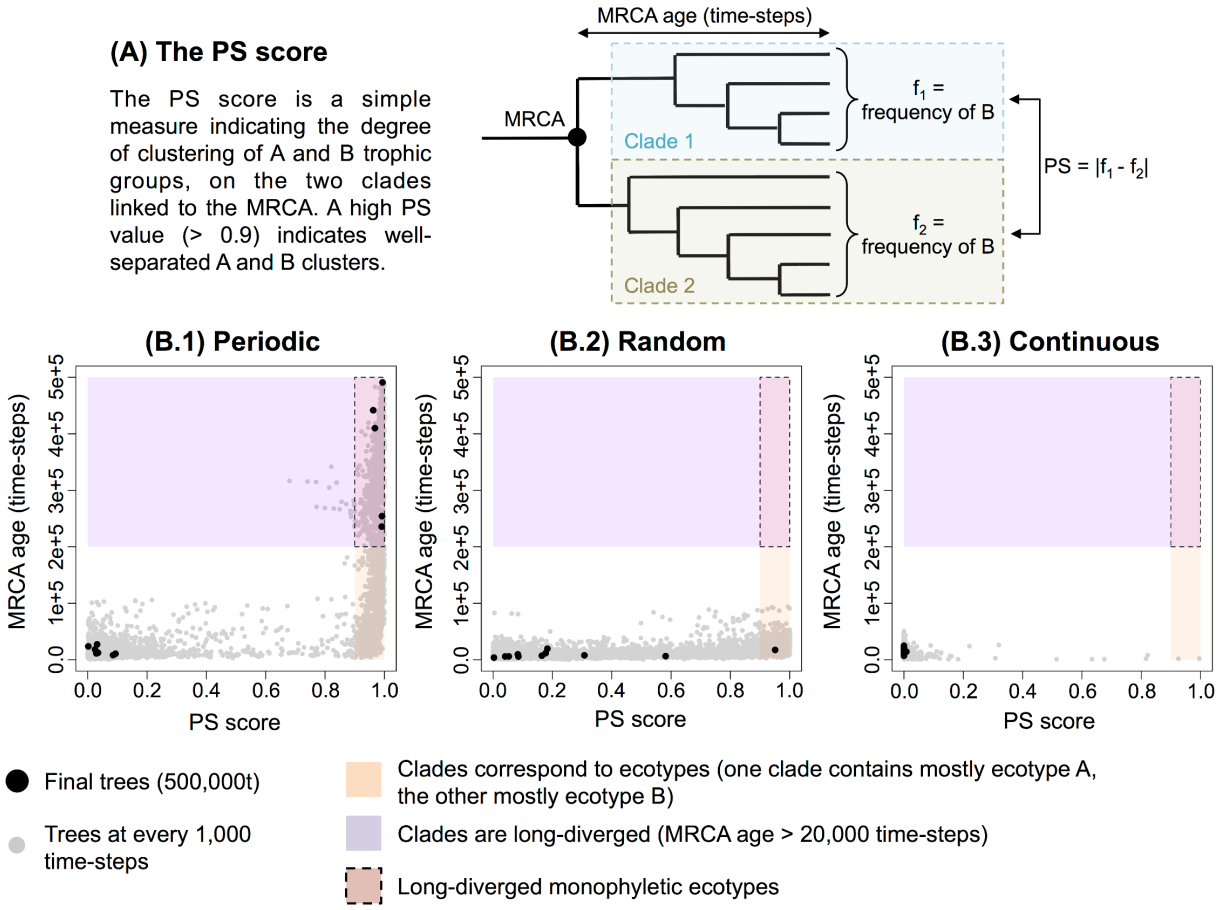


Figure IV.5 – Phylogenetic structure score against the MRCA age. **(A)** The PS score. The PS score is a measure indicating the degree of clustering of A and B trophic groups, on the two clades linked to the MRCA. A high PS value (> 0.9) indicates well separated A and B clusters. **(B)** For each environment, the PS score is plotted against the MRCA age every 1,000 time-steps, for all repetitions, with the points corresponding to the final trees (at 500,000 time-steps) colored in black. Purple area: long-diverged clades (MRCA age higher than 200,000 time-steps). Orange area: clades corresponding to ecotypes (PS score > 0.9). Intersection (inside dashed borders): long-diverged monophyletic ecotypes. **(B.1)** Periodic environment. **(B.2)** Random environment. **(B.3)** Continuous environment.

is typical from multiple niche selection, where beneficial mutations do not spread in all the population owing to competitive exclusion, but are confined in one specific niche. In the continuous environment (Fig. IV.6B), group A was predominant in all simulations, periodic selection affecting the whole population. In these conditions, the level of cross-feeding was maintained but the interactions were not stable (as shown by Figs. IV.2A.3 and IV.2B.3).

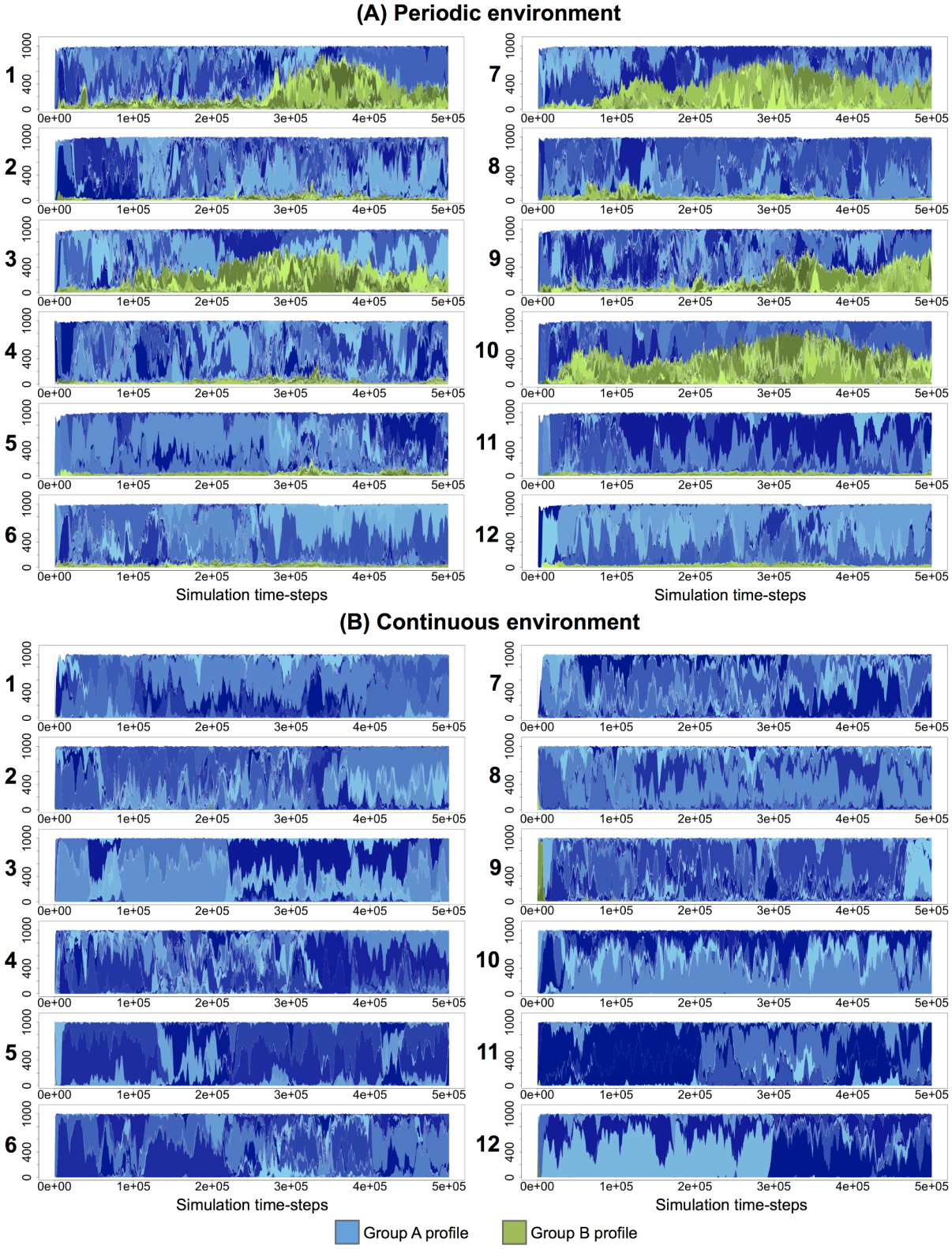


Figure IV.6 – Evolution of trophic profiles in the population for the continuous and periodic environments. Trophic profiles gather organisms that own the exact same metabolic activity (see Methods). Blue and green profiles belong to trophic groups A and B, respectively. **(A)** Continuous environment simulations. **(B)** Periodic environment simulations.

These results reinforce the fact that stable cross-feeding interactions were only possible in the periodic environment. Specifically, in the periodic environment, evolution was driven by multiple niche selection, with periodic selection events independently occurring in ecotypes A and B. On the opposite, in the continuous environment, evolution was driven by periodic selection and competitive exclusion, indicating that there was less opportunity for niche construction.

Ecological dynamics in the periodic environment

Comparative analysis of phylogenetic structure in the different environments revealed that the periodic environment especially favored the evolution of stable cross-feeding interactions, leading to two monophyletic ecotypes A and B in 5 of 12 repetitions, with the ecotype A feeding on the primary resource and possibly on some by-products, while ecotype B consumed by-products. In the LTEE, it has been shown that the coexistence of S and L ecotypes is driven by negative frequency-dependent interactions (Rozen et al., 2005, 2009). We analyzed in details the 5 populations to see whether the stable cross-feeding interactions were comparable to the S/L interaction.

Mutational history of ecotypes A and B.

In the 5 populations that evolved a stable cross-feeding, we recovered the mutational history of the lineages of ecotypes A and B. Final phylogenetic trees of the 5 populations are represented in Figure IV.7. For each tree, the trophic group of the MRCA, as well as the generation at which one of the monophyletic ecotypes switched from the ancestor group to the other one (*i.e.*, when one ecotype lost or gained inflowing pumps for the primary resource), are shown. In all 5 populations, the same pattern emerged: the population was primarily of group A, but niche construction on by-products resulted in adaptive diversification, with one ecotype strongly specializing on by-products, such that it lost the ability to uptake the primary resource.

Interestingly, in all simulations, the loss of this ability was not the source of the adaptive diversification. The diversification event occurred a few hundreds of generations *before* the loss of the pump provoking the change of trophic group. In the LTEE, the S ecotype specialized on acetate, but was still able to grow on glucose. However, recent work has shown that while the S ecotype improved its ability to grow on acetate since the diversification event, it was not the case on glucose, presaging a possible complete loss of its ability to grow on glucose in the longer term (Großkopf et al., 2016). Conversely, the L ecotype improved its ability to grow on glucose, but not on acetate, also presaging a loss of ability to grow on acetate at a longer term.

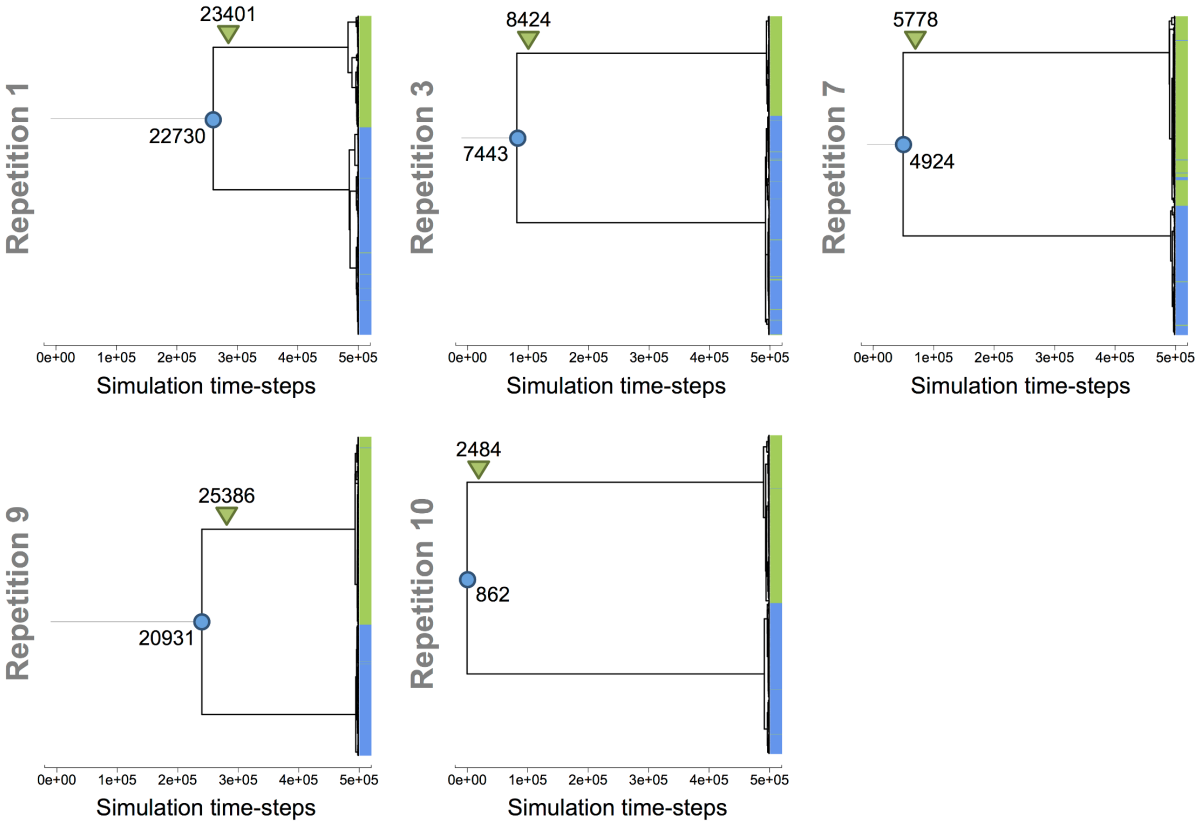


Figure IV.7 – Analysis of the adaptive diversification event leading to the monophyletic ecotypes A and B. In the phylogenetic trees of final evolved populations, the colored circles indicate the trophic group and the generation of the common ancestor. The colored triangles indicate the generation when one monophyletic ecotype moved from one trophic group to the other (*i.e.*, losing or gaining pumps to feed on external nutrient). Group A (blue) grows on the primary resource and possibly on by-products. Group B (green) exclusively grows on by-products.

Ecotype B frequency-dependent fitness in short term competition experiments.

To test whether ecotypes A and B coexistence is maintained by negative frequency-dependent interactions, we performed short term competition experiments with the 5 populations that evolved a stable cross-feeding interaction at the end of the simulations in the periodic environment (repetitions 1, 3, 7, 9 and 10). Initial populations were seeded at 9 different initial frequencies of B (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 and 0.9—each with 10 repetitions) and were propagated in the same periodic environment during 1 cycle (*i.e.*, 333 time-steps). Then, we computed the log-fitness (Chevin, 2011) of ecotype B, taking into account its initial frequency and its frequency at the end of the first cycle. Figure IV.8 demonstrates that the ecotypes A and B interaction was frequency-dependent, the ecotype B being favored when initially rare, and penalized when initially abundant. Since the external conditions varied during the seasons, the B organisms were not favored during the whole cycle. Video IV.8.6 shows the variation of B relative fitness over the 333 time-steps of the first cycle, at a full temporal resolution. This video shows the es-

tablishment of the negative frequency-dependent interaction along the cycle, and reveals that the relative fitness of B was initially negative at all initial frequencies. Indeed, at each cycle, B ecotype growth was delayed compared to A ecotype, the former growing on by-products during the second season, while the latter grew on fresh primary resource during the first season. At low initial frequencies of B, their small number can randomly lead to their extinction, thus artificially reducing its mean relative fitness. Those results are in full agreement with the LTEE (Rozen and Lenski, 2000; Ribeck and Lenski, 2015).

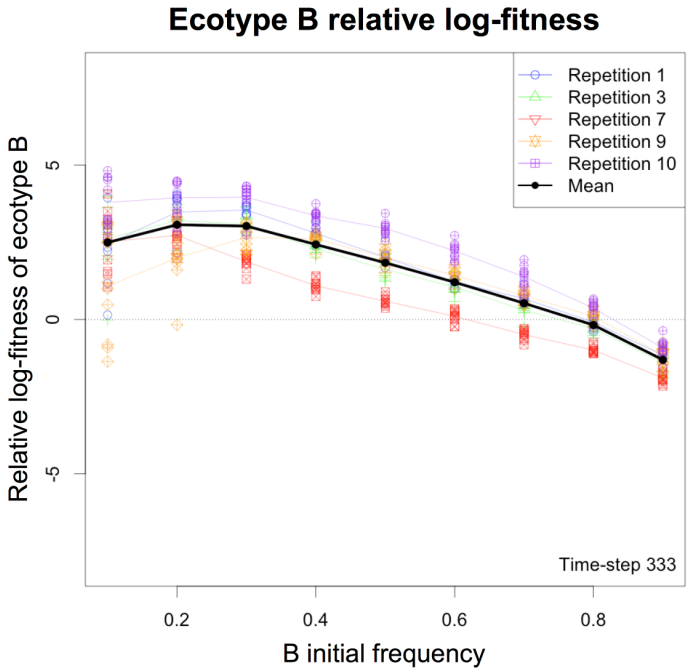


Figure IV.8 – Frequency-dependent relative fitness in short-term competition experiments. The frequency-dependent fitness was computed using log-fitness Chevin (2011); Ribeck and Lenski (2015) in short term competition experiments, starting with different initial frequencies of B ecotype. For each of the 5 populations that evolved monophyletic ecotypes at the end of the simulations, 10 repetitions were run per initial frequency of B (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 and 0.9). The global mean frequency-dependent fitness is represented in black. Mean fitness per population is shown in shaded colors. Each individual experiment is plotted in shaded color dots, related to their mean color.

Convergence to an oscillatory dynamics.

Owing to their negative frequency-dependent interaction, the relative frequencies of ecotypes A and B should stabilize over time, as in the LTEE (Rozen and Lenski, 2000). We extended the previous competition experiments to 10 cycles and recorded the A and B proportions at each time-step (Fig. IV.9). Trajectories show that at all initial frequencies of B, a stable oscillatory dynamics was reached for each repetition (Fig. IV.9A for repetition 1, Fig. IV.9B for rep. 3, Fig. IV.9C for rep. 7, Fig. IV.9D for rep. 9 and Fig. IV.9E for rep. 10). The observed variability was due to contingent evolutionary differences between the 5 populations, and to a sampling effect when the initial frequency of B was low.

Here again we observed exactly the dynamics observed in the LTEE (Rozen and Lenski, 2000; Ribeck and Lenski, 2015), even if the small population size artificially increased the oscillatory dynamics.

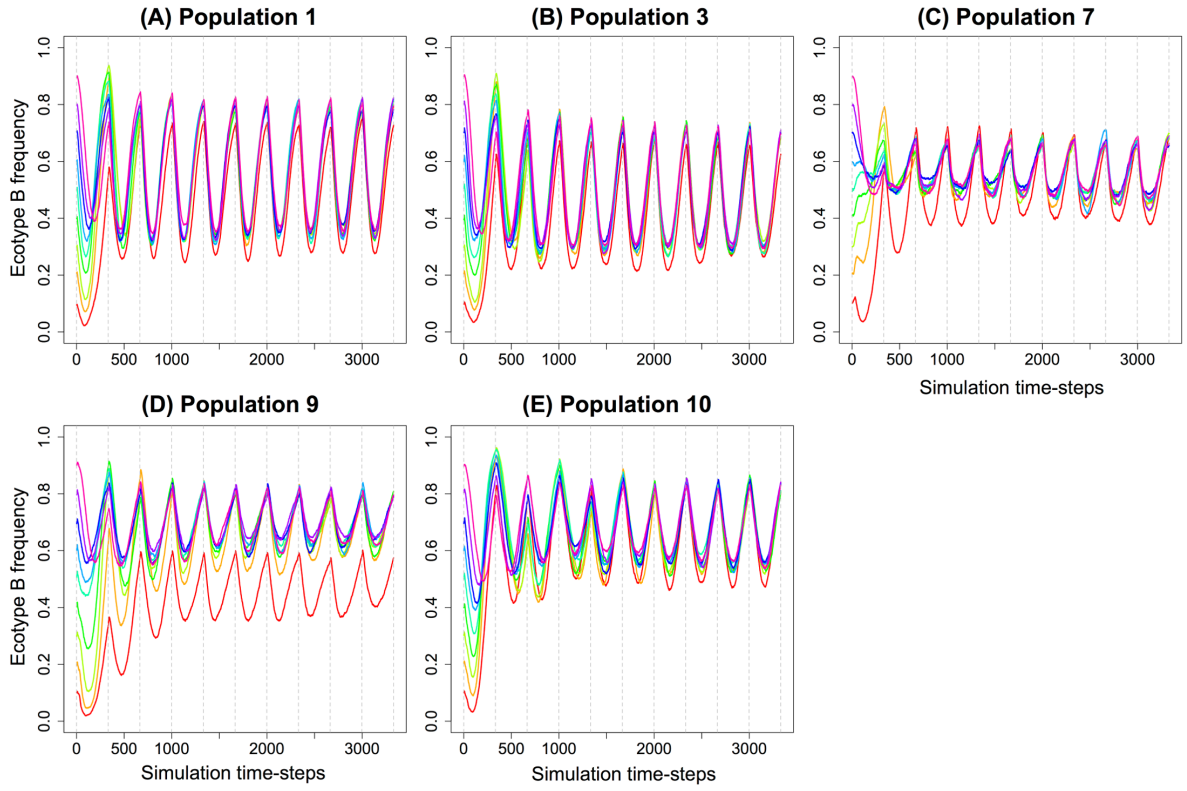


Figure IV.9 – Convergence to an oscillatory dynamics over 10 serial transfer cycles. Ecotype B is advantaged when rare, but is penalized when initially common, leading to a balanced polymorphism. Nine different initial frequencies of B have been tested (0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8 and 0.9). Each trajectory is the mean of the B frequency among the 10 repetitions of each of the 5 populations. **(A)** Population 1. **(B)** Population 3. **(C)** Population 7. **(D)** Population 9. **(E)** Population 10.

Stability of the A/B cross-feeding interactions when transferred in the continuous environment

In the continuous environment, no stable cross-feeding interaction evolved. This result is in apparent contradiction with wet experiments during which *E. coli* populations evolved in a continuous culture with glucose as a single limiting carbon source (Helling et al., 1987; Rosenzweig et al., 1994; Treves et al., 1998). In those experiments, cross-feeding interactions emerged after a few hundreds of generations. Nonetheless, our results showed that cross-feeding interactions quickly emerged in the continuous environment, but these interactions were not stable (Fig. IV.2).

To test whether a population with two stable A and B ecotypes (evolved in the periodic

environment) could persist when placed in the continuous one, we let populations from the periodic environment evolve in chemostat-like environments for 500,000 time-steps. The 5 populations that evolved a stable cross-feeding in the periodic environment were propagated in a continuous environment at two different stages of their evolution: **(i)** just after adaptive diversification (early populations, see Fig. IV.7), **(ii)** and at the end of the simulations (late populations, after 500,000 time-steps). As a control, these populations were also propagated in the periodic environment. For each population, 10 repetitions were run in each environment. Then, we evaluated the stability of the A/B cross-feeding interaction by counting the number of simulations where the interaction persisted, the number of simulations where the interaction failed, and the time before interaction failure.

The proportion of simulations where the interaction persisted are displayed in Table IV.2 for the continuous environment, and in Table IV.3 for the periodic environment. The evolution of the proportions of groups A and B is also shown in Figures IV.10A and IV.10C for the continuous environment (early and late populations, respectively), and in Figures IV.10B and IV.10D for the periodic environment (early and late populations, respectively). First, Table IV.2 shows that, for early populations in the continuous environment, the interaction was not robust and persisted in only 6% of the assays. For late populations in the continuous environment, the interaction was more robust, as the polymorphism persisted in 50% of the assays. For early populations in the periodic environment, the interaction was also not robust (the interaction persisted in only 18% of the assays), even if more populations maintained the interaction than in the continuous environment. This low percentage is probably due to the experimental protocol: populations were transferred in a periodic environment at the beginning of a new cycle, whatever the previous seasonal context of the population. The interaction was then destabilized while the diversification event was still recent, leading to a high probability to loose the interaction (a situation similar to what occurred in the random environments). However, for late populations in the periodic environment, most assays kept the polymorphism stable (the interaction persisted in 78% of the assays), indicating that seasonality is of primary importance to stabilize the interaction.

Table IV.2 – Proportion of assays where polymorphism persisted in chemostat conditions. For each population, 10 assays were simulated. The stable polymorphism was considered to be lost if the MRCA age changed, indicating that one of the two monophyletic groups was outcompeted.

	Pop 1	Pop 3	Pop 7	Pop 9	Pop 10
Early populations	0%	0%	0%	30%	0%
Late populations	90%	70%	30%	40%	20%

Table IV.3 – Proportion of assays where polymorphism persisted in batch conditions. For each population, 10 assays were simulated. The stable polymorphism was considered to be lost if the MRCA age changed, indicating that one of the two monophyletic groups was outcompeted.

	Pop 1	Pop 3	Pop 7	Pop 9	Pop 10
Early populations	30%	20%	0%	40%	0%
Late populations	100%	100%	90%	60%	40%

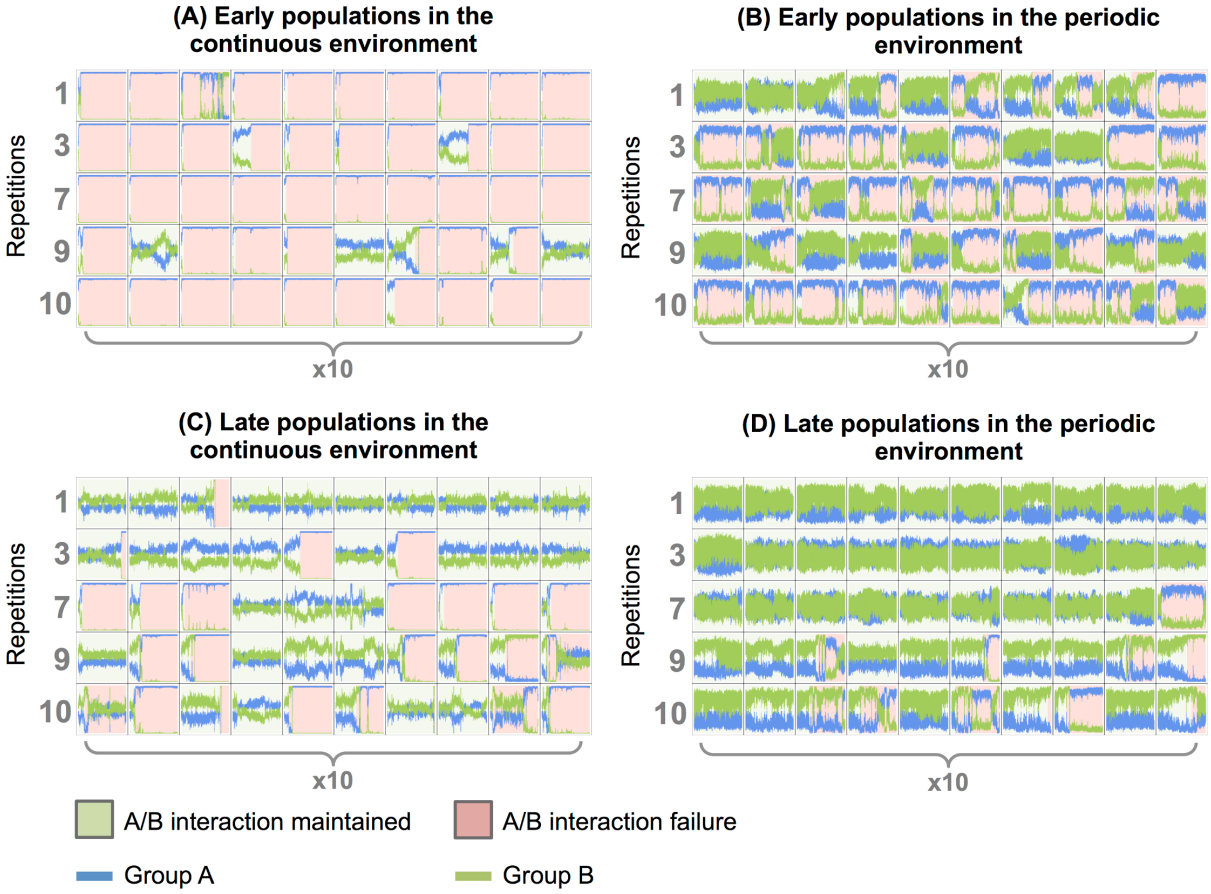


Figure IV.10 – Stability of the A/B interaction evolved in the periodic environment, when placed in the continuous one. Early populations were transferred just after adaptive diversification. Late populations were transferred at the end of the simulations (500,000 time-steps). For each repetition that evolved two ecotypes A and B (rep. 1, 3, 7, 9 and 10), 10 repetitions of 500,000 time-steps were run. The stable polymorphism was considered to be lost if the MRCA age changed, indicating that one of the two monophyletic groups was outcompeted. In this case, the simulation is colored in green, and red before and after this event, respectively (simulations where the A/B interaction was maintained during the whole experiment are fully green). **(A)** Early populations transferred in the continuous environment. **(B)** Early populations transferred in the periodic environment. **(C)** Late populations transferred in the continuous environment. **(D)** Late populations transferred in the periodic environment.

