Individual-based modeling of microbial systems under consideration of consumer-resource interactions and evolution

Dissertation for the degree of Doctor of Natural Sciences (Dr. rer. nat.)

Osnabrück University

Department of Mathematics/Computer Science

Submitted by André Bogdanowski

April 2022

Abstract

Ecological systems are difficult to understand, let alone predict. The reason is their enormous complexity that arises from numerous organisms interacting with each other and their environment in a multitude of ways. However, this understanding is crucial to secure a plentitude of services that are provided by ecological systems. A substantial proportion of these services are carried out by microorganisms such as bacteria, fungi, and archaea. For example, microorganisms contribute to degradation of organic matter, water purification, and even regulation of the global climate. Therefore, a thorough understanding of the ecology of microorganisms is particularly relevant for our future well-being.

While microorganisms are comparatively well-suited for experimental studies, owing also to recent technological advances in molecular biology, it is necessary to apply theory and modeling in order to fully benefit from the empirical data. A widely used theoretical method in microbial ecology is individual-based modeling, in which population or community dynamics emerge from the behavior and interplay of individual entities that are simulated according to a predefined set of rules. However, existing individual-based models of microbial communities are often specialized on particular research questions or require proficiency in specific programming languages or software. These limitations can be hampering for a broad and systematic application of individual-based modeling in microbial ecology.

For this thesis, McComedy, a framework and software tool for the creation and running of individual-based models of microbial consumer-resource systems, was developed. It allows for fast and user-friendly model development by flexibly combining pre-implemented building blocks that represent physical, biological, and evolutionary processes. The ability of McComedy to capture the essential dynamics of microbial consumer-resource systems was demonstrated by reproducing and furthermore adding to the results of two distinct studies from the literature.

McComedy was furthermore applied to study the evolution of metabolic interactions between bacteria. More specifically, it was assessed whether cooperative exchange of costly metabolites can evolve in bacterial multicellular aggregates. The results indicate that this is in principle possible, however, it depends on the mechanism by which the metabolites are exchanged. If metabolites are exchanged via diffusion through extracellular space, cooperation is not expected to evolve. On the other hand, if metabolites are transferred by contact-dependent means, for instance via intercellular nanotubes, cooperation is likely to evolve.

Overall, contributions from this thesis comprise, first, a user-friendly modeling tool that can be used by microbial ecologists, second, insights into the evolution of metabolic interactions in bacterial systems, and, third, awareness of how the mechanistic consideration of a process can drastically affect the outcome of a modeling study.

Content

1 Introduction

1.1 Towards mechanistic understanding of eco-evolutionary dynamics

1.1.1 The individual-based perspective on ecology

In a rapidly transforming world that is witnessing climate change, mass extinction, land degradation, and pandemics, thorough understanding and prediction of ecological systems is key to preserve ecosystem services (Loreau 2010b, Evans 2012, Grimm and Berger 2016). However, ecological systems are rarely easy to understand. In contrast, they often exhibit extraordinary complexity due to numerous species interacting in various ways and across multiple scales (Loreau 2010b). To tackle this complexity, ecologists often work with models, that is, simplified representations of the complex systems (Schmitz 2008, Railsback and Grimm 2019). A traditional approach is to use empirical demographic rates of the populations at question to model their dynamics. However, results obtained by this approach are limited to the specific conditions at which the underlying demographic rates were measured (Grimm *et al.* 2016). In the understanding and prediction of unknown ecological systems or known systems under novel conditions, major advances have been made by shifting the perspective from demographic thinking towards trait-based and individual-based ecology, in which the traits and adaptive behaviors of individual organisms are regarded as the determinants of higher-level dynamics (Stillman *et al.* 2015, Grimm *et al.* 2016, Zakharova *et al.* 2019). Together with this new theoretical foundation, an exponential increase of computational capacities over the last three decades has supported the rise of complex individual-based models (IBMs) that have the power to simulate entire ecosystems at the level of individual organisms (DeAngelis and Mooij 2005, DeAngelis and Grimm 2014). This led to considerable advances in understanding and predicting complex ecological systems. Further progress is attributed to the implementation of individual traits and behaviors according to mechanistically sound and generic principles, also referred to as first principles (Grimm *et al.* 2016, Grimm and Berger 2016). For example, the metabolic theory of ecology (Brown *et al.* 2004, Schramski *et al.* 2015) provides universal understanding of how the metabolic activity of organisms varies with body size and temperature, which can be used to characterize individual organisms across a multitude of ecological systems. In addition, explicit consideration of resources and compliance with physical laws during their uptake, release, and conversion to energy and biomass are fundamental to capture the real functioning of an ecological system (Martin *et al.* 2012).

1.1.2 Integration of ecology and evolution

Another dimension that needs to be accounted for when attempting to understand ecological systems and their response to change is evolution. There is increasing recognition that ecological and evolutionary processes can act on the same time-scale and that there is reciprocal interplay,

referred to as eco-evolutionary dynamics (Pelletier *et al.* 2009, Brunner *et al.* 2019). In order to model this interplay, literature of this field suggests coupling evolutionary processes with ecological theory across scales. That means, using models of evolutionary processes at the level of genes or individuals to obtain the demographic rates for population-level ecological models (Coulson *et al.* 2006, Loreau 2010b). However, individual-based modeling would allow both to integrate evolutionary dynamics at the individual level and to observe the ecological dynamics emerging at higher levels without the intermediate step of transforming low-level process outcomes to demographic rates (DeAngelis and Mooij 2005, Romero-Mujalli *et al.* 2018).

1.1.3 The cell-level perspective on eco-evolutionary dynamics

Following the first principles approach, integrating evolution into an IBM presupposes a mechanistic understanding of evolutionary processes. This, however, can only be achieved at the cellular level, as all evolutionary products are manifestations of structural and functional modifications of cells (Lynch *et al.* 2014). Considering that a multicellular organism may consist of trillions of cells (Bianconi *et al.* 2013), modeling ecological systems on a cellular level seems highly impractical. While sacrificing some extent of mechanistic detail for the sake of feasibility may be a justified approach, another option is to address eco-evolutionary questions with systems of unicellular organisms, that is, microbial populations and communities. Therefore, using microorganisms as a model systems allows integrating evolutionary processes into models at a very mechanistic level while maintaining an individual-based perspective. An advantage of the work with microbial communities is that accompanying experiments can be conducted in the laboratory under controlled conditions and within reasonable time-scales. However, microbial communities are not only a convenient model system but also central to research questions in various research disciplines as outlined in the following.

1.2 Modeling microbial communities

1.2.1 The diversity of microbial community models

For several decades, mathematical and computational models have been used to understand and predict microbial systems that are highly relevant in health, biotechnology, and the global climate (Costello *et al.* 2012, McCarty and Ledesma-Amaro 2019, Cavicchioli *et al.* 2019). This issue has been partly addressed with ‘black box’ models, which predict microbial ecological dynamics based on population-level data, but provide no mechanistic insight into the inner functioning of the systems (Widder *et al.* 2016b). Examples of such models range from ordinary differential equation models in the context of wastewater treatment (Henze *et al.* 2015) to artificial neural networks for the prediction of community structure of soil microbes (Santos *et al.* 2014). In order to provide mechanistic insights into microbial systems, models need to incorporate well-founded principles of biological processes (Zaccaria *et al.* 2017). Such principles can be adopted from

thermodynamics (González-Cabaleiro *et al.* 2013, Gogulancea *et al.* 2019, Marsland *et al.* 2019), microbial resource allocation (Litchman *et al.* 2015, Sharma and Steuer 2019), or metabolic networks (Orth *et al.* 2011, Biggs and Papin 2013, Bauer *et al.* 2017, Popp and Centler 2020). However, if microbial systems are modeled at the level of whole populations, dynamics that emerge from spatial organization cannot be accounted for. This is overcome by a variety of spatially explicit models of microbial systems. Classic examples are partially differential equation models (Gómez-Mourelo and Ginovart 2009, Labarthe *et al.* 2019, Uppal and Vural 2020). Grid and graph-based models (Harcombe *et al.* 2014, Allen *et al.* 2013) account for spatial structure by simulating populations in individual patches that can interact with neighboring patches. As discussed before (Section 1.1.1) for ecological models in general, a straightforward way to integrate biological processes and spatial dynamics into a microbial system is by simulating each bacterial cell individually, that is, with individual-based models.

1.2.2 Individual-based models of microbial communities

Within the variety of ecological models of microbial systems, individual-based models (IBMs) are particularly important (Ferrer *et al.* 2008, Hellweger and Bucci 2009, Hellweger *et al.* 2016). Given their potential for spatially explicit modeling of complex systems, they are often applied when population-based models are deemed insufficient. Examples of such systems that are frequently studied with IBMs include bacterial biofilms (Wimpenny and Kreft 2001, Lardon *et al.* 2011a, Biggs and Papin 2013, Li *et al.* 2019, Koshy-Chenthittayil *et al.* 2021), soil systems (Ginovart *et al.* 2005, Masse *et al.* 2007, Banitz *et al.* 2015, König *et al.* 2020), and the gut microbiome (Bucci and Xavier 2014, Coyte *et al.* 2015, Shashkova *et al.* 2016, Bauer *et al.* 2017, Lin *et al.* 2018). Moreover, IBMs of microbial communities are often used to address fundamental questions in microbial ecology and evolution, for example, how spatial patterns emerge (Mabrouk *et al.* 2010, Momeni *et al.* 2013a, Momeni *et al.* 2013b, Mitri *et al.* 2015, Pande *et al.* 2016b). As proposed by Hellweger *et al.* (2016) and Zaccaria *et al.* (2017), there is an increasing number of studies where IBMs are coupled with laboratory experiments. For example, Mitri *et al.* (2015) have shown experimentally that two previously intermixed strains of *Pseudomonas aeruginosa* segregate from each other under limited resources and applied an IBM to discover that this pattern was caused by a ‘bottle-neck effect’ (this is elaborated in Chapter 3). In another example, Bauer *et al.* (2017) devised an IBM that integrates metabolic data of representative gut microbiome species and compared the predicted production of metabolites with corresponding experimentally measured data.

1.2.3 Evolution in individual-based models of microbial communities

Research questions related to evolutionary biology have been addressed with IBMs of microbial systems in different ways. The most common approach is to examine the changes in frequency of different already existing genotypes while the emergence of new mutations in the community is

neglected. For example, in several studies addressing the evolution of cooperation in microbial systems, IBMs were used to investigate whether genes encoding costly cooperative traits can persist in the presence of non-cooperative individuals lacking these genes (e.g. Momeni *et al.* 2013b, Dobay *et al.* 2014, Pande *et al.* 2015a, Stump *et al.* 2018a, Stump *et al.* 2018b, Wechsler *et al.* 2019). However, this evolutionary approach is not clearly distinguishable from approaches in ecology that model the invasion of habitats by new species (e.g. Champagnat and Meleard 2007, Vila *et al.* 2019), underlining the intertwining of ecology and evolution (Section 1.1.2).

So far, only few examples of IBMs of microbial systems have included evolutionary processes in an explicit and mechanistically sound way. (e.g. Gregory *et al.* 2004, Clark *et al.* 2011, Hellweger *et al.* 2018). However, evolutionary and eco-evolutionary IBMs are frequently highlighted as particularly useful in the microbial context (Hellweger *et al.* 2016) as well as in general (Grimm and Berger 2016, Romero-Mujalli *et al.* 2018). This points to a great unexploited potential in this research direction.

1.2.4 Multi-purpose tools for developing IBMs of microbial communities

Most IBMs of microbial communities were developed for a specific purpose and are hence mostly applicable to a narrow space of research questions and specific study sites. This is reasonable because ecological models should always address ecological questions, not only represent systems (Railsback and Grimm 2019). However, a substantial number of microbial ecologists who are not trained in developing IBMs may miss out on the opportunity to complement their research with individual-based modeling. For this reason, several multi-purpose modeling tools have been developed to address a broader range of research questions regarding microbial systems. Prominent examples are Simbiotics (Naylor *et al.* 2017), iDynoMiCS (Lardon *et al.* 2011a), NUFEB (Li *et al.* 2019), and Biocellion (Kang *et al.* 2014). Although these tools were developed for public use and have proven useful in various studies, they still may be challenging to non-experts as their operation requires knowledge of programming languages or third-party software. This highlights the demand for a tool to devise microbial IBMs in a fast and user-friendly manner. Such a tool could be particularly useful if it is FAIR (findable, accessible, interoperable, reusable), a standard that was originally established for scientific data (Wilkinson *et al.* 2016) and adapted for research software (Lamprecht *et al.* 2020).

1.3 Evolution of cooperation in microbial communities

1.3.1 Microbial cross-feeding

A central aspect in the ecology of microbial systems is the metabolic interaction between microorganisms of the same and different species (e.g. D'Souza *et al.* 2018a, Hibbing *et al.* 2010). In general, linking the flow of energy and matter (i.e. resources) to the metabolism of organisms and, thus, their population dynamics has been suggested to be key for a thorough understanding of

ecological systems (Loreau 2010a). This seems to be especially true for natural microbial communities, given that they extensively exchange resources such as amino acids, vitamins, and nucleotides. This is underlined by the high frequency of auxotrophies (i.e. the inability to synthesize an essential resource) and interdependencies (Embree *et al.* 2015, Zengler and Zaramela 2018, Gorter *et al.* 2020, Johnson *et al.* 2020, Zelezniak *et al.* 2015). If a microorganism produces and releases a resource that can be consumed by another microorganism of the same community, the process is called *cross-feeding* (D'Souza *et al.* 2018a). D'Souza *et al.* (2018a) provided a scheme to classify different forms of cross-feeding. According to this scheme, cross-feeding can be classified as either *unidirectional* (which means one microorganism provides resources for another but not *vice versa*) or *bidirectional* (which means both microorganisms provide each other different resources). Another dimension to classify cross-feeding is the degree of investment: if an exchanged resource is a by-product of a microorganism's usual metabolism, the cross-feeding is referred to as *by-product cross-feeding*. On the other hand, if the resource is produced 'purposefully', that is, if the microorganisms provides the resource to another microorganism at the cost of its own fitness (which must have been previously favored by selection), this is referred to as *cooperative cross-feeding*.

1.3.2 The tragedy of the commons

If cooperative cross-feeding is essential for the functioning of a system, the possible emergence of non-cooperative competitors can be detrimental for that system as illustrated by the so called *tragedy of the commons* (Hardin 1968). In the context of evolutionary biology, *tragedy of the commons* refers to a situation in which competing individuals selfishly maximize their individual fitness to an extent that the overall fitness at the system-level decreases (Rankin *et al.* 2007). In the case of cooperative cross-feeding this could occur in the following way: Initially, two strains of cooperative cross-feeders reciprocally exchange costly metabolites. Mutations in members of these strains result in down-regulation of the costly overproduction of these metabolites. The saved costs for overproduction results in increased fitness of these members, which consequently outcompete the producers. However, in the long term, the total exchange of metabolites in that system decreases to a level at which the overall fitness of the individuals is significantly reduced.

The *tragedy of the commons* as well as other closely related concepts such as the *public goods dilemma*, *prisoner's dilemma*, and *free rider problem* originate from game theory in economics and political science (Olson 1965, Luce and Raiffa 1957). They can all be traced back to the circumstance that decisions (e.g. whether to cooperate or not) are made by individuals although the preferable outcome may be perceived differently from the individual and system-level perspective. These problems are readily transferable to evolutionary questions as natural selection acts on the genes of individual organisms (Williams 1966, Dawkins 1976, Rankin *et al.* 2007). In this comparison, 'decision' can be interpreted as carrying a gene that encodes a certain trait (e.g.

cooperation). Natural selection rewards or punishes the carrier of this gene according to its reproductive success. Thus, traits that maximize individual fitness are selected for, regardless of the consequences at the system level.

1.3.3 Mechanisms that support the evolution of cooperation

The question of how cooperation (e.g. in terms of cooperative cross-feeding) can evolve without causing a *tragedy of the commons* (Section 1.3.2) is a major problem in evolutionary biology (Axelrod and Hamilton 1981, Sachs *et al.* 2004, Frank 2010). Hamilton (1964b) has provided a fundamental extension to Darwin's theory, which has been later termed *kin selection* and *inclusive fitness*. The idea is that a cooperative act towards close relatives may contribute to the prevalence of the responsible gene as close relatives are likely to carry that same cooperative gene, as well. However, this explanation does not account for cooperative interactions between taxonomically distant or unrelated individuals, which are frequent in microbial systems.

Another mechanism that supports the evolution of cooperation is called multi-level selection or group-level selection (Wilson 1987, Kingma *et al.* 2014, Gardner 2015, Doulcier *et al.* 2020). This concept is based on the idea that natural selection acts not only at the level of individuals but also at other levels of biological organization. Thus, costly cooperative traits may be disfavored at the level of individuals but selected for at higher organizational levels, such as groups or colonies (Kramer and Meunier 2016). Assuming organization of individuals in groups, the frequency of cooperators is expected to decrease in each group, but groups that contain, by chance, a large proportion of cooperators may outcompete other groups. This way, cooperative traits can evolve despite their individual-level disadvantage. However, due to different conceptions of appropriate wording and partial overlap with *kin selection* and *inclusive fitness*, the validity of this mechanism is strongly debated (Leigh 2010).

There are several other mechanisms that support the evolution of cooperation, of which several are characterized by reciprocity (e.g. bidirectional cross-feeding). For example, if cooperative organisms have the ability to recognize other cooperative partners, they can block non-cooperative competitors from the exchanged benefits (Bull and Rice 1991, Sachs *et al.* 2004, Kaltenpoth *et al.* 2014). Such *partner selection* is not necessarily required if partners are interacting repeatedly over evolutionary time and, thus, positive feedback loops between cooperating partners reward such interactions, which is referred to as *partner fidelity feedback* (Bull and Rice 1991, Sachs *et al.* 2004, Fletcher and Doebeli 2009). This can be, for example, due to spatial proximity, provided the organisms remain sufficiently long at the same positions (Yamamura *et al.* 2004, Kümmerli *et al.* 2009, Hol *et al.* 2013, Momeni *et al.* 2013b, Dobay *et al.* 2014, Pande *et al.* 2016b, Germerodt *et al.* 2016, Stump *et al.* 2018a, Stump *et al.* 2018b). For this mechanism, however, a key assumption is that the cooperative interaction is limited in distance, that is, only close neighbors are affected by the cooperative act (Allison 2005, Allen *et al.* 2013, Borenstein *et al.* 2013). Thus, large

interaction distances (for example due to high diffusivity of the shared good) can hinder the selection for cooperation. Furthermore, if interactions between cooperators and non-cooperators results in specific fitness pay-offs, spatial structure can actually inhibit cooperation, as illustrated by the *spatial snowdrift game* (Hauert and Doebeli 2004).

The role of spatial structure in some of the described mechanisms leads to the question which spatial environments facilitate the evolution of cooperation. So far, it was demonstrated that, for example, attached biofilms and colonies can promote the cooperative exchange of metabolites (Kreft 2004, Drescher *et al.* 2014, Nadell *et al.* 2016, Pande *et al.* 2015a). In liquid environments, microorganisms can be spatially structured by forming multicellular aggregates (Trunk *et al.* 2018, Cai 2020). In such multicellular aggregates of initially non-cooperative strains of *Escherichia coli*, cooperation was observed to evolve within less than 150 generations (Preussger *et al.* 2020). However, it remains elusive if and how these aggregates contributed to the evolution of cooperation. To answer this question, the metabolic interactions of the microorganisms in the aggregates, their emergence, and consequences must be understood in a mechanistic way.

1.4 Research objectives

As outlined above (Sections 1.1 – 1.3), individual-based modeling of eco-evolutionary dynamics, particularly in microbial communities, is a promising research direction with a substantial foundation of preliminary work. However, multiple gaps in methodology and theory have been identified.

1 Individual-based models of microbial systems that integrate evolutionary processes are considered to be promising. However, to this day they have not been widely applied.

2 A generic and user-friendly framework for the development of eco-evolutionary IBMs of microbial communities is lacking. Existing frameworks are either specific to certain applications or not accessible to a large part of the researchers' community due to required proficiency in programming or software tools. The FAIR standard for research software has not been widely applied to IBMs of microbial systems.

3 The evolution of cooperation is not understood for systems in which microorganisms form multicellular aggregates. There is empirical evidence that such systems can facilitate the evolution of cooperation. However, a clear and mechanistic understanding of the underlying processes is usually lacking.

The overarching goal of this thesis is to address these identified gaps. To do so, the first main objective is to devise a framework for the development and application of eco-evolutionary IBMs for the analysis of the functioning of microbial systems by explicitly accounting for consumer-resource interactions. The idea is to facilitate versatile and systematic modeling by implementing

the framework in a modular design with a library of generic submodels of biological, physical, and evolutionary processes as building blocks that can be flexibly combined to IBMs according to the research question. The framework shall also allow for thorough simulation analyses. Expertise from the fields of evolution and microbial ecology has to flow into the development of the framework.

The second main objective is to make the framework available to a broad community of researchers by implementing it as an open-access stand-alone software with an intuitive and user-friendly graphical interface. No programming skills or proficiency with specific third-party software must be required to use the software. The framework is supposed to satisfy the FAIR standard. To ensure quality and accuracy of both framework and software, previously published studies of microbial systems will be reproduced.

The third main objective of this thesis is to apply the framework to a current research question in the field of microbial ecology and evolution that is linked to consumer-resource-interactions. The framework will be used to examine whether and how cooperative cross-feeding can evolve in microbial multicellular aggregates. Utilizing the required features of the framework, microbial multicellular aggregates will be simulated in a spatially explicit (in three dimensions) and mechanistically sound model, allowing the effects of cell-level processes to propagate across scales of biological organization. In particular, assumptions how metabolites are exchanged and how these assumptions impact the evolution of cooperators and their frequency will be tested in a systematic analysis.

Each of the main objectives is addressed in a chapter of this thesis. The workflow is illustrated in Figure 1.

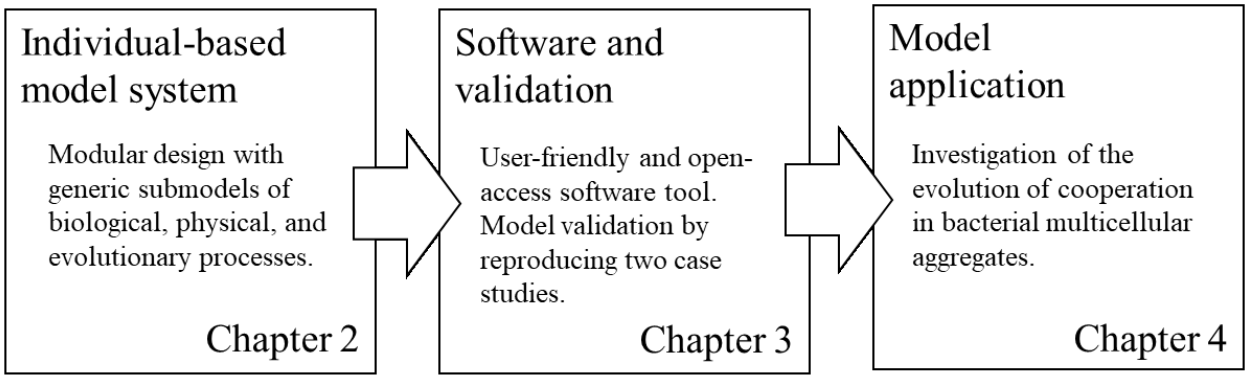

Figure 1. Scheme of objectives and chapter overview.

2 McComedy: Individual-based model framework[1]

2.1 Introduction

Ecological systems are notoriously complex, comprising networks of numerous organisms that interact in various ways with each other and their environment (Green *et al.* 2005, Loreau 2010b). However, understanding and predicting the behavior of ecological systems and particularly their response to changing environmental conditions is key to preserve various ecosystem services (Evans 2012). A substantial amount of such ecosystem services is provided by microorganisms, which play a significant role in nutrient cycling, climate regulation, water purification, and primary production (Ducklow 2008). Their essential contributions to our well-being but also recent advances in molecular technologies motivate intense research of microbial ecology (Prosser *et al.* 2007). However, in order to rigorously exploit the data collected in experimental approaches and to tackle questions that cannot be addressed solely empirically, mathematical and computational modeling is necessary (Gunawardena 2014, Hellweger *et al.* 2016). Early mathematical models of microorganisms focused on population-level growth of genetically homogeneous cultures (Monod 1949) and, to this day, equation-based population-level models are often the appropriate tool in microbial ecology (e.g. Marsland *et al.* 2020, Estrela *et al.* 2021, Pacciani-Mori *et al.* 2020). On the other hand, some ecological processes can only be recapitulated if local interactions between individual microbial cells, a spatially structured environment, and mass balance of resources are explicitly considered (e.g. Momeni *et al.* 2013b, Mitri *et al.* 2015, Pande *et al.* 2016b, Bauer *et al.* 2017). This is in line with the view of (Loreau 2010a) who underpinned that the functioning of ecosystems can only be fully understood if organismal aspects are adequately linked to matter and energy fluxes and frameworks are needed which allow analyses of the propagation of effects across organizational levels: from individuals to the entire ecosystem. All these challenges are taken up and addressed with individual-based models (IBMs) (Hellweger *et al.* 2016).

In general, IBMs are used to simulate the dynamics of populations and communities at the level of individual organisms, which have specific traits and behaviors, interact with each other, and respond to their environment (DeAngelis and Mooij 2005). From these traits and interactions, higher-level dynamics emerge. Traditional IBMs describe population or community dynamics in the language of individual-level demographic process rates (establishment, growth, fecundity, mortality) or behavioral rules (social status, dispersal behavior). This demographic approach, however, comes to the cost that the resulting model is specific and only applicable to the site at

[1] Based on Bogdanowski A., Banitz T., Muhsal L. K., Kost C. and Frank K. 2022. Mccomedy: A user-friendly tool for next-generation individual-based modeling of microbial consumer-resource systems. PLoS Comput Biol 18: e1009777.; Supporting Information; see also: https://git.ufz.de/bogdanow/mccomedy

which the demographic rates were empirically measured. The model is not transferable to other site-conditions that limits its applicability (Grimm *et al.* 2016). This drawback can be overcome with so-called next-generation IBMs, in which individual-level process rates are related to first principles, i.e. well-understood fundamental functional relationships rooted in theories from physics, chemistry, physiology, and evolution (Grimm and Berger 2016, Grimm *et al.* 2016). Such next-generation IBMs can be applied to scenarios with altered environmental conditions through appropriate parameterization as long as all processes are based on these fundamental principles.

Microbial systems are particularly well suited for analysis with next-generation IBMs as microbial behavior is much closer bound to first principles than that of, for example, social animals. Thus, there are highly useful examples of microbial IBMs that are based on first principles (e.g. Lardon *et al.* 2011a, Li *et al.* 2019, Naylor *et al.* 2017, Sharma and Steuer 2019, Kang *et al.* 2014, Bauer *et al.* 2017). In part, these models run on different platforms and their implementations are based on different designs. A systematic, cross-discipline analysis could therefore be hindered by such inconsistencies. One solution to overcome this problem was suggested by Grimm and Berger (2016): public libraries of generic, mechanistically sound, and tested submodels that can be combined to next-generation IBMs of various ecological systems and scenarios.

This thesis is based on the idea to adjust this strategy and to develop a modeling framework for advanced investigations of the functioning of microbial systems by explicitly accounting for consumer-resource interactions and their evolution. This chapter presents *McComedy* (**Mic**robial **Co**mmunities, **Me**tabolism, and **Dy**namics), a framework for flexibly designing next-generation individual-based models of microbial systems through combining pre-implemented, generic, but mechanistically sound submodels (referred to as process modules) according to the needs of the research question to be answered. In the following, the framework *McComedy* is elaborated with a particular focus on the process modules and their implementation.

2.2 Standardized model description

The description follows the ODD (Overview, Design concepts, Details) protocol for describing individual- and agent-based models (Grimm *et al.* 2006, Grimm *et al.* 2020) in a standardized way. Originally ODD protocols are intended to describe specific models whereas *McComedy* is a generic modeling framework. Hence, where the ODD protocol requires particular information with regard to single models, possibilities in different implementations of specific models using the *McComedy* platform are described.

2.2.1 Purpose and patterns

The purpose of *McComedy* is to provide a framework for flexibly designing spatially explicit, individual-based models (IBMs) to investigate microbial systems by accounting for consumer-resource interactions and their evolution. The framework allows for simple and fast model creation

by selecting a set of process modules to be combined in one specific model. The IBMs created with *McComedy* can serve different purposes, however, they have in common that they simulate the spatiotemporal dynamics of microbial systems driven by metabolic processes such as the uptake, use, conversion, and synthesis of resources. Therefore, they allow for mechanistic analyses and understanding of these dynamics. In this manner, specific IBMs may be used to recapitulate and explain empirical findings, to study parameter setups that cannot (easily) be controlled for in experiments, and to generate new hypotheses on the mechanisms underlying microbial dynamics.

The accuracy of the models can be assessed by comparison of patterns emerging in the model and corresponding experiments, provided that such data exists. Such patterns can be, for example, growth dynamics over time, relative abundances of different types of microbes, and spatial organization as quantifiable by, for example, intermixing and cluster analysis.

2.2.2 Entities, state variables, and scales

The entities in *McComedy* are *microbes* and the *environment*. *Microbes* are the individuals that constitute the investigated systems. They can, for example, represent bacteria, fungi, or protists. These microbes have a spherical shape. Each individual is characterized by several state variables such as its genotype, biomass or spatial position (see Table 1 for all state variables). The individuals' size is determined by their biomass and a density parameter. Certain microbe state variables are included only when the corresponding process module is used (Table 1).

The *environment* is a three-dimensional array of discrete grid cells. It contains the resources as well as the individual *microbes*. The individual *microbes* have exact continuous spatial positions, meaning that they can overlap with more than one grid cell (Table 1). Different types of resources represent different metabolites which are consumed or released by the microbes. Resource concentrations are assigned to discrete spatial grid cells. Hence, the state variables characterizing the environmental grid cells are the local resource concentrations of the different types of resources (Table 1). The explicit consideration of *microbes* and their *environment* is the pre-requisite for operationalizing the mentioned consumer-resource interactions and their evolution.

The models created with *McComedy* act on the scale of individual microbes, simulating up to several thousand individuals. *McComedy* works with generic units for time (T), distance (S), resource mass (M) and microbial dry mass (M*). The modeler is required to decide in which specific units the generic ones translate and parametrize the model accordingly. For example, assuming that T corresponds to 1 s and S corresponds to either 1 µm or 5 µm, a *diffusion constant* of 25 $\mu m^2/s$ needs to be defined as either 25 S^2/T or 1 S^2/T, respectively. For best computational performance, S should be in the same order of magnitude as a typical microbe's diameter. For instance, when modeling the bacteria *Escherichia coli*, S could be considered 1 µm. The extent of the environment and the number of microbes are only restricted by the computational time needed

to simulate all processes. We suggest starting with a spatial extent of 125 x 125 x 1 S^3 (2D) or 25 x 25 x 25 S^3 (3D) and a maximum microbe number of 2000 (these number can be increased when simulating on high-performance computing systems).

Table 1. State variables in McComedy. The symbols are used for the state variables in formulas in this ODD protocol, but not in the source code of McComedy. The column "Process module" indicates if a state variable is only included when the specified process module is used.

Name	Symbol	Variable type	Explanation	Process module
ID	-	Integer	A unique ID which allows tracking an individual microbe over simulation time.	-
Genotype	-	String	Defines the type of a microbe	-
Biomass	B	Decimal	A microbes dry weight	-
X-position	X	Decimal	X-position of a microbe's center	-
Y-position	Y	Decimal	Y-position of a microbe's center	-
Z-position	Z	Decimal	Z-position of a microbe's center	-
Local concentrations	C_R	3D-Array of Decimals	The concentration of a resource R in each grid cell of the discretized simulated space	-
Attached microbes	-	List of Microbes	Lists all microbes that are currently attached to a microbe	Attachment (Section 2.2.7.2.1)
Substrate pool	S_R	Decimal	Intracellular pool of resources in a microbe that have been consumed but not utilized yet	SubstrateUtilization (Section 2.2.7.2.16)
Starving	-	Boolean	Indicates whether a microbe is starving	SubstrateUtilization (Section 2.2.7.2.16)
Growth resources	G	Decimal	Resources allocated to biomass growth in a microbe	Growth (Section 2.2.7.2.7)
Product pool	P_R	Decimal	Intracellular pool of resource R that have been produced by a microbe	ConstantProduction (Section 2.2.7.2.4)
Connected microbes	-	List of microbes	A list containing all microbes that are currently connected to the focal microbe	NanoTubeExchange (Section 2.2.7.2.10)

2.2.3 Process overview and scheduling

Processes are the building blocks of every *McComedy* model. The processes are encapsulated into mostly independent *process modules*, which can be combined in several ways, resulting in a high flexibility in model design. Restrictions are that some process modules require others to be included as well and the process module InitModel (Section 2.2.7.1.3) is obligatory for initialization (Section 2.2.7).

There are five classes of process modules: *Initial processes*, *Microbial processes, Resource processes*, *Global processes*, and *Postprocessing modules*. Upon starting a model simulation, the *Initial processes* are executed once to initialize the model (Section 2.2.7.1). For example, the

process module InitCluster (Section 2.2.7.1.2) positions all microbes in the center of the environment to simulate a culture that starts growing from a single aggregate. All other process modules are executed repeatedly throughout the simulation. However, they are not necessarily always executed at once as different process modules may use different time step lengths (Figure 2). *Microbial processes* are executed for each microbe (Section 2.2.7.2). *Resource processes* are executed for each spatial grid cell and each resource, respectively (Section 2.2.7.3). *Global processes* are used when access to a single microbe or resource grid cell is not sufficient to simulate the process (Section 2.2.7.4). For example, Diffusion (Section 2.2.7.4.1) is implemented as a *Global process* instead of as a *Resource process* because the algorithm requires access to the entire array of grid cells. The latter three types of process modules are assumed to run simultaneously and are thus updated accordingly (Figure 2). *Postprocessing modules* (Section 2.2.7.5) are executed at the end of a time step after all other process modules (that were to be executed in that time step). They correct for undesired effects, such as the spatial overlap of microbial individuals (e.g. Shoving, Section 2.2.7.5.3).

Each process module (except for *Initial processes*) is defined with an individual time step dt. Thus, different processes can be modeled with different temporal resolution. They are only executed if their specific time step *dt* is an integer divisor of the current time *t* (Figure 2).

Process modules have read-only access to the current values of all state variables. The change ∂_p that a process needs to apply to a state variable V is stored in a temporary variable $\tilde{V} = \partial_p(V)$. Note that each process module defines its own $\tilde{V}$. After a set of simultaneously running processes is completely executed, all state variables V are updated according to

$$V(\tau + 1) = V(\tau) + \sum \tilde{V},$$

where τ indicates the state of the modeled system before the update and $\tau + 1$ after the update. The sum is taken over all $\tilde{V}$ from different process modules. The decoupling of computation and updating is necessary to ensure synchronous updates of state variables when processes are assumed to run simultaneously.

Throughout the simulation, the time variable is iteratively incremented by the smallest possible time step 1 mT. Figure 2 illustrates the procedure following the increment of the time variable in an exemplary model. The simulation stops if either the time variable or the number of microbes exceeds the predefined respective maximum value and optionally if the number of microbes becomes zero.

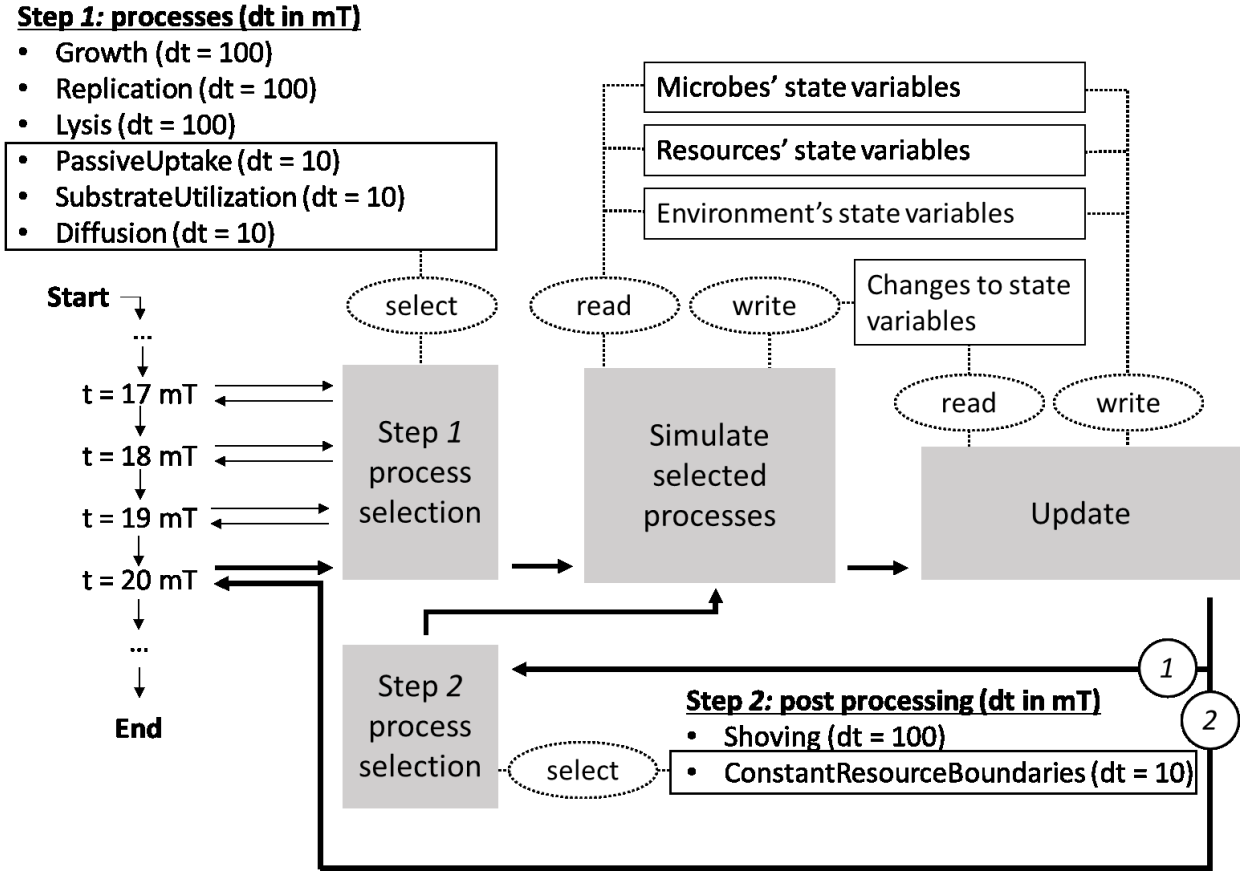

Figure 2. Process scheduling in McComedy. The simulation is organized in an iterative workflow. After incrementing the time variable process modules are checked whether they are ready for execution, i.e. if their specific time step dt *is an integer divisor of the current time. If no process modules are ready for execution the iteration is over and the time variable is incremented again. If any process modules are ready for execution they can read the entities' state variables and execute their algorithms. Resulting changes to the state variables are written into temporary variables. After that, a synchronous update is applied by adding the values of the temporary variables to the state variables of the entities. Then (following arrow number 1) it is checked if any Postprocessing modules are ready for execution. If so, the selected Postprocessing modules are executed in the same manner as the process modules before. After updating the changes made by the Postprocessing modules, or if no Postprocessing modules needed to be executed, (following arrow number 2) the iteration is over and the time variable is incremented again.*

Due to synchronized access to state variables by several entities and processes, it can happen that a state variable is changed beyond physically meaningful values (e.g. a negative resource concentration). If this happens the model is reset to the previous state and the simulation is continued with tenfold shorter time steps (ceiled to mT) of all process modules. If all process modules are run with the minimal time step 1 mT and the problem still occurs, the simulation is stopped with an error message.

