

DISS. ETH NO. XX

Eco-evolutionary dynamics in complex adaptive systems

Bridging theory and data

A thesis submitted to attain the degree of
DOCTOR OF SCIENCES of ETH ZURICH
(Dr. sc. ETH Zurich)

presented by

VICTOR BOUSSANGE

M.Sc. Energy and environmental sciences, Institut National des Sciences
Appliquées de Lyon
born January 10th, 1995
citizen of Bordeaux, France

accepted on the recommendation of

Prof. Dr. Loïc Pellissier, (doctoral thesis supervisor), examiner
Prof. Dr. Didier Sornette, co-examiner
Prof. Dr. Arnulf Jentzen, co-examiner
Prof. Dr. Samir Suweis, co-examiner

2022

Eco-evolutionary dynamics in complex adaptive systems

Bridging theory and data

Victor Boussange

March 18, 2022
Version: 0.0.1

Victor Boussange

Eco-evolutionary dynamics in complex adaptive systems

Bridging theory and data

March 18, 2022

Reviewers: Didier Sornette Samir Suweis and Arnulf Jentzen

Supervisors: Loïc Pellissier

ETH Zürich

Ecology Landscape and Evolution

Institute of Terrestrial Ecosystems

Department of Environmental Sciences

Universitätstrasse 18

8055 and Zürich

Summary

- Since life emerged 4 billion years ago, its complexity has evolved.
 - Complex Systems (CS) are generally defined as a category of dynamical systems composed of many individual entities, be they biological, socio-cultural or economic, spatially organised and interacting locally in a nonlinear way. The adjective Adaptive is used to define CS which are subjected to evolutionary mechanisms (Levin [2002]). These include the biosphere, socio-cultural systems and economical systems. The agents adapt to local conditions and are subjected to selection processes acting at a macro level.
- In this thesis, a novel framework for bridging mechanistic models of CAS and data is presented.
 - An eco-evolutionary model of interacting organisms is theoretically investigated
 - * to understand the emergence of biodiversity in complex landscapes
 - A set of tools are developed to
 - Those tools are used to investigate the processes that drive the macroscopic dynamics of economies across countries
- It is shown that bridging theoretical models and data deepens our current understanding of processes and can bring a new perspective.
 - Bridging disciplines provides a remarkably clear understanding of universal mechanisms that have shaped the economics dynamics.
- This thesis moves beyond the dichotomy between theoretical and data science approaches and provides a novel framework for formalizing and exploring multiple hypotheses and reconstructions associated with the processes that drive CAS. Model comparison with empirical data serves as hindcast, which might inform evolutionary trajectories. By advancing our understanding on the processes that dictate the dynamics of CAS, we can better anticipate the radical changes that we will face in the next decades

Résumé

- Same as above, but in french

Acknowledgement

I thank my parents, my sister, my friends, and my beloved Flora.

Contents

1	Introduction	1
1.1	History of modelling	1
1.2	Complex adaptive systems	1
1.3	Biological systems	2
1.4	Economic systems	2
1.5	Models	2
1.6	Thesis Structure	2
I	An eco-evolutionary model on spatial graphs	5
2	Neutral and adaptive differentiation in spatial graphs	9
2.1	Introduction	10
2.2	Results	12
2.2.1	Eco-evolutionary model on spatial graphs	12
2.2.2	Deterministic approximation of the population dynamics under no selection	13
2.2.3	Effect of graph topology on neutral differentiation under no selection	15
2.2.4	Deterministic approximation of the population dynamics and adaptation under heterogeneous selection	18
2.2.5	Effect of graph topology on adaptive differentiation under heterogeneous selection	20
2.2.6	Effect of habitat assortativity on neutral differentiation under heterogeneous selection	21
2.3	Discussion	25
2.4	Conclusion	27
2.5	Methods	27
2.5.1	Mean field approximation	27
2.5.2	Adaptive dynamics on graphs	28
2.5.3	Numerical simulations	29

2.6	Acknowledgements	30
2.7	Data accessibility	30
2.A	Supplementary Methods	30
2.A.1	Mathematical construction of the model	30
2.A.2	Deterministic approximation	31
2.A.3	Trait-dependent competition	34
2.A.4	Derivation of the habitat assortativity metric r_Θ in binary environments	34
II	Scientific machine learning for eco-evolutionary modelling	53
3	Deep learning approximations for non-local nonlinear PDEs with Neumann boundary conditions	59
3.1	Introduction	60
3.2	Machine learning-based approximation method in a special case	62
3.2.1	Partial differential equations (PDEs) under consideration	62
3.2.2	Simulation of time discrete reflected processes	63
3.2.3	Description of the proposed approximation method in a special case	63
3.3	Machine learning-based approximation method in the general case	65
3.3.1	PDEs under consideration	65
3.3.2	Description of the proposed approximation method in the general case	65
3.4	Multilevel Picard approximation method for non-local PDEs	67
3.4.1	Description of the proposed approximation method	67
3.4.2	Examples for the approximation method	68
3.5	Numerical simulations	70
3.5.1	Fisher–KPP PDEs with Neumann boundary conditions	73
3.5.2	Non-local competition PDEs	75
3.5.3	Non-local sine-Gordon type PDEs	75
3.5.4	Replicator-mutator PDEs	76
3.5.5	Allen–Cahn PDEs with conservation of mass	84
4	HighDimPDE.jl: A Julia package for solving high dimensional, non-local, non-linear PDEs	87
5	Interpretable Machine Learning for forecasting dynamical processes in ecosystems at different ecological states	89

III Briding eco-evolutionary models and data	91
6 Econobiology: quantifying interactions and evolution in economic systems	93
7 Discussion	95
7.1 Conclusion	95
Declaration	99

Introduction

“ Le bout du monde et le fond du jardin
contiennent la même quantité de merveilles ”

— Christian Bobin
(French poet)

1.1 History of modelling

Nature has been fascinating since the beginning of humankind. Human's curiosity lead us to propose models capturing our belief on how things work. More than three centuries ago, Isaac Newton became the revered founder of modern Mechanics due to his intuition, gathered by empirical evidence, about a possible mathematical formalization for the law of universal gravitation. [Equations2021]

1.2 Complex adaptive systems

Complex Systems (CS) are generally defined as a category of dynamical systems composed of many individual entities, be they biological, socio-cultural or economic, spatially organised and interacting locally in a nonlinear way. The adjective Adaptive is used to define CS which are subjected to evolutionary mechanisms (Levin [2002]). These include the biosphere, socio- cultural systems and economical systems. The agents adapt to local conditions and are subjected to selection processes acting at a macro level.

1.3 Biological systems

1.4 Economic systems

1.5 Models

- Agent Based model: hard to scale up
- PDE: hard to scale up
- Machine learning: scale up

1.6 Thesis Structure

Part I

An eco-evolutionary model on spatial graphs

It is not clear how landscape connectivity and habitat heterogeneity influence differentiation in biological populations. To obtain a mechanistic understanding of underlying processes, we construct an individual-based model that accounts for eco-evolutionary and spatial dynamics over graphs. Individuals possess both neutral and adaptive traits, whose co-evolution results in differentiation at the population level. In agreement with empirical studies, we show that characteristic length, heterogeneity in degree and habitat assortativity drive differentiation. By using analytical tools that permit a macroscopic description of the dynamics, we further link differentiation patterns to the mechanisms that generate them. This part provides support for a mechanistic understanding of how landscape features affect diversification.

Part II

Scientific machine learning for eco-evolutionary modelling

It is a daunting task to obtain an agreement between mechanistic models and real world systems. In particular, there is a need to account for the dimensionality of the evolutionary and spatial structures over which agents interact and evolve. Furthermore, the calibration of such models is difficult. To address the difficulties that arise due to the dimensionality of models, we develop two numerical methods to solve high-dimensional non-local nonlinear PDES that arise in eco-evolutionary models. We implement those methods in a software, `HighDimPDE.jl`, that integrates within

an open source ecosystem for Scientific Machine Learning in the Julia programming language. We further present a scheme to estimate the parameters of a mechanistic model from empirical data sets. We show with analytical arguments that the use of different shallow time series allows for a better estimation than a unique, possibly deeper time series. This part provides ready-to-use modeling tools to address the intrinsic complexity of complex adaptive systems.

Part III

Bridging eco-evolutionary models and data

Despite evidences that alike biological systems, economic systems are complex adaptive systems that continuously adapt and experience evolutionary processes, economists have discarded biological models and have rather relied on mechanistic models inspired from physics. Building upon an analogy between economic sectors and biological functional groups, we use a biological model to quantitatively investigate whether eco-evolutionary processes characterise the dynamics of economic sectors. Overall, we find that interactions across economic sectors, evolution of new economic sectors, and international transfers play a major role in the dynamics of economic sectors at the national level. The significance and the strength of such processes strongly vary across countries and correlate with standard macroeconomic indices such as the Economic Complexity Index. We relate such patterns to documented patterns in ecology and evolution. This part provides a new perspective on the understanding of the dynamics of economic systems.

Part I

An eco-evolutionary model on spatial
graphs

“ A picture is worth a thousand words. An interface is worth a thousand pictures.

— Ben Shneiderman

(Professor for Computer Science)

Neutral and adaptive differentiation in spatial graphs

by Victor Boussange ^{a,b,1} and Loïc Pellissier ^{a,b,2}

^a Swiss Federal Research Institute WSL, CH-8903 Birmensdorf, Switzerland

^b Landscape Ecology, Institute of Terrestrial Ecosystems, Department of Environmental System Science, ETH Zürich, CH-8092 Zürich, Switzerland

Preprint: doi:10.1101/2021.07.06.451404

Revision requested from Communications Biology.

Differentiation mechanisms are influenced by the properties of the landscape over which individuals interact, disperse and evolve. Here, we investigate how habitat connectivity and habitat heterogeneity affect phenotypic differentiation by formulating a stochastic eco-evolutionary model where individuals are structured over a spatial graph. We combine analytical insights into the eco-evolutionary dynamics with numerical simulations to understand how the graph topology and the spatial distribution of habitat types affect differentiation. We show that not only low connectivity but also heterogeneity in connectivity promotes neutral differentiation, due to increased competition in highly connected vertices. Habitat assortativity, a measure of habitat spatial auto-correlation in graphs, additionally drives differentiation under habitat-dependent selection. While assortative graphs systematically amplify adaptive differentiation, they can foster or depress neutral differentiation depending on the migration regime. By formalising the eco-evolutionary and spatial dynamics of biological populations on graphs, our study establishes fundamental links between landscape features and phenotypic differentiation.

biodiversity | eco-evolutionary dynamics | co-evolution | adaptation | phenotypic differentiation | heterogeneous habitats | individual-based model

Author contributions: V.B. and L.P. designed research; V.B. performed research; V.B. and L.P. wrote the paper.

The authors declare no competing interest.

2.1 Introduction

Biodiversity results from differentiation processes influenced by the features of the landscape over which populations are distributed [Hubbel2001]. The documentation of high levels of species diversity in mountain regions and riverine systems suggests that complex connectivity patterns and habitat heterogeneity foster differentiation [Rahbek2019a, Ding2020a, Dias2014, Guegan1998]. However, hypotheses formulated based on empirical evidence should be complemented by mechanistic models to crystallise a causal understanding between processes and patterns [Levin2002]. While the number of simulation studies is growing steadily [Cabral2017], such studies often lack a mathematical formalism to facilitate the interpretation of the model outcomes by providing an analytical underpinning to the simulation results [Lion2016].

Phenotypic differentiation processes emerge as a result of mutation, selection and migration and can be classified as neutral or adaptive [Holderegger2006]. Neutral differentiation is initiated by the stochastic drift of local phenotypes when spatial isolation and limited dispersal create barriers to gene flow, allowing distinct phenotypes to emerge in spatially structured populations [Slatkin1993a]. In contrast, adaptive differentiation results from heterogeneous selection, which promotes distinct, locally well-adapted phenotypes in populations occupying patches with different habitat conditions [Dieckmann1999]. The evolution of neutral phenotypes and of adaptive phenotypes are not independent, as selective forces can indirectly select for those neutral phenotypes that happen to be linked to the fittest adaptive phenotypes, a mechanism called the “hitchhiking effect” [Kaplan1989]. Moreover, selection can generate barriers to gene flow between populations in heterogeneous habitat landscapes [Orsini2013, Wang2014], a phenomenon coined “isolation by environment”, which can amplify neutral differentiation. How neutral processes, adaptive processes and their interplay are affected by landscape features is difficult to comprehend without a formalised mechanistic model [GARANT2007].

Models link patterns to processes [Levin2002], and the explicit representation of the landscape within an eco-evolutionary model can lead to a causal understanding of how landscape features shape differentiation. Spatial graphs provide a convenient mathematical representation of landscapes, where vertices represent suitable habitats hosting populations, and edges capture the connectivity between habitats [Dale2010]. Under ecological dynamics, metapopulation models have been used to study the role of graph topology in

the persistence and stability of metapopulation [Holland2008, Gilarranz2012, Mari2014, Gravel2016] and community diversity [Carrara2012, Thompson2017, Suzuki2021]. Evolutionary mechanisms are nevertheless fundamental drivers of diversity, and should therefore be explicitly integrated into models [Pelletier2009]. Evolutionary game theory explores how graph topology impacts the fixation probability and the fixation time of a mutated phenotype [Tkadlec2019]. However, the framework does not consider the continuous accumulation of mutations, and is therefore not suited to addressing the emergence of phenotypic differentiation. By combining a metapopulation model with a model of neutral evolution, [Economou2007, Economou2010] investigated how graph topology affects neutral diversity. Their approach demonstrated the key role of topological properties in shaping diversity, and its predictions could be matched with empirical data from e.g. river basins [Muneepeerakul2008]. Nonetheless, diversity results from the combination of neutral and adaptive processes developing at the population level. A first principles modelling approach considering spatial graphs, but also building upon the elementary processes of ecological interactions, reproduction, mutation and migration may therefore be promising to investigate the emergence of diversity.

Stochastic models for structured populations, rooted in the microscopic description of individuals, offer a generic framework for modelling eco-evolutionary dynamics [Champagnat2006, Bansaye2015]. In particular, these models can capture the interplay between population dynamics, spatial dynamics and phenotypic evolution, while providing a rigorous set-up for analytical investigation. By anchoring this modelling paradigm in a mathematical framework, the work of Champagnat et al. [Champagnat2006] generalises models of population genetics [Burger2000] (investigating the evolution of the frequencies of alleles) and quantitative genetics [Slatkin1978, Lande1991, Nagylaki1994] (investigating the evolution of phenotypic traits), which stimulated research into the link between spatial population structure and neutral differentiation. The framework embraces density-dependent selection, which could explain the emergence of phenotypic differentiation from competition processes [Dieckmann1999], and how spatial segregation can emerge as a byproduct of these adaptive processes along environmental gradients [Doebeli2003]. Related models have addressed the effects of landscape dynamics and habitat heterogeneity on adaptive differentiation, providing mathematical insights into the dynamics [Meszena1997, Aguilee2012, Debarre2013, Wickman2017, Polechova2018, Mirrahimi2020]. Because it accounts for finite population size, the baseline model of Champagnat et al. [Champagnat2006] can also capture neutral differentiation dynamics and therefore the coupling between neutral and adaptive processes [Billiard2015, Anceschi2019]. Nonetheless, the aforementioned studies were not spatially explicit [Billiard2015, Anceschi2019] or they assumed regular spatial structures (regular graphs [Meszena1997, Aguilee2012, Debarre2013, Mirrahimi2020] or continuous space [Doebeli2003, Wickman2017, Polechova2018]), therefore not addressing the role of the spatial complexity of landscapes. A stochastic individual-based model using spatial graphs as a representation of the landscape could help formalise fundamental links between landscape features and phenotypic differentiation.

A key challenge is to understand how individual dynamics result in the emergence of differentiation in complex landscapes [Manel2003]. Here, we investigate how complex

connectivity patterns and habitat heterogeneity affect both neutral and adaptive phenotypic differentiation by constructing an individual-based model (IBM) that accounts for eco-evolutionary dynamics on spatial graphs. The individuals disperse between habitat patches and possess co-evolving neutral and adaptive traits. The finite size of local populations generates neutral differentiation by inducing a stochastic drift in the neutral trait evolution, while heterogeneous selection gives rise to adaptive differentiation. Macroscopic properties of the model are analytically tractable, and we obtain a deterministic approximation of population size and adaptive trait dynamics which connects the emerging patterns to the graph properties that generate them. However, neutral differentiation is stochastic by nature, which complicates its analytical underpinning. We therefore rely on numerical simulations of the IBM to measure the effect of graph topology on neutral differentiation. In the case where heterogeneous selection is absent, we investigate how graph topology affects neutral differentiation. In the case of heterogeneous selection, we investigate how the graph topology, in combination with the spatial distribution of habitat types, affects levels of (i) adaptive and (ii) neutral differentiation. By combining analytical methods with numerical simulations, we expect to identify graph properties that determine the level of differentiation. Overall, our study establishes causal links between landscape properties and population differentiation and contributes to a fundamental understanding of how landscape features promote biodiversity.

2.2 Results

2.2.1 Eco-evolutionary model on spatial graphs

We establish an individual-based model (IBM) where individuals are structured over a trait space and a graph representing a landscape. For the sake of simplicity, we consider the case of asexual reproduction and haploid genetics [**Champagnat2006**]. Individuals die, reproduce, mutate and migrate in a stochastic fashion, which together results in macroscopic properties. The formulation of the stochastic IBM allows an analytical description of the dynamics at the population level, which links emergent properties to the elementary processes that generate them.

The trait space $\mathcal{X} \subseteq \mathbb{R}^d$ is continuous and can be split into a neutral trait space \mathcal{U} and an adaptive trait space \mathcal{S} . We refer to neutral traits $u \in \mathcal{U}$ as traits that are not under selection, in contrast to adaptive traits $s \in \mathcal{S}$, which experience selection. The graph denoted by G is composed of a set of vertices $\{v_1, v_2, \dots, v_M\}$ that correspond to habitat patches (suitable geographical areas), and a set of edges that constrain the movement of individuals between the habitat patches. We use the original measure of genetic differentiation for quantitative traits Q_{ST} (standing for Q -statistics) in the case of haploid populations [**Lande1992**, **WHITLOCK2008**]. We denote the neutral trait value of the k -th individual on v_i as $u_k^{(i)}$, the number of individuals on v_i as $N^{(i)}$, the mean neutral trait on v_i as $\bar{u}^{(i)}$, and the mean

neutral trait in the metapopulation as \bar{u} . It follows that we quantify neutral differentiation $Q_{ST,u}$ as

$$Q_{ST,u} = \sigma_{B,u}^2 / (\sigma_{B,u}^2 + \sigma_{W,u}^2) \quad (2.1)$$

where $\sigma_{B,u}^2 = \mathbb{E} \left[\frac{1}{M} \sum_i (\bar{u}^{(i)} - \bar{u})^2 \right]$ denotes the expected neutral trait variance between the vertices and $\sigma_{W,u}^2 = \frac{1}{M} \sum_i \mathbb{E} \left[\frac{1}{N^{(i)}} \sum_k (u_k^{(i)} - \bar{u}^{(i)})^2 \right]$ denotes the average expected neutral trait variance within vertices. We similarly quantify adaptive differentiation $Q_{ST,s}$.

Following the Gillespie update rule [Gillespie1976], individuals with trait $x_k \in \mathcal{X}$ on vertex v_i are randomly selected to give birth at rate $b^{(i)}(x_k)$ and die at rate $d(N^{(i)}) = N^{(i)}/K$, where K is the local carrying capacity. The definition of d therefore captures competition, which is proportional to the number of individuals on a vertex and does not depend on the individuals' traits (we relax this assumption later on). The offspring resulting from a birth event inherits the parental traits, which can independently be affected by mutations with probability μ . A mutated trait differs from the parental trait by a random change that follows a normal distribution with variance σ_μ^2 (corresponding to the continuum of alleles model [Kimura1964]). The offspring can further migrate to neighbouring vertices by executing a simple random walk on G with probability m . Under the setting with no selection, individuals are only characterised by neutral traits so that $\mathcal{X} = \mathcal{U}$. For individuals on a vertex with trait $x_k \equiv u_k$ we define $b^{(i)}(x_k) \equiv b$, so that the birth rate is constant. This ensures that neutral traits do not provide any selective advantage. Under the setting with heterogeneous selection, each vertex of the graph v_i is labelled by a habitat type Θ_i that specifies the optimal adaptive trait value on v_i . It follows that, for individuals with traits $x_k = (u_k, s_k) \in \mathcal{U} \times \mathcal{S}$ on v_i , we define

$$b^{(i)}(x_k) \equiv b^{(i)}(s_k) = b(1 - p(s_k - \Theta_i)^2) \quad (2.2)$$

where p is the selection strength [Mirrahimi2020]. This ensures that the maximum birth rate on v_i is attained for $s_k = \Theta_i$, which results in a differential advantage that acts as an evolutionary stabilising force. In the following we assume that habitat types are binary and symmetric, so that $\Theta_i \in \{\theta_\bullet, \theta_\circ\}$, $\theta_\circ = -\theta_\bullet = \theta$, where θ can be viewed as the habitat heterogeneity [Mirrahimi2020].

2.2.2 Deterministic approximation of the population dynamics under no selection

The model can be formulated as a measure-valued point process ([Bansaye2015] and Supplementary Methods). Under this formalism, we demonstrate in Supplementary Methods how the population size and the trait dynamics show a deterministic behaviour when a stabilising force dampens the stochastic fluctuations. This makes it possible to express the dynamics of the macroscopic properties with deterministic differential equations, connecting emergent patterns to the processes that generate them. In particular, under the setting of no

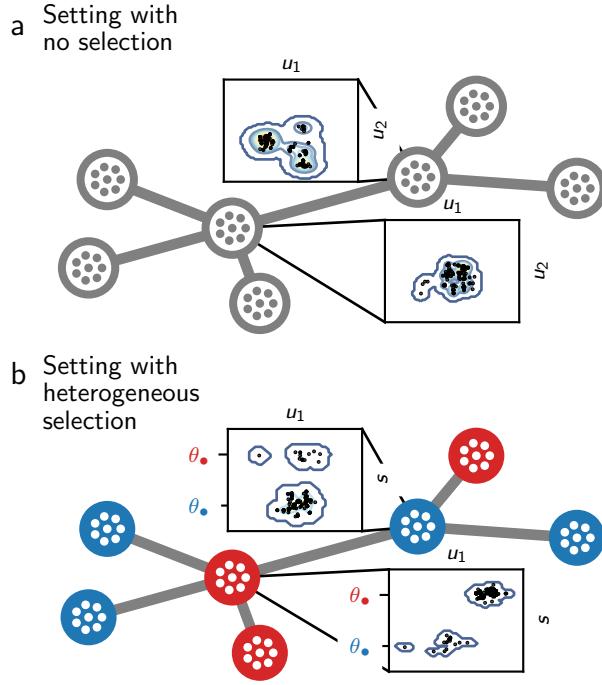


Fig. 2.1.: Graphical representation of the structure of individuals in the eco-evolutionary model. (a) Setting with no selection, where individuals are characterised by a set of neutral traits $u \in \mathcal{U}$. The scatter plots represent a projection of the first two components of u for the individuals present on the designated vertices at time $t = 1000$, obtained from one simulation of the IBM. (b) Setting with heterogeneous selection. In this setting, individuals are additionally characterised by adaptive traits $s \in \mathcal{S}$. Blue vertices favour the optimal adaptive trait value θ_\bullet , while red vertices favour θ_\bullet . The scatter plots represent a projection of the first component of u and s for the individuals present on the designated vertices at time $t = 1000$, obtained from one simulation. The majority of individuals are locally well-adapted and have an adaptive trait close to the optimal value, but some maladaptive individuals originating from neighbouring vertices are also present. $m = 0.05$.

selection, competition stabilises the population size fluctuations, and the dynamics can be considered deterministic and expressed as

$$\partial_t N_t^{(i)} = N_t^{(i)} \left[b(1 - m) - \frac{N_t^{(i)}}{K} \right] + mb \sum_{j \neq i} \frac{a_{i,j}}{d_j} N_t^{(j)} \quad (2.3)$$

where $A = (a_{i,j})_{1 \leq i,j \leq M}$ is the adjacency matrix of the graph G and $D = (d_1, d_2, \dots, d_M)$ is a vector containing the degree of each vertex (number of edges incident to the vertex). The first term on the right-hand side corresponds to logistic growth, which accounts for birth and death events of non-migrating individuals. The second term captures the gains due to migrations, which depend on the graph topology. Assuming that all vertices with the same degree have an equivalent position on the graph, corresponding to a “mean

field” approach (see Methods), one can obtain a closed-form solution from Eq. (2.3) (see Eq. (2.12)), which shows that the average population size \bar{N} scales with $\langle \sqrt{k} \rangle^2 / \langle k \rangle$, where $\langle k \rangle$ is the average vertex degree and $\langle \sqrt{k} \rangle$ is the average square-rooted vertex degree. The quantity $\langle \sqrt{k} \rangle^2 / \langle k \rangle$, denoted as h_d , relates to the homogeneity in vertex degree of the graph and can therefore be viewed as a measure negatively associated with heterogeneity in connectivity. Simulations of the IBM illustrate that h_d can explain differences in population size for complex graph topologies with varying migration regimes (Fig. 2.2a for graphs with $M = 7$ vertices and Fig. S1a for $M = 9$). This analytical result is connected to theoretical work on reaction diffusion processes [Colizza2007] and highlights that irregular graphs (graphs whose vertices do not have the same degree) result in unbalanced migration fluxes that affect the ecological balance between births and deaths. Highly connected vertices present an oversaturated carrying capacity ($N^{(i)} > bK$, see Methods), increasing local competition and lowering total population size compared with regular graphs (Fig. 2.2a). Because populations with small sizes experience more drift ([Burger2000] and Fig. S2), this result indicates that graph topology affects neutral differentiation not only through population isolation, but also by affecting population dynamics.