Figure IV.11 shows the distribution of the time before A/B interaction failure for early and late populations in the continuous environment. Late populations were much more robust, since extinctions happened significantly later (with a mean of 142,291.2 time-steps) than for early populations (with a mean of 37,173.68 time-steps). Student test gives a p-value < 0.001.

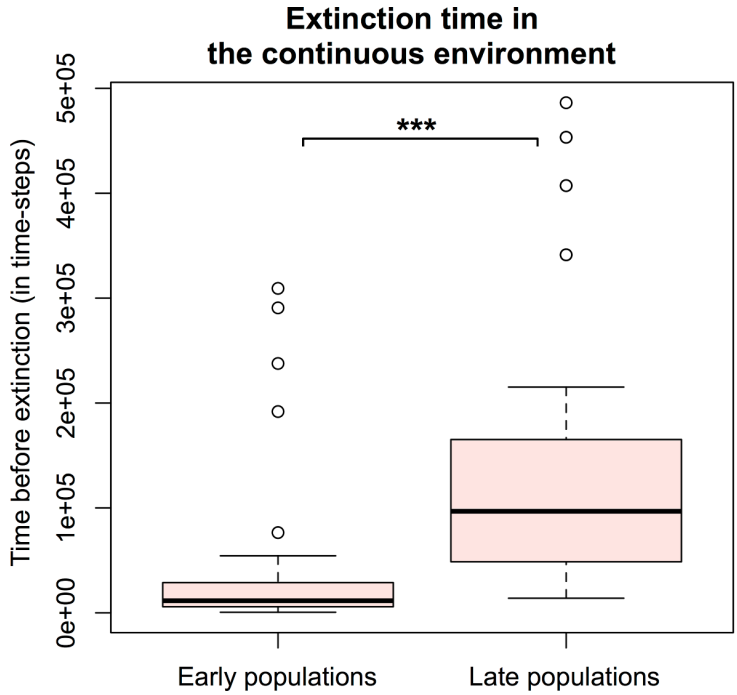


Figure IV.11 – Time before A/B interaction failure in the continuous environment. The time (in time-steps) is measured for all competition experiments in the continuous environment (50 simulations) where the interaction failed. Persistent interactions are not considered here. Early populations loose the interaction significantly earlier than the late ones (Student test is significant with a p-value < 0.001).

Vulnerability of ecotype B to A fast-growing mutants when transferred in the continuous environment.

In order to understand why the A/B interaction failed in half of the continuous environment experiments (50% of the assays in the late populations), and why the A/B interaction failures implied the extinction of ecotype B in most cases (80% of the failures in the late populations), we studied in details the evolution of digital organisms in this environment.

For each late population propagated in the continuous environment (Fig. IV.10C), we compared the initial A ecotype to the final A ecotype (after 500,000 time-steps of evolution in the continuous environment). We performed the same genomic and metabolic analysis as in section "Impact of environmental resource dynamics on evolved genome and network organization". Table IV.4 shows that A organisms **(i)** significantly increased their genome size (from 189.99 to 269.73 units, p-value $< 0.001/5$), their mean metabolic redundancy (from 6.69 to 10.86 units, p-value $< 0.001/5$) and mean uptake diversity (from 1.17 to 1.69 metabolites, p-value $< 0.001/5$), and **(ii)** significantly decreased their mean production diversity (from 7.14 to 5.66 metabolites, p-value $< 0.001/5$), when they evolved in the continuous environment for 500,000 time-steps. Indeed, the relaxation of selective pressures to maintain concentrations under the lethal toxicity thresholds led to a restructuring of A organisms towards a genome and metabolic network well-adapted to continuous conditions (see Table IV.1). This modification of ecotype A phenotypes im-

paired the negative frequency-dependent interaction between the A and B ecotypes: since B organisms consumed A-secreted by-products, the reduction of the production diversity of A organisms led to the extinction of ecotype B in half of the assays.

Table IV.4 – Comparison of the genome and metabolic network structure of initial and final ecotype A, when transferred in the continuous environment. Five variables were evaluated: **(i)** the mean genome size, **(ii)** the mean amount of non-coding DNA, **(iii)** the mean number of E encoding the same metabolic reaction (the "metabolic redundancy"), **(iv)** the mean number of different essential metabolites pumped in (the "uptake diversity"), and **(v)** the mean number of different essential metabolites produced (the "production diversity"). The standard deviation is also shown (mean ± sd.). For each measure, we performed a two-samples Wilcoxon test, with Bonferroni correction ($n = 5$).

Variable	Initial pop.	Final pop. (500,000t)	Wilcoxon test	Units
Genome size	189.99 ± 49.85	269.73 ± 96.69	***	Genome units
Non-coding DNA	4.69 ± 2.10	5.18 ± 2.01	-	Genome units
Metabolic redundancy	6.69 ± 1.46	10.86 ± 4.81	***	Genome units
Uptake diversity	1.17 ± 0.54	1.69 ± 1.13	***	Metabolites
Production diversity	7.14 ± 0.73	5.66 ± 1.78	***	Metabolites

To exemplify these statistical results, we studied in details the evolution of ecotypes A and B in the 10 repetitions of the late population 3, when propagated in the continuous environment (Fig. IV.10C.3). At the beginning of the assays, ecotypes A and B interacted through a negative frequency-dependent cross-feeding: ecotype A organisms produced essential metabolites 2, 3, 5, 7, 11, 13, 17 and 23; ecotype B organisms consumed metabolites 2, 3, 5 and 7 (all secreted by ecotype A organisms). We evaluated the evolution of the 8 essential metabolites that were produced by ecotype A organisms at the beginning of the assays (Fig. IV.12). Ecotype A organisms reduced their production of essential metabolites in all assays. However, when ecotype A organisms stopped producing metabolites 2, 3, 5 and/or 7, ecotype B organisms systematically went to extinction. On the opposite, when ecotype A organisms stopped producing metabolites 17 and/or 23 (but maintained the production of 2, 3, 5 and 7), ecotype B was not affected. These results confirm the mechanism of B extinction: when placed in continuous conditions, ecotype A organisms reorganized their metabolism and produced fewer essential metabolites. Now, while doing so they may stop producing metabolites that were necessary for the survival of ecotype B organisms, leading to their extinction.

The robustness of the A/B interaction in late populations is explained by character displacement and niche specialization.

In order to understand why the interaction between A and B ecotypes was more robust in late than early populations, we compared the genomic and metabolic structures of early and late populations, independently for A and B ecotypes. We performed the same statistical tests as in Tables IV.1 and IV.4 (two-samples Wilcoxon test with Bonferroni correction of $n = 5$). The results are presented in Table IV.5. In both ecotypes, the

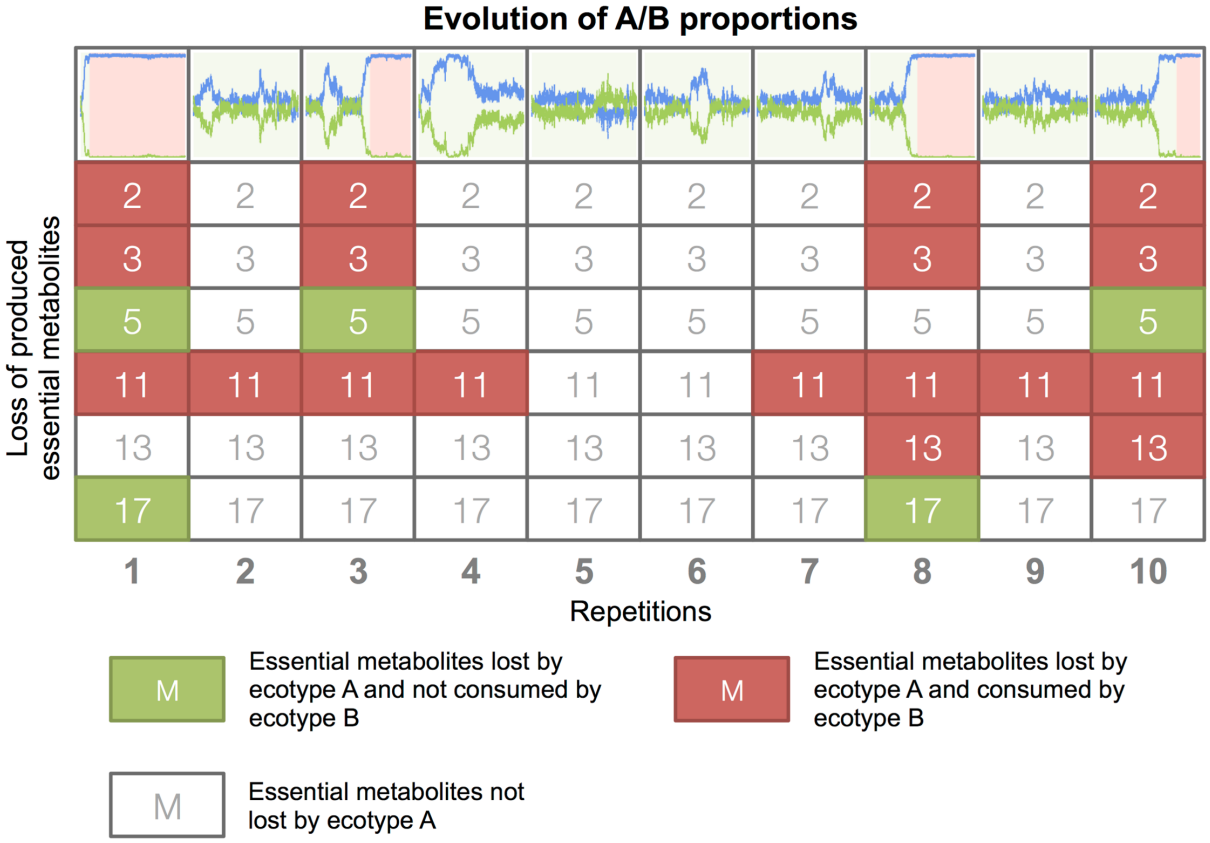


Figure IV.12 – Loss of essential metabolites production of ecotype A organisms in the 10 repetitions of the late population 3 in the continuous environment assays. The 10 repetitions of the population 3 are displayed. The 8 essential metabolites (2, 3, 5, 7, 11, 13, 17, and 23) that were produced by ecotype A organisms at the beginning of the assays are represented vertically for each repetition. Colored metabolites indicate a production loss. Essential metabolites that are consumed by ecotype B organisms are colored in red, the others in green. At the top, the evolution of groups A and B proportions is represented in green when the A/B interaction persisted and in red when the interaction failed. In all simulations where A ceased to produce a metabolite pumped-in by B, B went to extinction.

genome size, amount of non-coding DNA and metabolic redundancy were significantly reduced. However, if ecotype A significantly reduced its uptake diversity and increased its production diversity, ecotype B evolved in an opposite way (*i.e.*, ecotype B significantly increased its uptake diversity and reduced its production diversity). The traits of ecotypes A and B diverged: **(i)** ecotype A strongly specialized on m_{exo} (with a mean uptake diversity of 1.17 metabolites), and optimized metabolic fluxes according to the trade-off between avoiding lethal toxicity thresholds and maximizing the score (as explained above, this selective pressure resulted in reduced genome size and metabolic redundancy, and increased production diversity). **(ii)** Ecotype B specialized on by-products by increasing the uptake diversity and reducing the production diversity. This demonstrates that ecotypes A and B specialized to their own niches, and that ecotypes A and B traits diverged by character displacement, in complete agreement with the LTEE (Legac et al., 2012; Großkopf et al., 2016).

Table IV.5 – Comparison of the genome and metabolic network structure of initial ecotypes A and B in the early and late populations. For each ecotype, five variables were evaluated: **(i)** the mean genome size, **(ii)** the mean amount of non-coding DNA, **(iii)** the mean number of E encoding the same metabolic reaction (the "metabolic redundancy"), **(iv)** the mean number of different essential metabolites pumped in (the "uptake diversity"), and **(v)** the mean number of different essential metabolites produced (the "production diversity"). The standard deviation is also shown (mean ± sd.). For each measure, we performed a two-sample Wilcoxon test with Bonferroni correction ($n = 5$).

Variable	Early pop.	Late pop.	Wilcoxon test	Units
A genome size	257.09 ± 42.31	189.99 ± 49.85	***	Genome units
A non-coding DNA	6.37 ± 1.14	4.69 ± 2.10	**	Genome units
A metabolic redundancy	8.20 ± 0.83	6.69 ± 1.46	***	Genome units
A uptake diversity	3.12 ± 0.51	1.17 ± 0.54	***	Metabolites
A production diversity	6.82 ± 0.12	7.14 ± 0.73	*	Metabolites
B genome size	274.93 ± 31.98	203.14 ± 59.59	***	Genome units
B non-coding DNA	9.70 ± 8.21	4.20 ± 2.05	***	Genome units
B metabolic redundancy	9.86 ± 2.48	8.04 ± 2.54	**	Genome units
B uptake diversity	3.74 ± 0.76	4.59 ± 0.84	***	Metabolites
B production diversity	6.26 ± 0.70	5.85 ± 0.57	**	Metabolites

Character displacement explained the apparent robustness of the A/B interaction in late populations. Indeed, niche specialization led A organisms to specialize on m_{exo} and increase their production diversity. On the other hand, B organisms specialized on a large number of by-products. However, character displacement and niche specialization was stronger in late than early populations. For this reason, late A organisms needed more mutations (and then more evolution time) to adapt to continuous conditions (*i.e.*, reducing the production diversity and increasing metabolic redundancy, see Table IV.1) than early A organisms. Thus, B organisms were slower out-competed in late than early populations.

To assess those conclusions, we studied in details the evolution of ecotypes A and B in the 10 repetitions of the early population 3 (Fig. IV.10A.3), in the exact same way than in Figure IV.12. The result is available on Figure IV.8.4, and shows that A organisms from early populations reduced their production diversity (mean of 3.4 metabolites) more than A organisms in late populations (mean of 2.0 metabolites). Indeed, A organisms were less specialized and thus needed less time to adapt to the continuous conditions, thereby favoring the extinction of ecotype B.

The fact that beneficial mutations from ecotype A spread all over the population in the continuous environment (Fig. IV.4) indicates that competitive exclusion occurred. In this case, according to (Cohan, 2002), A and B groups cannot be considered as separate ecotypes in the continuous environment, although the same organisms were separate ecotypes in the periodic environment. Note that in 5 assays over all experiments, ecotype B fixed in the population (1 assay for early populations in the continuous environment, 3 assays for late populations in the continuous environment, and 1 assay for late populations in the periodic environment). When ecotype B invaded the population, by-products were not

produced anymore by ecotype A, therefore dooming the whole population to extinction (as exemplified in repetition 3 of late population 1).

Hence, the stability of the A/B cross-feeding interaction in the continuous environment relied on the evolutionary time elapsed in the periodic environment since adaptive diversification. This shows that the co-evolution of ecotypes A and B in the periodic environment strengthened their interaction, meaning that niche specialization stabilized the cross-feeding and fostered a robust negative frequency-dependence. However, even if the cross-feeding interaction seemed stable over few thousands of generations, in the continuous environment a beneficial mutation in ecotype A lineage can lead to the extinction of ecotype B lineage. Therefore, the stability of the A/B polymorphism in the periodic environment did not rely only on their cross-feeding interaction, but also on the seasonality of the environment.

IV.6 Discussion

Using *in silico* experimental evolution, we have shown that the long-term maintenance of cross-feeding interactions is favored in a seasonal environment, where the environment is reset and primary resource is supplied at regular intervals. In this environment, 5 simulations over 12 evolved a stable cross-feeding interaction at the end of the simulations, with two monophyletic ecotypes coexisting via a negative frequency-dependent interaction. At each cycle, ecotype A grows during the first season, feeding on the primary resource and releasing by-products, while ecotype B exclusively feeds on by-products during the second season. The stable coexistence of ecotypes A and B is then based on niche construction, followed by a negative frequency-dependent interaction, as the S and L ecotypes in the LTEE. According to our model, batch culture experiments seem to especially favor the evolution of stable cross-feeding polymorphisms owing to the cyclic nature of the environment that generates the conditions for the existence of at least two stable seasons: a first season is externally generated by the cyclic mechanism (thus being intrinsically stable) while the second one is generated by the replacement of the exogenously-provided nutrient by the secreted by-products through a mechanism of niche construction.

In the continuous environment, where the primary resource is constantly provided (like in a chemostat), cross-feeding interactions emerged, but were not stable because of competitive exclusion. In this case, organisms enriched their environment via their metabolic activity, such that mutants were temporarily able to feed on by-products. But the absence of seasonality precludes any possibility for the stabilization of cross-feeding interactions.

Our multi-scale model allowed us to investigate the impact of resource dynamics on the organization of genome (*e.g.*, gene amplification) and of the metabolic network. It also allowed us to dissect the precise mechanism behind the evolved robustness of the cross-feeding interaction. We demonstrated that those results are robust to model parameter variation. Indeed, stable cross-feeding interactions emerged in the periodic environment

for a wide range of parameter values, including well-mixed populations and infinite diffusion rate, while they never appeared in the continuous environment, thus reinforcing our conclusions (Appendix IV.8.5).

Previous wet experiments in chemostat demonstrated the emergence of cross-feeding interactions (Helling et al., 1987; Rosenzweig et al., 1994; Treves et al., 1998). In those experiments, *E. coli* populations have been propagated in a chemostat with glucose as a single limiting carbon source for at most 1,900 generations. When isolated and evolved together in competition experiments, the different mutants identified to contribute to the cross-feeding interactions reached a stable equilibrium owing to frequency-dependent interactions (Rosenzweig et al., 1994). Several reasons were invoked to explain why cross-feeding interactions could be stable in chemostat, despite the competitive exclusion principle. According to Pfeiffer and Bonhoeffer (2004), cross-feeding may evolve in microbial populations as a consequence of the maximization of ATP production, and the minimization of enzyme concentrations and intermediate products. Those constraints may hinder the emergence of mutants completely degrading glucose (or uptaking glucose and acetate), and outcompeting other cells by competitive exclusion. In our model, organisms do not need to explicitly produce energy carriers. However, competition for resources, toxicity thresholds and division impose metabolic flux optimization. Based on the same conclusions, Doebeli (2002) also suggested that this trade-off between uptake efficiency on the primary and the secondary resources should favor the emergence of cross-feeding polymorphism in chemostat but not in batch culture, because in a chemostat, by-products are more abundant and constantly provided. However, the limit of this model is that the rate of by-product production did not rely on the rate of primary resource consumption. Besides, a more recent theoretical work concluded that, in a continuous and well-mixed environment, the diversity of cross-feeding polymorphism was negatively correlated with primary resource abundance (Gerlee and Lundh, 2010a).

Our results shed a new light on this question. First, in our model, cross-feeding polymorphisms emerged both in the periodic and continuous environments. However the stabilization of the cross-feeding interactions was favored in the periodic environment, leading to the evolution of specialized ecotypes. Cohan (2002) defined an ecotype as an independent monophyletic cluster occupying a specific ecological niche. Ecotypes are at the heart of the bacterial species concept: what makes the genetic cohesion of an asexual bacterial species is periodic selection that regularly purges the genetic diversity in the same ecological niche (Cohan, 2002). As a consequence, ecotypes occupying different niches independently experience selective sweeps, the mutants from one niche not invading the ones from the other niche. Thus, the stability of a cross-feeding polymorphism should only be analyzed in the light of the robustness of each ecotype against selective sweeps by other ecotypes (Cohan, 2002). This mechanism is observed in the LTEE, as well as in our model. In the periodic environment, ecotypes A and B independently experience periodic selection events. In the continuous environment, competitive exclusion implied that only one ecotype evolved in this environment.

Secondly, when ecotypes A and B evolved in the periodic environment were transferred in the continuous environment, they retained their negative frequency-dependent interaction

for hundreds of generations, until a selective sweep purged the whole population diversity, and destroyed the cross-feeding interaction. Moreover, ecotypes A and B that evolved for a long time in the periodic environment had a more robust interaction in continuous conditions, because of niche specialization and character displacement on the long-term. In the light of those results, we suggest to distinguish between ecological stability and evolutionary stability. Even if different monophyletic clusters, related by cross-feeding interactions, have frequency-dependent interactions, they are not necessarily robust to competitive exclusion on the long-term. In this sense, ecotypes A and B are no longer ecotypes in the continuous environment. By contrast, in the periodic environment, A and B ecotypes can be considered as proto-species.

Those remarks lead us to hypothesize that the S and L interaction observed in the LTEE, which is still at an early stage, should not be stable in a chemostat on the long-term, even if it could become more and more stable. We also hypothesize that the S/L polymorphism is an ongoing speciation event. On the long run, the S ecotype could even loose the ability to consume glucose.

In a more general view, what we observed is strongly related to known results about temporal niche partitioning in ecology (Spencer et al., 2007). Bacterial communities commonly undergo adaptive diversification or niche specialization in sympatry, when the environment is seasonal. For example, this mechanism has been observed in marine microbial communities (Gilbert et al., 2012), and in lake phytoplankton (Grover, 1988). In the LTEE (Rozen et al., 2009) and in our model, seasonality of glucose originates from the serial transfer, but the seasonality of acetate is due to cross-feeding and niche construction. Moreover, we demonstrated in our model than negative frequency-dependent cross-feeding is not enough to stabilize the interaction between multiple ecotypes. External factors are necessary, such a regular serial transfer. While the environment is intentionally simplified in those experiments, we can expect much more complex environmental conditions in nature.

Such complex interactions between external factors, emergent cross-feeding interactions and niche construction are therefore of primary importance to understand the evolution of microbial communities in well-mixed environments. Using a computational model of ISEE to decipher those interactions seems to be a rich complementary approach to wet experiments and mathematical modeling.

IV.7 Conclusion

Using a multi-scale computational model of ISEE, we studied the evolution and stability of cross-feeding interactions in well-mixed environments, providing a single limiting resource periodically or continuously, as in batch cultures or chemostat devices. Our results led us to consider a stable cross-feeding polymorphism as the stable coexistence of different ecotypes, defined as different monophyletic clusters undergoing independent periodic

selection events in their own ecological niche (Cohan, 2002). We observed that, even if cross-feeding polymorphism systematically appears in all the simulations, the evolution of stable ecotypes coexisting via cross-feeding is favored in the periodic environment, similarly to the S/L polymorphism observed in the LTEE (Rozen and Lenski, 2000). In the continuous environment, competitive exclusion precludes the stabilization of cross-feeding interactions, in apparent contradiction with wet experiments. Indeed, while ecotypes interacting via cross-feeding can temporarily coexist, a mutant always eventually outcompetes them. Then, we suggest to study the evolution of cross-feeding polymorphism by fully integrating the notion of ecotype, and distinguishing between ecological stability and evolutionary stability, the latter including long-term evolutionary dynamics such as periodic selection. Our results contributed to understand temporal niche partitioning, by modeling various mechanisms such as cross-feeding, niche construction and seasonality. At a more general scale, our results may contribute to the study of the evolution of bacterial communities, by deciphering the conditions of sympatric speciation in asexual populations.

IV.8 Supporting Information

IV.8.1 Table S1. Common simulation parameters for the entire experimental protocol.

Parameters for the initialization of genomes	Value	Unit
Initial number of non-coding units (NC)	10	genomic-units
Initial number of promoter units (P)	10	genomic-units
Initial number of enzyme units (E)	10	genomic-units
Range for the random drawing of β in initial genes	$[0, 1]$	ACU.centi-time-step^{-1}
Range for the random drawing of s and p in initial genes	#1 to #20	dimensionless
Range for the random drawing of k_{cat} in initial genes	$[10^{-3}, 10^{-1}]$	centi-time-step^{-1}
Range for the random drawing of k_{cat}/K_M ratio in initial genes	$[10^{-5}, 10^{-4}]$	centi-time-step^{-1}.ACU^{-1}
Parameters of the intracellular dynamics	Value	Unit
Duration of one population time-step	100	centi-time-steps
Protein degradation rate ϕ	0.1	centi-time-step^{-1}
Non essential metabolites toxicity threshold	1.0	ACU
Essential metabolites toxicity threshold	1.0	ACU
Minimum score	10^{-3}	ACU
Parameters of population dynamics	Value	Unit
Total simulation time	500,000	time-steps
Grid width W	32	gridsteps
Grid height H	32	gridsteps
Death probability p_{death}	0.02	organism^{-1}.time-step^{-1}
Metabolite tag of the primary resource m_{exo}	#10	dimensionless
Diffusion parameter D	0.1	gridstep^2.time-step^{-1}
Parameters of point mutations	Value	Unit
Point mutation rate	1e-03	attribute^{-1}replication^{-1}
Substrate tag mutation size	1	dimensionless
Product tag mutation size	1	dimensionless
$log(k_{cat})$ tag mutation size	0.1	dimensionless
$log(k_{cat}/K_M)$ tag mutation size	0.1	dimensionless
β mutation size	0.1	ACU.centi-time-step^{-1}
Probability that a genomic unit changes type	1e-03	genomic-unit^{-1}.replication^{-1}
Parameters of genomic rearrangements	Value	Unit
Duplication rate	1e-03	genomic-unit^{-1}.replication^{-1}
Deletion rate	1e-03	genomic-unit^{-1}.replication^{-1}
Translocation rate	1e-03	genomic-unit^{-1}.replication^{-1}
Inversion rate	1e-03	genomic-unit^{-1}.replication^{-1}
Probability of attribute swap at breakpoint	1e-03	attribute^{-1}.breakpoint^{-1}
Maximum genome size	10000	genomic units

Those parameters are common to all the simulations of the experimental protocol.

IV.8.2 Figure S1. Final phylogenetic trees of each simulation.