2.2.4 Design concepts

2.2.4.1 Basic principles

IBMs created with *McComedy* represent the interaction of microbes and resources under consideration of space and physical and biological processes. These processes mostly account for consumer-resource dynamics (e.g. release, diffusion, and uptake of resources) and evolutionary mechanisms (e.g. mutations).

The underlying idea is that IBMs can be designed by combining distinct pre-implemented processes. This is possible because the pre-implemented processes are independent of assumptions that are specific to certain systems and therefore generic. Different processes are assumed to act on different temporal scales and can therefore be modeled with different time step lengths. This is possible due to the strict modularization of the processes (Section 2.2.3).

2.2.4.2 Emergence

Emerging patterns from IBMs created with *McComedy* depend on the specific design. Usually, the abundance of microbes of different types over time as well as spatial patterns constitute key results of the simulations. They emerge on the basis of local interactions and metabolic parameters rather than being imposed by superordinate rules. Assumed initial patterns such as microbes being located in a cluster or biofilm can be imposed by the initial positioning.

2.2.4.3 Adaptation

The microbes are assumed to make no active decisions regarding their behavior. For instance, microbes always consume as much resources as possible in order to increase biomass. However, in evolutionary terms, adaptation plays a critical role in the model. As mutations of certain traits can be explicitly modeled, the best adapted individuals will benefit from their advantages in terms of survival or reproduction and eventually dominate the system.

2.2.4.4 Objectives

As there is no direct adaptive behavior, objectives for decision-making are not required.

2.2.4.5 Learning

No learning mechanisms are implemented.

2.2.4.6 Prediction

As there is no direct adaptive behavior, predictions for decision-making are not required.

2.2.4.7 Sensing

If resource uptake is implemented in a model, microbes are assumed to sense local concentrations of the resources which they can consume.

2.2.4.8 Interaction

If uptake and release of resources are modeled, direct interaction takes place between microbes and the environment as the microbes consume or release resources. Moreover, microbes directly compete with each other for these resources and for space. The consumption and release of resources can mediate cooperative interaction between microbes of different types. There is direct metabolic interaction between microbes if nanotubes are included in the model. Furthermore, microbes can directly interact by physically attaching to each other.

2.2.4.9 Stochasticity

Stochasticity is used in the model initialization. Microbes are placed randomly in the simulated environment whereby the X, Y, and Z positions are drawn from a uniform distribution. Further stochasticity is used in the process modules InitCluster (Section 2.2.7.1.2), ChangeGenotype (Section 2.2.7.2.3), Flow (Section 2.2.7.2.6), NanoTubeExchange (Section 2.2.7.2.10), Parameter-Mutator (Section 2.2.7.2.11), Replication (Section 2.2.7.2.14), and LongTermExperiment (Section 2.2.7.4.3).

2.2.4.10 Collectives

For some model designs, simulations will result in the emergence of collectives in the form of one or several spatial aggregates of microbes.

2.2.4.11 Observation

For each simulation run, the model creates two result files: one for the state of each microbe and one for the concentration profiles of every resource. All state variables that are used in the selected process modules are written into the result files. These are updated in predefined time intervals (which can vary between the two files). The result data is compiled as tables in long format and saved as .txt files.

2.2.5 Initialization

Upon simulation start, five parameter files which define the model by listing the (1) process modules, (2) resource parameters, (3) microbe parameters, (4) general parameters, and (5) settings are read in and the environment is instantiated according to the specified spatial extent (Section 2.2.2). The parameter files are .txt files that contain tables with parameter names and values in separate columns (separated by tab stops). It is strongly recommended to create the parameter files using the graphical user interface of *McComedy* (Figure 3). Further initialization is performed by the *InitialProcess* modules (Section 2.2.7.1).

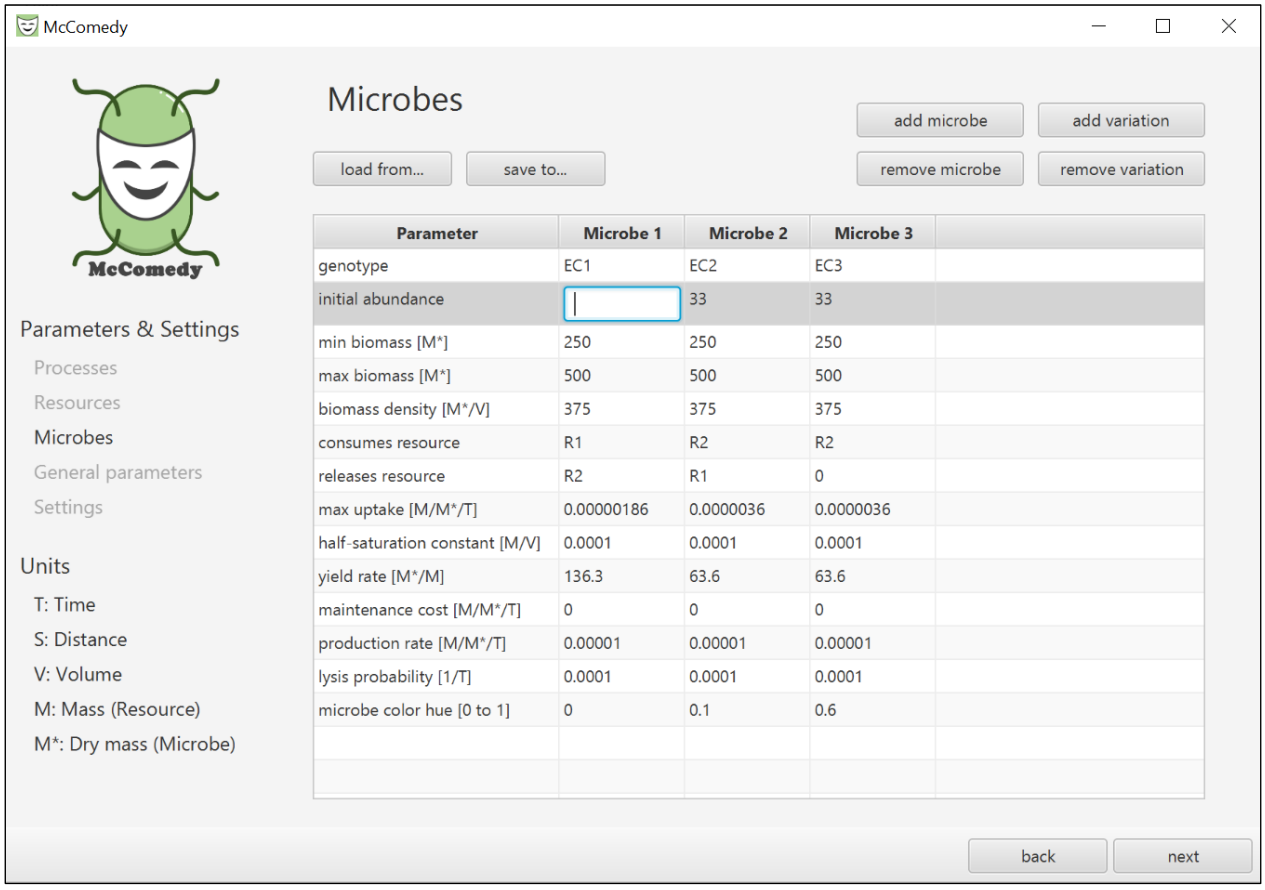

Figure 3. Screenshot of the graphical user interface of McComedy. According to the selected processes, necessary parameters are listed. For each type of microbe, the user can edit the parameter values.

2.2.6 Input data

The model does not use input data to represent time-varying processes.

2.2.7 Submodels

Since *McComedy* is a generic modeling framework, the following process modules (i.e. the submodels) to be combined in one specific simulation model can be flexibly selected. They are grouped into five classes: *Initial processes*, *Microbial processes*, *Resource processes*, *Global processes*, and *Postprocessing modules*. Within these classes, the process modules are ordered alphabetically for a convenient lookup. However, this order does not imply a corresponding order in execution. All process modules within a group are executed synchronously. As process modules do not modify state variables directly, changes are written into temporary variables which are indicated with '~' (Section 2.2.3 describes the updating scheme for synchronous processes).While the goal was to make the process modules as independent as possible, some process modules require other process modules to function properly. For example, the release of resources (Section 2.2.7.2.12) presupposes their prior production (Section 2.2.7.2.4). Such dependencies are listed in Table 25.

2.2.7.1 Initial processes

2.2.7.1.1 InitBiofilm

Biofilms are the predominant microbial life form (Costerton *et al.* 1995). This process module can be used to initialize the microbes such that they resemble a simple biofilm. The Y-position of all microbes is set to a value defined by the global parameter *biofilm Y-position*, resulting in an initial spatial distribution on a plane orthogonal to the Y-axis. When simulating biofilms, it is recommended to use this process module together with ImpermeableMicrobeBoundaries (Section 2.2.7.2.8) with the parameter values *impermeable Y-boundaries* true and *impermeable boundary offset* equal to *biofilm Y-position*.

Table 2. Parameters of process module InitBiofilm.

Parameter	Dimension	Type	Variable type	Explanation
biofilm Y-position	S	global parameter	Decimal	Position of the bottom of the biofilm on the Y-axis

2.2.7.1.2 InitCluster

Bacteria are often observed to aggregate to three-dimensional multicellular clusters, as opposed to free floating planktonic cells (Trunk *et al.* 2018, Cai 2020). If the microbes need to be initialized in an aggregated state, this process module can be used. All microbes will be positioned close to the center of the simulated environment, within the radius

$$r_{init} = r_{m0}\sqrt[3]{N}$$

, where r_{m0} is the radius of the first microbe in the list of all microbes and N is the initial number of microbes. In combination with the process module Shoving (Section 2.2.7.5.3), this results in a three-dimensional ball-shaped structure.

2.2.7.1.3 InitModel

This process module is necessary in every IBM in *McComedy*. Therefore, the user is not supposed to add it to the model manually. Instead, it is integrated automatically. According to the values of the necessary parameters (Table 3), the microbes and resources are added to the environment. The positions of the microbes are drawn randomly from a uniform distribution in the range of the entire environment.

Table 3. Parameters of process module InitModel.

Parameter	Dimension	Type	Variable type	Explanation
spatial extent X	S	global parameter	Integer	Size of environment in X-direction
spatial extent Y	S	global parameter	Integer	Size of environment in Y-direction
spatial extent Z	S	global parameter	Integer	Size of environment in Z-direction
simulation time	T	global parameter	Decimal	Maximum time before simulation stops
stop when all microbes die	-	global parameter	Boolean	Indicates whether simulation should be stopped when no microbes are contained in the environment
max microbes number	-	global parameter	Integer	If the number of microbes exceeds this value the simulation is stopped
constant initial position	-	global parameter	Boolean	Defines whether the initial microbe positions should be the same across replicates and variations
random generator seed	-	global parameter	Integer	Seed for random generator. If it is 0 a random seed is generated
genotype	-	microbe parameter	String	The name of a microbial type (e.g. strain or species). It serves as an identifier
min biomass	M*	microbe parameter	Decimal	The minimum dry mass of a microbe
max biomass	M*	microbe parameter	Decimal	The maximum dry mass of one microbe
initial abundance	-	microbe parameter	Integer	Initial number of microbes of respective type
biomass density	M^*/S^3	microbe parameter	Decimal	Biomass density of microbe as dry mass per volume
microbe color hue	-	microbe parameter	Decimal	Between 0 and 1. Color hue of respective type in visual model output as defined by HSB color representation
resource name	-	resource parameter	String	The name of this resource. It serves as an identifier
initial concentration	M/ S^3	resource parameter	Decimal	The initial concentration in each grid cell
resource color hue	-	resource parameter	Decimal	Between 0 and 1. Color hue of resource in visual model output as defined by HSB color representation
max render concentration	-	resource parameter	Decimal	Maximum concentration to which resource concentrations are scaled in visual model output. Concentrations equal to or above this maximum are rendered at full brightness

2.2.7.2 Microbial processes

2.2.7.2.1 Attachment

Attachment of microorganisms in order to form biofilms or multicellular aggregates by secreting an adhesive extracellular matrix is common in nature (Cai 2020, Costerton *et al.* 1995, Trunk *et al.* 2018). We developed an algorithm to mimic such attachment between microbial cells. Firstly, before two microbial cells can attach, it is checked whether their types are compatible for

attachment (this is defined by the microbial parameter *attach to genotype*). It is necessary to allow this distinction because some microbes only attach to the same type (autoaggregation) while other can attach to different strains and species (co-aggregation) (Trunk *et al.* 2018). Next, each microbe checks if other microbes of a compatible type are located within the *attachment distance*. This distance represents the thickness of the adhesive extracellular matrix. If compatible microbes are found within this distance, they get attached to the focal microbe and are added to its list *Attached microbes*. The algorithm pulls attached microbes towards each other. This is not fully justified by first principles but considered a reasonable simplification and ensures that attached microbes stay together even if they move. Thus, the position of every microbe in the environment is changed according to

$$\tilde{X} = d_x \, c_a \, dt,$$

$$\tilde{Y} = d_y \, c_a \, dt,$$

$$\tilde{Z} = d_z \, c_a \, dt$$

, where $\tilde{X}$, $\tilde{Y}$, and $\tilde{Z}$ are the temporary variables for the changes made by this process module with respect to the spatial axes X, Y, and Z. d_x, d_y, and d_z are the X, Y, and Z components of the mean directional vector from the focal microbe to all attached microbes and c_a is the microbe parameter *attachment coefficient*, which represents the firmness and adhesiveness of the extracellular matrix. If the *attachment coefficient* is low (i.e. the extracellular matrix loose or little adhesive), attached microbes are easier to separate from each other. If attached microbes are separated further away than the *detachment distance* they become detached from the focal microbe and are removed from its list *Attached microbes*. The detachment distance represents how far the adhesive extracellular matrix can be stretched before microbes detach from each other.

Table 4. Parameters of process module Attachment.

Parameter	Dimension	Type	Variable type	Explanation
attach to genotype	-	microbe parameter	String	The type of microbes that can attach to this type. Multiple types can be defined, separated by ';'. The string 'all' indicates that all types attach to this type
attachment distance	S	microbe parameter	Decimal	Defines how close Microbes need to be in order to attach. The distance is measured between the microbes' surfaces
detachment distance	S	microbe parameter	Decimal	Defines how far attached Microbes need to be forced away from each other in order to detach
attachment coefficient	1/T	microbe parameter	Decimal	Between 0 and 1. Defines how much a microbe is pulled towards its attached neighbors per time unit T, whereby 1 means the microbe is pulled completely towards the attached microbes and 0 means it is not pulled at all

2.2.7.2.2 CellPartition

This process module does not represent a biological or physical process but is required by some other process modules for convenience (Table 25). It estimates in a computationally efficient way how much a microbe (with its spherical shape) overlaps with the spatial grid cells of the environment. To this end, the grid cell that overlaps with the microbe's midpoint and all 26 neighboring grid cells are considered. First, for each of these 27 grid cells, a grid cell is considered to overlap with the microbe if the microbes' radius is larger than the shortest distance between its midpoint and the grid cell. Second, for all grid cells that overlap with the microbe, the reciprocal values of the squared distances between the grid cell's midpoint and the microbe's midpoint are computed and these values are normalized such that their sum equals 1. Thus, each value estimates the relative overlap of the microbe with the respective grid cell.

2.2.7.2.3 ChangeGenotype

Although this process module is called ChangeGenotype, it can be used to model all kinds of changes as long as they occur stochastically. When the type of the microbe is changed, all parameter values that are associated with the new type (as defined in the microbes parameter file) are changed, accordingly. The probability of changing the genotype is defined by the microbe parameter *change genotype probability*. The genotype that the microbe is changed to is defined by the microbe parameter *change genotype to*. Depending on how this process module is used, it can simulate a specific process in a mechanistically accurate manner (e.g. by representing a specific mutation) or implement an abstracted model assumption (e.g. by simulating random transitions between life stages).

Table 5. Parameters of process module ChangeGenotype.

Parameter	Dimension	Type	Variable type	Explanation
change genotype to	-	microbe parameter	String	The type of microbe this microbe is changed to. The string must equal a microbe's genotype parameter
change genotype probability	1/T	microbe parameter	Decimal	The probability at which a microbe's type is changed

2.2.7.2.4 ConstantProduction

Microbes are known to frequently exchange resources, either 'purposefully' as the result of an evolutionary process that resulted in cooperative behavior or by releasing by-products of the organism's metabolism (D'Souza *et al.* 2018a). In either way, the shared resource must be synthesized or degraded in a metabolic process. Using Flux Balance Analysis (Orth *et al.* 2010), we observed that the rate of synthesis of a metabolite is constant (only depending on the producers biomass) if all components required for this synthesis are available *ad libitum*. In many applications

of *McComedy*, this can be a reasonable assumption as usually only few resources are modeled explicitly and all others assumed to be sufficiently available. In such case, this process module simulates production of resource R in the intracellular product pool P_R according to

$$\tilde{P}_R = B\, p_R\, dt$$

, where $\tilde{P}_R$ is the temporary variable for the changes made by this process module with respect to the intracellular product pool P_R, B is the biomass of the microbe and p_R is the microbe's *production rate* of resource R. This process module only accounts for the synthesis of exchanged resources. Actual exchange of produced resources can be modeled with the process modules PassiveRelease (Section 2.2.7.2.12) and NanoTubeExchange (Section 2.2.7.2.10). The parameter value of *production rate* can be manipulated by the process module CooperativityFitnessCost (Section 2.2.7.2.5) in order to account for the trade-off between resource production and biomass growth.

Table 6. Parameters of process module ConstantProduction.

Parameter	Dimension	Type	Variable type	Explanation
production rate	M/M*/T	microbe parameter	Decimal	The rate at which resources are produced

2.2.7.2.5 CooperativityFitnessCost

Some metabolites are costly to synthesize and, thus, can reduce the producers growth rate (D'Souza *et al.* 2014). Using Flux Balance Analysis (Orth *et al.* 2010) we observed that increase of production rates of metabolites result in a linearly dependent decrease of biomass growth. This can be explained by the cells limited capacity of pathways such as the TCA-cycle and glycolysis for the production of ATP for either biomass growth or metabolic overproduction. This process module can be used to model this trade-off and thereby to incorporate consequences of the inner-cell metabolism in an aggregated way (instead of integrating a full FBA). It associates the production of resources with a fitness cost that results in reduced biomass growth. Using this process module manipulates the parameter values *production rate* from the process module ConstantProduction (Section 2.2.7.2.4) and *yield* from the process module Growth (2.2.7.2.7) according to

$$p_R^{new} = c\, p_R,$$

$$y^{new} = (1 - c)\, y$$

, where p_R is the original parameter value of *production rate*, p_R^{new} is the new parameter value of *production rate*, y is the original parameter value of *yield*, y^{new} is the new parameter value of *yield*, and c is the microbe parameter *cooperativity*. If the parameter value of *cooperativity* changes during simulation (e.g. by the process module ParameterMutator (Section 2.2.7.2.11), the

parameters *production rate* and *yield* will be updated based on their original values at model initialization.

Table 7. Parameters of process module CooperativityFitnessCost.

Parameter	Dimension	Type	Variable type	Explanation
cooperativity	-	microbe parameter	Decimal	Defines how much biomass yield is sacrificed for the production of resources. A value of 1 results in full resource production but no biomass yield while a value of 0 results in no resource production but full biomass yield

2.2.7.2.6 Flow

In liquid environments, passive movement of microorganisms can be attributed for example to flow and Brownian motion. This process module allows to simulate different kinds of movement that can be described by a constant velocity and a random walk. The movement of the microbes is computed by

$$\tilde{X} = (v_x + \varepsilon_x)\, dt,$$

$$\tilde{Y} = \left(v_y + \varepsilon_y\right) dt,$$

$$\tilde{Z} = (v_z + \varepsilon_z)\, dt$$

, where $\tilde{X}$, $\tilde{Y}$, and $\tilde{Z}$ are the temporary variables for the changes made by this process module, v_x, v_y, and v_z are the components of the velocity vector as defined by the global parameters *mean flow X*, *mean flow Y*, and *mean flow Z*, and ε_x, ε_y, and ε_z are the components of a random vector drawn from a normal distribution with a mean of zero and standard deviations corresponding to the global parameters *flow SD X*, *flow SD Y*, and *flow SD Z*. On one hand, this allows to integrate Brownian motion on a very mechanistic level according to Einstein (1905) by setting the standard deviations to

$$SD = \sqrt{dt} * \sqrt{\frac{k_B T}{3\pi k P}}$$

, where k_B is the Boltzmann constant, T is the absolute temperature, k is the viscosity of the fluid, and P is the mean diameter of the microbes. On the other hand, this process module can be also used in a less mechanistic way by assuming constant flow ignoring turbulences and the displacement of resources.

Table 8. Parameters of process module Flow.

Parameter	Dimension	Type	Variable type	Explanation
mean flow X	S/T	global parameter	Decimal	The velocity at which microbes move in X direction
mean flow Y	S/T	global parameter	Decimal	The velocity at which microbes move in Y direction
mean flow Z	S/T	global parameter	Decimal	The velocity at which microbes move in Z direction
flow SD X	S/T	global parameter	Decimal	Standard deviation of movement in X direction
flow SD Y	S/T	global parameter	Decimal	Standard deviation of movement in Y direction
flow SD Z	S/T	global parameter	Decimal	Standard deviation of movement in Z direction

2.2.7.2.7 Growth

All organisms transform consumed resources partly into biomass. With this process module, the biomass growth of the microbes is modeled by

$$\tilde{B} = y\,G,$$

$$\tilde{G} = -G$$

, where $\tilde{B}$ and $\tilde{G}$ are the temporary variables for the changes made by this process module with respect to the biomass B and the amount G of resources that has been allocated to growth by the process module SubstrateUtilization (Section 2.2.7.2.16). y is the microbe parameter *yield*. Note that this process module only utilizes the substrate that has been previously allocated for growth and, thus, makes no assumptions with regard to rates of resource uptake or metabolism.

Table 9. Parameters of process module Growth.

Parameter	Dimension	Type	Variable type	Explanation
yield	M*/M	microbe parameter	Decimal	The rate at which resources are transformed into biomass

2.2.7.2.8 ImpermeableMicrobeBoundaries

For processes that affect the position of the microbes, this process module allows to change from default periodic (also referred to as toroidal) boundaries to closed boundaries, separately for each spatial dimension. This can be controlled by the parameters *impermeable X-boundaries*, *impermeable Y-boundaries*, and *impermeable Z-boundaries*, respectively. This is implemented as follows: Microbes that approach the impermeable boundaries closer than the value of the parameter *impermeable boundary offset* are moved away from the boundary such that their distance to the

boundary equals the value of the parameter *impermeable boundary offset*. It is therefore important to set a large enough offset (the modeler needs to estimate how much a microbe can move within one time step) because if a microbe moves far enough to cross the actual border of the environment within a time step, its new location will be computed according to the default periodic boundaries.

Table 10. Parameters of process module ImpermeableMicrobeBoundaries

Parameter	Dimension	Type	Variable type	Explanation
impermeable X-boundaries	-	global parameter	Boolean	Indicates whether X-boundaries are impermeable
impermeable Y-boundaries	-	global parameter	Boolean	Indicates whether Y-boundaries are impermeable
impermeable Z-boundaries	-	global parameter	Boolean	Indicates whether Z-boundaries are impermeable
impermeable boundary offset	S	global parameter	Decimal	The distance from the impermeable point to the true boundary

2.2.7.2.9 Lysis

Microbial cell death is typically followed by disintegration of the cell membrane, whereby the cytoplasm is released to the environment. This process is called lysis and is known to be governed by a complex regulatory system (Rice and Bayles 2008). However, so far, we have implemented microbial cell death as a non-mechanistic process that randomly removes microbes from the environment according to a probability defined by the parameter *lysis probability*.

Table 11. Parameters of process module Lysis.

Parameter	Dimension	Type	Variable type	Explanation
lysis probability	1/T	microbe parameter	Decimal	Between 0 and 1. Indicates the probability of a cell lysing

2.2.7.2.10 NanoTubeExchange

Bacteria have been observed to exchange cellular material by connecting each other with intercellular nanotubes that consist of membrane-derived lipids (Pande *et al.* 2015a, Dubey and Ben-Yehuda 2011). We developed an algorithm to mimic such resource exchange via intercellular nanotubes. Each microbe can connect to a number of microbes that is defined by the microbe parameter *max nanotubes* as it can be assumed that each microbe can only form a limited number of nanotubes. Nanotubes are also assumed to have a limited length, thus, the connected microbes are randomly chosen within a distance that is defined by the microbe parameter *max nanotube length*. Connected microbes disconnect if they move away further than the parameter value of *max nanotube length* or by chance, with a probability defined by the microbe parameter *nanotube disconnection probability*. Nanotubes are relatively little researched and concrete mechanisms that

govern the transfer of cellular material remain unknown. Thus, we assume that resources contained in a microbes *Product pool* (Section 2.2.7.2.4) are equally divided between all connected microbes. These fractions of the *Product pool* are added to the connected microbes *Substrate pool* if, and only if, the connected microbe's parameter *consumes resource* equals the focal microbe's parameter *releases resource*. Otherwise it is assumed that the recipient microbe has no use for the resource, which is why it is ignored.

Table 12. Parameters of process module NanoTubeExchange.

Parameter	Dimension	Type	Variable type	Explanation
consumes resource	-	microbe parameter	String	The resource that is consumed by respective type
releases resource	-	microbe parameter	String	The resource that is released by respective type
max nanotubes	-	microbe parameter	Integer	Number of microbes that a microbe can connect to
max nanotube length	S	microbe parameter	Decimal	Maximum distance over which microbes can connect
nanotube disconnection probability	1/T	microbe parameter	Decimal	The probability that an existing connection between two microbes disappears

2.2.7.2.11 ParameterMutator

Mutations are, together with natural selection, the drivers of evolution. Therefore, this process module facilitates evolutionary studies by simulating mutations that change the value of a parameter. The microbe parameter *mutation parameter* defines the parameter that is subject to mutations. The microbe parameter *mutation rate* defines the probability at which the value of the mutated parameter is changed per time step. Upon mutation, the respective parameter value is changed by a random value, drawn from a uniform distribution between *max mutation delta* and negative *max mutation delta*. The value of the mutated parameter cannot change beyond its limits which are defined by the microbe parameters *min mutation value* and *max mutation value*. This process module is not exactly a mechanistically sound representation of mutations, which would require to explicitly model genes which are translated into functional proteins (Gregory *et al.* 2004). However, it is a reasonable simplification that the modeled process rates, which are defined by parameter values, are controlled by a set of enzymes. Random mutations can lead to a shift in abundance of these enzymes or to altered efficiency, which ultimately results in the respective process rates being changed. Therefore, this process module still allows for a meaningful investigation of the effects of evolution with respect to specific traits.

Table 13 Parameters of process module ParameterMutator.

Parameter	Dimension	Type	Variable type	Explanation
mutation parameter	-	microbe parameter	String	The name of the parameter that is supposed to be mutated. For example *yield* from the process module Growth
mutation rate	1/T	microbe parameter	Decimal	The probability at which the mutated parameter value is changed per time step
max mutation delta	-	microbe parameter	Decimal	The upper limit of the absolute of the random value by which the mutated parameter is changed
max mutation value	-	microbe parameter	Decimal	The upper limit for the mutated parameter. If it is attempted to be changed above this limit, it will be set to the value of the limit
min mutation value	-	microbe parameter	Decimal	The lower limit for the mutated parameter. If it is attempted to be changed below this limit, it will be set to the value of the limit

2.2.7.2.12 PassiveRelease

Microorganisms can use various mechanisms to exchange resources (e.g. via intercellular nanotubes, membrane fusion, vesicles, and via diffusion through the environment) (D'Souza *et al.* 2018a). This process module simulates the release of intracellular resources into the environment, which is necessary for passive resource exchange via diffusion. For the sake of simplicity, we assume that at each time step, the entire pool of products is released. The amount of released resources is therefore controlled by the process module ConstantProduction (Section 2.2.7.2.4). The type of the released resource is specified by the microbial parameter *releases resource*. As microbes have spherical shapes with explicit sizes and continuous spatial positions, it can happen that an individual overlaps with several discrete grid cells of the environment. Therefore, at each time step, the resources contained in the *Product pool* P_R are distributed among the grid cells that the microbes overlap with, according to

$$\tilde{C}_{R,x,y,z} = B\ a_{x,y,z}\ P_R$$

, where $\tilde{C}_{R,x,y,z}$ is the temporary variable for the changes made by this process module with respect to $C_{R,x,y,z}$, the amount of resource R in the grid cell at position X, Y, Z. B is the biomass of the microbe and $a_{x,y,z}$ is the proportional overlap of the microbe with the grid cell at position X, Y, Z. The entire *Product pool* is consequently depleted:

$$\tilde{P}_R = -P_R$$

. Here $\tilde{P}_R$ is the temporary variable for changes made by this process module with respect to the *Product pool* P_R. Approximate proportions of the overlaps of the microbes with different grid cells are computed by the process module CellPartition (Section 2.2.7.2.2).

Table 14. Parameters of process module PassiveRelease.

Parameter	Dimension	Type	Variable type	Explanation
releases resource	-	microbe parameter	String	The resource that is released by respective type. The string must equal a resource's name parameter

2.2.7.2.13 PassiveUptake

The uptake of resources is a central process in numerous ecological models as it determines the organisms' metabolic activity (Brown *et al.* 2004). This process module simulates the consumption of resources from the environment by microbes. As microbes have continuous spatial coordinates while resources are discretized to a grid, it can happen that a microbe overlaps with several grid cells. Resources from grid cells that the microbe overlaps with are transferred into the intracellular pool S_R of resource R according to Monod-kinetics (Monod 1949)

$$\tilde{S}_R = B\ q\ dt = B \sum a_{x,y,z} \frac{q_{max} C_{R,x,y,z}}{K_M + C_{R,x,y,z}}\ dt,$$

$$\tilde{C}_{R,x,y,z} = -B\ a_{x,y,z}\ q\ dt$$

, where $\tilde{S}_R$ and $\tilde{C}_{R,x,y,z}$ are temporary variables for the changes made by this process module with respect to the intracellular pool S_R and $C_{R,x,y,z}$ is the amount of resource R in the grid cell at position X, Y, Z. B is the biomass of the microbe, a_{xyz} is the proportional overlap of the microbe with the grid cell at position X, Y, Z, q_{max} is the maximum uptake rate, and K_M is the Monod half-saturation constant. Approximate proportions of the overlaps of the microbes with different grid cells are computed by the process module CellPartition (Section 2.2.7.2.2).

Table 15. Parameters of process module PassiveUptake.

Parameter	Dimension	Type	Variable type	Explanation
consumes resource	-	microbe parameter	String	The resource that is consumed by respective type. The string must equal a resource's name parameter
half-saturation constant	M/S³	microbe parameter	Decimal	Resource concentration in the vicinity of the microbe at which the uptake rate is half maximal (according to the microbe parameter *max uptake*)
max uptake	M/M*/T	microbe parameter	Decimal	Theoretical maximum uptake rate of resource

2.2.7.2.14 Replication

Microbes reproduce by cell-division, a complex and highly regulated process (Harry *et al.* 2006). This process module implements cell-division based on the simplified assumption that microbial cells divide into two identical daughter cells as soon as a critical biomass is reached. This critical

biomass is defined by the microbe parameter *maximum biomass* (Section 2.2.7.1.3). Upon division, the microbe's biomass is divided by two and another microbe with identical state variable values is added to the environment. As the two microbes occupy the exact same location, the newly added microbe is shifted in a random direction by a very small distance (i.e. < 0.0001 S). This allows the process module Shoving (Section 2.2.7.5.3) to operate and minimize the overlap of both microbes.

2.2.7.2.15 Starvation

This process module simulates death due to starvation. The implementation does not represent a mechanistic process. If a microbe's state variable *Starvation* indicates starving, it is removed from the environment.

2.2.7.2.16 SubstrateUtilization

This process module represents the core of the microbes' metabolism. In ecology, there are several approaches to model metabolism in a mechanistic way. For example, constraint based modeling (in particular Flux Balance Analysis, Orth *et al.* 2010) can be used to model the fluxes through all metabolic pathways based on metabolic reconstructions of genetic data (e.g. Bauer *et al.* 2017, Biggs and Papin 2013). We implement a more simplified approach according to Herbert (1958) in which a fraction of the consumed resources is allocated for maintenance and the remainder for biomass growth. Thus, the amount of consumed resources allocated to maintenance M is given by

$$M = -B\, m\, dt$$

, where B is the biomass of the microbe and m is the microbe parameter *maintenance cost*. If this amount exceeds S_R, this microbe is flagged as *starving* (see also Starvation, Section 2.2.7.2.15). Otherwise, the remainder of S_R is transferred to the microbe's state variable *Growth resources* (G), that is to the pool of resources allocated to biomass growth

$$\tilde{G} = \max(S_R - M, 0),$$

$$\tilde{S}_R = -S_R$$

. Here $\tilde{G}$ and $\tilde{S}_R$ are the temporary variables for the changes made by the process modules with respect to the *Growth resources* G and the consumed resources S_R. Note that the chosen implementation of maintenance results in a restriction regarding the choice of time step lengths. The time step of the process module PassiveUptake (Section 2.2.7.2.13), which provides resources into the *Substrate pool,* is supposed to be smaller than or equal to the time step of Substrate-Utilization (Section 2.2.7.2.16). If it were chosen larger and, thus, SubstrateUtilization (Section 2.2.7.2.16) executed multiple times between executions of PassiveUptake (Section 2.2.7.2.13), the entire *Substrate pool* would be depleted during the first execution and hence the maintenance would be impossible to satisfy in subsequent executions. However, this restriction does not apply if the *maintenance cost* is set to zero.

Table 16. Parameters of process module SubstrateUtilization.

Parameter	Dimension	Type	Variable type	Explanation
maintenance cost	M/M*/T	microbe parameter	Decimal	The amount diverted into maintenance from the *Substrate pool*

2.2.7.3 Resource processes

2.2.7.3.1 ResourceDecay

It is widely reported that the decay of resources affects the ecology and evolution of microbial systems (e.g. Kümmerli and Brown 2010, Allen *et al.* 2013, Dobay *et al.* 2014). This process module simulates decay of resources over time in the environment. In accordance with the models of Kümmerli and Brown (2010) and Allen *et al.* (2013), the concentration of the resources is reduced by a constant rate, according to

$$\tilde{C}_{R,x,y,z} = (1 - d_R) C_{R,x,y,z}$$

, where $\tilde{C}_{R,x,y,z}$ is the temporary variable for changes made by this process module with respect to $C_{R,x,y,z}$, the concentration of resource R in position X, Y, Z, and d_R is the decay rate of R as specified by the resource parameter *decay rate*.

Table 17. Parameters of process module ResourceDecay.

Parameter	Dimension	Type	Variable type	Explanation
decay rate	1/T	resource parameter	Decimal	Between 0 and 1. Defines the rate at which each resource decays in the environment

2.2.7.4 Global processes

2.2.7.4.1 Diffusion

Diffusion is one of the fundamental processes accounted for in the majority of spatially explicit models of microbial interactions (e.g. Mitri *et al.* 2015, Bauer *et al.* 2017, Allen *et al.* 2013, Momeni *et al.* 2013b, Pande *et al.* 2016b). With this process module, diffusion of resources is modeled according to Fick's second law of diffusion. We applied the finite difference method in three dimensions as described by Mugler and Scott (1988) for two dimensions, which results in

$$\tilde{C}_{R,x,y,z} = D_R \frac{dt}{dS^2}\left(C_{R,x+dS,y,z} + C_{R,x-dS,y,z} + C_{R,x,y+dS,z} + C_{R,x,y-dS,z} + C_{R,x,y,z+dS} + C_{R,x,y,z-dS} - 6\, C_{R,x,y,z}\right),$$

where $\tilde{C}_{R,x,y,z}$ is the temporary variable for changes made by this process module with respect to $C_{R,x,y,z}$, the concentration of resource R in position X, Y, Z, D_R is the diffusion constant of that resource according to the parameter *diffusion constant*, and dS is the distance between the

midpoints of two adjacent resource grid cells which is always 1 S. The solution of the equation above is unstable if $D_R * dt > \frac{1}{6}$, hence the parameter *diffusion constant* and the time step dt must be sufficiently small.