Nonetheless, the stochasticity of the processes at the individual level can propagate to the population level and substantially affect the macroscopic properties. In particular, neutral differentiation emerges from the stochastic fluctuations of the populations’ neutral trait distribution. These fluctuations complicate an analytical underpinning of the dynamics, and in this case simulations of the IBM offer a straightforward approach to evaluate the level of neutral differentiation.

2.2.3 Effect of graph topology on neutral differentiation under no selection

We study a setting with no selection and investigate the effect of the graph topology on neutral differentiation. When migration is limited, individuals’ traits are coherent on each vertex but stochastic drift at the population level generates neutral differentiation between the vertices. Migration attenuates neutral differentiation because it has a correlative effect on local trait distributions. Following [Economou2007, Carrara2012, Thompson2017], we expect that the intensity of the correlative effect depends on the average path length of the graph $\langle l \rangle$, defined as the average shortest path between all pairs of vertices [Bounova2012]. For a constant number of vertices, $\langle l \rangle$ is strictly related to the mean betweenness centrality and quantifies the graph connectivity [Bounova2012]. High $\langle l \rangle$ implies low connectivity and a greater isolation of populations, and hence we expect that graphs with high $\langle l \rangle$ are associated with high differentiation levels. We consider various graphs with an identical number of vertices and run simulations of the IBM to obtain the neutral differentiation level $Q_{ST,u}$ attained after a time long enough to discard transient dynamics (see Methods). We then interpret the discrepancies in $Q_{ST,u}$ across the simulations by relating them to the underlying graph topologies.

We observe strong differences in $Q_{ST,u}$ across graphs for varying m , and find that $\langle l \rangle$ explains at least 55% of the variation in $Q_{ST,u}$ across all graphs with $M = 7$ vertices for (Fig. 2.2b). Nonetheless, some specific graphs, such as the star graph, present higher levels of $Q_{ST,u}$ than expected by their average path length. To explain this discrepancy, we explore the effect of homogeneity in vertex degree h_d , as we showed in Eq. (2.12) that it decreases population size, which should in turn increase $Q_{ST,u}$ by intensifying stochastic drift. We find that h_d explains 57% of the variation for low m (Fig. 2.2b). However, the fit remains similar after correcting for differences in population size (see Table S1), indicating that irregular graphs structurally amplify the isolation of populations. Unbalanced migration fluxes lead central vertices to host more individuals than allowed by their carrying capacity. This causes increased competition that results in a higher death rate, so that migrants have a lower probability of further spreading their trait. Highly connected vertices therefore behave as bottlenecks, increasing the isolation of peripheral vertices and consequently amplifying $Q_{ST,u}$.

We then evaluate the concurrent effect of $\langle l \rangle$ and h_d on $Q_{ST,u}$ with a multivariate regression model that we fit independently for low and high migration regimes (Fig. 2.2d). The multivariate regression model explains at least 70% of the variation in $Q_{ST,u}$ for the migration regimes considered and for graphs with $M = 7$ vertices (see Table S2 for details). Moreover, we find that $\langle l \rangle$ and h_d have similar contributions to neutral differentiation for low m , but the effect of $\langle l \rangle$ increases for higher migration regimes while the effect of h_d decreases. To ensure that these conclusions can be generalised to larger graphs, we conduct the same analysis on a subset of graphs with $M = 9$ vertices and find congruent results (Fig. S1). In the absence of selection and with competitive interactions, graphs with a high average path length $\langle l \rangle$ and low homogeneity in vertex degree h_d , or similarly graphs with low connectivity and high heterogeneity in connectivity, show high levels of neutral differentiation.

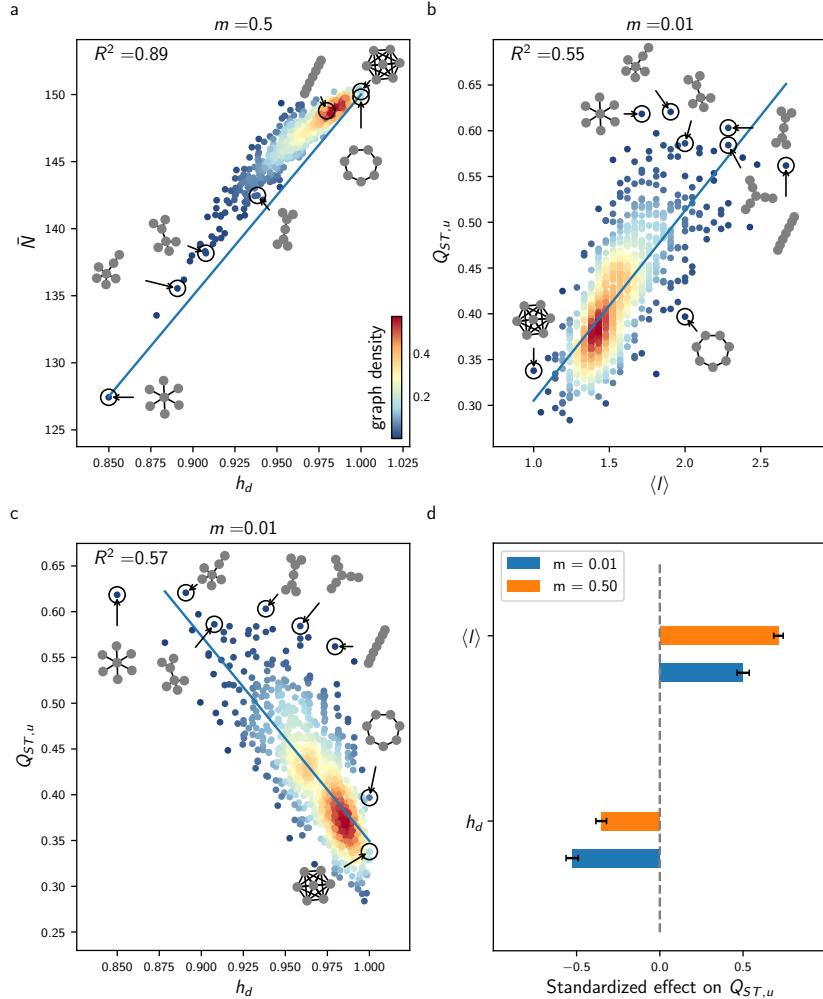


Fig. 2.2.: Effect of the topology metrics $\langle l \rangle$ and h_d on average population size \bar{N} and neutral differentiation $Q_{ST,u}$ under the setting with no selection. Each plot is based on results from simulations of the IBM on all undirected connected graphs with $M = 7$ vertices. (a) Response of \bar{N} to homogeneity in degree $h_d = (\sqrt{k})^2/\langle k \rangle$, for $m = 0.5$. (b) Response of $Q_{ST,u}$ to average path length $\langle l \rangle$ for $m = 0.01$. (c) Response of $Q_{ST,u}$ to homogeneity in degree h_d for $m = 0.01$. In (a), (b) and (c) the colour scale corresponds to the proportion of the graphs with similar x and y axis values (graph density), while the blue line corresponds to a linear fit. (d) Standardized effect of h_d and $\langle l \rangle$ on $Q_{ST,u}$, obtained from multivariate regression models independently fitted for $m = 0.01$ and $m = 0.5$. The contributions of $\langle l \rangle$ and h_d to $Q_{ST,u}$ are similar for low migration regimes. Error bars show 95% confidence intervals. Analogous results on graphs with $M = 9$ vertices are presented in Fig. S1.

2.2.4 Deterministic approximation of the population dynamics and adaptation under heterogeneous selection

We next consider heterogeneous selection and investigate the response of adaptive differentiation to the spatial distribution of habitat types, denoted as the Θ -spatial distribution. Adaptive differentiation emerges from local adaptation, but migration destabilises adaptation as a result of the influx of maladaptive migrants. We expect that higher connectivity between vertices of similar habitat type increases the level of adaptive differentiation, because it increases the proportion of well-adapted migrants. Local adaptation can be investigated by approximating the stochastic dynamics of the trait distribution with a deterministic partial differential equation (PDE). We demonstrate under mean field assumption how the deterministic approximation can be reduced to an equivalent two-habitat model. We analyse the reduced model with the theory of adaptive dynamics [Meszena1997, Mirrahimi2020] and find a critical migration threshold m^* that determines local adaptation. m^* depends on a quantity coined the habitat assortativity r_Θ , and we demonstrate with numerical simulations that r_Θ determines the overall adaptive differentiation level $Q_{ST,s}$ reached at steady state in the deterministic approximation.

Heterogeneous selection, captured by the dependence of the birth rate on Θ_i , generates a stabilising force that dampens the stochastic fluctuations of the adaptive trait distribution. The dynamics of the adaptive trait distribution consequently shows a deterministic behavior and we demonstrate in Supplementary Methods and Figs. S4 and S6 that the number of individuals on v_i with traits $s \in \Omega \subset \mathcal{S}$ can be approximated by the quantity $\int_\Omega n^{(i)}(s)ds$, where $n^{(i)}$ is a continuous function solution of the PDE

$$\partial_t n_t^{(i)}(s) = n_t^{(i)}(s) \left[b^{(i)}(s)(1-m) - \frac{1}{K} \int_S n_t^{(i)}(\mathbf{s})d\mathbf{s} \right] + m \sum_{j \neq i} b_j(s) \frac{a_{i,j}}{d_j} n_t^{(j)}(s) + \frac{1}{2} \mu \sigma_\mu^2 \Delta_s \left[b^{(i)}(s) n_t^{(i)}(s) \right] \quad (2.4)$$

Equation (2.4) is similar to Eq. (2.3), except that it incorporates an additional term corresponding to mutation processes and that the birth rate is trait dependent. We show how Eq. (2.4) can be reduced to an equivalent two-habitat model under mean field assumption. The mean field approach differs slightly from the setting with no selection because vertices are labelled with Θ_i . Here we assume that vertices with similar habitat types have an equivalent position on the graph (see Fig. S7 for a graphical representation), so that all vertices with habitat type $\Theta_i = \theta_\bullet$ are characterised by the identical adaptive trait distribution that we denote by \bar{n}^\bullet . Let $P(\bullet, \bullet)$ denote the proportion of edges connecting a vertex v_i with $\Theta_i = \theta_\bullet$ to a vertex v_j with $\Theta_j = \theta_\bullet$, and let $P(\bullet)$ denote the proportion of vertices v_i with $\Theta_i = \theta_\bullet$. By further assuming that habitats are homogeneously distributed on the graphs so that $P(\bullet) = P(\bullet) = \frac{1}{2}$, Eq. (2.4) transforms into

$$\begin{aligned} \partial_t \bar{n}_t^\bullet(s) &= \bar{n}_t^\bullet(s) \left[b_\bullet(s)(1-m) - \frac{1}{K} \int_S \bar{n}_t^\bullet(\mathbf{s})d\mathbf{s} \right] + \frac{1}{2} \mu \sigma_\mu^2 (\Delta_s b_\bullet \bar{n}_t^\bullet)(s) \\ &\quad + \frac{m}{2} [(1-r_\Theta)b_\bullet(s)\bar{n}_t^\bullet(s) + (1+r_\Theta)b_\bullet(s)\bar{n}_t^\bullet(t)] \end{aligned} \quad (2.5)$$

(see Methods), where we define

$$r_\Theta = 2(P(\bullet, \bullet) - P(\bullet, \bullet)) \quad (2.6)$$

as the habitat assortativity of the graph, which ranges from -1 to 1 . When $r_\Theta = -1$, all edges connect dissimilar habitat types (disassortative graph), while as r_Θ tends towards 1 the graph is composed of two clusters of vertices with identical habitat types (assortative graph). Eq. (2.5) can be analysed with the theory of adaptive dynamics [Meszena1997, Debarre2013, Mirrahimi2020], a mathematical framework that provides analytical insights by assuming a “trait substitution process”. Following this assumption, the mutation term in Eq. (2.5) is omitted and the phenotypic distribution results in a collection of discrete individual types that are gradually replaced by others until evolutionary stability is reached (see Methods and [Meszena1997, Debarre2013, Mirrahimi2020] for details). By applying the theory of adaptive dynamics, we find a critical migration rate m^*

$$m^* = \frac{1}{(1 - r_\Theta)} \frac{4p\theta^2}{(1 + 3p\theta^2)} \quad (2.7)$$

so that when $m < m^*$, a single type of individual exists with adaptive trait $s^* = \frac{\theta_\bullet + \theta_\bullet}{2} = 0$ in the steady state (see Methods for the derivation of Eq. (2.7)). In this case, adaptive differentiation $Q_{ST,s}$ is nil and the average population size is given by $\bar{N} = bK(1 - p\theta)^2$. In contrast, when $m = 0$ and/or $r_\Theta = 1$, all individuals are locally well-adapted with trait $\bar{\Theta}_i$ on v_i , and it follows that the average population size is higher and equal to $\bar{N} = bK$, while adaptive differentiation is strong and equal to $Q_{ST,s} = \text{Var}(\Theta) = \theta^2$. When $0 < m < m^*$, the coexistence of two types of individuals on each vertex v_i is predicted but the calculation of the trait values is more subtle. To understand the effect of m and r_Θ on the local trait distributions and on $Q_{ST,s}$, we therefore leave behind the adaptive dynamics framework and numerically solve Eq. (2.5) by including the mutation term. When $0 < m < m^*$, the local trait distributions are bimodal with peaks corresponding to the two types of individuals predicted by the adaptive dynamics. The highest peak corresponds to the well-adapted individuals, whose adaptation is destabilised by the influx of maladaptive migrants (Fig. 2.3a). This phenomenon is dampened as r_Θ increases, since the proportion of maladaptive migrants is reduced in assortative graphs (Fig. 2.3b). As a consequence, the habitat assortativity r_Θ increases the differentiation $Q_{ST,s}$ when $0 < m < m^*$ (Fig. 2.3c). The simulations further confirm that the adaptive dynamics prediction given by Eq. (2.7) is still valid when the continuous accumulation of mutations is considered, so that for $m > m^*$ the local trait distributions obtained from Eq. (2.5) are unimodal and $Q_{ST,s}$ vanishes (Fig. 2.3a,c). Our analysis of the mean field deterministic approximation Eq. (2.5) therefore demonstrates that assortative graphs present high levels of adaptive differentiation $Q_{ST,s}$. On the other hand, the analysis shows that $Q_{ST,s}$ rapidly declines with increasing m on disassortative graphs, until $Q_{ST,s}$ vanishes when $m > m^*$.

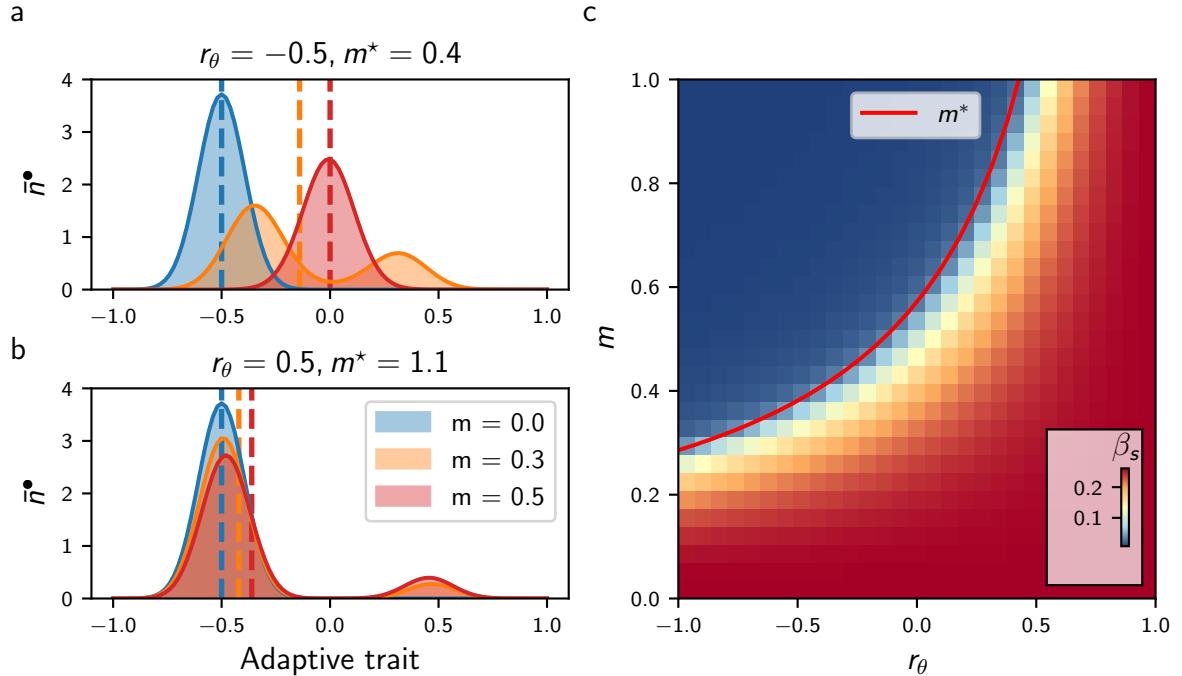


Fig. 2.3.: Effect of habitat assortativity r_Θ and migration m on the local adaptive trait distribution \bar{n}^* and on the adaptive differentiation level $Q_{ST,s}$ under the mean field, deterministic approximation Eq. (2.5). (a) Effect of m and r_Θ on \bar{n}^* . Migration induces the apparition of maladaptive individuals (centred around $\theta_0 = 0.5$), which destabilise local adaptation by displacing the mean value of the well-adapted individuals (centred around $\theta_0 = -0.5$). Together with the decrease in local adaptation, migration causes a displacement of the mean value of the local trait distribution (represented by the vertical dashed lines), which decreases local population size and adaptive differentiation $Q_{ST,s}$. (b) Similar data for higher r_Θ . Increasing r_Θ increases population size and $Q_{ST,s}$. (c) Effect of r_Θ on $Q_{ST,s}$. The red line indicates the critical migration threshold m^* predicted by Eq. (2.7); $Q_{ST,s}$ vanishes when $m > m^*$.

2.2.5 Effect of graph topology on adaptive differentiation under heterogeneous selection

To generalise the conclusions drawn from the mean field deterministic approximation Eq. (2.5), we generate different Θ -spatial distributions for varying graph topology, and compare outputs of the IBM simulations with those of Eq. (2.5) (see Methods for the details of the simulations). For each combination of Θ -spatial distribution and graph, we compute the habitat assortativity r_Θ , since r_Θ can be generalised from Eq. (2.6) to any graph topology following the original definition of [Newman2003a] as

$$r_\Theta = \frac{\text{Cov}(\Theta_x, \Theta_\wedge)}{\sigma_{\Theta_x} \sigma_{\Theta_\wedge}} \quad (2.8)$$

where Θ_{\times} and Θ_{\wedge} denote the sets of habitats found at the toe and tip of each directed vertex of graph V , and $\langle \Theta_{\times} \rangle, \langle \Theta_{\wedge} \rangle$ and $\sigma_{\Theta_{\times}}, \sigma_{\Theta_{\wedge}}$ denote their respective means and standard deviations (see Supplementary Methods). The mean field deterministic approximation Eq. (2.5) is in very good agreement with the IBM simulations for general graph ensembles at low migration regimes, and captures the response of \bar{N} and $Q_{ST,s}$ to r_{Θ} (Fig. 2.4). Nonetheless, under high migration regimes, higher levels of $Q_{ST,s}$ are observed in the stochastic simulations compared with the mean field deterministic approximation (Fig. S8). We hypothesize that this reinforcement is generated by stochastic drift, which must become the main driver of differentiation when local adaptation is lost for $m > m^*$, and perform a multivariate regression analysis to investigate the additional effect of $\langle l \rangle$ and h_d on $Q_{ST,s}$. As expected, the analysis highlights that the effect of $\langle l \rangle$ and h_d are substantial and complement the effect of r_{Θ} for high m (Fig. 2.5c for graphs with $M = 7$ vertices and Fig. S9a for $M = 9$), further explaining the discrepancies observed (see Table S3).

We extend our analyses to realistic landscapes with a continuum of habitat types by running simulations on graphs obtained from real spatial habitat datasets and by considering mean annual temperature as a proxy for habitat type (see Fig. S10 and Table S4). We also consider simulations accounting for trait-dependent competition to test whether our results hold under more complex ecological processes (see Supplementary Methods for the implementation details and Table S5 for the results). The simulations are congruent and show that the effects of r_{Θ} , h_d and $\langle l \rangle$ are similar under these alternative settings, underlining the robustness of these metrics and the generality of our conclusions. Taken together, these results indicate that under sufficiently strong selection and sufficiently high habitat heterogeneity, adaptive differentiation $Q_{ST,s}$ is mainly driven by habitat assortativity r_{Θ} . Nonetheless, local adaptation is lost in disassortative graphs when $m > m^*$, such that $\langle l \rangle$ and h_d become complementary determinants of $Q_{ST,s}$ for high migration regimes.

2.2.6 Effect of habitat assortativity on neutral differentiation under heterogeneous selection

We finally consider a setting with heterogeneous selection where individuals carry both neutral and adaptive traits. With distinct habitat types, selection promotes neutral differentiation by reducing the birth rate of maladaptive migrants, reinforcing the isolation of local populations. We have shown above that adaptive differentiation $Q_{ST,s}$ is driven by habitat assortativity r_{Θ} , so we expect r_{Θ} , together with the topological metrics found in the setting with no selection, to influence the level of neutral differentiation $Q_{ST,u}$. We first investigate how the response of $Q_{ST,u}$ to migration compares between the setting with no selection and the setting with heterogeneous selection for graphs with an identical topology. We then examine how the response compares between graphs with an identical topology but different r_{Θ} . We finally consider simulations on different graphs with varying r_{Θ} to assess the concurrent effect of assortativity and topology metrics on $Q_{ST,u}$.

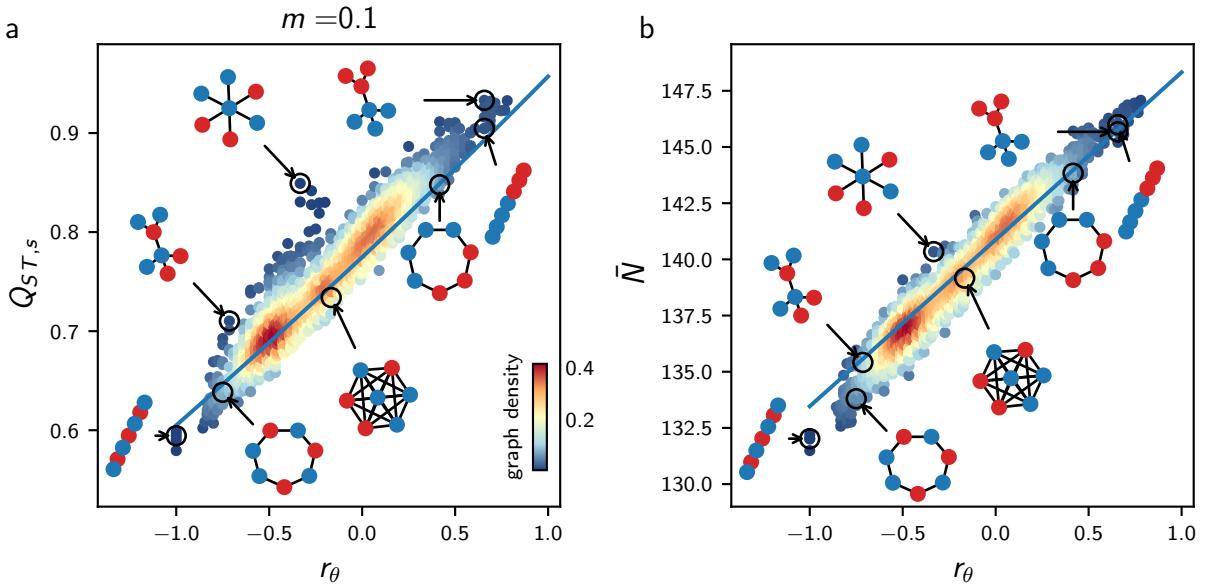


Fig. 2.4.: Effect of habitat heterogeneity r_Θ on $Q_{ST,s}$ and average population size \bar{N} for general graph ensembles. Each plot is based on results from simulations of the IBM on all undirected connected graphs with $M = 7$ vertices and varying r_Θ , for $m = 0.1$. Insights from Eq. (2.5) are congruent with the IBM simulations for complex habitat connectivity patterns at low m . (a) Effect of r_Θ on $Q_{ST,s}$. (b) Effect of r_Θ on average population size \bar{N} . Similar results with $m = 0.5$ are presented in Fig. S8. The colour scale corresponds to the proportion of the graph with similar x and y axis values (graph density). The blue lines correspond to results obtained from the mean field approximation Eq. (2.5).