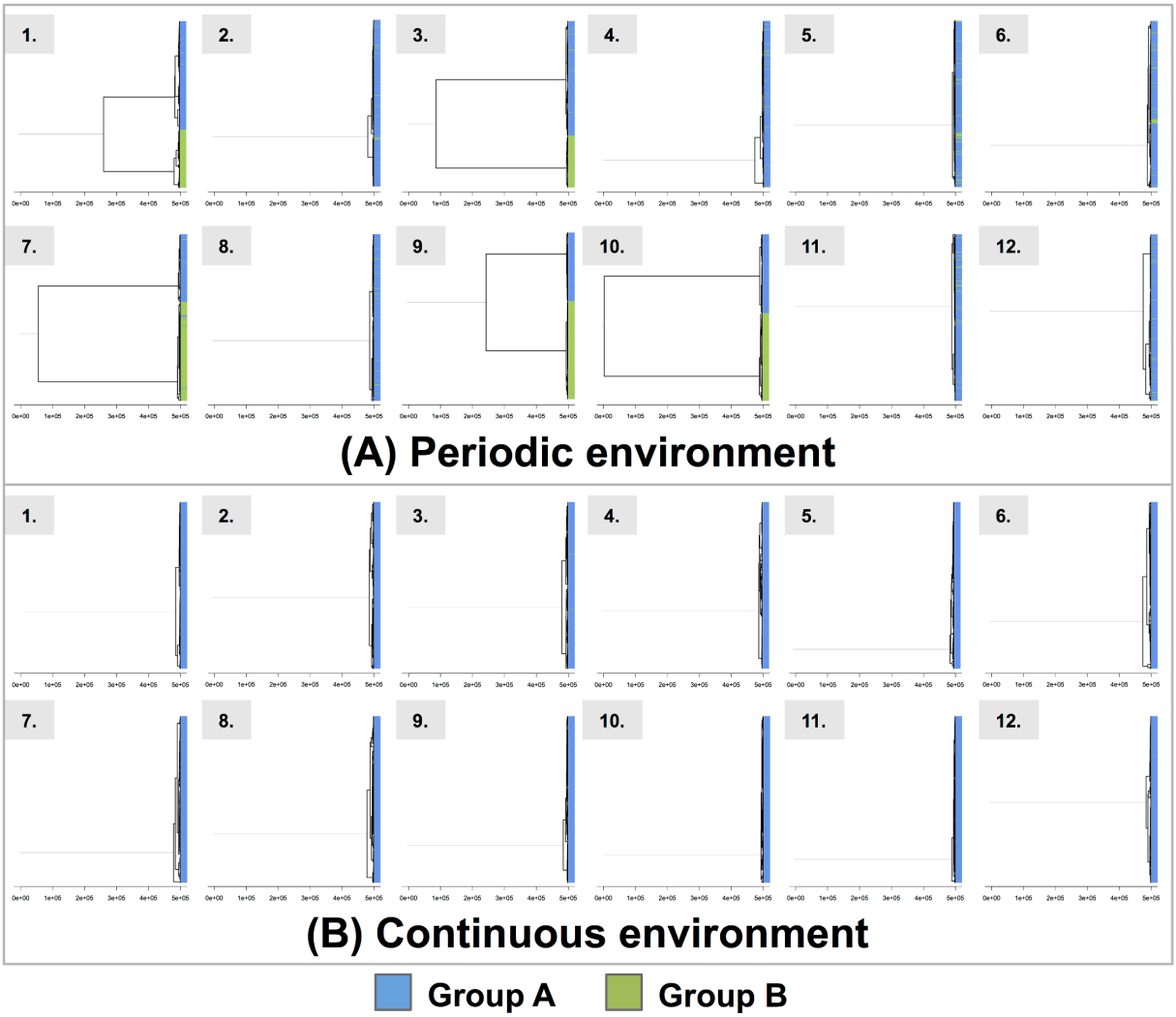


Figure IV.13 – Final phylogenetic trees of each simulation. (A) Phylogenetic trees of the 12 repetitions in the periodic environment. (B) Phylogenetic trees of the 12 repetitions in the continuous environment. Tree leaves are colored depending on their trophic group: group A in blue, group B in green. Phylogenetic trees are numbered by repetition. For each tree, the scale is represented in simulation time-steps.

IV.8.3 Figure S2. Evolution of the MRCA age during simulations, for the three types of environments.

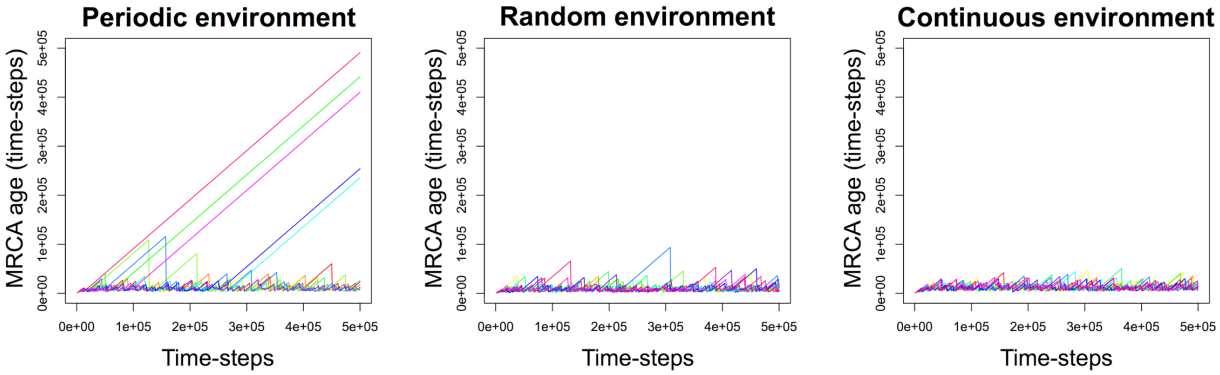


Figure IV.14 – Evolution of the MRCA age during simulations, for the three types of environments. For each environment, all the repetitions are represented in different colors. **(A)** Periodic environment. **(B)** Random environment. **(C)** Continuous environment.

IV.8.4 Figure S3. Loss of essential metabolites production of ecotype A organisms, in the 10 repetitions of the early population 3 in the continuous environment assays.

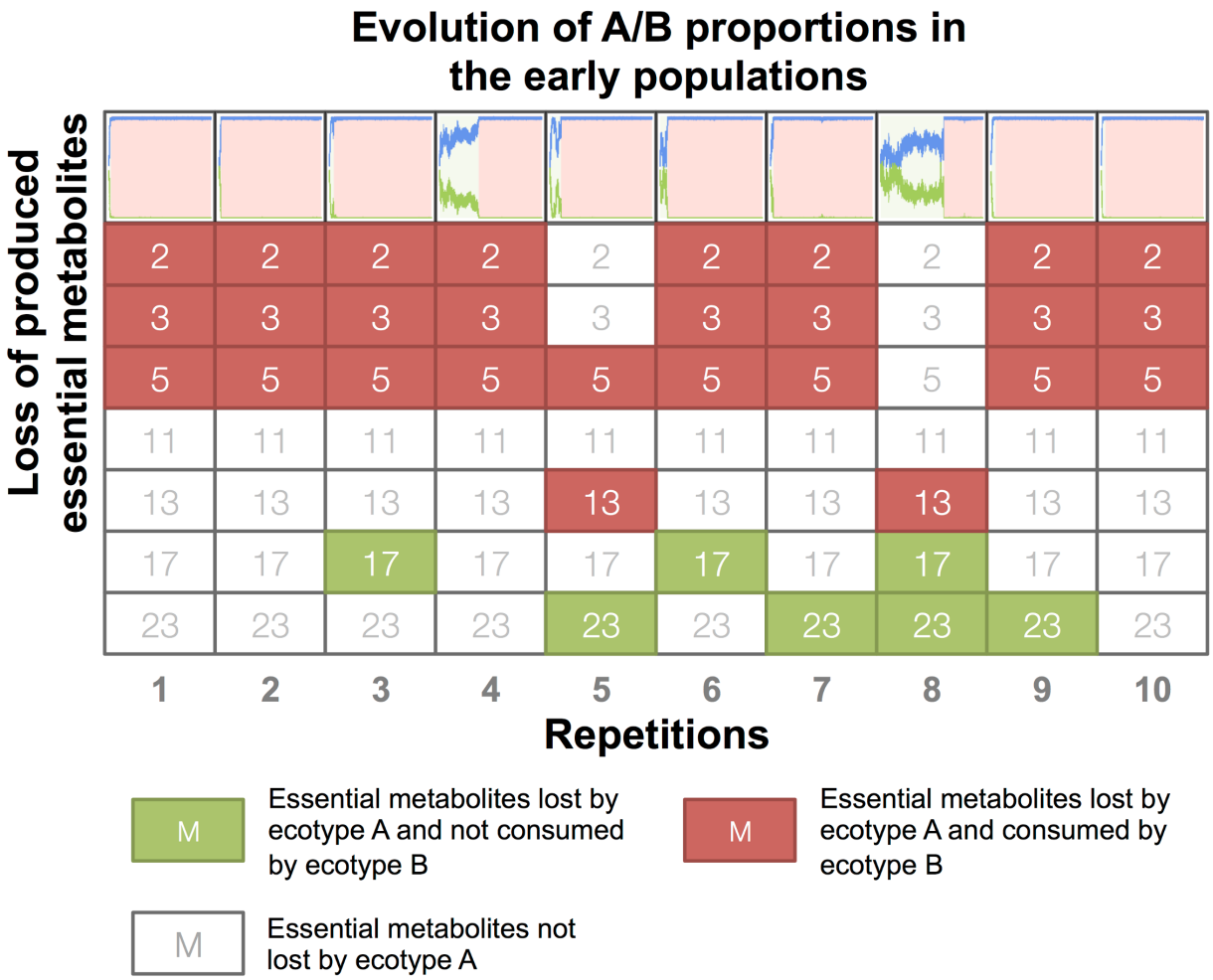


Figure IV.15 – Loss of essential metabolites production of ecotype A organisms, in the 10 repetitions of the early population 3 in the continuous environment assays. The 10 repetitions of population 3 are displayed. The 7 essential metabolites (2, 3, 5, 11, 13, 17, and 23) that were produced by ecotype A organisms at the beginning of the assays are represented vertically for each repetition. Background colors indicate a production loss. Essential metabolites that are consumed by ecotype B organisms are colored in red, the other in green. At the top, the evolution of groups A and B proportions is represented, and is colored in green when the A/B interaction persisted, or in red when the interaction failed. In all simulations where A have ceased to produce a metabolite pumped-in by B, B has gone to extinction.

IV.8.5 Appendix S1. Sensitivity analysis on six key parameters.

To evaluate the robustness of our results to the variation of main simulation parameters, we performed a sensitivity analysis. We explored six key parameters, selected for their importance and their influence on evolutionary dynamics:

(1) The death probability p_{death}, that controls the probability to die at each simulation time-step (the same for every individuals, see "Population and environment" section). We explored p_{death} around the default value ($p_{death} = 0.02$), *i.e.*, $p_{death} = 0.005$ and $p_{death} = 0.1$. Because this parameter is highly sensitive, we also explored intermediate values, *i.e.*, $p_{death} = 0.01$ and $p_{death} = 0.05$

(2) The diffusion rate (see "Population and environment" section). We explored the diffusion rate around the default value (diffusion parameter $D = 0.1$ gridstep^2.time-step^{-1}), *i.e.*, $D = 0.02$ gridstep^2.time-step^{-1} and the special condition of a perfectly well-mixed environment.

(3) The mutation rates (see "Genome structure" section). We explored the mutation rates around the default value (10^{-3}), *i.e.*, 2.10^{-4} and 5.10^{-3}.

(4) The toxicity thresholds, that impose a lethal upper threshold to internal cell's metabolic concentrations. We explored the mutation rates around the default value (1.0), *i.e.*, 0.1 ACU (Arbitrary Concentration Unit) and 5.0 ACU.

(5) The "migration rate" r_{mig}: this parameter controls, at each time-step, the fraction of exchanged pairs among all possible pairs of individuals. As competition is local, this parameter thus controls whether an individual directly compete with its siblings or with more distantly related individuals. By default, the migration rate is $r_{mig} = 0.0$. We then explored this parameter with to higher values: $r_{mig} = 0.5$ (half of the locations are randomized) and $r_{mig} = 1.0$ (every locations are randomized).

(6) The grid size. The default grid size being 32×32, we tested sizes 20×20 and 50×50.

Because of computational loads, we varied each parameter separately around our default parameters set. We computed 10 repetitions for each set of values (with a total of 140 simulations of 500,000 time steps and 1.5 months of computation).

To assess whether a A/B-like stable polymorphism evolved in a given run, we analyzed the phylogenetic tree of the final population and computed both the PS score (see the paragraph on cross-feeding interactions in the manuscript) and the time to the most recent common ancestor (MRCA age). As in Fig 5, we considered that a A/B stable polymorphism had evolved if **(i)** the MRCA age was higher than 200,000 time-steps, which indicates the existence of long-diverged clades, and **(ii)** the PS score was higher than 0.9, which indicates that clades match well with ecotypes (ecotypes are monophyletic). The result of the sensitivity is presented in Fig 1.

• First, no stable polymorphism evolved in the continuous environment, whatever the parameter values, while it regularly evolved in the periodic environment, thereby supporting our main conclusion.

• Second, in the periodic environment, some parameters are more sensitive than others in the periodic environment.

(i) For example, a lower or higher death probability (Figs 1A.1 and 1B.1) inhibits the emergence of stable polymorphism. Interestingly, we calibrated the death probability (0.02 per organism per time-step) and the duration of a cycle in the periodic environment (333 times-steps) to obtain in theory 6.67 generations per cycle - since $333 * 0.02 = 6.67$ - like in the LTEE. A lower death probability exposes individuals to several seasons, facilitating the evolution of generalists. On the opposite, a higher death rate forbids the survival of B individuals, that would have too short a lifespan to survive the famine that (for them) necessarily follows the environment refresh.

(ii) Lower or higher toxicity thresholds (Figs 1A.4 and 1B.4) strongly influence the structuring of the metabolic network, and then the ability for the A/B interaction to be stable (see part "Stability of the A/B cross-feeding interactions in the continuous environment" of the manuscript for details).

(iii) The variation of the mutation rates (Figs 1A.3 and 1B.3) also influences the outcome of the simulations. Stable polymorphism is observed at higher mutation rate but not at a lower one which may be due to a too slow evolution rate compared with the duration of the experiment (structured trees are indeed observed at a low mutation rate but the MRCA ages remain low).

• Third, and most importantly, the exploration of the diffusion rate (Figs 1A.2 and 1B.2) and of the migration rate (Figs 1A.5 and 1B.5) reinforces our conclusions. Indeed, previous theoretical studies highlighted the fact that spatial structure could inhibit the emergence of stable polymorphism, while well-mixed conditions could enhance it (Hauert and Doebeli, 2004; Gerlee and Lundh, 2012). In the case of the periodic environment, there is a clear correlation between the rate of diffusion and the number of simulations exhibiting stable polymorphism ($D = 0.02 \rightarrow 0\%$, $D = 0.1 \rightarrow 42\%$, well-mixed$\rightarrow 80\%$). Conclusions are the same for the migration rate ($r_{mig} = 0.0 \rightarrow 42\%$, $r_{mig} = 0.5 \rightarrow 50\%$, $r_{mig} = 1.0 \rightarrow 100\%$). This result is in agreement with previous studies (Hauert & Doebeli, 2004 ; Gerlee & Lundh, 2012). However, there is no stable polymorphism in the continuous environment in any cases, thus reinforcing our conclusion that stable polymorphism cannot evolve on the long-term in chemostat-like environments because of competitive exclusion.

• We also explored the grid size, by decreasing the grid size to 20×20, and increasing it to 50×50 (Figs 1A.6 and 1B.6). A small grid size (20×20) seems to inhibit the emergence of a stable polymorphism. Indeed, the population size is probably too small to sustain the polymorphism. In the large grid, we observed a slightly less stable polymorphism (30%) than in the default grid (42%), but the difference is not significant.

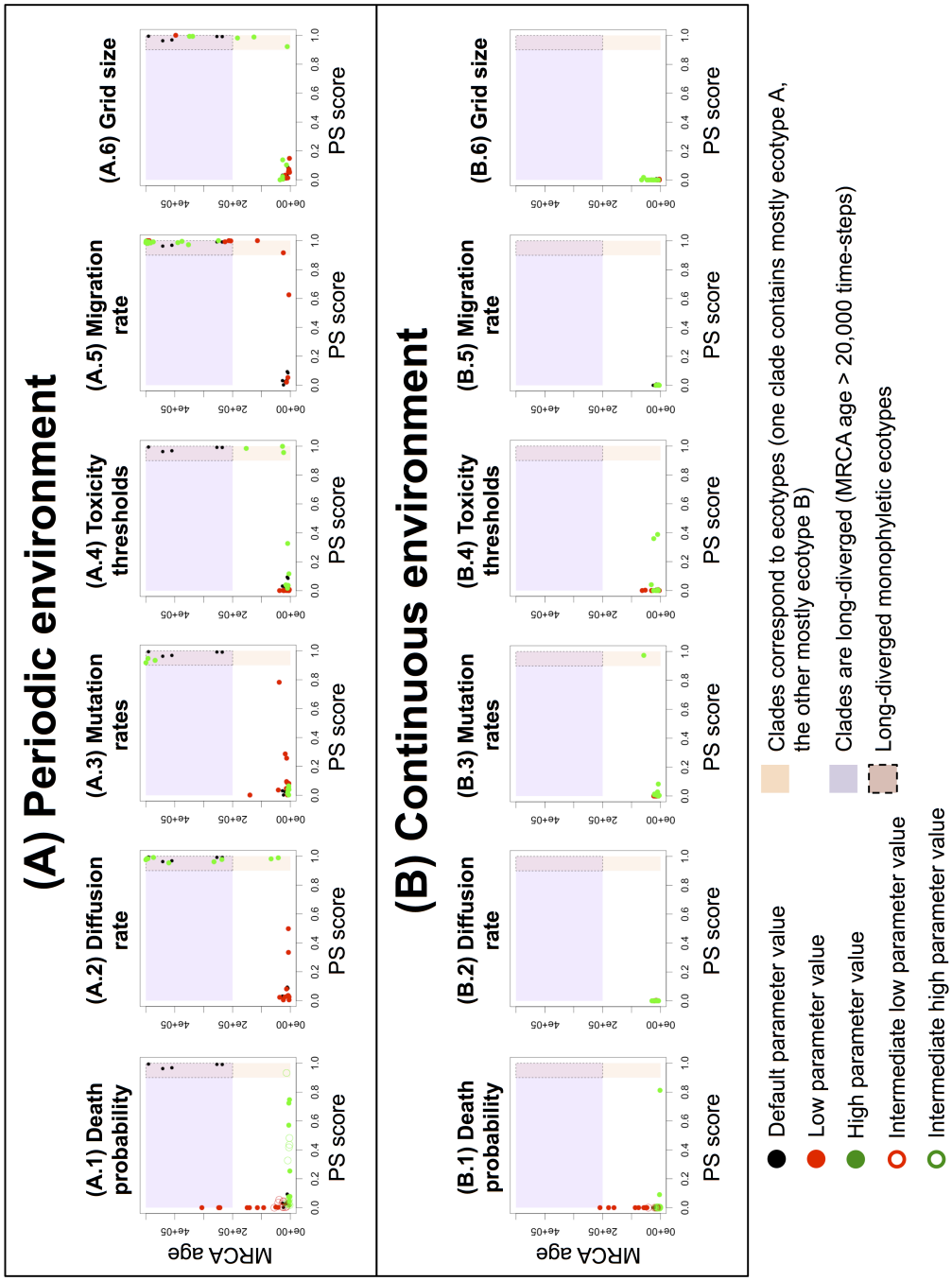


Figure 1 - **Sensitivity analysis on six key parameters of Evo²Sim.** Six parameters are explored: **(i)** the death probability p_{death}, **(ii)** the diffusion rate, **(iii)** the mutation rates, **(iv)** the toxicity thresholds, **(v)** the migration rate and **(vi)** the grid size. For each type of environment, the PS score is plotted against the MRCA age for the final trees (at 500,000 time-steps): **(i)** default parameter value simulations are plotted in black, **(ii)** low parameter value simulations are plotted in red, **(iii)** high parameter value simulations are plotted in green. **(iv)** For p_{death}, intermediates low and high values are also explored (resp. red and green circles). On each plot, three areas are identified: purple area indicates long-diverged clades (MRCA age higher than 200,000 time-steps), orange area indicates that clades correspond to ecotypes (PS score > 0.9). The intersection of the two previous areas (inside dashed borders) indicates long-diverged monophyletic ecotypes. **(A)** Periodic environment. **(B)** Continuous environment.

IV.8.6 Video S1. Variation of the relative fitness of ecotype B during an entire cycle.

Variation of the relative fitness of ecotype B during an entire cycle. This video shows the evolution of the ecotype B relative fitness during the first cycle of the short-term competition experiment. Each of the 333 frames corresponds to one time-step, the whole video presenting the entire cycle.

https://doi.org/10.1371/journal.pcbi.1005459.s006

Chapter V

Why do cells regulate? The fate of genetic regulation in an energy-limited cell's model

The results presented in this chapter are preliminary and unpublished.

> They are delighted he's going to eliminate regulations, let them make more profit; of course, it'll lead to another crash, but that's somebody else's problem. (Noam Chomsky, 2017)

V.1 Introduction

While studied for decades, the role of genetic regulation in cellular functions and its evolution are still not completely understood (Savageau, 1998; Shinar et al., 2006). In living systems, many metabolic pathways are controlled by enzymes, whose expression levels are under regulation. The best known example of such a regulation is probably the lactose operon (Jacob and Monod, 1961). When lactose is absent, the transcription of β-galactosidase, which degrades lactose in glucose and galactose, is repressed. When lactose is detected in the local environment, the repressor is inhibited, and β-galactosidase enzymes are produced.

It is often assumed that genetic regulation evolved to optimize cellular metabolism in variable environments. Many modeling tools in systems biology are based on this assumption. This is for example the case of **flux balance analysis** (FBA, Orth et al. 2010). FBA is based on mathematical optimization algorithms to find metabolic flux rates that maximize the production of one or more metabolites in a specific metabolic network, often linked to cellular fitness (*e.g.* ATP production). FBA assumes that the cell is able to finely regulate its metabolic activity depending on some constraints (*e.g.* the availability of a resource). For example, FBA has been used to study the evolution of cross-feeding in bacterial populations (Großkopf et al., 2016). Yet, this interpretation of the role of genetic regulation is undermined by recent works showing that "*the regulation of metabolic pathways may have evolved not to match expression of enzymes to levels of extracellular substrates in changing environments but rather to balance a trade-off between exploiting one type of nutrient over another*" (Weiße et al., 2015). Indeed, living systems do not escape thermodynamic laws. Energy and resource allocation to various cell components are an essential limitation to cell's activity. Weiße et al. (2015) recently showed with an *in silico* model of evolution that regulation may not evolve to adjust gene expression to external resource concentrations, but to balance internal trade-offs. Indeed, they showed that the best strategy in an environment providing a single nutrient is not to adjust gene expression to external nutrient concentrations, but to constitutively express enzymes to metabolize this nutrient at their maximum value, according to internal trade-offs. In environments providing two nutrients, the best strategy depends on the uptake efficiency of the cell for each nutrient. In this case, the expression level of enzymes metabolizing each nutrient only depends on relative uptake efficiencies, and not on external nutrient concentrations.

In Evo^{2}Sim, genetic regulation, while freely evolvable, did not emerged in typical con-

ditions, as shown in chapter IV. Indeed, in Rocabert et al. (2017), the cell model is energy-free, meaning that regulation and metabolic networks evolve without any energy allocation trade-offs. In the complex situation of niche construction and the evolution of stable cross-feeding, none of the simulations shown evolved functional genetic regulation networks, while resource fluctuations were significant. The evolution of efficient metabolic pathways thus seems sufficient in these conditions to regulate the metabolic activity, without the intervention of genetic regulation.

In unpublished preliminary experiments with Evo^{2}Sim, C. Knibbe initialized simulations with digital organisms owning carefully handcrafted regulation and metabolic networks, and undergoing energy constraints on their metabolism. She showed that even in tough environments and highly interdependent metabolic and regulation networks, digital organisms lost their regulation network and evolved a metabolic network with constitutively expressed enzymes. Moreover, organisms losing the regulation network had a better fitness than the ones keeping a finely regulated metabolism[1]. The study presented in this chapter is based on this preliminary work.

We first parameterized Evo^{2}Sim with realistic values when it was possible, for two reasons: **(i)** it is impossible to explore the whole parameter space of Evo^{2}Sim. We thus need an heuristic linked to our scientific question: in the case of an energy-limited model, realistic values are a good choice. **(ii)** Evo^{2}Sim is a multi-scale model including the interaction of many objects and structures. To obtain appropriate results and avoid artifactitious dynamics, the different objects and structures must be parameterized in a coherent way. This can be done by setting all parameters in the same orders of magnitude as in real bacteria, when possible.

Using this parameter setting, we tested two specific environmental and cell model conditions. The first environment alternatively provides two resources: the metabolites #20 and #22. We called this environment **env. A**. We also ran a complementary experiment in a second type of environment (**env. B**), similar to the first one but providing more resources. In the first model condition, the production of proteins costs energy. In the second one, this energetic cost is relaxed. However, in both conditions, pumping activity and anabolic reactions (*i.e.* the production of metabolites with higher metabolite tag than the source metabolite, see chapter III) consume energy, while catabolic reactions produce energy. Initial digital organisms own handcrafted genomes coding for carefully designed genetic regulation and metabolic networks (see below). When protein production energy costs exist, regulation is mandatory to survive, because it avoid the depletion of energy carrier molecules, as described below.

Our results show that the presence of protein production costs led to the evolution of "virus-like" organisms, having a small genome coding for a single operon, with no non-coding DNA. This operon codes for both regulation and metabolic networks. Doing so, digital organisms limit energy consumption in time by producing all proteins at once. In the absence of protein production costs, digital organisms evolved larger genomes with

[1]This work has been presented to the *Workshop of the International Laboratory EvoAct (Evolution in action)*, Autrans, April 2016.

multiple small operons, half of the genome being non-coding. These organisms completely lost regulation. Thus, our results suggest that protein production costs strongly influence the evolution of genome structure and regulation in Evo^{2}Sim. This preliminary work calls for further experiments in the model to assess the functional nature of genetic regulation.

V.2 Methods

V.2.1 Realistic parameterization in Evo^{2}Sim

The section V.2.1 is largely inspired from the final report of the EvoEvo project, available at www.evoevo.eu.

Evo^{2}Sim contains many parameters that must be set at the beginning of each simulation (chapter III and Appendix A). In the case of the study of internal cellular trade-offs, in order to compare the results of a simulation with *in vivo* experiments, model parameters must be tuned to fit typical values found in living cells, for protein and metabolic concentrations, cellular lifespan, enzymatic constants, and so on. Moreover, since parameters are interdependent, one must set the correct order of magnitude of each parameter to avoid the emergence of purely artificial dynamics in the model. For example, metabolic reactions must be fast enough to enable the cell to react to environmental changes, but too fast reactions must be avoided since they would not be possible in practice. We identified the correct order of magnitude for most of the parameters of Evo^{2}Sim. They are presented below.

• **Time units.** Parameters related to internal molecular processes were expressed per minute. At each simulation time-step and for each digital organism, the internal dynamics were computed by a time adaptive numerical solver (chapter III) from $t = 0$ to $t = 100$ minutes. The population dynamics was thus updated every 100 minutes depending on the current state of each cell. Given this timescale, we fixed the death rate p_{death} at 0.005 per organism per 100 minutes, meaning that each cell lives on average ∼14 days.

• **Protein degradation rate.** Proteins half-life in *E. coli* vary from 2 minutes to 70 hours (Maurizi, 1992), depending on the protein (the proteins with low half-life generally being mutant or badly folded ones). In Evo^{2}Sim, the protein half-life is fixed by the protein degradation rate ϕ. We used a degradation rate ϕ of 5.10^{-4} per protein per minute, corresponding to a protein half-life of ∼24 hours.