Table 18. Parameters of process module Diffusion.

Parameter	Dimension	Type	Variable type	Explanation
diffusion constant	S^2/T	resource parameter	Decimal	Diffusion constant

2.2.7.4.2 LocalSource

This process module allows to introduce local sources or sinks of resources. The positions of these points are defined by the resource parameters *local source X*, *local source Y*, and *local source Z*. The parameter *local source type* specifies whether the resource concentration as defined by the resource parameter *local source concentration* is set only at simulation start ('once'), in each time step ('set'), or the defined concentration is added in each time step ('add'). When the *local source type* is set to 'add', a negative value can be chosen in order to simulate a sink. However, this can result in a negative local resource concentration which will cause the simulation to stop with a warning message. Therefore, it is better to simulate a sink by using 'set' to 0 (or also a low positive value), if this is reasonable in the modeled context.

Table 19. Parameters of process module LocalSource.

Parameter	Dimension	Type	Variable type	Explanation
has local source	-	resource parameter	Boolean	Indicates whether this resource has any local sources
local source X	S	resource parameter	Decimal	X position(s) of local source(s). If multiple sources exist, the positions are separated by whitespaces ' '
local source Y	S	resource parameter	Decimal	Y position(s) of local source(s). If multiple sources exist the positions are separated by whitespaces ' '
local source Z	S	resource parameter	Decimal	Z position(s) of local source(s). If multiple sources exist, the positions are separated by whitespaces ' '
local source concentration	M/S^3	resource parameter	Decimal	Concentration(s) of local source(s). If multiple sources exist, the concentrations are separated by whitespaces ' '
local source type	-	resource parameter	String	'add', 'set', or 'once'. Indicates, whether the concentration of each source is added to the respective grid cell each time step (add), held constant (set), or set once upon model initialization

2.2.7.4.3 LongTermExperiment

Usually, exponential growth of the microbes limits the simulations to some generation times before computation becomes too demanding, which hinders the observation of long-term processes. Similarly, in experiments, microbial cultures often reach the capacity of the provided medium

before long-term dynamics can be assessed. Experimentally, this issue can be overcome with long-term experiments in which the cultured microorganisms are sequentially transferred into fresh medium upon reaching a certain density (e.g. Lenski 2017, Preussger *et al.* 2020). Analogously, this process module allows for longer simulations by reducing the number of microbes when they exceed a predefined abundance given by the global parameter *transfer at microbe count*. Then, the microbes are reduced to a random subset of a size defined by the global parameter *transfer microbes*. Furthermore, the resources are reset to their initial value and the positions of the remaining microbes are randomly changed. If any initial processes need to be repeated at this point, their names can be indicated in the global parameter *Initial processes on transfer*.

Table 20. Parameters of process module LongTermExperiment.

Parameter	Dimension	Type	Variable type	Explanation
transfer at microbe count	-	global parameter	Integer	Number of microbes upon which microbes are removed and resources are reset
transfer microbes	-	global parameter	Integer	Number of microbes that remain when microbes are removed and resources are reset
Initial processes on transfer	-	global parameter	String	Initial processes that need to be repeated upon removing microbes and resetting resources. Multiple processes can be separated by semicolons '; '

2.2.7.4.4 ProximityManager

This process module does not represent a biological or physical process but is required by some other process modules for convenience (Table 25). This process module discretizes the space of the simulated environment into raster cells of a size defined by the global parameter *proximity raster cell size* and groups all microbes together that overlap to the greatest extent with the same raster cell. The estimate of which microbes are in rough vicinity to a given microbe consists of all microbes in the same raster cell and all 26 adjacent raster cells (Note that the proximity raster cells may differ from the resource grid cells, which are always of size 1 S^3). This increases computational efficiency for process modules that simulate local interaction between microbes (e.g. Attachment, Section 2.2.7.2.1; Shoving, Section 2.2.7.5.3).

Table 21. Parameters of process module ProximityManager.

Parameter	Dimension	Type	Variable type	Explanation
proximity raster cell size	S	global parameter	Integer	The length of raster cells that are used to group microbes into discrete spatial positions

2.2.7.5 Postprocessing modules

2.2.7.5.1 ConstantResourceBoundaries

This process module allows to change from the default periodic boundary conditions for processes that affect resource concentrations to Dirichlet boundary conditions (i.e. a constant value is assumed at the boundaries), separately for each spatial dimension. This can be controlled with the parameters *constant X-boundaries*, *constant Y-boundaries*, and *constant Z-boundaries*, respectively. On execution of this process module, if applied to the corresponding dimension, all resource amounts in the outermost grid cells are set to the value of *concentration at boundary*.

Table 22. Parameters of process module ConstantResourceBoundaries.

Parameter	Dimension	Type	Variable type	Explanation
constant X-boundaries	-	resource parameter	Boolean	Indicates whether X-boundaries are constant
constant Y-boundaries	-	resource parameter	Boolean	Indicates whether Y-boundaries are constant
constant Z-boundaries	-	resource parameter	Boolean	Indicates whether Z-boundaries are constant
concentration at boundary	M/S^3	resource parameter	Decimal or String	If Decimal, the concentration that is maintained at fixed boundaries. If String "as initial", the resource parameter value *initial concentration* is used, which is convenient when the initial concentration is varied

2.2.7.5.2 ExtrapolatingResourceBoundaries

In spatially explicit models, the simulated space is often representing only a fraction of the modeled system. In order to avoid boundary value problems, often periodic (also referred to as toroidal) boundary conditions are assumed (e.g. Momeni *et al.* 2013b, Allison 2005). However, this assumes that sources and sinks of resources (e.g. microbes that consume or release metabolites) are present at the same density beyond the boundaries. This is, however, not always true, such as in the case of a single colony growing on an agar plate, where periodic boundaries would result in an accumulation of resources that would otherwise diffuse out of the system. To solve this issue we developed an algorithm that estimates the concentration of resources at open boundaries by extrapolating the resource concentration based on an exponential regression. For this, three grid cells in a row of which one is adjacent to the grid cell in question are used for the exponential regression. However, in order to prevent uncontrolled increase of resources, the extrapolated concentration is never higher than in the adjacent cell. These boundary conditions can be set for each spatial dimension individually with the parameters *extrapolate X-boundaries*, *extrapolate Y-boundaries*, and *extrapolate Z-boundaries*, respectively.

Table 23. Parameters of process module ExtrapolatingResourceBoundaries.

Parameter	Dimension	Type	Variable type	Explanation
extrapolate X-boundaries	-	resource parameter	Boolean	Indicates whether X-boundaries are extrapolated
extrapolate Y-boundaries	-	resource parameter	Boolean	Indicates whether Y-boundaries are extrapolated
extrapolate Z-boundaries	-	resource parameter	Boolean	Indicates whether Z-boundaries are extrapolated

2.2.7.5.3 Shoving

Shoving describes the process of the microbes pushing each other away if they overlap, also referred to as spatial relaxation. The shoving algorithm from iDynoMiCS (Lardon *et al.* 2011a; Algorithm S1) is used. It is not always possible to get rid of any overlap between microbes in reasonable computational time (the algorithm stops after 12 iterations), therefore the process aims to minimize the total overlap as much as possible.

2.3 Summary and outlook

McComedy is a framework for individual-based modeling of microbial systems which explicitly accounts for consumer-resource interactions and their evolution. Its special feature is the implementation of various process modules, that is, generic submodels of biological, physical and evolutionary processes. In parts, these process modules are based on first principles as they are implemented according to physical laws or well-understood biological principles, such as the process modules Diffusion (Section 2.2.7.4.1), CooperativityFitnessCost (Section 2.2.7.2.5), and PassiveUptake (Section 2.2.7.2.13). However, some other process modules are so far implemented according to abstract, non-mechanistic rules, such as the process modules Lysis (Section 2.2.7.2.9), Replication (Section 2.2.7.2.14), and Starvation (Section 2.2.7.2.15). Thus, *McComedy* takes a step towards next-generation IBMs, however, future development of the framework needs to address the process modules lacking mechanistic principles.

The process modules can be combined to IBMs of different microbial systems and scenarios. During simulations with *McComedy*, an innovative scheduling systems integrates the independently running process modules and ensures synchronous updates of the affected state variables.

Particularly, models that focus on metabolic interactions and evolution are the intended field of application for *McComedy*. Metabolic interactions can be addressed with the process modules ConstantProduction (Section 2.2.7.2.4), NanoTubeExchange (Section 2.2.7.2.10), Substrate-Utilization (Section 2.2.7.2.16), PassiveRelease (Section 2.2.7.2.12), and PassiveUptake (Section 2.2.7.2.13). Evolutionary studies are facilitated by the process modules ChangeGenotype (Section

2.2.7.2.3), ParameterMutator (Section 2.2.7.2.11), and LongTermExperiment (Section 2.2.7.4.3). At the interface of metabolic interactions and evolution is the process module Cooperativity-FitnessCost (Section 2.2.7.2.5), which assigns fitness costs to the overproduction of resources. Examples of how IBMs can be created and parametrized with *McComedy* and how the model output can be analyzed are provided in the following two chapters. These examples include two reproductions of previously reported studies, which validate that IBMs created with *McComedy* can accurately capture relevant dynamics of microbial systems.

In order to maximize the usability of *McComedy*, the framework was implemented as a user-friendly software tool with a graphical user interface (https://git.ufz.de/bogdanow/McComedy). In the next chapter, the software solution is presented in more detail.

3 McComedy: Software and validation[2]

3.1 Introduction

Microbial communities are pervasive across all ecosystems and most often essential for their functioning (Prosser *et al.* 2007, Widder *et al.* 2016a). However, the vast taxonomic diversity of their members, manifold interactions within communities and between microorganisms and their environments, as well as heterogeneities (e.g. in composition and functioning) across spatial and temporal scales pose a major challenge to understand their ecology (Prosser *et al.* 2007, Widder *et al.* 2016a, Curtis and Sloan 2005, Pacheco and Segre 2019, Ladau and Eloe-Fadrosh 2019). On the other hand, a better sense of how microbial communities assemble and respond to environmental conditions is essential to fuel advance in various research fields such as medicine, biotechnology, and climate change research (Cavicchioli *et al.* 2019, McCarty and Ledesma-Amaro 2019, Costello *et al.* 2012).

Microbial community dynamics usually involve metabolic interactions such as the exchange of and competition for resources (D'Souza *et al.* 2018b, Hibbing *et al.* 2010). Focusing on those interactions, microbial communities together with the resources can be viewed as consumer-resource systems. Traditionally, consumer-resource systems are modeled using differential equations for the densities of consumer and resource species at the level of populations (MacArthur 1970, Chesson 1990). Such population-level equations are still applied in microbial ecology (Marsland *et al.* 2020, Estrela *et al.* 2021, Pacciani-Mori *et al.* 2020), but recent research of microbial consumer-resource systems is increasingly concerned with the dynamics within populations, particularly when a spatial component needs to be explicitly considered (Momeni *et al.* 2013b, Mitri *et al.* 2015, Pande *et al.* 2016a, Bauer *et al.* 2017). Such spatially explicit approaches can provide insight on how localized processes (e.g. cross-feeding in a structured environment, Momeni *et al.* 2013b, Pande *et al.* 2016a) shape the community on a larger scale. For that, individual-based models (IBMs) are widely applied (Hellweger *et al.* 2016).

IBMs are commonly used to investigate the dynamics of populations or communities by simulating individual entities, which in ecology usually represent individual organisms (DeAngelis and Grimm 2014, Botkin *et al.* 1972). The dynamics of populations and communities then emerge from the simulated behavior of these individuals. This bottom-up approach has been shown to be particularly useful for modeling complex systems, where individuals exhibit trait variation, adaptive behavior, or localized interactions (Railsback 2001, Grimm and Berger 2016).

[2] Based on Bogdanowski A., Banitz T., Muhsal L. K., Kost C. and Frank K. 2022. Mccomedy: A user-friendly tool for next-generation individual-based modeling of microbial consumer-resource systems. PLoS Comput Biol 18: e1009777.

The relatively well-understood nature of individual microorganisms in terms of movement, metabolism, and reproduction (as opposed to the more complex dynamics at the level of populations and communities) makes microbial systems particularly well-suited for simulation in IBMs. For this reason, IBMs are frequently applied to analyze different aspects of microbial ecology and evolution (Hellweger *et al.* 2016). The simulation model results can be analyzed on different levels of organization, ranging from below (e.g. metabolic networks within individual microorganisms, Bauer *et al.* 2017, Biggs and Papin 2013, Harcombe *et al.* 2014), at (e.g. movement trajectories of individuals) and above the level of individuals (e.g. spatial distributions of entire populations or community compositions Mitri *et al.* 2015, Momeni *et al.* 2013b, Pande *et al.* 2016a). In addition, the resulting data can be directly compared to results derived from experiments, thus making IBMs very powerful to link empirical observations with theory.

IBMs can be distinguished between traditional ones and so-called next-generation IBMs (Grimm *et al.* 2016). Traditional IBMs are designed and parametrized on the basis of site-specific measurements (e.g. the interaction of two species is modeled according to their co-occurrence in the modeled ecosystem). This makes these models non-generic and non-transferable to other environments (Grimm *et al.* 2016). Next-generation IBMs overcome this drawback by constructing the individuals' behavior from generic submodels that are based on well-understood principles from physics, chemistry, physiology, and evolutionary biology (Grimm and Berger 2016, Grimm *et al.* 2016). This mechanistic approach increases the propensity of the models to capture the organization and functioning of the real system (i.e. *structural realism*) rather than only matching empirically observed patterns (Grimm and Berger 2016). In microbial ecology, this is reflected in several IBMs (e.g. Bauer *et al.* 2017, Lardon *et al.* 2011b, Gras and Ginovart 2006, Li *et al.* 2019), which result in strikingly realistic model behavior and a thorough understanding of ecological mechanisms. Besides providing specific insights in their respective fields of application, these models demonstrate the general potential of next-generation IBMs for microbial ecology. However, building and using next-generation IBMs usually requires good knowledge in programming or proficiency with specific software tools, which hinders a more widespread application by microbial ecologists. An easy-to-use framework that facilitates the development of such IBMs from pre-implemented, tested, generic and mechanistically sound submodels could therefore contribute significantly to the field.

Here we present the modeling tool *McComedy* (**Mic**robial **Co**mmunities, **Me**tabolism, and **Dy**namics), which constitutes a framework for individual-based modeling of microbial consumer-resource systems. A central idea of this framework is to provide generic submodels based on biological and physical principles, which we refer to as *process modules* and which can be combined and parametrized in a user-friendly graphical interface, resulting in ready-to-use next-generation IBMs. We tested the validity of our approach by using *McComedy* to implement two specific IBMs corresponding to two different studies of spatial and evolutionary dynamics of

microbial communities, which involved both experiments and IBMs. For both cases, we demonstrate that the respective model constructed with *McComedy* was able to robustly reproduce the general results and capture the essential mechanisms underlying the microbial community dynamics in the original studies. Furthermore, we demonstrate how *McComedy* can be used for tackling open research questions by extending the two original studies with additional insights.

3.2 Results

3.2.1 McComedy

McComedy is an open-source modeling tool for developing and using IBMs of microbial communities, with a focus on consumer-resource interactions and their implications for the functioning of the corresponding communities. This tool was developed to facilitate fast and user-friendly operation as well as to grant high flexibility in model design (Figure 4). The software can be downloaded from https://git.ufz.de/bogdanow/mccomedy, where also the source code and a tutorial on how to get started are provided. To create a new IBM, the user can select several process modules, which implement biological and physical processes of relevance for microbial consumer-resource systems such as consumption or production of resources, resource diffusion through the environment, and growth of individual microorganisms. Next, parameter values of the selected process modules can be defined according to the specifics of the modeled system on the basis of empirical observation or literature. Subsequently, simulations are executed and spatially explicit data on the modeled system at discrete time points is generated.

A specific IBM created with *McComedy* describes a three-dimensional environment in which individual microorganisms (in *McComedy* referred to as *microbes*) and resources interact. Microbes are modeled as individual objects with spherical shapes and continuous positions in the environment. Specific types can be defined that differ in certain traits, such as their metabolic requirement or growth parameters. Resources are not modeled as individual particles but instead as concentrations in each grid cell of a three-dimensional grid covering the simulated environment. Resources are different metabolites that can be consumed or produced by the microbes. If necessary, the three-dimensional environment can be reduced to represent two dimensions by constraining the third dimension to just one layer of grid cells.

Over the simulated time span, microbes and resources are subject to the modeled processes. These processes are encapsulated in so-called process modules, which mediate direct and indirect interactions among microbes and between microbes and the resources. Each process module simulates one component of the system dynamics, such as microbial growth or resource diffusion. In order to facilitate a flexible yet functional model design, each process module is implemented based on generic principles, which means that no *ad-hoc* assumptions are made for particular model applications. Instead, the dynamics of each modeled microbial system emerge entirely from the

same pool of generic principles. For example, the process module *Growth* transforms consumed resources into biomass under consideration of a yield to be defined (Section 2.2.7.2.7). The module *Replication* divides a microbe individual into two once a critical biomass has been reached (Section 2.2.7.2.14). These processes are mechanistically valid regardless of the specific modeled system and are therefore preferable to alternatives, such as imposed rules or *ad-hoc* assumptions (e.g. a microbe replicating by chance when it is close to resources). We use the term *generic principles* instead of *first principles*, which is also common in the literature (Grimm and Berger 2016), because we do not claim that our processes are completely described by scientific laws, as we also use reasonable simplifications if we consider them mechanistically sound. *McComedy* does not allow for imposing higher-level processes (e.g. spatial pattern formation or density-dependent regulation of population size) as such dynamics are supposed to emerge from the generic process modules.

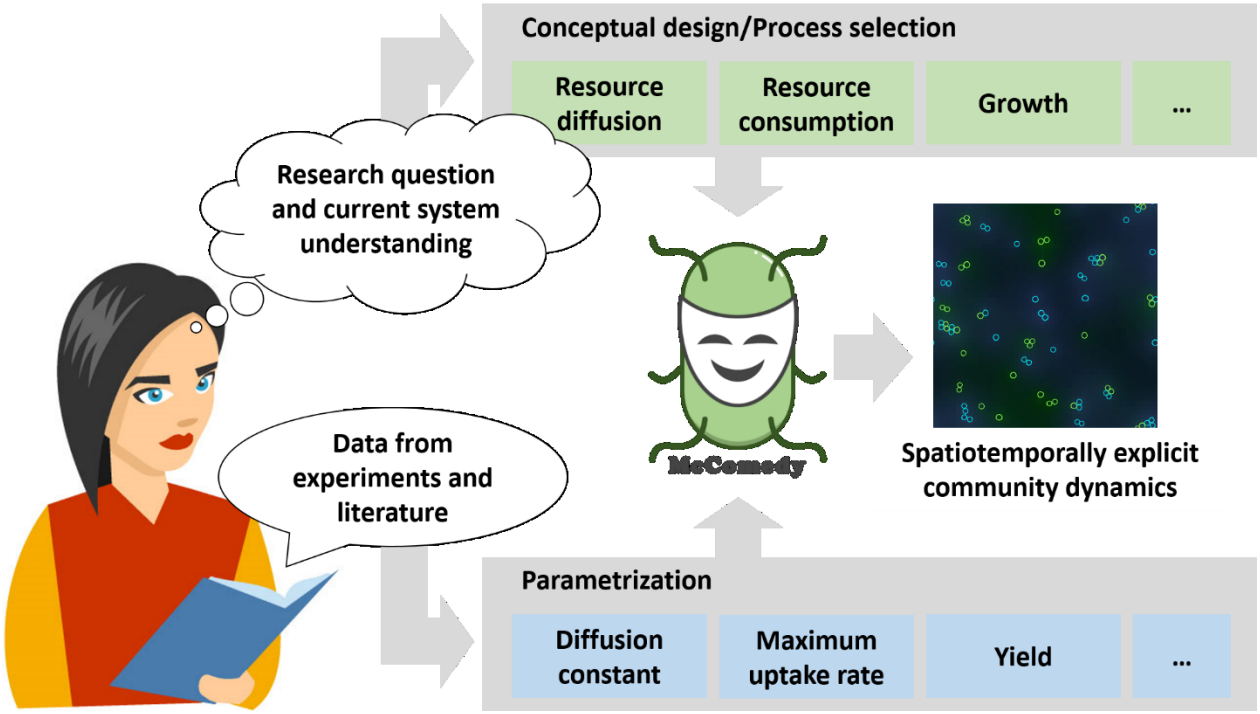

Figure 4. Intended workflow when using McComedy. The modeler designs an individual-based model (IBM) by selecting process modules under consideration of the research question and the current understanding of the system. The parameter values that are necessary for the simulation of the selected processes are set by the modeler, e.g. according to experimental data or literature. The resulting IBM generates spatiotemporally explicit data of the modeled microbial system.

The graphical user interface of *McComedy* supports a fast and user-friendly model development. The user is guided through different development stages, starting with the selection of process modules. According to the selection, *McComedy* shows tables containing the required parameters with editable default values. The user can also specify lists of values for single parameters and *McComedy* will run simulations for every combination of these parameter values. Moreover, the

user can control technical settings such as the number of replicates and the configuration of the model output.

The model output is generated separately for each individual simulation, in order to facilitate comparative analyses with regard to parameter variations as well as variance analyses due to stochasticity. For each simulation, the state of each microbe and resource grid cell is written into result files at predefined time intervals. The aforementioned state includes spatial coordinates, biomass, microbial type, resource concentration, as well as other properties, which allow not only for a highly-resolved and spatially explicit model analysis, but also for a direct statistical comparison with a variety of empirical data (e.g. growth kinetics, spatial patterns, functional responses, etc.).

The computation time for a simulation depends mostly on the size of the simulated environment, the number of microbes included, the time step lengths of the process modules, the termination condition, and the hardware used. Simulating a microbial community for 10 virtual hours on a regular computer can take between few minutes and several days. We provide an estimate of reference computation times on a current standard computer for different representative parameter choices in A.2 Computational performance.

For further details on the implementation and use of *McComedy* please consult the Methods Section as well as the ODD protocol (Section 2.2).

To demonstrate that the IBMs built with *McComedy* can capture and serve to analyze the dynamics of specific microbial systems, *McComedy* was used to reproduce the outcomes of two exemplary studies of microbial systems. The studies were chosen from the literature based on the close correspondence of their research questions to *McComedy*'s intended field of application. Thus, both studies assess spatial structuring in microbial communities as a consequence of consumer-resource interactions. We compared the outcomes of *McComedy* with the empirical and modeling results of the original studies. In the following, we show how *McComedy* can help to analyze and compare the respective results and how it can provide additional insight into the underlying mechanisms.

3.2.2 Example 1: Spatial organization model (Mitri *et al.* 2015)

To fully understand the functioning of microbial communities, it is essential to identify the drivers of spatial structuring and the diversity in their assemblage. In this context, Mitri *et al.* (2015) conducted experiments with bacteria to assess whether resource limitation leads to spatial separation of different strains in an initially well-mixed, growing colony. The authors used an IBM to identify the ecological mechanisms underlying their experimentally observed results.

In the experiment, two droplets of two differently labeled strains of *Pseudomonas aeruginosa*, visually discriminable by green and blue color, were spotted on nutrient-poor agar. After two weeks

of incubation, the colony had grown in size and exhibited a strong pattern of intermixing among the two strains in the center, yet a clear separation of green and blue bacteria in the outer rim of the colony. Increasing the initial resource concentration in the agar led to an increased demixing distance, defined as the distance between the initial inoculum and the region of spatial separation (Figure 5 A, Figure 2 a in Mitri *et al.* 2015). Quantitatively, the observed spatial structure of the colony was assessed by measuring the degree of heterozygosity (i.e. how much the two strains were intermixed in a given location) across the colony (Figure 5 B, Figure 2 c in Mitri *et al.* 2015). The demixing distance corresponds to the distance from the initial inoculum, at which heterozygosity showed the steepest decrease.

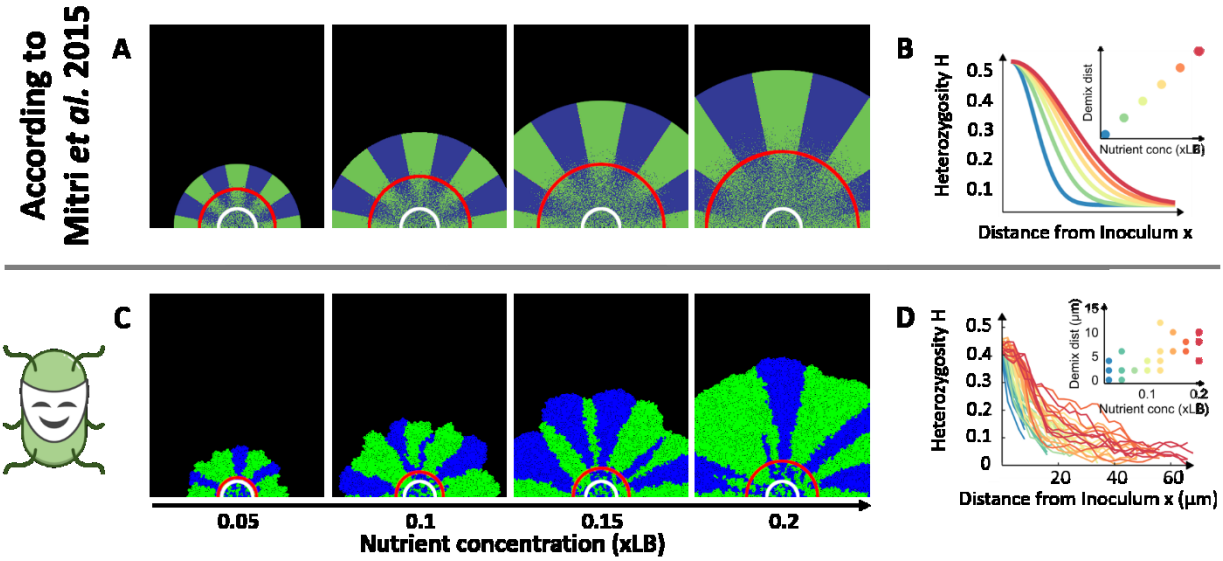

Figure 5. McComedy can reproduce the results of experiments and simulations by Mitri et al. (2015) both quantitatively and qualitatively. Top views on colonies at different initial resource (nutrient) concentrations and degree of heterozygosity over the distance to the inoculum. The unit xLB is defined as the fold-concentration of LB medium. Blue and green colors on colony images indicate the two bacterial strains. White circles on the colony images indicate the inoculum. Red circles indicate where the demixing area begins. Analyses with McComedy were conducted after 45 simulated hours of growth. A: Stylized recreation of top views on colonies at different resource concentrations according to Figures 2a and 4a in (Mitri et al. 2015). B: Stylized recreation of the heterozygosity over distance from inoculum and corresponding demixing distances at different resource concentrations according to Figures 2c and 4b in (Mitri et al. 2015). Axis labels of distances are not shown as they varied between experimental and simulation results and were of no consequence for the qualitative pattern. C: Representive top views on colonies at different resource concentrations in the McComedy IBM. D: Heterozygisity over distance from inoculum and estimated demixing distances at different resource concentrations in the McComedy IBM. Images A and B were recreated due to copyright issues. Refer to Figures 2 and 4 in (Mitri et al. 2015) to view the original data.

The correlation of resource concentration and demixing distance was hypothesized to be attributable to the varying resource accessibility at the periphery of the colony. At high resource concentrations, more resources diffuse into the colony, which support the growth of a higher number of bacterial cells, thus reducing the chance of excluding one strain from growth at a given location. Analogously, at low resource concentrations, growth of fewer bacterial cells is supported

at the edge of the colony, resulting in a more immediate loss of the local diversity. In population genetics, this mechanism is known as the bottleneck effect (Nei *et al.* 1975). Mitri *et al.* (2015) applied an IBM to recapitulate the empirical pattern (Figure 4 a and 4 b in Mitri *et al.* 2015). In agreement with the hypothesis, also in the model the demixing distance increased with increasing initial resource concentrations. Based on the analysis of their model, the authors thus concluded that the bottleneck effect in an expanding colony is indeed the mechanism that most likely explains spatial separation of bacteria under resource-limited conditions.

To validate *McComedy*, we created an IBM to recapitulate the results presented by Mitri *et al.* (2015). In accordance with the original system, the model was specified with two types of bacteria having exactly the same properties (except for their color) and a resource at varied, initially homogeneous concentrations in a two-dimensional environment. Process modules were selected to account for resource diffusion, resource consumption, microbial growth, and replication. A detailed description of the model implementation is provided in the Methods section and a complete list of the selected process modules and parameter values is available in Section A.5.

The model simulations generated very similar bacterial community dynamics compared to the original study (Figure 5 C and Figure 5 D) and also the spatial organization of the two strains (Figure 5 C) qualitatively matched those described by Mitri *et al.* (2015) (Figure 5 A and Figures 2 a and 4 a in Mitri *et al.* 2015). The resulting colonies showed a clear separation (demixing) of the two strains towards the edge of the colony, whereby the demixing distance also increased with higher resource concentrations. For a quantitative analysis of the demixing dynamics in response to different initial resource levels, the measure of heterozygosity was calculated based on the exact position of each single bacterial individual in the simulations (Figure 5 D) as was done for the original model results (Figure 4 b in Mitri *et al.* 2015). This analysis showed that heterozygosity dropped from approximately 0.5 (highly mixed strains) at the inoculum to almost zero (segregated strains) at the edge of the colony. Moreover, the demixing distance increased with initial resource concentrations. Quantitatively, both simulation models do not precisely match the experimental data and show slight discrepancies between each other, which may originate from different implementation details or choices of parameter values. However, the consistent qualitative response of the spatial pattern to the varied resource concentrations demonstrates that also the new model is well-suited to study the mechanisms generating such patterns. This is facilitated by *McComedy's* capabilities to observe and quantify the characteristics of spatiotemporal colony dynamics that emerge from suites of different scenarios.

To further test the potential of *McComedy* for understanding mechanisms operating in microbial consumer-resource systems, we used it to simulate colony growth under different resource diffusion constants, while keeping the initial resource concentration constant. This type of analysis should additionally corroborate the explanation by Mitri *et al.* (2015), which attributes the spatial

separation to the bottleneck effect. Here we hypothesized that increasing the resource diffusion constant should have an effect that is similar to increasing the initial resource concentration. Indeed, simulating increased rates of diffusion revealed that resources diffused deeper into the colony, thus resulting in less spatial segregation of both strains and an increased demixing distance (Figure 6). These results confirm that the bottleneck-effect drives the separation of the two populations and that *McComedy* is a powerful tool to investigate the spatiotemporal dynamics and mechanisms underlying experimental observations.

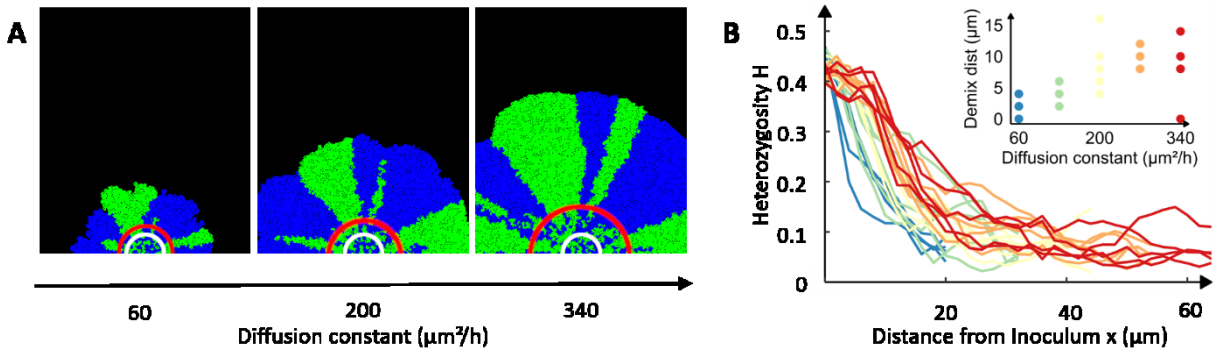

Figure 6. Increased diffusion resulted in an increased demixing distance. Simulations were performed with McComedy. Analysis after 39 simulated hours of growth. A: Top views on representative colonies as simulated using McComedy using different resource diffusion constants. Blue and green colors on colony images indicate the two bacterial strains. White circles on the colony images indicate the size of the inoculum. Red circles indicate where the demixing area begins. B: Heterozygosity over distance from inoculum and estimated demixing distance at different resource diffusion constants.

3.2.3 Example 2: Cooperation model (Momeni *et al.* 2013b)

The second example concerns research on the maintenance of cooperation in spatially structured environments, where pairs of individual microorganisms can interact repeatedly (as opposed to a well-mixed, spatially unstructured environment). In this context, it is important to understand how metabolic interactions affect the spatial organization of resident strains and thus the distribution of different strategists within microbial communities. Momeni *et al.* (2013b) performed experiments with yeast strains to investigate how spatial self-organization affects the abundance of cooperative and non-cooperative strains. For this, they used synthetically engineered cooperative and non-cooperative strains of *Saccharomyces cerevisiae*, of which the former two strains provided the resources lysine and adenine to the community. In their study, the observation that cooperators intermix, while non-cooperators spatially segregate, was explained using an IBM.

In particular, the authors designed a system with three strains of *Saccharomyces cerevisiae,* two complementary cooperators and one non-cooperator. One cooperator strain $G_{\rightarrow L}^{\leftarrow A}$ required adenine and released lysine upon cell death, while the other cooperator $R_{\rightarrow A}^{\leftarrow L}$ required lysine and continuously released adenine. In contrast, the non-cooperating strain $C^{\leftarrow L}$ also required lysine for growth, yet did not release any resource to enhance the growth of other cells. This latter strain

gained a fitness advantage from not sharing resources. After mixing the three strains and plating them on agar at low density, individual yeast cells formed colonies that increased in diameter, until the entire agar plate was covered after which the yeast cells started to grow upwards. During this process, the two cooperating strains intermixed with each other, grew well, and formed a thick layer, whereas the non-cooperators spatially segregated from the cooperators and only formed a thin layer of cells (Figure 7 B). In a control experiment, the growth medium was supplemented with adenine and lysine, such that the two cooperators could grow independently of the cooperation of their corresponding partners. Under these conditions, cooperation turned into competition for space and other limiting resources. As a consequence, none of the strains intermixed to a significant extent and the thickness of the microbial layer was almost uniform, independent of which strain formed it (Figure 7 A).

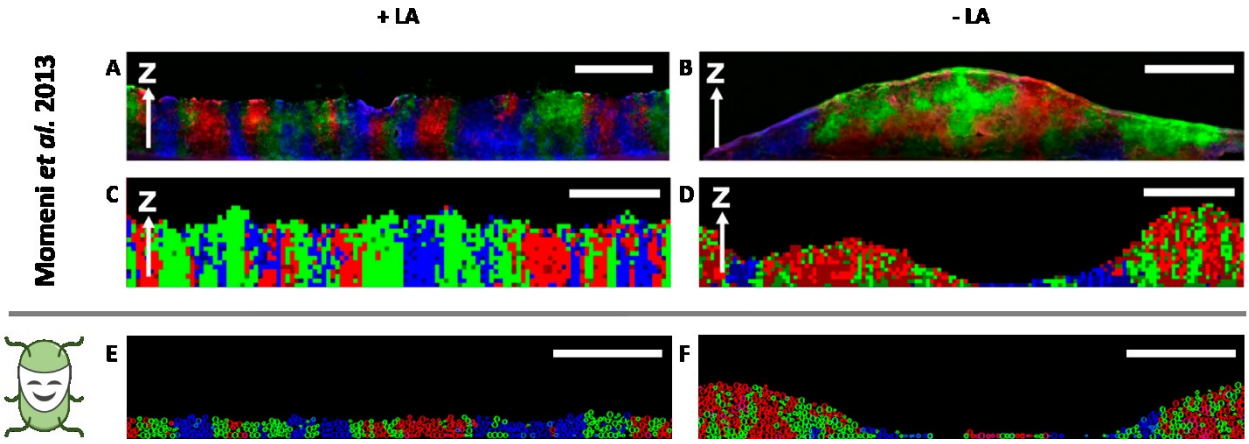

Figure 7. McComedy reproduces qualitative results of experiments and simulations by Momeni et al. (Momeni et al. 2013b). Vertical cross-section views on layers of yeast cells grown on media supplemented with lysine and adenine (+ LA) and on media without these resources (- LA). Red and green color indicates the two cooperative yeast strains, blue color indicates the non-cooperative yeast strain. Simulations performed with McComedy were visualized after 6 generations. A, C, E: Representative cross-sections of yeast cells grown with supplemented lysine and adenine (+LA) in the experiment, original IBM, and McComedy IBM, respectively. B, D, F: Representative cross-sections of yeast cells grown without lysine and adenine (-LA) in the experiment, original IBM, and McComedy IBM, respectively. Scale bar: 100 µm. Images A, B, C, D adapted from (Momeni et al. 2013b).