Migration has a fitness cost because maladaptive migrants present lower fitness. Under an equivalent migration regime, migrants therefore have a lower probability of reproduction, increasing the populations' isolation compared with a setting without selection. Simulations with varying m on the complete graph confirm that selection in heterogeneous habitats reinforces $Q_{ST,u}$ compared with a setting without selection (Fig. 2.5a). Nonetheless, previous results show that adaptive differentiation $Q_{ST,s}$ vanishes on a disassortative graph when $m > m^*$, implying that individuals become equally fit in all habitats. In this case, the isolation effect of heterogeneous selection is lost and $Q_{ST,u}$ reaches a similar level as in the setting with no selection for $m > m^*$ (Fig. 2.5a), although $Q_{ST,u}$ is slightly higher in the setting with heterogeneous selection due to a lower population size ($\bar{N} = bK(1 - p\theta)$ vs. $\bar{N} = bK$, see section above and Methods). This suggests that r_Θ reinforces $Q_{ST,u}$, as assortative graphs sustain higher levels of adaptive differentiation (Figs. 2.3 and 2.4). Simulations on the path graph with varying Θ -spatial distribution support this conclusion for high migration regimes, but show the opposite relationship under low migration regimes, where the habitat assortativity r_Θ decreases $Q_{ST,u}$ (Fig. 2.5b). Assortative graphs are composed of large clusters of vertices with similar habitats, within which migrants can circulate without fitness losses. Local neutral trait distributions become more correlated within these clusters, resulting in a decline in $Q_{ST,u}$ for assortative graphs compared with disassortative graphs. Figure 2.5b therefore highlights the ambivalent effect of r_Θ on $Q_{ST,u}$.

r_Θ reinforces $Q_{ST,u}$ by favouring adaptive differentiation, but also decreases $Q_{ST,u}$ by decreasing population isolation within clusters of vertices with the same habitat type.

We compare the effect of r_Θ on $Q_{ST,u}$ to the effect of the topology metrics $\langle l \rangle$ and h_d found in the setting with no selection using a multivariate regression analysis on simulation results obtained for different graphs with varying Θ -spatial distribution (Fig. 2.5d for graphs with $M = 7$ vertices and Fig. S9b for $M = 9$). The multivariate model explains the discrepancies in $Q_{ST,u}$ across the simulations for low and high migration regimes (see Table S3 for details), and we find that r_Θ , $\langle l \rangle$ and h_d contribute similarly to neutral differentiation. Hence, the effects of r_Θ and the topology metrics $\langle l \rangle$ and h_d add up under heterogeneous selection. A change in sign of the standardized effect of r_Θ on $Q_{ST,s}$ for low and high migration regimes verifies that the ambivalent effect of r_Θ on $Q_{ST,u}$ found on the path graph holds for general graph ensembles. Simulations with trait-dependent competition and simulations on realistic graphs with a continuum of habitat types equally confirm the ambivalent effect of r_Θ and further support the complementary effect of $\langle l \rangle$ and h_d on $Q_{ST,u}$ (see Fig. S10). $\langle l \rangle$ and h_d therefore drive neutral differentiation with and without heterogeneous selection. r_Θ becomes an additional determinant of neutral differentiation under heterogeneous selection. In contrast to the non-ambivalent, positive effect of habitat assortativity on adaptive differentiation, r_Θ can amplify or depress neutral differentiation depending on the migration regime considered.

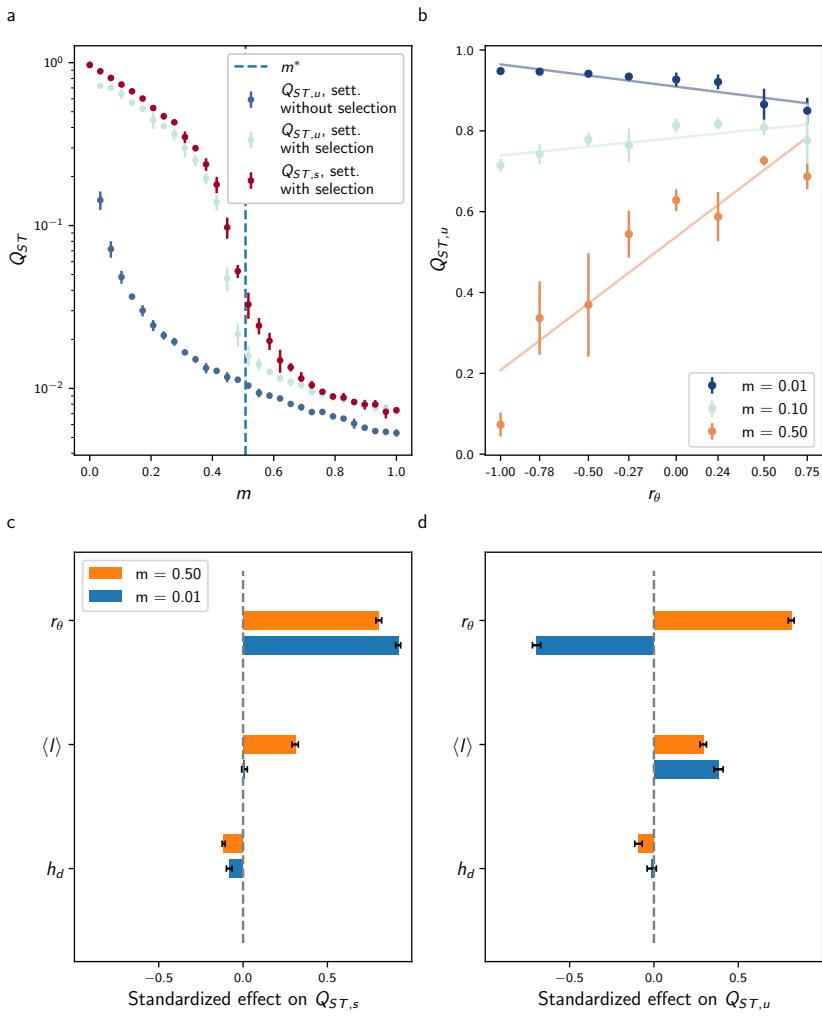


Fig. 2.5.: Effect of habitat assortativity r_Θ and the topology metrics $\langle l \rangle$ and h_d on adaptive differentiation $Q_{ST,s}$ and neutral differentiation $Q_{ST,u}$ under the setting with heterogeneous selection. (a) Comparison of the response of $Q_{ST,u}$ to migration with the response of $Q_{ST,u}$ in the setting with no selection for the complete graph. The dashed vertical blue line corresponds to the critical migration regime m^* predicted by Eq. (2.7). Heterogeneous selection increases $Q_{ST,u}$ when $m < m^*$, but local adaptation is lost when $m > m^*$, and in this case $Q_{ST,u}$ reaches similar levels as $Q_{ST,u}$ in the setting with no selection. (b) Response of $Q_{ST,u}$ to r_Θ and migration for the path graph. r_Θ correlates positively with $Q_{ST,u}$ for high m , but correlates negatively for low m . (c-d) Standardized effect of h_d , $\langle l \rangle$ and r_Θ on $Q_{ST,s}$ and $Q_{ST,u}$ obtained from a multivariate regression model independently fitted for low and high migration regimes on results from simulations of the IBM on all undirected connected graphs with $M = 7$ vertices and varying r_Θ (see Methods). The ambivalence of the effect of r_Θ on $Q_{ST,u}$ found for the path graph holds for general graph ensembles and adds up to that of $\langle l \rangle$ and h_d . Error bars show 95% confidence intervals. Analogous results on graphs with $M = 9$ vertices are presented in Fig. S9.

2.3 Discussion

Using analytical tools and simulations, we have built upon a graph representation of landscapes and a stochastic individual-based model to investigate how landscape features drive phenotypic differentiation. Our study is based on a first principles modelling approach [Champagnat2006] describing the stochastic dynamics of individuals and capturing the interplay between population dynamics, phenotypic evolution and spatial dynamics in heterogeneous habitats. In contrast to metacommunity models [Holland2008, Gilarranz2012, Mari2014, Gravel2016, Carrara2012, Thompson2017, Suzuki2021] and evolutionary metacommunity models [Economou2007, Economou2010], we have focused on differentiation at the population level. Quantitative genetics studies have investigated the effect of topology on differentiation under the assumption of non-overlapping generations, constant population sizes and regular spatial structures [Burger2000, Kimura1964, Lande1991, Nagylaki1994]. Generalising beyond these assumptions, our modelling framework accounts for population dynamics and includes frequency-dependent selection and competition. The systematic investigation of the effect of topology on differentiation over general graph ensembles and under different ecological settings shows that average path length $\langle l \rangle$, homogeneity in vertex degree h_d and habitat assortativity r_Θ contribute equally to differentiation. These results support correlative studies that have associated population differentiation (see [Manel2003] for a review) and species richness [Liu2018, Dias2014, Rahbek2001, Kreft2007a, Davies2007, Veech2007, Guegan1998, Stein2014] with a variety of metrics used as surrogates for connectivity, connectivity heterogeneity and habitat heterogeneity. To further our understanding of the origin of spatial biodiversity patterns, the contribution of landscape properties to discrepancies in population differentiation could be investigated at large scales by (i) using techniques to project real landscapes on graphs (see Fig. S10a–b); (ii) characterising the landscape features with $\langle l \rangle$, h_d and r_Θ ; and (iii) relating the obtained metrics maps to observation data. Overall, our results point to topology metrics that can connect spatial biodiversity patterns to the generating eco-evolutionary and spatial processes.

In the absence of selection, neutral differentiation is more pronounced on graphs with a high average path length $\langle l \rangle$, but is also negatively associated with homogeneity in degree h_d (Fig. 2.2c–d). $\langle l \rangle$ generalises the concept of dimensionality in [Kimura1964, Lande1991, Nagylaki1994], where it is shown that differentiation is lower for two-dimensional grid graphs compared with path graphs. At the species level, a similar effect of $\langle l \rangle$ on β -diversity (pairwise differences in species composition) has been reported with the graph metacommunity model of [Carrara2012] and with the graph eco-evolutionary metacommunity model of [Economou2007]. Additionally, we have shown that competition processes can substantially influence the effect of graph topology on neutral differentiation, such that h_d appears as an equally strong determinant of neutral differentiation in graphs (Fig. 2.2c,d). Our model realistically assumes that population growth is limited by the local carrying capacity. The latter becomes saturated on highly connected vertices in irregular graphs, an effect that has been experimentally documented in microcosm experiments [Altermatt2018]. As a consequence, central vertices behave as bottlenecks and amplify the isolation of peripheral

vertices [Orsini2013]. The role of h_d cannot be captured with classical metapopulation and quantitative genetics models or with models of evolutionary dynamics in graphs, as they assume constant population size. This behaviour should be prevalent in patchy landscapes where interspecific competition is high because of limiting resources. Our study highlights that heterogeneity in connectivity can reinforce differentiation patterns through the creation of unbalanced migration fluxes which affect ecological equilibrium.

Habitat assortativity r_Θ is a useful indicator for assessing how the spatial distribution of habitat types modulates local adaptation and adaptive differentiation in complex landscapes [Richardson2014]. While adaptation has been extensively studied along environmental gradients [Slatkin1973, Slatkin1978, Kirkpatrick1997, Polechova2015, Polechova2018, AndradeRestrepo2019, Doeblei2003], landscapes can be patchy and it is unrealistic to assume regularity [Dale2010]. Our work generalises the two-habitat models in [Meszena1997, Debarre2013, Mirrahimi2020] to spatial graphs, which can capture irregularity in connectivity between distinct habitats [Dale2010]. We have found a critical migration regime m^* that dictates the possibility of adaptation. Equation (2.7) indicates that m^* increases with increasing selection strength p and with increasing environmental heterogeneity θ , the latter playing a similar role as the slope of the environmental gradient in [Slatkin1973, Slatkin1978, Polechova2015, Polechova2018]. Local adaptation would consequently be sustained under higher migration regimes following an increase in these parameters. The critical migration regime m^* further involves the habitat assortativity r_Θ , which must be regarded as a measure of habitat spatial auto-correlation based on the dispersal range of a species [Richardson2014]. Our results indicate that for general habitat distributions, r_Θ is the main determinant of adaptive differentiation under sufficiently strong selection p and high habitat heterogeneity θ , irrespective of the graph topology (Fig. 2.5c, Fig. S9a and Fig. S10). As p decreases, however, the effect of stochastic drift on $Q_{ST,s}$ should increase, and in this case the topology metrics $\langle l \rangle$ and h_d should become the most important determinants of $Q_{ST,s}$. Our results predict that in landscapes with heterogeneous habitats and where selection is strong, populations structured over assortative habitats are larger, support higher adaptive differentiation, and can be locally well-adapted even in the case where migration rates are high.

Spatial eco-evolutionary feedbacks in heterogeneous habitats can critically affect differentiation [Richardson2014]. While most eco-evolutionary studies have investigated diversification by considering a unique adaptive trait [Doeblei2003, Kirkpatrick1997, Polechova2015, Polechova2018], distinguishing between neutral and adaptive processes is crucial [Holderegger2006] and our work underlines the distinct responses of neutral and adaptive differentiation to landscape features (Fig. 2.5c vs. Fig. 2.5d). Our study builds upon recent mathematical models that consider the co-evolution of neutral and adaptive traits [Billiard2015, Anceschi2019] and extends those works to a spatial context. Our work provides an analytical framework to the concept of isolation by environment (IBE) [Orsini2013], which has been suggested to be one of the most important mechanisms governing differentiation in nature [Wang2014]. Heterogeneous selection leads to more isolation by modifying the fitness of migrants [Polechova2018], which further reduces gene flow [Richardson2014] and therefore affects the level of neutral differentiation (Fig. 2.5a) [GARANT2007]. Our

work proposes a mechanism by which habitat assortativity, relative to the migration regime, controls the direction of the effect of habitat heterogeneity on differentiation (Fig. 2.5d). Patchy, heterogeneous habitats can promote neutral differentiation as a result of selection that reduces effective migration [Stein2014]. Nonetheless, adaptive differentiation decreases substantially when migration is high relative to the critical migration regime m^* . In this case, neutral differentiation should be higher in landscapes with more aggregated habitats [Richardson2014]. Our study suggests that habitat assortativity must be considered for a complete understanding of differentiation in complex environments [Stein2014].

2.4 Conclusion

Using an analytical description of micro-evolutionary processes explicitly accounting for spatial dynamics over graphs, we have established how differentiation can emerge at the population level from eco-evolutionary feedbacks in spatial graphs representing complex landscapes. Our study formalises how differentiation emerges from the interplay between spatial dynamics, the co-evolution of neutral and adaptive traits, and landscape properties. Connectivity and habitat assortativity emerge as core determinants of differentiation in spatial graphs. These results resonate with empirical findings and previous theoretical works. Our study further stresses that habitat assortativity can depress or foster neutral differentiation depending on the migration regime. Additionally, our work highlights that heterogeneity in connectivity is an equally strong determinant of differentiation because highly connected habitats behave as bottlenecks, increasing the isolation of peripheral habitats. The present approach offers a promising framework for studying complex adaptive systems, as it can elucidate how macroscopic properties emerge from microscopic processes acting upon agents structured over complex spatio-evolutionary structures.

2.5 Methods

2.5.1 Mean field approximation

In the setting with no selection, the mean field approach involves the assumption that all vertices having the same degree are equivalent. For this, let $P(k, k')$ denote the proportion of edges that map a vertex with degree k to a vertex with degree k' , and consider the average population size $\bar{N}_t^{(k)}$ in each vertex with degree k at time t . An individual has probability $P(k, k')/k'$ to migrate from a vertex with degree k' to a vertex with degree k . Viewing $a_{i,j}/d_j$ as the probability that an individual on v_i chosen for migration moves to v_j , Eq. (2.3) then transforms into

$$\partial_t \bar{N}_t^{(k)} = \bar{N}_t^{(k)} \left[b(1-m) - \frac{\bar{N}_t^{(k)}}{K} \right] + mbk \sum_{k' \in V} \frac{P(k, k')}{k'} \bar{N}_t^{(k')} \quad (2.9)$$

Assuming uncorrelated graphs for which $P(k,k')/k' = P(k')k'/\langle k \rangle$, where $\langle k \rangle$ denotes the average degree of the graph [Colizza2007], yields

$$\partial_t \bar{N}_t^{(k)} = \bar{N}_t^{(k)} \left[b(1-m) - \frac{\bar{N}_t^{(k)}}{K} \right] + mb \frac{k}{\langle k \rangle} \bar{N}_t \quad (2.10)$$

where

$$\bar{N}_t = \sum_k P(k) \bar{N}_t^{(k)}. \quad (2.11)$$

When solving for the stationary state and setting $m = 1$, one obtains $\bar{N}^{(k)} = \sqrt{bK \frac{k}{\langle k \rangle} \bar{N}}$ from Eq. (2.10). Combining this with Eq. (2.11) yields

$$\bar{N} = bK \langle \sqrt{k} \rangle^2 / \langle k \rangle \quad (2.12)$$

In the setting with heterogeneous selection, the mean field approach involves the assumption that all vertices with a similar habitat are equivalent. In this case, an individual from a vertex with habitat θ_{\bullet} has the probability $P(\bullet, \bullet)/P(\bullet)$ of migrating to a vertex with habitat θ_{\bullet} , and therefore Eq. (2.4) transforms into

$$\begin{aligned} \partial_t \bar{n}_t^{\bullet}(s) &= \bar{n}_t^{\bullet}(s) \left[b_{\bullet}(s)(1-m) - \frac{1}{K} \int_S \bar{n}_t^{\bullet}(\mathbf{s}) d\mathbf{s} \right] + \frac{1}{2} \mu \sigma_{\mu}^2 \Delta_s [b_{\bullet}(s) \bar{n}_t^{\bullet}(s)] \\ &\quad + m \sum_{\circ \in \{\bullet, \bullet\}} b_{\circ}(s) \frac{P(\bullet, \circ)}{P(\circ)} \bar{n}_t^{\circ}(s) \end{aligned} \quad (2.13)$$

Considering that $P(\bullet) = P(\bullet) = \frac{1}{2}$ (habitats are equally distributed), $P(\bullet, \bullet) + P(\bullet, \bullet) = P(\bullet)$ (sum of conditional expectations), and $r_{\Theta} = 2(P(\bullet, \bullet) - P(\bullet, \bullet))$ (Eq. (2.6)), one obtains

$$P(\bullet, \bullet) = \frac{1}{4}(1 - r_{\Theta}) \quad \text{and} \quad P(\bullet, \bullet) = \frac{1}{4}(1 + r_{\Theta}) \quad (2.14)$$

Combining Eq. (2.14) with Eq. (2.13) yields Eq. (2.5). We show in the Supplementary Methods how one can derive Eq. (2.6) from the general definition of assortativity given in Eq. (2.8) and initially introduced in [Newman2003a].

2.5.2 Adaptive dynamics on graphs

The adaptive dynamics theory considers a monomorphic population that evolves following a “trait substitution process” [Meszena1997]. Accordingly, the trait s of the monomorphic metapopulation evolves gradually along the direction given by its fitness gradient, until it reaches a singular strategy s^* for which the fitness gradient vanishes. By omitting the mutation term, Eq. (2.6) can be written in the matrix form

$$\partial_t \bar{\mathbf{n}}_t(s) = M(s, \bar{\mathbf{N}}_t) \bar{\mathbf{n}}_t(s) \quad (2.15)$$

where $\bar{\mathbf{n}}_t = (\bar{n}_t^\bullet, \bar{n}_t^\bullet)$ and $\bar{\mathbf{N}}_t = (\bar{N}_t^\bullet, \bar{N}_t^\bullet)$ are the vectors containing the population densities and the population size on each habitat type, and

$$M(s, \bar{\mathbf{N}}) = \begin{bmatrix} \mathfrak{r}^\bullet(s, \bar{N}^\bullet) & \frac{m}{2}(1 - r_\Theta)b^\bullet(s) \\ \frac{m}{2}(1 - r_\Theta)b^\bullet(s) & \mathfrak{r}^\bullet(s, \bar{N}^\bullet) \end{bmatrix} \quad (2.16)$$

is the so-called projection matrix [Meszena1997], with $\mathfrak{r}^\bullet(s, \bar{N}^\bullet) = b_\bullet(s)(1 + \frac{m}{2}(r_\Theta - 1)) - \bar{N}^\bullet/K$. The overall fitness of individuals with trait s is the leading eigenvalue of M , which we denote with $\lambda(s, \bar{\mathbf{N}})$. We obtain the singular strategy s^* by setting the fitness gradient $\frac{\partial \lambda}{\partial s}(s, \bar{\mathbf{N}}) = 0$, from which we further obtain the demographic equilibrium $\bar{\mathbf{N}}^{s^*}$. Because of symmetries, we must have $\bar{N}^{\bullet, s^*} = \bar{N}^{\bullet, s^*}$ and $s^* = \frac{\theta_\bullet + \theta_\bullet}{2} = 0$, such that $\bar{N}^{\bullet, s^*} = \bar{N}^{\bullet, s^*} = bK(1 - p\theta^2)$. s^* is said to be evolutionary stable if no mutants can invade, i.e. if s^* locally maximises the fitness of a mutant with trait y in the resident population with trait s^* , given by $\lambda(y, \bar{\mathbf{N}}^{s^*})$ (see [Meszena1997] for details). One can show that $\left[\frac{\partial \lambda}{\partial y}(y, \bar{\mathbf{N}}^{s^*}) \right]_{y=s^*} = 0$ and the condition for evolutionary stability becomes $\left[\frac{\partial^2 \lambda}{\partial y^2}(y, \bar{\mathbf{N}}^{s^*}) \right]_{y=s^*} < 0$. We compute and simplify this inequality through computer algebra (see Mathematica notebook at <https://github.com/vboussange/differentiation-in-spatial-graphs>), which leads to Eq. (2.7).

2.5.3 Numerical simulations

We performed Monte Carlo simulations by running five replicate simulations for each result presented, with $b = 1$, local carrying capacity $K = 150$, selection strength $p = 1$, mutation rate $\mu = 0.1$, mutation range $\sigma_\mu = 5 \cdot 10^{-2}$, and total time span $t = 1000$. This parameter choice made it possible to discard transient dynamics while obtaining results in a reasonable computational time (see Fig. S11). In settings (1) and (2), we ran simulations on all of the 853 undirected connected graphs with $M = 7$ vertices and on 1126 of the 261,080 undirected connected graphs with $M = 9$ vertices, listed at <http://oeis.org/A001349>. Graphs with $M = 9$ vertices were selected with a stratified sampling method: we randomly sampled without replacement a maximum of 50 graphs for each class of graphs with an equal number of vertices. For the setting with heterogeneous selection, we randomly generated different habitat type spatial distributions for each graph. For graphs with $M = 7$ and $M = 9$, we selected at most three and two habitat spatial distributions with a unique r_Θ value, respectively, corresponding to a total of 2537 labelled graphs for $M = 7$ and 2250 labelled graphs for $M = 9$. We then computed $Q_{ST,u}$ and $Q_{ST,s}$, which we further averaged over the last time steps and across the replicates. Since the dynamics of $Q_{ST,u}$ is characterised by large quadratic variations, we simulated individuals with $d = 300$ neutral traits, where each trait can independently be affected by mutations. $Q_{ST,u}$ values presented were then obtained from the average $Q_{ST,u}$ for each trait. This reduced the variance of the numerical simulations and is also biologically meaningful because populations are characterised by many traits, most of which are neutral [Holderegger2006]. As initial conditions, MK individuals were homogeneously distributed over all of the vertices, with traits centred on 0

and with standard deviation σ_μ . Graph metrics used for the meta-analysis were calculated using the **LightGraphs.jl** library [Bromberger2017]. We numerically solved the PDEs with a finite difference scheme using **DifferentialEquations.jl** [Rackauckas2017], ensuring that the domain was large enough to avoid border effects.

2.6 Acknowledgements

We thank Thomas Poulet, Sylvian Billiard, Sepideh Mirrahimi, Heike Lischke, Joshua Payne, Conor Waldock, Yaquan Chang, Flora Desmet, Benjamin Flück and Alexander Skeels for helpful discussions and comments on the manuscript. L.P. was supported by the Swiss National Science Foundation grant (N° 310030_188550).

2.7 Data accessibility

The code used in this article is available online at <https://github.com/vboussange/differentiation-in-spatial-graphs>.