• **Protein concentration units.** Following known values in model bacteria[1], we fixed the cell volume and the grid patch volume to 4 μm^3, corresponding to the estimated volume of an *E. coli* cell. In order to properly scale the concentrations, we chose the protein production rate as a relative reference. Let's first define the unit Z such that 1 Z = 10^{-8} M. In EVO^{2}SIM, the protein production rate varies between 0 and 1. Given the protein degradation rate ϕ, the maximum protein concentration at equilibrium is $1/5.10^{-4} = 2000$ Z. In *E. coli*, enzymatic concentrations are estimated to vary between 5 and 500 nM. Hence, to fit these values, we considered that 1 Z was equal to 10^{-8} M, the protein concentrations thus varying between 0 and 20000 nmol/L. When a protein concentration is below than 1 Z, we consider that it has disappeared from the cell (since a concentration lower than 1 Z corresponds to less than one molecule in the cell).

• **Metabolic concentration units.** In bacteria, intracellular concentration of metabolites varies between 10^{-7} and 10^{-2} M (*i.e.*, 10 to 10^6 Z). Controlled environments used to cultivate *E. coli* usually contain between 1 and 20 g/l of glucose, corresponding to metabolic concentrations of 5.10^5 and 10^7 Z in EVO^{2}SIM (the minimal concentration below which *E. coli* does not grow being 4∼5 g/l).

• K_M **and** k_{cat} **values.** K_M and k_{cat} are the two parameters of the Michaelis-Menten equation used in EVO^{2}SIM to model the metabolic network dynamics. These values are encoded in the genome, are enzyme-specific and evolve freely. According to Bar-even et al. (2011), in natural enzymes the observed values are usually between 10^{-7} and 10^{-1} M for K_M (10^1 to 10^7 Z), and between 6 and 60000 minute^{-1} for k_{cat} (the median value being 600 minute^{-1}). Given the range of variation of both values, they are encoded in logarithmic scale, resulting in a range of 1 to 7 for K_M and from 0.8 to 4.8 for k_{cat}. However, these values raise an unanticipated difficulty: independent mutations on K_M and k_{cat} could result in a ratio k_{cat}/K_M varying between $10^{-5.2}$ and $10^{2.8}$ min^{-1}Z^{-1}, which is a nonsense both mathematically (introducing artificial stiffness in ODEs) and biologically, since in natural enzymes, the k_{cat}/K_M ratio varies between 6.10^{-4} and 6 min^{-1}Z^{-1}. Indeed, a trade-off exists between K_M and k_{cat} values (Bar-even et al., 2011). We thus decided to parameterize the Michaelis-Menten reaction with two evolvable parameters: k_{cat} and k_{cat}/K_M ratio (see chapter III). In consequence, in EVO^{2}SIM, k_{cat} varies between 0.8 and 3.8, with a median of 2.8. Compared to realistic values (0.8 to 4.8), we restrained the range of k_{cat} to avoid very stiffed and intractable ODEs, but keeping the same median. The ratio k_{cat}/K_M varies between -3.22 and -1.22.

Apart from the realistic parameterization, the dynamics of EVO^{2}SIM model and the methodology used to solve ODE systems are exactly the ones presented in chapter III.

[1]For a global reference on the biological values, see http://bionumbers.hms.harvard.edu/.

Table V.1 – Initial genome structure. In Evo^{2}Sim, genomes are composed of **genetic units**, of five different types (promoters, binding sites, transcription factor coding units, enzyme coding units and non-coding units, see chapter III). The genetic units used for generate the initial genome in this work are listed in the right order here (there is only one strand with a single reading frame in Evo^{2}Sim). For all enzymes, $log_{10}(k_{cat}) = 2.8$ and $log_{10}(K_M/k_{cat}) = -1.22$.

Genetic unit type	Number	Main parameter values
Op. 1		
Promoter	1	$\beta = 0.5$
Binding site	1	$\text{TF}_{tag} = 1$
Transcription factor	1	$\text{BS}_{tag} = 1$; $\text{coE}_{tag} = \#20$
Enzyme (pump)	1	$s = \#20$
Enzyme	1	$s = \#20$; $p = \#5$
Non-coding	50	–
Op. 2		
Promoter	1	$\beta = 0.5$
Binding site	2	$\text{TF}_{tag} = 2$
Transcription factor	2	$\text{BS}_{tag} = 2$; $\text{coE}_{tag} = \#22$
Enzyme (pump)	1	$s = \#22$
Enzyme	1	$s = \#22$; $p = \#3$
Non-coding	50	–

V.2.2 Initial handcrafted genome structure

For all the simulations in **env. A**, we initialized digital organisms with the same handcrafted genome. As shown in Figure V.1, this genome contains two functional regions coding for two independent operons (**Op. 1** and **Op. 2**), each allowing for the production of an essential metabolite (respectively the essential metabolites #5 and #3), from two different external resources (respectively #20 and #22). As described in chapter III, enzymatic reactions #20 → #5 and #22 → #3 are catabolic and provide energy to the cell, but pumps for #20 and #22 require energy. Each operon is self-inhibiting, unless its primary metabolite is present in the environment. To this aim, each operon encodes its self-repressing transcription factor (inhibited by its co-enzyme, the primary resource), a pump for the primary resource and an enzyme to convert it into an essential metabolite (*i.e.* a prime number) (Fig. V.1a). The corresponding metabolic pathways (**metabolic pathway 1** and **metabolic pathway 2**) are rather simple: each is dedicated to the production of an essential metabolite and each is regulated by an operon (Fig. V.1b). In the case where protein production energy costs are high, this regulation scheme ensures a minimal energy consumption in the absence of the energy source. Indeed, we parameterized the protein production costs such that without such a self-inhibiting regulation pattern, digital organisms would die. The exact structure of the initial genome are shown in Table V.1.

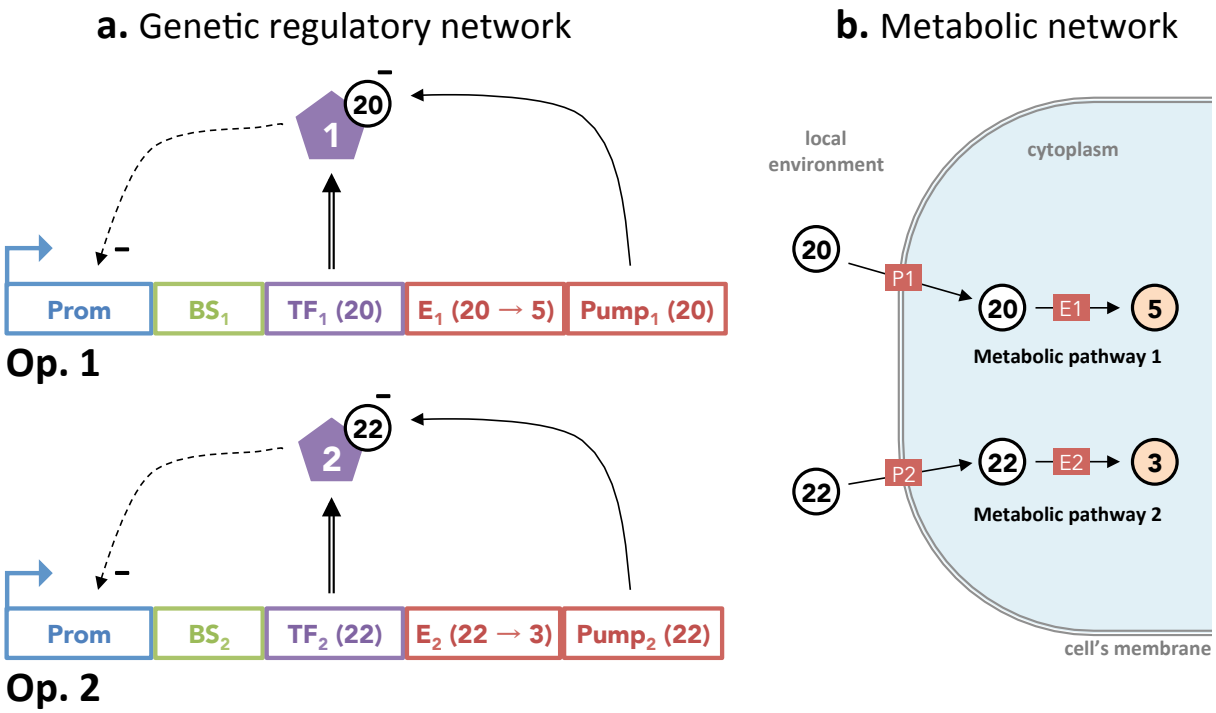


Figure V.1 – Initial handcrafted genome codes for two auto-repressed operons. a. The initial handcrafted genome contains two independent operons **Op. 1** and **Op. 2**. Each operon (**Op. 1** or **Op. 2**). Both operons contain each a promoter (blue rectangle), an operator site composed of one binding site (green rectangles), a transcription factor coding unit (purple rectangles), and two enzyme coding units (red rectangles). TF_1 and TF_2 respectively encode transcription factors 1 and 2 (purple polygons) that repress their own expression (dashed arrow), except if they are inhibited by their co-enzyme (respectively metabolites #20 and #22). Thus, each operon is self-repressed in the absence of the co-enzyme. **b.** Coding units E_1 and $Pump_1$ (respectively E_2 and $Pump_2$) encode two enzymes P1 and E1 (respectively P2 and E2), constituting **metabolic pathway 1** and **metabolic pathway 2**. Each metabolic pathway produces an essential metabolite (respectively the essential metabolites #5 and #3, dark circles filled in orange color), from two different external resources (respectively #20 and #22, dark circles), each being a co-enzyme of their respective transcription factor (TF_1 and TF_2). Thus, each operon is self-inhibiting, unless its primary metabolite is present in the environment.

V.2.3 Evaluation of the handcrafted digital organisms

To evaluate our handcrafted genomes, we run simulations in **env. A** with null mutation rates, for 500,000 time-steps. We tested the two conditions cited above, namely with or without protein production energy costs, with 10 repetitions each. In **env. A**, two external metabolites (#20 and #22) are introduced at random in each environmental grid location, following a Poisson process $\mathcal{P}(\lambda)$, with λ the introduction rate. In all simulations, $\lambda = 0.01$ per location per time-step. The degradation rate D_g is set to 0.0001 per gridspot per time-step in the first environment, and the diffusion rate D is set to 0.01 per gridstep^2 per timestep (a gridstep being the width of a gridspot).

The resulting typical cell's dynamics with protein production costs is shown in Figure

V.2. The six panels display the behavior of a single cell through time, since its birth (one time-step corresponding to 100 minutes, see above). Each time one of the external resources (#20 or #22) is present in the local environment, the corresponding operon (**Op. 1** or **Op. 2**) produces the corresponding pump and enzyme (Fig. V.2a). The cytoplasm then contains the external resource (#20 or #22) that is transformed into the corresponding essential metabolite (#5 or #3) (Fig. V.2c). Each time an operon transcription is initialized, and before energy supply from the corresponding imported resource is sufficient, small drops in energy are visible due to temporarily unfavorable energy balance in the cell (Fig. V.2e black circles). The cell's score directly depends on the concentrations of essential metabolites (Fig. V.2f). At division, the tracked cell inherits half of protein and metabolite amounts of its mother, as clearly visible on Figure V.2b. Since cell's content is released in the environment at death, the concentration of cell's final products progressively increases in the environment (Fig. V.2d).

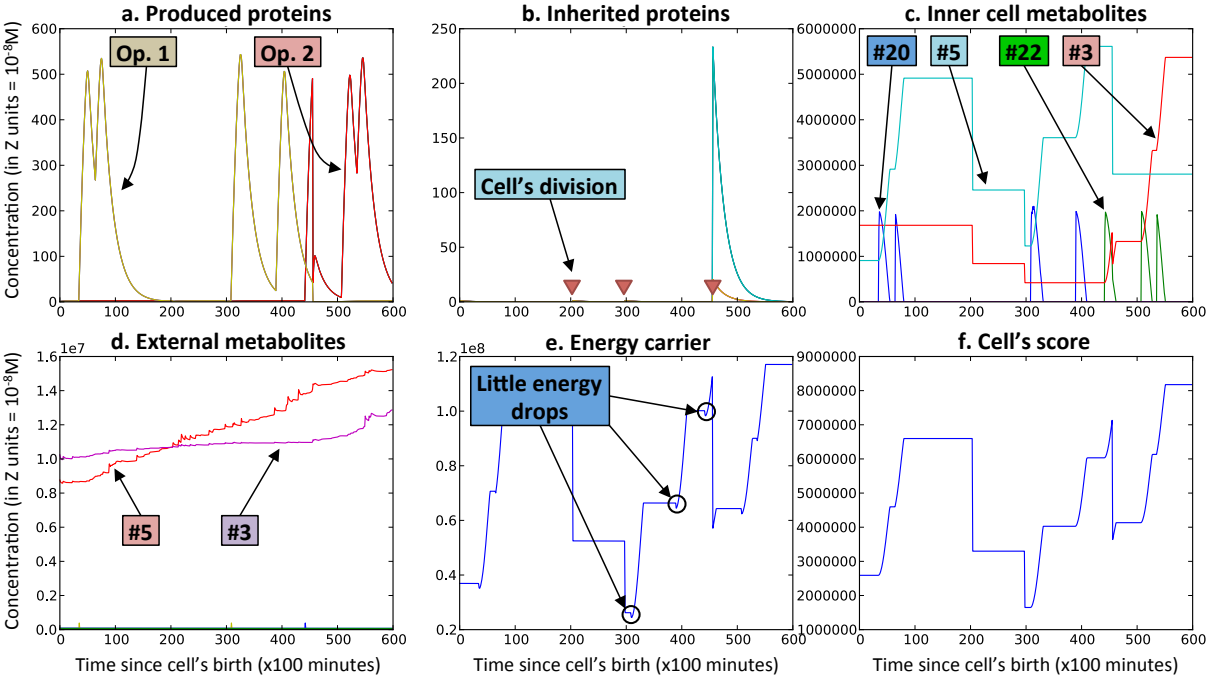


Figure V.2 – Dynamics of initial digital organisms in environment A with null mutation rates. The six panels display the behavior of a single cell through time, since its birth (one time-step corresponding to 100 minutes, see above). Each time one of the external resources (#20 or #22) is present in the local environment, the corresponding operon (**Op. 1** or **Op. 2**) produces the corresponding pump and enzyme **(panel a)**. The cytoplasm then contains the external resource (#20 or #22) that is transformed into the corresponding essential metabolite (#5 or #3) **(panel c)**. At the beginning of an operon transcription, and before energy is sufficiently produced by degrading the resource, small drops in energy are visible **(panel e black circles)**. The cell's score is directly dependent on the concentrations of essential metabolites **(panel f.)**. At division, the tracked cell inherits half of protein and metabolite amounts of its mother, as clearly visible on **panel b** (red triangles). Since cellular content is released in the environment at death, the concentration of cellular final products progressively increases in the environment **(panel d)**.

According to the cell's dynamics presented here, we were expecting that digital populations evolving in this conditions (env. A and null mutation rates) would never go to

extinction. However, most of the simulations did not reach 500,000 time-steps. As shown on Figure V.3, the extinction time was significantly higher (Wilcoxon-Mann-Whitney test gave a p-value of 0.017) for populations evolving without protein production costs (with a mean extinction time of 326,000t, 3 simulations out of 10 reached 500,000t), than populations evolving with protein production costs (mean extinction time of 160,000t, none of the simulations reached 500,000t). Two reasons explain these elevated extinction rate: **(i)** In environment A, external resources are provided at random, following a Poisson process. This could lead to prolonged periods of famine, possibly leading to whole population extinction, as it is surely the case for simulations without protein production costs. **(ii)** When protein production costs exist, energy drops at the beginning of each protein production (before the resource is sufficiently degraded to compensate for the associated energy cost) can lead to population's extinction, especially when energy level is already low (for example, after a cell division, or a prolonged famine, see Fig. V.2e). Thus, our handcrafted digital organisms are not very robust to environmental conditions as is, when no mutation occur in the genome. Extinctions occur especially early when protein production energy costs are applied (Fig. V.3).

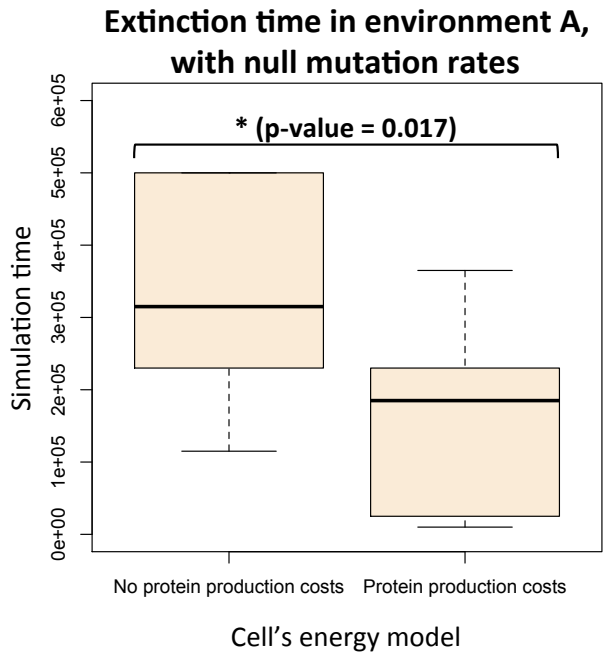


Figure V.3 – Extinction time in environment A with null mutation rates. The dynamics of handcrafted digital organisms were tested in the environment A, with null mutation rates, and in both situations where protein production costs were imposed or not. The mean extinction time of populations evolving without protein production costs is 326,000 time-steps, 3 simulations out of 10 reaching the 500,000 time-steps. The extinction time of populations evolving with protein production costs is significantly lower (160,000t in mean), a Wilcoxon-Mann-Whitney test indicating a p-value of 0.017 (none of simulations reached 500,000t). The reasons of this elevated extinction rate are due to the cellular internal dynamics, and to the environmental variability, as explained above.

V.2.4 Experimental protocol

Based on the global parameter settings and the handcrafted genome presented above, we used Evo^{2}Sim to run simulations in two different scenarios, with or without protein production energy costs.

(1) Mutation rates. Evo^{2}Sim includes seven types of mutation rates (point mutation rate, duplications, deletions, inversions, translocations, breakpoints and type transitions, see chapter III). They were all set to 0.01 (per attribute per replication for the point mutation rate, per genetic unit per replication for rearrangement and transition rates, and per attribute per breakpoint for the breakpoint rate)[1];

(2) Environment. The environment A described above was used in all simulations;

(3) Protein production costs. In Evo^{2}Sim, it is possible to set independent energy costs on each of the main activities of a cell (protein production, protein degradation, enzymatic reactions, pumps). For all the simulations, a positive cost were applied to enzymatic reactions and pumps (a cost of 1 Z of energy per Z of transformed, or pumped, metabolite). However, two costs were tested for the protein production: either an elevated cost (1000 Z per Z of produced proteins), or no cost at all (0 Z). The protein degradation cost was set to 0 for all the simulations;

(4) Initial genomes. For all the simulations, we initialized digital organisms with the same handcrafted genome described above;

(5) Simulation time. All the simulations have been run for 500,000 time-steps;

We also tested a second environment (**env. B**), providing external resources ranging from #20 to #30. Environmental parameters were the same than for environment A, except that multiple resources could be provided at the same time. To compensate for the higher quantity of resource provided in the environment, the degradation rate D_g was higher ($D_g = 0.01$). In this environment, initial digital organisms own a handcrafted containing only one operon (**Op. 1**, see Fig. V.1), in order to evaluate their capacity to innovate by creating new metabolic functions from their initial operon.

[1] We also run simulations with low mutation rates (all mutation rates being set at 0.001). However, the very low number of fixed mutations in evolved populations prevented any relevant analysis. We discussed this point in the discussion.

V.3 Results

V.3.1 Digital populations evolving under positive mutation rates are more robust to extinctions

First, we evaluated the extinction time of populations evolving under positive mutation rates, compared to the test case with null mutation rates presented above. As shown on Figure V.4, more populations were able to reach the 500,000 time-steps with mutations enabled than without, both without and with protein production energy costs.

Importantly, for technical reasons, we were not able to compute some simulations without protein production costs to the end. The main reason is the evolution of large metabolic networks in these simulations, with unseen dynamics in previous works with EVO^{2}SIM. In some simulations, the time needed to finish the simulations (several months) would not have allowed us to conclude this chapter in reasonable delays. Hence, only 4 repetitions out of 10 in environment A without protein production costs were completed. They all reached 500,000 time-steps. Comparing these 4 simulations with the test case with a Wilcoxon-Mann-Whitney mean comparison test gives a p-value of 0.041, which is slightly significant.

For populations evolving with protein production costs, the Wilcoxon-Mann-Whitney mean comparison test is not significant (p-value of 0.623), even if 2 repetitions out of 10 reached 500,000 time-steps (other simulations went extinct).

Hence, evolution seems to allow digital organisms to fix mutation events that reduce the risk of extinction, thus making digital organisms more robust to famine episodes. In the next section, we will describe in more details the modifications undergone by the digital organisms, depending on the protein production energy costs. To this aim, we will focus on the simulations that reached the 500,000 time-steps.

V.3.2 Digital populations without protein production cost lost regulation

When mutation rates are enabled, for populations evolving with protein production costs, 2 repetitions reached 500,000 time-steps. For populations evolving without protein production costs, the 4 terminated repetitions reached 500,000 time-steps. The next results are based on these 6 simulations.

We evaluated the capacity of digital populations to keep their genetic regulation network through evolution. To this aim, we evaluated the last best individual (*i.e.*, the individual having the best score, see chapter III) of all simulations that reached the 500,000 time-steps, to see whether the genetic regulation network was lost or not. If the last best

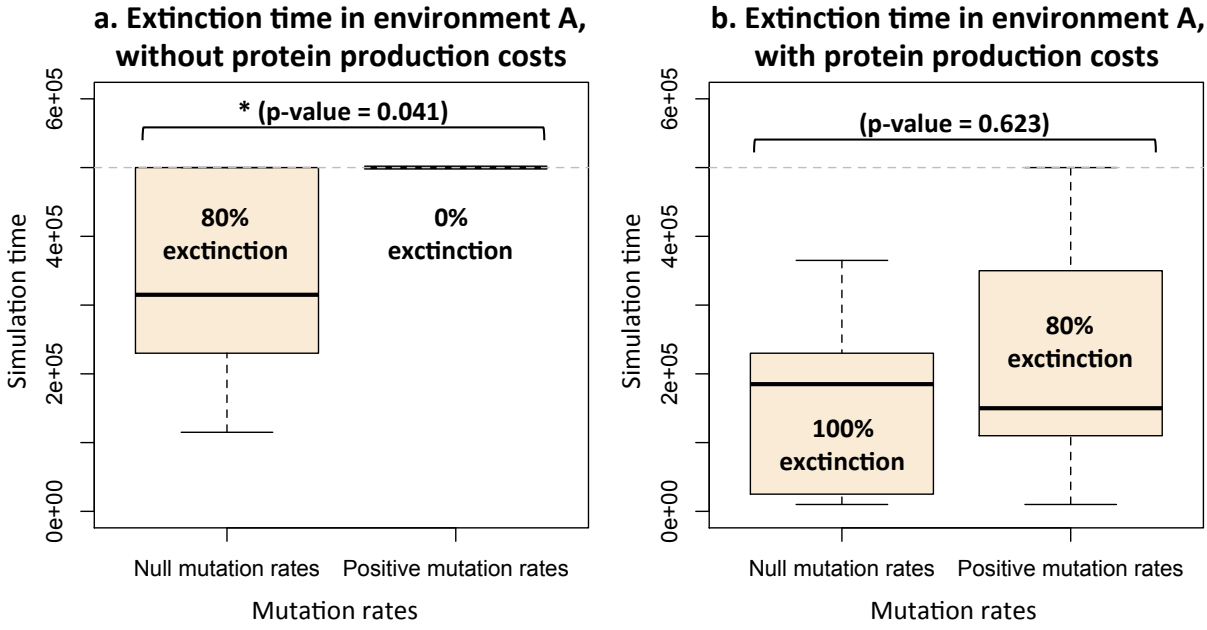


Figure V.4 – Extinction time in environment A. The dynamics of handcrafted digital organisms were tested in the environment A, with either null or positive mutation rates, and in both situations where protein production costs were imposed or not. **a.** Extinction time without protein production costs. 80% of the simulations went extinct with null mutation rates, while 0% went extinct with positive mutation rates. A Wilcoxon-Mann-Whitney mean comparison test gives a p-value of 0.041, which is slightly significant. **b.** Extinction time with protein production costs. 100% of the simulations went extinct with null mutation rates, and 80% went extinct with positive mutation rates. The Wilcoxon-Mann-Whitney mean comparison test is not significant.

individual possess a regulation network at the end of the simulation (even if it is different from the initial one), the organism was considered to have kept regulation. As shown in Table V.2, digital organisms evolving in environment A without protein production cost (and with positive mutation rates) lost their regulation network in 7 repetitions out of 10 (Table V.2). The 3 repetitions that kept regulation have non-functional networks. Thus, all the simulations evolving without protein production costs lost efficient regulation. On the contrary, all the populations evolving with protein production costs kept genetic regulation.

Thus, all the digital populations evolved towards a regulation-free metabolic network without internal energy trade-offs, their proteins being constitutively expressed. However, when significant internal trade-offs are introduced, by setting an energy cost to proteins production, all the simulations kept genetic regulation. This result is in agreement with Weiße et al. (2015).

Table V.2 – Proportion of simulations that kept a regulation network. At the end of each simulation, if the last best individual had no regulation at all, the simulation was considered to have lost regulation.

Scenario	Null mutation rates	Positive mutation rates
Env. A, no protein production cost	100%	30% (0% functional)
Env. A, protein production cost	100%	100%

V.3.3 Protein production costs constrain the evolution of the genome structure

The results above suggest that the existence of internal cellular trade-offs (here an energy cost to the production of proteins) is a condition to at least keep, and possibly evolve genetic regulation networks. But what are the exact differences between digital organisms evolving with, or without protein production costs? One of the main advantage of *in silico* experimental evolution is the possibility to get insights in the details of the structure of each organism. Using the same 6 simulations that reached 500,000 time-steps (4 without protein production costs, 2 with), we evaluated the structure of the last best individual of each simulation. We studied the structure of the genome, the regulation network, the metabolic network, as well as the metabolic content of the cytoplasm.

As shown in Table V.3, the genome structure evolved in very different directions depending on the presence or not of protein production costs. Indeed, populations evolving without production costs own bigger genomes, with many functional regions of small size, and a large proportion of non-coding DNA (except for repetition 8). However, populations evolving with production costs all evolved much smaller genomes, with a single functional region occupying almost 100% of the genome. The latter population thus own a "virus-like" genome, with a single operon coding for all cellular functions. This situation is examplified in Figure V.5, representing the genomes of the last best individuals of repetition 6 without protein production costs and of the repetition 2 with protein production costs. The single operon of the last best genome of repetition 2 is clearly visible (Fig. V.5b), while 9 smaller operons are visible all along the last best genome of repetition 6 (Fig. V.5a). Moreover, the latter genome does not contain any binding site (green triangles) and thus no regulation at all.

V.3.4 For digital populations evolving with protein production costs, reducing genome complexity enhances metabolic complexity

The analysis of the genome structure of populations evolving with protein production costs (Table V.3 and Fig. V.5b) indicates that the initial structure of the genetic regulation network has been modified in the course of evolution. Indeed, initial digital organisms owned two operons (as described above), while Table V.3 indicates that evolved genomes

Table V.3 – Genome structure of the last best individuals in environment A. For each last best individual, we extracted the genome size, the proportion of coding sequences, the number of functional regions, and the mean size of functional regions. 4 genomes are evaluated without protein production costs, 2 with protein production costs.