An IBM was used to recapitulate these experimental results and examine the mechanism that explains the observed exclusion of non-cooperating types. The simulations robustly reproduced the partner intermixing of the two cooperative strains (Figure 7 C and Figure 7 D). Moreover, it was shown that the spatial association of the cooperators $G_{\rightarrow L}^{\leftarrow A}$ with their partners $R_{\rightarrow A}^{\leftarrow L}$ increased over time compared to their association with the non-cooperating strain $C^{\leftarrow L}$. These association differences were quantified by computing the association index $A_{RG/CG}^{3D}$, which is the ratio between the frequencies of individuals of type $G_{\rightarrow L}^{\leftarrow A}$ in the direct vicinity of individuals of type $R_{\rightarrow A}^{\leftarrow L}$, and in the direct vicinity of individuals of type $C^{\leftarrow L}$ (Figure 8 A). Furthermore, the abundance of the cooperator $R_{\rightarrow A}^{\leftarrow L}$ relative to the corresponding non-cooperator $C^{\leftarrow L}$ (both of which compete for lysine) increased (Figure 8 C). The following mechanism drove the observed spatial self-

organization: distinct populations that reciprocally provide each other with localized benefits are expected to intermix as individual yeast cells grow best in the vicinity of a cooperating partner (Momeni *et al.* 2013a). By the same logic, populations that provide no localized benefits to the community are expected to segregate. The IBM demonstrated that this mechanism alone was sufficient to generate the observed spatial patterns and no additional rules implemented by the modeler such as e.g. partner recognition or positive chemotaxis were required.

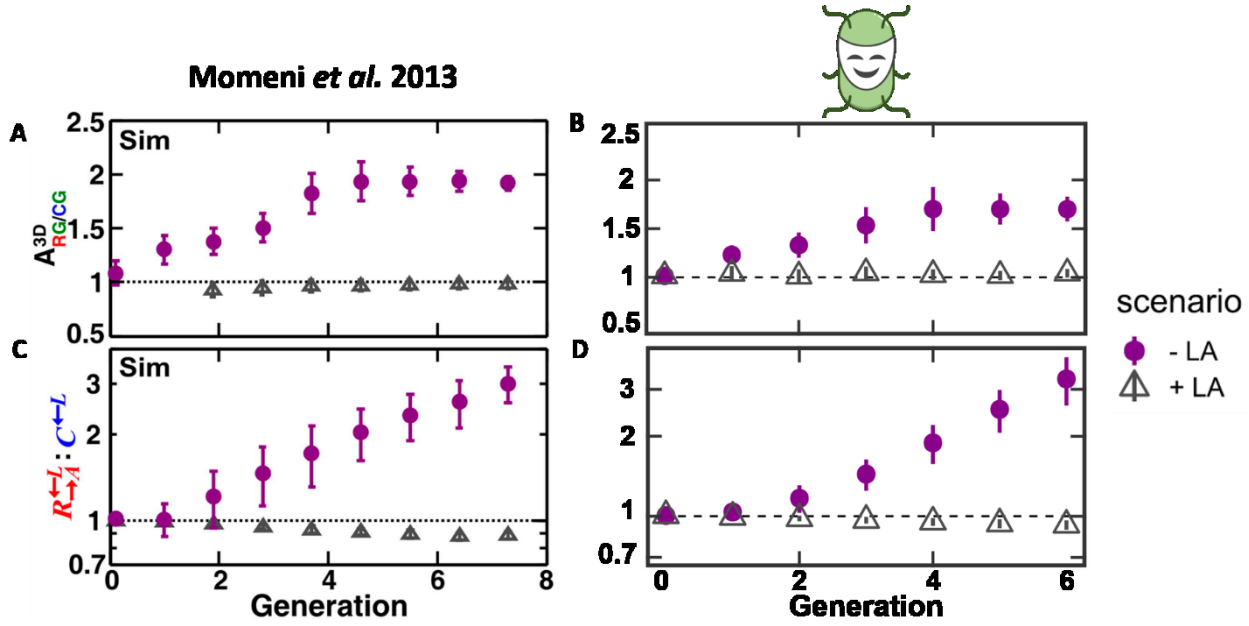

Figure 8. McComedy reproduces quantitative results of simulations by Momeni et al. (Momeni et al. 2013b). The quantitative metrics were assessed for yeast cells grown on media supplemented with lysine and adenine (+ LA) and on media without these resources (- LA). A, B: Association index of the two cooperative strains ($R^{\leftarrow L}_{\rightarrow A}$ with $G^{\leftarrow A}_{\rightarrow L}$) and the non-cooperators $C^{\leftarrow L}$ in the original IBM and McComedy IBM, respectively. C, D: Abundance ratio between the cooperators $R^{\leftarrow L}_{\rightarrow A}$ and the non-cooperators $C^{\leftarrow L}$ in the original IBM and McComedy, respectively. Note the logarithmic scales of the vertical axes. Images A, C adapted from (Momeni et al. 2013b).

To verify whether *McComedy* can reproduce the results of Momeni *et al.* (Momeni *et al.* 2013b), a corresponding model of a microbial system with two cooperating and one non-cooperating yeast strains and two resource types was implemented in *McComedy*. Simulations were performed in a three-dimensional environment and yeast cells were initially distributed on a plane at the bottom. Process modules were selected to account for the production, release, diffusion, and consumption of resources, microbial growth, replication and mortality, and a weak gravitational force that kept the yeast cells at the bottom of the environment. Other resources than lysine and adenine were not explicitly modeled but assumed to be not limiting and constantly available for microbial uptake. This allows for the production of lysine or adenine, respectively, also for non-growing individuals. A detailed description of the model implementation is provided in the Methods section and a complete list of the selected process modules and parameter values is available in Section A.5.

The IBM created with *McComedy* succeeded in qualitatively reproducing the self-organized pattern observed in both the original IBM and the experimental setup. In the competitive scenario with additional adenine and lysine provided (+ LA), the microbial layer consisted of strongly separated yeast strains, which exhibited uniform thickness (Figure 7 E). In the scenario without additional resource providing (- LA), cooperating partners intermixed and formed thick layers, whereas non-cooperators were spatially excluded from the cooperative benefit and only formed thin layers (Figure 7 F). For a quantitative comparison, the two measures from the original study (i) association index and (ii) ratio of the abundances of $R_{\rightarrow A}^{\leftarrow L}$ and $C^{\leftarrow L}$ were calculated based on the new simulation results.

Both measures match very well between the two models. The association index increases in both cases initially, before plateauing after approximately four generations between values of 1.5 and 2 (Figure 8 A and Figure 8 B). The ratio of abundances of $R_{\rightarrow A}^{\leftarrow L}$ and $C^{\leftarrow L}$ increases in both models exponentially (Figure 8 C and Figure 8 D, note the logarithmic vertical axes). Both the original model and the new *McComedy* model fit the empirical evidence, as the experimental setup was evaluated once after six to eight generations, showing an increased intermixing of cooperators (Momeni *et al.* 2013b).

According to Momeni *et al.* (2013b), the intermixing of two cooperative strains depends on the amount of the essential resources that is exchanged between strains. This means that if a cooperative strain reduces the release of the shared resource, it will also intermix less with its cooperation partners and, thus, be inferior to another, more cooperative strain, even though it saves some of the cost for producing the cooperative benefit (Momeni *et al.* 2013b). This finding raises a follow-up question: How would the system behave if both genotypes $R_{\rightarrow A}^{\leftarrow L}$ and $G_{\rightarrow L}^{\leftarrow A}$ would simultaneously exhibit an increased or reduced cooperativity? Using *McComedy*, we examined this situation. A reduced overall cooperativity in terms of resource release by both cooperative strains led to an increased intermixing and relative abundance of cooperators (Figure 9), which might seem counterintuitive. However, a reduced resource release results in less resources that diffuse in the environment. Thus, resources are mostly available in short distances to the respective producing (cooperative) individuals, which leads to a stronger localization of cooperative benefit, thus favoring intermixing as discussed by Momeni *et al.* (2013b). Note that this strong spatial intermixing due to reduced cooperation coincided with considerably slower growth of the entire population (i.e. longer generation times, Figure 9 C). For very low rates of resource release, cooperators were not able to sustain the whole population, resulting in extinction after few generations. Therefore, the question arises, whether the model system is evolutionary unstable, albeit robust against non-cooperators in the short term. This example shows that the model implemented with *McComedy* serves as a powerful tool to understand mechanisms that drive spatial self-organization of microorganisms and also hints to possible challenges when evolutionary dynamics are taken into account.

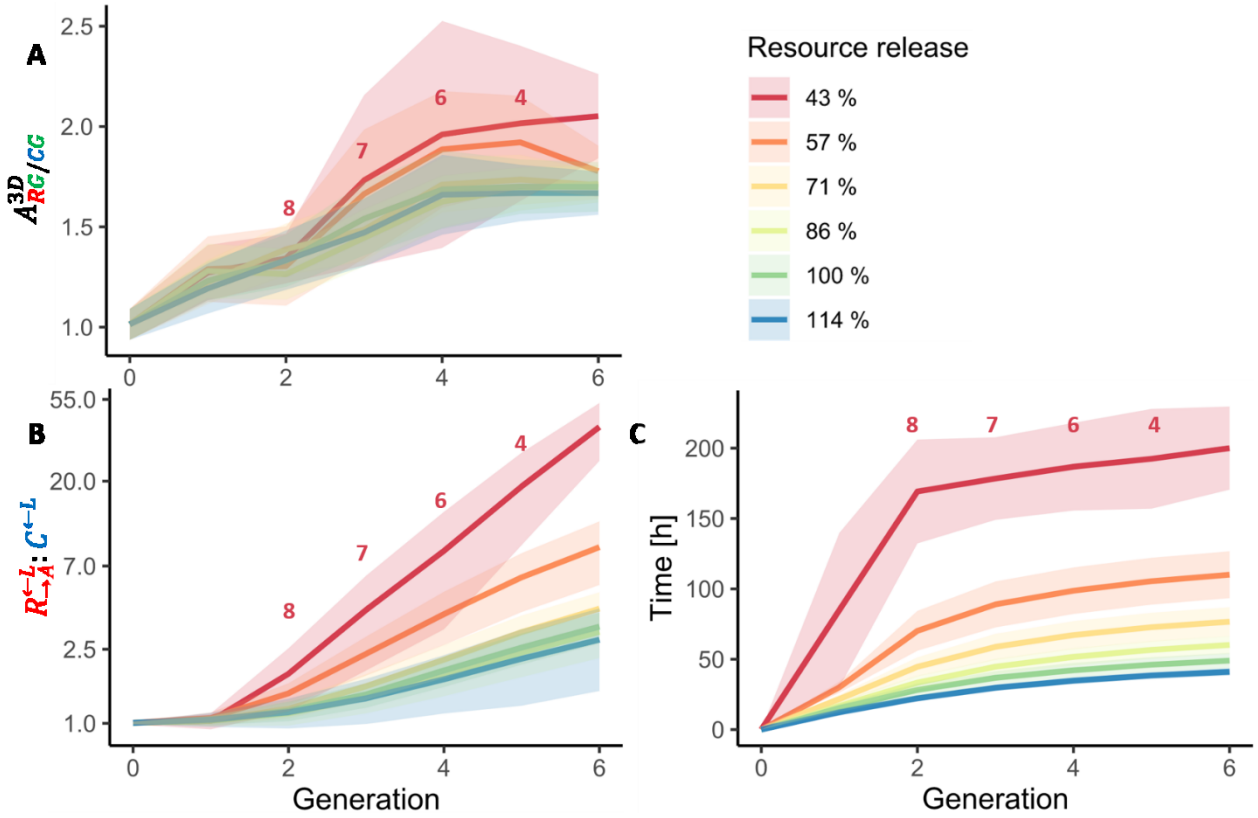

Figure 9. Reduced resource release rates increase abundance and intermixing of cooperators but also their generation time. Simulations were performed with McComedy with varied resource release rates and all other parameter values corresponding to scenario without supplemented lysine and adenine (- LA) in Fig 5. At low resource release rates, not all simulated communities achieved six generations. Numbers indicate how many of the initial 10 simulations contributed to the data visualized in the same color, starting from the respective X-position. Ribbons indicate the standard deviation. A: Association index of the two cooperative strains ($R^{\leftarrow L}_{\rightarrow A}$ with $G^{\leftarrow A}_{\rightarrow L}$) and the non-cooperators $C^{\leftarrow L}$. B: Abundance ratio between the cooperators $R^{\leftarrow L}_{\rightarrow A}$ and the non-cooperators $C^{\leftarrow L}$. C: Mean time until respective generation time is reached. One generation corresponds to the biomass doubling time of the simulated community.

3.3 Discussion

In this study, we introduced *McComedy*, a tool for developing and analyzing next-generation IBMs of microbial consumer-resource systems. The goal was to create a modeling framework that (i) allows to accurately capture the spatiotemporal dynamics of microbial communities as a consequence of mechanistically sound simulations of the relevant processes, (ii) is highly modularized to facilitate simulation of various distinct microbial systems by combining the relevant processes, and (iii) is user-friendly and accessible to researchers without profound programming knowledge. *McComedy* was successfully tested by reproducing the experimental and model results of two published studies that analyzed the spatiotemporal dynamics of microbial communities with consumer-resource interactions. In both cases, *McComedy* was additionally applied to investigate the studied systems beyond the scope of the original publications, providing additional insights that

complement the original studies. Thus, it was also demonstrated how *McComedy* can be flexibly adjusted to assess new scenarios.

We consider the good correspondence of our model results with previously published data a consequence of *McComedy*'s generic process modules that were combined and interact in the specific IBMs. Such simulation of standardized low-level processes and their interplay favors the emergence of structural realism, which means that a model largely captures the functional and organizational structure of a system (Grimm and Berger 2016). As the two IBMs that we implemented build on similar sets of process modules to be combined (Table 24) and adhere to *McComedy*'s fundamental design concepts and assumptions (e.g. continuous spatial positions of the microorganism individuals), they are very similar with respect to the model structure and implementation details. In contrast, the two IBMs provided by Mitri *et al.* (2015) and Momeni *et al.* (2013b) are designed quite differently. For example, the former model is based on explicit individuals representing microorganisms, whereas the latter model is based on local densities of microorganisms in discrete spatial grid cells. While both models accurately represent the respective microbial system, an attempt to compare or synthesize the results of the two studies would be hampered by the different designs used. This raises the question of how much of the different observations can be attributed to distinct ecological processes and how much is a consequence of the different design choices. By simulating both scenarios with *McComedy*, we firstly corroborate the generality of the respective findings and secondly demonstrate how the approach of building IBMs from generic process modules contributes to a coherent understanding of different ecological processes.

In terms of providing a framework for IBMs of microbial communities, *McComedy* is not the first of its kind. There are several other prominent and highly useful examples like Simbiotics (Naylor *et al.* 2017), iDynoMiCS (Lardon *et al.* 2011b), NUFEB (Li *et al.* 2019), COMETS (Harcombe *et al.* 2014), and Biocellion (Kang *et al.* 2014). Also NetLogo (Wilensky 1999) is a versatile and widely used framework for individual-based modeling of, for instance, microbial communities (van der Wal *et al.* 2013, Banitz *et al.* 2015). Although tailored for a broader community, these frameworks can be challenging and time-consuming to master for non-experts. Therefore, *McComedy* can be particularly useful to microbial ecologists and modelers who have little experience with programming and cannot invest much time into learning the specifics of other frameworks. This also offers the possibility of using *McComedy* for teaching purposes. The intuitive user interface and high flexibility allow an easy entry into individual-based modeling. In this context, student projects could for example constitute the reproduction of existing studies, as presented in this work.

With *McComedy*, the output data of the IBMs allows for sophisticated analysis of the simulated community dynamics across spatial scales and organizational levels. For example, simulation data

on the biomass of each individual microbe over time can be aggregated into population dynamics (Figure 8 C, Figure 8 D, Figure 9 B, and Figure 9 C) and data on the spatial position of each individual microbe can be used for spatial pattern analysis (Figure 5 D, Figure 6 B, Figure 8 A, Figure 8 B, and Figure 9 A). The possibility to analyze the microbial communities across spatial scales and organizational levels allows for pattern-oriented modeling, a technique where the model output is matched with as many different empirical patterns as possible to increase structural realism and reduce complexity (Grimm *et al.* 2005).

Current limitations for a broad application of *McComedy* for the modeling of various microbial systems are given by the set of available process modules. With the presented version of *McComedy*, we provide a library of selected process modules that allow for microbial community modeling with a focus on spatially explicit interactions and consumer-resource dynamics. At the current stage, *McComedy* facilitates modeling communities that consist of sessile microorganisms (e.g. in colonies and biofilms), planktonic individuals that move randomly in a liquid medium, and microbial aggregates suspended in liquid medium.

Process modules encompass diffusion and decay of resources, passive movement, attachment, and shoving of microbes, different initial microbe distributions, boundary conditions, a simple metabolism of microbes that optionally involves production and/or consumption of resources, and microbial growth and replication. However, there are additional processes that might be relevant in microbial communities but are not yet covered by the currently available catalogue of process modules. Therefore, we will continue the development of *McComedy* and provide more process modules in future versions that will allow for a wider range of microbial IBMs. For example, we are currently working on other forms of metabolic interactions such as direct resource exchange via nanotubes (Dubey and Ben-Yehuda 2011, Pande *et al.* 2015b) and evolutionary mechanisms such as mutation of microbial traits. Furthermore, future versions of *McComedy* will facilitate active microbial movement (e.g. based on chemotaxis) and negative metabolic interactions (e.g. release of growth-inhibitory by-products). Due to the free access to the code of *McComedy*, further process modules could be developed by other modelers too, if they are proficient with the programming language Java.

Another limitation of *McComedy* concerns the size of both the environment and the microbial communities that can be simulated. Although, technically, there are no hard limits for either of these, we recommend to not exceed community sizes of 2,000 individuals or environments of 50,000 grid cells to ensure reasonable computation times (see also computation times for representative simulations in A.2 Computational performance). This recommended scale allows to analyze the community dynamics at the level of individual cells, which is the intended use of *McComedy*. Larger-scale simulations, for example on the scale of an entire test tube, should be rather conducted on a more aggregated level using other modeling frameworks.

Based on the successful testing of *McComedy* with different studies from the literature and due to the simple and fast model creation, we conclude that *McComedy* is a promising tool for users who require next-generation IBMs of spatially explicit microbial consumer-resource systems. As shown in this work, the flexibility to model different systems does not come at the cost of accuracy because the system dynamics emerge from the mechanistic interplay of the generic process modules, close to what happens in the real systems. The development of *McComedy* goes on and we invite all researches from fields related to microbial ecology to try applying *McComedy* within their own projects.

3.4 Methods

3.4.1 Implementation of McComedy

In this section, an overview over the implementation of *McComedy* is provided. For a complete description and implementation details, refer to the ODD (overview, design concepts and details) protocol for standardized descriptions of individual-based models (Grimm *et al.* 2020, Grimm *et al.* 2006) (Section 2.2). *McComedy* is implemented in Java 11 using the JavaFX library for the graphical user interface. The process modules, each represented by one Java class, have been tested with the Junit unit-testing framework. *McComedy* is open-source and the code can be viewed and downloaded from https://git.ufz.de/bogdanow/mccomedy.

A microbial community modeled with *McComedy* is represented by a spatially explicit three-dimensional environment in which resources have local concentrations in discrete grid cells and microbe entities with a spherical shape have continuous position coordinates. At each time step the state of the system is defined by a set of state variables that are attached to the environment and individual microbes. The list of state variables is provided Table 1.

Over simulated time, the state variables are subject to change by a set of process modules. The process modules are repeatedly executed at a specific frequency and simulate natural processes, which are assumed to shape the dynamics of the modeled system (Table 24). Please see *Submodels* of the ODD protocol (Section 2.2) for detailed descriptions. As the simulation runs, the state of the system (i.e. values of all state variables) is saved in model output files in predefined intervals. This enables analyses of the temporal progression of the modeled system.

Table 24. Process modules currently available in McComedy. The columns SOM (Spatial organization model, Mitri et al. 2015) and CM (Cooperation model, Momeni et al. 2013b) indicate with an 'X' which process modules were integrated in the corresponding McComedy models. A more detailed description of each process module is provided in the ODD protocol (Section 2.2).

Process module	Description	SOM	CM
Attachment	Upon physical contact, microbes can attach to each other.		
CellPartition	This module estimates the overlap of each microbial cell with resource grid cells. It is required by some other processes and enhances the computation time of the simulation.	X	X
ChangeGenotype	Microbes change their type with a predefined probability.		X
ConstantProduction	Microbes produce a resource at a predefined constant rate. The resource remains in an intracellular pool.		X
ConstantResourceBoundaries	Resource concentrations are held constant at the boundaries of the environment (opposed to default periodic boundary conditions).	X	X
Diffusion	Resources diffuse through the environment at a predefined rate.	X	X
Flow	Microbes move in a predefined direction. This does not affect resources.		X
Growth	Consumed resources that have been allocated for growth are transformed into biomass with a predefined yield.	X	X
ImpermeableMicrobeBoundaries	Microbes cannot penetrate the boundaries of the system (opposed to default periodic boundary conditions).		X
InitBiofilm	Upon simulation start, microbes are placed on a two-dimensional plane at the bottom of the simulated environment.		X
InitCluster	Upon simulation start, microbes are distributed within a sphere at the center of the simulated environment.	X	
InitModel	This is the only obligatory process module. It attaches the initial resources and microbes to the environment and sets the initial values of the state variables.	X	X
LocalSource	Resource concentrations are increased or reduced at one or multiple locations according to a predefined rate.		
Lysis	Microbes are removed from the environment at a given probability.		
PassiveRelease	Microbes release produced intracellular resources into resource grid cells that the microbes overlap with.		X
PassiveUptake	Microbes consume resources from grid cells that the microbes overlap with, according to Monod-kinetics.	X	X
ProximityManager	This module groups microbes that are close to each other in order to boost searching algorithms. It is required by some other processes and enhances the computation time of the simulation.	X	X
Replication	When a microbe's biomass exceeds a predefined value, it is divided into two individuals.	X	X
ResourceDecay	At all grid positions, the concentration of resources decays at predefined rates.		
Shoving	Microbes that overlap spatially push each other away.	X	X
Starving	Microbes that are marked as starving are removed from the environment.		
SubstrateUtilization	Intracellular resources are reduced at a predefined rate to account for maintenance costs. The remainder is allocated to biomass growth. If the maintenance cost exceeds the amount of intracellular resources, the microbe is marked as starving.	X	X

3.4.2 Spatial organization model (Mitri *et al.* 2015)

Corresponding to the original model by Mitri *et al.* (Mitri *et al.* 2015), the system was simulated in an approximately two-dimensional environment of size 250 µm x 250 µm x 1 µm. The model was initialized with one homogeneously distributed resource *R* and two types of microbes, *M1* and *M2*, both of which could consume *R* and were also identical in all other respects except for the name and color. From each type, 100 microbes were randomly placed in a cluster at the center of the simulated environment (process module *InitCluster*).

The process modules *PassiveUptake*, *SubstrateUtilization*, and *Growth* were integrated into the model to account for resource uptake and biomass growth according to Monod-kinetics (Monod 1949). Microbes were assumed to divide into two individuals upon exceeding a critical biomass (process module *Replication*). Mechanical interaction between microbes (i.e. pushing each other away when overlapping spatially) was simulated by the process module *Shoving* according to the algorithm described in (Lardon *et al.* 2011b). The process module *Diffusion* was used to simulate resource diffusion throughout the environment. The 'agar plate' that contained the resources and on which the microbes grew was assumed to extend far beyond the simulation boundaries. Therefore, resource concentrations at the boundaries of the simulated environment were maintained at the initial resource concentration to account for diffusion into the simulated system (process module *ConstantResourceBoundaries*).

Across different simulations, the parameter values for the initial resource concentration and diffusion constant were varied. Five replicates for each variant of parametrization were simulated, each of which differed in the initial distribution of microbes due to different random generator seeds. However, the replicates for different variants of parametrization were initialized and simulated with the same random generator seeds. The generic units of *McComedy* for time T, distance S, resource mass M, and microbial dry mass M* were treated as seconds, micrometers, femtograms, and femtograms (dry weight), respectively. A complete list of model parameters and their values is provided in Section A.5.

Simulations were set to run for a maximum of 100 hours or until the community reached a total abundance of 20,000 microbes. The statistical analysis was conducted at the time point, at which the first simulation stopped, which was the case after 45 simulated hours, when initial concentrations were varied and after 39 hours, when the diffusion constant was varied.

Top-views on colonies (Figure 5 C and Figure 6 A) were rendered with *McComedy*. The quantitative analysis of the heterozygosity and the demixing distances was performed according to Mitri *et al.* (Mitri *et al.* 2015). The heterozygosity was calculated by sampling boxes of 5 µm x 5 µm along transects from the initial inoculum to the edge of the colony and counting individuals of

M1 and *M2* in each box. The heterozygosity as a function of distance from the inoculum is given by

$$H(x) = \frac{2}{\Phi}\sum_{\varphi}^{\Phi} f_1(x,\varphi)(1 - f_1(x,\varphi)),$$

where $f_1(x,\varphi)$ is the proportion of microbes *M1* at distance x from the inoculum location in transect φ and Φ is the number of transects. The demixing distance is defined as the point where $\frac{dH}{dx}$, the derivative of the heterozygosity function, is minimal.

3.4.3 Cooperation model (Momeni *et al.* 2013b)

The microbial system was simulated in a three-dimensional environment of size 480 µm x 100 µm x 240 µm. Two resources, *L* and *A* (representing lysine and adenine, respectively) and three types of microbes, $R_{\rightarrow A}^{\leftarrow L}$, $G_{\rightarrow L}^{\leftarrow A}$, and $C^{\leftarrow L}$, were added to the environment. The model was initialized with either empty resource grid cells for the scenario in which lysine and adenine were not provided (*– LA*) or with inexhaustibly high resource concentrations (i.e. 9999999 fmole/125 µm³) for the scenario in which lysine and adenine were provided (*+ LA*). Initially, 115 microbes of each type were randomly distributed on a two-dimensional plane (orthogonal to the Y-axis) close to the bottom of the simulated environment (process module *InitBiofilm*). This plane represented the surface of the agar, on which microbes were growing. Note that in the original study (Momeni *et al.* 2013b), vertical positions are described by Z-coordinates, whereas in *McComedy*, vertical positions are described by Y-coordinates.

Microbes were restricted from movement below the surface of the agar by the process module *ImpermeableMicrobeBoundaries* and a weak gravitational force was simulated by moving the microbes towards the agar surface with the process module *Flow*. As in the previous example, resource consumption and metabolism were modeled with the process modules *PassiveUptake*, *SubstrateUtilization*, and *Growth.* Additionally, resource overproduction and release were integrated with the process modules *ConstantProduction*, *PassiveRelease, and ChangeGenotype*. To account for mortality, the strain R, which constantly produced adenine, changed with a low probability to a metabolically inactive type. Strain G, which released lysine only upon cell death, was modeled such that it did not produce lysine when active. In the case of mortality (also occurring with a low probability) it first changed to a temporary type that produced a high amount of lysine and after one more time step to a metabolically inactive type. The process module *Diffusion* was used to simulate resource diffusion throughout the environment. The boundaries in X- and Z-direction were kept periodic (*McComedy* default) and resource concentrations at Y-boundaries were maintained at the initial concentration to simulate open boundaries (process module *ConstantResourceBoundaries*).

10 replicates were simulated for every variant of parametrization, each of which differed in the initial distribution of microbes due to different random generator seeds. However, the replicates for different variants of parametrization were initialized and simulated with the same random generator seeds. The generic units of *McComedy* for time T, distance S, resource mass M, and microbial dry mass M* were treated as seconds, 5 micrometers, femtomoles, and 10 picograms (dry weight), respectively. A complete list of model parameters and their values is provided in Section A.5.

The simulations ran until the community reached a total abundance of 22,080 microbes (i.e. 6 doublings of the initial 345 microbes) or until all microbes were dead. The analyses were performed at all time points at which the community size doubled (i.e. when the community abundance was closest to 345; 690; 1,380; 2,760; 5,520; 11,040; and 28,080 microbes, respectively).

Vertical cross-section views (Figure 7 E and Figure 7 F) were rendered with *McComedy*. The ratio between two types was calculated with respect to the biomass of each type. The association index is given by

$$A = \frac{\frac{1}{n_{R_{\to A}^{\leftarrow L}}} \sum_i^{n_{Y1}} a(R_{\to A i}^{\leftarrow L}, G_{\to L}^{\leftarrow A})}{\frac{1}{n_{C^{\leftarrow L}}} \sum_j^{n_{Y3}} a(C^{\leftarrow L}{}_j, G_{\to L}^{\leftarrow A})},$$

where $n_{R_{\to A}^{\leftarrow L}}$ is the number of microbes of type $R_{\to A}^{\leftarrow L}$ that are in proximity of at least one microbe of a different type and $a(R_{\to A i}^{\leftarrow L}, G_{\to L}^{\leftarrow A})$ is the number of microbes of type $G_{\to L}^{\leftarrow A}$ that are in proximity of the i-th microbe of type $R_{\to A}^{\leftarrow L}$ (microbes of type $R_{\to A}^{\leftarrow L}$ that have no microbes of a different type in their proximity are excluded). Here, '*in proximity*' means a maximum distance of 7.5 µm between the midpoints of the two microbes, which includes almost only directly adjacent microbes. The variables in the denominator are defined analogously.

3.4.4 Statistical analysis

Statistical analysis was conducted with R 4.0.3 (Team 2020). Plots were created with the package 'ggplot2' (Wickham 2016).

4 Model application: The cooperation paradox of diffusive goods[3]

4.1 Introduction

Bacteria provide crucial ecosystem services (Cavicchioli *et al.* 2019, Ducklow 2008) and play key roles in many applied contexts including health and disease (Costello *et al.* 2012), agriculture (Wakelin 2018), food processing (Min *et al.* 2019), or biofuel production (Kallio *et al.* 2014). Thus, a quantitative mechanistic understanding of the principles that govern the ecology and evolution of bacterial communities is essential to maintain and optimize these benefits. However, most bacterial systems are inherently complex, involving large numbers of taxonomically diverse cells that interact with each other and their environment in multifarious ways. In this context, metabolic interactions between bacterial cells appear to be particularly important, as indicated by the high abundance of auxotrophic genotypes (i.e. cells lacking the ability to synthetize essential metabolites) and the resulting obligate metabolic interdependencies between bacterial cells (Embree *et al.* 2015, Zengler and Zaramela 2018, Gorter *et al.* 2020, Johnson *et al.* 2020, Zelezniak *et al.* 2015).

The diversity of metabolic cross-feeding interactions that emerges within microbial communities is generally classified based on whether the production of the traded commodity incurs a cost to the producing individual or not (D'Souza *et al.* 2018a). If a shared metabolite is released as a by-product of an individual's metabolism (Fernandez-Veledo and Vendrell 2019, Koch *et al.* 2015) or simply as an unavoidable consequence of leakiness (Morris 2015), the interaction is termed a *by-product* interaction. In contrast, when an individual produces a metabolite to benefit another individual at a cost to itself and the interaction has evolved because of this reason, the interaction is termed *cooperative cross-feeding* (D'Souza *et al.* 2018a). Given this important distinction, the question arises whether metabolic cross-feeding in natural bacterial systems can be classified as truly cooperative or not (Oliveira *et al.* 2014). Unfortunately, however, it is generally difficult if not impossible to unambiguously decide whether a naturally evolved cross-feeding interaction is cooperative. This is because of problems to cultivate or genetically manipulate isolated strains. Moreover, the ecological conditions under which a given interaction evolved is frequently unknown, thus making it difficult to reenact the relevant circumstances under laboratory conditions. Most importantly, the ancestral strains from which the interaction evolved are usually not available for experimentation, thereby hampering a clear-cut assignment of a certain behavior as cooperative. As a consequence, evidence of cooperative cross-feeding is mostly limited to synthetic consortia of well-characterized laboratory strains that evolved a cooperative interaction

[3] Based on Bogdanowski A., Banitz T., Muhsal L. K., Kost C. and Frank K. 2022. Contact-dependent metabolite transfer can solve the cooperation paradox of diffusive goods (submitted).

under controlled laboratory conditions (e.g. Harcombe 2010, Preussger *et al.* 2020, Hillesland and Stahl 2010).

In order to assess whether cooperative cross-feeding can exist in natural communities, it is first of all necessary to identify the conditions under which this type of interaction can evolve. The main problem with explaining the emergence of cooperative behaviors is the fact that as long as both cooperating and non-cooperating individuals are equally likely to benefit from the cooperative act, non-cooperative strategies are favored (Williams 1966, Dawkins 1976, Rankin *et al.* 2007).

To overcome this problem, several mechanisms that facilitate the evolution of cooperation have been suggested (e.g. Hamilton 1964a, Wilson 1987, Doulcier *et al.* 2020). One such mechanism is *partner fidelity feedback* (Bull and Rice 1991, Sachs *et al.* 2004, Fletcher and Doebeli 2009), which favors cooperative interactions if partners interact reciprocally and repeatedly over extended periods of time. Such a situation can, for example, emerge in spatially structured environments (Nowak *et al.* 1994, Nowak and May 1992, Yamamura *et al.* 2004, Allison 2005, Stump *et al.* 2018b). Under these conditions, cooperators may by chance colocalize with other cooperators. Repeated interactions among cooperators will then strongly enhance their growth relative to interactions between cooperators and non-cooperators or pairs of non-cooperators. This results in a pattern of spatial self-organization that favors cooperative individuals and excludes non-cooperators from cooperative benefits (Momeni *et al.* 2013b, Germerodt *et al.* 2016). In bacteria, this mechanism has been demonstrated in surface-attached biofilms (Kreft 2004, Nadell *et al.* 2016), colonies on agar plates (Pande *et al.* 2016b), and in constructed microhabitats consisting of small habitat patches connected by corridors (Hol *et al.* 2013). However, spatial structure was also found to inhibit cooperation if interactions between cooperators and non-cooperators result in specific fitness pay-offs (Hauert and Doebeli 2004) or when local competition for resources outweighs the benefits of cooperation (Platt and Bever 2009, Griffin *et al.* 2004).

A frequently documented type of self-generated spatial structures is the formation of multicellular aggregates within bacterial populations (Trunk *et al.* 2018, Cai 2020). This raises the question whether such aggregates also promote the evolution of cooperative metabolic exchange. During an evolution experiment, in which auxotrophic strains of *Escherichia coli* that initially exchanged by-products have evolved cooperative cross-feeding, Preussger *et al.* (2020) observed a high prevalence of multicellular aggregates in cocultured bacteria. Based on these observation, the authors argued that the spatial structure provided by these aggregates played an essential role for the evolutionary outcome.

Here we use an individual-based model to examine whether multicellular bacterial aggregates can indeed promote the evolution of metabolic cooperation as has been previously reported for surface-attached biofilms and colonies (Kreft 2004, Nadell *et al.* 2016, Pande *et al.* 2016b). To do so, we devised an individual-based model to simulate the behavior of bacterial strains from the evolution

experiment of Preussger *et al.* (2020). These include non-cooperators that are either auxotrophic for tyrosine or tryptophan (here referred to as *NCΔY* and *NCΔW*, respectively) and the evolved cooperators that are also auxotrophic for one of these amino acids, but also overproduce the other amino acid (hereafter referred to as *CoopΔY* and *CoopΔW*, respectively). We investigate under which conditions the evolved cooperative bacteria are favored over their non-cooperative ancestors and whether the observed selection for an increased cooperativity can be reproduced in the simulations. Our results indicate that a diffusion-based exchange of metabolites within multicellular aggregates strongly selects against cooperators, thus preventing an increase in the cooperativity on a cluster-level. Furthermore, our findings suggest that other contact-dependent mechanisms of metabolite transfer, for example intercellular nanotubes (Dubey and Ben-Yehuda 2011, Pande *et al.* 2015a), have likely evolved as a key prerequisite for the emergence of cooperative cross-feeding in bacteria.

4.2 Results

Using the simulation platform *McComedy* (Bogdanowski *et al.* 2022), we devised a mechanistic and spatially explicit individual-based model (IBMs) of consortia that consist of the two cooperative strains, *CoopΔY* and *CoopΔW*, and one non-cooperative strain *NCΔY*. This composition allowed *CoopΔY* and *CoopΔW* to engage in cooperative cross-feeding by reciprocally exchanging the two amino acids tryptophan and tyrosine. The non-cooperative *NCΔY* also benefitted from the provided tyrosine, yet saved the costs for producing public goods (Figure 10 A). In accordance with the observations by Preussger *et al.* (2020), the consortia were modeled as three-dimensional multicellular aggregates.

To parametrize the model, growth kinetics of the strains described in the study by Preussger *et al.* (2020) were experimentally determined by providing the focal strains with the amino acid they required for growth. In addition, the maximum uptake rates of the amino acids, the degree of metabolite overproduction in cooperators, and the fitness cost for overproduction were quantified (Section 4.4.5).

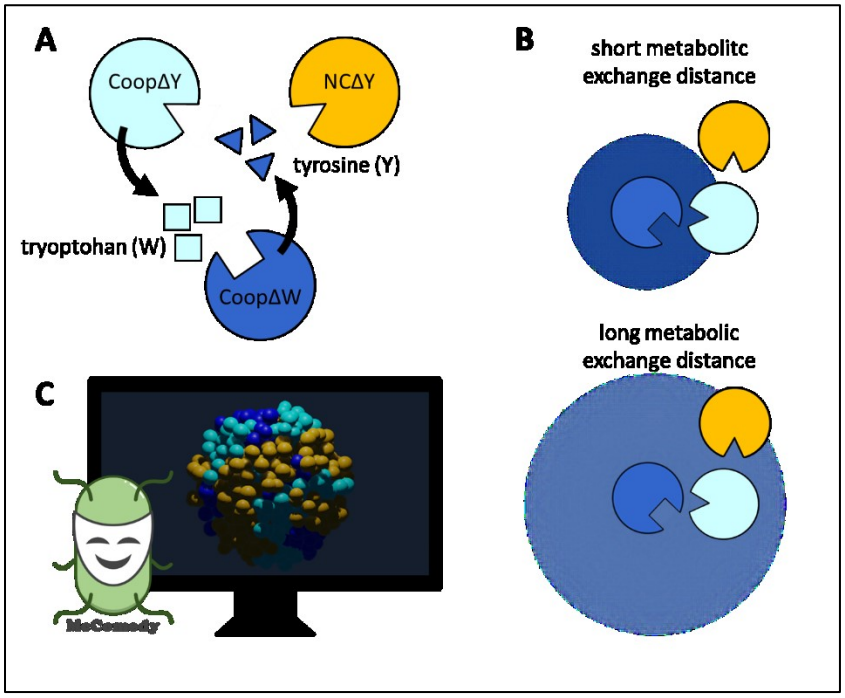

Figure 10. Spatially-explicit modeling of a bacterial consortia with cooperative cross-feeders and non-cooperative competitors. A: The modeled consortia consist of three different strains of E. coli exchanging essential amino acids: the cooperator CoopΔY (cyan, consuming tyrosine, overproducing tryptophan), the cooperator CoopΔW (blue, consuming tryptophan, overproducing tyrosine), and the non-cooperator NCΔY (orange, consuming tyrosine, no overproduction). B: The diffusion coefficient of metabolites determines its concentration gradient after release (here: tyrosine, visualized by blue color around the blue cell). This gradient translates into an exchange distance determining the access to metabolites for bacteria in the vicinity. Slightly remote bacteria (here: the orange cell) only have access to the metabolite when the exchange distance is sufficiently long (i.e. the diffusion coefficient is sufficiently high, bottom illustration). C: Bacterial consortia are simulated with McComedy, a tool for individual-based modeling of microbial consumer-resource systems (Bogdanowski et al. 2022). In the model simulations, the bacteria form three-dimensional multicellular aggregates.