2.A Supplementary Methods

2.A.1 Mathematical construction of the model

The model is a measure-valued point process [Bansaye2015], so that individuals are represented as dirac functions $\delta_{x_k^{(i)}}$, where $x_k^{(i)} \in \mathcal{X}$ corresponds to the traits' value of individual k located on vertex v_i . Under this formalism, the population on v_i is represented as a sum of dirac functions $\nu^{(i)} = \sum_k^{N^{(i)}} \delta_{x_k^{(i)}}$, where $N^{(i)}$ is the local population size. It follows that the time variation of the process can be described by the so-called infinitesimal generator L , defined for all real valued functions ϕ as

$$L\phi(\nu_t^{(i)}) = \partial_t \mathbb{E} [\phi(\nu_t^{(i)})] \quad (\text{S1})$$

(see [Linke2015] for an introduction to infinitesimal generators). Equation (S1) provides the expected time variation at time t of e.g. the population size by choosing $\phi(\nu_t^{(i)}) = \int_{\mathcal{X}} \nu_t^{(i)}(dx)$. Recall that we use b_i to denote the birth rate on vertex v_i , d for the death rate, μ for the mutation probability, m for the migration probability, $\mathcal{M}(x, y) = \frac{1}{\sqrt{2\pi}\sigma_\mu} \exp\left(-\frac{\|x-y\|^2}{2\sigma_\mu^2}\right)$ for the mutation kernel, K for the carrying capacity, $A = (a_{i,j})_{1 \leq i,j \leq M}$ for the adjacency matrix of the graph G , and $D = (d_1, d_2, \dots, d_M)$

for the vector containing the degree of each vertex. In order to explicitly write the generator L , let us recall that five events of different natures can alter the number of individuals with trait x on vertex v_i :

- an individual on v_i with trait x can give birth to an offspring with a similar trait that stays on v_i , at rate $(1 - \mu)(1 - m)b_i(x)$,
- an individual on v_i with trait y can give birth to an offspring with trait x that stays on v_i , at rate $\mu(1 - m)\mathcal{M}(x, y)b_i(y)$,
- an individual on v_i with trait x can die, at rate $d(N^{(i)}) = \frac{N^{(i)}}{K} = \frac{1}{K} \int_{\mathcal{X}} \nu_t^{(i)}(dx)$,
- an individual on V_j with trait x can give birth to an offspring that migrates to v_i , at rate $\frac{a_{i,j}}{d_j}(1 - \mu)m b_j(x)$,
- an individual on V_j with trait y can give birth to an offspring that migrates to v_i , at rate $\frac{a_{i,j}}{d_j}\mu m \mathcal{M}(x, y)b_j(x)$.

Summing over all all individuals and all vertices yields

$$\begin{aligned}
L\phi(\nu_t^{(i)}) &= \int_{\mathcal{X}} \left\{ b_i(\mathbf{x})(1 - \mu)(1 - m)(\phi(\nu_t^{(i)} + \delta_{\mathbf{x}}) - \phi(\nu_t^{(i)})) \right\} \nu_t^{(i)}(d\mathbf{x}) && \text{births w/o mutations, w/o } \\
&\quad + \int_{\mathcal{X}} \left\{ \mu(1 - m) \int_{\mathcal{X}} b_i(y)(\phi(\nu_t^{(i)} + \delta_z) - \phi(\nu_t^{(i)}))\mathcal{M}(\mathbf{x}, y)dy \right\} \nu_t^{(i)}(d\mathbf{x}) && \text{births w/ mutations, w/o } \\
&\quad + \iint_{\mathcal{X}} \left\{ \frac{1}{K}(\phi(\nu_t^{(i)} - \delta_{\mathbf{x}})) - \phi(\nu_t^{(i)}) \right\} \nu_t^{(i)}(dy) \nu_t^{(i)}(dx) \\
&\quad + \sum_{j \neq i} \frac{a_{i,j}}{d_j} \int_{\mathcal{X}} \mu m \left\{ \int_{\mathcal{X}} b_j(y)(\phi(\nu^{(j)} + \delta_{\mathbf{x}}) - \phi(\nu^{(j)}))\mathcal{M}(\mathbf{x}, y)dy \right\} \nu_t^{(j)}(d\mathbf{x}) && \text{migrations w/o } \\
&\quad + \sum_{j \neq i} \frac{a_{i,j}}{d_j} \int_{\mathcal{X}} \left\{ b_j(\mathbf{x})(1 - \mu)m(\phi(\nu^{(j)} + \delta_{\mathbf{x}}) - \phi(\nu^{(j)})) \right\} \nu_t^{(j)}(d\mathbf{x}). && \text{migrations w/ }
\end{aligned} \tag{S2}$$

Taking expectations in Eq. (S2), one can obtain an equation for the mean trajectory of the quantity of interest, $\mathbb{E} [\phi(\nu_t^{(i)})]$. Nonetheless, Eq. (S2) involves an integral with respect to $\nu_t^{(i)}(dx)\nu_t^{(i)}(dy)$, making it impossible to obtain an explicit solution. It is therefore unclear whether one can gain insight into the stochastic dynamics from Eq. (S2) without simplifying assumptions. We refer to [Champagnat2006] for a detailed discussion on the topic.

2.A.2 Deterministic approximation

One strategy to overcome the difficulties encountered above is to assimilate the process to its mean trajectory, assuming that $\mathbb{E} [\nu_t^{(i)}] \approx \nu_t^{(i)}$ and further approximating

$\nu_t^{(i)}$ with a continuous deterministic function $n_t^{(i)}$. Such strategy inherently neglects the stochasticity of the process, which is reasonable provided that a force dampens the stochastic fluctuations of the quantity of interest.

Setting with no selection

Consider a setting with no selection and recall that in this setting where $x \equiv u \in \mathcal{X} = \mathcal{U}$ we define

$$b_i(x) \equiv b \quad (\text{S3})$$

By applying the strategy mentioned above and choosing $\phi(n_t^{(i)}) = \int_{\mathcal{X}} n_t^{(i)}(x) dx$, Eq. (S2) transforms into the deterministic approximation of the population size dynamics given in the main-text by

$$\partial_t N_t^{(i)} = N_t^{(i)} \left[b(1-m) - \frac{N_t^{(i)}}{K} \right] + mb \sum_{j \neq i} \frac{a_{i,j}}{d_j} N_t^{(j)}. \quad (\text{S4})$$

Competition stabilises the population size dynamics, which behaves deterministically. This is supported by Fig. S5A, which shows how Eq. (S4) accurately describes the population size for varying migration regimes. Nonetheless, stochastic fluctuations drive the dynamics of the neutral trait distribution. Attempting to characterise the neutral trait distribution with the same strategy, this time setting $\phi(n_t^{(i)}) = n_t^{(i)}(u)$, yields

$$\begin{aligned} \partial_t n_t^{(i)}(u) &= n_t^{(i)}(u) \left[b(1-m)(1-\mu) - \frac{1}{K} \int_{\mathcal{U}} n_t^{(i)}(\mathbf{u}) d\mathbf{u} \right] \\ &\quad + (1-m)\mu b \int_{\mathcal{U}} n_t^{(i)}(\mathbf{u}) \mathcal{M}(u, \mathbf{u}) d\mathbf{u} \\ &\quad + m\mu b \sum_{j \neq i} \frac{a_{i,j}}{d_j} \int_{\mathcal{U}} n_t^{(j)}(u) \mathcal{M}(u, \mathbf{u}) d\mathbf{u} \\ &\quad + m(1-\mu)b \sum_{j \neq i} \frac{a_{i,j}}{d_j} b n_t^{(j)}(u). \end{aligned} \quad (\text{S5})$$

Solving for Eq. (S5), one can show that the variance of $n_t^{(i)}$ continuously grows in time (see Fig. S5) and tends to infinity as time goes to infinity, which is an unrealistic behaviour considering finite populations. Intuitively, this reflects the fact that no stabilising force acts on the neutral trait distribution, such that random fluctuations play a major role in driving the dynamics of the stochastic process. Figure S5 shows how IBM trajectories significantly differ from Eq. (S5), and Fig. S3 illustrates how diversity metrics obtained from Eq. (S5) do not match those obtained from simulations of the IBM.

Setting with heterogeneous selection

In contrast to the neutral trait dynamics, the adaptive distribution can successfully be approximated by a deterministic description because selection pressure acts as a stabilising force and stabilises the populations' adaptive trait, dampening the stochastic fluctuations. Consider the setting with heterogeneous selection and recall that in this setting where $x \equiv (s, u) \in \mathcal{X} = \mathcal{S} \times \mathcal{U}$ we define

$$b_i(x) \equiv b(1 - p(s - \theta_i)^2). \quad (\text{S6})$$

By applying the same strategy as above to characterise the adaptive trait distribution $n_t^{(i)}(s)$ by choosing $\phi(n_t^{(i)}) = n_t^{(i)}(s) \equiv \int_{\mathcal{U}} n_t^{(i)}(u, s) du$, Eq. (S2) transforms into

$$\begin{aligned} \partial_t n_t^{(i)}(s) &= n_t^{(i)}(s) \left[b_i(s)(1-m)(1-\mu) - \frac{1}{K} \int_{\mathcal{S}} n_t^{(i)}(\mathbf{s}) d\mathbf{s} \right] \\ &\quad + (1-m)\mu \int_{\mathcal{S}} b_i(\mathbf{s}) n_t^{(i)}(\mathbf{s}) \mathcal{M}(\mathbf{s}, s) d\mathbf{s} \\ &\quad + m\mu \sum_{j \neq i} \frac{a_{i,j}}{d_j} \int_{\mathbb{R}} b_j(\mathbf{s}) n_t^{(j)}(s) \mathcal{M}(\mathbf{s}, s) ds \\ &\quad + m(1-\mu) \sum_{j \neq i} \frac{a_{i,j}}{d_j} b_j(s) n_t^{(j)}(s). \end{aligned} \quad (\text{S7})$$

Assuming that the variance of the mutation kernel is small, one can use a diffusion approximation for the mutation term [Kimura1965, Debarre2013, Mirrahimi2020]

$$\int_{\mathcal{S}} b_i(\mathbf{s}) n_t^{(i)}(\mathbf{s}) \mathcal{M}(\mathbf{s}, s) d\mathbf{s} = b_i(s, t) n_t^{(i)}(s) + \frac{1}{2} \sigma_\mu^2 \Delta_s (b_i n_t^{(i)})(s). \quad (\text{S8})$$

Neglecting the terms in $m\mu$, we obtain

$$\begin{aligned} \partial_t n_t^{(i)}(s) &= n_t^{(i)}(s) \left[b_i(s, t)(1-m-\mu) - \frac{1}{K} \int_{\mathcal{S}} n_t^{(i)}(\mathbf{s}) d\mathbf{s} \right] \\ &\quad + \mu \left[b_i(s, t) n_t^{(i)}(s) + \frac{1}{2} \sigma_\mu^2 \Delta_s (b_i n_t^{(i)})(s) \right] \\ &\quad + m \sum_{j \neq i} b_j(s, t) n_t^{(j)}(s) a_{i,j} \end{aligned} \quad (\text{S9})$$

which, after rearranging terms, yields the elegant deterministic approximation of the adaptive trait dynamics

$$\partial_t n_t^{(i)}(s) = n_t^{(i)}(s) \left[b_i(s)(1-m) - \frac{1}{K} \int_{\mathcal{S}} n_t^{(i)}(\mathbf{s}) d\mathbf{s} \right] + m \sum_{j \neq i} b_j(s) \frac{a_{i,j}}{d_j} n_t^{(j)}(s) + \frac{1}{2} \mu \sigma_\mu^2 \Delta_s \left[b_i(s) n_t^{(i)}(s) \right]. \quad (\text{S10})$$

Setting $m = 0$ [Mirrahimi2020] shows that Eq. (S10) admits a stationary solution that is Gaussian, with variance $\sqrt{\mu}\sigma_\mu^2/\sqrt{p}$. Therefore, the variance of the adaptive trait distribution stabilises to a finite value. Intuitively, this reflects the fact that the random fluctuations of the adaptive trait distribution are damped by the stabilising force of selection. Provided that the selection strength p is large enough, Eq. (S10) is a good approximation of the adaptive trait distribution obtained from the stochastic process. Figure S6 shows how IBM trajectories are similar to the ones obtained from Eq. (S5), and Fig. S4 illustrates how diversity metrics obtained from Eq. (S5) match those obtained from simulations of the IBM.

2.A.3 Trait-dependent competition

To test whether the effects of the metrics hold under more complex ecological processes, we designed an extra experiment considering heterogeneous selection and adaptive trait-dependent competition, where the death rate of individuals on v_i with traits $x_k^{(i)} = (u_k^{(i)}, s_k^{(i)}) \in \mathcal{U} \times \mathcal{S}$ is given by

$$d(x_k^{(i)}, \nu^{(i)}) = \frac{1}{K} \int_S \exp\left(-\frac{(s_k^{(i)} - \mathbf{s})^2}{2\sigma_\alpha^2}\right) \nu^{(i)}(\mathbf{s}) \quad (\text{S11})$$

where σ_α is the competition bandwidth. This competition kernel tends to increase the population size, as it decreases the overall competition. The adaptive dynamics theory predicts that when $m = 0$, competition promotes two distinct types of individuals at either side of the adaptive trait optimum for a competition bandwidth $\sigma_\alpha < 1/\sqrt{2p}$, while a single type is observed when $\sigma_\alpha > 1/\sqrt{2p}$ [DoebeliMichael2011Ad]. We performed simulations in both cases for graphs with $M = 7$ vertices and show results of the multivariate regression analyses in Table S5. The analyses demonstrate that the trends reported in the main manuscript remain unchanged in both cases.

2.A.4 Derivation of the habitat assortativity metric r_Θ in binary environments

We demonstrate here how the habitat assortativity r_Θ relates to the conditional probability of habitats being connected, and we show how r_Θ simplifies under mean field assumption.

Following the original definition of [Newman2003a], habitat assortativity r_Θ is defined as the Pearson correlation of environmental conditions θ at either ends of the vertices V of graph G , that is

$$r_\Theta = \frac{\text{Cov}(\Theta_x, \Theta_\wedge)}{\sqrt{\text{Var}(\Theta_x)\text{Var}(\Theta_\wedge)}} = \frac{\langle\Theta_x\Theta_\wedge\rangle - \langle\Theta_x\rangle\langle\Theta_\wedge\rangle}{\sqrt{(\langle\Theta_x^2\rangle - \langle\Theta_x\rangle^2)(\langle\Theta_\wedge^2\rangle - \langle\Theta_\wedge\rangle^2)}} \quad (\text{S12})$$

where Θ_x and Θ_\wedge denote the sets of environmental conditions found at the toe and tip of each directed vertex of graph V , and $\langle\Theta_x\rangle$ and $\langle\Theta_\wedge\rangle$ denote their respective mean values.

Let $P(\bullet, \bullet)$ be the proportion of edges that connect a vertex with habitat θ_\bullet to a vertex with habitat θ_\bullet . One can also view $P(\bullet, \bullet)$ as the conditional probability that a vertex of type θ_\bullet is connected to a vertex of type θ_\bullet . Let $P(\bullet)$ denote the proportion of vertices that are of type θ_\bullet . First observe that for undirected graphs, one has $\langle\Theta_x\rangle = \langle\Theta_\wedge\rangle$ and $\langle\Theta_x^2\rangle = \langle\Theta_\wedge^2\rangle$. Assuming that habitats are symmetric and binary, it follows that $\theta_\bullet = -\theta_\bullet$. Then

$$\begin{aligned} \langle\Theta_x\Theta_\wedge\rangle &= P(\bullet, \bullet)\theta_\bullet^2 + P(\bullet, \bullet)\theta_\bullet^2 + [P(\bullet, \bullet) + P(\bullet, \bullet)]\theta_\bullet\theta_\bullet \\ &= \theta_\bullet^2(P(\bullet, \bullet) + P(\bullet, \bullet) - [P(\bullet, \bullet) + P(\bullet, \bullet)]), \end{aligned} \quad (\text{S13})$$

$$\begin{aligned} \langle\Theta_x\rangle &= P(\bullet)\theta_\bullet + P(\bullet)\theta_\bullet \\ &= \theta_\bullet[P(\bullet) - P(\bullet)], \end{aligned} \quad (\text{S14})$$

$$\begin{aligned} \langle\Theta_x^2\rangle &= P(\bullet)\theta_\bullet^2 + P(\bullet)\theta_\bullet^2 \\ &= \theta_\bullet^2[P(\bullet) + P(\bullet)] \\ &= \theta_\bullet^2. \end{aligned} \quad (\text{S15})$$

Combining Eq. (S13), Eq. (S14) and Eq. (S15) with Eq. (S12) one gets

$$\begin{aligned} r_\Theta &= \frac{\langle\Theta_x\Theta_\wedge\rangle - \langle\Theta_x\rangle\langle\Theta_\wedge\rangle}{\langle\Theta_x^2\rangle - \langle\Theta_x\rangle^2} \\ &= \frac{P(\bullet, \bullet) + P(\bullet, \bullet) - [P(\bullet, \bullet) + P(\bullet, \bullet)] - (P(\bullet) - P(\bullet))^2}{P(\bullet) + P(\bullet) - (P(\bullet) - P(\bullet))^2} \\ &= \frac{P(\bullet, \bullet) + P(\bullet, \bullet) - [P(\bullet, \bullet) + P(\bullet, \bullet)] - (P(\bullet) - P(\bullet))^2}{1 - (P(\bullet) - P(\bullet))^2}. \end{aligned} \quad (\text{S16})$$

Assuming that habitats are homogeneously distributed, we have $P(\bullet) = P(\bullet) = \frac{1}{2}$ and thus we obtain

$$r_\Theta = P(\bullet, \bullet) + P(\bullet, \bullet) - [P(\bullet, \bullet) + P(\bullet, \bullet)]. \quad (\text{S17})$$

The mean field approximation involves the assumption that all vertices with similar habitats are equivalent in terms of their connections with other habitats, so that $P(\bullet, \bullet) = P(\bullet, \bullet)$ and $P(\bullet, \bullet) = P(\bullet, \bullet)$, which yields $r_\Theta = 2(P(\bullet, \bullet) - P(\bullet, \bullet))$.

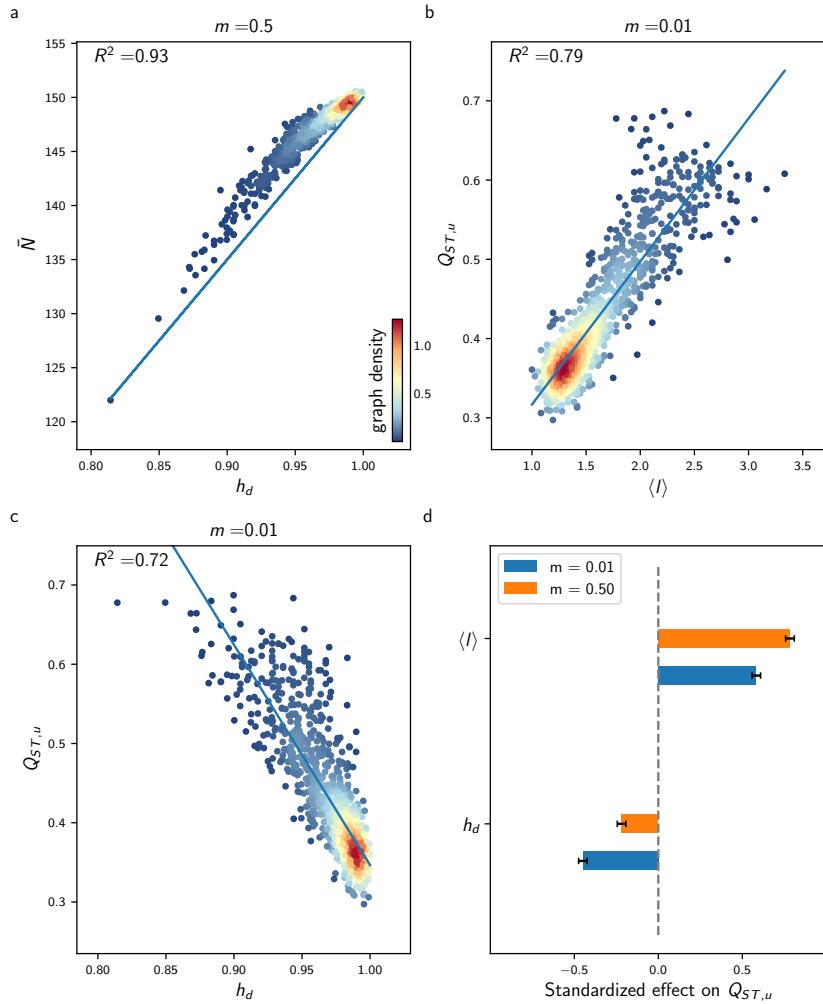


Fig. S1.: Effect of the topology metrics $\langle l \rangle$ and h_d on average population size \bar{N} and neutral differentiation $Q_{ST,u}$ under the setting with no selection. Each plot is based on results from simulations of the IBM on all undirected connected graphs with $M = 9$ vertices.

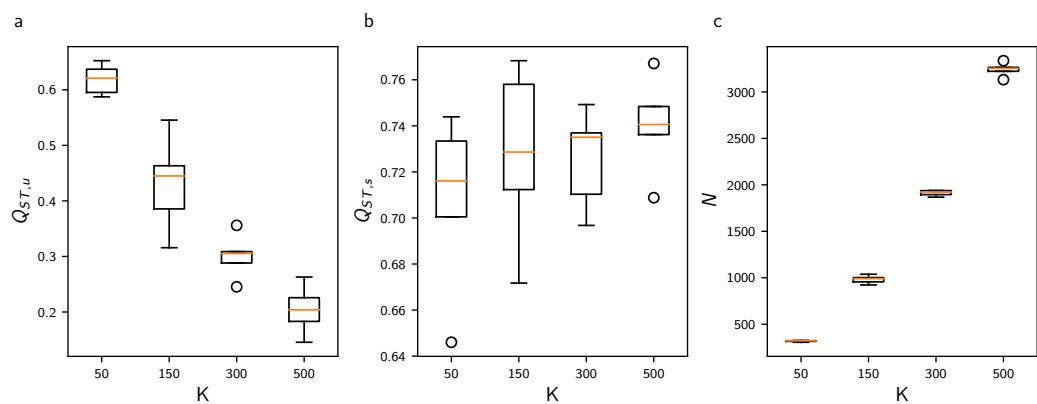


Fig. S2.: Effect of the carrying capacity K on $Q_{ST,u}$, $Q_{ST,s}$ and metapopulation size N for the line graph with $M = 7$ vertices for $m = 0.1$. Decreasing K increases $Q_{ST,u}$ as it favours drift, but it does not influence $Q_{ST,s}$.

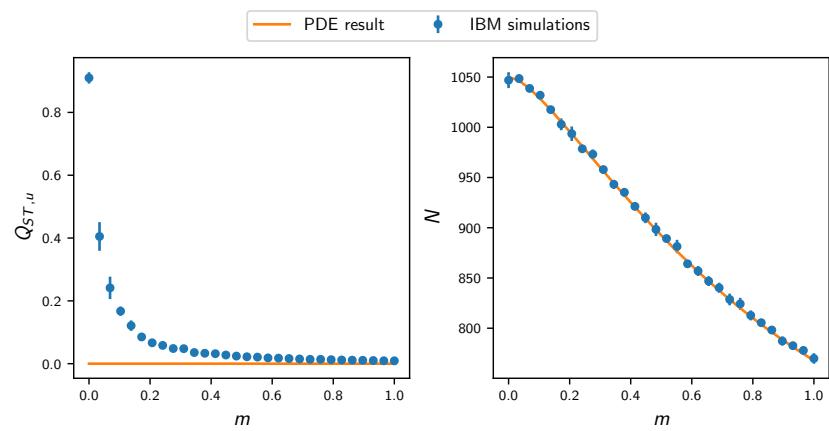


Fig. S3.: Comparison of results obtained from the deterministic approximations Eqs. (S4) and (S5) and from IBM simulations in the setting with no selection, on the star graph with $M = 7$ vertices. While Eq. (S4) can capture population size, Eq. (S5) is not able to capture $Q_{ST,u}$.

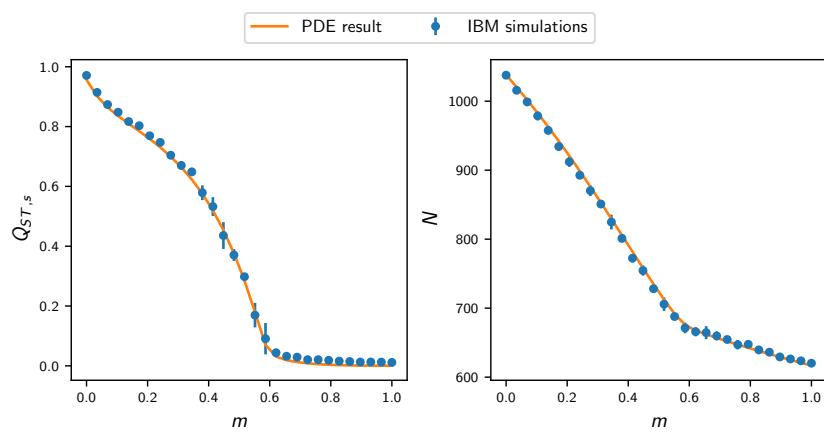


Fig. S4.: Comparison of results obtained from the deterministic approximation Eq. (S10) and IBM simulations in the setting with heterogeneous selection, on the star graph with $M = 7$ vertices. $Q_{ST,s}$ and population size obtained from Eq. (S10) closely match the IBM simulations.