Repetition	Genome size	Proportion of coding sequences	Nb. functional regions	Functional regions mean size
a. Without protein production costs				
4	439	54.9%	58	4.16
5	282	66.7%	37	5.08
6	60	76.7%	9	5.11
8	30	100%	1	30
b. With protein production costs				
2	32	90.6%	1	29
3	68	100%	1	68

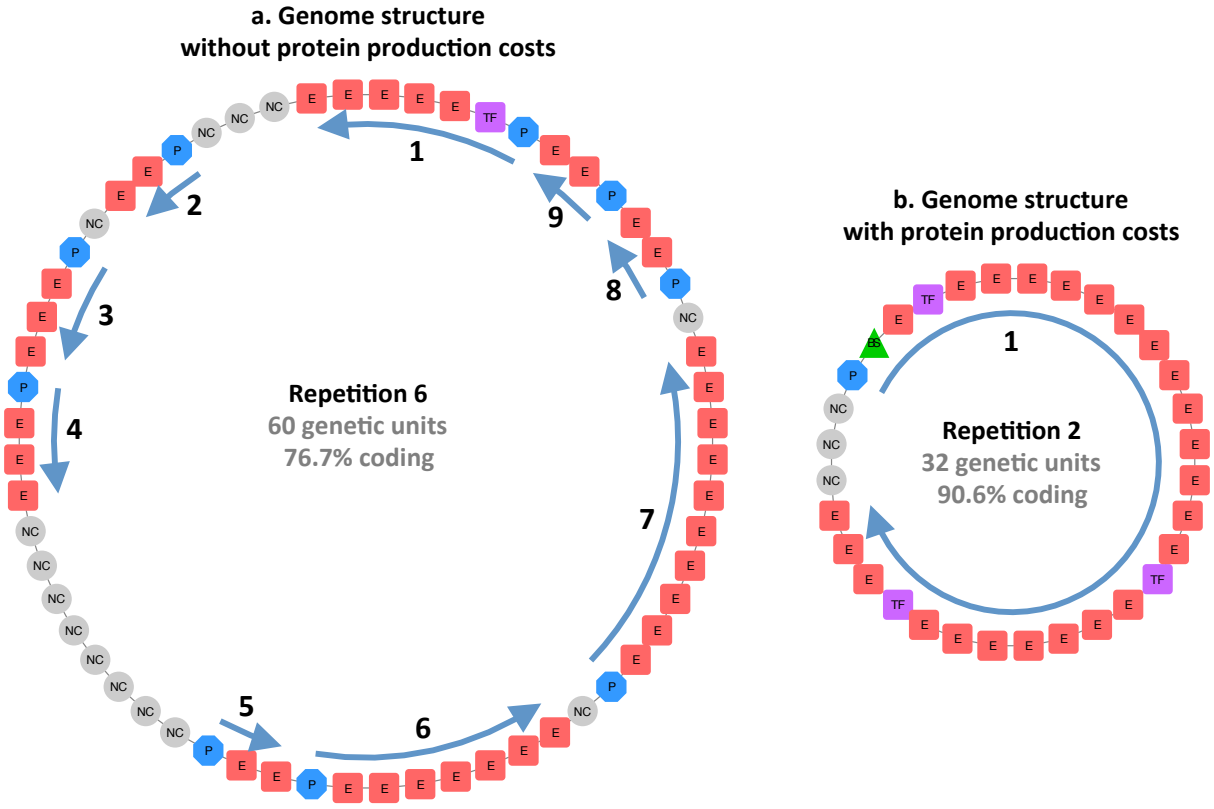


Figure V.5 – Two examples of evolved genome structures depending on protein production costs. The genomes of the last best individuals of two examples are represented. Each functional region is indicated by a blue arrow. Grey circles: non-coding units (NC). Blue octagons: promoters (P). Green triangles: binding sites (BS). Purple squares: transcription factor coding units (TF). Red squares: enzyme coding units (E). **a.** Last best individual's genome of the repetition 6 without protein production costs. ∼24% of the genome is non-coding. 9 functional regions are visible, none of them having binding sites. **b.** Last best individual's genome of the repetition 2 with protein production costs. ∼91% of the genome is coding for a single operon owning an operator site.

own a single operon. In order to get more insights in this phenomenon, we recovered the lineage of the last best individual of the 2 repetitions that reached 500,000 time-steps and evaluated the main indicators of the evolution of the genome, the regulation

network and the metabolic network. As show on Figure V.6 for repetition 2, the loss of a functional region leading to a "virus-like" genome (Fig. V.6a red line) seems to allow for the evolution of more complex biochemical networks. Indeed, just after this important modification of the genome structure, the number of metabolic nodes and edges increased significantly all along evolution, as well as the functional genome size and the number of edges in the regulation network. Hence, the regulation network became smaller, but more connected, while the metabolic network globally grew. The situation is exactly the same for repetition 3 (data not shown).

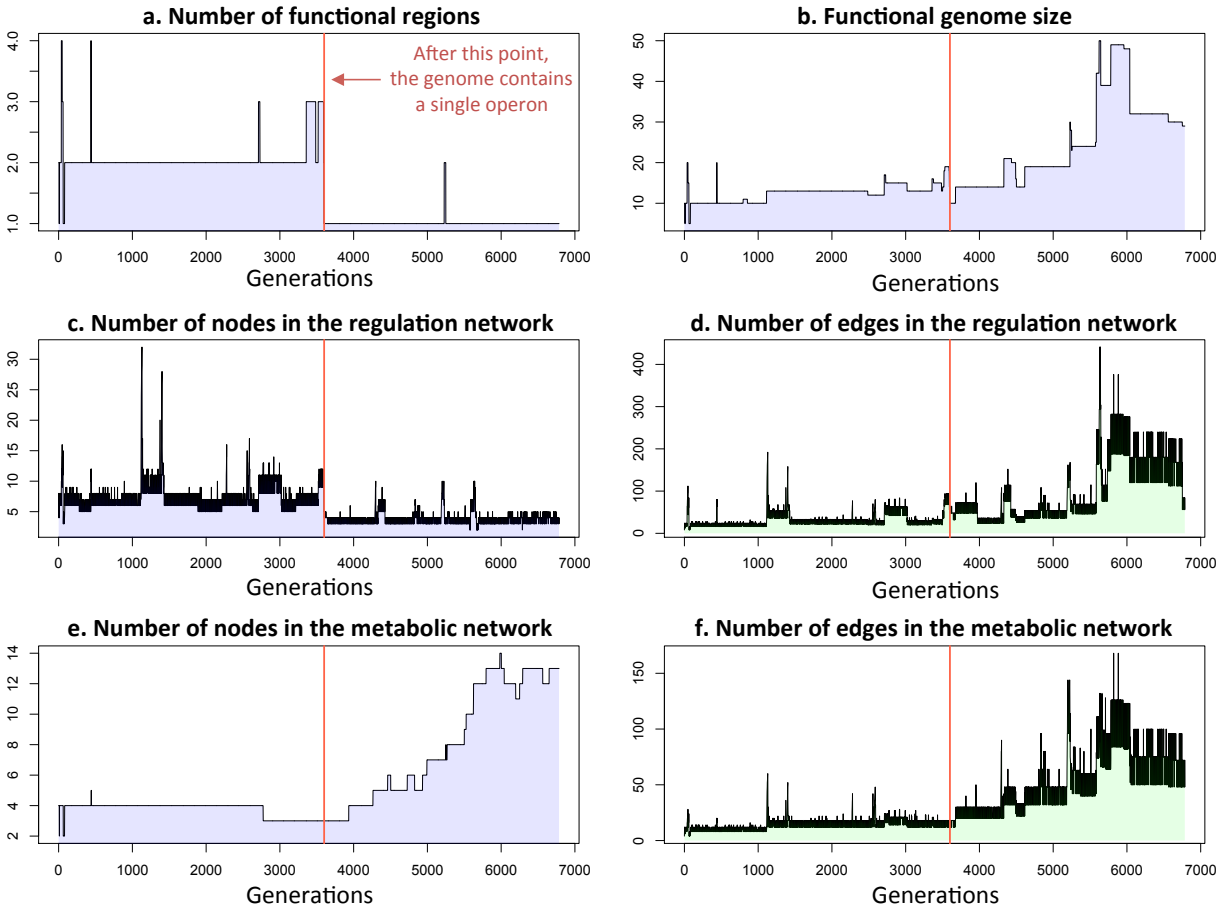


Figure V.6 – Evolution of genome and network structures in the lineage of an evolved organism in environment A with protein production costs. The lineage of the last best individual of the repetition 2 is recovered. **a.** Evolution of the number of functional regions in the genome. Red line: at this point, the lineage undergone a genomic deletion and kept only one operon. **b.** Evolution of the functional genome size (the genome size minus the non-coding DNA). **c.** Evolution of the number of nodes in the genetic regulation network. **d.** Evolution of the number of edges in the genetic regulation network. **e.** Evolution of the number of nodes in the metabolic network. **f.** Evolution of the number of edges in the metabolic network.

Figure V.7 shows the regulation network and the metabolic network corresponding to the last best individual of repetition 2. The structure of the genetic regulation network (Fig. V.7a) shows that a single transcription factor (purple rectangle at the center of the network) inhibits all the enzyme coding units. This transcription factor is repressed by co-enzyme #20. The corresponding metabolic network (Fig. V.7b) is much more complex

and connected than the initial metabolic network (see methods). 4 essential metabolites are produced (#3, #17, #19 and #23, red rectangles), from 6 different external resources (#17, #19, #20, #22, #23 and #26, blue rectangles). The environment A providing only external resources #20 and #22, other resources are wastes of population metabolic activity. Grey nodes correspond to non functional reactions. The properties of the last best individual of repetition 3 are similar (data not shown).

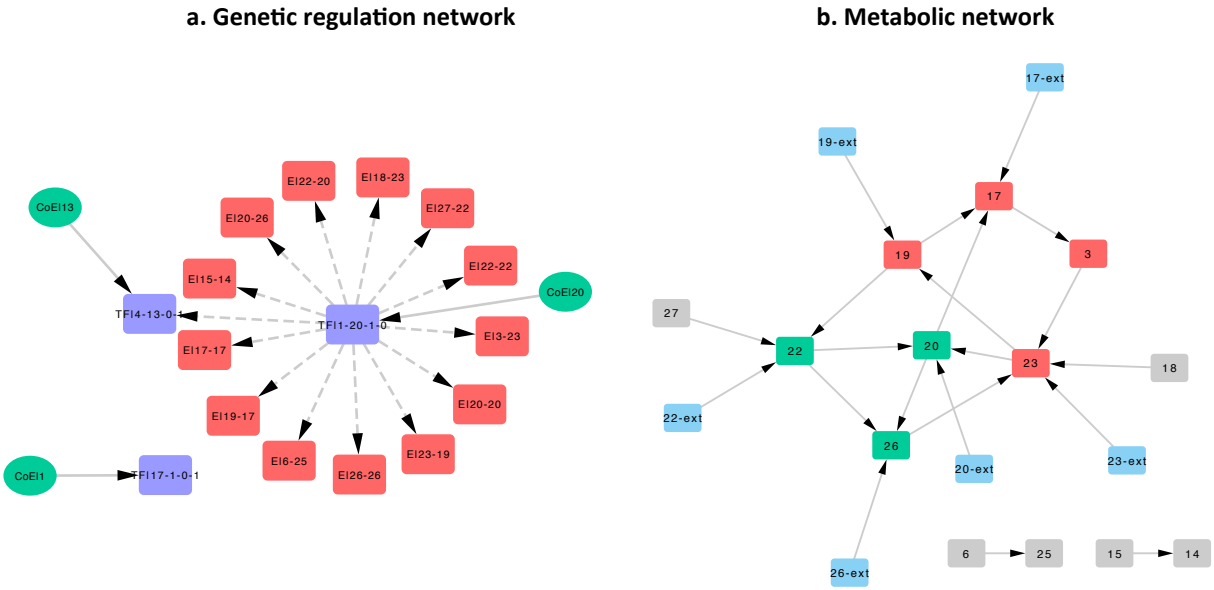


Figure V.7 – Genetic regulation network and metabolic network of an evolved organism in environment A with protein production costs. The networks of the last best individual of the repetition 2 are represented. **a.** Genetic regulation network. Green ellipses: co-enzymes (CoE|<metabolite tag>). Purple rectangles: transcription factors (TF|<BS tag; CoE tag; free activity; bound activity>, see chapter III). Red rectangles: enzymes (E|<substrate; product>, see chapter III). Solid arrows: positive regulation. Dashed arrows: negative regulation. **b.** Metabolic network. Blue rectangles: external metabolite. Green rectangles: non-essential metabolites (see chapter III). Red rectangles: essential metabolites (see chapter III). Arrows: enzymatic reaction.

Finally, looking at the metabolic content of the cytoplasm of the last best individual and at its local environment (Fig. V.8), we see that many metabolites are produced and released by digital organisms (at death since no outflowing pumps are coded in the genome). These cellular products are then available for other organisms, thus leading to a higher complexity of the metabolic network.

V.3.5 Digital populations evolving in diversified environments with protein production energy costs also evolved a single operon

As presented above, we also evaluated our model with digital populations evolving in a second environment: environment B (see Methods). In this environment, multiple external resources are provided, ranging from metabolite #20 to #30. Thus, this environment

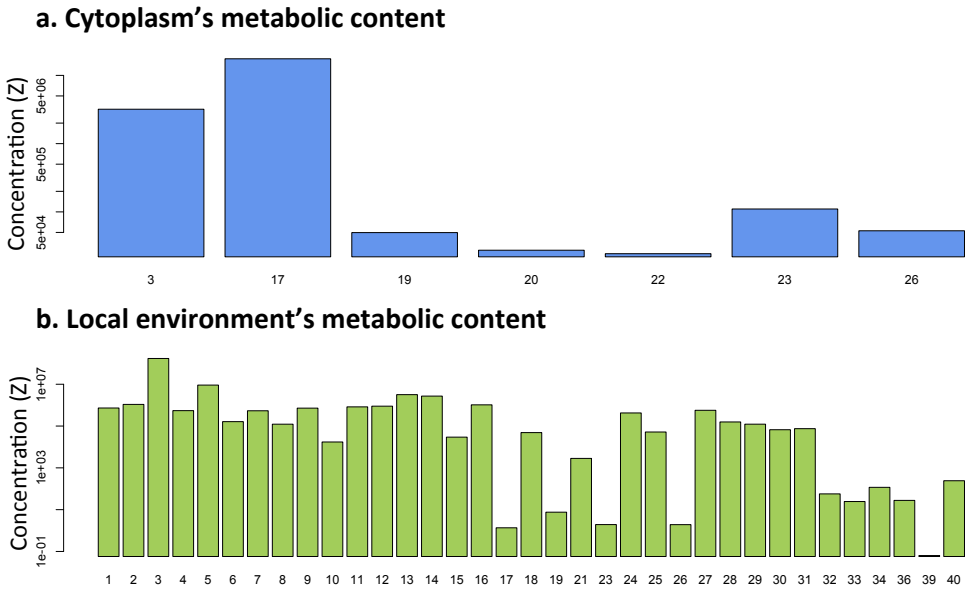


Figure V.8 – Cytoplasmic metabolic content of an evolved organism in environment A with protein production costs. a. List of metabolites present in the cytoplasm of the last best individual of repetition 2. **b.** List of metabolites present in the local environment of this individual.

is richer (many metabolites are provided at the same time) and more diversified than environment A. Initial digital organisms own a handcrafted genome only containing one operon: the **Op. 1** described above, that allows an organism to uptake metabolite #20 and to produce essential metabolite #5.

On the 10 populations that evolved in this environment, only two reached the 500,000 time-steps (repetitions 1 and 2)[1]. In the same manner than for the previous section, we evaluated in detail the structure of the last best individual of each repetition. As shown in Table V.4, final genomes present the same virus-like structure as for environment A with protein production costs, with only a single operon occupying the whole genome. However, in environment A, digital organisms had lost one operon (**Op. 2**), keeping and complexifying the remaining one (**Op. 1**). Coherently, in environment B, digital organisms kept their single operon.

However, as exemplified on Figure V.9 for the last best individual of the repetition 2, digital organisms evolved the ability to exploit the various resources provided in environment B, without losing the self-repressed regulation controlled by the co-enzyme metabolite #20. Contrary to environment A, there is less chance for prolonged famine in environment B, since multiple resources are possibly provided at the same time at each environmental location. Thus, there is no particular reason to use the metabolite #20

[1]The same simulations have been run with null mutation rates: 5 repetitions out of 10 reached 500,000 time-steps. The mean extinction time is 390,500 time-steps with no mutation, and 294,500 time-steps with positive mutation rates (a Wilcoxon-Mann-Whitney test gives a p-value of 0.152). Thus in the particular case of the environment B, it seems that digital organisms are more robust with null mutation rates, when they keep their initial structure. The reason is the abundance of external metabolite #20, and thus the absence of prolonged famine. Evolving organisms are more exposed to whole population extinctions, meaning that evolution does not lead to more robust organisms in this case.

Table V.4 – Genome structure of the last best individuals in environment B. For each last best individual, we extracted the genome size, the proportion of coding sequences, the number of functional regions, and the mean size of functional regions. 2 genomes are evaluated.

Repetition	Genome size	Proportion of coding sequences	Nb. functional regions	Functional regions mean size
1	11	100%	1	11
2	40	100%	1	40

than any other in the environment, except for contingent historical reasons, independently from the environmental variability.

V.4 Discussion

Cellular metabolism is often considered to be finely tuned by the genetic regulation network by precisely adjusting enzymatic concentrations in response to environmental resource fluctuations. However, as theoretically demonstrated by Weiße et al. (2015), it seems that the role of genetic regulation is not to adapt the metabolic activity to environmental changes, but to balance internal energy and resource trade-offs.

In previous experiments with Evo^{2}Sim (Rocabert et al., 2017), where the cell model was not energy-limited, no functional regulation network evolved (see chapter IV). In the attempt to study the maintenance and the evolution of genetic regulation, we parameterized Evo^{2}Sim with realistic parameters values, and introduced strong energy trade-offs, by imposing energy costs to the main cellular functions (protein production, anabolism and active pumps). We then let digital organisms with handcrafted initial regulation and metabolic networks evolve in various conditions.

Although many simulations led to population extinctions, our preliminary results suggest that genetic regulation indeed evolved not to cope with environmental changes, but to balance internal energy trade-offs, and avoid premature cell death due to the depletion of energy carrier molecules (as suggested by Weiße et al. 2015). First, we showed that populations evolving without protein production costs lost genetic regulation, with no negative effect on the evolution of their metabolic network. On the contrary, populations that survive under strong protein production costs all kept regulation (Table V.2). Moreover, our results showed that the genome structure of digital organisms evolving in Evo^{2}Sim is strongly impacted by the existence of protein production costs: this includes the non-coding elements, while there is no cost to DNA replication in this model. Indeed, while digital populations without protein production costs evolved larger genomes, with a significant amount of non-coding DNA and many functional regions each coding for a few proteins, populations with protein production costs evolved compacted genomes, with no non-coding DNA and having a single functional region coding for a large operon. This self-repressed operon codes all the functions of the cell (regulation and metabolism), its expression being activated by single co-enzyme.

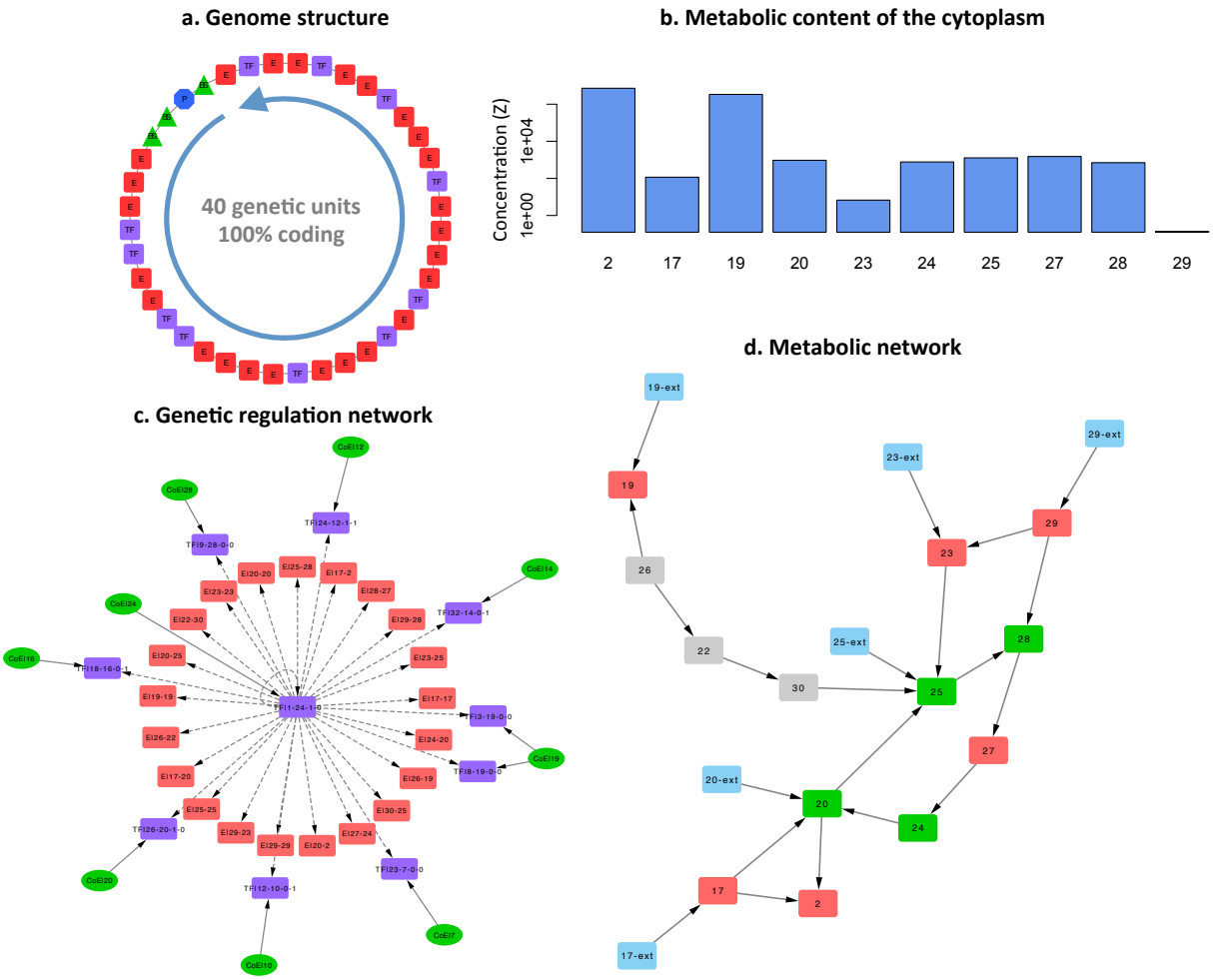


Figure V.9 – Structure of an evolved digital organism in environment B, with protein production costs. The structure of the last best individual of the repetition 2 is represented. **a.** Genome structure. The single functional region is indicated by a blue arrow. Blue octagon: promoter (P). Green triangles: binding sites (BS). Purple squares: transcription factor coding units (TF). Red squares: enzyme coding units (E). **b.** Cytoplasmic metabolic content. Concentrations are in arbitrary concentration units (ACU, see chapter III). **c.** Genetic regulation network. Green ellipses: co-enzymes (CoE|<metabolite tag>). Purple rectangles: transcription factors (TF|<BS tag; CoE tag; free activity; bound activity>, see chapter III). Red rectangles: enzymes (E|<substrate; product>, see chapter III). Solid arrows: positive regulation. Dashed arrows: negative regulation. **d.** Metabolic network. Blue rectangles: external metabolite. Green rectangles: non-essential metabolites (see chapter III). Red rectangles: essential metabolites (see chapter III). Arrows: enzymatic reaction.

Digital populations evolving with protein production costs in diversified environments, providing more resources, also evolved this "virus-like" structure, with a single self-repressed operon, also activated by a single co-enzyme. This co-enzyme has no clear relation with environmental dynamics, but is the result of historical constraints and purely internal trade-offs, unlinked to environmental variability.

As a whole, our results suggest that in EVO^{2}SIM, digital populations evolving with protein production energy costs undergo important constraints on their genome structure, while

there is no cost to DNA replication. Indeed, such a "virus-like" structure could be a good way to limit energy consumption due to protein production, by expressing all cellular functions at a single time. This protein expression pattern could limit the number of energy drops. However, as shown on V.9, this specific genome structure does not seem to limit the evolution of complexity, since digital organisms evolved in the diversified environment still exploit many resources.

As a next step, these preliminary results could be improved, at least in two directions:

(i) First, a parametric exploration should be performed on energy cost parameters and mutation rates. Indeed, in these simulations, we set the protein production costs such that digital organisms losing regulation under elevated protein production costs systematically die. Moreover, our energy model could be more realistic. In the current version of Evo^{2}Sim, energy carrier molecules are not included in ordinary differential equations, and have no effect on reaction speed. It could be interesting to couple energy more intimately to the artificial chemistry of Evo^{2}Sim, as it may provide more realistic cell's dynamics. In this work, we also run simulations with low mutation rates, but the very low number of mutation events fixed in those simulations prevented any relevant analysis. However, previous modeling and mathematical works suggested that mutation rates have a strong effect on the genome structure: populations evolving with high mutation rates own small genomes, with few non-coding DNA and large operons. On the contrary, populations evolving with low mutation rates own large genomes, with many non-coding DNA and many small operons (Knibbe et al., 2007a; Beslon et al., 2010b; Fischer et al., 2014). The relation between genome size and mutation rates follows a power law, linked to a long-term adjustment between the robustness and the evolvability of digital organisms. Thus, it could be interesting to study this effect in Evo^{2}Sim, when high energy costs are applied. For example, by using ævol software, Knibbe et al. (2007b) showed that a coupling exists between the amount of non-coding DNA and the deleteriousness of gene mutations. The more deleterious the gene mutations, the shorter the intergenic sequences (and thus the genome size). In this work, gene mutations are probably much more deleterious when protein production costs are applied, thus leading to smaller genomes with few non-coding DNA;

(ii) Second, a clear step is waiting to be crossed between these results and those from Rocabert et al. (2017) (see chapter IV). Indeed, the long-term evolution experiment (LTEE, Elena and Lenski 2003) provided insights in the evolution of stable cross-feeding (Rozen et al., 2005), but also in the evolution of genetic regulation and metabolic networks. In the LTEE, mutations that led to bacterial diversification in *E. coli* are often linked to the regulation of metabolic pathways (see *e.g.*, Großkopf et al. 2016). Thus, replaying the experimental protocol from Rocabert et al. (2017) with more realistic parameters and energy costs could provide important insights in the understanding of bacterial diversification.

Nonetheless, our preliminary results are in agreement with the work of Weiße et al. (2015). Moreover, we were able to identify evolutive relationship between selective pressures at

the level of the metabolism, and the structure of the genome. Even if the rationals of this relationship are still to be unraveled, this is a beautiful demonstration of the ability of multi-scale models to generate counter-intuitive outcomes, when multiple biological structures interact and evolve together.

Conclusion and outlook

The beauty of Darwinian evolution is relying on the remarkable simplicity of its core principles—variation and selection. With these two basic ingredients and the help of billions of years of evolution, extraordinarily complex and diversified living systems populated the earth. This apparent contradiction between the simplicity of Darwinian principles and the complexity of observed organisms is due to the powerful emergent properties of evolution, bearing no comparison with other scientific theories.

As discussed in introduction, long-term evolution, indirect selection and multi-level selection are partly responsible, as far as we know, for these counter-intuitive outcomes of evolution on earth. Indeed, many complex properties of living organisms emerged from their capacity to accumulate information from past and variable environments (Hogeweg, 2011), through mutations and selection, leading to complex properties such as evolvability or robustness. The belief that long-term evolution leads to evolution of evolution is now largely spreading in theoretical evolutionary biology, as demonstrated by the attempt to extend the modern synthesis (Watson and Szathmáry, 2016), and the idea that evolution "*learnt how to learn*" (Mattick, 2009).