4.2.1 High diffusivity selects against metabolic cooperation in multicellular clusters

First, the development of consortia of *CoopΔY*, *CoopΔW*, and *NCΔY* was simulated under varied diffusivities of both amino acids. Given that the diffusivity of released metabolites determined the availability of the released compounds to bacterial cells and thus also the distribution of the three focal strains in space (Figure 10 B), population dynamics were expected to change in response to varying the diffusion coefficient. Indeed, examining the selection coefficient for cooperation SC_{Coop} (Section 4.4.6) across different diffusion coefficients revealed an increased selection against cooperation with increased diffusivity (Figure 11 B). This pattern can be understood by analyzing the positions of individual bacteria from the three different strains and the spatial distribution of tyrosine (i.e. the amino acid that *CoopΔY* and *NCΔY* were competing for, Figure 11 A). At low diffusivity, the distribution of tyrosine was heterogeneous and increased concentrations in the vicinity of the overproducers (*CoopΔY*, cyan circles) were available to other cells. With high diffusivity, however, the metabolites spread out so rapidly that the concentration of tyrosine

appeared homogeneous. As a consequence, all bacteria had similar access to the metabolites and non-cooperators prevailed, because they saved the fitness cost for metabolite overproduction. Notably, actual diffusion coefficients of amino acids in aqueous solutions are reported to be between 500 and 1.000 $\mu m^2 \cdot s^{-1}$ (Ma *et al.* 2005) and simulations with a diffusion in this range favored exclusively non-cooperators. Only at much lower diffusion coefficients (i.e. 0.01 $\mu m^2 \cdot s^{-1}$) did selection favor cooperators over non-cooperators (Figure 11 B).

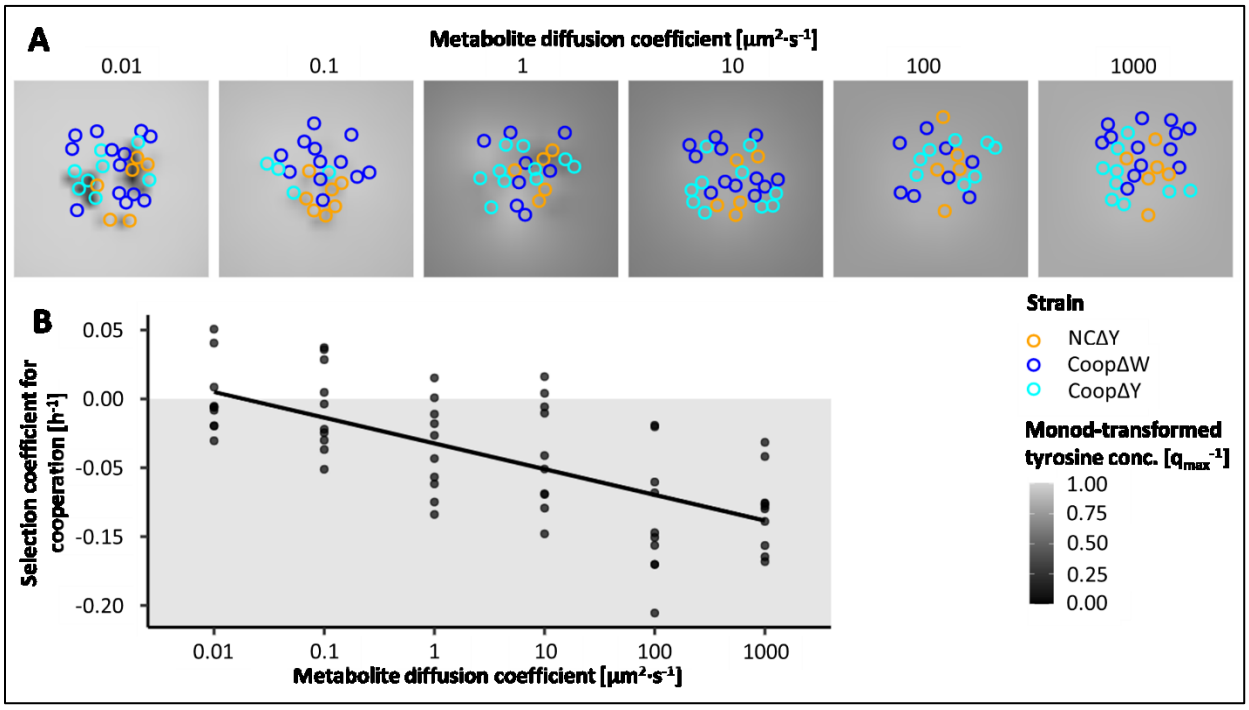

Figure 11. Selection for cooperation decreases with increasing diffusion coefficients. A: Example cross-sections of bacterial aggregates and distribution of tyrosine along X- and Y-axis at Z = 12.5 µm (midpoint of Z-axis) after two hours of growth at different diffusivity of metabolites. Orange circles represent the locations of non-cooperative bacteria. Blue and cyan circles represent the locations of cooperative bacteria. Different shadings of grey represent the local concentration of tyrosine. Concentration values are transformed into relative uptake rates according to Monod-dynamics. This value ranges from 0 (no uptake) to 1 (maximum possible uptake) B: Relationship between the selection coefficient for cooperation (SC_{COOP}) and metabolite diffusion. The grey area ($SC_{COOP} < 0$) indicates an overall advantage for non-cooperators. The line represents the general trend as log-linear regression. Each group consists of 10 replicates. Spearman's rank correlation: ρ = -0.69, P = 1.4e-9, n = 60.

4.2.2 The cooperation paradox of diffusive goods: cooperation is favored most strongly at lowest cooperativity

So far, we have found that pre-existing cooperators could prevail despite the presence of non-cooperating types. Our result showed that this is in principle possible, yet only when unreasonably low rates of metabolite diffusion are assumed. However, can mutualistic cooperation evolve under these conditions? Here we expected that the degree of cooperativity in cooperators that emerge from previously non-cooperative bacteria should slowly increase, because multiple mutations are necessary until an optimal level is achieved. Therefore, the previous simulations were repeated

with varied cooperativity values of *CoopΔY* and *CoopΔW*. In our model, the cooperativity value defined the overproduction rate of the released metabolite as well as the associated fitness cost. The results of this analysis clearly showed that cooperativity was negatively correlated with selection for cooperation and that this pattern was independent of the diffusion coefficient of the metabolites (Figure 12 A). Thus, strongest selection for cooperation occurred when cooperators were least cooperative. This counterintuitive observation, which we termed the 'cooperation paradox of diffusible goods' can be best understood by considering again the spatial distribution of individual bacterial cells and the traded metabolite (here: tyrosine, Figure 12 B). At low levels of cooperativity, small amounts of tyrosine were released and thus only available to other cells that were closely neighboring the producing bacteria. In contrast, when tyrosine was released in increased amounts (i.e. when cooperativity was high), metabolites could also reach more distant bacterial cells. Thus, increasing the degree of cooperativity had a similar effect to increasing the diffusion coefficient of the metabolites as both parameters determined the exchange distance of metabolites (Figure 10 B). To rule out that these effects were only due to the higher fitness cost of high cooperativity, additional test simulations were performed. This time, however, instead of varying the rate of overproduction of the traded metabolite and its fitness cost simultaneously (i.e. cooperativity), only the metabolite overproduction rate of the metabolites was varied and the fitness cost was kept constant. The results of these simulations revealed a qualitatively similar pattern as before (Figure 16), thus suggesting that indeed the increased release of metabolites undermined the selection for cooperation.

To examine these results analytically, we devised a simple mathematical model, in which a cooperator released a metabolite, which was taken up (according to Monod-dynamics) by two recipients at different distances. The mathematical model revealed that increasing the amount of metabolites produced by a cooperator led to higher relative benefits for a more distant recipient than for a closer localized one (Section A.3), thus corroborating our previous result that increased levels of a public good undermines the evolution of cooperation. By homogenizing the availability of the public good to both cooperators and non-cooperators, the otherwise cooperation-enhancing effect of spatial structure is lost.

Our previous results suggested that, independent of the diffusivity of the exchanged metabolites, low levels of cooperativity are favored over higher ones (Figure 12 A). This observation complicates the evolution of cooperation in initially non-cooperative bacteria, which requires conditions in which an increased cooperativity is favored. To test this directly, we devised a simulated evolution experiment, in which consortia of poorly cooperating auxotrophs were simulated for 20 days and in which the cooperativity value of individual bacterial cells occasionally changed due to rare mutational events (Section 4.4.3). A diffusion coefficient of 0.01 $\mu m^2 \cdot s^{-1}$ was set in order to initialize the experiment at conditions that slightly favor cooperators. The general trend of the evolved degree of cooperativity in the consortia corresponded well with the previous

result (Figure 12 C). Over the simulated period of time, the cooperativity of cross-feeding bacteria did not increase, but rather decreased slightly. We therefore concluded that the evolution of cooperative cross-feeding is hampered by diffusion, even if the diffusivity is unreasonably low.

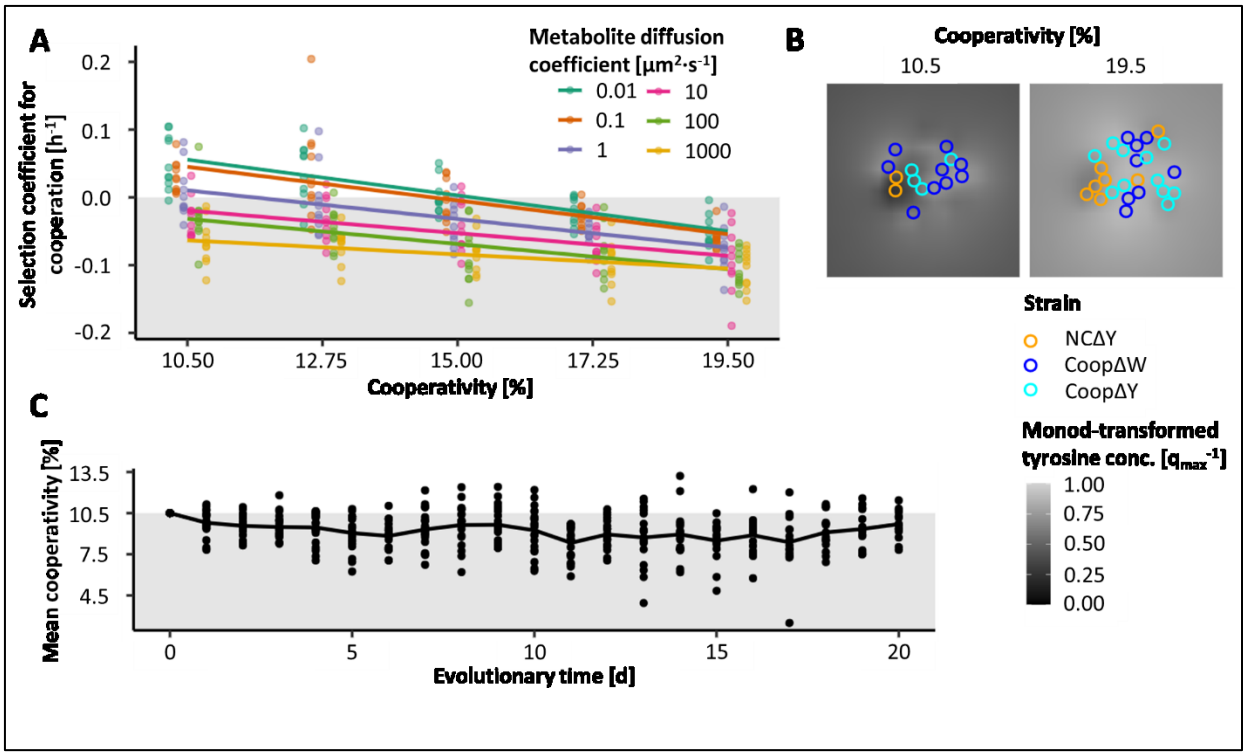

Figure 12. Selection favors reduced cooperativity if metabolites are exchanged via diffusion. A: Selection coefficient for cooperation (SC_{COOP}) against cooperativity is plotted for different metabolite diffusion coefficients (different colors). Cooperativity is a measure of metabolite overproduction and the associated fitness costs. 15 % corresponds to the empirical data (Section 4.4.3). The grey area ($SC_{COOP} < 0$) indicates an overall advantage for non-cooperators. The lines represent the general trend as obtained from multiple linear regressions. Each group (i.e. combination of cooperativity and diffusion coefficient) consists of 10 replicates. Pearson's moment correlation for each metabolite diffusion coefficient D [$\mu m^2 \cdot s^{-1}$]: D = 0.01: r = -0.77, P = 8.8e-11, n = 50; D = 0.1: r = -0.68, P = 5.1e-8, n = 50; D = 1: r = -0.65, P = 3e-7, n = 50; D = 10: r = -0.55, P = 3.1e-5, n = 50; D = 100: r = -0.58, P = 9.1e-6, n = 50; D = 1000: r = -0.46, P = 7.1e-4, n = 50. B: Example cross-sections of bacterial aggregates and distribution of tyrosine along X- and Y-axis at Z = 12.5 µm (midpoint of Z-axis) after two hours of growth at different cooperativity values. The metabolite diffusion coefficient was set to 1 $\mu m^2 \cdot s^{-1}$. Orange circles represent the locations of non-cooperative bacteria. Blue and cyan circles represent the locations of cooperative bacteria. Different shades of grey represent the local concentration of tyrosine. The concentration values are transformed into relative uptake rates according to Monod-dynamics and range from 0 (no uptake) to 1 (maximum possible uptake) C: Mean cooperativity value from each replicate in the simulated evolution experiment over time (started with 20 replicates). The metabolite diffusion coefficient was set to 0.01 $\mu m^2 \cdot s^{-1}$. All bacterial cells were initialized with 10.5 % cooperativity at day 0. The grey area indicates cooperativity below the initial value. The black line indicates the mean at each time point.

4.2.3 Contact-dependent metabolite transfer can solve the cooperation paradox of diffusive goods

Given that diffusion of metabolites through the environment hinders the evolution of cooperative cross-feeding, how can mutualistic cooperation evolve that is based on a reciprocal exchange of

diffusible metabolites? One solution could be to exchange metabolites in a contact-dependent manner. This type of mechanism includes, for example, an exchange of metabolites via intercellular nanotubes (Dubey and Ben-Yehuda 2011, Pande *et al.* 2015a). Nanotubes are tubular structures that consist of membrane-derived lipids and which allow for an exchange of cytoplasmic materials between interconnected cells. Hence, the distance over which metabolites can be exchanged is determined by the length of nanotubes and thus independent of the diffusivity of the shared metabolites. Using the simulation platform *McComedy*, we tested whether a transfer of metabolites via intercellular nanotubes could facilitate the maintenance and evolution of cooperative cross-feeding and thus solve the previously described cooperation paradox of diffusive goods.

As before, multicellular aggregates of the strains *CoopΔY*, *CoopΔW*, and *NCΔY* were simulated. This time, however, the maximum length of the nanotubes (analogously to the diffusion coefficient as both parameters determine the metabolic exchange distance) and the cooperativity value (Figure 13 A) was varied. The results of these simulations revealed that short nanotubes (≤ 4 µm) selected for cooperation. When nanotubes were very short (≤ 2 µm), selection for cooperation slightly increased with increased cooperativity (i.e. with increased fitness cost), albeit this increase was not significant (Pearson's moment correlation $r = 0.22$ $P = 0.12$, $n = 50$). This finding suggested that a contact-dependent exchange of metabolites can facilitate the evolution of cooperative cross-feeding.

To test this hypothesis more directly, the simulated evolution experiment was repeated, yet with a nanotube-mediated instead of a diffusion-based exchange of metabolites. In contrast to the previous results (Figure 12 C), a significant increase of the mean cooperativity in the bacterial consortia over time was observed (Figure 13 C). After approximately 5 days, the mean cooperativity plateaued at a value of approximately 15 %, which coincided with the cooperativity value that was estimated from the experimental growth kinetics of the modeled strains (Section 4.4.5). These results strongly indicate that a contact-dependent metabolic exchange, for example, via intercellular nanotubes, can facilitate the evolution of cooperative cross-feeding under more realistic assumptions than was previously observed for a diffusion-based exchange of metabolites.

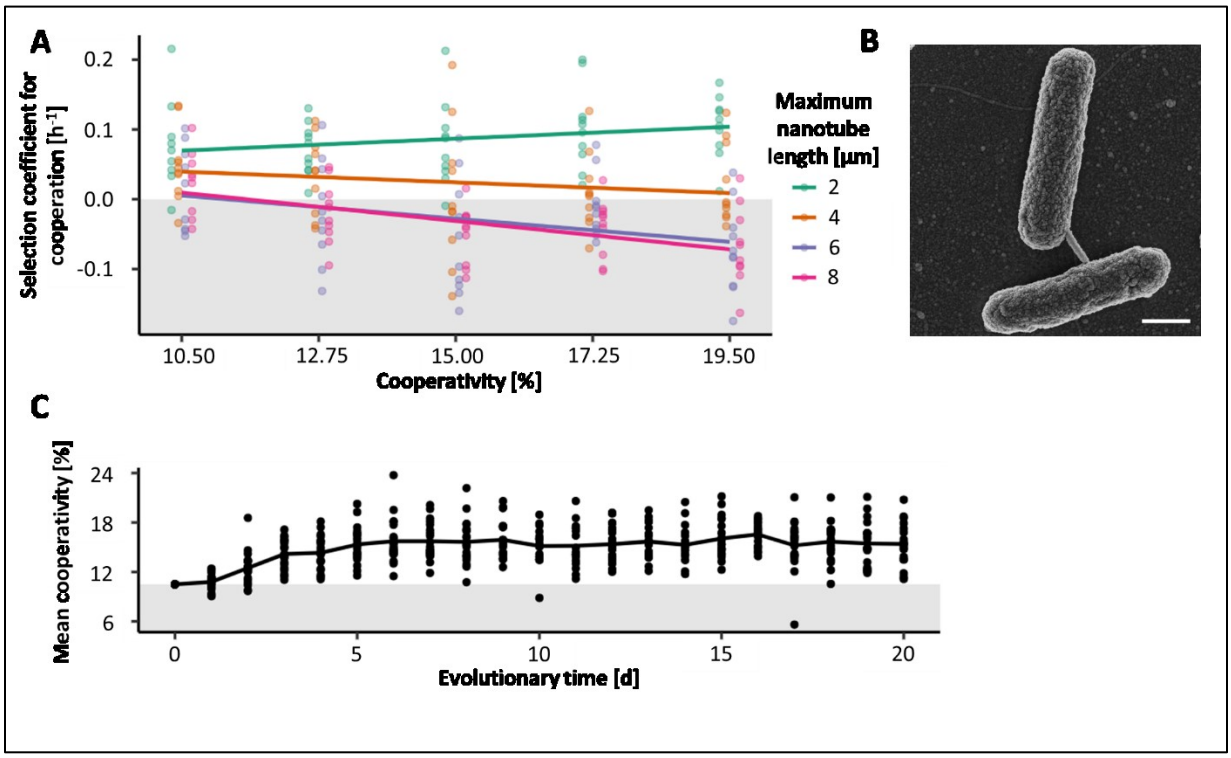

Figure 13. Selection favors increased cooperativity if metabolites are exchanged via short nanotubes (≤ 2 µm). A: Selection coefficient for cooperation (SC_{COOP}) against cooperativity plotted for different maximum nanotube lengths (different colors). Cooperativity is a measure of metabolite overproduction and associated fitness costs. 15 % corresponds to the empirical data (Section 4.4.3). The grey area ($SC_{COOP} < 0$) indicates an overall advantage for non-cooperators. The lines represent the general trend as obtained from multiple linear regression. Each group (i.e. combination of cooperativity and maximum nanotube lengths) consists of 10 replicates. Pearson's moment correlation for each maximum nanotube length L [µm]: L = 2: r = 0.22 P = 0.12 (not significant), n = 50; L = 4: r = -0.17, P = 0.25 (not significant), n = 50; L = 6: r = -0.33, P = 0.02, n = 50; L = 8: r = -0.53, P = 8.8e-5, n = 50. B: Scanning electron microscopy image of nanotubes in a coculture of synethtically engineered cooperative strains of E. coli ΔtrpBΔhisL and E. coli ΔhisDΔtrpR. Scale bar: 0.5 µm. C: Mean cooperativity value from each replicate in the simulated evolution experiment over time (initiated with 20 replicates). The maximum nanotube length was set to 2 µm. All bacterial cells were initialized with 10.5 % cooperativity at day 0. The grey area indicates cooperativity below the initial value. The black line indicates the mean at each time point.

4.3 Discussion

We investigated if and how cooperative cross-feeding of diffusive metabolites can evolve in multicellular bacterial aggregates using an individual-based model. Our results demonstrate that a diffusion-based transfer of essential metabolites between bacterial cells hinders the evolution of cooperation. In contrast, a direct transfer of metabolites between cells in a contact-dependent manner, for example via intercellular nanotubes, favored the evolution and long-term persistence of cooperation.

Why does a nanotube-based metabolic exchange support the evolution of cooperation in multicellular aggregates, while metabolic exchange via diffusion does not? Our results indicate that

the diffusion coefficient of metabolites that are shared between cells is required to be extremely low in order to favor cooperating over non-cooperating genotypes. In fact, cooperation was only selected for when diffusion coefficients were assumed that were orders of magnitude below the actual diffusion coefficient of amino acids in aqueous solutions. Only under those conditions was the concentration of released metabolites higher in the immediate vicinity of cooperators, thus, strongly benefitting pairs of randomly colocalized cooperators. This finding is in line with other studies demonstrating that in spatially structured environments cooperation is only favored over non-cooperation when the diffusivity of the exchanged public good is sufficiently low (Allen *et al.* 2013, Borenstein *et al.* 2013, Allison 2005, Germerodt *et al.* 2016, Dobay *et al.* 2014). Empirical support of this notion can be found in experiments with *Pseudomonas aeruginosa* in media with different degrees of viscosity, in which cooperation was favored at high viscosity (and hence low diffusivity) (Kümmerli *et al.* 2009). At a short time scale that allows to neglect evolutionary change in the degree of cooperativity, our results are in agreement with this observation. However, this observation should be seen as a snapshot during a long evolutionary process. When a longer time frame is considered and therefore also evolutionary adaptations of cooperativity are taken into account, natural selection is not favoring an overproduction of metabolites (i.e. increased cooperativity), as an increased release of diffusible metabolites has a similar effect to increased diffusivity.

On the other hand, if metabolites are exchanged in a contact-dependent manner via for example nanotubes, only connected cells receive the benefit. This appears to be a key advantage compared to a diffusion-based exchange as it makes the metabolites unavailable to cells outside of the cooperators' vicinity. In such a scenario, non-cooperators may still gain access to the metabolites they require, because they can also be linked to cooperators via nanotubes. Therefore, when nanotubes were allowed to be rather long, non-cooperators could persist and were also competitively superior to cooperators, thus selecting against cooperation. However, with shorter nanotubes, the exchange involved exclusively partners in close vicinity, leading to the process of spatial self-organization as described above. In this case, however, increasing cooperativity by sharing a higher amount of metabolites was profitable for the cooperators as the benefit was received exclusively by the immediate interaction partners which, after spatial self-organization commenced, were predominantly cooperative. Thus, the degree of cooperativity that emerged in our simulations with shorter nanotubes was much higher than the one that was possible when metabolites were exchanged by diffusion through the extracellular environment – even if a favorable diffusivity was assumed.

Our results raise the question whether the bacteria in the evolution experiment of Preussger *et al.* (2020), which served as the biological model system for the here presented simulations, necessarily needed to exchange metabolites in a contact-dependent manner in order to evolve cooperation. In fact, Preussger *et al.* (2020) have discussed intercellular nanotubes as a possible explanation for

the formation of multicellular aggregates during their evolution experiment, which was corroborated by the detection of cells that had exchanged cytoplasmic material. It is therefore likely that nanotubes have indeed played a significant role for the evolution of cooperation in that system. However, other alternative explanations may also account for the evolution of cooperation within multicellular aggregates when the exchange of metabolites between cells relies on a diffusion through the extracellular environment. For example, bacteria can actively prevent diffusible metabolites from spreading by secreting extracellular substances that form impenetrable barriers, as has been shown in experiments with *Vibrio cholera* (Drescher *et al.* 2014). Another strategy is to ensure that the majority of diffusing metabolites is consumed before they can reach other cells, which can be achieved for example by dense cell packing and high uptake rates (Dal Co *et al.* 2020). Furthermore, if growing bacterial aggregates form new aggregates via, for example, budding or an aggregation-disaggregation process (Pande and Kost 2017), multi-level selection could promote the best growing aggregates, which likely contain a higher proportion of cooperative individuals (Doulcier *et al.* 2020, Kingma *et al.* 2014, Gardner 2015, Wilson 1987). New bacterial aggregates could be formed either passively, for example by shearing forces (Uppal and Vural 2018) or actively by weakening enzymatically the cohesion of multicellular aggregates as during dispersion of biofilms (Petrova and Sauer 2016, Kaplan 2010).

In this work, we focused our attention on nanotubes as one example for a contact-dependent metabolic interactions. Whether or not nanotubes are commonly used by bacteria to exchange cellular material is debated (Pospisil *et al.* 2020). However, our findings should also apply to other contact-dependent mechanisms that are used by bacteria to exchange metabolite between different bacterial cells. For instance, bacteria in synthetic communities consisting of *Clostridium acetobutylicum* and *Desulfovibrio vulgaris* established tight cell-to-cell interactions that allowed for a direct transfer of metabolites between these strains (Benomar *et al.* 2015). Another example is *Pelotomaculum thermopropionicum*, which was shown to use flagellum-like filaments to exchange hydrogen with *Methanothermobacter thermautotrophicus* (Ishii *et al.* 2005, Shimoyama *et al.* 2009). Material exchange was also observed after membrane fusion between cells of *Myxococcus xanthus* (Ducret *et al.* 2013). All of these mechanisms could help to promote the evolution of cooperative cross-feeding.

Taken together, this work sheds light on the evolutionary dynamics of cooperative metabolic exchange in bacterial multicellular aggregates. We found that the spatial structure of such aggregates can promote the evolution of cooperation, however, this depends on the mechanism by which the metabolites are transferred between cells. If exchange is carried out vial diffusion through extracellular space, cooperative cross-feeding cannot evolve unless additional mechanisms ensure that the metabolites cannot spread. On the opposite, contact dependent means of metabolite transfer, such as via intercellular nanotubes, allow for such evolution. It is therefore demonstrated that in bacterial ecological systems, two cell-level mechanisms with a similar function but

differences in realization can have fundamentally different consequences for the entire system. For researchers working with simulation models and complex ecological patterns, this may be encouragement to pay attention to biological details.

4.4 Methods

4.4.1 Growth experiments

The strains *CoopΔY*, *CoopΔW*, *NCΔY*, and *NCΔW* were the *E. coli* strains *Av5b*, *Av5r*, Δ*tyrA*, and Δ*trpB* from Preussger *et al.* (2020), respectively. Cultures were grown in minimal medium for Azospirillium brasilense (MMAB) without biotin using 0.5 % glucose instead of malate as a carbon source. Single strains were grown overnight supplemented with the amino acid they require for growth and washed twice in MMAB without amino acid supplementation. Afterwards, strains were inoculated with an initial optical density at 600 nm (OD_{600nm}) of 0.005 in 96-well plates and incubated at 30 °C under shaking conditions in the SpectraMax microplate reader (Molecular Devices). The OD_{600nm} was measured every 30 minutes for 72 hours. In this experiment, MMAB was supplemented with 100 µM tyrosine when growing monocultures of *CoopΔY* and *NCΔY* or with 100 µM tryptophan when growing monocultures of *EVOΔW* and *NCΔW*. Cocultures (of *NCΔY* + *NCΔW* and *CoopΔY* + *CoopΔW*) were grown without amino acid supplementation and both partners were inoculated in a 1:1 ratio.

4.4.2 Estimating growth rates

To estimate growth rates from experimental data, only OD_{600nm} values measured during the exponential growth phase were considered. The exponential growth phase was assumed to start immediately as no considerable lag-phase was observed and to last until the OD_{600nm} values reached the half-maximal OD_{600nm} value for the first time. The exponential growth rate in each replicate was estimated by fitting a linear model to the log transformed OD_{600nm} values and time.

4.4.3 Individual-based modeling

The individual-based model was created and executed with the modeling platform *McComedy* (**Mic**robial **co**mmunities, **me**tabolism, and **dy**namics) (Bogdanowski *et al.* 2022). In *McComedy*, bacterial cells are simulated in a spatially explicit, three-dimensional environment. Metabolites are represented via three-dimensional grids of local concentrations. Changes in bacterial cells (e.g. position and biomass) and metabolite concentrations are driven by various process modules (e.g. diffusion of metabolites, growth of microbial cells) that can be combined in different ways. A full description of *McComedy* following the standard ODD protocol (Grimm *et al.* 2006, Grimm *et al.* 2020) is provided in Section 2.2.

In each simulation, one microbial aggregate was modeled in a three-dimensional environment of size 25 µm x 25 µm x 25 µm. The model was initialized with 100 bacterial cells (either 50 *CoopΔY*

and 50 *CoopΔW* or 40 *CoopΔY*, *40 CoopΔW*, and 20 *NCΔY*). Initially, bacterial cells were randomly distributed in an aggregate in the center of the respective environment. Bacterial cells consumed either tryptophan or tyrosine, while all other essential metabolites were assumed to be consumed implicitly according to the respective demands without limiting its availability in the environment. The consumed metabolites were transformed into biomass according to a predefined yield, which was estimated to be 63.6 for tyrosine and 136.3 for tryptophan (Section 4.4.5). Cooperative strains overproduced either tryptophan or tyrosine at a rate that was dependent on the individual cooperativity and a predefined maximum overproduction rate. The latter defined how much amino acids could theoretically be overproduced if a bacterium would invest all of its resources into overproduction (and no resources into biomass growth). Based on this, the cooperativity value defined what percentage of the maximum overproduction rate was realized and how much of the biomass yield was sacrificed for this overproduction. If not varied, a cooperativity value of 15 % was used as it was experimentally determined in both *CoopΔY* and *CoopΔW* (Section 4.4.5). In simulations with a diffusion-based metabolite exchange, all overproduced metabolites were released into the extracellular environment and diffused according to Fick's second law. For numerical realization, the finite difference method was applied in three dimensions as described by Mugler and Scott (1988) for two dimensions, which results in

$$\Delta C_{x,y,z} = D\frac{dt}{dS^2}(C_{x+dS,y,z} + C_{x-dS,y,z} + C_{x,y+dS,z} + C_{x,y-dS,z} + C_{x,y,z+dS} + C_{x,y,z-dS} - 6\,C_{x,y,z})$$

, where $\Delta C_{x,y,z}$ is the difference in metabolite concentration in position X, Y, Z due to diffusion, D is the diffusion coefficient, dt is the time step length of the diffusion process and dS is the distance between the midpoints of two adjacent grid cells. To ensure numerical stability, dt was kept below $\frac{1}{6}D$. To simulate open boundaries, the metabolite concentration at the grid boundaries was extrapolated from the concentration in the interior of the environment assuming exponential decay over space. Uptake of amino acids in the environment was modeled according to Monod dynamics:

$$q(C) = q_{max}\frac{C}{K_m + C}$$

, where $q(C)$ is the uptake rate at concentration C, q_{max} is the highest possible uptake rate, and K_m is the half-saturation constant which satisfies $q(K_m) = \frac{q_{max}}{2}$.

In simulations with metabolite exchange via nanotubes, overproduced metabolites were directly transferred to other bacterial cells that were connected via nanotubes. In these cases, no metabolites were released to the environment. The transfer of amino acids occurred independently of the recipient's cooperativity or ability to utilize them. The formation of nanotubes was implemented according to following rules: (1) each time step, bacterial cells without nanotubes could form a maximum of one nanotube to connect to another cell. (2) Bacterial cells could be connected to

several other cells if these formed nanotubes. (3) Upon forming a nanotube, a bacterial cell connected to another cell, which was randomly picked from all cells that were located around the focal cell within a radius that corresponded to the maximum nanotube length. (4) Established nanotubes disconnected by chance with a probability of 0.001 s^{-1} or if the connected cells moved farther away from each other than the maximum nanotube length.

Bacterial cells divided into two cells when its dry biomass increased beyond a threshold (i.e. 500 fg). Bacterial cells would eventually overlap due to growth and replication. In this case, the Shoving algorithm (Lardon *et al.* 2011a) was applied to push overlapping microbes away from each other. To account for mortality, bacterial cells were removed from the simulated system with a probability of 0.0001 s^{-1}.

Generally, the dynamics of bacterial aggregates were simulated over 10 hours or until the population reached 0 or 5,000 cells.

In the simulated long term evolution experiment, bacterial aggregates were reduced to 100 randomly chosen cells and repositioned upon reaching a population of 1,000 individuals. Simultaneously, metabolite concentrations in the environment were reset to zero. Simulations were conducted over 20 days or until all bacterial cells died.

Complete model descriptions (process modules, time step lengths, and parameter values) are provided in Section A.5 and implementation details of *McComedy* and all process modules are provided in Section 2.2.

4.4.4 Model calibration

The maximum overproduction rates of metabolites were used to calibrate the model to scenarios with different diffusion coefficients and nanotube lengths. This was necessary to ensure that the overall availability of metabolites was independent of the assumed diffusion coefficient and nanotube length. For every scenario, the maximum overproduction rates were set such that simulated cocultures of *CoopΔY* and *CoopΔW* grew (without supplemented amino acids) at the same rate as in the experiment (Figure 17 C). These maximum overproduction rates were identified by testing different values, calculating the resulting growth rate, and interpolating according to the experimentally-determined growth rate.

4.4.5 Estimating parameter values

The individual-based model was parametrized according to data from the literature and experiments. A complete list of parameter values with explanations and references for each IBM is provided in Section A.5. Given that not all parameter values could be directly obtained from literature or experimentally quantified, some had to be estimated based on the available data as follows.

The biomass yield resulting from the uptake of tyrosine and tryptophan was estimated based on (i) the codon usage of tryptophan and tyrosine in the genome of *E. coli* (0.014 and 0.03, respectively, Maloy *et al.* 1996) and (ii) the protein ratio in the dry mass of *E. coli* (0.524, Stouthamer 1973). Assuming that, for simplicity, all consumed amino acids are integrated into the proteome and all other metabolites are available *ad libitum*, the biomass yield from the uptake of the amino acid AA was estimated according to

$$y_{AA} = \frac{1}{f_{AA} * x_p}$$

, where f_{AA} is the codon usage of the amino acid AA and x_p is the protein ratio of the dry biomass.

The maximum uptake rates of metabolites, as required according to Monod dynamics, were calculated using the yield y_{AA} and the growth rate of the non-cooperative strain in minimal medium with the required amino acid at saturation (Figure 17 A, B), according to

$$q_{max,AA} = \frac{r_{AA}}{y_{AA}}$$

, where r_{AA} is the growth rate of the respective non-cooperative strain at saturation of AA. While the overproduction rates resulted from model calibration, the ratio of tryptophan to tyrosine overproduction was kept constant (at 0.88). This ratio was estimated by performing a flux balance analysis (Orth *et al.* 2010) on the metabolic model iJO1366 (Orth *et al.* 2011) using the COBRA Toolbox 3.0 (Heirendt *et al.* 2019).

The cooperativity of the cooperative strains (i.e. how much of the biomass growth was sacrificed to overproduce metabolites) was estimated by the ratio of the growth rates of cooperative and non-cooperative strains in monoculture with the respective metabolites at saturation (Figure 17 A, B).

4.4.6 Selection coefficient

To assess the selection for cooperators in a system consisting of cooperators and non-cooperators, the selection coefficient for cooperation SC_{Coop} was calculated according to

$$SC_{Coop} = \frac{ln\left(\frac{BM_{Coop}(t)}{BM_{Coop}(0)}\right)}{t} - \frac{ln\left(\frac{BM_{NC}(t)}{BM_{NC}(0)}\right)}{t}$$

, where t is the simulation time, $BM_{Coop}(x)$ is the total biomass of all cooperators at time x, and $BM_{NC}(x)$ is the total biomass of all non-cooperators at time x. A positive SC_{Coop} indicates that selection favors cooperators and a negative SC_{Coop} indicates that selection favors non-cooperators.

4.4.7 Statistical analysis and visualization

Correlations between variables were determined by using Pearson's product-moment correlation whenever appropriate or by using Spearman's rank correlation. The assumptions for Pearson's product-moment correlation were tested by visually inspecting the Tukey Anscobe Plot and the normal Q-Q plot from linear models of the corresponding data.