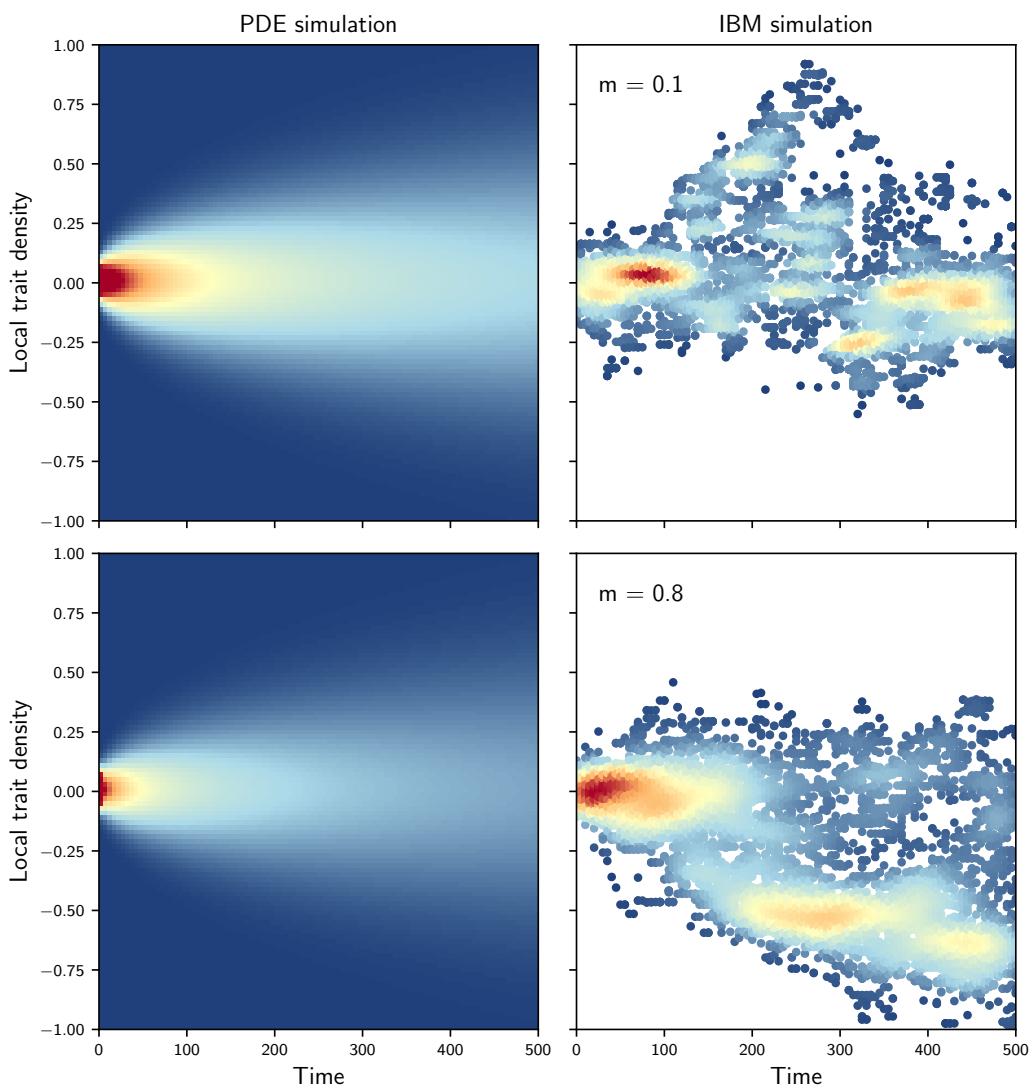


Fig. S5.: Comparison of the neutral trait density on one vertex obtained from Eq. (S5) (left) and from the IBM simulations (right) in the setting with no selection, for the chain graph. The densities obtained from Eq. (S5) and from the IBM are dissimilar.

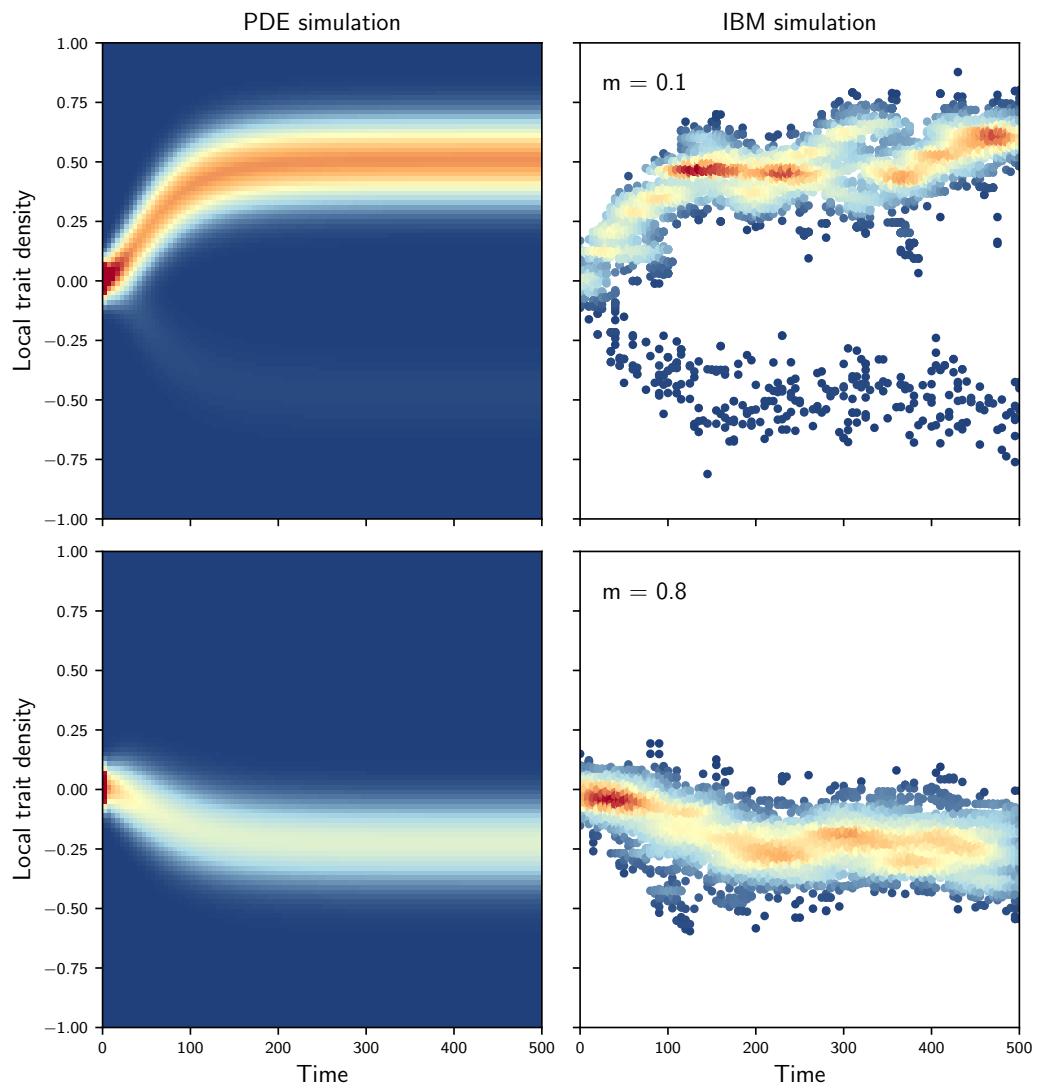


Fig. S6.: Comparison of the adaptive trait density on one vertex obtained from Eq. (S10) (left) and from the IBM simulations (right) in the setting with heterogeneous selection, for the star graph with $M = 7$ vertices. The densities obtained from Eq. (S10) and from the IBM are qualitatively similar.

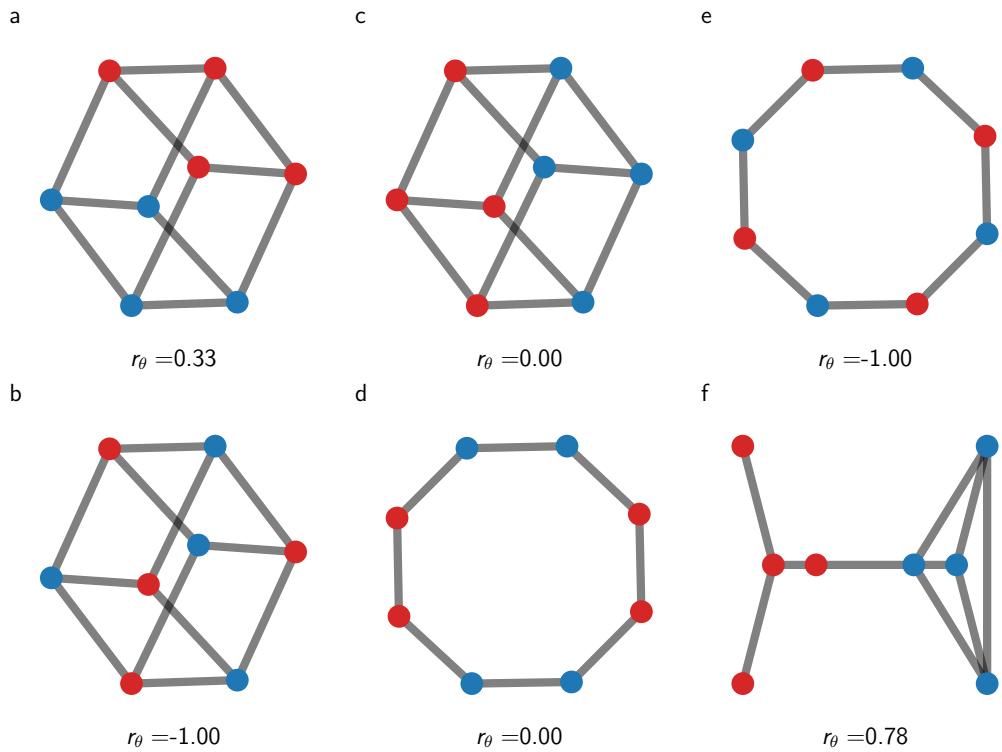


Fig. S7.: Graphs with spatial distribution of habitat types corresponding to different habitat assortativity r_Θ . Graphs (a–d) can be described exactly with a mean field approach, as blue and red vertices have an equivalent position on the graph.

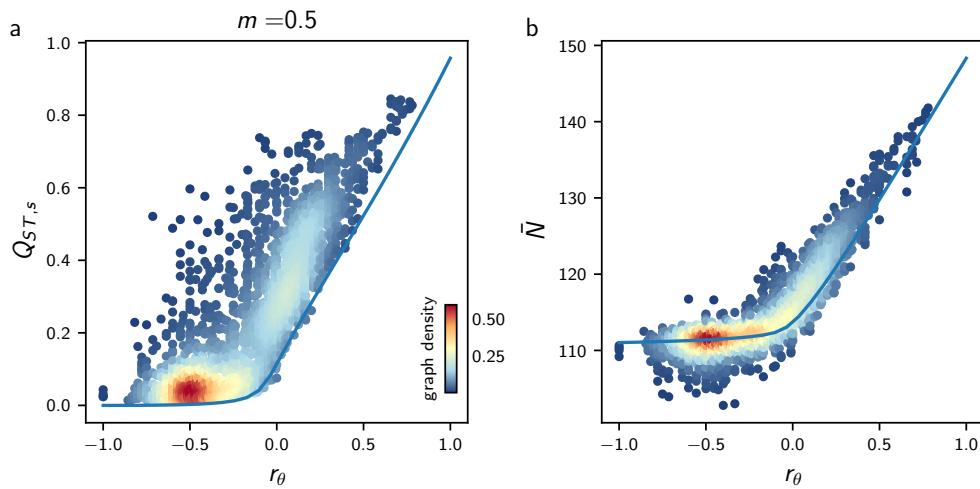


Fig. S8.: Effects of habitat heterogeneity r_Θ on $Q_{ST,s}$ and average population size \bar{N} based on results from simulations of the IBM on all undirected connected graphs with $M = 7$ vertices and varying r_Θ , for $m = 0.5$. The colour scale corresponds to the proportion of the graph with similar x and y axis values (graph density). The blue lines correspond to results obtained from the mean field, deterministic approximation Eq. (2.5). Deviations from the mean field, deterministic approximation Eq. (2.5) can be explained by differences in $\langle l \rangle$ and h_d between the graphs.

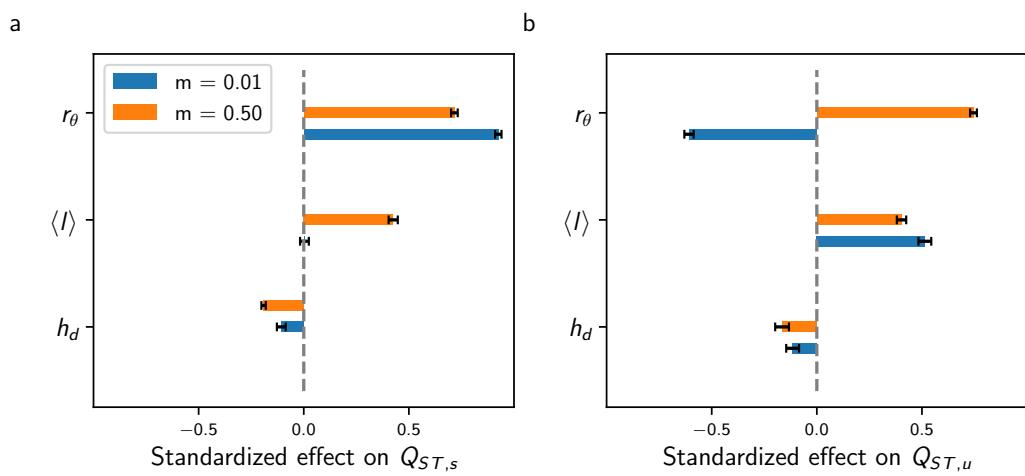


Fig. S9.: Standardized effects of h_d , $\langle l \rangle$ and r_θ on $Q_{ST,s}$ and $Q_{ST,u}$ obtained from multivariate regression models independently fitted for low and high migration regimes on results from simulations of the IBM on 1126 of the 261,080 undirected connected graphs with $M = 9$ vertices and varying r_Θ (see Methods for details). Error bars show 95% confidence intervals.

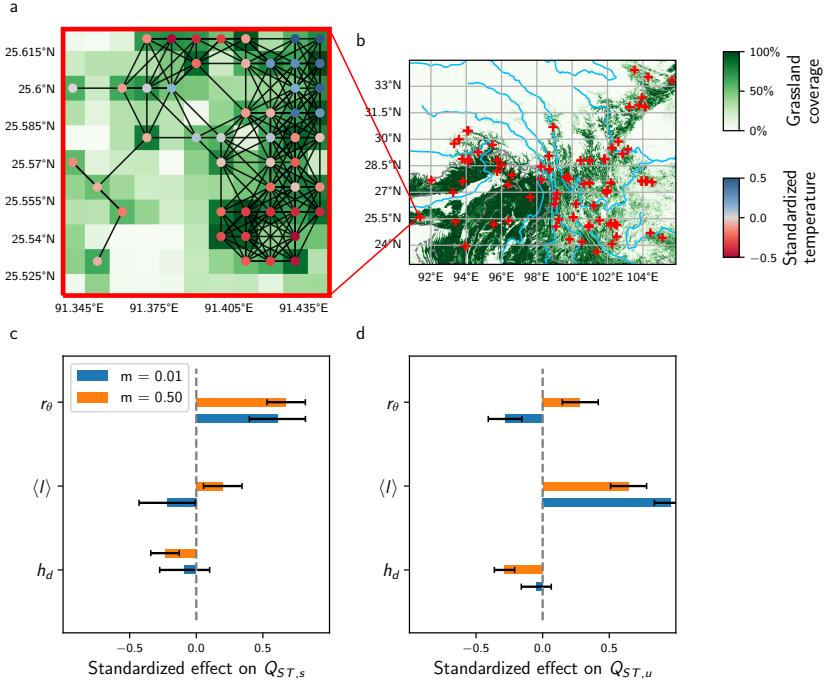


Fig. S10.: Simulations on graphs with $M = 49$ vertices obtained from real spatial habitat datasets, in the setting with heterogeneous selection. The region from where graphs are obtained is centred on the Hengduan Mountains in Southwest China, one of the most species-rich temperate mountain biota globally [Ding2020a]. (a) Graphical representation of a geographical area of size $0.11^\circ \times 0.11^\circ$. To create the graph, we considered biological populations living in grasslands, and used the dataset provided in [Jung2020] containing global grassland coverage at 0.01° resolution. We assigned a vertex to a geographical area of size $0.01^\circ \times 0.01^\circ$ if its grassland coverage was above a threshold arbitrarily set to 50%. We further assumed that two vertices were connected if their Euclidean distance was below a certain dispersal range, which we let vary from 1 to 2.5 km. Local annual average temperature was considered as the value that captures the habitat type at each vertex. Temperature data was obtained from the CHELSA dataset [Karger2017]. (b) Grassland coverage for the considered region. Blue lines correspond to rivers and dashed grey lines correspond to country borders. Red crosses indicate the locations of the 83 graphs sampled for the simulations used in (c–d). (c–d) Standardized effects of h_d , $\langle l \rangle$ and r_Θ on $Q_{ST,s}$ and $Q_{ST,u}$ obtained from multivariate regression models independently fitted for low and high migration regimes to results from simulations of the IBM on the 83 graphs which location is illustrated in (c) (see Table S4 for simulation details). Error bars show 95% confidence intervals.

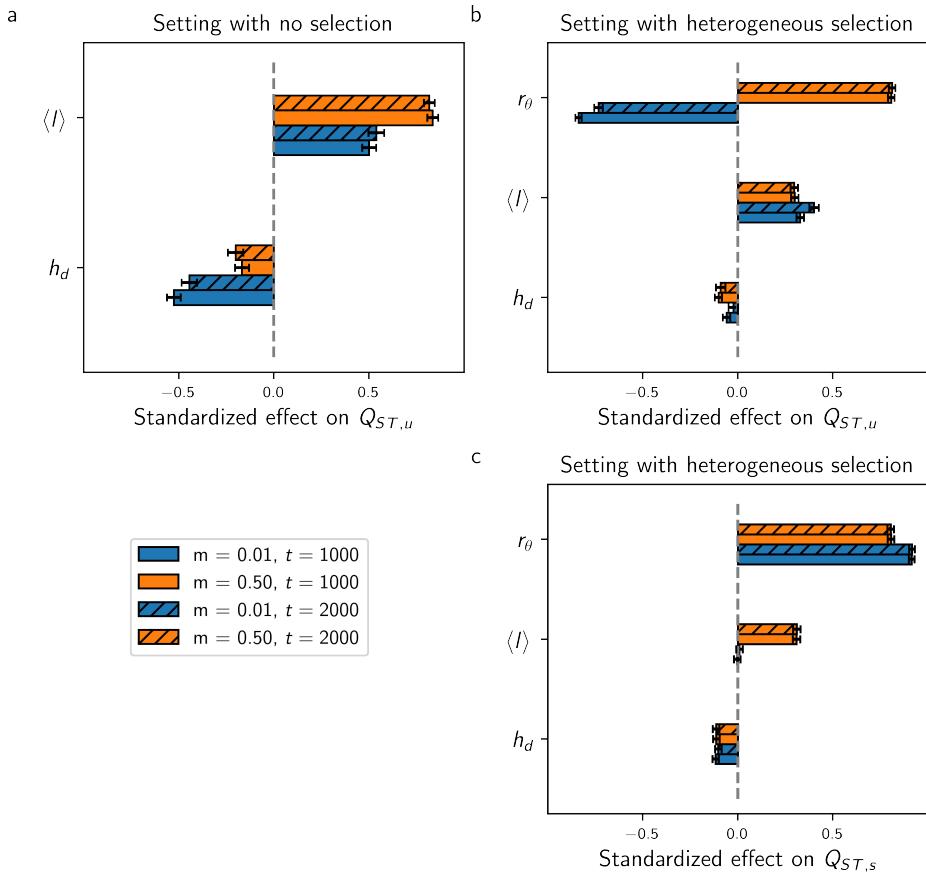


Fig. S11.: Standardized effects of h_d , $\langle I \rangle$ and r_Θ on $Q_{ST,u}$ in the setting with no selection and in the setting with heterogeneous selection for the time horizons $t = 1000$ and $t = 2000$, obtained from multivariate regression models independently fitted for low and high migration regimes to results from simulations of the IBM on all undirected connected graphs with $M = 7$ vertices and varying r_Θ (see Methods for details). (a–c) illustrate that the effects of the topology metrics on $Q_{ST,u}$ and $Q_{ST,s}$ remain constant for $t > 1000$ in both the settings without selection and with heterogeneous selection. Error bars show 95% confidence intervals.