Scientists – and thus scientific theories – are part of a society, with its culture, its economy and its beliefs. From the animal machine theory of R. Descartes to the digesting duck of J. de Vaucanson, influenced by scientific and technical progress due to Newtonian physics, many examples show how biology has been influenced by the historical context. This is not a coincidence if the dogmatic view that living organisms own a genetic program encoded in the DNA molecule emerged as computer science was largely spreading in industrialized countries. In consequence, we cannot ignore that current advances in theoretical evolutionary biology have obvious links with the current craze for machine learning theories. Is there a nascent dogmatism? Nonetheless, as stated by J. Monod, any scientific theory carries its part of unavoidable dogmatism. If we assume that scientific theories are tools to better understand the world, being aware of the limits of a theory never prevented its explanation power and its utility. After all, Newtonian theory is still used in many industrial applications.

In this thesis, we used a complementary modeling approach in the hope of deciphering the emerging properties of evolution leading to evolution of evolution. First, we studied the evolution of phenotypic noise with a classical, mathematical approach. Such a model, with few parameters and the possibility to perform analytical resolutions, allowed us to

rigorously and completely study some of its properties. The domain and the dynamics are well-defined: the outcomes of the model could be sometimes counter-intuitive (as it was the case with σFGM), but they were always completely described by equations. For example, the conditions in which phenotypic noise would increase in directional selection, when the population is far from the fitness optimum, and then be minimized in stabilizing selection, when the population reaches the fitness optimum, are well-defined: the fitness landscape should be convex, at least locally, and noise correlations between characters should be evolvable. However, the simplicity of this model comes with an inevitable lack of information on what would be the mechanisms at work in real organisms. Regarding evolution, and even more evolution of evolution, this calls for more complex multi-scale models, in order to better understand the evolution of phenotypic noise when the genome structure or the genetic regulation network evolve for example.

In the second part of this manuscript, we used a multi-scale model of *in silico* evolution to decipher some aspects of indirect and multi-level selection on bacterial-like digital organisms. Contrary to simpler mathematical models, we have seen that this kind of models almost always provide results that are complex and difficult to apprehend. Moreover, the approach is radically different from the previous one: the number of parameters, the amount of data produced and the complexity of the behavior impose an experimental approach, like for real systems (Peck, 2004). In evolutionary biology, we have the chance to be able to compare numerical simulations with real evolution experiments, thanks to the *in silico* experimental evolution approach (Hindré et al., 2012), which allowed us to compare our results with the long-term evolution experiment with *Escherichia coli* (Rozen et al., 2005; Rocabert et al., 2017). However, the observations made in complex simulations often need to be evaluated in a more robust and comprehensible way, and to be generalized. As such, complex models in turn call back for simpler models that can be perfectly defined ... In other words, analytical models. In the case of the evolution of stable cross-feeding in bacterial populations, this work has partially be done (see *e.g.* Rozen et al. 2009, or Ribeck and Lenski 2015). But other examples demonstrated the utility to transfer hypotheses raised by *in silico* experimental evolution models into simpler mathematical models, as it is the case with ævol software, for at least two results: the link between the genome size and the mutation rates (Knibbe et al., 2007a; Fischer et al., 2014), and the evolution of cooperation (Frénoy et al., 2013, 2017).

Somehow, complex and multi-scale numerical models provide an intuition of a mechanism, an hypothesis or a theory, and mathematical models provide a robust and well-defined solution to this intuition. Thus, studying emergent properties of evolution consists in alternating between both approaches. But in all cases, the results obtained with these models call for experimental validation. Regarding the evolution of phenotypic noise, it would be very interesting to initiate, or exploit, experimental evolution protocols where micro-organisms are placed in directional selection, and where single-cell data are acquired at the level of the phenotype (for example, by exploiting barcoding technologies, see Levy et al. 2015; Venkataram et al. 2016, or by evolving micro-organisms under artificial directional selection, see Ito et al. 2009). Regarding the results obtained with EVO^{2}SIM, a possibility would be to replay the experimental protocol of Rocabert et al. (2017) with an energy-limited cell model, and to compare the results with the precise mutational data

from the long-term evolution experiment, including mutations in the regulation network (see e.g. Großkopf et al. 2016).

As a whole, we think that the results obtained along this thesis have a great potential for further motivating research work and experiments. The paths to take have been cleared; we hope to follow them and pursue this exciting scientific adventure.

Bibliography

Acar, M., Mettetal, J. T., and van Oudenaarden, A. (2008). Stochastic switching as a survival strategy in fluctuating environments. *Nature Genetics*, 40(4):471–475.

Adami, C. (2006). Digital genetics: unravelling the genetic basis of evolution. *Nature reviews Genetics*, 7(2):109–18.

Alberch, P. (1991). From genes to phenotype: dynamical systems and evolvability. *Genetica*, 84(1):5–11.

Alberts, B., Bray, D., Hopkin, K., Johnson, A., Lewis, J., Raff, M., Roberts, K., and Walter, P. (2013). *Essential cell biology*. Garland Science, New York City, New York (United States).

Anderson, T. W., Olkin, I., and Underhill, L. G. (1987). Generation of random orthogonal matrices. *SIAM Journal on Scientific and Statistical Computing*, 8(4):625–629.

Andrews, P. S., Stepney, S., Hoverd, T., Polack, F. A. C., Sampson, A. T., and Timmis, J. (2011). CoSMoS process, models, and metamodels. *CoSMoS 2011: Proceedings of the 2011 Workshop on Complex Systems Modelling and Simulation*, pages 1–13.

Avery, O. T., MacLeod, C. M., and McCarty, M. (1944). Studies on the chemical nature of the substance inducing transformation of Pneumococcal types. *The Journal of Experimental Medicine*, 79(2):137–158.

Avery, S. V. (2006). Microbial cell individuality and the underlying sources of heterogeneity. *.Nature reviews Microbiology*, 4(8):577–87.

Bahar, R., Hartmann, C. H., Rodriguez, K. a., Denny, A. D., Busuttil, R. a., Dollé, M. E. T., Calder, R. B., Chisholm, G. B., Pollock, B. H., Klein, C. a., and Vijg, J. (2006). Increased cell-to-cell variation in gene expression in ageing mouse heart. *Nature*, 441(7096):1011–1014.

Balaban, N. Q., Merrin, J., Chait, R., Kowalik, L., and Leibler, S. (2004). Bacterial Persistence as a Phenotypic Switch. *Science*, 305(5690):1622–1625.

Banzhaf, W., Baumgaertner, B., Beslon, G., Doursat, R., Foster, J. A., McMullin, B., de Melo, V. V., Miconi, T., Spector, L., Stepney, S., and White, R. (2016). Defining and simulating open-ended novelty: requirements, guidelines, and challenges. *Theory in Biosciences*, 135(3):131–161.

Banzhaf, W. and Yamamoto, L. (2015). *Artificial chemistries*. MIT Press, Cambridge, Massachusetts (United States).

Bar-even, A., Noor, E., Savir, Y., Liebermeister, W., Davidi, D., Tawfik, D. S., Milo, R., Taw, D. S., and Milo, R. (2011). The Moderately Efficient Enzyme : Evolutionary and Physicochemical. *Biochemistry*, 50(21):4402–4410.

Batut, B., Parsons, D. P., Fischer, S., Beslon, G., and Knibbe, C. (2013). In silico experimental evolution: a tool to test evolutionary scenarios. *BMC Bioinformatics*, 14 Suppl 1(Suppl 15):S11.

Beaumont, H. J., Gallie, J., Kost, C., Ferguson, G. C., and Rainey, P. B. (2009). Experimental evolution of bet hedging. *Nature*, 462(7269):90–93.

Bedau, M. A. and Packard, N. H. (2003). Evolution of evolvability via adaptation of mutation rates. *BioSystems*, 69(2-3):143–162.

Beslon, G. (2008). Apprivoiser la vie: Modélisation individu-centrée de systèmes biologiques complexes. *Habilitation à Diriger des Recherches*, page 138.

Beslon, G., Parsons, D. P., Pea, J. M., Rigotti, C., and Sanchez-Dehesa, Y. (2010a). From digital genetics to knowledge discovery: Perspectives in genetic network understanding. *Intelligent Data Analysis*, 14(2):173–191.

Beslon, G., Parsons, D. P., Sanchez-Dehesa, Y., Peña, J. M., and Knibbe, C. (2010b). Scaling laws in bacterial genomes: A side-effect of selection of mutational robustness? *BioSystems*, 102(1):32–40.

Bódi, Z., Farkas, Z., Nevozhay, D., Kalapis, D., Lázár, V., Csörgő, B., Nyerges, Á., Szamecz, B., Fekete, G., Papp, B., Araújo, H., Oliveira, J. L., Moura, G., Santos, M. A., Székely, T., Balázsi, G., and Pál, C. (2017). Phenotypic heterogeneity promotes adaptive evolution. *PLOS Biology*, 15(5):e2000644.

Boukhibar, L. M. and Barkoulas, M. (2016). The developmental genetics of biological robustness. *Annals of Botany*, 117(5):699–707.

Boyle, R. A., Williams, H. T. P., and Lenton, T. M. (2012). Natural selection for costly nutrient recycling in simulated microbial metacommunities. *Journal of Theoretical Biology*, 312:1–12.

Chalancon, G., Ravarani, C. N. J., Balaji, S., Martinez-Arias, A., Aravind, L., Jothi, R., and Babu, M. M. (2012). Interplay between gene expression noise and regulatory network architecture. *Trends in Genetics*, 28(5):221–232.

Chalmers, A. F. (1990). *Qu'est-ce que la science ?* Biblio Essais. Le Livre de Poche, Paris (France).

Charlebois, D. A. (2015). Effect and evolution of gene expression noise on the fitness landscape. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 92(2):1–7.

Charlebois, D. A., Abdennur, N., and Kaern, M. (2011). Gene expression noise facilitates adaptation and drug resistance independently of mutation. *Physical Review Letters*, 107(21):218101.

Charlebois, D. A., Balázsi, G., and Kærn, M. (2014). Coherent feedforward transcriptional regulatory motifs enhance drug resistance. *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 89(5):052708.

Chevin, L.-M. (2011). On measuring selection in experimental evolution. *Biology Letters*, 7(2):210–3.

Chow, S. S. (2004). Adaptive Radiation from Resource Competition in Digital Organisms. *Science*, 305(5680):84–86.

Cohan, F. M. (2002). What are bacterial species? *Annual Review of Microbiology*, 56:457–487.

Cohen, A. A., Geva-Zatorsky, N., Eden, E., Frenkel-Morgenstern, M., Issaeva, I., Sigal, A., Milo, R., Cohen-Saidon, C., Liron, Y., Kam, Z., Cohen, L., Danon, T., Perzov, N., and Alon, U. (2008). Dynamic Proteomics of Individual Cancer Cells in Response to a Drug. *Science*, 322(5907):1511–1516.

Correia, M. B. and Fonseca, C. M. (2007). How redundancy and neutrality may affect evolution on NK fitness landscapes. *2007 IEEE Congress on Evolutionary Computation, CEC 2007, Singapore (Singapore)*, pages 2842–2849.

Costa, E., Pérez, J., and Kreft, J. U. (2006). Why is metabolic labour divided in nitrification? *Trends in Microbiology*, 14(5):213–219.

Cressler, C. E., Bengtson, S., and Nelson, W. A. (2017). Unexpected nongenetic individual heterogeneity and trait covariance in Daphnia And its consequences for ecological and evolutionary dynamics. *The American Naturalist*, 190(1):E13–E27.

Crick, F. H. (1958). On protein synthesis. *The Symposia of the Society for Experimental Biology*, 12(1):138–163.

Crombach, A. and Hogeweg, P. (2007). Chromosome rearrangements and the evolution of genome structuring and adaptability. *Molecular Biology and Evolution*, 24(5):1130–1139.

Crombach, A. and Hogeweg, P. (2008). Evolution of evolvability in gene regulatory networks. *PLoS Computational Biology*, 4(7):1–13.

Crombach, A. and Hogeweg, P. (2009). Evolution of resource cycling in ecosystems and individuals. *BMC Evolutionary Biology*, 9(1):122.

Cuypers, T. D. and Hogeweg, P. (2012). Virtual genomes in flux: An interplay of neutrality and adaptability explains genome expansion and streamlining. *Genome Biology and Evolution*, 4(3):212–229.

Cuypers, T. D., Rutten, J. P., and Hogeweg, P. (2017). Evolution of evolvability and phenotypic plasticity in virtual cells. *BMC Evolutionary Biology*, 17(1):60.

Dar, R. D., Hosmane, N. N., Arkin, M. R., Siliciano, R. F., and Weinberger, L. S. (2014). Screening for noise in gene expression identifies drug synergies. *Science*, 344(6190):1392–1396.

Darwin, C. (1859). *On the Origin of the Species by Natural Selection*. Murray, London (UK).

Dawkins, R. (1976). *The selfish gene*. Oxford University Press, Oxford (UK).

De Jong, I. G., Haccou, P., and Kuipers, O. P. (2011). Bet hedging or not? A guide to proper classification of microbial survival strategies. *BioEssays*, 33(3):215–223.

Dejonghe, W., Berteloot, E., Goris, J., Boon, N., Crul, K., Maertens, S., Höfte, M., De Vos, P., Verstraete, W., and Top, E. M. (2003). Synergistic degradation of linuron by a bacterial consortium and isolation of a single linuron-degrading Variovorax strain. *Applied and Environmental Microbiology*, 69(3):1532–1541.

Denamur, E. and Matic, I. (2006). Evolution of mutation rates in bacteria. *Molecular Microbiology*, 60(4):820–827.

Dittrich, P., Ziegler, J., and Banzhaf, W. (2001). Artificial Chemistries - A Review. *Artificial Life*, 7(3):225–275.

Doebeli, M. (2002). A model for the evolutionary dynamics of cross-feeding polymorphisms in microorganisms. *Population Ecology*, 44(2):59–70.

Dwight Kuo, P., Banzhaf, W., and Leier, A. (2006). Network topology and the evolution of dynamics in an artificial genetic regulatory network model created by whole genome duplication and divergence. *BioSystems*, 85(3):177–200.

Eldar, A. and Elowitz, M. B. (2010). Functional roles for noise in genetic circuits. *Nature*, 467(7312):167–173.

Elena, S. F. and Lenski, R. E. (2003). Evolution experiments with microorganisms: the dynamics and genetic bases of adaptation. *Nature reviews Genetics*, 4(6):457–69.

Elena, S. F. and Sanjuán, R. (2008). The effect of genetic robustness on evolvability in digital organisms. *BMC Evolutionary Biology*, 8(1):284.

Elowitz, M. B., Levine, A. J., Siggia, E. D., and Swain, P. S. (2002). Stochastic Gene Expression in a Single Cell. *Science*, 297(5584):1183–1186.

Fischer, S., Bernard, S., Beslon, G., and Knibbe, C. (2014). A Model for Genome Size Evolution. *Bulletin of Mathematical Biology*, 76(9):2249–2291.

Fisher, R. A. (1930). *The genetical theory of natural selection: a complete variorum edition*. Oxford University Press, Oxford (UK).

Foster, P. L. (2007). Stress-induced mutagenesis in bacteria. *Critical Reviews in Biochemistry and Molecular Biology*, 42(5):373–97.

Fraser, D. and Kærn, M. (2009). A chance at survival: Gene expression noise and phenotypic diversification strategies. *Molecular Microbiology*, 71(6):1333–1340.

Fraser, H. B., Hirsh, A. E., Giaever, G., Kumm, J., and Eisen, M. B. (2004). Noise minimization in eukaryotic gene expression. *PLoS Biology*, 2(6):834–838.

Frénoy, A., Taddei, F., and Misevic, D. (2013). Genetic Architecture Promotes the Evolution and Maintenance of Cooperation. *PLoS Computational Biology*, 9(11):1–12.

Frénoy, A., Taddei, F., and Misevic, D. (2017). Second-order cooperation: Cooperative offspring as a living public good arising from second-order selection on non-cooperative individuals. *Evolution*, 71(7):1802–1814.

Gandrillon, O., Kolesnik-Antoine, D., Kupiec, J.-J., and Beslon, G. (2012). *Chance at the heart of the cell*. Pergamon, Oxford (UK).

Gascoigne, K. E. and Taylor, S. S. (2008). Cancer Cells Display Profound Intra- and Interline Variation following Prolonged Exposure to Antimitotic Drugs. *Cancer Cell*, 14(2):111–122.

Gerlee, P. and Lundh, T. (2010a). Productivity and diversity in a cross-feeding population of artificial organisms. *Evolution*, 64(9):2716–2730.

Gerlee, P. and Lundh, T. (2010b). Rock-Paper-Scissors Dynamics in a Digital Ecology. *Proceedings of the Artificial life XII Conference, Odense (Denmark)*, pages 285–292.

Gerlee, P. and Lundh, T. (2012). Effect of space in the game "war of attrition". *Physical Review E - Statistical, Nonlinear, and Soft Matter Physics*, 85(4):6–11.

Gilbert, J. A., Steele, J. A., Caporaso, J. G., Steinbrück, L., Reeder, J., Temperton, B., Huse, S., McHardy, A. C., Knight, R., Joint, I., Somerfield, P., Fuhrman, J. A., and Field, D. (2012). Defining seasonal marine microbial community dynamics. *The ISME Journal*, 6(2):298–308.

Gillespie, D. T. (2007). Stochastic simulation of chemical kinetics. *Annual review of Physical Chemistry*, 58:35–55.

Grimm, V. (1999). Ten years of individual-based modelling in ecology: What have we learned and what could we learn in the future? *Ecological Modelling*, 115(2-3):129–148.

Großkopf, T., Consuegra, J., Gaffé, J., Willison, J. C., Lenski, R. E., Soyer, O. S., and Schneider, D. (2016). Metabolic modelling in a dynamic evolutionary framework predicts adaptive diversification of bacteria in a long-term evolution experiment. *BMC Evolutionary Biology*, 16:163.

Grover, J. P. (1988). Dynamics of Competition in a Variable Environment : Experiments with Two Diatom Species. *Ecology*, 69(2):408–417.

Gudelj, I., Kinnersley, M., Rashkov, P., Schmidt, K., and Rosenzweig, F. (2016). Stability of Cross-Feeding Polymorphisms in Microbial Communities. *PLoS Computational Biology*, 12(12):1–17.

Hauert, C. and Doebeli, M. (2004). Spatial structure often inhibits the evolution of cooperation in the snowdrift game. *Nature*, 428(6983):643–646.

Helling, R. B., Vargas, C. N., and Adams, J. (1987). Evolution of Escherichia coli during growth in a constant environment. *Genetics*, 116(3):349–358.

Hindré, T., Knibbe, C., Beslon, G., and Schneider, D. (2012). New insights into bacterial adaptation through in vivo and in silico experimental evolution. *Nature Reviews Microbiology*, 10(5):352–365.

Hogeweg, P. (1978). Simulating the growth of cellular forms. *Simulation*, 31(3):90–96.

Hogeweg, P. (2011). The roots of bioinformatics in theoretical biology. *PLoS Computational Biology*, 7(3):1–5.

Hogeweg, P. and Hesper, B. (1978). Interactive instructions on population interactions.

Hogeweg, P. and Takeuchi, N. (2003). Multilevel selection in models of prebiotic evolution: Compartments and spatial self-organization. *Origins of Life and Evolution of the Biosphere*, 33(4-5):375–403.

Holland, S. L., Reader, T., Dyer, P. S., and Avery, S. V. (2014). Phenotypic heterogeneity is a selected trait in natural yeast populations subject to environmental stress. *Environmental Microbiology*, 16(6):1729–1740.

Huang, S. (2012). Tumor progression: Chance and necessity in Darwinian and Lamarckian somatic (mutationless) evolution. *Progress in Biophysics and Molecular Biology*, 110(1):69–86.

Huxley, J. (1942). *Evolution the modern synthesis*. George Allen and Unwin, Crows Nest (Australia).

Ito, Y., Toyota, H., Kaneko, K., and Yomo, T. (2009). How selection affects phenotypic fluctuation. *Molecular Systems Biology*, 5(264):264.

Jacob, F. (1970). *La logique du vivant*. Gallimard, Paris (France).

Jacob, F. and Monod, J. (1961). Genetic regulatory mechanisms in the synthesis of proteins. *Journal of Molecular Biology*, 3(3):318–356.

Jo, J., Kang, H., Choi, M. Y., and Koh, D.-S. (2005). How noise and coupling induce bursting action potentials in pancreatic β-cells. *Biophysical journal*, 89(3):1534–42.

Johannsen, W. (1911). The genotype conception of heredity. *The American Naturalist*, 45(531):129–159.

Johnson, T. J. and Wilke, C. O. (2004). Evolution of resource competition between mutually dependent digital organisms. *Artificial Life*, 10(2):145–56.

Katsuyama, C., Nakaoka, S., Takeuchi, Y., Tago, K., Hayatsu, M., and Kato, K. (2009). Complementary cooperation between two syntrophic bacteria in pesticide degradation. *Journal of Theoretical Biology*, 256(4):644–654.

Kauffman, S. and Levin, S. (1987). Towards a general theory of adaptive walks on rugged landscapes. *Journal of Theoretical Biology*, 128(1):11–45.

Keren, L., Hausser, J., Lotan-Pompan, M., Vainberg Slutskin, I., Alisar, H., Kaminski, S., Weinberger, A., Alon, U., Milo, R., and Segal, E. (2016). Massively Parallel Interrogation of the Effects of Gene Expression Levels on Fitness. *Cell*, 166(5):1282–1294.e18.

Kirschner, M. and Gerhart, J. (1998). Evolvability. *Proceedings of the National Academy of Sciences of the United States of America*, 95(15):8420–7.

Knibbe, C. (2007). Structuration des génomes par sélection indirecte de la variabilité mutationnelle : une approche de modélisation et de simulation. *Thèse de Doctorat*, page 177.

Knibbe, C., Coulon, A., Mazet, O., Fayard, J. M., and Beslon, G. (2007a). A long-term evolutionary pressure on the amount of noncoding DNA. *Molecular Biology and Evolution*, 24(10):2344–2353.

Knibbe, C., Mazet, O., Chaudier, F., Fayard, J. M., and Beslon, G. (2007b). Evolutionary coupling between the deleteriousness of gene mutations and the amount of non-coding sequences. *Journal of Theoretical Biology*, 244(4):621–630.

Knibbe, C. and Parsons, D. P. (2014). What happened to my genes? Insights on gene family dynamics from digital genetics experiments. In *ALIFE 14 (14th Intl. Conf. on the Synthesis and Simulation of Living Systems), New York, New York (United States)*, volume 14, pages 33–40. MIT Press.

Kupiec, J. and Sonigo, P. (2000). *Ni Dieu Ni Gène Pour Une Autre Théorie de L'Hérédité*. Seuil, Paris (France).

Kupiec, J.-J. (2008). *L'origine des individus*. Fayard, Paris (France).

Kussell, E. and Leibler, S. (2005). Phenotypic Diversity, Population Growth, and Information in Fluctuating Environments. *Science*, 309(5743):2075–2078.

Laland, K. N., Uller, T., Feldman, M. W., Sterelny, K., Müller, G. B., Moczek, A., Jablonka, E., and Odling-Smee, J. (2015). The extended evolutionary synthesis: its structure, assumptions and predictions. *Proceedings of the Royal Society B*, 282(1813):20151019.

Lande, R. (1976). Natural selection and random genetic drift in phenotypic evolution. *Evolution*, 30(2):314–334.

Lande, R. and Arnold, S. J. (1983). The measurement of selection on correlated characters. *Evolution*, 37(6):1210–1226.

Lavelle, C., Berry, H., Beslon, G., Ginelli, F., Giavitto, J. L., Kapoula, Z., Le Bivic, a., Peyrieras, N., Radulescu, O., and Six, a. (2008). From Molecules to Organisms: Towards Multiscale Integrated Models of Biological Systems. *Theoretical Biology Insights*, 1:13–22.

Legac, M. L., Plucain, J., Hindré, T., Lenski, R. E., and Schneider, D. (2012). Ecological and evolutionary dynamics of coexisting lineages during a long-term experiment with Escherichia coli. *Proceedings of the National Academy of Sciences of the United States of America*, 109(24):9487–9492.

Lehner, B. (2008). Selection to minimise noise in living systems and its implications for the evolution of gene expression. *Molecular Systems Biology*, 4(170):170.

Lempe, J., Lachowiec, J., Sullivan, A. M., and Queitsch, C. (2013). Molecular mechanisms of robustness in plants. *Current Opinion in Plant Biology*, 16(1):62–69.

Levy, S. F., Blundell, J. R., Venkataram, S., Petrov, D. A., Fisher, D. S., and Sherlock, G. (2015). Quantitative evolutionary dynamics using high-resolution lineage tracking. *Nature*, 519(7542):181–6.

Levy, S. F. and Siegal, M. L. (2008). Network hubs buffer environmental variation in Saccharomyces cerevisiae. *PLoS Biology*, 6(11):2588–2604.

Liu, J., Martin-Yken, H., Bigey, F., Dequin, S., François, J. M., and Capp, J. P. (2015). Natural yeast promoter variants reveal epistasis in the generation of transcriptional-mediated noise and its potential benefit in stressful conditions. *Genome Biology and Evolution*, 7(4):969–984.

Liu, Y. and Sumpter, D. (2017). Insights into resource consumption, cross-feeding, system collapse, stability and biodiversity from an artificial ecosystem. *Journal of The Royal Society Interface*, 14:20160816.

Loehle, C. (1990). A Guide to Increased Creativity in Research: Inspiration or Perspiration? *BioScience*, 40(2):123–129.

Marbach, D., Mattiussi, C., and Floreano, D. (2009). Replaying the evolutionary tape: Biomimetic reverse engineering of gene networks. *Annals of the New York Academy of Sciences*, 1158:234–245.

Martin, G. (2014). Fisher's geometrical model emerges as a property of complex integrated phenotypic networks. *Genetics*, 197(1):237–255.

Martin, G. and Lenormand, T. (2006). A general multivariate extension of Fisher's geometrical model and the distribution of mutation fitness effects across species. *Evolution*, 60(5):893–907.

Mattick, J. S. (2009). Deconstructing the dogma: A new view of the evolution and genetic programming of complex organisms. *Annals of the New York Academy of Sciences*, 1178:29–46.

Mattiussi, C. and Floreano, D. (2007). Analog genetic encoding for the evolution of circuits and networks. *IEEE Transactions on Evolutionary Computation*, 11(5):596–607.

Maurizi, M. R. (1992). Proteases and protein degradation in Escherichia coli. *Experientia*, 48(2):178–201.

Metzger, B. P. H., Yuan, D. C., Gruber, J. D., Duveau, F. D., and Wittkopp, P. J. (2015). Selection on noise constrains variation in a eukaryotic promoter. *Nature*, 521(521):344–347.

Mineta, K., Matsumoto, T., Osada, N., and Araki, H. (2015). Population genetics of non-genetic traits: Evolutionary roles of stochasticity in gene expression. *Gene*, 562(1):16–21.

Monod, J. (1970). *Le hasard et la nécessité*. Seuil, Paris (France).

Mozhayskiy, V. and Tagkopoulos, I. (2013). Microbial evolution in vivo and in silico: methods and applications. *Integrative Biology*, 5(2):262–277.

Mustonen, V. and Lässig, M. (2009). From fitness landscapes to seascapes: non-equilibrium dynamics of selection and adaptation. *Trends in Genetics*, 25(3):111–119.

Newman, J. R. S., Ghaemmaghami, S., Ihmels, J., Breslow, D. K., Noble, M., DeRisi, J. L., and Weissman, J. S. (2006). Single-cell proteomic analysis of S. cerevisiae reveals the architecture of biological noise. *Nature*, 441(7095):840–846.

Ofria, C. and Wilke, C. O. (2004). Avida: A Software Platform for Research in Computational Evolutionary Biology. *Artificial Life*, 10(2):191–229.

Ohya, Y., Kimori, Y., Okada, H., and Ohnuki, S. (2015). Single-cell phenomics in budding yeast. *Molecular Biology of the Cell*, 26(22):3920–3925.