All statistical analyses were conducted with R 4.1.0 (R-Core-Team 2021). Data was visualized with the R-package 'ggplot2' (Wickham 2016).

5 General Discussion

5.1 Thesis in a nutshell

5.1.1 Scope of this thesis

The overarching aim of this thesis was to provide and apply an innovative model-based methodology to advance the understanding of the functioning of microbial systems under explicit consideration of consumer-resource interactions and their evolution. A particular focus was on clarifying how individual-level processes shape the evolution of metabolic interactions. To enable such analyzes, the objective was to develop an approach that facilitates systematic individual-based modeling of microbial systems. For this, a framework for building IBMs from generic submodels of biological, physical, and evolutionary processes has been developed, tested by reproducing previously published studies on microbial ecology, and applied to address important open questions from microbial ecology. It was supposed to be implemented as an open-source user-friendly software tool. The expectation was to fulfill a number of standards for model building and analysis as well as for the design of research software:

1. A high degree of genericity due to a modular design with building blocks which consist of tested generic submodels that can be flexibly combined to IBMs for the analysis of microbial systems and adjusted to the research question of interest. The advantage of such submodels is that the composed IBM can be applied to a specific situation or new environmental conditions by appropriate parameterization without the need to develop a new model from the scratch. (Grimm and Berger 2016). The modular design additionally allows using the modeling framework as 'virtual laboratory' in which causal understanding on the relevance of a particular process can be gained through comparative analyses of scenarios based on alternative submodels for this process (Stillman *et al.* 2015, Schlüter *et al.* 2019, Reynolds and Acock 1997).
2. A model output that can be compared with patterns from empirical observations. This allows applying the strategy of pattern-oriented modeling, a technique where the model output is matched with as many empirical patterns as possible to achieve predictive power and structural realism (Grimm *et al.* 1996, Grimm *et al.* 2005, Gallagher *et al.* 2021).
3. Model validation to demonstrate a reasonable accuracy in the intended field of application by reproducing previously published studies on metabolic interactions and evolution in microbial systems (Rykiel 1996).
4. Fulfilment of the FAIR principles (Findable, Accessible, Interoperable, Reusable) for the software implementation (Lamprecht *et al.* 2020). This includes easy access for the

community of researchers by implementing the framework as an open-access and user-friendly software tool

A further objective was to apply this approach to a current research question in the field of microbial ecology and evolution. More specifically, the question whether and how cooperative cross-feeding can evolve in microbial multicellular aggregates should be examined.

5.1.2 McComedy: A framework for individual-based modeling

An analytical framework for individual-based modeling of microbial systems, named *McComedy* (**Mic**robial **Co**mmunities, **Me**tabolism, and **Dy**namics), was developed in the context of this thesis. A special feature of *McComedy* is its modular design, which is based on several submodels (in *McComedy* referred to as process modules) that represent relevant biological, physical, and evolutionary processes (Section 2.2.7). These process modules can be combined to IBMs for analyzing microbial systems and comparing scenarios. The modular design of *McComedy* allows to extend the framework with additional process modules that simulate other processes or account for different assumptions. This is a step towards the outlook on individual-based modeling given by Grimm and Berger (2016) who envisioned public libraries of generic and mechanistic submodels.

With the so far provided process modules, *McComedy* can be used to address research questions concerning metabolic interactions and evolution in microbial systems. Passive metabolite exchange via diffusion through extracellular space, as commonly assumed in IBMs (e.g. Mitri *et al.* 2015, Bauer *et al.* 2017, Allen *et al.* 2013, Momeni *et al.* 2013b, Pande *et al.* 2016b) as well as contact-dependent resource transfer via intercellular nanotubes (Dubey and Ben-Yehuda 2011, Pande *et al.* 2015a) can be modeled. It is possible to account for the fact that some metabolites are costly to produce (D'Souza *et al.* 2014) by using a process module that implements a trade-off between resource production and biomass growth. This allows to simulate microorganisms with different levels of cooperativity (and associated fitness costs) as has been previously done to study the evolution of cooperation (e.g. Momeni *et al.* 2013b, Zhang and Perc 2016). Furthermore, microevolution can be directly integrated in the models by enabling mutations of trait parameter values.

In order to customize an IBM to a specific microbial system, general parameter values (e.g. the size of the simulated environment in each spatial dimension) and parameter values specific to the selected process modules (e.g. the metabolites' diffusion coefficients for the process module Diffusion, Section 2.2.7.4.1) can be defined according to the modeled scenario. These parameter values can be obtained from laboratory experiments. For example, fitness assays can be conducted to measure the cost for metabolic overproduction in bacteria (D'Souza *et al.* 2014). The output data of the IBMs can be then compared with further empirical data (e.g. growth kinetics, spatial patterns,

and functional responses) and therefore facilitate pattern-oriented modeling (Gallagher *et al.* 2021, Grimm *et al.* 1996, Grimm *et al.* 2005). According to the standard for individual-based models, *McComedy* was documented in the format of an ODD (Overview, Design concepts, Details) protocol in Chapter 2 (Grimm *et al.* 2006, Grimm *et al.* 2020).

5.1.3 McComedy: User-friendly software tool

McComedy was implemented as a stand-alone software tool for building and running IBMs for analyzing the functioning of microbial systems. In order to achieve accessibility to a broad range of researchers, it was designed as a fast, intuitive, and user-friendly tool that requires no proficiency in programming or third-party software. The graphical user interface supports the design and parametrization of IBMs as well as systematic analyses with parameter variations. While the simulations run, a real time visualization allows to observe the current number and positions of microbes as well as the concentrations of the resources in the environment. In addition to the graphical user interface, *McComedy* can also be operated from the command prompt, which enables large-scale simulations on high performance computing systems. The software has options to flexibly adjust the extent and frequency of the model output, thus allowing for a broad range of spatially and temporally explicit analyses. The software, the source code, and the ODD protocol of *McComedy* are publically available at https://git.ufz.de/bogdanow/mccomedy.

5.1.4 Model validation

The suitability of *McComedy* to capture the essential dynamics of microbial systems was demonstrated by reproducing and furthermore adding to the results of two distinct studies from the literature. The studies were chosen such that their focal research questions closely matched *McComedy*'s intended field of application.

5.1.4.1 Example 1: Spatial organization model (Mitri et al. 2015)

Mitri *et al.* (2015) assessed whether resource limitation leads to spatial separation of different strains in an initially well-mixed, growing colony. By conducting experiments with differently marked strains of *Pseudomonas aeruginosa* on agar plates with varied concentrations of nutrients and with an IBM, the authors demonstrate that resource limitation indeed causes spatial separation of the two strains of bacteria. Using *McComedy*, the findings of Mitri *et al.* (2015) were recapitulated. In agreement with the results of the original study, it was demonstrated that spatial separation of the two strains occurred closer to the initial inoculum where nutrients levels were lower. Complementing the original work, it was shown with *McComedy* that also reduced diffusivity of the nutrients resulted in separation of the strains closer to the initial inoculum.

5.1.4.2 Example 2: Cooperation model (Momeni et al. 2013b)

Momeni *et al.* (2013b) investigated whether cooperative metabolic exchange between microorganisms can be maintained in the presence of non-cooperative competitors in a spatially

structured environment. By experimenting with synthetically engineered cooperative and non-cooperative strains of *Saccharomyces cerevisiae* and by applying an IBM, it was demonstrated that cooperators intermix and grow well, while non-cooperators spatially segregate and grow poorly. Also in this example, the findings were successfully reproduced with *McComedy*. Both the frequency of the cooperators and their degree of intermixing increased while the yeast cells grew on a simulated agar plate, which was in line with the results of Momeni *et al.* (2013b). Adding to the original work, it was demonstrated that, in contrast to expectations, reduced release of exchanged metabolites resulted in a decreased advantage for the cooperators. This phenomenon is discussed in more detail in Sections 4.3 and 5.1.5.

The successful recapitulation of the results of the two exemplary studies indicated a reasonable accuracy of IBMs built with *McComedy*. Furthermore, by simulating these distinct systems with IBMs that are based on the same set of processes, it was also demonstrated that *McComedy* can be used as a unifying framework for a coherent understanding of different ecological processes.

5.1.5 Model application: Evolution of cooperation in cross-feeding bacteria

Metabolic interaction is prevalent in natural microbial systems (Embree *et al.* 2015, Zengler and Zaramela 2018, Gorter *et al.* 2020, Johnson *et al.* 2020, Zelezniak *et al.* 2015). However, so far it is not clear whether these interactions always represent by-product interactions, in which only metabolic by-products are released or if this exchange can also be truly cooperative, that is, costly for the donor (Oliveira *et al.* 2014). Thus, a major question in microbial ecology and evolution is if and how cooperative traits can evolve and be maintained despite the burden of fitness costs that are associated with cooperation. In this context, there is evidence that, in contrast to well-mixed environments in which individuals interact randomly and only temporary, environments that provide spatial structure such as biofilms, soil particles, and colonies can promote cooperation (Nowak *et al.* 1994, Nowak and May 1992, Yamamura *et al.* 2004, Allison 2005, Stump *et al.* 2018b). The reason for this is that spatial structure allows for repeated interactions between adjacent individuals. If cooperators colocalize by chance, reciprocal cooperation strongly enhances their growth as compared to pairs including non-cooperators. This results in a pattern of spatial self-organization that favors cooperative individuals and excludes non-cooperators from the cooperative benefits (Momeni *et al.* 2013b, Germerodt *et al.* 2016). In liquid environments, bacteria often form multicellular aggregates (Trunk *et al.* 2018), which could as well provide the spatial structure that is needed for the self-organization of cooperators and non-cooperators (Preussger *et al.* 2020). *McComedy* was applied to test if and how such multicellular aggregates can promote cooperation.

5.1.5.1 Diffusion-based exchange

By simulating multicellular aggregates of cooperative and non-cooperative bacteria with *McComedy*, it was demonstrated that diffusion of metabolites hinders the evolution of cooperation.

It was shown that cooperators were favored over non-cooperators only at diffusion coefficients of metabolites that were several orders of magnitude lower than realistic values. This was in line with previous work that suggests that cooperation is favored over non-cooperation in spatially structured environments only at low diffusivity of the shared public goods (Allen *et al.* 2013, Borenstein *et al.* 2013, Allison 2005, Germerodt *et al.* 2016, Dobay *et al.* 2014, Kümmerli *et al.* 2009). However, in simulations with *McComedy*, even at low diffusion coefficients favoring cooperators over non-cooperators, cooperators became less cooperative in the long term. This observation was termed the cooperation paradox of public goods. Analyzing the spatial distribution of bacteria and the diffusing metabolites led to the conclusion that high cooperativity in terms of high release of shared metabolites resulted in increased interaction distance between the bacteria, which undermined the role of the spatial structure of the multicellular aggregate and thus inhibited the mechanism to favor cooperation. This finding challenges the existing notion that, in spatially structured environments, low diffusivity of shared public goods can be sufficient to promote the evolution of cooperation (Dobay *et al.* 2014).

5.1.5.2 Nanotube-based exchange

In order to examine whether cooperative bacteria can still evolve in multicellular aggregates, another mechanism for the exchange of metabolites was tested in another set of simulations with *McComedy*. Intercellular nanotubes that bacteria form to exchange cytoplasmic material between different cells (Dubey and Ben-Yehuda 2011, Pande *et al.* 2015a) were assumed to mediate metabolic exchange. Under these conditions, the interaction distance of this contact-dependent exchange was only limited by the length of the nanotube and thus independent of the metabolites' diffusivity or bacterial cooperativity. In this scenario, cooperation was favored under reasonable assumptions and, as opposed to the scenario with diffusion-based metabolic exchange, cells became more cooperative in the simulated evolution experiment. This findings suggest that intercellular nanotubes and other means of contact-dependent metabolic exchange may have evolved as a prerequisite for cooperation.

Note that comparative simulations with varied assumptions of cellular processes (i.e. the extent and mechanism of metabolic exchange) were a prerequisite for these findings. This underlines the importance of *McComedy*'s mechanistic process implementation and modular design, which facilitated this type of analysis.

5.2 Synthesis and outlook

5.2.1 Assessment of the modeling framework McComedy

As outlined in the Introduction of this thesis (Chapter 1), prediction and thorough understanding of ecological systems can be achieved with IBMs that consist of tested, generic submodels that are based on first principles. From the mechanistic interaction of individuals, of which traits and

behaviors are defined according to such first principles, the system's organization and functioning can emerge naturally (Grimm and Berger 2016, Grimm *et al.* 2016). How well does *McComedy* meet this standard? Overall, IBMs created with *McComedy* can be indeed classified as mechanistic. A subset of the implemented process modules is already based on first principles. For example, the process module Diffusion (Section 2.2.7.4.1) implements a numeric approximation of Fick's second law of diffusion (finite difference method, Mugler and Scott 1988). Another example, the trade-off between cooperativity and resource overproduction in the process module Cooperativity-FitnessCost (Section 2.2.7.2.5), is based on a Flux Balance Analysis (Orth *et al.* 2010). This is a powerful aggregated alternative to incorporating such metabolic modeling dynamically (e.g. Bauer *et al.* 2017, Biggs and Papin 2013, Harcombe *et al.* 2014). Despite its heuristic character, this trade-off view is in line with implications from prominent first principles such as Dynamic Energy Budget (DEB) theory (Kooijman and Troost 2007), stating that each activity costs energy. For some process modules, it depends on the user whether they are used to represent processes mechanistically. It is outlined in Section 2.2.7.2.6 how the process module Flow can be used to accurately model Brownian motion according to Einstein (1905). On the other hand, some process modules are not based on first principles, so far (e.g. Lysis, Section 2.2.7.2.9 and Replication, Section 2.2.7.2.14). However, the modular design of *McComedy* allows to eventually replace these process modules by more mechanistic implementations. Future process modules could, for example, rely on the Metabolic Theory of Ecology (Brown *et al.* 2004, Schramski *et al.* 2015). Once such extensions for *McComedy* exist, comparative studies with process modules of varied mechanistic detail could be insightful for determining an appropriate level of model complexity.

Modular ecological models have the advantage that they can be easily adjusted to different scenarios or new experimental results and facilitate alternative hypothesis testing by replacing individual modules (Reynolds and Acock 1997). This plays a central role in the design of *McComedy* as all biological, physical, and evolutionary processes are encapsulated in mostly independent process modules. Accordingly, *McComedy* was suitable to simulate different microbial systems (i.e. colonies of *Pseudomonas aeruginosa*, Section 3.2.2; biofilm-like communities of *Saccharomyces cerevisiae*, Section 3.2.3; and multicellular aggregates of *Escherichia coli*, Chapter 4) and to test alternate hypotheses regarding the mechanism of metabolite transfer between bacterial cells (i.e. diffusion, Section 4.2.2, versus intercellular nanotubes, Section 4.2.3). The integration of the process modules to a functional IBM is accomplished by an innovative scheduling system that runs process modules according to their individual time step and ensures synchronous updates of simultaneously executed process modules (Figure 2).

The parametrization of the process modules permits integrating experimentally obtained data and the model output is suitable for comparison with results from the laboratory. This allows to test predictions and, by matching the model output to as many empirical patterns as possible, to increase confidence that the model captures the relevant structure and functioning of the modeled system,

that is, conducting pattern-oriented modeling (Grimm *et al.* 1996, Grimm *et al.* 2005, Gallagher *et al.* 2021). These possibilities bridge the gap between experimental approaches on one side and ecological theory and modeling on the other side, thus, addressing the repeatedly emphasized demand of coupling experimental and theoretical ecology (Zaccaria *et al.* 2017, Hellweger *et al.* 2016, Widder *et al.* 2016b).

McComedy's open-source access, user-friendliness, and standardized documentation as an ODD (Overview, Design concepts, Details) protocol (Grimm *et al.* 2006, Grimm *et al.* 2020) is a step towards being FAIR (findable, accessible, interoperable, reusable), a standard that was originally established for scientific data (Wilkinson *et al.* 2016) and later also adapted to research software (Lamprecht *et al.* 2020). Standards for model building and analysis are regarded as a major goal to serve the modelers' community and to increase the value of models, as promoted by various initiatives fostering open modeling (e.g. The Open Modeling Foundation, https://openmodelingfoundation.github.io/).

A technical limitation of *McComedy* is set by the computational demand that is associated to large simulations. So far, the development of *McComedy* was mainly dictated by standards in modeling and research. Further work on *McComedy* could optimize algorithms, software architecture, and working pipelines. Improvements in these aspects could reduce *McComedy*'s demand for simulation time and other resources (e.g. working memory, electricity, carbon footprint, etc.).

In summary, the main advantages of *McComedy* are, first, the providing of generic process modules that facilitate the development of a variety of mechanistic models of microbial systems, second, a high degree of modularity that allows for flexible model design and systematic testing of alternative hypotheses, third, the suitability to integrate experimental data and compare results with empirically observed patterns, and fourth, an open-access implementation as a user-friendly software tool. Future improvements should address the mechanistic detail of some process modules, extend the framework with new process modules, and optimize *McComedy*'s source code for a better computational performance.

5.2.2 Eco-evolutionary insights from applying McComedy

In Chapter 3, the potential of *McComedy* to accurately capture the structure and functioning of microbial systems was validated by successfully reproducing results from two previously published studies (Mitri *et al.* 2015, Momeni *et al.* 2013b). In addition to reproducing the results, *McComedy* was further applied to extend the understanding of the two modeled systems. In the first example, Mitri *et al.* (2015) have suggested that in environments with limited nutrient availability, colony expansion leads to spatial self-organization due to a bottle-neck effect at the edge of the colony (this is explained in more detail in Section 3.2.2). While this hypothesis was strongly supported by the presented data, the authors did not manipulate the diffusion coefficient of the nutrients, which

would have altered the nutrient availability in the outer rim of the colony and, thus, the bottle-neck effect. Therefore, this was addressed with *McComedy*. The results were fully in line with the original findings: a weaker bottle-neck effect due to a higher diffusion coefficient resulted in a reduced spatial self-organization and *vice versa* (Figure 6). This result does not only corroborate the findings of Mitri *et al.* (2015) but it also points to a connection with studies on the evolution of cooperation, which demonstrate that increased diffusion of public goods disfavors cooperation in spatially structured environments (Allen *et al.* 2013, Borenstein *et al.* 2013, Allison 2005, Germerodt *et al.* 2016, Dobay *et al.* 2014, Kümmerli *et al.* 2009). This connection leads to a coherent picture: high diffusivity of public goods increases their availability in microbial structures, which has a negative effect on the spatial self-organization of the microorganisms and, ultimately, spatial organization is known to facilitate cooperation (Nowak *et al.* 1994, Nowak and May 1992, Yamamura *et al.* 2004, Allison 2005, Stump *et al.* 2018b, Momeni *et al.* 2013b, Pande *et al.* 2016b). The fact that both nutrient availability and the diffusion coefficient have a similar effect on the spatial organization of the microorganisms also links to the 'cooperation paradox of diffusive goods' presented in Chapter 4. This paradox states that increasing the cooperative release of diffusible goods (i.e. increasing the availability of these goods) has a similarly negative effect on cooperation as increasing the diffusion coefficient.

The second example for model validation of *McComedy* was one of the mentioned studies on the evolution of cooperation (Momeni *et al.* 2013b). *McComedy* successfully reproduced the finding that in a spatially structured environment, reciprocally cross-feeding cooperators can be favored over non-cooperators (details are provided in Section 3.4.3). Based on simulations, in which non-cooperators were replaced by 'intermediate cooperators' (i.e. they released a resource at an intermediate rate) but they were still disfavored, Momeni *et al.* (2013b) suggested that self-organization favors cooperators that supply the most benefits. However, simulations with *McComedy* indicated that increased resource release by the cooperators reduces their selective advantage against non-cooperators. Thus, it was suggested that high cooperativity could be disfavored in the long term. This phenomenon was examined more thoroughly in Chapter 4, which led to the discovery of the aforementioned cooperation paradox of diffusive goods.

The cooperation paradox of diffusive goods constitutes a novel insight into the evolution of cooperation in spatially structured environments. In that context, it was so far assumed that cooperation can evolve at low diffusion rates of the common good (Allen *et al.* 2013, Borenstein *et al.* 2013, Allison 2005, Germerodt *et al.* 2016, Dobay *et al.* 2014, Kümmerli *et al.* 2009). Dobay *et al.* (2014) even stated that '*reduced public good diffusion (which keeps the public goods closer to the producer) is not only essential but also sufficient for cooperation to be promoted*'. While this seems to be true in the short term, the cooperation paradox of diffusive goods clearly suggests that even at decreased diffusion rates, reduced cooperativity is selected for in the long term if the public goods diffuse through extracellular space.

In order to observe this long-term effect, it was necessary to enable mutations of the microorganisms' cooperativity parameter. The importance of this micro-evolutionary process for the model outcome underpins the necessity of considering evolution in ecological models, as suggested throughout literature (e.g. Grimm *et al.* 2016, Grimm and Berger 2016, Loreau 2010b, Hellweger *et al.* 2016, Coulson *et al.* 2006, Pelletier *et al.* 2009, Widder *et al.* 2016b). Moreover, it was demonstrated that the mechanism of metabolic exchange (i.e. via diffusion or nanotubes) determined whether cooperation was selected for or against. The vast majority of IBMs in this context either assume metabolic exchange via diffusion (e.g. Allison 2005, Momeni *et al.* 2013b, Germerodt *et al.* 2016, Bauer *et al.* 2017) or directly between adjacent individuals (e.g. Nowak *et al.* 1994, Hauert and Doebeli 2004). However, the results of this thesis indicate that these are strong assumptions that can be decisive for the emergent system-level outcome.

The work presented in this thesis contributes to the general understanding of the evolution of cooperation in spatially structured microbial systems. The discovery of the cooperation paradox of diffusive goods opens a new perspective on the role of public good diffusivity and intercellular nanotubes were identified as potential pre-requisites for the evolution of cooperation.

5.2.3 Limitations and further research directions

Following on from the here presented work, further progress can be made by both continuing the development of *McComedy* and investigating the evolution of cooperative cross-feeding in microbial systems.

Further work on *McComedy* should firstly address the shortcomings mentioned in Section 5.2.1, that is, the mechanistic detail of some process modules and computational performance. A limitation for the wide application of *McComedy* is so far set by the amount and diversity of the implemented process modules. For example, process modules that account for chemotaxis or the interaction with toxic compounds are not available, yet. However, the modular design of *McComedy* fully supports extension by implementing additional process modules. This is further supported by the open-access of the source code, allowing all researchers to develop extensions for *McComedy*.

The theoretical findings of this thesis can be extended in the future by continuing the systematic analysis of the evolution of cooperative cross-feeding in bacterial multicellular aggregates. For example, the mutation of other central parameter values can be taken into account. Dal Co *et al.* (2020) have suggested that increased uptake of metabolites can favor the evolution of cooperation. It would be interesting to investigate whether this effect can compensate for the cooperation paradox of diffusive goods. It would be also interesting to test whether the presented results can be reproduced in spatially structured environments other than bacterial multicellular aggregates.

Furthermore, it would be intriguing to assess whether a stressful environment (e.g. fluctuating availability of resources) would affect the results of this thesis.

Moreover, *McComedy* could be used to aggregate results to population or community-level parameters that can, in turn, be used to model higher-level dynamics. For example, demographic rates of individual multicellular aggregates could be obtained from simulations with *McComedy* and integrated in a different model to assess the dynamics of microbial systems that consist of numerous aggregates.

A framework similar to *McComedy* can be implemented for virtually any ecological system that is worthwhile being investigated with IBMs. Therefore, *McComedy* could also serve well as an inspiration for other generic modeling tools.

Appendix

A.1 McComedy process module dependencies

Table 25. Dependencies of all process modules (Section 2.2.7) in McComedy.

Process module	Dependency	Explanation
Attachment	Proximity-Manager	The computationally intensive test which microbes are close enough for attachment requires information on which microbes are in each other's vicinity, provided by ProximityManager
Cooperativity-FitnessCost	Constant-Production	Parameter of ConstantProduction is manipulated
Cooperativity-FitnessCost	Growth	Parameter of Growth is manipulated
NanoTube-Exchange	Constant-Production	Exchanged resources are taken from the state variable *Product pool* which is added by the process module ConstantProduction
NanoTube-Exchange	Substrate-Utilization	Exchanged resources are added to the state variable *Substrate pool* which is added by the process module SubstrateUtilization
PassiveRelease	Constant-Production	Produced resources are stored in the state variable *Product pool* which is added by the process module ConstantProduction
PassiveRelease	CellPartition	Resources are released into overlapping grid cells proportionally to the overlap. The overlap is estimated by CellPartition
PassiveUptake	Substrate-Utilization	Consumed resources are stored in the state variable *Substrate pool* which is added by the process module SubstrateUtilization
PassiveUptake	CellPartition	Resources are consumed from overlapping grid cells proportionally to the overlap. The overlap is estimated by CellPartition
Starvation	Substrate-Utilization	The state variable *Starvation* that indicates whether a microbe is starving is added by the process module SubstrateUtilization
Substrate-Utilization	Growth	Resources allocated to growth are stored in the state variable *Growth resources* which is added by the process module Growth
Shoving	Proximity-Manager	The test which microbes overlap can be computationally boosted if information which microbes are close to each other is provided. This is done by ProximityManager

A.2 Computational performance

We measured the computation time for a set of representative simulations. The time step length, microbial abundance, and spatial extent of the simulated environment were varied. The simulations were performed on an ordinary laptop. The parametrization of the simulations is provided in (Section A.5).

Hardware

Table 26. Test simulation executed on following hardware.

Device	Notebook Lenovo T480
CPU	Intel 8th Gen Quad-Core i5-8250U
CPU speed	1.6 GHz
Memory	B DDR4 RAM

Performance

The duration to simulate one virtual second was assessed for different parametrizations (Figure 14 and Figure 15). Overall the computation time increased with higher microbe numbers, larger environment size, and shorter time steps (ΔT). However, at very high microbe numbers, reducing the environment sizes (i.e. increasing the density) resulted in increased computation time as the computationally expensive *Shoving* algorithm (Section 2.2.7.5.3) had to shift more microbes.

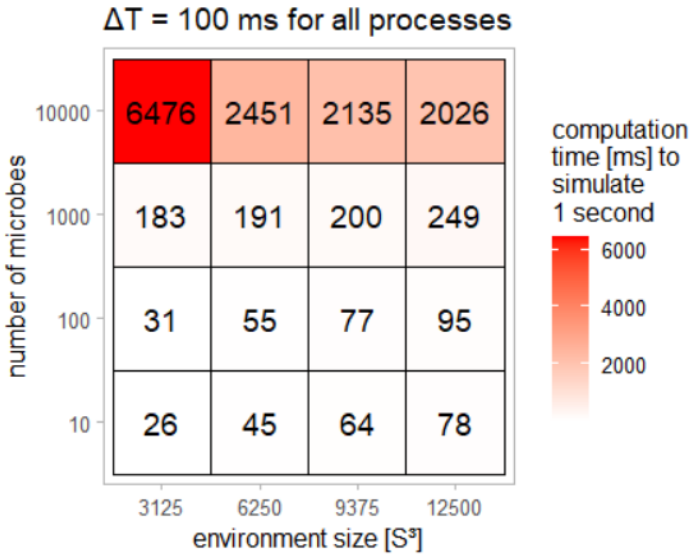

Figure 14. Computation time for ΔT = 100 ms for all processes.

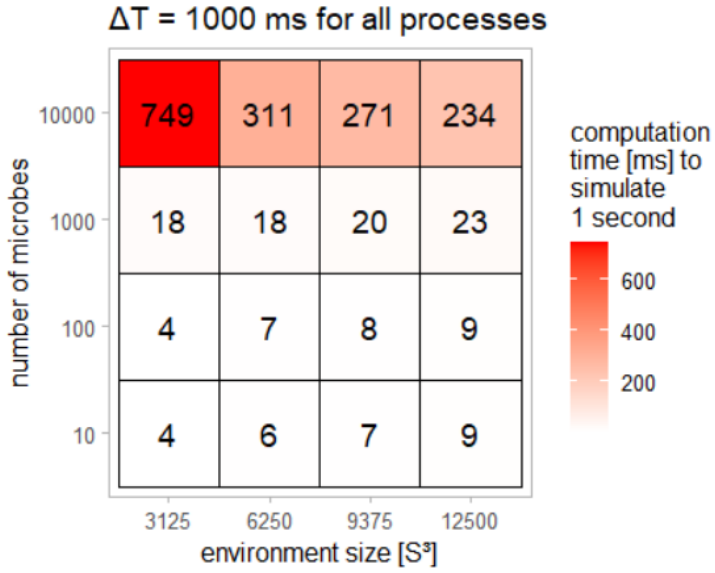

Figure 15.Computation time for ΔT = 1000 ms for all processes.

A.3 Mathematical modeling

A spatially structured environment can promote cooperation by imposing repeated interactions on individuals in the vicinity of each other. Pairs of randomly adjacent cooperators reciprocally benefit each other and, thus, perform better than pairs including non-cooperators. However, the results presented in the main text of this study indicate that cooperation is disfavored in bacterial cross-feeders that exchange metabolites via diffusion in a spatially structured environment. The suggested reason is that increased release of public goods results in a greater distance at which these public goods can be consumed, attenuating the purpose of the spatially structured environment. This effect is examined with a simple mathematical model.

Public goods released by the cooperators propagate through space as described by Fick's second law of diffusion. The concentration of the public good decreases with distance x and time t according to the fundamental solution of the diffusion equation in 1D

$$F(x,t) = \frac{1}{\sqrt{4\pi D t}} e^{-\frac{x^2}{4Dt}}$$

, where D is the diffusion coefficient of the public good (Murray 2002). Uptake rates of the public goods by cooperators and non-cooperators depend on the local concentration C of the public good, as described by the Monod function

$$q(C) = q_{max} \frac{C}{K_m + C}$$

, where q_{max} is the highest possible uptake rate and K_m is the Monod constant, that is, the concentration of C that satisfies $\mu(C) = \frac{\mu_{max}}{2}$.

For the sake of simplicity, it is assumed that reproductive fitness depends linearly on the uptake rate of public goods. Furthermore it is assumed that all parameter values in the following equations are positive as this is sensible from a physical point of view.

As the concentration of the diffusing public good continuously decreases with distance, it is obvious that proximate recipients of the public good exhibit a greater uptake rate than distant ones, according to:

$$q\big(C_0 F(d,t)\big) > q\big(C_0 F(d+\alpha,t)\big)$$

, where C_0 is the concentration of the public good released by a cooperator, d is the distance between the cooperator and a proximate recipient, $d + \alpha$ is the distance between the cooperator and a more distant recipient, and t is the time between release and uptake.

So far, this does not contradict promotion of cooperation by a spatially structured environment as release of public goods mostly benefits recipients in the immediate vicinity, favoring pairs of cooperators. However, assuming that mutations cause the cooperators to increase the release of the public good by β, the relative increase of the uptake by the proximate recipient (at distance d) is less than the relative increase of the uptake by the distant recipient (at distance $d + a$). This is shown by:

$$\frac{q\big((C_0+\beta)F(d,t)\big)}{q\big(C_0F(d,t)\big)} < \frac{q\big((C_0+\beta)F(d+\alpha,t)\big)}{q\big(C_0F(d+\alpha,t)\big)}$$

, which can be written as:

$$\frac{q_{max}\dfrac{1}{1+\dfrac{K_m}{(C_0+\beta)F(d,t)}}}{q_{max}\dfrac{1}{1+\dfrac{K_m}{C_0F(d,t)}}} < \frac{q_{max}\dfrac{1}{1+\dfrac{K_m}{(C_0+\beta)F(d+\alpha,t)}}}{q_{max}\dfrac{1}{1+\dfrac{K_m}{C_0F(d+\alpha,t)}}}$$

and simplified to:

$$\frac{\dfrac{1}{K_m}+\dfrac{1}{C_0F(d,t)}}{\dfrac{1}{K_m}+\dfrac{K_m}{(C_0+\beta)F(d,t)}} < \frac{\dfrac{1}{K_m}+\dfrac{K_m}{C_0F(d+\alpha,t)}}{\dfrac{1}{K_m}+\dfrac{K_m}{(C_0+\beta)F(d+\alpha,t)}}$$

$$<=> \quad \frac{\dfrac{1}{K_m}+\dfrac{1}{F(d,t)}\left(\dfrac{1}{C_0+\beta}+\dfrac{1}{C_0}-\dfrac{1}{C_0+\beta}\right)}{\dfrac{1}{K_m}+\dfrac{1}{(C_0+\beta)F(d,t)}} < \frac{\dfrac{1}{K_m}+\dfrac{1}{F(d+\alpha,t)}\left(\dfrac{1}{C_0+\beta}+\dfrac{1}{C_0}-\dfrac{1}{C_0+\beta}\right)}{\dfrac{1}{K_m}+\dfrac{1}{(C_0+\beta)F(d+\alpha,t)}}$$

$$<=> \quad 1+\frac{\dfrac{1}{F(d,t)}\left(\dfrac{1}{C_0}-\dfrac{1}{C_0+\beta}\right)}{\dfrac{1}{K_m}+\dfrac{1}{(C_0+\beta)F(d,t)}} < 1+\frac{\dfrac{1}{F(d+\alpha,t)}\left(\dfrac{1}{C_0}-\dfrac{1}{C_0+\beta}\right)}{\dfrac{1}{K_m}+\dfrac{1}{(C_0+\beta)F(d+\alpha,t)}}$$

$$<=> \quad \frac{1}{F(d,t)\dfrac{1}{K_m}+\dfrac{1}{(C_0+\beta)}} < \frac{1}{F(d+\alpha,t)\dfrac{1}{K_m}+\dfrac{1}{(C_0+\beta)}}$$

$$<=> \quad F(d,t)\frac{1}{K_m}+\frac{1}{(C_0+\beta)} > F(d+\alpha,t)\frac{1}{K_m}+\frac{1}{(C_0+\beta)}$$

$$<=> \quad F(d,t) > F(d+\alpha,t)$$

. Accordingly, decreased release of public goods results in a smaller relative decrease of the uptake by the proximate recipient than by the distant recipient. Considering that promotion of cooperation by a spatially structured environment requires short interaction distances, this results in a dilemma: cooperation (i.e. releasing public goods) is favored in a spatially structured environment the most when the least public goods are released.

A.4 Supplementary figures

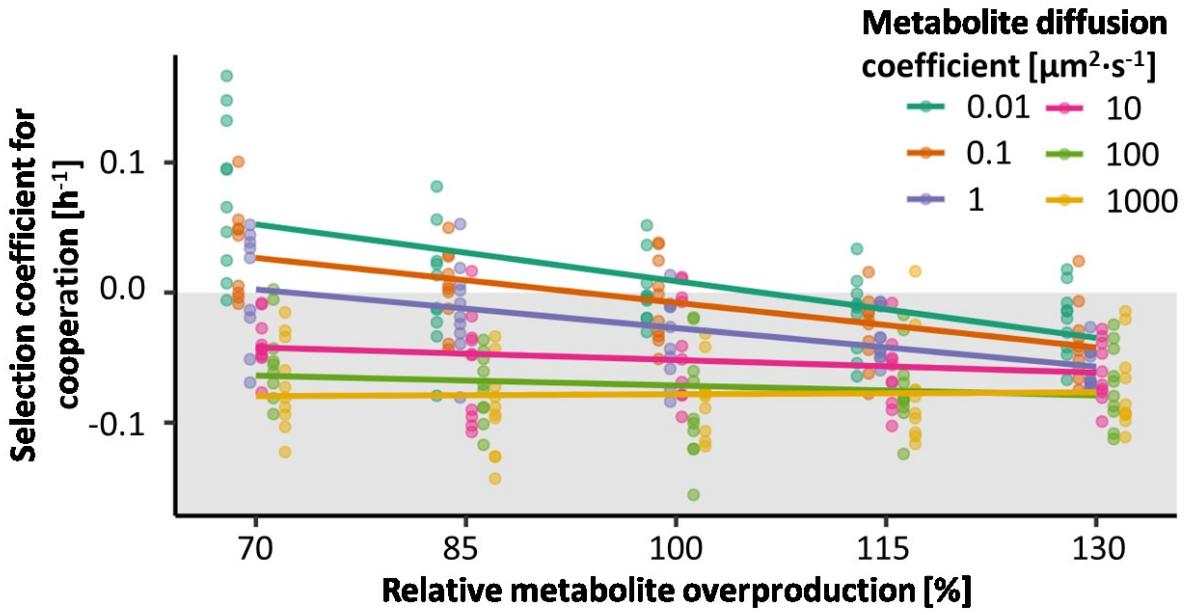

Figure 16. Selection favors reduced metabolite overproduction if metabolites are exchanged via diffusion. Selection coefficient for cooperation (SC_{COOP}) against relative metabolite overproduction is plotted for different metabolite diffusion coefficients (different colors). The relative metabolite overproduction is based on the overproduction rate that was fitted to experimental data, whereby 100 % corresponds to the fitted value. The grey area ($SC_{COOP} < 0$) indicates an overall advantage for non-cooperators. The lines represent the general trend as obtained from multiple linear regressions. Each group (i.e. combination of cooperativity and diffusion coefficient) consists of 10 replicates. Pearson's moment correlation for each metabolite diffusion coefficient D [µm²·s⁻¹]: D = 0.01: r = -0.59, p = 5.5e-6, n = 50; D = 0.1: r = -0.63, p = 1.2e-6, n = 50; D = 1: r = -0. 57, p = 1.8e-5, n = 50; D = 10: r = -0.22, p = 0.12 (not significant), n = 50; D = 100: r = -0.15, p = 0.28 (not significant), n = 50; D = 1000: r = 0.03, p = 0.83 (not significant), n = 50.