Tab. S1.: Linear regression model coefficients for the effect of topology metrics on $Q_{ST,u}$ in the setting with no selection, based on all graphs with $M = 7$ vertices. ***
 $P < 0.001$

	$Q_{ST,u}$				$Q_{ST,u} - bN$	
	0.01	0.50	0.01	0.50	0.01	0.50
m	0.01	0.50	0.01	0.50	0.01	0.50
(Intercept)	0.000 (0.023)	-0.000 (0.017)	-0.000 (0.023)	-0.000 (0.025)	-0.000 (0.023)	-0.000 (0.028)
$\langle l \rangle$	0.739*** (0.023)	0.872*** (0.017)				
h_d			-0.753*** (0.023)	-0.674*** (0.025)	-0.753*** (0.023)	-0.143*** (0.028)
Number of sim.	853	853	853	853	853	853
R^2	0.546	0.760	0.567	0.454	0.567	0.030

Tab. S2.: Multivariate linear regression model coefficients for the effect of topology metrics on $Q_{ST,u}$ in the setting with no selection. *** $P < 0.001$

	$M = 7$		$M = 9$		
	$Q_{ST,u}$				
		0.01	0.50	0.01	0.50
m					
(Intercept)	-0.000 (0.017)	-0.000 (0.013)	0.000 (0.009)	-0.000 (0.010)	
h_d	-0.527*** (0.019)	-0.352*** (0.014)	-0.449*** (0.013)	-0.218*** (0.013)	
$\langle l \rangle$	0.500*** (0.019)	0.712*** (0.014)	0.583*** (0.013)	0.784*** (0.013)	
Number of sim.	853	853	1,126	1,126	
R^2	0.766	0.858	0.899	0.896	

Tab. S3.: Multivariate linear regression model coefficients for the effect of the topology metrics on $Q_{ST,u}$ and $Q_{ST,s}$ in the setting with heterogeneous selection. *** $P < 0.001$

m	$M = 7$				$M = 9$			
	$Q_{ST,s}$		$Q_{ST,u}$		$Q_{ST,s}$		$Q_{ST,u}$	
	0.01	0.50	0.01	0.50	0.01	0.50	0.01	0.50
(Intercept)	-0.000 (0.008)	-0.000 (0.009)	0.000 (0.011)	-0.000 (0.009)	0.000 (0.008)	0.000 (0.008)	0.000 (0.011)	0.000 (0.008)
h_d	-0.103*** (0.009)	-0.114*** (0.010)	-0.024 (0.013)	-0.091*** (0.010)	-0.107*** (0.011)	-0.191*** (0.011)	-0.115*** (0.016)	-0.165*** (0.011)
$\langle l \rangle$	0.009 (0.009)	0.310*** (0.010)	0.400*** (0.013)	0.297*** (0.010)	0.003 (0.011)	0.425*** (0.011)	0.514*** (0.016)	0.403*** (0.011)
r_Θ	0.915*** (0.008)	0.804*** (0.009)	-0.733*** (0.011)	0.812*** (0.009)	0.925*** (0.008)	0.717*** (0.008)	-0.608*** (0.011)	0.745*** (0.008)
Number of sim.	2,548	2,548	2,548	2,548	2,250	2,250	2,250	2,250
R^2	0.847	0.807	0.680	0.800	0.867	0.855	0.715	0.848

Tab. S4.: Multivariate linear regression model coefficients for the effect of topology metrics on $Q_{ST,u}$ and $Q_{ST,s}$ on real graphs with $M = 49$ vertices in the setting with heterogeneous selection. * $P < 0.05$, ** $P < 0.01$, *** $P < 0.001$

m	$Q_{ST,s}$		$Q_{ST,u}$	
	0.01	0.50	0.01	0.50
(Intercept)	-0.000 (0.093)	-0.000 (0.063)	-0.000 (0.056)	-0.000 (0.059)
h_d	-0.087 (0.094)	-0.235*** (0.064)	-0.048 (0.057)	-0.286*** (0.060)
$\langle l \rangle$	-0.218* (0.106)	0.199** (0.073)	0.965*** (0.064)	0.645*** (0.068)
r_Θ	0.609*** (0.106)	0.675*** (0.072)	-0.282*** (0.063)	0.282*** (0.068)
Number of sim.	83	83	83	83
R^2	0.311	0.678	0.752	0.717

Tab. S5.: Multivariate linear regression model coefficients for the effect of topology metrics on $Q_{ST,u}$ and $Q_{ST,s}$ in the setting of trait-dependent competition and heterogeneous selection (Section 2.A.3), based on all graphs with $M = 7$ vertices. ** $P < 0.01$, *** $P < 0.001$

m	$\sigma_a = 0.5 < 1/\sqrt{2p}$				$\sigma_a = 1 > 1/\sqrt{2p}$			
	$Q_{ST,s}$		$Q_{ST,u}$		$Q_{ST,s}$		$Q_{ST,u}$	
	0.01	0.50	0.01	0.50	0.01	0.50	0.01	0.50
(Intercept)	-0.000 (0.007)	-0.000 (0.010)	-0.000 (0.012)	-0.000 (0.011)	0.000 (0.007)	-0.000 (0.008)	-0.000 (0.011)	-0.000 (0.007)
h_d	-0.189*** (0.008)	-0.116*** (0.011)	-0.136*** (0.013)	-0.132*** (0.012)	-0.099*** (0.008)	-0.131*** (0.009)	-0.040** (0.012)	-0.124*** (0.008)
$\langle l \rangle$	0.029*** (0.008)	0.378*** (0.011)	0.382*** (0.013)	0.553*** (0.012)	-0.001 (0.008)	0.296*** (0.009)	0.370*** (0.012)	0.278*** (0.008)
r_Θ	0.913*** (0.007)	0.733*** (0.010)	-0.672*** (0.012)	0.535*** (0.011)	0.936*** (0.007)	0.814*** (0.008)	-0.755*** (0.011)	0.844*** (0.007)
Number of sim.	2,548	2,548	2,548	2,548	2,548	2,548	2,548	2,548
R^2	0.877	0.760	0.638	0.704	0.885	0.826	0.695	0.858

Part II

Scientific machine learning for
eco-evolutionary modelling

Introduction

This introduction is greatly inspired from Arnulf Jentzen lecture notes "On Deep Artificial Neural Networks and Machine Learning for PDEs" (DEEP_NEURAL_NETWORK_028.PDF) and "Numerical Analysis of Stochastic Ordinary Differential Equations" (NASODE_LECTURE_NOTES.PDF)

Towards a Deep Learning-based approximation for PDEs

Curse of dimensionality

Standard approximation methods for nonlinear PDEs, such as finite difference approximation methods, finite element approximation methods, spectral Galerkin approximation methods, sparse grid approximation methods, and standard nested Monte Carlo approximation methods, suffer from the so-called curse of dimensionality in the sense that the number of computational operations of the employed approximation scheme grows exponentially in the PDE dimension $d \in N$ or in the reciprocal $1/\epsilon$ of the prescribed approximation accuracy $\epsilon \in (0, \infty)$ and it is a very challenging problem to design and analyze approximation methods for nonlinear PDEs which overcome the curse of dimensionality in the sense that the number of computational operations of the proposed approximation algorithm grows at most polynomially in the PDE dimension $d \in N$ and in the reciprocal $1/\epsilon$ of the prescribed approximation accuracy $\epsilon \in (0, \infty)$. [Beck2020]

A concrete example for the finite difference scheme

For an ODE which solution is $u: [0, T] \rightarrow \mathbb{R}$, one needs to discretize the interval $[0, T]$ in N_t time steps to find variables u_1, \dots, u_{N_t} approximating $u(t_1), \dots, u(t_{N_t})$. Such variables can be found by solving a system of linear equations, in the case of an explicit scheme. For a PDE which solution is $u: [0, T] \times D \rightarrow \mathbb{R}$, $D \subseteq \mathbb{R}^d$ one needs to discretize the time space and the domain D . Assuming that dimension i of D is discretized in N_i , then one needs to write a linear system of equations with $N_t N_1 \dots N_d$

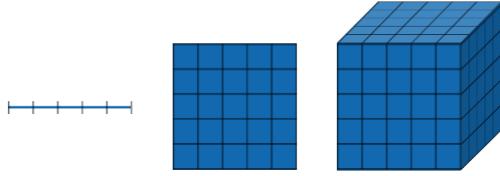


Fig. 2.12.: Illustration of the curse of dimensionality with a finite difference scheme

variables corresponding to $u(t_1, x_1^{(1)}, \dots, x_d^{(1)}), \dots, u(t_{N_t}, x_1^{(N_1)}, \dots, x_d^{(N_d)})$. Assuming that $N_t, N_1, \dots, N_d > N$, this means the number of variables to solve for is greater than N^{d+1} . This means that the computational cost grows exponentially in the number of the dimension of the PDE.

Diffusion approximation

Proposition 2.1.1 (Diffusion approximation). *Let $d \in \mathbb{N}$, $t \in (0, \infty)$, $\sigma \in \mathbb{R}$, $a \in \mathbb{R}$, $b \in (0, \infty)$, let $\phi: \mathbb{R}^d \rightarrow \mathbb{R}$ be a function, let $u \in C^{1,2}([0, T] \times \mathbb{R}^d, \mathbb{R})$ be a function with at most polynomially growing partial derivatives which satisfies for every $t \in [0, T]$, $x \in \mathbb{R}^d$ that $u(0, x) = \phi(x)$ and*

$$\partial_t u(t, x) = \frac{1}{2} \sigma^2 \Delta_x u(t, x), \quad (2.18)$$

let $(\omega, \mathcal{F}, \mathbb{P})$ be a probability space, and let $\mathbb{W}: \Omega \rightarrow \mathbb{R}^d$ be a standard normal random variable. Then

- (i) it holds that the function $\phi: \mathbb{R}^d \rightarrow \mathbb{R}$ is twice continuously differentiable with at most polynomially growing derivatives and
- (ii) it holds for every $x \in \mathbb{R}^d$ that $u(T, x) = \mathbb{E} [\phi(\sigma \mathbb{W} \sqrt{T} + x)]$

Note that here above we think of $\sigma \mathbb{W} \sqrt{T}$ as σB_T where B_T is a standard Brownian motion.

Monte Carlo integration

Here we show that the above expectation can be approximated by a Monte Carlo averaging

$$\mathbb{E} [\phi(\sigma \mathbb{W} \sqrt{T} + x)] \approx \frac{1}{M} \sum_{m=1}^M [\phi(\sigma \mathbb{W}^{(m)} \sqrt{T} + x)] \quad (2.19)$$

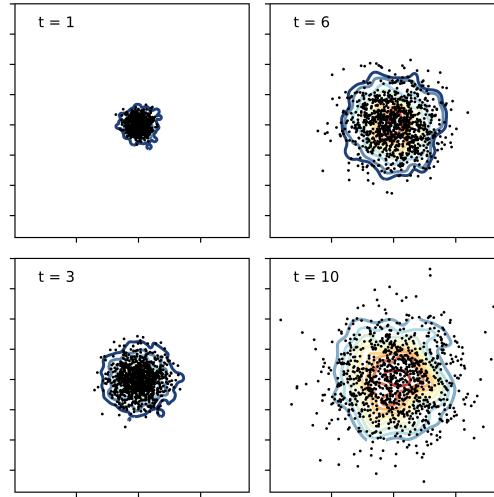


Fig. 2.13.: Brownian motion diffusion. Contour line represent estimated particle density.

and that the approximation error is proportional to \sqrt{N} so that a Monte Carlo approach combined with the diffusion approximation overcomes the curse of dimensionality.

Deep Learning-based approximation

A genuine idea to approximate u on the whole interval $[a, b] \subseteq \mathbb{R}^d$ is to introduce a function approximator $NN_\theta(x) = \sigma_l(W_l(\dots(W_2\sigma_1(W_1x+b_1)+b_2)\dots)+b_l$ where W_i , b_i are respectively the weights and biases at level i , and $\theta = \{W_1, \dots, W_l, b_1, \dots, b_l\}$ denotes the ensemble of parameters. Note that in the following we adopt an other perspective on neural networks, dealing with layers and neurons instead of matrix multiplications.

Cost function

$$\mathcal{L}_\theta = \frac{1}{M} \sum_{m=1}^M \left[NN_\theta^{(m)}(\xi^{(m)}) - \phi(\sigma B_T^{(m)} + \xi^{(m)}) \right]^2 \quad (2.20)$$

Prerequisites

This leads to the

Theorem 2.1.2 (Non-linear Feynman Kac for initial value problems). Consider the PDE

$$\partial_t u(t, x) = \mu(t, x) \nabla_x u(t, x) + \frac{1}{2} \sigma^2(t, x) \Delta_x u(t, x) + f(x, u(t, x)) \quad (2.21)$$

with initial conditions $u(0, x) = g(x)$, where $u: \mathbb{R}^d \rightarrow \mathbb{R}$. Then

$$u(t, x) = \int_0^t \mathbb{E} [f(X_{t-s}^x, u(T-s, X_{t-s}^x)) ds] + \mathbb{E} [u(0, X_t^x)] \quad (3)$$

with

$$X_t^x = \int_0^t \mu(X_s^x) ds + \int_0^t \sigma(X_s^x) dB_s + x. \quad (2.22)$$

State contribution

Parameter inference for Differential Equations

Reverse engineering

Approximate Bayesian Computation in Evolution and Ecology : [Beaumont2010]

Parametric estimation applied to predict community stability: [Cenci2019]

adjoint for solver

State contribution

Deep learning approximations for non-local nonlinear PDEs with Neumann boundary conditions

by Victor Boussange^{1,2}, Sebastian Becker³, Arnulf Jentzen^{4,5}, and Loïc Pellissier^{6,7}

¹ Unit of Land Change Science, Swiss Federal Research Institute for Forest, Snow and Landscape (WSL), Switzerland

² Landscape Ecology, Institute of Terrestrial Ecosystems, Department of Environmental Systems Science, ETH Zürich, Switzerland

³ Risklab, Department of Mathematics, ETH Zürich, Switzerland

⁴ School of Data Science and Shenzhen Research Institute of Big Data, The Chinese University of Hong Kong, Shenzhen, China

⁵ Applied Mathematics: Institute for Analysis and Numerics, Faculty of Mathematics and Computer Science, University of Münster, Germany

⁶ Unit of Land Change Science, Swiss Federal Research Institute for Forest, Snow and Landscape (WSL), Switzerland

⁷ Landscape Ecology, Institute of Terrestrial Ecosystems, Department of Environmental System Science, ETH Zürich, Switzerland

Preprint: doi:XXX

Non-local nonlinear partial differential equations (PDEs) with Neumann boundary conditions are used to model dynamical processes in many scientific fields, ranging from physics to biology. Here we propose two numerical methods based on machine learning and Picard iterations to approximately solve non-local nonlinear PDEs. The proposed machine learning-based method is an extended variant of the deep splitting scheme and utilizes neural networks to provide approximate solutions on a subset of the domain of the solution $[a, b]^d \subseteq D$. The Picard iterations-based method is an extended variant of the multilevel Picard approximation scheme and provides an approximate solution for a single point of the domain $x \in D$. Both methods are mesh-free and allow to solve non-local nonlinear PDEs with Neumann

boundary conditions in high dimensions. In the two methods, the numerical difficulties arising due to the dimensionality of the PDEs are avoided by (i) using the correspondence between the expected trajectory of reflected stochastic processes and the solution of PDEs (given by the Feynman-Kac formula), and by (ii) using a plain vanilla Monte Carlo integration to handle the non-local term. We evaluate the performance of the two methods with 5 different PDEs arising in physics and biology. In all cases, the methods yield good results in up to 10 dimensions with short run times. Our work extends recently developed methods to overcome the curse of dimensionality in solving PDEs.

scientific machine learning | curse of dimensionality | non-local PDEs

Author contributions: to be completed.

The authors declare no competing interest.

3.1 Introduction

In this paper, we derive numerical schemes to approximately solve high-dimensional non-local nonlinear partial differential equations (PDEs) with Neumann boundary conditions. Such PDEs are used to describe a variety of processes in physics, engineering, finance and biology, but can generally not be solved analytically, requiring numerical methods to provide approximate solutions. Nonetheless, traditional numerical methods suffer from the curse of dimensionality, calling for the development of novel approximation methods.

The need for solving non-local nonlinear PDEs has been expressed in various fields as they provide a more general description of the dynamical systems than their local counterparts [Kavallaris2018, Sunderasan2020]. In physics and engineering, non-local nonlinear PDEs are found, e.g., in models of Ohmic heating production [Lacey1995], in the investigation of the fully turbulent behavior of real flows [Caglioti1992], in phase field models allowing non-local interactions [Barone1971, Gajewski2003, Coleman1994, Hairer2016], or in phase transition models with conservation of mass [RUBINSTEIN1992, Stoleriu2011]; see [Kavallaris2018] for further references. In finance, non-local PDEs are used, e.g., in jump-diffusion models for the pricing of derivatives where the dynamics of stock prices is described by stochastic processes experiencing large jumps [Merton1976, Chan1999, Kou2000, Abergel2010, Benth2001, Sunderasan2020, Cruz2020, Tankov2003]. Penalty methods for pricing American put options such as in Kou's jump-diffusion model [Huang2013, Gan2020], considering large investor where the agent policy affects the assets prices [Amadori2003, Abergel2010], or considering default risks [Pham2009, Henry-Labordere2012],

can further introduce nonlinear terms in non-local PDEs. In economics, non-local nonlinear PDEs appear, e.g., in evolutionary game theory with the so-called replicator-mutator equation capturing continuous strategy spaces [Oechssler2001, Kavallaris2017, Hamel2020, Alfaro2016, Alfaro2019], or in growth model where consumption is non-local [Banerjee2021]. In biology, non-local nonlinear PDEs are used, e.g., to model processes determining the interaction and evolution of organisms. Examples include models of morphogenesis and cancer evolution [Lorz2013, Chen2020, Villa2021], models of gene regulatory networks [Pajaro2017], population genetics models with the non-local Fisher–Kolmogorov–Petrovsky–Piskunov (Fisher–KPP) equations [FISHER1937, Hamel2001, Bian2017, Perthame2007, Berestycki2009b, Houchmandzadeh2017, Wang2021], and quantitative genetics models where populations are structured on a phenotypic and/or a geographical space [Burger1994, Genieys2006a, Berestycki2016, Nordbotten2016, Nordbotten2018, Roques2016, Doebeli2010, Nordbotten2020]. In such models, Neumann boundary conditions are used, e.g., to model the effect of the borders of the geographical domain on the movement of the organisms.

To accurately capture the dynamics of the systems under considerations, it is important to account for their complexity [Doebeli2010]. This increases the dimensionality of the above models, leading to complications in obtaining numerical approximations. The number of dimensions of the PDEs corresponds in finance to the number of financial asset (such as stocks, commodities, exchange rates and interest rates) in the involved portfolio. In evolutionary dynamics, it relates to the dimension of the strategy space. In biology, it coincides, e.g., with the number of genes modelled [Pajaro2017], or with the dimension of the geographical space and/or with the dimension of the phenotypic space over which the organisms are structured. Standard approximation methods for PDEs such as finite difference approximation methods, FEM methods, spectral Galerkin approximation methods, sparse grid approximation methods and Monte Carlo approximation methods all suffer from the so called *curse of dimensionality* [Bellman], meaning that their computational costs increases exponentially in the number of dimensions of the PDE under consideration.

Numerical methods exploiting the stochastic representation of the solution can overcome the curse of dimensionality. Simple Monte Carlo averages of the underlying stochastic processes have been proposed since a long time to solve high-dimensional linear PDEs, such as, e.g., Black Scholes and Kolmogorov PDEs [E2020]. Nonetheless, this approach cannot easily be extended to more challenging nonlinear PDEs. Recently, two novel classes of methods have proved successful in high dimensions, namely deep learning-based and full history recursive multilevel Picard approximations. Deep learning-based methods take advantage of the recent development of deep learning [LeCun2015] and reformulate the PDE as a stochastic learning problem through the Feynman–Kac formula [Beck2017a, Han2018] (for an overview on deep learning-based approximation methods for PDEs; see [Beck2020]). In particular, the deep splitting scheme consists in splitting the differential operator into a linear and a nonlinear part. The PDE approximation is then decomposed into a sequence of separate learning problems. The deep splitting has proven to reasonably approximate the solution of nonlinear PDEs in up to 10000 dimensions [Beck2019]. On the other hand, the multilevel Picard iteration method utilises the Feynman–Kac formula to reformulate the PDE problem as a fixed point equation. It further reduces the complexity of the numerical approximation

of the time integral through a multilevel Monte Carlo approach. Nonetheless, those methods do not, as of now, account for non-localness and Neumann boundary conditions.

Here, we generalise the deep splitting method introduced in [Beck2019] and the full history multilevel Picard approximation method introduced in [Weinan2019, Hutzenthaler2020, Weinan2021] to solve for non-local nonlinear PDEs with Neumann boundary conditions. We handle the non-local term by a plain vanilla Monte Carlo integration and address Neumann boundary conditions by constructing reflected stochastic processes. While the multilevel Picard method can only provide an approximate solution on a single point $x \in D$ in one run, the machine learning-based method can provide an approximate solution on a full subset $[a, b]^d \subseteq D$. We use both methods to solve 5 non-local nonlinear PDEs arising in models from biology and physics, and cross-validate the results of the simulations. We manage to solve the non-local nonlinear PDEs with reasonable accuracy in up to 10 dimensions.

The remainder of the paper is organized as follows. Section 3.2 discusses a special case of the proposed machine learning-based method, for the reader to grasp the general idea of the method. Section 3.3 discusses a general case, which can include batch normalisation. Section 3.4 presents an extension of the multilevel Picard approximation method to non-local nonlinear PDEs, that we use to obtain reference solutions in Section 3.5. Section 3.5 provides numerical simulations for 5 concrete non-local nonlinear PDE examples. ?? provides the source codes used for the PDE examples in Section 3.5.

Approximation scheme

[Zang2019, Kolluru2019]

3.2 Machine learning-based approximation method in a special case

3.2.1 Partial differential equations (PDEs) under consideration

Let $T \in (0, \infty)$, $d \in \mathbb{N}$, let $\mathbb{D} \subseteq \mathbb{R}^d$ be a domain with a sufficiently smooth boundary $\partial_{\mathbb{D}}$, let $\mathbf{n}: \partial_{\mathbb{D}} \rightarrow \mathbb{R}^d$ be the outer unit normal vector field associated to \mathbb{D} , let $g: \mathbb{D} \rightarrow \mathbb{R}$ be continuous, let $\nu_x: \mathbb{D} \rightarrow [0, 1]$, $x \in \mathbb{D}$, be probability measures, let $f: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ be measurable, and let $u = (u(t, x))_{(t, x) \in [0, T] \times \mathbb{D}} \in C^{1,2}([0, T] \times \mathbb{D}, \mathbb{R})$ have at most polynomially growing partial derivatives, assume for every $t \in [0, T]$, $x \in \partial_{\mathbb{D}}$ that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$, and assume for every $t \in [0, T]$, $x \in \mathbb{D}$ that $u(0, x) = g(x)$, $\int_{\mathbb{D}} |f(u(t, x), u(t, \mathbf{x}))| \nu_x(d\mathbf{x}) < \infty$, and

$$(\frac{\partial}{\partial t} u)(t, x) = (\Delta_x u)(t, x) + \int_{\mathbb{D}} f(u(t, x), u(t, \mathbf{x})) \nu_x(d\mathbf{x}). \quad (3.1)$$

Our goal is to approximately calculate under suitable hypotheses the solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.1).

3.2.2 Simulation of time discrete reflected processes

Framework 3.2.1 (Simulation of time discrete reflected processes). Let $d \in \mathbb{N}$, let $\mathbb{D} \subseteq \mathbb{R}^d$ be a domain with a sufficiently smooth boundary $\partial_{\mathbb{D}}$, let $\mathbf{n}: \partial_{\mathbb{D}} \rightarrow \mathbb{R}^d$ be the outer unit normal vector field associated to \mathbb{D} , let $\mathfrak{c}: (\mathbb{R}^d)^2 \rightarrow \mathbb{R}^d$ satisfy for every $a, b \in \mathbb{R}^d$ that

$$\mathfrak{c}(a, b) = a + [\inf(\{r \in [0, 1]: a + r(b - a) \notin \mathbb{D}\} \cup \{1\})](b - a), \quad (3.2)$$

let $\mathcal{R}: (\mathbb{R}^d)^2 \rightarrow (\mathbb{R}^d)^2$ satisfy for every $a, b \in \mathbb{R}^d$ that

$$\mathcal{R}(a, b) = \begin{cases} (\mathfrak{c}(a, b), b - 2\mathbf{n}(\mathfrak{c}(a, b)) \langle b - \mathfrak{c}(a, b), \mathbf{n}(\mathfrak{c}(a, b)) \rangle) & : \mathfrak{c}(a, b) \neq b \\ (b, b) & : \mathfrak{c}(a, b) = b, \end{cases} \quad (3.3)$$

let $P: (\mathbb{R}^d)^2 \rightarrow \mathbb{R}^d$ satisfy for every $a, b \in \mathbb{R}^d$ that $P(a, b) = b$, let $\mathcal{R}_n: (\mathbb{R}^d)^2 \rightarrow (\mathbb{R}^d)^2$, $n \in \mathbb{N}_0 = \{0\} \cup \mathbb{N}$, satisfy for every $n \in \mathbb{N}_0$, $x, y \in \mathbb{R}^d$ that $\mathcal{R}_0(x, y) = (x, y)$ and $\mathcal{R}_{n+1}(x, y) = \mathcal{R}(\mathcal{R}_n(x, y))$, and let $R: (\mathbb{R}^d)^2 \rightarrow \mathbb{R}^d$ satisfy for every $x, y \in \mathbb{R}^d$ that

$$R(x, y) = \lim_{n \rightarrow \infty} P(\mathcal{R}_n(x, y)). \quad (3.4)$$

3.2.3 Description of the proposed approximation method in a special case

Framework 3.2.2 (Special case of the machine learning-based approximation method). Assume Lemma 3.2.1, let $T, \gamma \in (0, \infty)$, $N, M, K \in \mathbb{N}$, $g \in C^2(\mathbb{R}^d, \mathbb{R})$, $\mathfrak{h} \in \mathbb{N} \setminus \{1\}$, $\nu = \mathfrak{h}(N+1)d(d+1)$, $t_0, t_1, \dots, t_N \in [0, T]$ satisfy

$$0 = t_0 < t_1 < \dots < t_N = T, \quad (3.5)$$

let $\tau_0, \tau_1, \dots, \tau_n \in [0, T]$ satisfy for every $n \in \{0, 1, \dots, N\}$ that $\tau_n = T - t_{N-n}$, let $f: \mathbb{R} \times \mathbb{R} \rightarrow \mathbb{R}$ be measurable, let $(\Omega, \mathcal{F}, \mathbb{P}, (\mathcal{F}_t)_{t \in [0, T]})$ be a filtered probability space, let $\xi^m: \Omega \rightarrow \mathbb{R}^d$, $m \in \mathbb{N}$, be i.i.d. $\mathcal{F}_0/\mathcal{B}(\mathbb{R}^d)$ -measurable random variables, let $B^m: [0, T] \times \Omega \rightarrow \mathbb{R}^d$, $m \in \mathbb{N}$, be i.i.d. standard $(\mathcal{F}_t)_{t \in [0, T]}$ -Brownian motions, for every $m \in \mathbb{N}$ let $\mathcal{Y}^m: \{0, 1, \dots, N\} \times \Omega \rightarrow \mathbb{R}^d$ be the stochastic process which satisfies for every $n \in \{0, 1, \dots, N-1\}$ that $\mathcal{Y}_0^m = \xi^m$ and

$$\mathcal{Y}_{n+1}^m = R(\mathcal{Y}_n^m, \mathcal{Y}_n^m + \sqrt{2}(B_{\tau_{n+1}}^m - B_{\tau_n}^m)), \quad (3.6)$$

let $\mathcal{L}_d: \mathbb{R}^d \rightarrow \mathbb{R}^d$ satisfy for every $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ that

$$\mathcal{L}_d(x) = \left(\frac{\exp(x_1)}{\exp(x_1) + 1}, \dots, \frac{\exp(x_d)}{\exp(x_d) + 1} \right), \quad (3.7)$$

for every $\theta = (\theta_1, \dots, \theta_\nu) \in \mathbb{R}^\nu$, $k, l, v \in \mathbb{N}$ with $v + l(k+1) \leq \nu$ let $A_{k,l}^{\theta,v}: \mathbb{R}^k \rightarrow \mathbb{R}^l$ satisfy for every $x = (x_1, \dots, x_k) \in \mathbb{R}^k$ that