O'Neill, B. (2003). Digital Evolution. *PLoS Biology*, 1(1):e18.

Orr, H. A. (2000). Adaptation and the Cost of Complexity. *Evolution*, 54(1):13–20.

Orr, H. A. (2005). The genetic theory of adaptation: a brief history. *Nature Reviews Genetics*, 6(2):119–127.

Orth, J., Thiele, I., and Bernhard (2010). What is flux balance analysis? *Nature Biotechnology*, 28(3):245–248.

Ozbudak, E. M., Thattai, M., Kurtser, I., Grossman, A. D., and van Oudenaarden, A. (2002). Regulation of noise in the expression of a single gene. *Nature Genetics*, 31(1):69–73.

Paaby, A. B. and Rockman, M. V. (2013). The many faces of pleiotropy. *Trends in Genetics*, 29(2):66–73.

Peck, S. L. (2004). Simulation as experiment: A philosophical reassessment for biological modeling. *Trends in Ecology and Evolution*, 19(10):530–534.

Pennisi, E. (1998). How the Genome Readies Itself for Evolution. *Science*, 281(21):1131–1133.

Pfeiffer, T. and Bonhoeffer, S. (2004). Evolution of cross-feeding in microbial populations. *The American Naturalist*, 163(6):E126–E135.

Philippe, N., Crozat, E., Lenski, R. E., and Schneider, D. (2007). Evolution of global regulatory networks during a long-term experiment with Escherichia coli. *BioEssays*, 29(9):846–860.

Pisco, A. O., Brock, A., Zhou, J., Moor, A., Mojtahedi, M., Jackson, D., and Huang, S. (2013). Non-Darwinian dynamics in therapy-induced cancer drug resistance. *Nature Communications*, 4:2467.

Rainey, P. and Rainey, K. (2003). Evolution of cooperation and conflict in experimental bacterial populations. *Nature*, 425(September):72–74.

Rainey, P. B. and Travisano, M. (1998). Adaptive radiation in a heterogeneous environment. *Nature*, 394(6688):69–72.

Raser, J. M. and O'Shea, E. K. (2005). Noise in Gene Expression: Orgins, Consequences, and Control. *Science*, 309(5743):2010–2013.

Ray, T. S. (1993). An Evolutionary Approach to Synthetic Biology: Zen and the Art of Creating Life. *Artificial Life*, 1(1–2):179–209.

Reisinger, J. and Miikkulainen, R. (2006). Selecting for evolvable representations. *Proceedings of the 8th annual conference on Genetic*, pages 1297–1304.

Reynolds, C. W. (1987). Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, 21(4):25–34.

Ribeck, N. and Lenski, R. E. (2015). Modeling and quantifying frequency-dependent fitness in microbial populations with cross-feeding interactions. *Evolution*, 69(5):1313–1320.

Richard, A., Boullu, L., Herbach, U., Bonnafoux, A., Morin, V., Vallin, E., Guillemin, A., Papili Gao, N., Gunawan, R., Cosette, J., Arnaud, O., Kupiec, J. J., Espinasse, T., Gonin-Giraud, S., and Gandrillon, O. (2016). Single-Cell-Based Analysis Highlights a Surge in Cell-to-Cell Molecular Variability Preceding Irreversible Commitment in a Differentiation Process. *PLoS Biology*, 14(12):e1002585.

Rivoire, O. and Leibler, S. (2014). A model for the generation and transmission of variations in evolution. *Proceedings of the National Academy of Sciences*, 111(19):E1940–E1949.

Rocabert, C., Knibbe, C., Consuegra, J., Schneider, D., and Beslon, G. (2017). Beware batch culture: Seasonality and niche construction predicted to favor bacterial adaptive diversification. *PLoS Computational Biology*, 13(3):1–32.

Rosenzweig, R. F., Sharp, R. R., Treves, D. S., and Adams, J. (1994). Microbial evolution in a simple unstructured environment: Genetic differentiation in Escherichia coli. *Genetics*, 137(4):903–917.

Rozen, D. E. and Lenski, R. E. (2000). Long-Term Experimental Evolution in Escherichia coli. VIII. Dynamics of a Balanced Polymorphism. *The American Naturalist*, 155(1):24–35.

Rozen, D. E., Philippe, N., Arjan De Visser, J., Lenski, R. E., and Schneider, D. (2009). Death and cannibalism in a seasonal environment facilitate bacterial coexistence. *Ecology Letters*, 12(1):34–44.

Rozen, D. E., Schneider, D., and Lenski, R. E. (2005). Long-term experimental evolution in Escherichia coli. XIII. Phylogenetic history of a balanced polymorphism. *Journal of Molecular Evolution*, 61(2):171–180.

Ryall, B., Eydallin, G., and Ferenci, T. (2012). Culture history and population heterogeneity as determinants of bacterial adaptation: the adaptomics of a single environmental transition. *Microbiology and Molecular Biology Reviews*, 76(3):597–625.

Savageau, M. A. (1998). Demand theory of gene regulation. I. Quantitative development of the theory. *Genetics*, 149(4):1665–1676.

Schrödinger, E. (1944). *What is life? With mind and matter and autobiographical sketches*. Cambridge University Press, Cambridge (UK).

Servedio, M. R., Brandvain, Y., Dhole, S., Fitzpatrick, C. L., Goldberg, E. E., Stern, C. A., Van Cleve, J., and Yeh, D. J. (2014). Not Just a Theory - The Utility of Mathematical Models in Evolutionary Biology. *PLoS Biology*, 12(12):1–5.

Shen, X., Pettersson, M., Rönnegård, L., and Carlborg, Ö. (2012). Inheritance Beyond Plain Heritability: Variance-Controlling Genes in Arabidopsis thaliana. *PLoS Genetics*, 8(8):e1002839.

Shinar, G., Dekel, E., Tlusty, T., and Alon, U. (2006). Rules for biological regulation based on error minimization. *Proceedings of the National Academy of Sciences of the United States of America*, 103(11):3999–4004.

Singh, D. K., Ku, C.-J., Wichaidit, C., Steininger, R. J., Wu, L. F., and Altschuler, S. J. (2010). Patterns of basal signaling heterogeneity can distinguish cellular populations with different drug sensitivities. *Molecular Systems Biology*, 6(1).

Smith, J. M. and Szathmary, E. (1997). *The major transitions in evolution*. Oxford University Press, Oxford (UK).

Spencer, C. C., Saxer, G., Travisano, M., and Doebeli, M. (2007). Seasonal resource oscillations maintain diversity in bacterial microcosms. *Evolutionary Ecology Research*, 9(5):775–787.

Spencer, H. (1864). *The Principles of Biology*. Williams and Norgate, London (UK).

Stams, A. (1994). Metabolic Interactions Between Anaerobic-Bacteria in Methanogenic Environments. *Antonie Van Leeuwenhoek International Journal of General and Molecular Microbiology*, 66(1-3):271–294.

Stearns, S. C. (1989). The Evolutionary Significance of Phenotypic Plasticity. *BioScience*, 39(7):436– 445.

Stewart, F. M. and Levin, B. R. (1973). Partitioning of resources and the outcome of interspecific competition: a model and some general considerations. *The American Naturalist*, 107(954):171–198.

Symmons, O. and Raj, A. (2016). What's Luck Got to Do with It: Single Cells, Multiple Fates, and Biological Nondeterminism. *Molecular Cell*, 62(5):788–802.

ten Tusscher, K. H. W. J. and Hogeweg, P. (2009). The role of genome and gene regulatory network canalization in the evolution of multi-trait polymorphisms and sympatric speciation. *BMC Evolutionary Biology*, 9:159.

Tenaillon, O. (2014). The Utility of Fisher's Geometric Model in Evolutionary Genetics. *Annual Review of Ecolology, Evolution, and Systematics*, 45:179–201.

Tenaillon, O., Taddei, F., Radman, M., and Matic, I. (2001). Second-order selection in bacterial evolution: Selection acting on mutation and recombination rates in the course of adaptation. *Research in Microbiology*, 152(1):11–16.

Tenaillon, O., Toupance, B., Nagard, H. L., Taddei, F., and Godelle, B. (1999). Mutators, population size, adaptive landscape and the adaptation of asexual populations of bacteria. *Genetics*, 152(2):485–493.

Treves, D. S., Manning, S., and Adams, J. (1998). Repeated evolution of an acetate-crossfeeding polymorphism in long-term populations of Escherichia coli. *Molecular Biology and Evolution*, 15(7):789–97.

Tsuru, S., Yasuda, N., Murakami, Y., Ushioda, J., Kashiwagi, A., Suzuki, S., Mori, K., Ying, B.-W., and Yomo, T. (2011). Adaptation by stochastic switching of a monostable genetic circuit in Escherichia coli. *Molecular Systems Biology*, 7(1):493.

Turner, P. E., Souza, V., and Lenski, R. E. (1996). Tests of ecological mechanisms promoting the stable coexistence of two bacterial genotypes. *Ecology*, 77(7):2119–2129.

Veening, J.-W., Smits, W. K., and Kuipers, O. P. (2008). Bistability, Epigenetics, and Bet-Hedging in Bacteria. *Annual Review of Microbiology*, 62(1):193–210.

Venkataram, S., Dunn, B., Li, Y., Agarwala, A., Chang, J., Ebel, E. R., Geiler-Samerotte, K., Hérissant, L., Blundell, J. R., Levy, S. F., Fisher, D. S., Sherlock, G., and Petrov, D. A. (2016). Development of a Comprehensive Genotype-to-Fitness Map of Adaptation-Driving Mutations in Yeast. *Cell*, 166(6):1585–1596.e22.

Viñuelas, J., Kaneko, G., Coulon, A., Beslon, G., and Gandrillon, O. (2012). Towards experimental manipulation of stochasticity in gene expression. *Progress in Biophysics and Molecular Biology*, 110(1):44–53.

Waddington, C. H. (1942). Canalization of development and the inheritance of acquired characters. *Nature*, 150(3805):405–406.

Wagner, A. (2005). Robustness, evolvability, and neutrality. *FEBS Letters*, 579(8):1772–1778.

Wagner, A. (2008). Neutralism and selectionism: a network-based reconciliation. *Nature Reviews Genetics*, 9(12):965–974.

Wagner, A. (2012). The role of robustness in phenotypic adaptation and innovation. *Proceedings of the Royal Society B*, 279(1732):1249–1258.

Wagner, A. (2013). *Robustness and evolvability in living systems*. Princeton University Press, Princeton, New Jersey (United States).

Wagner, G. P. and Zhang, J. (2011). Fundamental concepts in genetics: The pleiotropic structure of the genotype–phenotype map: the evolvability of complex organisms. *Nature Reviews Genetics*, 12(3):204–213.

Wang, Z. and Zhang, J. (2011). Impact of gene expression noise on organismal fitness and the efficacy of natural selection. *Proceedings of the National Academy of Sciences of the United States of America*, 108(16):E67–76.

Watson, H. W. and Galton, F. (1875). On the Probability of the Extinction of Families. *The Journal of the Anthropological Institute of Great Britain and Ireland*, 4(1875):138–144.

Watson, J. D. and Crick, F. H. C. (1953). Molecular Structure Of Nucleic Acids. *Nature*, 171(4356):737–738.

Watson, R. A. and Szathmáry, E. (2016). How Can Evolution Learn? *Trends in Ecology and Evolution*, 31(2):147–157.

Weiße, A. Y., Oyarzún, D. A., Danos, V., and Swain, P. S. (2015). Mechanistic links between cellular trade-offs, gene expression, and growth. *Proceedings of the National Academy of Sciences of the United States of America*, 112(9):E1038–1047.

Welch, J. J. and Waxman, D. (2003). Modularity and the cost of complexity. *Evolution*, 57(8):1723–1734.

Wilke, C. O., Wang, J. L., Ofria, C., Lenski, R. E., and Adami, C. (2001). Evolution of digital organisms at high mutation rates leads to survival of the flattest. *Nature*, 412(6844):331–333.

Williams, H. T. P. and Lenton, T. M. (2010). Evolutionary regime shifts in simulated ecosystems. *Oikos*, 119(12):1887–1899.

Wright, S. (1932). The roles of mutation, inbreeding, crossbreeding and selection in evolution.

Yvert, G., Ohnuki, S., Nogami, S., Imanaga, Y., Fehrmann, S., Schacherer, J., and Ohya, Y. (2013). Single-cell phenomics reveals intra-species variation of phenotypic noise in yeast. *BMC Systems Biology*, 7(1):54.

Zhang, Z., Qian, W., and Zhang, J. (2009). Positive selection for elevated gene expression noise in yeast. *Molecular Systems Biology*, 5(1):299.

Zhang, Q.,Lambert, G., Liao, D., Kim, H., Robin, K., Tung, C.-k., Pourmand, N., Austin, R. H. (2011). Acceleration of Emergence of Bacterial Antibiotic Resistance in Connected Microenvironments. *Science*, 1764(September):1764–1767.

Zhuravel, D., Fraser, D., St-Pierre, S., Tepliakova, L., Pang, W. L., Hasty, J., and Kærn, M. (2010). Phenotypic impact of regulatory noise in cellular stress-response pathways. *Systems and Synthetic Biology*, 4(2):105–116.

Appendix A

Evo^{2}Sim user manual

Evo^{2}Sim User Manual

for version 1.0.1 (October 29, 2017)

Contents

Chapter 1

Introduction

1.1 Introducing Evo²Sim

Evo²Sim is a multi-scale model of *in silico* experimental evolution, the virtual pendant of experimental evolution in wet laboratory (see Fig 1.1). The software is equiped with the whole tool case of experimental setups, competition assays, phylogenetic analysis, and, most importantly, allowing for evolvable ecological interactions. Digital organisms with an evolvable genome structure, encoding evolvable genetic regulation and metabolic networks are evolved for tens of thousands of generations in environments mimicking the dynamics of real controlled environments, including chemostat or batch culture.

Evo²Sim was developed under EvoEvo (http://www.evoevo.eu/), a FP7-ICT project funded by the European Commission (FP7-ICT-610427). The source code is written in C++.

You can find more details on software description and development on Github page charlesrocabert/Evo2Sim. A website fully dedicated to Evo²Sim is coming soon.

1.2 License

1.3 Community

Evo²Sim was developed by Charles Rocabert, Carole Knibbe and Guillaume Beslon, under the EvoEvo project. The list of contributors is displayed in text-file `AUTHORS` of Evo²Sim package. You shall find more details on http://www.evoevo.eu/community/.

1.4 Download

Evo^{2}Sim last releases are available on Github page charlesrocabert/Evo2Sim.

1.5 Contact

For any question about the software, do not hesitate to contact us at http://www.evoevo.eu/contact-us/.

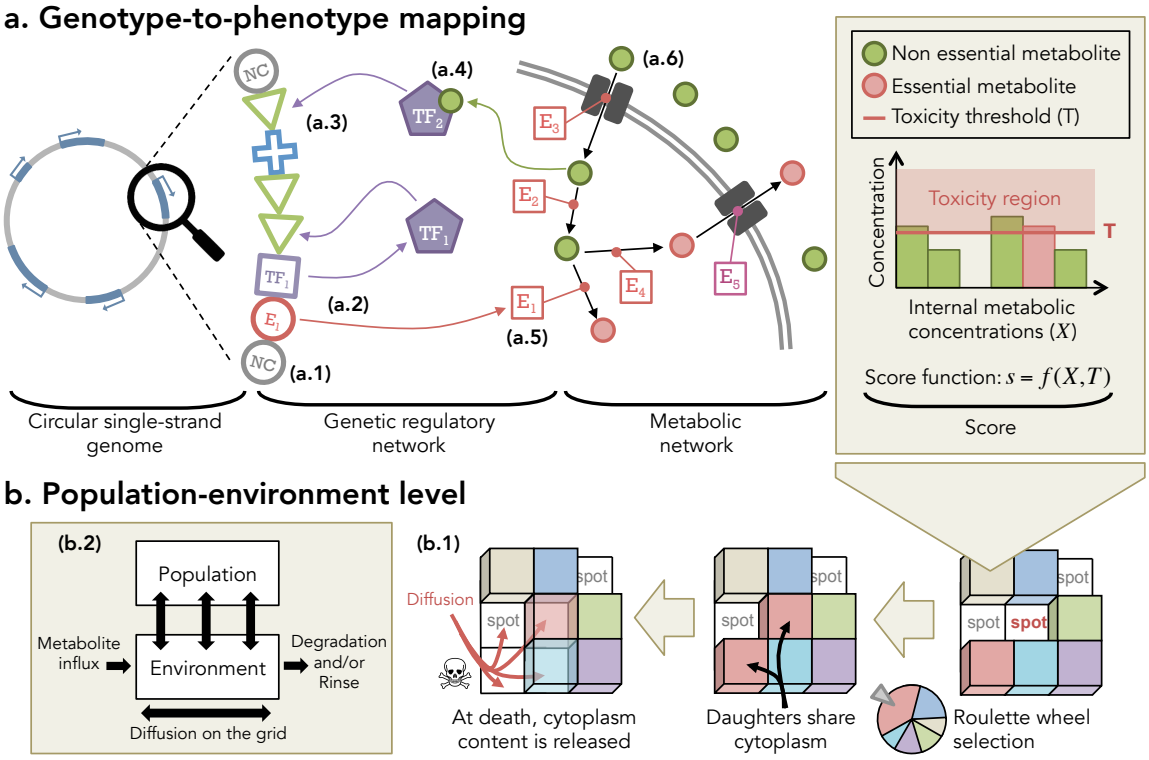


Figure 1.1: **Global picture of Evo²Sim. a. Description of the genotype-to-phenotype mapping.** Organisms own a coarse-grained genome made of units. This genome is a circular single-strand sequence, with a unique reading frame. Non coding **(NC)** units are not functional **(a.1)**. The arrangement of the units on the sequence defines functional regions, where a promoter (**P**, blue cross) controls the expression of enzyme coding units (**E**, red circles) or transcription factor coding units (**TF**, purple squares), thereby allowing for operons (here, one E and one TF). When coding units are expressed **(a.2)**, they contribute to the genetic regulatory network (for TFs) and the metabolic network (for Es). Depending on their attributes, transcription factors bind on binding sites. **(a.3)** If they bind on the enhancer sequence (binding sites flanking the promoter upstream), the promoter activity is up-regulated. If they bind on the operator sequence (binding sites flanking the promoter downstream), the promoter activity is down-regulated. **(a.4)** Metabolites can bind on a transcription factor as co-enzymes, and activate or inhibit it, depending on transcription factor attributes. Enzymes perform metabolic reactions in the cytoplasm **(a.5)**, or pump metabolites in or out **(a.6)**. The score of an organism is computed from its "essential metabolites" (usually the score is the sum of essential metabolite concentrations). Lethal toxicity thresholds are applied to each metabolic concentration and forbid organisms to accumulate resources. **b. Description of the population and environment levels.** Organisms are placed on a 2D toroidal grid, and compete for resources and space. When an organism dies, it leaves its grid cell empty and organisms in the Moore neighborhood (if any) compete to divide in available space. The competition is based on scores, a minimal threshold being applied on scores to forbid worst organisms to divide. At division, daughters share cytoplasm content (enzymes and metabolites). At death, metabolites from the cytoplasm are released in the local environment, and diffuse on the grid **(b.1)**. On the largest scale, the population evolves on the environment by up-taking, transforming and releasing metabolites. Metabolites then diffuse and are degraded. This strong interaction between the population and the environment allows for the evolution of complex ecological situations, depending on environmental properties **(b.2)**.

Chapter 2

Installation instructions

Download the latest release of Evo^{2}Sim on Github page charlesrocabert/Evo2Sim and save it to a directory of your choice. Open a terminal and use the `cd` command to navigate to this directory. Then follow the steps below to compile and build the executables.

2.1 Supported platforms

Evo^{2}Sim software has been successfully tested on Ubuntu 12.04 LTS, Ubuntu 14.04 LTS, OSX 10.9.5 (Maverick) and OSX 10.10.1 (Yosemite).

2.1.1 Required dependencies

- A `C++` compiler (GCC, LLVM, ...)
- `CMake` (command line version)
- `zlib`
- `GSL`
- `CBLAS`
- `TBB`
- `R` (packages `ape` and `RColorBrewer` are needed)

2.1.2 Optional dependencies (for graphical outputs)

- `X11` (or `XQuartz` on latest OSX version)
- `SFML 2`
- `matplotlib` (this python library is needed for the script `track_cell.py` (see below)

2.1.3 HTML viewer dependencies

- Javascript must be activated in your favorite internet browser

Note, however, that Evo^{2}Sim can be compiled without graphical outputs, and hence no need for X and SFML libraries (see compilation instructions below for more information). This option is useful if you want to run Evo^{2}Sim on a computer cluster, for example.

2.2 Software compilation

2.2.1 User mode

To compile Evo^{2}Sim, run the following instructions on the command line:

```
$ cd cmake/
```

and

```
$ bash make.sh
```

To gain performances during large experimental protocols, or on computer cluster, you should compile the software without graphical outputs:

```
$ bash make_no_graphics.sh
```

2.2.2 Debug mode

To compile the software in `DEBUG` mode, use `make_debug.sh` script instead of `make.sh`:

```
$ bash make_debug.sh
```

When Evo^{2}Sim is compiled in `DEBUG` mode, a lot of tests are computed on the fly during a simulation (*e.g.* integrity tests on phylogenetic trees, or on the ODE solver ...). For this reason, this mode should only be used for test or development phases. Moreover, unitary and integrated tests must be ran in `DEBUG` mode (see below).

2.2.3 Executable files emplacement

Binary executable files are in `build/bin` folder.

Chapter 3

Typical usage

Evo^{2}Sim includes three main executables (`evo2sim_create`, `evo2sim_bootstrap` and `evo2sim_run`), and a set of executables dedicated to post-treatments, data recovery or tests.

Everything in Evo^{2}Sim relies on an ad-hoc file organization where all the data for a simulation is stored: populations in the `population` directory, environments in `environment`, phylogenetic and lineage trees in `tree` and so on. It is not recommended to manually modify these files since this may cause some inconsistency leading to undefined behavior. Besides, most of these files are compressed.

Open a terminal and use the `cd` command to navigate to Evo^{2}Sim directory. A typical parameters file is provided in Evo^{2}Sim package, in folder `example` (an exhaustive description of the parameters is available in chapter "Parameters description"). Navigate to this folder using the `cd` command. Then follow the steps below for a first usage of the software.

3.1 Creating a simulation

Create a fresh simulation from the parameters file (by default `parameters.txt`):

```
$ ../build/bin/evo2sim_create
```

Several folders have been created. They mainly contain simulation backups (population, environment, trees, parameters, ...). Additional files and folders have also been created:

- `version.txt`: this file indicates the version of the software. This information is useful to ensure that the code version is compatible with the backup files (*e.g.*, in case of post-treatments).
- `track_cell.py`: when executed, this python script displays on the fly the internal protein and metabolic concentrations of the cell at position 0×0 on the grid. This script is useful to get an idea of internal cell's dynamics (metabolic fluxes, regulation, ...).
- `viewer` folder: the viewer is central to the usage of Evo^{2}Sim (see chapter "Simulation viewer"). To access the viewer, open the html page `viewer/viewer.html` in an internet browser.

3.2 Generating viable initial conditions with a bootstrap

Alternatively to the `evo2sim_create` executable, use a bootstrap to find a simulation with good initial properties from the parameters file:

```
$ ../build/bin/evo2sim_bootstrap
```

A fresh simulation with an updated parameters file will be automatically created if a suitable seed is found.

3.3 Running a simulation

In Evo^{2}Sim, running a simulation necessitates to load it from backup files. Here, we will run a simulation from freshly created backups (see above):

```
$ ../build/bin/evo2sim_run -b 0 -t 10000 -g
```

with `-b` the date of the backup, here 0 (fresh simulation), `-t` the simulation time, here 10,000 time-steps. Option `-g` activates the graphical output (not works if the software has been compiled with the no-graphics option). At any moment during the simulation, you can take a closer look at the evolution of the system by opening `viewer/viewer.html` in an internet browser. You can track internal cell's dynamics by executing the script `track_cell.py`.

Other main executables are described below in section "Main executables description". You can also obtain help by running the executable with the `-h` option (e.g. `evo2sim_create -h`)

Chapter 4

Simulation viewer

Evo^{2}Sim comes with an HTML viewer displaying a very complete set of live statistics. Each new simulation owns a dedicated viewer, which is frequently actualized on the fly (by default, every 500 simulation time-steps). This viewer has been developed using `Bootstrap`, `DyGraph`, `CytoscapeJS`, `ChartJS` and `JQuery`.

To access the viewer, simply open `viewer/viewer.html` in an internet browser (Javascript must be enabled). The different tabs are described below.

4.1 Population

This page displays the evolution of main population statistics (population size, mean genome size, mean score, ...), as well as the evolution of the trophic network.

4.2 Best lineage

This page displays the evolution of last best individual statistics. These informations are the most representative of evolutionary dynamics, since they contains all the mutations fixed since the beginning of the simulation.

4.3 Best individual

This page displays some informations about the last best individual at the moment of the visualization (genome state, genetic regulation network, metabolic network, internal metabolic state, ...).

4.4 Environment

This page displays the evolution of main environment statistics, as well as its current state.

4.5 Phylogeny

This page displays various rendering of the current phylogenetic tree, as well as some evolution statistics (number of nodes, common ancestor age, ...).

4.6 Parameters

This page displays the parameters file used to create the simulation, as well as a short description of parameters usage.

Appendix A

Main executables description

A.1 `evo2sim_create` executable

Create a fresh simulation from a parameters file.

Usage:

$ `evo2sim_create -h` or `--help`

or

$ `evo2sim_create [options]`

Options are:

`-h, --help`**:** print this help, then exit (optional)

`-v, --version`**:** print the current version, then exit (optional)

`-f, --file`**:** specify the parameters file (default: `parameters.txt`)

`-rs, --random-seed`**:** the prng seed is drawn at random (optional)

Be aware that creating a simulation in a folder completely erases previous simulation.

A.2 `evo2sim_bootstrap` executable

Run a bootstrap to find viable initial conditions.

Usage:

$ `evo2sim_bootstrap -h` or `--help`

or

$ `evo2sim_bootstrap [options]`

Options are:

`-h, --help:` print this help, then exit (optional)

`-v, --version:` print the current version, then exit (optional)

`-f, --file:` specify the parameters file (default: `parameters.txt`)

`-min, --minimum-time:` specify the minimum time the new population must survive (default: 100)

`-pop, --minimum-pop-size:` specify the minimum size the new population must maintain (default: 500)

`-t, -trials:` specify the number of trials (default: 1000)

`-g, --graphics:` activate graphic display (optional)

A simulation is automatically created if good conditions are found. The parameters file is also edited to include the corresponding prng seed value. Be aware that creating a simulation in a folder completely erases previous simulation.

A.3 `evo2sim_run` executable

Run a simulation from backup files.

Usage:

$ `evo2sim_run -h` or `--help`

or

$ `evo2sim_run [options]`

Options are:

`-h, --help:` print this help, then exit (optional)

`-v, --version:` print the current version, then exit (optional)

`-b, --backup-time:` set the date of the backup to load (default: 0)

`-t, --simulation-time:` set the duration of the simulation (default: 10000)

`-g, --graphics:` activate graphic display (optional)

Statistic files content is automatically managed when a simulation is reloaded from backup to avoid data loss.

A.4 `evo2sim_generate_figures` executable

Extract statistics and generate viewer figures from backup files.

Usage:

$ `evo2sim_generate_figures -h` or `--help`

or

$ `evo2sim_generate_figures [options]`

Options are:

`-h, --help`: print this help, then exit (optional)

`-v, --version`: print the current version, then exit (optional)

`-b, --backup-time`: set the date of the backup to load (mandatory)

A.5 `evo2sim_recover_parameters` executable

Recover the parameters file from backup files.