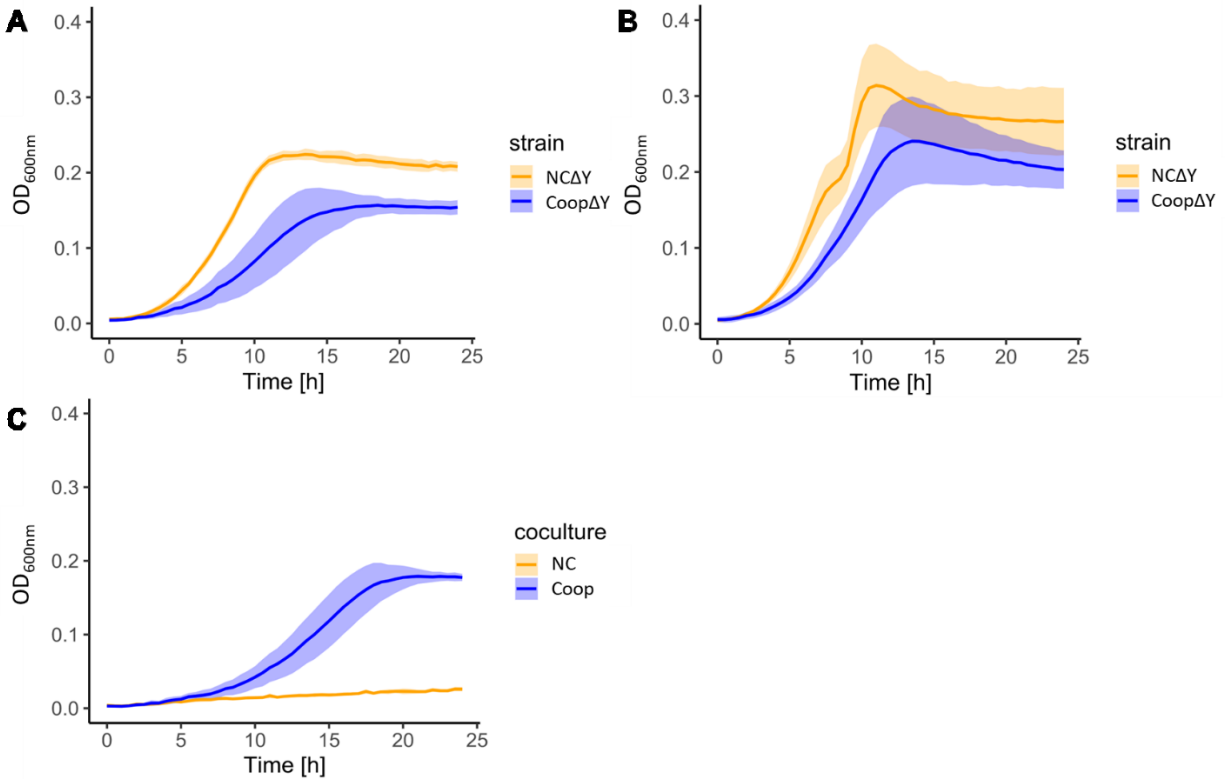

Figure 17. Growth kinetics of the model strains grown in minimal medium. A: Non-cooperative (orange) and cooperative (blue) strains that are auxotrophic for tyrosine were grown in monoculture with supplemented tyrosine. B: Non-cooperative (orange) and cooperative (blue) strains that are auxotrophic for tryptophan were grown in monoculture with supplemented tryptophan. C: Non-cooperators (orange) and cooperators (blue) were grown in coculture without amino acids supplemented. All growth curves are based on 8 biological replicates. Lines indicate mean values and ribbons indicate the standard deviation.

A.5 Model parametrizations

Spatial organization model (according to Mitri *et al.* 2015)

Table 27. Process modules

Process	Time step [ms]
CellPartition	1000
ConstantResourceBoundaries	100
Diffusion	100
Growth	1000
InitCluster	-
InitModel	-
PassiveUptake	1000
ProximityManager	1000
Replication	1000
Shoving	10000
SubstrateUtilization	1000

Table 28. Resource parameters

Parameter	Value for resource R	Reference
resource name	'R'	Model assumption
initial concentration [fg/µm³]	20, 40, 60, 80, 100, 120, 140, 160 (120 when diffusion constant was varied)	The range of variation adapted from (Mitri *et al.* 2015). Absolute values set such that colonies grew within two days
diffusion constant [µm²/s]	0.015, 0.035, 0.055, 0.075, 0.095 (0.055 when initial concentration was varied)	Model assumption
constant X-boundary	TRUE	Model assumption
constant Y-boundary	TRUE	Model assumption
constant Z-boundary	FALSE	Model assumption
concentration at boundary [fg/µm³]	'As initial concentration'	Model assumption

Table 29. Microbe parameters

Parameter	Values for type M1	Values for type M2	Reference
genotype	'M1'	'M2'	Model assumption
initial abundance	100	100	Adapted from (Mitri *et al.* 2015)
biomass density [fg/µm³]	375	375	Estimated from (Neidhardt and Umbarger 1996)
min biomass [fg]	250	250	Estimated from (Neidhardt and Umbarger 1996)
max biomass [fg]	500	500	Estimated from (Neidhardt and Umbarger 1996)
consumes resource	R	R	Estimated from (Neidhardt and Umbarger 1996)
maintenance cost [1/s]	0	0	Model assumption
half-saturation constant [fg/µm³]	1.5	1.5	Adapted from (Mitri *et al.* 2015)
max uptake [1/s]	0.00039	0.00039	Adapted from (Mitri *et al.* 2015)
yield rate	0.5	0.5	Adapted from (Mitri *et al.* 2015)

Table 30. Model parameters

Parameter	Value	Reference
spatial extent X [µm]	250	Model assumption
spatial extent Y [µm]	250	Model assumption
spatial extent Z [µm]	1	Model assumption
simulation time [s]	360000	-
max microbes number	20000	-
stop when all microbes die	TRUE	-
constant initial position	FALSE	-

random generator seed	1	-
proximity raster cell size [µm]	3	Model assumption

Cooperation model (according to Momeni *et al.* 2013b)

Table 31. Process modules

Process	time step [ms]
CellPartition	10000
ChangeGenotyoe	10000
ConstantProduction	100
ConstantResourceBoundaries	100
Diffusion	100
Flow	10000
Growth	10000
ImpermeableMicrobeBoundaries	10000
InitBiofilm	-
InitModel	-
PassiveRelease	100
PassiveUptake	100
ProximityManager	10000
Replication	10000
Shoving	10000
SubstrateUtilization	10000

Table 32. Resource parameters

Parameter	Value for Resource L	Value for Resource A	Reference
resource name	'L'	'A'	Adapted from (Momeni *et al.* 2013b)
initial concentration [fmole/(5 µm)³]	0, 9999999 (0 when release rate was varied)	0, 9999999 (0 when release rate was varied)	Model assumption
diffusion constant [(5 µm)²/s]	0.01	0.01	Model assumption
constant X-boundary	FALSE	FALSE	Model assumption
constant Y-boundary	TRUE	TRUE	Model assumption
constant Z-boundary	FALSE	FALSE	Model assumption
concentration at boundary [fmole/(5 µm)³]	As initial resource concentration	As initial resource concentration	Model assumption

Table 33. Microbe parameters (living microbes)

Parameter	Value for type $R_{\rightarrow A}^{\leftarrow L}$	Value for type $G_{\rightarrow L}^{\leftarrow A}$	Value for type $C^{\leftarrow L}$	Reference
genotype	'R'	'G'	'C'	Adapted from (Momeni *et al.* 2013b)
initial abundance	115	115	115	Adapted from (Momeni *et al.* 2013b)

biomass density [fmole/(5 µm)³]	4.688	4.688	4.688	Estimated from (Klis *et al.* 2014)
min biomass [10 pg]	1.1	1.1	1.1	Estimated from (Klis *et al.* 2014)
max biomass [10 pg]	2.2	2.2	2.2	Estimated from (Klis *et al.* 2014)
consumes resource	L	A	L	Adapted from (Momeni *et al.* 2013b)
releases resource	A	L	-	Adapted from (Momeni *et al.* 2013b)
maintenance cost [fmole/10 pg/s]	0	0	0	Model assumption
half-saturation constant [fmole/(5 µm)³]	0.00013	0.00013	0.00013	Adapted from (Momeni *et al.* 2013b)
max uptake [fmole/10 pg/s]	0.000018	0.000018	0.000018	Calculated from minimal doubling time (Momeni *et al.* 2013b) and yield rate
release rate [fmole/10 pg/s]	0.000003, 0.000004, 0.000005, 0.000006, 0.000007, 0.000008 (0.000007 when initial concentration was varied)	0.000003, 0.000004, 0.000005, 0.000006, 0.000007, 0.000008 (0.000007 when initial concentration was varied)	0.000003, 0.000004, 0.000005, 0.000006, 0.000007, 0.000008 (0.000007 when initial concentration was varied)	Model assumption
yield rate [10 pg/fmole]	10	10	10.2	Model assumption. Fitness advantage of $C^{\leftarrow L}$ adapted from (Momeni *et al.* 2013b)
impermeable X-boundaries	FALSE	FALSE	FALSE	Model assumption
impermeable Y-boundaries	TRUE	TRUE	TRUE	Model assumption
impermeable Z-boundaries	FALSE	FALSE	FALSE	Model assumption
impermeable boundary offset [5 µm]	3	3	3	Model assumption
change genotype to	D1	D2	D3	Model assumption
change genotype probability [1/s]	0.000015	0.000015	0.000015	Adapted from (Momeni *et al.* 2013b)

Table 34. Microbe parameters (dummies, representing dead cells)

Parameter	**Value for type D1**	**Value for type D2**	**Value for type D3**	**Reference**
genotype	'D1'	'D2'	'D3'	Model assumption
initial abundance	0	0	0	Adapted from (Momeni *et al.* 2013b)

biomass density [fmole/(5 µm)³]	4.688	4.688	4.688	Estimated from (Klis *et al.* 2014)
min biomass [fmole]	0	0	0	Model assumption
max biomass [fmole]	9999	9999	9999	Model assumption
consumes resource	-	-	-	Model assumption
releases resource	-	-	-	Model assumption
maintenance cost [fmole/10 pg/s]	-	-	-	Model assumption
half-saturation constant [fmole/(5 µm)³]	-	-	-	Model assumption
max uptake [fmole/10 pg/s]	-	-	-	Model assumption
release rate [fmole/10 pg/s]	-	-	-	Model assumption
yield rate [10 pg/fmole]	-	-	-	Model assumption
impermeable X-boundaries	FALSE	FALSE	FALSE	Model assumption
impermeable Y-boundaries	TRUE	TRUE	TRUE	Model assumption
impermeable Z-boundaries	FALSE	FALSE	FALSE	Model assumption
impermeable boundary offset [5 µm]	3	3	3	Model assumption
change genotype to	-	-	-	Model assumption
change genotype probability [1/s]	-	-	-	Model assumption

Table 35. Model parameters

Parameter	**Value**	**Reference**
spatial extent X [5 µm]	96	Model assumption
spatial extent Y [5 µm]	20	Model assumption
spatial extent Z [5 µm]	48	Model assumption
simulation time [s]	864000	-
max microbes number	22080	-
stop when all microbes die	TRUE	-
constant initial position	FALSE	-
random generator seed	1	-
proximity raster cell size [5 µm]	1	Model assumption
biofilm Y-position [5 µm]	3	Model assumption
mean flow X [5 µm/s]	0	Model assumption
mean flow Y [5 µm/s]	-0.001	Model assumption
mean flow Z [5 µm/s]	0	Model assumption
flow SD X [5 µm/s]	0	Model assumption
flow SD Y [5 µm/s]	0	Model assumption
flow SD Z [5 µm/s]	0	Model assumption

Estimating resource production rates

Production rates were estimated for different diffusion constants and nanotubes individually.

Diffusion constant D = 0.01 µm²/s

Table 36: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 37: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	0.01	0.01	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 38: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	-
initial abundance	50	50	Model assumption
min biomass [fg dry weight]	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)

biomass density [fg dry weight/µm³]	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	Both types are cooperative and therefore release the resource required by the other
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	Estimated from the respective codon usage in *E.coli* and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	Model assumption
production rate [fg/fg dry weight/s]	1e-5, 1.5e-5, 2e-5, 2.5e-5, 3e-5	9e-6, 1.3e-5, 1.8e-5, 22e-5, 2.6e-5	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 39: Model parameters

Parameter	**Value**	**Reference/Explanation**
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	1000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 0.1 µm²/s

Table 40: Process modules

Process	**time step**
ProximityManager	1000

Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 41: Resource parameters

Parameter	**Value for TRP**	**Value for TYR**	**Reference/Explanation**
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	0.1	0.1	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 42: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	-
initial abundance	50	50	Model assumption
min biomass [fg dry weight]	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	Both types are cooperative and therefore release the resource required by the other

max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	Model assumption
production rate [fg/fg dry weight/s]	1e-5, 1.5e-5, 2e-5, 2.5e-5, 3e-5	9e-6, 1.3e-5, 1.8e-5, 22e-5, 2.6e-5	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 43: Model parameters

Parameter	**Value**	**Reference/Explanation**
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	1000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 1 µm²/s

Table 44: Process modules

Process	**time step**
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000

PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 45: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	1	1	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 46: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	-
initial abundance	50	50	Model assumption
min biomass [fg dry weight]	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	Both types are cooperative and therefore release the resource required by the other
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	Estimated from the respective codon usage in E.coli and its ratio of

			protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.2e-5, 1.8e-5, 2.4e-5, 3e-5, 3.6e-5	1.1e-5, 1.6e-5, 2.1e-5, 2.6e-5, 3.2e-5	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 47: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	1000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 10 µm²/s

Table 48: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	10
Replication	1000
SubstrateUtilization	1000
ConstantProduction	10
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	10
PassiveUptake	10
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	10

Table 49: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-

initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	10	10	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 50: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	-
initial abundance	50	50	Model assumption
min biomass [fg dry weight]	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	Both types are cooperative and therefore release the resource required by the other
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	Model assumption
production rate [fg/fg dry weight/s]	3e-5, 4.5e-5, 6e-5, 7.5e-5, 9e-5	2.6e-5, 4e-5, 5.3e-5, 6.6e-5, 7.9e-5	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	Estimated from maximum growth rates of

	cooperators and non-cooperators (Table 136)

Table 51: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	1000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 100 µm²/s

For high diffusion constants (i.e. 100 and 1000 µm²/s) the spatial resolution was reduced in order to save computational time. Note the different units and parameter values.

Table 52: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	10
Replication	1000
SubstrateUtilization	1000
ConstantProduction	10
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	10
PassiveUptake	10
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	10

Table 53: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/(2.5 µm)³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [(2.5 µm)²/s]	16	16	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open

extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 54: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	-
initial abundance	50	50	Model assumption
min biomass [fg dry weight]	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/(2.5 µm)³]	5859	5859	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	Both types are cooperative and therefore release the resource required by the other
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/(2.5 µm)³]	1.56e-3	1.56e-3	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.5e-4, 2.25e-4, 3e-4, 3.75e-4, 4.5e-4	1.32e-4, 1.98e-4, 2.63e-4, 3.29e-4, 3.95e-4	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 55: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [2.5 µm]	10	large enough for 1000 microbes
spatial extent Y [2.5 µm]	10	large enough for 1000 microbes
spatial extent Z [2.5 µm]	10	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	1000	-
stop when all microbes die	TRUE	-
proximity raster cell size [2.5 µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 1000 µm²/s

For high diffusion constants (i.e. 100 and 1000 µm²/s) the spatial resolution was reduced in order to save computational time. Note the different units and parameter values.

Table 56: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	1
Replication	1000
SubstrateUtilization	1000
ConstantProduction	1
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	1
PassiveUptake	1
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	1

Table 57: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/(2.5 µm)³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [(2.5 µm)²/s]	160	160	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 58: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	-
initial abundance	50	50	Model assumption
min biomass [fg dry weight]	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/(2.5 µm)³]	5859	5859	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	Both types are cooperative and therefore release the resource required by the other
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/(2.5 µm)³]	1.56e-3	1.56e-3	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.25e-3, 1.875e-3, 2.5e-3, 3.125e-3, 3.75e-3	1.098e-3, 1.647e-3, 2.196e-3, 2.744e-3, 3.293e-3	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 59: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [2.5 µm]	10	large enough for 1000 microbes
spatial extent Y [2.5 µm]	10	large enough for 1000 microbes
spatial extent Z [2.5 µm]	10	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	1000	-
stop when all microbes die	TRUE	-

proximity raster cell size [2.5 µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Nanotube-based resource exchange

Table 60: Process modules

Process	**time step**
ProximityManager	1000
Shoving	1000
CellPartition	1000
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
CooperativityFitnessCost	1000
NanoTubeExchange	1000

Table 61: Resource parameters

Parameter	**Reference/Explanation**
-	No resources simulated explicitly

Table 62: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	-
initial abundance	50	50	Model assumption
min biomass [fg dry weight]	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	Both types are cooperative and therefore release the resource required by the other
yield [fg dry weight/fg]	136.3	63.6	Estimated from the respective codon usage in E.coli and its ratio of

			protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	Model assumption
production rate [fg/fg dry weight/s]	3e-5, 3.5e-5, 4e-5, 4.5e-5, 5e-5, 5.5e-5, 6e-5	2.6e-5, 3.1e-5, 3.5e-5, 4e-5, 4.4e-5, 4.8e-5, 5.3e-5	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)
max nanotubes	1,3,5,7	1,3,5,7	Varied parameter
max nanotube length	2,4,6,8	2,4,6,8	Varied parameter
nanotube disconnection probability [1/s]	0.001	0.001	Model assumption

Table 63: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	1000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	8	Must at least equal longest nanotube length
constant initial position	FALSE	-
random generator seed	1	-

Selection coefficient at varied diffusion constant and cooperativity

For each diffusion constant a set of simulations with different cooperativity values were conducted.

Diffusion constant D = 0.01 µm²/s

Table 64: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100

PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 65: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	0.01	0.01	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 66: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of

				protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.9e-5	1.7e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Varied Parameter

Table 67: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 0.1 µm²/s

Table 68: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 69: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	0.1	0.1	Varied parameter

extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 70: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.9e-5	1.7e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Varied Parameter

Table 71: Model parameters

Parameter	Value	Reference/Explanation

spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 1 µm²/s

Table 72: Process modules

Process	**time step**
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 73: Resource parameters

Parameter	**Value for TRP**	**Value for TYR**	**Reference/Explanation**
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	1	1	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 74: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Value for ANCΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry

				weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	2.3e-5	2e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Varied Parameter

Table 75: Model parameters

Parameter	**Value**	**Reference/Explanation**
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 10 µm²/s

Table 76: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	10
Replication	1000
SubstrateUtilization	1000
ConstantProduction	10
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	10
PassiveUptake	10
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	10

Table 77: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	10	10	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 78: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type

max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	5.5e-5	4.8e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Varied Parameter

Table 79: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 100 µm²/s

For high diffusion constants (i.e. 100 and 1000 µm²/s) the spatial resolution was reduced in order to save computational time. Note the different units and parameter values.

Table 80: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	10
Replication	1000
SubstrateUtilization	1000
ConstantProduction	10
InitCluster	-
Growth	1000
Lysis	1000

PassiveRelease	10
PassiveUptake	10
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	10

Table 81: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/(2.5 µm)³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [(2.5 µm)²/s]	16	16	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 82: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	5859	5859	5859	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1.56e-3	1.56e-3	1.56e-3	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage

				in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	2.51e-4	2.2e-4	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Varied Parameter

Table 83: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [2.5 µm]	10	large enough for 1000 microbes
spatial extent Y [2.5 µm]	10	large enough for 1000 microbes
spatial extent Z [2.5 µm]	10	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [2.5 µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 1000 µm²/s

For high diffusion constants (i.e. 100 and 1000 µm²/s) the spatial resolution was reduced in order to save computational time. Note the different units and parameter values.

Table 84: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	1
Replication	1000
SubstrateUtilization	1000
ConstantProduction	1
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	1
PassiveUptake	1
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	1

Table 85: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-

initial concentration [fg/(2.5 µm)³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [(2.5 µm)²/s]	160	160	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 86: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Value for ANCΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	5859	5859	5859	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1.56e-3	1.56e-3	1.56e-3	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	2.791e-3	2.451e-3	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Varied Parameter

Table 87: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [2.5 µm]	10	large enough for 1000 microbes
spatial extent Y [2.5 µm]	10	large enough for 1000 microbes
spatial extent Z [2.5 µm]	10	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [2.5 µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Selection coefficient at varied diffusion constant and resource production

For each diffusion constant a set of simulations with different resource production values were conducted.

Diffusion constant D = 0.01 µm²/s

Table 88: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 89: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	0.01	0.01	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open

extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 90: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.3e-5, 1.6e-5, 1.9e-5, 2.2e-5, 2.5e-5	1.2e-5, 1.4e-5, 1.7e-5, 1.9e-5, 2.2e-5	0	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 91: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 0.1 µm²/s

Table 92: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 93: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	0.1	0.1	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 94: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption

min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.3e-5, 1.6e-5, 1.9e-5, 2.2e-5, 2.5e-5	1.2e-5, 1.4e-5, 1.7e-5, 1.9e-5, 2.2e-5	0	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 95: Model parameters

Parameter	**Value**	**Reference/Explanation**
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-

random generator seed	1	-

Diffusion constant D = 1 µm²/s

Table 96: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100

Table 97: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	1	1	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 98: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)

biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.7e-5, 1.9e-5, 2.3e-5, 2.6e-5, 3e-5	1.4e-5, 1.7e-5, 2e-5, 2.3e-5, 2.6e-5	0	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 99: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 10 µm²/s

Table 100: Process modules

Process	time step

ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	10
Replication	1000
SubstrateUtilization	1000
ConstantProduction	10
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	10
PassiveUptake	10
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	10

Table 101: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/µm³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [µm²/s]	10	10	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 102: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type

max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/µm³]	1e-4	1e-4	1e-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	3.8e-5, 4.7e-5, 5.5e-5, 6.3e-5, 7.1e-5	3.4e-5, 4.1e-5, 4.8e-5, 5.5e-5, 6.3e-5	0	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 103: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 100 µm²/s

For high diffusion constants (i.e. 100 and 1000 µm²/s) the spatial resolution was reduced in order to save computational time. Note the different units and parameter values.

Table 104: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000

Diffusion	10
Replication	1000
SubstrateUtilization	1000
ConstantProduction	10
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	10
PassiveUptake	10
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	10

Table 105: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/(2.5 µm)³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [(2.5 µm)²/s]	16	16	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 106: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/(2.5 µm)³]	5859	5859	5859	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains

releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/(2.5 µm)³]	1.56e-3	1.56e-3	1.56e-3	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.75e-4, 2.13e-4, 2.51e-4, 2.88e-4, 3.26e-4	1.54e-4, 1.87e-4, 2.2e-4, 2.53e-4, 2.86e-4	0	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 107: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [2.5 µm]	10	large enough for 1000 microbes
spatial extent Y [2.5 µm]	10	large enough for 1000 microbes
spatial extent Z [2.5 µm]	10	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [2.5 µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Diffusion constant D = 1000 µm²/s

For high diffusion constants (i.e. 100 and 1000 µm²/s) the spatial resolution was reduced in order to save computational time. Note the different units and parameter values.

Table 108: Process modules

Process	time step
ProximityManager	1000

Shoving	1000
CellPartition	1000
Diffusion	1
Replication	1000
SubstrateUtilization	1000
ConstantProduction	1
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	1
PassiveUptake	1
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	1

Table 109: Resource parameters

Parameter	**Value for TRP**	**Value for TYR**	**Reference/Explanation**
resource name	'TRP'	'TYR'	-
initial concentration [fg/(2.5 µm)³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [(2.5 µm)²/s]	160	160	Varied parameter
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 110: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Value for ANCΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/(2.5 µm)³]	5859	5859	5859	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type

max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/(2.5 µm)³]	1.56e-3	1.56e-3	1.56e-3	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.954e-3, 2.373e-3, 2.791e-3, 3.21e-3, 3.629e-3	1.716e-3, 2.084e-3, 2.451e-3, 2.819e-3, 3.187e-3	0	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.15	0.15	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

Table 111: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [2.5 µm]	10	large enough for 1000 microbes
spatial extent Y [2.5 µm]	10	large enough for 1000 microbes
spatial extent Z [2.5 µm]	10	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [2.5 µm]	1	-
constant initial position	FALSE	-
random generator seed	1	-

Selection coefficient at varied maximum nanotube length and cooperativity

For each maximum nanotube length a set of simulations with different cooperativity values were conducted.

Maximum length L = 2

Table 112: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000

Replication	1000
SubstrateUtilization	1000
ConstantProduction	1000
InitCluster	-
Growth	1000
Lysis	1000
CooperativityFitnessCost	1000
NanoTubeExchange	1000

Table 113: Resource parameters

Parameter	Reference/Explanation
-	No resources simulated explicitly

Table 114: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	3.8e-5	3.3e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)

max nanotube	1	1	1	Varied parameter
max nanotube length [µm]	2	2	2	Varied parameter
nanotube disconnection probability [1/s]	0.001	0.001	0.001	Model assumption

Table 115: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	8	-
constant initial position	FALSE	-
random generator seed	1	-

Maximum length L = 4

Table 116: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Replication	1000
SubstrateUtilization	1000
ConstantProduction	1000
InitCluster	-
Growth	1000
Lysis	1000
CooperativityFitnessCost	1000
NanoTubeExchange	1000

Table 117: Resource parameters

Parameter	Reference/Explanation
-	No resources simulated explicitly

Table 118: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry

				weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	3.5e-5	3e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)
max nanotube	1	1	1	Varied parameter
max nanotube length [µm]	4	4	4	Varied parameter
nanotube disconnection probability [1/s]	0.001	0.001	0.001	Model assumption

Table 119: Model parameters

Parameter	**Value**	**Reference/Explanation**
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	8	-
constant initial position	FALSE	-
random generator seed	1	-

Maximum length L = 6

Table 120: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Replication	1000
SubstrateUtilization	1000
ConstantProduction	1000
InitCluster	-
Growth	1000
Lysis	1000
CooperativityFitnessCost	1000
NanoTubeExchange	1000

Table 121: Resource parameters

Parameter	Reference/Explanation
-	No resources simulated explicitly

Table 122: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	3.4e-5	2.9e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Estimated from maximum growth rates

				of cooperators and non-cooperators (Table 136)
max nanotube	1	1	1	Varied parameter
max nanotube length [µm]	6	6	6	Varied parameter
nanotube disconnection probability [1/s]	0.001	0.001	0.001	Model assumption

Table 123: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	8	-
constant initial position	FALSE	-
random generator seed	1	-

Maximum length L = 8

Table 124: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Replication	1000
SubstrateUtilization	1000
ConstantProduction	1000
InitCluster	-
Growth	1000
Lysis	1000
CooperativityFitnessCost	1000
NanoTubeExchange	1000

Table 125: Resource parameters

Parameter	Reference/Explanation
-	No resources simulated explicitly

Table 126: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40	40	20	Model assumption

min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	3.4e-5	2.9e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.1275, 0.15, 0.1725, 0.195	0.105, 0.1275, 0.15, 0.1725, 0.195	0	Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)
max nanotube	1	1	1	Varied parameter
max nanotube length [µm]	8	8	8	Varied parameter
nanotube disconnection probability [1/s]	0.001	0.001	0.001	Model assumption

Table 127: Model parameters

Parameter	**Value**	**Reference/Explanation**
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	5000	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	8	-
constant initial position	FALSE	-
random generator seed	1	-

In-silico evolution experiment

Diffusion-based resource exchange

Table 128: Process modules

Process	time step
ProximityManager	1000
Shoving	1000
CellPartition	1000
Diffusion	100
Replication	1000
SubstrateUtilization	1000
ConstantProduction	100
InitCluster	-
Growth	1000
Lysis	1000
PassiveRelease	100
PassiveUptake	100
CooperativityFitnessCost	1000
ExtrapolatingResourceBoundaries	100
LongTermExperiment	1000
ParameterMutator	1000

Table 129: Resource parameters

Parameter	Value for TRP	Value for TYR	Reference/Explanation
resource name	'TRP'	'TYR'	-
initial concentration [fg/(2.5 µm)³]	0	0	No initial amino acids were assumed in the medium
diffusion constant [(2.5 µm)²/s]	0.1	0.1	Low diffusion constant chosen to increase likelihood of selection for cooperation
extrapolate X-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Y-boundaries	TRUE	TRUE	The environment was assumed to be open
extrapolate Z-boundaries	TRUE	TRUE	The environment was assumed to be open

Table 130: Microbe parameters

Parameter	Value for EVOΔW	Value for EVOΔY	Value for ANCΔY	Reference/Explanation
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40, 50	40, 50	20, 0	Model assumption. Varied to include and exclude Non-cooperators
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry

				weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/(2.5 µm)³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains
releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
max uptake [fg/fg dry weight/s]	1.82e-6	3.47e-6	3.47e-6	Estimated from maximum growth rate (Table 136) and yield
half-saturation constant [fg/(2.5 µm)³]	1e-4	1e-4	1-4	Used half-saturation constant of tryptophan uptake of S. typhimurium (Ames 1964)
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	1.9e-5	1.7e-5	0	Varied parameter. The ratio between the production rate of EVOΔY and EVOΔW was held constant (Table 136)
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.105, 0.15	0.105, 0.15	0	Varied parameter
mutation parameter	cooperativity	cooperativity	none	Model assumption
min mutation value	0	0	0	Model assumption
max mutation delta	0.1	0.1	0	Model assumption
mutation rate	1e-5	1e-5	0	Model assumption
max mutation value	1	1	0	Model assumption

Table 131: Model parameters

Parameter	**Value**	**Reference/Explanation**
spatial extent X [2.5 µm]	25	large enough for 1000 microbes
spatial extent Y [2.5 µm]	25	large enough for 1000 microbes
spatial extent Z [2.5 µm]	25	large enough for 1000 microbes
simulation time [s]	1.728e6	20 days
max microbes number	9999	this number will never be reached
stop when all microbes die	TRUE	-
proximity raster cell size [2.5 µm]	1	-

constant initial position	FALSE	-
random generator seed	1	-
transfer microbes	100	initial number
Initial processes on transfer	InitCluster	after each transfer microbes will be aggregated into new cluster
transfer at microbe count	1000	Model assumption

Nanotube-based resource exchange

Table 132: Process modules

Process	**time step**
ProximityManager	1000
Shoving	1000
CellPartition	1000
Replication	1000
SubstrateUtilization	1000
ConstantProduction	1000
InitCluster	-
Growth	1000
Lysis	1000
CooperativityFitnessCost	1000
NanoTubeExchange	1000
LongTermExperiment	1000
ParameterMutator	1000

Table 133: Resource parameters

Parameter	**Reference/Explanation**
-	No resources simulated explicitly

Table 134: Microbe parameters

Parameter	**Value for EVOΔW**	**Value for EVOΔY**	**Value for ANCΔY**	**Reference/Explanation**
genotype	'EVO_W'	'EVO_Y'	'ANC_Y'	-
initial abundance	40, 50	40, 50	20, 0	With and without non-cooperators
min biomass [fg dry weight]	250	250	250	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
max biomass [fg dry weight]	500	500	500	Biomass ranges between 100 and 1000 fg dry weight (Neidhardt and Umbarger 1996)
biomass density [fg dry weight/µm³]	375	375	375	At this density the average microbe has a size of 1 µm³
consumes resource	TRP	TYR	TYR	Auxotrophies of the strains

releases resource	TYR	TRP	-	Cooperative types release the resource required by the other cooperative type
yield [fg dry weight/fg]	136.3	63.6	63.6	Estimated from the respective codon usage in E.coli and its ratio of protein to dry weight (Table 136)
maintenance cost [fg/fg dy weight/s]	0	0	0	Model assumption
production rate [fg/fg dry weight/s]	3.8e-5	3.3e-5	0	Fitted parameter
lysis probability	1e-4	1e-4	1e-4	Model assumption
cooperativity	0.15, 0.105	0.15, 0.105	0	Varied parameter. Estimated from maximum growth rates of cooperators and non-cooperators (Table 136)
max nanotube	1	1	1	Model assumption
max nanotube length [µm]	2	2	2	Model assumption
nanotube disconnection probability [1/s]	0.001	0.001	0.001	Model assumption

Table 135: Model parameters

Parameter	Value	Reference/Explanation
spatial extent X [µm]	25	large enough for 1000 microbes
spatial extent Y [µm]	25	large enough for 1000 microbes
spatial extent Z [µm]	25	large enough for 1000 microbes
simulation time [s]	36000	10 hours
max microbes number	9999	-
stop when all microbes die	TRUE	-
proximity raster cell size [µm]	8	-
constant initial position	FALSE	-
random generator seed	1	-
transfer microbes	100	initial number
Initial processes on transfer	InitCluster	after each transfer microbes will be aggregated into new cluster
transfer at microbe count	1000	Model assumption

Parameters not directly used in the models

Table 136. Values that were not directly used in the McComedy models

Parameter	Value	Reference/Explanation
maximum net growth rate of ANCΔY [1/s]	1.21E-04	measured by growing strain in monoculture in minimal medium (MMAB) + tyr
maximum net growth rate of ANCΔW [1/s]	1.47E-04	measured by growing strain in monoculture in minimal medium (MMAB) + trp

maximum net growth rate of EVOΔY [1/s]	8.8E-05	measured by growing strain in monoculture in minimal medium (MMAB) + tyr
maximum net growth rate of EVOΔW [1/s]	1.12E-04	measured by growing strain in monoculture in minimal medium (MMAB) + trp
maximum growth rate of ANCΔY [1/s]	2.21E-04	Estimated from maximum net growth rate and assumed lysis probability of 1e-4 1/s
maximum growth rate of ANCΔW [1/s]	2.47E-04	Estimated from maximum net growth rate and assumed lysis probability of 1e-4 1/s
maximum growth rate of EVOΔY [1/s]	1.88E-04	Estimated from maximum net growth rate and assumed lysis probability of 1e-4 1/s
maximum growth rate of EVOΔW [1/s]	2.12E-04	Estimated from maximum net growth rate and assumed lysis probability of 1e-4 1/s
ratio: *E. coli* protein/dry weight	0.524	Stouthamer (1973)
E. coli codon usage: tyrosine	0.03	Maloy *et al.* (1996)
E. coli codon usage: tryptophan	0.014	Maloy *et al.* (1996)
Ratio: *E coli* tryptophan production/ tyrosine production	0.8782	Flux balance analysis conducted on metabolic model of *E. coli*. The constraint based model was optimized to maximize tyrosine or tryptophan release. The ratio of the resulting release rates was calculated after transforming molarities into mass units.

Computational performance test

Table 137. Process modules

Process	**Time step [ms]**
CellPartition	100, 1000
Diffusion	100, 1000
InitModel	-
PassiveUptake	100, 1000
ProximityManager	100, 1000
Growth	100, 1000
Shoving	100, 1000
SubstrateUtilization	100, 1000
Flow	100, 1000

Table 138. Resource parameters

Parameter	**Value for resource R1**
resource name	'R1'
initial concentration [M/V]	999999
diffusion constant [S^2/T]	1
resource color hue (0 to 1)	0.6
max render concentration [M/V]	1

Table 139. Microbe parameters

Parameter	**Values for type M1**
genotype	'EC1'

initial abundance	10, 100, 1000, 10000
biomass density [M*/V]	375
min biomass [M*]	250
max biomass [M*]	500
consumes resource	'R1'
maintenance cost [1/s]	0
half-saturation constant [fg/µm³]	0.0001
max uptake [1/s]	0.00000186
yield rate	136.3
microbe color hue [0 to 1]	0

Table 140. Model parameters

Parameter	**Value**
spatial extent X [S]	25, 50, 75, 100
spatial extent Y [S]	25
spatial extent Z [S]	25
simulation time [T]	120
max microbes number	99999
mean flow X [S/T]	0
mean flow Y [S/T]	0
mean flow Z [S/T]	0
flow SD X [S/T]	0.1
flow SD Y [S/T]	0.1
flow SD Z [S/T]	0.1
stop when all microbes die	TRUE
constant initial position	FALSE
random generator seed	1
proximity raster cell size [µm]	3

Table 141. Settings

Parameter	**Value**
replicates	1
result directory name	Results
save microbe data every [T]	0
save resource data every [T]	0
save microbe image every [T]	0
save resource images every [T]	0
log frequency [mT]	5000
draw in 3D	TRUE
simultaneous runs	1

List of Figures

Figure 4 Intended workflow when using McComedy. The modeler designs an individual-based model (IBM) by selecting process modules under consideration of the research question and the current understanding of the system. The parameter values that are necessary for the simulation of the selected processes are set by the modeler, e.g. according to experimental data or literature. The resulting IBM generates spatiotemporally explicit data of the modeled microbial system. 40

Figure 5 McComedy can reproduce the results of experiments and simulations by Mitri *et al.* (2015) both quantitatively and qualitatively. Top views on colonies at different initial resource (nutrient) concentrations and degree of heterozygosity over the distance to the inoculum. The unit xLB is defined as the fold-concentration of LB medium. Blue and green colors on colony images indicate the two bacterial strains. White circles on the colony images indicate the inoculum. Red circles indicate where the demixing area begins. Analyses with McComedy were conducted after 45 simulated hours of growth. A: Stylized recreation of top views on colonies at different resource concentrations according to Figures 2a and 4a in (Mitri *et al.* 2015). B: Stylized recreation of the heterozygosity over distance from inoculum and corresponding demixing distances at different resource concentrations according to Figures 2c and 4b in (Mitri *et al.* 2015). Axis labels of distances are not shown as they varied between experimental and simulation results and were of no consequence for the qualitative pattern. **C**: Representive top views on colonies at different resource concentrations in the McComedy IBM. **D**: Heterozygisity over distance from inoculum and estimated demixing distances at different resource concentrations in the McComedy IBM. Images **A** and **B** were recreated due to copyright issues. Refer to Figures 2 and 4 in (Mitri *et al.* 2015) to view the original data. 42

Figure 6 Increased diffusion resulted in an increased demixing distance. Simulations were performed with McComedy. Analysis after 39 simulated hours of growth. A: Top views on representative colonies as simulated using McComedy using different resource diffusion constants. Blue and green colors on colony images indicate the two bacterial strains. White circles on the colony images indicate the size of the inoculum. Red circles indicate where the 44

reached. One generation corresponds to the biomass doubling time of the simulated community.