$$A_{k,l}^{\theta,v}(x) = \left(\theta_{v+k+l+1} + \left[\sum_{i=1}^k x_i \theta_{v+i} \right], \dots, \theta_{v+k+l+l} + \left[\sum_{i=1}^k x_i \theta_{v+(l-1)k+i} \right] \right), \quad (3.8)$$

let $\mathbb{V}_n: \mathbb{R}^\nu \times \mathbb{R}^d \rightarrow \mathbb{R}$, $n \in \{0, 1, \dots, N\}$, satisfy for every $n \in \{1, 2, \dots, N\}$, $\theta \in \mathbb{R}^\nu$, $x \in \mathbb{R}^d$ that $\mathbb{V}_0(\theta, x) = g(x)$ and

$$\begin{aligned} \mathbb{V}_n(\theta, x) &= \\ &\left(A_{d,1}^{\theta,(\mathfrak{h}n+\mathfrak{h}-1)d(d+1)} \circ \mathcal{L}_d \circ A_{d,d}^{\theta,(\mathfrak{h}n+\mathfrak{h}-2)d(d+1)} \circ \dots \circ \mathcal{L}_d \circ A_{d,d}^{\theta,(\mathfrak{h}n+1)d(d+1)} \circ \mathcal{L}_d \circ A_{d,d}^{\theta,\mathfrak{h}nd(d+1)} \right)(x), \end{aligned} \quad (3.9)$$

let $\nu_x: \mathcal{B}(\mathbb{D}) \rightarrow [0, 1]$, $x \in \mathbb{D}$, be probability measures, for every $x \in \mathbb{D}$ let $Z_{x,k}^{n,m}: \Omega \rightarrow \mathbb{D}$, $k, n, m \in \mathbb{N}$, be i.i.d. random variables which satisfy for every $A \in \mathcal{B}(\mathbb{D})$ that $\mathbb{P}(Z_{x,1}^{1,1} \in A) = \nu_x(A)$, let $\Theta^n: \mathbb{N}_0 \times \Omega \rightarrow \mathbb{R}^\nu$, $n \in \{0, 1, \dots, N\}$, be stochastic processes, for every $n \in \{1, 2, \dots, N\}$, $m \in \mathbb{N}$ let $\phi^{n,m}: \mathbb{R}^\nu \times \Omega \rightarrow \mathbb{R}$ satisfy for every $\theta \in \mathbb{R}^\nu$, $\omega \in \Omega$ that

$$\begin{aligned} \phi^{n,m}(\theta, \omega) &= \left[\mathbb{V}_n(\theta, \mathcal{Y}_{N-n}^m(\omega)) - \mathbb{V}_{n-1}(\Theta_M^{n-1}(\omega), \mathcal{Y}_{N-n+1}^m(\omega)) \right. \\ &\quad \left. - \frac{(t_n - t_{n-1})}{K} \left[\sum_{k=1}^K f(\mathbb{V}_{n-1}(\Theta_M^{n-1}(\omega), \mathcal{Y}_{N-n+1}^m(\omega)), \mathbb{V}_{n-1}(\Theta_M^{n-1}(\omega), Z_{\mathcal{Y}_{N-n+1}^m,k}^{n,m}(\omega))) \right] \right]^2, \end{aligned} \quad (3.10)$$

for every $n \in \{1, 2, \dots, N\}$, $m \in \mathbb{N}$ let $\Phi^{n,m}: \mathbb{R}^\nu \times \Omega \rightarrow \mathbb{R}^\nu$ satisfy for every $\theta \in \mathbb{R}^\nu$, $\omega \in \Omega$ that $\Phi^{n,m}(\theta, \omega) = (\nabla_\theta \phi^{n,m})(\theta, \omega)$, and assume for every $n \in \{1, 2, \dots, N\}$, $m \in \mathbb{N}$ that

$$\Theta_m^n = \Theta_{m-1}^n - \gamma \cdot \Phi^{n,m}(\Theta_{m-1}^n). \quad (3.11)$$

In the setting of Lemma 3.2.2 we think under suitable hypotheses for sufficiently large $N, M, K \in \mathbb{N}$, sufficiently small $\gamma \in (0, \infty)$, every $n \in \{0, 1, \dots, N\}$, and every $x \in \mathbb{R}^d$ of $\mathbb{V}_n(\Theta_M^n, x): \Omega \rightarrow \mathbb{R}$ as a suitable approximation

$$\mathbb{V}_n(\Theta_M^n, x) \approx u(t_n, x) \quad (3.12)$$

of $u(t_n, x)$ where $u = (u(t, x))_{(t,x) \in [0,T] \times \mathbb{D}} \in C^{1,2}([0, T] \times \mathbb{D}, \mathbb{R})$ is a function with at most polynomially growing derivatives which satisfies for every $t \in [0, T]$, $x \in \partial_{\mathbb{D}}$ that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$ and which satisfies for every $t \in [0, T]$, $x \in \mathbb{D}$ that $u(0, x) = g(x)$, $\int_{\mathbb{D}} |f(u(t, x), u(t, \mathbf{x}))| \nu_x(d\mathbf{x}) < \infty$, and

$$(\frac{\partial}{\partial t} u)(t, x) = (\Delta_x u)(t, x) + \int_{\mathbb{D}} f(u(t, x), u(t, \mathbf{x})) \nu_x(d\mathbf{x}) \quad (3.13)$$

(cf. (3.1)). In (3.11) in Lemma 3.2.2 we merely employ the plain vanilla stochastic gradient descent (SGD) optimisation method to train the considered neural networks (see (3.10) and (3.11) in Lemma 3.2.2). We refer to (3.19) and (3.20) in Lemma 3.3.1 for a set-up covering

more general SGD type optimisation methods such as the Adam optimizer (see Kingma & Ba [Kingma2014]).

3.3 Machine learning-based approximation method in the general case

3.3.1 PDEs under consideration

Let $T \in (0, \infty)$, $d \in \mathbb{N}$, let $\mathbb{D} \subseteq \mathbb{R}^d$ be a domain with a sufficiently smooth boundary $\partial_{\mathbb{D}}$, let $\mathbf{n}: \partial_{\mathbb{D}} \rightarrow \mathbb{R}^d$ be the outer unit normal vector field associated to \mathbb{D} , let $g: \mathbb{D} \rightarrow \mathbb{R}$, $\mu: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}^d$, and $\sigma: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}^{d \times d}$ be continuous, let $\nu_x: \mathbb{D} \rightarrow [0, 1]$, $x \in \mathbb{D}$, be probability measures, let $f: [0, T] \times \mathbb{D} \times \mathbb{D} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ be measurable, let $u = (u(t, x))_{(t,x) \in [0,T] \times \mathbb{D}} \in C^{1,2}([0, T] \times \mathbb{D}, \mathbb{R})$ have at most polynomially growing partial derivatives, assume for every $t \in [0, T]$, $x \in \partial_{\mathbb{D}}$ that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$, and assume for every $t \in [0, T]$, $x \in \mathbb{D}$ that $u(0, x) = g(x)$, $\int_{\mathbb{D}} |f(t, x, \mathbf{x}, u(t, x), u(t, \mathbf{x}), (\nabla_x u)(t, x), (\nabla_x u)(t, \mathbf{x}))| \nu_x(d\mathbf{x}) < \infty$, and

$$\begin{aligned} (\frac{\partial}{\partial t} u)(t, x) &= \int_{\mathbb{D}} f(t, x, \mathbf{x}, u(t, x), u(t, \mathbf{x}), (\nabla_x u)(t, x), (\nabla_x u)(t, \mathbf{x})) \nu_x(d\mathbf{x}) \\ &\quad + \langle \mu(t, x), (\nabla_x u)(t, x) \rangle + \frac{1}{2} \text{Trace}(\sigma(t, x)[\sigma(t, x)]^*(\text{Hess}_x u)(t, x)). \end{aligned} \quad (3.14)$$

Our goal is to approximately calculate under suitable hypotheses the solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.14).

3.3.2 Description of the proposed approximation method in the general case

Framework 3.3.1 (General case of the machine learning-based approximation method). Assume Lemma 3.2.1, let $T \in (0, \infty)$, $N, d, \varrho, \nu, \varsigma \in \mathbb{N}$, $(M_n)_{n \in \{0, 1, \dots, N\}} \subseteq \mathbb{N}$, $(K_n)_{n \in \mathbb{N}} \subseteq \mathbb{N}$, $(J_m)_{m \in \mathbb{N}} \subseteq \mathbb{N}$, $t_0, t_1, \dots, t_N \in [0, T]$ satisfy

$$0 = t_0 < t_1 < \dots < t_N = T, \quad (3.15)$$

let $\tau_0, \tau_1, \dots, \tau_n \in [0, T]$ satisfy for every $n \in \{0, 1, \dots, N\}$ that $\tau_n = T - t_{N-n}$, let $\nu_x: \mathcal{B}(\mathbb{D}) \rightarrow [0, 1]$, $x \in \mathbb{D}$, be probability measures, for every $x \in \mathbb{D}$ let $Z_{x,k}^{n,m,j}: \Omega \rightarrow \mathbb{D}$, $k, n, m, j \in \mathbb{N}$, be i.i.d. random variables which satisfy for every $A \in \mathcal{B}(\mathbb{D})$ that $\mathbb{P}(Z_{x,1}^{1,1,1} \in A) = \nu_x(A)$, let $f: [0, T] \times \mathbb{D} \times \mathbb{D} \times \mathbb{R} \times \mathbb{R} \times \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}$ be a function, let $(\Omega, \mathcal{F}, \mathbb{P}, (\mathcal{F}_t)_{t \in [0, T]})$ be a filtered probability space, for every $n \in \{1, 2, \dots, N\}$ let $B^{n,m,j}: [0, T] \times \Omega \rightarrow \mathbb{R}^d$, $m, j \in \mathbb{N}$, be i.i.d. standard $(\mathcal{F}_t)_{t \in [0, T]}$ -Brownian motions, for every $n \in \{1, 2, \dots, N\}$ let $\xi^{n,m,j}: \Omega \rightarrow \mathbb{R}^d$,

$m, j \in \mathbb{N}$, be i.i.d. $\mathcal{F}_0/\mathcal{B}(\mathbb{R}^d)$ -measurable random variables, let $H: [0, T]^2 \times \mathbb{R}^d \times \mathbb{R}^d \rightarrow \mathbb{R}^d$ be a function, let $\mathbb{V}_n^{j,s}: \mathbb{R}^\nu \times \mathbb{R}^d \rightarrow \mathbb{R}$, $(j, s, n) \in \mathbb{N} \times \mathbb{R}^c \times \{0, 1, \dots, N\}$, be functions, for every $n \in \{1, 2, \dots, N\}$, $m, j \in \mathbb{N}$ let $\mathcal{Y}^{n,m,j}: \{0, 1, \dots, N\} \times \Omega \rightarrow \mathbb{R}^d$ be a stochastic process which satisfies for every $k \in \{0, 1, \dots, N-1\}$ that $\mathcal{Y}_0^{n,m,j} = \xi^{n,m,j}$ and

$$\mathcal{Y}_{k+1}^{n,m,j} = H(\tau_{k+1}, \tau_k, \mathcal{Y}_k^{n,m,j}, B_{\tau_{k+1}}^{n,m,j} - B_{\tau_k}^{n,m,j}), \quad (3.16)$$

let $\Theta^n: \mathbb{N}_0 \times \Omega \rightarrow \mathbb{R}^\nu$, $n \in \{0, 1, \dots, N\}$, be stochastic processes, for every $n \in \{1, 2, \dots, N\}$, $m \in \mathbb{N}$, $s \in \mathbb{R}^c$ let $\phi^{n,m,s}: \mathbb{R}^\nu \times \Omega \rightarrow \mathbb{R}$ satisfy for every $\theta \in \mathbb{R}^\nu$, $\omega \in \Omega$ that

$$\begin{aligned} \phi^{n,m,s}(\theta, \omega) &= \frac{1}{J_m} \sum_{j=1}^{J_m} \left[\mathbb{V}_n^{j,s}(\theta, \mathcal{Y}_{N-n}^{n,m,j}(\omega)) - \mathbb{V}_{n-1}^{j,s}(\Theta_{M_{n-1}}^{n-1}(\omega), \mathcal{Y}_{N-n+1}^{n,m,j}(\omega)) \right. \\ &\quad - \frac{(t_n - t_{n-1})}{K_n} \left[\sum_{k=1}^{K_n} f(t_{n-1}, \mathcal{Y}_{N-n+1}^{n,m,j}(\omega), Z_{\mathcal{Y}_{N-n+1}^{n,m,j}, k}^{n,m,j}(\omega), \right. \\ &\quad \left. \left. \mathbb{V}_{n-1}^{j,s}(\Theta_{M_{n-1}}^{n-1}(\omega), \mathcal{Y}_{N-n+1}^{n,m,j}(\omega)), \mathbb{V}_{n-1}^{j,s}(\Theta_{M_{n-1}}^{n-1}(\omega), Z_{\mathcal{Y}_{N-n+1}^{n,m,j}, k}^{n,m,j}(\omega)), \right. \right. \\ &\quad \left. \left. (\nabla_x \mathbb{V}_{n-1}^{j,s})(\Theta_{M_{n-1}}^{n-1}(\omega), \mathcal{Y}_{N-n+1}^{n,m,j}(\omega)), (\nabla_x \mathbb{V}_{n-1}^{j,s})(\Theta_{M_{n-1}}^{n-1}(\omega), Z_{\mathcal{Y}_{N-n+1}^{n,m,j}, k}^{n,m,j}(\omega)) \right] \right]^2, \end{aligned} \quad (3.17)$$

for every $n \in \{1, 2, \dots, N\}$, $m \in \mathbb{N}$, $s \in \mathbb{R}^c$ let $\Phi^{n,m,s}: \mathbb{R}^\nu \times \Omega \rightarrow \mathbb{R}^\nu$ satisfy for every $\omega \in \Omega$, $\theta \in \{\eta \in \mathbb{R}^\nu: \phi^{n,m,s}(\cdot, \omega): \mathbb{R}^\nu \rightarrow \mathbb{R} \text{ is differentiable at } \eta\}$ that

$$\Phi^{n,m,s}(\theta, \omega) = (\nabla_\theta \phi^{n,m,s})(\theta, \omega), \quad (3.18)$$

let $\mathcal{S}^n: \mathbb{R}^c \times \mathbb{R}^\nu \times (\mathbb{R}^d)^{\{0,1,\dots,N\} \times \mathbb{N}} \rightarrow \mathbb{R}^c$, $n \in \{1, 2, \dots, N\}$, be functions, for every $n \in \{1, 2, \dots, N\}$, $m \in \mathbb{N}$ let $\psi_m^n: \mathbb{R}^\varrho \rightarrow \mathbb{R}^\nu$ and $\Psi_m^n: \mathbb{R}^\varrho \times \mathbb{R}^\nu \rightarrow \mathbb{R}^\varrho$ be functions, and for every $n \in \{1, 2, \dots, N\}$ let $\mathbb{S}^n: \mathbb{N}_0 \times \Omega \rightarrow \mathbb{R}^c$ and $\Xi^n: \mathbb{N}_0 \times \Omega \rightarrow \mathbb{R}^\varrho$ be stochastic processes which satisfy for every $m \in \mathbb{N}$ that

$$\mathbb{S}_m^n = \mathcal{S}^n(\mathbb{S}_{m-1}^n, \Theta_{m-1}^n, (\mathcal{Y}_k^{n,m,i})_{(k,i) \in \{0,1,\dots,N\} \times \mathbb{N}}), \quad (3.19)$$

$$\Xi_m^n = \Psi_m^n(\Xi_{m-1}^n, \Phi^{n,m,\mathbb{S}_m^n}(\Theta_{m-1}^n)), \quad \text{and} \quad \Theta_m^n = \Theta_{m-1}^n - \psi_m^n(\Xi_m^n). \quad (3.20)$$

In the setting of Lemma 3.3.1 we think under suitable hypotheses for sufficiently large $N \in \mathbb{N}$, sufficiently large $(M_n)_{n \in \{0,1,\dots,N\}} \subseteq \mathbb{N}$, sufficiently large $(K_n)_{n \in \mathbb{N}} \subseteq \mathbb{N}$, every $n \in \{0, 1, \dots, N\}$, and every $x \in \mathbb{R}^d$ of $\mathbb{V}_n^{1,\mathbb{S}_{M_n}^n}(\Theta_{M_n}^n, x): \Omega \rightarrow \mathbb{R}$ as a suitable approximation

$$\mathbb{V}_n^{1,\mathbb{S}_{M_n}^n}(\Theta_{M_n}^n, x) \approx u(t_n, x) \quad (3.21)$$

of $u(t_n, x)$ where $u = (u(t, x))_{(t,x) \in [0,T] \times \mathbb{R}^d} \in C^{1,2}([0, T] \times \mathbb{R}^d, \mathbb{R})$ is a function with at most polynomially growing derivatives which satisfies for every $t \in [0, T]$, $x \in \partial_D$ that

$\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$ and which satisfies for every $t \in [0, T]$, $x \in \mathbb{D}$ that $u(0, x) = g(x)$, $\int_{\mathbb{D}} |f(t, x, \mathbf{x}, u(t, x), u(t, \mathbf{x}), (\nabla_x u)(t, x), (\nabla_x u)(t, \mathbf{x}))| \nu_x(d\mathbf{x}) < \infty$, and

$$\begin{aligned} \left(\frac{\partial}{\partial t} u\right)(t, x) &= \int_{\mathbb{D}} f(t, x, \mathbf{x}, u(t, x), u(t, \mathbf{x}), (\nabla_x u)(t, x), (\nabla_x u)(t, \mathbf{x})) \nu_x(d\mathbf{x}) \\ &\quad + \langle \mu(t, x), (\nabla_x u)(t, x) \rangle + \frac{1}{2} \operatorname{Trace}(\sigma(t, x)[\sigma(t, x)]^*(\operatorname{Hess}_x u)(t, x)) \end{aligned} \quad (3.22)$$

(cf. (3.14)). The role of the processes $\mathbb{S}^n: \mathbb{N}_0 \times \Omega \rightarrow \mathbb{R}^s$, $n \in \{1, 2, \dots, N\}$, is to describe the variables needed for batch normalization.

3.4 Multilevel Picard approximation method for non-local PDEs

3.4.1 Description of the proposed approximation method

Framework 3.4.1 (Multilevel Picard approximation method). *Assume Lemma 3.2.1, let $\langle \cdot, \cdot \rangle: (\cup_{n \in \mathbb{N}} (\mathbb{R}^n \times \mathbb{R}^n)) \rightarrow \mathbb{R}$ and $\|\cdot\|: (\cup_{n \in \mathbb{N}} \mathbb{R}^n) \rightarrow \mathbb{R}$ satisfy for every $x = (x_1, \dots, x_d)$, $y = (y_1, \dots, y_d) \in \mathbb{R}^d$ that*

$$\langle x, y \rangle = \sum_{j=1}^d x_j y_j \quad \text{and} \quad \|x\| = \left[\sum_{j=1}^d |x_j|^2 \right]^{1/2}, \quad (3.23)$$

let $c, T \in (0, \infty)$, $\Theta = \cup_{n \in \mathbb{N}} \mathbb{Z}^n$, $f \in C([0, T] \times D \times D \times \mathbb{R} \times \mathbb{R}, \mathbb{R})$, $g \in C(D, \mathbb{R})$, $u \in C([0, T] \times \mathbb{D}, \mathbb{R})$, let $(K_{n,l,m})_{n,l,m \in \mathbb{N}_0} \subseteq \mathbb{N}$, let $\nu_x: \mathcal{B}(\mathbb{D}) \rightarrow [0, 1]$, $x \in \mathbb{D}$, be probability measures, for every $x \in \mathbb{D}$ let $Z_x^\theta: \Omega \rightarrow \mathbb{D}$, $\theta \in \Theta$, be i.i.d. random variables, assume for every $A \in \mathcal{B}(\mathbb{D})$, $\theta \in \Theta$ that $\mathbb{P}(Z_x^\theta \in A) = \nu_x(A)$, let $\phi_r: \mathbb{R} \rightarrow \mathbb{R}$, $r \in [0, \infty]$, satisfy for every $r \in [0, \infty]$, $y \in \mathbb{R}$ that

$$\phi_r(y) = \min\{r, \max\{-r, y\}\}, \quad (3.24)$$

let $(\Omega, \mathcal{F}, \mathbb{P})$ be a probability space, let $\mathcal{V}^\theta: \Omega \rightarrow (0, 1)$, $\theta \in \Theta$, be independent $\mathcal{U}_{(0,1)}$ -distributed random variables, let $V^\theta: [0, T] \times \Omega \rightarrow [0, T]$, $\theta \in \Theta$, satisfy for every $t \in [0, T]$, $\theta \in \Theta$ that

$$V_t^\theta = t + (T - t)\mathcal{V}^\theta, \quad (3.25)$$

let $W^\theta: [0, T] \times \Omega \rightarrow \mathbb{R}^d$, $\theta \in \Theta$, be independent standard Brownian motions, assume that $(\mathcal{V}^\theta)_{\theta \in \Theta}$ and $(W^\theta)_{\theta \in \Theta}$ are independent, let $\mu: \mathbb{R}^d \rightarrow \mathbb{R}^d$ and $\sigma: \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$ be globally Lipschitz continuous, let $X_t^{x,\theta} = (X_{t,s}^{x,\theta})_{s \in [t, T]}: [t, T] \times \Omega \rightarrow \mathbb{R}^d$, $x \in \mathbb{R}^d$, $\theta \in \Theta$, $t \in [0, T]$, be

stochastic processes with continuous sample paths, let $U_{n,M,r}^\theta: [0, T] \times \mathbb{R}^d \times \Omega \rightarrow \mathbb{R}^k$, $\theta \in \Theta$, $n, M \in \mathbb{N}_0$, $r \in [0, \infty]$, satisfy for every $\theta \in \Theta$, $n, M \in \mathbb{N}_0$, $r \in [0, \infty]$, $t \in [0, T]$, $x \in \mathbb{R}^d$ that

$$\begin{aligned} U_{n,M,r}^\theta(t, x) = & \sum_{l=0}^{n-1} \frac{(T-t)}{M^{n-l}} \sum_{m=1}^{M^{n-l}} \frac{1}{K_{n,l,m}} \sum_{k=1}^{K_{n,l,m}} \left[f\left(V_t^{(\theta,l,m)}, X_{t,C_t^{(\theta,l,m)}}^{x,(\theta,l,m)}, Z_{X_{t,V_t^{(\theta,l,m)}}^{x,(\theta,l,m)}}^{(\theta,l,m,k)}, \right. \right. \\ & \phi_r\left(U_{l,M,r}^{(\theta,l,m)}(V_t^{(\theta,l,m)}, X_{t,V_t^{(\theta,l,m)}}^{x,(\theta,l,m)})\right), \phi_r\left(U_{l,M,r}^{(\theta,l,m)}(V_t^{(\theta,l,m)}, Z_{X_{t,V_t^{(\theta,l,m)}}^{x,(\theta,l,m)}}^{(\theta,l,m,k)})\right) \Big) \\ & - \mathbb{1}_{\mathbb{N}}(l) f\left(V_t^{(\theta,l,m)}, X_{t,V_t^{(\theta,l,m)}}^{x,(\theta,l,m)}, Z_{X_{t,V_t^{(\theta,l,m)}}^{x,(\theta,l,m)}}^{(\theta,l,m,k)}, \phi_r\left(U_{\max\{l-1,0\},M,r}^{(\theta,l,-m)}(V_t^{(\theta,l,m)}, X_{t,V_t^{(\theta,l,m)}}^{x,(\theta,l,m)})\right)\right), \\ & \left. \phi_r\left(U_{\max\{l-1,0\},M,r}^{(\theta,l,-m)}(V_t^{(\theta,l,m)}, Z_{X_{t,V_t^{(\theta,l,m)}}^{x,(\theta,l,m)}}^{(\theta,l,m,k)})\right)\right) \Big] + \frac{\mathbb{1}_{\mathbb{N}}(n)}{M^n} \left[\sum_{m=1}^{M^n} g(X_{t,T}^{x,(\theta,0,-m)}) \right], \\ (3.26) \end{aligned}$$

assume for every $t \in [0, T]$, $x \in \partial_D$ that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$, assume for every $t \in [0, T]$, $x = (x_1, \dots, x_d) \in \mathbb{D}$ that $u|_{[0,T] \times \mathbb{D}} \in C^{1,2}([0, T] \times \mathbb{D}, \mathbb{R})$, $\|u(t, x)\| \leq c(1 + \|x\|^c)$, $u(T, x) = g(x)$, and

$$\begin{aligned} (\frac{\partial}{\partial t} u)(t, x) + \int_D f(t, x, \mathbf{x}, u(t, x), u(t, \mathbf{x})) \nu_x(d\mathbf{x}) + \frac{1}{2} \text{Trace}(\sigma(x)[\sigma(x)]^*(\text{Hess}_x u)(t, x)) \\ + \langle \mu(x), (\nabla_x u)(t, x) \rangle = 0. \quad (3.27) \end{aligned}$$

3.4.2 Examples for the approximation method

Example 3.4.2 (Fisher–KPP PDEs with Neumann boundary conditions). In this example we specialize Lemma 3.4.1 in the case of certain Fisher–KPP PDEs with Neumann boundary conditions (cf., e.g., Bian et al. [Bian2017] and Wang et al. [Wang2021]).