Usage:

$ `evo2sim_recover_parameters -h` or `--help`

or

$ `evo2sim_recover_parameters [options]`

Options are:

`-h, --help`: print this help, then exit (optional)

`-v, --version`: print the current version, then exit (optional)

`-f, --file`: specify the name of the parameters file to save (mandatory)

A.6 `evo2sim_unitary_tests` executable

Run unitary tests.

Usage:

$ `evo2sim_unitary_tests -h` or `--help`

or

$ `evo2sim_unitary_tests [options]`

Options are:

`-h, --help`: print this help, then exit (optional)

`-v, --version`: print the current version, then exit (optional)

`-f, --file`: specify the parameters file (default: `parameters.txt`)

To use the unitary tests, the software must be compiled in DEBUG mode (see installation instructions below).

A.7 `evo2sim_integrated_tests` executable

Run integrated tests.

Usage:

```
$ evo2sim_integrated_tests -h or --help
```

or

```
$ evo2sim_integrated_tests [options]
```

Options are:

`-h, --help`: print this help, then exit (optional)

`-v, --version`: print the current version, then exit (optional)

`-f, --file`: specify the parameters file (default: `parameters.txt`)

`-tests, --number-of-tests`: specify the number of tests with different seeds (default: 1)

`-steps, --number-of-steps`: specify the number of steps by test (default: 1)

`-rs, --random-seed`: the prng seed is drawn at random for each test (optional)

`-rp, --random-parameters`: the parameters are drawn at random for each test (optional)

To use the unitary tests, the software must be compiled in DEBUG mode (see installation instructions below).

A.8 Other executables

For all the other executables, you can obtain help by running the executable with the `-h` option (e.g. `evo2sim_create -h`)

Appendix B

Parameters description

All the parameters of the parameters file are described in details below. Each parameters receive at least on value. There is three types of values:

- **integer**: integer number
- **float**: floating point number
- **string**: characters string

For each parameter, the type is possibly bounded. In this case, boundaries are indicated.

B.1 Pseudorandom numbers generator

SEED <seed> (integer > 0)

Simply set the seed of the pseudorandom numbers generator (prng). The seed value is important since it allows to exactly replay a simulation if needed.

B.2 Parallel computing

PARALLEL_COMPUTING <choice> (YES/NO)

This parameter allows to activate, or deactivate, parallel computing at will. Parallel computing is managed by the external library `TBB`.

B.3 Simulation schemes

B.3.1 Energy costs scheme

ENERGY_COSTS_SCHEME <choice> (YES/NO)

Choose the energy scheme. By default, biochemical reactions are energy free in Evo^{2}Sim. When energy costs are activated, inner cell's chemical reactions produce or consume energy (an abstract view of energy carriers, like ATP). Transcription, enzymatic reactions and pumps else produce or cost energy to the cell, which must maintain its energy level to survive. Specific parameters are used to precisely set energy costs (see below).

B.3.2 Membrane permeability scheme

MEMBRANE_PERMEABILITY_SCHEME <choice> (YES/NO)

Choose membrane permeability scheme. If membrane permeability is activated, metabolites diffuse through the cell's membrane at a specific rate (see MEMBRANE_PERMEABILITY parameter below).

B.3.3 Metabolic inheritance scheme

METABOLIC_INHERITANCE_SCHEME <choice> (YES/NO)

Choose metabolic inheritance scheme. If this parameter is activated, the two daughter cells share the metabolic content of their parent. Each daughter cell inherits half of metabolic concentrations.

B.3.4 Enzymatic inheritance scheme

ENZYMATIC_INHERITANCE_SCHEME <choice> (YES/NO)

Choose enzymatic inheritance scheme. If this parameter is activated, the two daughter cells share the enzymatic content of their parent. Each daughter cell inherits half of enzymatic concentrations.

B.3.5 Co-enzymes scheme

CO_ENZYME_ACTIVITY <choice> (YES/NO)

Choose co-enzyme scheme. If this parameter is activated, some metabolites act as co-enzymes. Each transcription-factor owns a site where a specific metabolite can bind, activating or inhibiting the transcription factor depending on its properties. Activating this parameter increases the complexity of the genetic regulation network, and more importantly, allows cells to evolve environmental sensing.

B.3.6 Score scheme

SCORE_SCHEME <choice> (SUM/SUM_MINUS_DEV/COMBINATORIAL)

Choose the score scheme. The score of a cell is computed from its internal metabolic concentrations:

- SUM scheme: the score is simply the sum of essential metabolite concentrations;
- SUM_MINUS_DEV scheme: the score is the sum of essential metabolite concentrations, minus the standard deviation of the concentrations. This score adds an homeostatic constraint on cells.

- COMBINATORIAL scheme: the score is computed depending on relative essential metabolite concentrations. Basically, essential metabolites are considered to form complex molecules similar to RNA polymerases. Bigger is the polymerase, higher is its contribution to the score. Then, the bigger polymerase including all the essential metabolites is defined by the lowest concentration. Since the lowest metabolite is exhausted for this polymerase, the next one is the contribution of remaining metabolites, and so forth.

B.3.7 Selection threshold

SELECTION_THRESHOLD <threshold> (float $\in [0, 1]$)

Define a score threshold, above which cell's division is forbidden. When neighboring cells compete for a gap in the environment, one cell is elected at random by a roulette wheel draw, based on relative scores. However, a minimum threshold is mandatory to avoid individuals owning a very low score to divide, and drive the population in an artificial dead-end (where everybody is very bad, but nobody dies).

B.4 Space

B.4.1 Grid width

WIDTH <width> (integer > 0)

Simply define the width of the environmental grid.

B.4.2 Grid height

HEIGHT <height> (integer > 0)

Simply define the height of the environmental grid.

B.5 Output

B.5.1 Simulation backup step

SIMULATION_BACKUP_STEP <step> (integer ≥ 0)

Define the frequency at which backups of the simulation are saved. The resolution is in simulation time-steps. It is possible to exactly replay a simulation from backup files. Be aware that backup files size is large, the backup frequency must be reasonable (*e.g.*, 1,000 time-steps).

B.5.2 Figures generation step

FIGURES_GENERATION_STEP <step> (integer ≥ 0)

Define the frequency at which figures are generated for the html viewer. Some scripts used to generate figures may take more time to execute for very long simulations, the backup frequency must be reasonable (*e.g.*, 1,000 time-steps).

B.6 Genome

B.6.1 Load the genome from file

LOAD_GENOME_FROM_FILE <choice> (YES/NO)

Choose to generate genomes at random (NO, in this case, random generation depends on parameters below), or load a handcrafted genome from a file (YES). In case the handcrafted genome is loaded, it must be encoded in a file named `initial_genome.txt`. The structure of this file is specific and must respect the following scheme:

1. To encode non-coding units (NC), insert the following line: `NC <number of units>`. The specified number of random NC units will be inserted (`<number of units>` > 0);
2. To encode a promoter unit (P), insert the following line: `P <basal expression level>`. A promoter unit with a basal expression level $\beta =$ `<basal expression level>` will be inserted ($\beta \in [0, 1]$);
3. To encode a binding site unit (BS), insert the following line: `BS <TF tag>`. A binding site unit owning the specified transcription factor tag value will be inserted (`<TF tag>` $\in \mathbb{Z}$);
4. To encode a transcription factor coding unit (TF), insert the following line: `TF <BS tag> <CoE tag> <free activity> <bound activity> <binding window>`. A transcription factor coding unit with specified attributes will be inserted (`<BS tag>` $\in \mathbb{Z}$, `<CoE tag>` $\in \mathbb{N}^*$, `<free activity>` $\in \{true, false\}$, `<bound activity>` $\in \{true, false\}$, `<binding window>` ≥ 0);
5. To encode an enzyme coding unit (E), insert the following line: `E <substrate> <product>` `<`k_{cat}`>` `<`K_M`>`. An enzyme coding unit with specified attributes will be inserted (`<substrate>` > 0, `<product>` > 0, `<`k_{cat}`>` and `<`K_M`>` $\in$ specified boundaries).

B.6.2 Metabolite tags initial range

METABOLITE_TAG_INITIAL_RANGE <min> <max> (integer > 0; min $\leq$ max)

Define the initial distribution of metabolite tags encoded in the initial random genome (*i.e.*, in transcription factor and enzyme units). `min` and `max` values define the boundaries of a uniform law, used to draw the metabolite tags.

B.6.3 Binding site tags initial range

BINDING_SITE_TAG_INITIAL_RANGE <min> <max> (float > 0; min $\leq$ max)

Define the initial distribution of binding site tags encoded in the initial random genome (*i.e.*, in transcription factor units). `min` and `max` values define the boundaries of a uniform law, used to draw the binding site tags.

B.6.4 Co-enzyme tags initial range

CO_ENZYME_TAG_INITIAL_RANGE <min> <max> (float > 0; min $\leq$ max)

Define the initial distribution of co-enzyme tags encoded in the initial random genome (*i.e.*, in transcription factor units). `min` and `max` values define the boundaries of a uniform law, used to draw the co-enzyme tags.

B.6.5 Transcription factor tags initial range

TRANSCRIPTION_FACTOR_TAG_INITIAL_RANGE <min> <max> (float > 0; min $\leq$ max)

Define the initial distribution of transcription factor tags encoded in the initial random genome (*i.e.*, in binding site units). `min` and `max` values define the boundaries of a uniform law, used to draw the transcription factor tags.

B.6.6 Transcription factors binding window

TRANSCRIPTION_FACTOR_BINDING_WINDOW <window> (integer $\geq$ 0)

Define the "binding window" of a transcription factor on a binding site. If transcription factors and binding site tags are similar enough, the binding is allowed. More precisely if $\text{tag}_{TF} \in [\text{tag}_{TF} - \text{window} \ , \ \text{tag}_{TF} + \text{window}]$, the binding is possible.

B.6.7 Initial number of non-coding units

INITIAL_NUMBER_OF_NON_CODING_UNITS <number> (integer ≥ 0)

Define the number of random non-coding units in the initial random genome.

B.6.8 Initial number of enzyme coding units

INITIAL_NUMBER_OF_ENZYME_UNITS <number> (integer ≥ 0)

Define the number of random enzyme units in the initial random genome.

B.6.9 Initial number of transcription factor coding units

INITIAL_NUMBER_OF_TRANSCRIPTION_FACTOR_UNITS
<number> (integer ≥ 0)

Define the number of random transcription factor units in the initial random genome.

B.6.10 Initial number of binding site units

INITIAL_NUMBER_OF_BINDING_SITE_UNITS <number> (integer ≥ 0)

Define the number of random binding site units in the initial random genome.

B.6.11 Initial number of promoter units

INITIAL_NUMBER_OF_PROMOTER_UNITS <number> (integer ≥ 0)

Define the number of random promoter units in the initial random genome.

B.6.12 Point mutation rate

POINT_MUTATION_RATE <rate> (float $\in [0, 1]$)

Define the point mutation rate (in attribute^{-1}.replication^{-1}).

B.6.13 Duplication rate

DUPLICATION_RATE <rate> (float $\in [0, 1]$)

Define the duplication rate (in genomic-unit^{-1}.replication^{-1}).

B.6.14 Deletion rate

DELETION_RATE <rate> (float $\in [0, 1]$)

Define the deletion rate (in genomic-unit^{-1}.replication^{-1}).

B.6.15 Translocation rate

TRANSLOCATION_RATE <rate> (float $\in [0, 1]$)

Define the translocation rate (in genomic-unit^{-1}.replication^{-1}).

B.6.16 Inversion rate

INVERSION_RATE <rate> (float $\in [0, 1]$)

Define the inversion rate (in genomic-unit^{-1}.replication^{-1}).

B.6.17 Transition rate

TRANSITION_RATE <rate> (float $\in [0, 1]$)

Define the transition rate (in genomic-unit^{-1}.replication^{-1}).

B.6.18 Breakpoint rate

BREAKPOINT_RATE <rate> (float $\in [0, 1]$)

Define the breakpoint rate (in attribute^{-1}.breakpoint^{-1}).

B.6.19 Substrate tag mutation size

SUBSTRATE_TAG_MUTATION_SIZE <size> (integer ≥ 0)

Define the size of the uniform distribution used to mutate substrate tags (in enzyme units). The mutation is defined as tag + $\mathcal{U}$(-size, +size).

B.6.20 Product tag mutation size

PRODUCT_TAG_MUTATION_SIZE <size> (integer ≥ 0)

Define the size of the uniform distribution used to mutate product tags (in enzyme units). The mutation is defined as tag + $\mathcal{U}$(-size, +size).

B.6.21 k_{cat} mutation size

KCAT_MUTATION_SIZE <size> (float ≥ 0.0)

Define the standard deviation of the gaussian distribution used to mutate k_{cat} constant (in enzyme units). The mutation is defined as $log_{10}(k_{cat})$ + $\mathcal{N}$(0, size).

B.6.22 k_{cat}/k_M ratio mutation size

KCAT_KM_RATIO_MUTATION_SIZE <size> (float ≥ 0.0)

Define the standard deviation of the gaussian distribution used to mutate k_{cat}/k_M ratio (in enzyme units). The mutation is defined as $log_{10}(k_{cat}/k_M)$ + $\mathcal{N}$(0, size).

B.6.23 Binding site tag mutation size

BINDING_SITE_TAG_MUTATION_SIZE <size> (integer ≥ 0)

Define the size of the uniform distribution used to mutate binding site tags (in transcription factor units). The mutation is defined as tag + $\mathcal{U}$(-size, +size).

B.6.24 Co-enzyme tag mutation size

CO_ENZYME_TAG_MUTATION_SIZE <size> (integer ≥ 0)

Define the size of the uniform distribution used to mutate co-enzyme tags (in transcription factor units). The mutation is defined as tag + $\mathcal{U}$(-size, +size).

B.6.25 Transcription factor tag mutation size

TRANSCRIPTION_FACTOR_TAG_MUTATION_SIZE <size> (integer ≥ 0)

Define the size of the uniform distribution used to mutate transcription factor tags (in binding site units). The mutation is defined as tag + $\mathcal{U}$(-size, +size).

B.6.26 Basal expression level mutation size

BASAL_EXPRESSION_LEVEL_MUTATION_SIZE <size> (float ≥ 0.0)

Define the standard deviation of the gaussian distribution used to mutate β constant (in promoter units). The mutation is defined as $\beta + \mathcal{N}(0, \text{size})$.

B.7 Genetic regulation network

B.7.1 Genetic regulation network time-steps ratio

GENETIC_REGULATION_NETWORK_TIMESTEP <time-step> (float > 0.0)

Define the number of ODE time-steps used to solve the genetic regulation network per simulation time-step.

B.7.2 Hill function theta parameter

HILL_FUNCTION_THETA <theta> (float $\in [0, 1]$)

Define the parameter θ of the Hill function used to compute the contribution of the regulation on each promoter transcription.

B.7.3 Hill function n parameter

HILL_FUNCTION_N <n> (float ≥ 0.0)

Define the parameter n of the Hill function used to compute the contribution of the regulation on each promoter transcription.

B.7.4 Protein degradation rate

PROTEIN_DEGRADATION_RATE <rate> (float $\in [0, 1]$)

Define the protein degradation rate per genetic regulation ODE time-step.

B.8 Metabolic network

B.8.1 Metabolism time-steps

METABOLISM_TIMESTEP <time-step> (float > 0.0)

Define the number of ODE time-steps used to solve the metabolic network per simulation time-step.

B.8.2 Essential metabolites toxicity threshold

ESSENTIAL_METABOLITES_TOXICITY_THRESHOLD <threshold> (float > 0.0)

Define the maximum cell's toxicity threshold of essential metabolites. If one essential metabolite overreaches this threshold in cell's cytoplasm, the cell dies.

B.8.3 Non-essential metabolites toxicity threshold

NON_ESSENTIAL_METABOLITES_TOXICITY_THRESHOLD <threshold> (float > 0.0)

Define the maximum cell's toxicity threshold of non-essential metabolites. If one non-essential metabolite overreaches this threshold in cell's cytoplasm, the cell dies.

B.8.4 Initial metabolite amount in cells

INITIAL_METABOLITES_AMOUNT_IN_CELLS <initial_amount> (float ≥ 0.0)

Define the initial amount of metabolites found in cells when the simulation is created from scratch.

B.8.5 Maximum reaction size

MAXIMUM_REACTION_SIZE <size> (integer ≥ 0)

Define the maximum jump size of a metabolic reaction in the metabolic space. Considering s and p to be resp. the tags of the substrate and the product of a metabolic reaction (catalyzed by an enzyme), the reaction only occurs if $|s - p| \leq$ size.

B.9 Energy

B.9.1 Energy transcription cost

ENERGY_TRANSCRIPTION_COST <cost> (float ≥ 0)

Define the cost of producing proteins (mainly by transcription). When a enzyme or transcription factor unit is transcribed at a certain rate e, energy cost is $c = e * cost$. For computation reasons, the energy is not coupled to transcription equations (*i.e.*, the reaction speed of the transcription does not depend on energy concentration). If the cost is set to 0.0, the transcription comes with no energy cost. If the energy becomes negative, the cell dies.

B.9.2 Energy degradation cost

ENERGY_DEGRADATION_COST <cost> (float ≥ 0)

Define the cost of degrading proteins. When proteins are degraded at rate d, energy cost is $c = d * cost$. For computation reasons, the energy is not coupled to degradation equations (*i.e.*, the speed of the degradation does not depend on energy concentration). If the cost is set to 0.0, the degradation comes with no energy cost. If the energy becomes negative, the cell dies.

B.9.3 Energy enzymatic cost

ENERGY_ENZYMATIC_COST <cost> (float ≥ 0)

Define the cost or the production of energy when performing metabolic reactions. Metabolic reactions are performed by enzymes needing or producing energy carrier molecules. Let's consider s and p the tags of resp. the substrate and the product of a metabolic reaction catalyzed by enzyme E. If $s < p$, the reaction consumes energy at rate $c = (p - s) * cost$. If $s > p$, the reaction produces energy at rate $c = (s - p) * cost$. For computation reasons, the energy is not coupled to metabolic reaction equations (*i.e.*, the reaction speed does not depend on energy concentration). If the cost is set to 0.0, metabolic reactions come with no energy cost. If the energy becomes negative, the cell dies.

B.9.4 Energy pumping cost

ENERGY_PUMPING_COST <cost> (float ≥ 0)

Define the cost of pumping in or out metabolites. When metabolites are pumped in or out at rate r by a pump, energy cost is $c = r * cost$. For computation reasons, the energy is not coupled to pump equations (*i.e.*, the reaction speed does not depend on energy concentration). If the cost is set to 0.0, the pumping activity comes with no energy cost. If the energy becomes negative, the cell dies.

B.9.5 Energy dissipation rate

ENERGY_DISSIPATION_RATE <rate> (float $\in [0, 1]$)

Define the rate at which a cell looses its energy stock by dissipation.

B.9.6 Energy toxicity threshold

ENERGY_TOXICITY_THRESHOLD <threshold> (float ≥ 0)

Define a maximum threshold to cell's energy. If a cell energy stock overreaches this threshold, the cell dies.

B.9.7 Initial energy amount in cells

INITIAL_ENERGY_AMOUNT_IN_CELLS <amount> (float ≥ 0)

Define the initial energy amount available in cells when the simulation is created from scratch. This parameter allow for random initialization of complex cells needing energy production to survive.

B.10 Cell

B.10.1 Membrane permeability

MEMBRANE_PERMEABILITY <permeability> (float $\in [0, 1]$)

Define the membrane permeability. Metabolites in cell's cytoplasm or in the local environment diffuse through the cell's membrane depending on their concentrations and the permeability.

B.11 Population

B.11.1 Death probability

DEATH_PROBABILITY <probability> (float $\in [0, 1]$)

Define the probability to die at random per simulation time-step. This probability is the same for every cell, and is constant during cell life. This rate is applied in addition to other death events linked to toxicity thresholds.

B.11.2 Migration rate

MIGRATION_RATE <rate> (float $\in [0, 1]$)

If the migration rate is not null, pairs of random cells exchange their location at a defined rate per simulation time-step. Depending on the strength of the random mixing, cell's behavior evolve differently (*e.g.*, to evolve cooperation).

B.11.3 HGT rate

HGT_RATE <rate> (float $\in [0, 1]$)

Define the probability for a genome to receive alien genetic sequences at replication. Genetic sequences are generated at random, and do not come from the simulated population.

B.12 Environment

B.12.1 Environment initialization cycles

ENVIRONMENT_INITIALIZATION_CYCLES <cycles> (integer ≥ 0)

Define the number of initialization loops applied to a newly created environment. Initialization loops are based on environment parameters defined below. For example, If a concentration $c = 0.1$ of metabolite 10 is introduced in the environment at every simulation time-step, and if 5 initialization cycles are requested, the initial concentration will be 0.1×5. This parameter is useful to allow the environment to reach dynamic equilibrium before introducing new cells.

B.12.2 Environment species tags range

ENVIRONMENT_SPECIES_TAG_RANGE <min> <max> (integer > 0; min $\leq$ max)

Define the boundaries of the uniform law used to draw a new metabolite introduced in the environment.

B.12.3 Environment concentrations range

ENVIRONMENT_CONCENTRATION_RANGE <min> <max> (float > 0.0; min $\leq$ max)

Define the boundaries of the uniform law used to draw the concentration of each new metabolite introduced in the environment.

B.12.4 Environment number of species range

ENVIRONMENT_NUMBER_OF_SPECIES_RANGE <min> <max> (integer > 0.0; min $\leq$ max)

Define the boundaries of the uniform law used to draw the number of metabolites introduced in the environment.

B.12.5 Environment interaction scheme

ENVIRONMENT_INTERACTION_SCHEME <choice>
(NO_INTERACTION/INTERACTION)

Define the interaction scheme between the population and the environment.

- NO_INTERACTION: environment concentrations are not modified by cells. Cells grow on ressources with constant concentrations.
- INTERACTION: cells modify their environment by uptaking or releasing food.

B.12.6 Environment renewal scheme

ENVIRONMENT_RENEWAL_SCHEME <choice>
(KEEP_MATTER/CLEAR_MATTER)

Define the renewal scheme of the environment at each new variation.

- CLEAR_MATTER: the environment is rinsed at each variation.
- KEEP_MATTER: the environment is NOT rinsed at each variation.

B.12.7 Environment variation scheme

ENVIRONMENT_VARIATION_SCHEME <choice>
(RANDOM/PERIODIC/CYCLIC)

Define the variation scheme of the environment.

- PERIODIC: variation periodically occurs with frequency INTRODUCTION_RATE
- RANDOM: variation occurs with probability INTRODUCTION_RATE
- CYCLIC: variation occurs at each time-step, but is pondered by a sinus function of period 1/INTRODUCTION_RATE

B.12.8 Environment localization scheme

ENVIRONMENT_LOCALIZATION_SCHEME <choice>
(GLOBAL/RANDOM/SPOT/CENTER)

Define the localization scheme of the environment.

- GLOBAL: the variation affects the whole environment at once (the same concentration(s) of the same new metabolite(s) is introduced everywhere).
- RANDOM: the variation affects the whole environment, but new concentrations and new metabolites are drawn for each location.
- SPOT: the variation affects only one random spot
- CENTER: the variation affects the center of the environment.

B.12.9 Environment metabolic scheme

ENVIRONMENT_VARIATION_SCHEME <choice>
(UNIQUE/MULTIPLE/BOUNDARIES)

Define the metabolic scheme of the environment.

- UNIQUE: only one metabolite is introduced at each new variation.
- MULTIPLE: multiple metabolites introduction is possible.
- BOUNDARIES: restricted multiple scheme: only boundaries of the environment species range are chosen.

B.12.10 Environment introduction rate

ENVIRONMENT_INTRODUCTION_RATE <rate> (float $\in [0, 1]$)

Define the rate at which environmental variations occur (depends on the variation scheme).

B.12.11 Environment diffusion coefficient

ENVIRONMENT_DIFFUSION_COEFFICIENT <coefficient> (float $\in [0, 1]$)

Define the diffusion coefficient in the environment grid. Diffusion is based on a simple algorithm diffusing every metabolites at the same rate in the Moore neighborhood. No ODEs are used here. For this reason, the algorithm becomes unstable for coefficient $>$ 0.1. Thus, if coefficient > 1, diffusion is infinite (well-mixed environment).

B.12.12 Environment degradation rate

ENVIRONMENT_DEGRADATION_RATE <rate> (float $\in [0, 1]$)

Define the rate at which metabolites are degraded. All metabolites degrade at the same rate. Degradation products are implicit, meaning that degraded metabolites simply disappear from the environment.

FOLIO ADMINISTRATIF

THESE DE L'UNIVERSITE DE LYON OPEREE AU SEIN DE L'INSA LYON

NOM : ROCABERT DATE de SOUTENANCE : 17/11/2017

Prénoms : Charles

TITRE : Étude de l'évolution des micro-organismes bactériens par des approches de modélisation et de simulation informatique

NATURE : Doctorat Numéro d'ordre : 2017LYSEI106

Ecole doctorale : Informatique et Mathématiques de Lyon (InfoMaths)

Spécialité : Informatique

RESUME :
Variation et sélection sont au coeur de l'évolution Darwinienne. Cependant, ces deux mécanismes dépendent de processus eux-mêmes façonnés par l'évolution. Chez les micro-organismes, qui font face à des environnements souvent variables, ces propriétés adaptatives sont particulièrement bien exploitées, comme le démontrent de nombreuses expériences en laboratoire. Chez ses organismes, l'évolution semble donc avoir optimisé sa propre capacité à évoluer, un processus que nous nommons évolution de l'évolution (EvoEvo). La notion d'évolution de l'évolution englobe de nombreux concepts théoriques, tels que la variabilité, l'évolvabilité, la robustesse ou encore la capacité de l'évolution à innover (open-endedness). Ces propriétés évolutives des micro-organismes, et plus généralement de tous les organismes vivants, sont soupçonnées d'agir à tous les niveaux d'organisation biologique, en interaction ou en conflit, avec des conséquences souvent complexes et contre-intuitives. Ainsi, comprendre l'évolution de l'évolution implique l'étude de la trajectoire évolutive de micro-organismes – réels ou virtuels – , et ce à différents niveaux d'organisation (génome, interactome, population, …). L'objectif de ce travail de thèse a été de développer et d'étudier des modèles mathématiques et numériques afin de lever le voile sur certains aspects de l'évolution de l'évolution. Ce travail multidisciplinaire, car impliquant des collaborations avec des biologistes expérimentateur•rice•s, des bio-informaticien•ne•s et des mathématicien•ne•s, s'est divisé en deux parties distinctes, mais complémentaires par leurs approches : **(i)** l'extension d'un modèle historique en génétique des populations – le modèle géométrique de Fisher – afin d'étudier l'évolution du bruit phénotypique en sélection directionnelle, et **(ii)** le développement d'un modèle d'évolution *in silico* multi-échelles permettant une étude plus approfondie de l'évolution de l'évolution. Cette thèse a été financée par le projet européen EvoEvo (FP7-ICT-610427), grâce à la commission européenne.

MOTS-CLÉS :
Évolution de l'évolution ; bruit phénotypique ; complexité phénotypique ; construction de niche ; cross-feeding stable ; diversification bactérienne ; modélisation mathématique ; modélisation multi-échelles ; modélisation individu-centrée ; évolution expérimentale *in silico*.

Laboratoire(s) de recherche : Laboratoire d'InfoRmatique en Images et Systèmes d'information (LIRIS, UMR 5205 CNRS).

Composition du jury :

Samuel Bernard	Chargé de recherche HDR, CNRS	Examinateur
Guillaume Beslon	Professeur, INSA de Lyon	Directeur de thèse
Bahram Houchmandzadeh	Directeur de recherche, CNRS	Rapporteur
Carole Knibbe	Maître de conférences, INSA de Lyon	Directrice de thèse
Jean-Baptiste Mouret	Directeur de recherche, INRIA	Président du jury
Olivier Tenaillon	Directeur de recherche, INSERM	Rapporteur
Karine Van Doninck	Professeur, University of Namur	Examinatrice