Figure 10 Spatially-explicit modeling of a bacterial consortia with cooperative cross-feeders and non-cooperative competitors. A: The modeled consortia consist of three different strains of E. coli exchanging essential amino acids: the cooperator CoopΔY (cyan, consuming tyrosine, overproducing tryptophan), the cooperator CoopΔW (blue, consuming tryptophan, overproducing tyrosine), and the non-cooperator NCΔY (orange, consuming tyrosine, no overproduction). B: The diffusion coefficient of metabolites determines its concentration gradient after release (here: tyrosine, visualized by blue color around the blue cell). This gradient translates into an exchange distance determining the access to metabolites for bacteria in the vicinity. Slightly remote bacteria (here: the orange cell) only have access to the metabolite when the exchange distance is sufficiently long (i.e. the diffusion coefficient is sufficiently high, bottom illustration). C: Bacterial consortia are simulated with McComedy, a tool for individual-based modeling of microbial consumer-resource systems (Bogdanowski et al. 2022). In the model simulations, the bacteria form three-dimensional multicellular aggregates. 60

Figure 11 Selection for cooperation decreases with increasing diffusion coefficients. A: Example cross-sections of bacterial aggregates and distribution of tyrosine along X- and Y-axis at Z = 12.5 µm (midpoint of Z-axis) after two hours of growth at different diffusivity of metabolites. Orange circles represent the locations of non-cooperative bacteria. Blue and cyan circles represent the locations of cooperative bacteria. Different shadings of grey represent the local concentration of tyrosine. Concentration values are transformed into relative uptake rates according to Monod-dynamics. This value ranges from 0 (no uptake) to 1 (maximum possible uptake) B: Relationship between the selection coefficient for cooperation (SC_{COOP}) and metabolite diffusion. The grey area ($SC_{COOP} < 0$) indicates an overall advantage for non-cooperators. The line represents the general trend as log-linear regression. Each group consists of 10 replicates. Spearman's rank correlation: ρ = -0.69, P = 1.4e-9, n = 60. 61

biological replicates. Lines indicate mean values and ribbons indicate the standard deviation.

List of Tables

The tables containing model parametrizations (Section A.5) are excluded from this list

Bibliography

Allen B., Gore J. and Nowak M. A. 2013. Spatial dilemmas of diffusible public goods. Elife 2: e01169.

Allison S. D. 2005. Cheaters, diffusion and nutrients constrain decomposition by microbial enzymes in spatially structured environments. Ecology Letters 8: 626-635.

Ames G. F. 1964. Uptake of amino acids by salmonella typhimurium. Archives of Biochemistry and Biophysics 104: 1-18.

Axelrod R. and Hamilton W. D. 1981. The evolution of cooperation. Science 211: 1390-1396.

Banitz T., Gras A. and Ginovart M. 2015. Individual-based modeling of soil organic matter in netlogo: Transparent, user-friendly, and open. Environmental Modelling & Software 71: 39-45.

Bauer E., Zimmermann J., Baldini F., Thiele I. and Kaleta C. 2017. Bacarena: Individual-based metabolic modeling of heterogeneous microbes in complex communities. PLoS Computational Biology 13: e1005544.

Benomar S., Ranava D., Cardenas M. L., Trably E., Rafrafi Y., Ducret A., Hamelin J., Lojou E., Steyer J. P. and Giudici-Orticoni M. T. 2015. Nutritional stress induces exchange of cell material and energetic coupling between bacterial species. Nat Commun 6: 6283.

Bianconi E.*et al.* 2013. An estimation of the number of cells in the human body. Ann Hum Biol 40: 463-471.

Biggs M. B. and Papin J. A. 2013. Novel multiscale modeling tool applied to pseudomonas aeruginosa biofilm formation. PLoS One 8: e78011.

Bogdanowski A., Banitz T., Muhsal L. K., Kost C. and Frank K. 2022. Mccomedy: A user-friendly tool for next-generation individual-based modeling of microbial consumer-resource systems. PLoS Comput Biol 18: e1009777.

Borenstein D. B., Meir Y., Shaevitz J. W. and Wingreen N. S. 2013. Non-local interaction via diffusible resource prevents coexistence of cooperators and cheaters in a lattice model. PLoS One 8: e63304.

Botkin D. B., Janak J. F. and Wallis J. R. 1972. Some ecological consequences of a computer model of forest growth. The Journal of Ecology 60.

Brown J. H., Gillooly J. F., Allen A. P., Savage V. M. and West G. B. 2004. Toward a metabolic theory of ecology. Ecology 85: 1771-1789.

Brunner F. S., Deere J. A., Egas M., Eizaguirre C. and Raeymaekers J. a. M. 2019. The diversity of eco-evolutionary dynamics: Comparing the feedbacks between ecology and evolution across scales. Functional Ecology 33: 7-12.

Bucci V. and Xavier J. B. 2014. Towards predictive models of the human gut microbiome. J Mol Biol 426: 3907-3916.

Bull J. J. and Rice W. R. 1991. Distinguishing mechanisms for the evolution of co-operation. Journal of Theoretical Biology 149: 63-74.

Cai Y. M. 2020. Non-surface attached bacterial aggregates: A ubiquitous third lifestyle. Front Microbiol 11: 557035.

Cavicchioli R.*et al.* 2019. Scientists' warning to humanity: Microorganisms and climate change. Nature Reviews Microbiology 17: 569-586.

Champagnat N. and Meleard S. 2007. Invasion and adaptive evolution for individual-based spatially structured populations. J Math Biol 55: 147-188.

Chesson P. 1990. Macarthur's consumer-resource model. Theoretical Population Biology 37: 26-38.

Clark J. R., Daines S. J., Lenton T. M., Watson A. J. and Williams H. T. P. 2011. Individual-based modelling of adaptation in marine microbial populations using genetically defined physiological parameters. Ecological Modelling 222: 3823-3837.

Costello E. K., Stagaman K., Dethlefsen L., Bohannan B. J. and Relman D. A. 2012. The application of ecological theory toward an understanding of the human microbiome. Science 336: 1255-1262.

Costerton J. W., Lewandowski Z., Caldwell D. E., Korber D. R. and Lappin-Scott H. M. 1995. Microbial biofilms. Annu Rev Microbiol 49: 711-745.

Coulson T., Benton T. G., Lundberg P., Dall S. R. X. and Kendall B. E. 2006. Putting evolutionary biology back in the ecological theatre: A demographic framework mapping genes to communities. Evolutionary Ecology Research 8: 1155-1171.

Coyte K. Z., Schluter J. and Foster K. R. 2015. The ecology of the microbiome: Networks, competition, and stability. Science 350: 663-666.

Curtis T. P. and Sloan W. T. 2005. Microbiology. Exploring microbial diversity--a vast below. Science 309: 1331-1333.

D'souza G., Shitut S., Preussger D., Yousif G., Waschina S. and Kost C. 2018a. Ecology and evolution of metabolic cross-feeding interactions in bacteria. Nat Prod Rep 35: 455-488.

D'souza G., Shitut S., Preussger D., Yousif G., Waschina S. and Kost C. 2018b. Ecology and evolution of metabolic cross-feeding interactions in bacteria. Natural Product Reports 35: 455-488.
D'souza G., Waschina S., Pande S., Bohl K., Kaleta C. and Kost C. 2014. Less is more: Selective advantages can explain the prevalent loss of biosynthetic genes in bacteria. Evolution 68: 2559-2570.
Dal Co A., Van Vliet S., Kiviet D. J., Schlegel S. and Ackermann M. 2020. Short-range interactions govern the dynamics and functions of microbial communities. Nat Ecol Evol 4: 366-375.
Dawkins R. 1976. The selfish gene. New York: Oxford University Press.
Deangelis D. L. and Grimm V. 2014. Individual-based models in ecology after four decades. F1000Prime Reports 6: 39.
Deangelis D. L. and Mooij W. M. 2005. Individual-based modeling of ecological and evolutionary processes. Annual Review of Ecology, Evolution, and Systematics 36: 147-168.
Dobay A., Bagheri H. C., Messina A., Kummerli R. and Rankin D. J. 2014. Interaction effects of cell diffusion, cell density and public goods properties on the evolution of cooperation in digital microbes. J Evol Biol 27: 1869-1877.
Doulcier G., Lambert A., De Monte S. and Rainey P. B. 2020. Eco-evolutionary dynamics of nested darwinian populations and the emergence of community-level heredity. Elife 9.
Drescher K., Nadell C. D., Stone H. A., Wingreen N. S. and Bassler B. L. 2014. Solutions to the public goods dilemma in bacterial biofilms. Curr Biol 24: 50-55.
Dubey G. P. and Ben-Yehuda S. 2011. Intercellular nanotubes mediate bacterial communication. Cell 144: 590-600.
Ducklow H. 2008. Microbial services: Challenges for microbial ecologists in a changing world. Aquatic Microbial Ecology 53: 13-19.
Ducret A., Fleuchot B., Bergam P. and Mignot T. 2013. Direct live imaging of cell-cell protein transfer by transient outer membrane fusion in myxococcus xanthus. Elife 2: e00868.
Einstein A. 1905. Über die von der molekularkinetischen theorie der wärme geforderte bewegung von in ruhenden flüssigkeiten suspendierten teilchen. Annalen der Physik 322: 549-560.
Embree M., Liu J. K., Al-Bassam M. M. and Zengler K. 2015. Networks of energetic and metabolic interactions define dynamics in microbial communities. Proc Natl Acad Sci U S A 112: 15450-15455.
Estrela S., Sanchez-Gorostiaga A., Vila J. C. and Sanchez A. 2021. Nutrient dominance governs the assembly of microbial communities in mixed nutrient environments. Elife 10.
Evans M. R. 2012. Modelling ecological systems in a changing world. Philos Trans R Soc Lond B Biol Sci 367: 181-190.
Fernandez-Veledo S. and Vendrell J. 2019. Gut microbiota-derived succinate: Friend or foe in human metabolic diseases? Rev Endocr Metab Disord 20: 439-447.
Ferrer J., Prats C. and Lopez D. 2008. Individual-based modelling: An essential tool for microbiology. J Biol Phys 34: 19-37.
Fletcher J. A. and Doebeli M. 2009. A simple and general explanation for the evolution of altruism. Proc Biol Sci 276: 13-19.
Frank S. A. 2010. A general model of the public goods dilemma. J Evol Biol 23: 1245-1250.
Gallagher C. A., Chudzinska M., Larsen-Gray A., Pollock C. J., Sells S. N., White P. J. C. and Berger U. 2021. From theory to practice in pattern-oriented modelling: Identifying and using empirical patterns in predictive models. Biol Rev Camb Philos Soc 96: 1868-1888.
Gardner A. 2015. The genetical theory of multilevel selection. J Evol Biol 28: 305-319.
Germerodt S., Bohl K., Luck A., Pande S., Schroter A., Kaleta C., Schuster S. and Kost C. 2016. Pervasive selection for cooperative cross-feeding in bacterial communities. PLoS Comput Biol 12: e1004986.
Ginovart M., López D. and Gras A. 2005. Individual-based modelling of microbial activity to study mineralization of c and n and nitrification process in soil. Nonlinear Analysis: Real World Applications 6: 773-795.
Gogulancea V.*et al.* 2019. Individual based model links thermodynamics, chemical speciation and environmental conditions to microbial growth. Front Microbiol 10: 1871.
Gómez-Mourelo P. and Ginovart M. 2009. The differential equation counterpart of an individual-based model for yeast population growth. Computers & Mathematics with Applications 58: 1360-1369.
González-Cabaleiro R., Lema J. M., Rodríguez J. and Kleerebezem R. 2013. Linking thermodynamics and kinetics to assess pathway reversibility in anaerobic bioprocesses. Energy & Environmental Science 6.
Gorter F. A., Manhart M. and Ackermann M. 2020. Understanding the evolution of interspecies interactions in microbial communities. Philos Trans R Soc Lond B Biol Sci 375: 20190256.

Gras A. and Ginovart M. 2006. Indisim-som, an individual-based model to study shortterm evolutions of carbon and nitrogen pools related to microbial activity in soil organic matter. In*:* ECMS 2006 Proceedings edited by: W Borutzky, A Orsoni, R Zobel, 2006, p. 554-559.

Green J. L.*et al.* 2005. Complexity in ecology and conservation: Mathematical, statistical, and computational challenges. BioScience 55.

Gregory R., Paton R., Saunders J. and Wu Q. H. 2004. Parallelising a model of bacterial interaction and evolution. Biosystems 76: 121-131.

Griffin A. S., West S. A. and Buckling A. 2004. Cooperation and competition in pathogenic bacteria. Nature 430: 1024-1027.

Grimm V., Ayllón D. and Railsback S. F. 2016. Next-generation individual-based models integrate biodiversity and ecosystems: Yes we can, and yes we must. Ecosystems 20: 229-236.

Grimm V. and Berger U. 2016. Structural realism, emergence, and predictions in next-generation ecological modelling: Synthesis from a special issue. Ecological Modelling 326: 177-187.

Grimm V.*et al.* 2006. A standard protocol for describing individual-based and agent-based models. Ecological Modelling 198: 115-126.

Grimm V., Frank K., Jeltsch F., Brandl R., Uchmański J. and Wissel C. 1996. Pattern-oriented modelling in population ecology. Science of The Total Environment 183: 151-166.

Grimm V.*et al.* 2020. The odd protocol for describing agent-based and other simulation models: A second update to improve clarity, replication, and structural realism. Journal of Artificial Societies and Social Simulation 23.

Grimm V., Revilla E., Berger U., Jeltsch F., Mooij W. M., Railsback S. F., Thulke H. H., Weiner J., Wiegand T. and Deangelis D. L. 2005. Pattern-oriented modeling of agent-based complex systems: Lessons from ecology. Science 310: 987-991.

Gunawardena J. 2014. Models in biology: 'Accurate descriptions of our pathetic thinking'. BMC Biol 12: 29.

Hamilton W. D. 1964a. The genetical evolution of social behaviour. I. Journal of Theoretical Biology 7: 1-16.

Hamilton W. D. 1964b. <hamilton1964 jtheoretbiol the genetical evolution of social behaviour.Pdf>. Journal of Theoretical Biology 7: 1-16.

Harcombe W. 2010. Novel cooperation experimentally evolved between species. Evolution 64: 2166-2172.

Harcombe W. R.*et al.* 2014. Metabolic resource allocation in individual microbes determines ecosystem interactions and spatial dynamics. Cell Reports 7: 1104-1115.

Hardin G. 1968. The tragedy of the commons. Science 162: 1243-1248.

Harry E., Monahan L. and Thompson L. 2006. Bacterial cell division: The mechanism and its precison. p. 27-94.

Hauert C. and Doebeli M. 2004. Spatial structure often inhibits the evolution of cooperation in the snowdrift game. Nature 428: 643-646.

Heirendt L.*et al.* 2019. Creation and analysis of biochemical constraint-based models using the cobra toolbox v.3.0. Nat Protoc 14: 639-702.

Hellweger F. L. and Bucci V. 2009. A bunch of tiny individuals—individual-based modeling for microbes. Ecological Modelling 220: 8-22.

Hellweger F. L., Clegg R. J., Clark J. R., Plugge C. M. and Kreft J. U. 2016. Advancing microbial sciences by individual-based modelling. Nature Reviews Microbiology 14: 461-471.

Hellweger F. L., Huang Y. and Luo H. 2018. Carbon limitation drives gc content evolution of a marine bacterium in an individual-based genome-scale model. ISME J 12: 1180-1187.

Henze M., Gujer W., Mino T. and Van Loosedrecht M. 2015. Activated sludge models asm1, asm2, asm2d and asm3. Water Intelligence Online 5: 9781780402369-9781780402369.

Herbert D. 1958. Some principles of continuous culture. In*:* Tunevall, G. (Ed.) Recent progress in microbiology, Stockholm: VII Intern. Congr. for Microbiology, p. 381–396.

Hibbing M. E., Fuqua C., Parsek M. R. and Peterson S. B. 2010. Bacterial competition: Surviving and thriving in the microbial jungle. Nature Reviews Microbiology 8: 15-25.

Hillesland K. L. and Stahl D. A. 2010. Rapid evolution of stability and productivity at the origin of a microbial mutualism. Proc Natl Acad Sci U S A 107: 2124-2129.

Hol F. J., Galajda P., Nagy K., Woolthuis R. G., Dekker C. and Keymer J. E. 2013. Spatial structure facilitates cooperation in a social dilemma: Empirical evidence from a bacterial community. PLoS One 8: e77042.

Ishii S., Kosaka T., Hori K., Hotta Y. and Watanabe K. 2005. Coaggregation facilitates interspecies hydrogen transfer between pelotomaculum thermopropionicum and methanothermobacter thermautotrophicus. Appl Environ Microbiol 71: 7838-7845.

Johnson W. M., Alexander H., Bier R. L., Miller D. R., Muscarella M. E., Pitz K. J. and Smith H. 2020. Auxotrophic interactions: A stabilizing attribute of aquatic microbial communities? FEMS Microbiol Ecol 96.

Kallio P., Pasztor A., Thiel K., Akhtar M. K. and Jones P. R. 2014. An engineered pathway for the biosynthesis of renewable propane. Nat Commun 5: 4731.

Kaltenpoth M., Roeser-Mueller K., Koehler S., Peterson A., Nechitaylo T. Y., Stubblefield J. W., Herzner G., Seger J. and Strohm E. 2014. Partner choice and fidelity stabilize coevolution in a cretaceous-age defensive symbiosis. Proc Natl Acad Sci U S A 111: 6359-6364.

Kang S., Kahan S., Mcdermott J., Flann N. and Shmulevich I. 2014. Biocellion: Accelerating computer simulation of multicellular biological system models. Bioinformatics 30: 3101-3108.

Kaplan J. B. 2010. Biofilm dispersal: Mechanisms, clinical implications, and potential therapeutic uses. J Dent Res 89: 205-218.

Kingma S. A., Santema P., Taborsky M. and Komdeur J. 2014. Group augmentation and the evolution of cooperation. Trends Ecol Evol 29: 476-484.

Klis F. M., De Koster C. G. and Brul S. 2014. Cell wall-related bionumbers and bioestimates of saccharomyces cerevisiae and candida albicans. Eukaryotic Cell 13: 2-9.

Koch H., Lucker S., Albertsen M., Kitzinger K., Herbold C., Spieck E., Nielsen P. H., Wagner M. and Daims H. 2015. Expanded metabolic versatility of ubiquitous nitrite-oxidizing bacteria from the genus nitrospira. Proc Natl Acad Sci U S A 112: 11371-11376.

König S., Vogel H.-J., Harms H. and Worrich A. 2020. Physical, chemical and biological effects on soil bacterial dynamics in microscale models. Frontiers in Ecology and Evolution 8.

Kooijman S. A. and Troost T. A. 2007. Quantitative steps in the evolution of metabolic organisation as specified by the dynamic energy budget theory. Biol Rev Camb Philos Soc 82: 113-142.

Koshy-Chenthittayil S., Archambault L., Senthilkumar D., Laubenbacher R., Mendes P. and Dongari-Bagtzoglou A. 2021. Agent based models of polymicrobial biofilms and the microbiome-a review. Microorganisms 9.

Kramer J. and Meunier J. 2016. Kin and multilevel selection in social evolution: A never-ending controversy? F1000Res 5.

Kreft J. U. 2004. Biofilms promote altruism. Microbiology (Reading) 150: 2751-2760.

Kümmerli R. and Brown S. P. 2010. Molecular and regulatory properties of a public good shape the evolution of cooperation. Proc Natl Acad Sci U S A 107: 18921-18926.

Kümmerli R., Griffin A. S., West S. A., Buckling A. and Harrison F. 2009. Viscous medium promotes cooperation in the pathogenic bacterium pseudomonas aeruginosa. Proc Biol Sci 276: 3531-3538.

Labarthe S., Polizzi B., Phan T., Goudon T., Ribot M. and Laroche B. 2019. A mathematical model to investigate the key drivers of the biogeography of the colon microbiota. J Theor Biol 462: 552-581.

Ladau J. and Eloe-Fadrosh E. A. 2019. Spatial, temporal, and phylogenetic scales of microbial ecology. Trends in Microbiology 27: 662-669.

Lamprecht A.-L.*et al.* 2020. Towards fair principles for research software. Data Science 3: 37-59.

Lardon L. A., Merkey B. V., Martins S., Dotsch A., Picioreanu C., Kreft J. U. and Smets B. F. 2011a. Idynomics: Next-generation individual-based modelling of biofilms. Environ Microbiol 13: 2416-2434.

Lardon L. A., Merkey B. V., Martins S., Dotsch A., Picioreanu C., Kreft J. U. and Smets B. F. 2011b. Idynomics: Next-generation individual-based modelling of biofilms. Environmental Microbiology 13: 2416-2434.

Leigh E. G., Jr. 2010. The group selection controversy. J Evol Biol 23: 6-19.

Lenski R. E. 2017. Experimental evolution and the dynamics of adaptation and genome evolution in microbial populations. ISME J 11: 2181-2194.

Li B., Taniguchi D., Gedara J. P., Gogulancea V., Gonzalez-Cabaleiro R., Chen J., Mcgough A. S., Ofiteru I. D., Curtis T. P. and Zuliani P. 2019. Nufeb: A massively parallel simulator for individual-based modelling of microbial communities. PLoS Computational Biology 15: e1007125.

Lin C., Culver J., Weston B., Underhill E., Gorky J. and Dhurjati P. 2018. Gutlogo: Agent-based modeling framework to investigate spatial and temporal dynamics in the gut microbiome. PLoS One 13: e0207072.

Litchman E., Edwards K. F. and Klausmeier C. A. 2015. Microbial resource utilization traits and trade-offs: Implications for community structure, functioning, and biogeochemical impacts at present and in the future. Front Microbiol 6: 254.

Loreau M. 2010a. From populations to ecosystems.

Loreau M. 2010b. Linking biodiversity and ecosystems: Towards a unifying ecological theory. Philosophical Transactions of the Royal Society B: Biological Sciences 365: 49-60.

Luce R. D. and Raiffa H. 1957. Games and decisions. John Wiley & Sons, Inc.

Lynch M., Field M. C., Goodson H. V., Malik H. S., Pereira-Leal J. B., Roos D. S., Turkewitz A. P. and Sazer S. 2014. Evolutionary cell biology: Two origins, one objective. Proc Natl Acad Sci U S A 111: 16990-16994.

Ma Y., Zhu C., Ma P. and Yu K. T. 2005. Studies on the diffusion coefficients of amino acids in aqueous solutions. Journal of Chemical & Engineering Data 50: 1192-1196.

Mabrouk N., Deffuant G., Tolker-Nielsen T. and Lobry C. 2010. Bacteria can form interconnected microcolonies when a self-excreted product reduces their surface motility: Evidence from individual-based model simulations. Theory Biosci 129: 1-13.

Macarthur R. 1970. Species packing and competitive equilibrium for many species. Theoretical Population Biology 1: 1-11.

Maloy S. V., Steward V. J. and Taylor R. K. 1996. Genetic analysis of pathogenic bacteria. Cold Spring Harbor Laboratory Press, NY.

Marsland R., 3rd, Cui W., Goldford J., Sanchez A., Korolev K. and Mehta P. 2019. Available energy fluxes drive a transition in the diversity, stability, and functional structure of microbial communities. PLoS Comput Biol 15: e1006793.

Marsland R., 3rd, Cui W. and Mehta P. 2020. A minimal model for microbial biodiversity can reproduce experimentally observed ecological patterns. Scientific Reports 10: 3308.

Martin B. T., Zimmer E. I., Grimm V. and Jager T. 2012. Dynamic energy budget theory meets individual-based modelling: A generic and accessible implementation. Methods in Ecology and Evolution 3: 445-449.

Masse D., Cambier C., Brauman A., Sall S., Assigbetse K. and Chotte J. L. 2007. Mior: An individual-based model for simulating the spatial patterns of soil organic matter microbial decomposition. European Journal of Soil Science 58: 1127-1135.

Mccarty N. S. and Ledesma-Amaro R. 2019. Synthetic biology tools to engineer microbial communities for biotechnology. Trends in Biotechnology 37: 181-197.

Min M., Bunt C. R., Mason S. L. and Hussain M. A. 2019. Non-dairy probiotic food products: An emerging group of functional foods. Crit Rev Food Sci Nutr 59: 2626-2641.

Mitri S., Clarke E. and Foster K. R. 2015. Resource limitation drives spatial organization in microbial groups. The ISME Journal 10: 1471-1482.

Momeni B., Brileya K. A., Fields M. W. and Shou W. 2013a. Strong inter-population cooperation leads to partner intermixing in microbial communities. Elife 2: e00230.

Momeni B., Waite A. J. and Shou W. 2013b. Spatial self-organization favors heterotypic cooperation over cheating. Elife 2: e00960.

Monod J. 1949. The growth of bacterial cultures. Annual Review of Microbiology 3: 371-394.

Morris J. J. 2015. Black queen evolution: The role of leakiness in structuring microbial communities. Trends Genet 31: 475-482.

Mugler D. H. and Scott R. A. 1988. Fast fourier transform method for partial differential equations, case study: The 2-d diffusion equation. Computers & Mathematics with Applications 16: 221-228.

Murray J. D. 2002. Mathematical biology. 3 ed, New York: Springer Verlag.

Nadell C. D., Drescher K. and Foster K. R. 2016. Spatial structure, cooperation and competition in biofilms. Nat Rev Microbiol 14: 589-600.

Naylor J., Fellermann H., Ding Y., Mohammed W. K., Jakubovics N. S., Mukherjee J., Biggs C. A., Wright P. C. and Krasnogor N. 2017. Simbiotics: A multiscale integrative platform for 3d modeling of bacterial populations. ACS Synthetic Biology 6: 1194-1210.

Nei M., Maruyama T. and Chakraborty R. 1975. The bottleneck effect and genetic variability in populations. Evolution 29: 1-10.

Neidhardt F. C. and Umbarger H. E. 1996. Escherichia coli and salmonella: Cellular and molecular biology. n. 1, 2 ed, Washington, D.C.: ASM Press.

Nowak M. A., Bonhoeffer S. and May R. M. 1994. Spatial games and the maintenance of cooperation. Proc Natl Acad Sci U S A 91: 4877-4881.

Nowak M. A. and May R. M. 1992. Evolutionary games and spatial chaos. Nature 359: 826-829.

Oliveira N. M., Niehus R. and Foster K. R. 2014. Evolutionary limits to cooperation in microbial communities. Proc Natl Acad Sci U S A 111: 17941-17946.

Olson M. 1965. The logik of collective action: Public goods and the theory of groups. Cambridge, Mass.: Harvard University Press.

Orth J. D., Conrad T. M., Na J., Lerman J. A., Nam H., Feist A. M. and Palsson B. O. 2011. A comprehensive genome-scale reconstruction of escherichia coli metabolism--2011. Mol Syst Biol 7: 535.

Orth J. D., Thiele I. and Palsson B. O. 2010. What is flux balance analysis? Nat Biotechnol 28: 245-248.

Pacciani-Mori L., Giometto A., Suweis S. and Maritan A. 2020. Dynamic metabolic adaptation can promote species coexistence in competitive microbial communities. PLoS Computational Biology 16: e1007896.

Pacheco A. R. and Segre D. 2019. A multidimensional perspective on microbial interactions. FEMS Microbiology Letters 366.
Pande S., Kaftan F., Lang S., Svatos A., Germerodt S. and Kost C. 2016a. Privatization of cooperative benefits stabilizes mutualistic cross-feeding interactions in spatially structured environments. The ISME Journal 10: 1413-1423.
Pande S., Kaftan F., Lang S., Svatos A., Germerodt S. and Kost C. 2016b. Privatization of cooperative benefits stabilizes mutualistic cross-feeding interactions in spatially structured environments. ISME J 10: 1413-1423.
Pande S. and Kost C. 2017. Bacterial unculturability and the formation of intercellular metabolic networks. Trends Microbiol 25: 349-361.
Pande S., Shitut S., Freund L., Westermann M., Bertels F., Colesie C., Bischofs I. B. and Kost C. 2015a. Metabolic cross-feeding via intercellular nanotubes among bacteria. Nat Commun 6: 6238.
Pande S., Shitut S., Freund L., Westermann M., Bertels F., Colesie C., Bischofs I. B. and Kost C. 2015b. Metabolic cross-feeding via intercellular nanotubes among bacteria. Nature Communications 6: 6238.
Pelletier F., Garant D. and Hendry A. P. 2009. Eco-evolutionary dynamics. Philos Trans R Soc Lond B Biol Sci 364: 1483-1489.
Petrova O. E. and Sauer K. 2016. Escaping the biofilm in more than one way: Desorption, detachment or dispersion. Curr Opin Microbiol 30: 67-78.
Platt T. G. and Bever J. D. 2009. Kin competition and the evolution of cooperation. Trends Ecol Evol 24: 370-377.
Popp D. and Centler F. 2020. Mubialsim: Constraint-based dynamic simulation of complex microbiomes. Front Bioeng Biotechnol 8: 574.
Pospisil J.*et al.* 2020. Bacterial nanotubes as a manifestation of cell death. Nat Commun 11: 4963.
Preussger D., Giri S., Muhsal L. K., Ona L. and Kost C. 2020. Reciprocal fitness feedbacks promote the evolution of mutualistic cooperation. Curr Biol 30: 3580-3590 e3587.
Prosser J. I.*et al.* 2007. The role of ecological theory in microbial ecology. Nature Reviews Microbiology 5: 384-392.
R-Core-Team. 2021. R: A language and environment for statistical computing [Online]. Vienna, Austria: R Foundation for Statistical Computing. Available: https://www.r-project.org/.
Railsback S. F. 2001. Concepts from complex adaptive systems as a framework for individual-based modelling. Ecological Modelling 139: 47-62.
Railsback S. F. and Grimm V. 2019. Agent-based and individual-based modeling: A practical introduction. 2 ed, Princeton and Oxford: Princeton University Press.
Rankin D. J., Bargum K. and Kokko H. 2007. The tragedy of the commons in evolutionary biology. Trends Ecol Evol 22: 643-651.
Reynolds J. F. and Acock B. 1997. Modularity and genericness in plant and ecosystem models. Ecological Modelling 94: 7-16.
Rice K. C. and Bayles K. W. 2008. Molecular control of bacterial death and lysis. Microbiol Mol Biol Rev 72: 85-109, table of contents.
Romero-Mujalli D., Jeltsch F. and Tiedemann R. 2018. Individual-based modeling of eco-evolutionary dynamics: State of the art and future directions. Regional Environmental Change 19: 1-12.
Rykiel E. J. 1996. Testing ecological models: The meaning of validation. Ecological Modelling 90: 229-244.
Sachs J. L., Mueller U. G., Wilcox T. P. and Bull J. J. 2004. The evolution of cooperation. Q Rev Biol 79: 135-160.
Santos E. C., Armas E. D., Crowley D. and Lambais M. R. 2014. Artificial neural network modeling of microbial community structures in the atlantic forest of brazil. Soil Biology and Biochemistry 69: 101-109.
Schlüter M., Müller B. and Frank K. 2019. The potential of models and modeling for social-ecological systems research: The reference frame modses. Ecology and Society 24.
Schmitz R. A. 2008. Ecological models and dynamics: An interactive textbook. New York, NY: Garland Science.
Schramski J. R., Dell A. I., Grady J. M., Sibly R. M. and Brown J. H. 2015. Metabolic theory predicts whole-ecosystem properties. Proc Natl Acad Sci U S A 112: 2617-2622.
Sharma S. and Steuer R. 2019. Modelling microbial communities using biochemical resource allocation analysis. J R Soc Interface 16: 20190474.
Shashkova T., Popenko A., Tyakht A., Peskov K., Kosinsky Y., Bogolubsky L., Raigorodskii A., Ischenko D., Alexeev D. and Govorun V. 2016. Agent based modeling of human gut microbiome interactions and perturbations. PLoS One 11: e0148386.
Shimoyama T., Kato S., Ishii S. and Watanabe K. 2009. Flagellum mediates symbiosis. Science 323: 1574.
Stillman R. A., Railsback S. F., Giske J., Berger U. and Grimm V. 2015. Making predictions in a changing world: The benefits of individual-based ecology. Bioscience 65: 140-150.

Stouthamer A. H. 1973. A theoretical study on the amount of atp required for synthesis of microbial cell material. Antonie Van Leeuwenhoek 39: 545-565.
Stump S. M., Johnson E. C. and Klausmeier C. A. 2018a. How leaking and overproducing resources affect the evolutionary robustness of cooperative cross-feeding. J Theor Biol 454: 278-291.
Stump S. M., Johnson E. C., Sun Z. and Klausmeier C. A. 2018b. How spatial structure and neighbor uncertainty promote mutualists and weaken black queen effects. J Theor Biol 446: 33-60.
Team R. C. 2020. R: A language and environment for statistical computing [Online]. Available: https://www.R-project.org/.
Trunk T., Khalil H. S. and Leo J. C. 2018. Bacterial autoaggregation. AIMS Microbiol 4: 140-164.
Uppal G. and Vural D. C. 2018. Shearing in flow environment promotes evolution of social behavior in microbial populations. Elife 7.
Uppal G. and Vural D. C. 2020. Evolution of specialized microbial cooperation in dynamic fluids. J Evol Biol 33: 256-269.
Van Der Wal A., Tecon R., Kreft J. U., Mooij W. M. and Leveau J. H. 2013. Explaining bacterial dispersion on leaf surfaces with an individual-based model (phyllosim). PLoS One 8: e75633.
Vila J. C. C., Jones M. L., Patel M., Bell T. and Rosindell J. 2019. Uncovering the rules of microbial community invasions. Nat Ecol Evol 3: 1162-1171.
Wakelin S. A. 2018. Managing soil microbiology: Realising opportunities for the productive land-based sectors. New Zealand Journal of Agricultural Research 61: 358-376.
Wechsler T., Kummerli R. and Dobay A. 2019. Understanding policing as a mechanism of cheater control in cooperating bacteria. J Evol Biol 32: 412-424.
Wickham H. 2016. Ggplot2: Elegant graphics for data analysis. Springer-Verlag New York.
Widder S.*et al.* 2016a. Challenges in microbial ecology: Building predictive understanding of community function and dynamics. The ISME Journal 10: 2557-2568.
Widder S.*et al.* 2016b. Challenges in microbial ecology: Building predictive understanding of community function and dynamics. ISME J 10: 2557-2568.
Wilensky U. 1999. Netlogo. http://ccl.northwestern.edu/netlogo/: Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL.
Wilkinson M. D.*et al.* 2016. The fair guiding principles for scientific data management and stewardship. Sci Data 3: 160018.
Williams G. C. 1966. Adaptation and natural selection. Princeton, New Jersey: Princeton University Press.
Wilson D. S. 1987. Altruism in mendelian populations derived from sibling groups: The haystack model revisited. Evolution 41.
Wimpenny J. W. and Kreft J. U. 2001. Effect of eps on biofilm structure and function as revealed by an individual-based model of biofilm growth. Water Science and Technology 43: 135-135.
Yamamura N., Higashi M., Behera N. and Yuichiro Wakano J. 2004. Evolution of mutualism through spatial effects. J Theor Biol 226: 421-428.
Zaccaria M., Dedrick S. and Momeni B. 2017. Modeling microbial communities: A call for collaboration between experimentalists and theorists. Processes 5.
Zakharova L., Meyer K. M. and Seifan M. 2019. Trait-based modelling in ecology: A review of two decades of research. Ecological Modelling 407.
Zelezniak A., Andrejev S., Ponomarova O., Mende D. R., Bork P. and Patil K. R. 2015. Metabolic dependencies drive species co-occurrence in diverse microbial communities. Proc Natl Acad Sci U S A 112: 6449-6454.
Zengler K. and Zaramela L. S. 2018. The social network of microorganisms - how auxotrophies shape complex communities. Nat Rev Microbiol 16: 383-390.
Zhang H. and Perc M. 2016. Evolution of conditional cooperation under multilevel selection. Sci Rep 6: 23006.

Acknowledgements

I would like to acknowledge and express my deepest gratitude to my supervisors Dr. Thomas Banitz, Prof. Christian Kost, and Prof. Karin Frank, who made this work possible. Thomas Banitz provided a helpful hand in an incredible number of ways. He taught me a lot about scientific research, knew an answer to all of my questions, engaged in very productive discussions about my ideas, gave valuable advice on writing manuscripts and this thesis, and provided mental support whenever necessary. Christian Kost introduced me to the world of microbial ecology and evolution, contributed to my research with brilliant ideas, and always provided incredibly helpful feedback. Karin Frank enthusiastically shared an impressive amount of knowledge, inspired me in countless vivid discussions, and always knew a way out of a tricky situation.

I would also like to thank Linea Muhsal. We were working in close collaboration and she complemented my theoretical studies perfectly with her experimental approaches. She was very supportive and always had good ideas. Further thanks goes to Adam Reichold for his extremely useful advice on software development. Moreover, I would like to thank all members from OESA, KostLab, and EvoCell for fruitful discussions, a great working environment, exciting coffee breaks, and numerous friendships.

My gratitude goes also to Onsabrück University for funding this thesis as a project of the graduate school EvoCell and to the Helmholtz Centre for Environmental Research – UFZ for providing additional financial support and granting me access to the graduate school HIGRADE.

Finally, I would like to thank my friends and family for the tremendous mental support, the comfort at stressful times, and the sometimes sorely needed distraction from work. I particularly thank Susan Kang for never letting me down and making me smile, even during the most difficult times.