Assume Lemma 3.4.1, let $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $D = [-\frac{1}{2}, \frac{1}{2}]^d$, and $T \in \{1/5, 1/2, 1\}$, assume for every $n, l, m \in \mathbb{N}$ that $K_{n,l,m} = 10$, assume for every $s, t \in [0, T]$, $x, \mathbf{x} \in \mathbb{D}$, $y, \mathbf{y} \in \mathbb{R}$, $v, z, \mathbf{z} \in \mathbb{R}^d$ that $g(x) = \exp(-\frac{1}{2}\|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \epsilon v$, and $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = y(1 - y)$, and assume that for every $x \in \mathbb{R}^d$, $\theta \in \Theta$, $t \in [0, T]$, $s \in [t, T]$ it holds \mathbb{P} -a.s. that

$$X_{t,s}^{x,\theta} = R\left(x, x + \int_t^s \mu(X_{t,r}^{x,\theta}) dr + \int_t^s \sigma(X_{t,r}^{x,\theta}) dW_r^\theta\right). \quad (3.28)$$

The solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.27) then satisfies that for every $t \in [0, T]$, $x \in \partial_D$ it holds that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$ and that for every $t \in [0, T]$, $x \in \mathbb{D}$ it holds that $u(0, x) = \exp(-\frac{1}{2}\|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + u(t, x)(1 - u(t, x)). \quad (3.29)$$

Example 3.4.3 (Non-local competition PDEs). *In this example we specialize Lemma 3.4.1 in the case of certain non-local competition PDEs (cf., e.g., Doebeli & Ispolatov [Doebeli2010], Berestycki et al. [Berestycki2009b], Perthame & Génieys [Perthame2007], and Génieys et al. [Genieys2006a]).*

Assume Lemma 3.4.1, let $\mathfrak{s} = \frac{1}{10}$, $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $D = \mathbb{R}^d$, and $T \in \{1/5, 1/2, 1\}$, assume for every $n, l, m \in \mathbb{N}$ that $K_{n,l,m} = 10$, assume for every $x \in D$, $A \in \mathcal{B}(D)$ that $\nu_x(A) = \frac{1}{(\pi)^{d/2}\mathfrak{s}^d} \int_A \exp(-\mathfrak{s}^{-2} \|x - \mathbf{x}\|^2) d\mathbf{x}$, assume for every $s, t \in [0, T]$, $x, \mathbf{x} \in \mathbb{D}$, $y, \mathbf{y} \in \mathbb{R}$, $v, z, \mathbf{z} \in \mathbb{R}^d$ that $g(x) = \exp(-\frac{1}{2}\|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \epsilon v$, and $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = y(1 - \mathbf{y}\pi^{d/2}\mathfrak{s}^d)$, and assume that for every $x \in \mathbb{R}^d$, $\theta \in \Theta$, $t \in [0, T]$, $s \in [t, T]$ it holds \mathbb{P} -a.s. that

$$X_{t,s}^{x,\theta} = x + \int_t^s \mu(X_{t,r}^{x,\theta}) dr + \int_t^s \sigma(X_{t,r}^{x,\theta}) dW_r^\theta. \quad (3.30)$$

The solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.27) then satisfies that for every $t \in [0, T]$, $x \in \mathbb{D}$ it holds that $u(0, x) = \exp(-\frac{1}{2}\|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + u(t, x) \left(1 - \int_{\mathbb{R}^d} u(t, \mathbf{x}) e^{-\frac{\|\mathbf{x}-x\|^2}{\mathfrak{s}^2}} d\mathbf{x} \right). \quad (3.31)$$

Example 3.4.4 (Non-local sine-Gordon PDEs). *In this example we specialize Lemma 3.4.1 in the case of certain non-local sine-Gordon type PDEs (cf., e.g., Hairer & Shen [Hairer2016], Barone et al. [Barone1971], and Coleman [Coleman1994]).*

Assume Lemma 3.4.1, let $\mathfrak{s} = \frac{1}{10}$, $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $D = \mathbb{R}^d$, and $T \in \{1/5, 1/2, 1\}$, assume for every $n, l, m \in \mathbb{N}$ that $K_{n,l,m} = 1$, assume for every $x \in D$, $A \in \mathcal{B}(D)$ that $\nu_x(A) = \frac{1}{(\pi)^{d/2}\mathfrak{s}^d} \int_A \exp(-\mathfrak{s}^{-2} \|x - \mathbf{x}\|^2) d\mathbf{x}$, assume for every $s, t \in [0, T]$, $x, \mathbf{x} \in \mathbb{D}$, $y, \mathbf{y} \in \mathbb{R}$, $v, z, \mathbf{z} \in \mathbb{R}^d$ that $g(x) = \exp(-\frac{1}{2}\|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \epsilon v$, and $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = \sin(y) - \mathbf{y}\pi^{d/2}\mathfrak{s}^d$, and assume that for every $x \in \mathbb{R}^d$, $\theta \in \Theta$, $t \in [0, T]$, $s \in [t, T]$ it holds \mathbb{P} -a.s. that

$$X_{t,s}^{x,\theta} = x + \int_t^s \mu(X_{t,r}^{x,\theta}) dr + \int_t^s \sigma(X_{t,r}^{x,\theta}) dW_r^\theta. \quad (3.32)$$

The solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.27) then satisfies that for every $t \in [0, T]$, $x \in \mathbb{D}$ it holds that $u(0, x) = \exp(-\frac{1}{2}\|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + \sin(u(t, x)) - \int_{\mathbb{R}^d} u(t, \mathbf{x}) e^{-\frac{\|\mathbf{x}-x\|^2}{\mathfrak{s}^2}} d\mathbf{x}. \quad (3.33)$$

Example 3.4.5 (Replicator-mutator PDEs). *In this example we specialize Lemma 3.4.1 in the case of certain d -dimensional replicator-mutator PDEs (cf., e.g., Hamel et al. [Hamel2020]).*

Assume Lemma 3.4.1, let $\mathfrak{m}_1, \mathfrak{m}_2, \dots, \mathfrak{m}_d, \mathfrak{s}_1, \mathfrak{s}_2, \dots, \mathfrak{s}_d, \mathfrak{u}_1, \mathfrak{u}_2, \dots, \mathfrak{u}_d \in \mathbb{R}$ satisfy for every $k \in \{1, 2, \dots, d\}$ that $\mathfrak{m}_k = \frac{1}{10}$, $\mathfrak{s}_k = \frac{1}{4\pi}$, and $\mathfrak{u}_k = 0$, assume that $d \in \{1, 2, 5, 10\}$, $\mathbb{D} = \mathbb{R}^d$, and $T \in \{1/5, 1/2, 1\}$, assume for every $n, l, m \in \mathbb{N}$ that $K_{n,l,m} = 10$, let $a: \mathbb{R}^d \rightarrow \mathbb{R}$

satisfy for every $x \in \mathbb{R}^d$ that $a(x) = -\frac{1}{2}\|x\|^2$, assume for every $x \in D$, $A \in \mathcal{B}(D)$ that $\nu_x(A) = \frac{1}{\pi^{d/2}} \int_A \exp(-\|\mathbf{x}\|^2) d\mathbf{x}$, assume for every $s, t \in [0, T]$, $x = (x_1, \dots, x_d)$, $\mathbf{x} \in D$, $v = (v_1, \dots, v_d)$, $y, \mathbf{y} \in \mathbb{R}$ that $g(x) = [\pi^{d/2} \prod_{i=1}^d (\mathfrak{s}_i)^{1/2}]^{-1} \exp(-\sum_{i=1}^d \frac{(x_i - \mathfrak{u}_i)^2}{2\mathfrak{s}_i})$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = (\mathfrak{m}_1 v_1, \dots, \mathfrak{m}_d v_d)$, and

$$f(t, x, \mathbf{x}, y, \mathbf{y}) = y \left(a(x) - \pi^{d/2} \mathbf{y} a(\mathbf{x}) \exp(\|\mathbf{x}\|^2) \right), \quad (3.34)$$

and assume that for every $x \in \mathbb{R}^d$, $\theta \in \Theta$, $t \in [0, T]$, $s \in [t, T]$ it holds \mathbb{P} -a.s. that

$$X_{t,s}^{x,\theta} = x + \int_t^s \mu(X_{t,r}^{x,\theta}) dr + \int_t^s \sigma(X_{t,r}^{x,\theta}) dW_r^\theta. \quad (3.35)$$

The solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.27) then satisfies that for every $t \in [0, T]$, $x = (x_1, \dots, x_d) \in \mathbb{D}$ it holds that $u(0, x) = (2\pi)^{-d/2} [\prod_{i=1}^d |\mathfrak{s}_i|^{-1/2}] \exp(-\sum_{i=1}^d \frac{(x_i - \mathfrak{u}_i)^2}{2\mathfrak{s}_i})$ and

$$(\frac{\partial}{\partial t} u)(t, x) = u(t, x) \left(a(x) - \int_D u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) + \sum_{i=1}^d \frac{1}{2} |\mathfrak{m}_i|^2 (\frac{\partial^2}{\partial x_i^2} u)(t, x). \quad (3.36)$$

Example 3.4.6 (Allen–Cahn PDEs with conservation of mass). In this example we specialize Lemma 3.4.1 in the case of certain Allen–Cahn PDEs with cubic nonlinearity, conservation of mass, and no-flux boundary conditions (cf, e.g., Rubinstein & Sternberg [RUBINSTEIN1992]).

Assume Lemma 3.4.1, let $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $D = [0, 1]^d$, and $T \in \{1/5, 1/2, 1\}$, assume for every $n, l, m \in \mathbb{N}$ that $K_{n,l,m} = 10$, assume for every $x \in D$, $A \in \mathcal{B}(D)$ that $\nu_x(A) = \int_A d\mathbf{x}$, assume for every $s, t \in [0, T]$, $x, \mathbf{x} \in \mathbb{D}$, $y, \mathbf{y} \in \mathbb{R}$, $v, z, \mathbf{z} \in \mathbb{R}^d$ that $g(x) = \exp(-\frac{1}{2}\|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \epsilon v$, and $f(t, x, \mathbf{x}, y, \mathbf{y}) = y - y^3 - (\mathbf{y} - \mathbf{y}^3)$, and assume that for every $x \in \mathbb{R}^d$, $\theta \in \Theta$, $t \in [0, T]$, $s \in [t, T]$ it holds \mathbb{P} -a.s. that

$$X_{t,s}^{x,\theta} = R \left(x, x + \int_t^s \mu(X_{t,r}^{x,\theta}) dr + \int_t^s \sigma(X_{t,r}^{x,\theta}) dW_r^\theta \right). \quad (3.37)$$

The solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.27) then satisfies that for every $t \in [0, T]$, $x \in \partial_{\mathbb{D}}$ it holds that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$ and that for every $t \in [0, T]$, $x \in \mathbb{D}$ it holds that $u(0, x) = \exp(-\frac{1}{2}\|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + u(t, x) - [u(t, x)]^3 - \int_{[0,1]^d} (u(t, \mathbf{x}) - [u(t, \mathbf{x})]^3) d\mathbf{x}. \quad (3.38)$$

3.5 Numerical simulations

In this section we illustrate the performance of the machine learning-based approximation method in Lemma 3.3.1 by means of numerical simulations for 5 concrete non-local nonlinear PDEs; see Sections 3.5.1 to 3.5.5 below. In each of these numerical simulations

we employ the general machine learning-based approximation method in Lemma 3.3.1 together with the Adam optimizer (cf. (3.42) and (3.43) in Lemma 3.5.1 below and Kingma & Ba [Kingma2014]).

In each of the numerical simulations in Sections 3.5.1 to 3.5.5 we employ N fully-connected feedforward neural networks to represent $\mathbb{V}_n^{j,s}(\theta, x)$ for $n \in \{1, \dots, N\}$, $j \in \{1, \dots, 8000\}$, $s \in \mathbb{R}^c$, $\theta \in \mathbb{R}^\nu$, $x \in \mathbb{R}^d$. These neural networks consist of 4 layers (corresponding to 3 affine linear transformations in the neural networks) where the input layer is d -dimensional (with d neurons on the input layer), where the two hidden layers are both $(d + 50)$ -dimensional (with $d + 50$ neurons on each of the two hidden layers), and where the output layer is 1-dimensional (with 1 neuron on the output layer). We also refer to Figure 3.1 for a graphical illustration of the neural network architectures used in the numerical simulations in Sections 3.5.1 to 3.5.5.

As activation functions just in front of the two hidden layers we employ multidimensional versions of the hyperbolic tangent function

$$\mathbb{R} \ni x \mapsto (e^x + e^{-x})^{-1}(e^x - e^{-x}) \in \mathbb{R}. \quad (3.39)$$

In addition, in Sections 3.5.1, 3.5.2, 3.5.4 and 3.5.5 we use the square function $\mathbb{R} \ni x \mapsto x^2 \in \mathbb{R}$ as activation function just in front of the output layer and in Section 3.5.3 we use the identity function $\mathbb{R} \ni x \mapsto x \in \mathbb{R}$ as activation function just in front of the output layer. Furthermore, we employ Xavier initialisation to initialize all neural network parameters; see Glorot & Bengio [glorot2010] for details.

Each of the numerical experiments presented below is performed with the Julia library HighDimPDE.jl on a NVIDIA TITAN RTX GPU with 1350 MHz core clock and 24 GB GDDR6 memory with 7000 MHz clock rate where the underlying system consists of an AMG EPYC 7742 64-core CPU with 2TB memory running Julia 1.6.1 on Ubuntu 20.04.3. We also refer to ?? below for the employed Julia source codes.

Framework 3.5.1. Assume Lemma 3.3.1, let $\langle \cdot, \cdot \rangle: (\cup_{n \in \mathbb{N}}(\mathbb{R}^n \times \mathbb{R}^n)) \rightarrow \mathbb{R}$ and $\|\cdot\|: (\cup_{n \in \mathbb{N}}\mathbb{R}^n) \rightarrow \mathbb{R}$ satisfy for every $x = (x_1, \dots, x_d)$, $y = (y_1, \dots, y_d) \in \mathbb{R}^d$ that

$$\langle x, y \rangle = \sum_{j=1}^d x_j y_j \quad \text{and} \quad \|x\| = [\sum_{j=1}^d |x_j|^2]^{1/2}, \quad (3.40)$$

let $\nu = (d + 50)(d + 1) + (d + 50)(d + 51) + (d + 51)$, $\varepsilon = 10^{-8}$, $\beta_1 = \frac{9}{10}$, $\beta_2 = \frac{999}{1000}$, $(\gamma_m)_{m \in \mathbb{N}_0} \subseteq (0, \infty)$, let $g: \mathbb{R}^d \rightarrow \mathbb{R}$, $\mu: \mathbb{R}^d \rightarrow \mathbb{R}^d$, and $\sigma: \mathbb{R}^d \rightarrow \mathbb{R}^{d \times d}$ be continuous, let $u = (u(t, x))_{(t,x) \in [0,T] \times \mathbb{D}} \in C^{1,2}([0, T] \times \mathbb{D}, \mathbb{R})$ have at most polynomially growing partial derivatives, assume for every $t \in [0, T]$, $x \in \partial \mathbb{D}$ that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$, assume for every $t \in [0, T]$, $x \in \mathbb{D}$, $j \in \mathbb{N}$, $s \in \mathbb{R}^c$ that $u(0, x) = g(x) = V_0^{j,s}(\theta, x)$, $\int_{\mathbb{D}} |f(t, x, \mathbf{x}, u(t, x), u(t, \mathbf{x}), (\nabla_x u)(t, x), (\nabla_x u)(t, \mathbf{x}))| \nu_x(d\mathbf{x}) < \infty$, and

$$\begin{aligned} (\frac{\partial}{\partial t} u)(t, x) &= \int_{\mathbb{D}} f(t, x, \mathbf{x}, u(t, x), u(t, \mathbf{x}), (\nabla_x u)(t, x), (\nabla_x u)(t, \mathbf{x})) \nu_x(d\mathbf{x}) \\ &\quad + \langle \mu(x), (\nabla_x u)(t, x) \rangle + \frac{1}{2} \text{Trace}(\sigma(x)[\sigma(x)]^*(\text{Hess}_x u)(t, x)), \end{aligned} \quad (3.41)$$

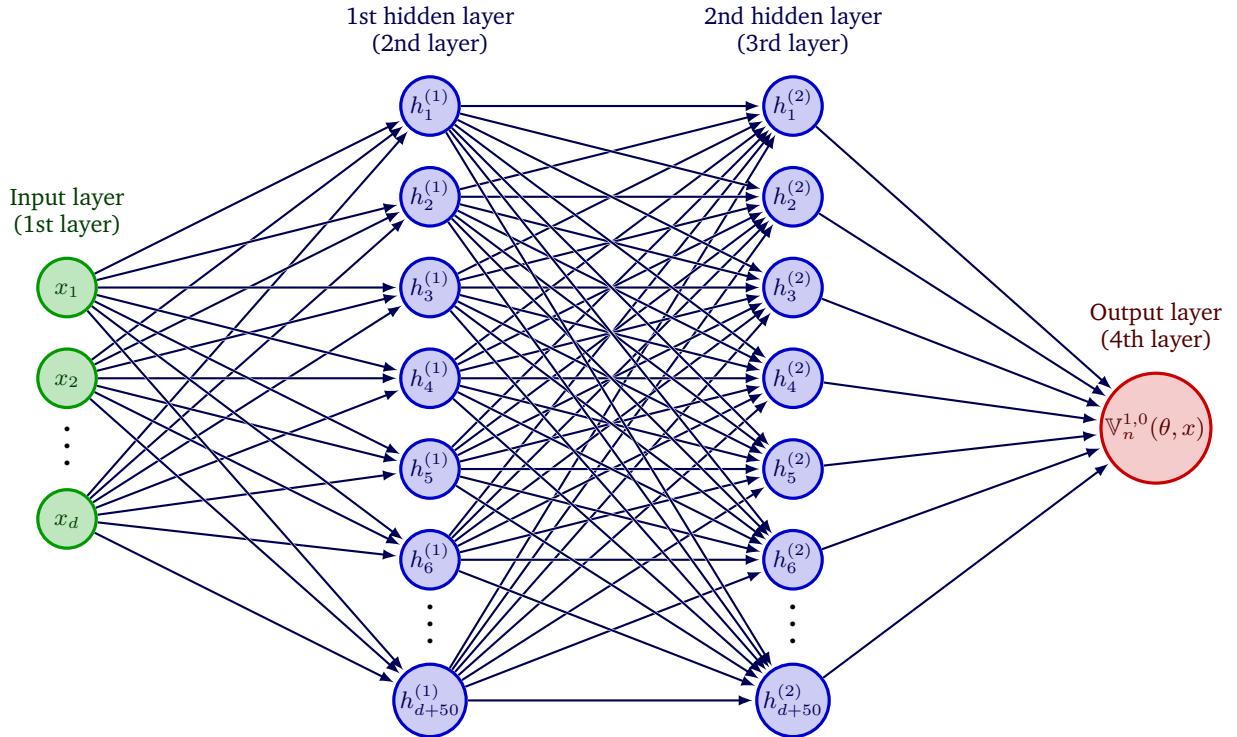


Fig. 3.1.: Graphical illustration of the neural network architectures used in the numerical simulations: in Sections 3.5.1 to 3.5.5 we employ neural networks with 4 layers (corresponding to 3 affine linear transformations in the neural networks) with d neurons on the input layer (corresponding to a d -dimensional input layer), with $d + 50$ neurons on the 1st hidden layer (corresponding to a $(d + 50)$ -dimensional 1st hidden layer), with $d + 50$ neurons on the 2nd hidden layer (corresponding to a $(d + 50)$ -dimensional 2nd hidden layer), and with 1 neuron on the output layer (corresponding to an 1-dimensional output layer) in the numerical simulations.

assume for every $m \in \mathbb{N}$, $i \in \{0, 1, \dots, N\}$ that $J_m = 8000$, $t_i = \frac{iT}{N}$, and $\varrho = 2\nu$, and assume for every $n \in \{1, 2, \dots, N\}$, $m \in \mathbb{N}$, $x = (x_1, \dots, x_\nu)$, $y = (y_1, \dots, y_\nu)$, $\eta = (\eta_1, \dots, \eta_\nu) \in \mathbb{R}^\nu$ that

$$\Psi_m^n(x, y, \eta) = (\beta_1 x + (1 - \beta_1)\eta, \beta_2 y + (1 - \beta_2)((\eta_1)^2, \dots, (\eta_\nu)^2)) \quad (3.42)$$

and

$$\psi_m^n(x, y) = \left(\left[\sqrt{\frac{|y_1|}{1 - (\beta_2)^m}} + \varepsilon \right]^{-1} \frac{\gamma_m x_1}{1 - (\beta_1)^m}, \dots, \left[\sqrt{\frac{|y_\nu|}{1 - (\beta_2)^m}} + \varepsilon \right]^{-1} \frac{\gamma_m x_\nu}{1 - (\beta_1)^m} \right). \quad (3.43)$$

3.5.1 Fisher–KPP PDEs with Neumann boundary conditions

In this subsection we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the solutions of certain Fisher–KPP PDEs with Neumann boundary conditions (cf., e.g., Bian et al. [Bian2017] and Wang et al. [Wang2021]).

Assume Lemma 3.5.1, let $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $D = [-\frac{1}{2}, \frac{1}{2}]^d$, $T \in \{1/5, 1/2, 1\}$, $N = 10$, $K_1 = K_2 = \dots = K_N = 1$, and $M_1 = M_2 = \dots = M_N = 400$, assume for every $n, m, j \in \mathbb{N}$, $\omega \in \Omega$ that $\xi^{n,m,j}(\omega) = (0, \dots, 0)$, assume for every $m \in \mathbb{N}$ that $\gamma_m = 10^{-2}$, and assume for every $s, t \in [0, T]$, $x, \mathbf{x} \in \mathbb{D}$, $y, \mathbf{y} \in \mathbb{R}$, $v, z, \mathbf{z} \in \mathbb{R}^d$ that $g(x) = \exp(-\frac{1}{2}\|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \epsilon v$, $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = y(1 - y)$, and

$$H(t, s, x, v) = R(x, x + \mu(x)(t - s) + \sigma(x)v) \quad (3.44)$$

(cf. (3.6) and (3.16)). The solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.41) then satisfies that for every $t \in [0, T]$, $x \in \partial_{\mathbb{D}}$ it holds that $\langle \mathbf{n}(x), (\nabla_x u)(t, x) \rangle = 0$ and that for every $t \in [0, T]$, $x \in \mathbb{D}$ it holds that $u(0, x) = \exp(-\frac{1}{2}\|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + u(t, x)(1 - u(t, x)). \quad (3.45)$$

In (3.45) the function u models the proportion of a particular type of alleles in a biological population spatially structured over D . The number $u(x, t) \in \mathbb{R}$ describes the proportion of individuals with a particular type of alleles located at position $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ at time $t \in [0, T]$. In Table 3.2 we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the mean of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the standard deviation of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the L^1 -approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the uncorrected sample standard deviation of the approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, and the average runtime in seconds needed for calculating one realization of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$ based on 5 independent realizations (5 independent runs). The reference value, which is used as an approximation for the unknown value $u(T, 0, 0, \dots, 0)$ of the exact solution of (3.45), has been calculated through the multilevel Picard approximation method for non-local nonlinear PDEs in Lemma 3.4.1 (see Lemma 3.4.2).

d	T	N	Mean of the approx. method	Standard deviation of the approx. method	Reference value	L^1 -approx. error	Standard deviation of the error	Average runtime in seconds
1	$1/5$	10	0.9986540	0.00005417	0.9986178	0.0000402	0.0000507	64.206
2	$1/5$	10	0.9971986	0.00012122	0.9972858	0.0001229	0.0000738	86.086
5	$1/5$	10	0.9933020	0.00013637	0.9931724	0.0001308	0.0001369	136.330
10	$1/5$	10	0.9862864	0.00014875	0.9863772	0.0001446	0.0000848	206.904
1	$1/2$	10	0.9989920	0.00006920	0.9990462	0.0000722	0.0000443	82.898
2	$1/2$	10	0.9979800	0.00012102	0.9980528	0.0001119	0.0000755	115.407
5	$1/2$	10	0.9948250	0.00022405	0.9951104	0.0002868	0.0002252	190.090
10	$1/2$	10	0.9897806	0.00043646	0.9901634	0.0004260	0.0003927	295.212
1	1	10	0.9994678	0.00010809	0.9994910	0.0000812	0.0000642	104.738
2	1	10	0.9988340	0.00012236	0.9990816	0.0002478	0.0001225	148.529
5	1	10	0.9967888	0.00044382	0.9971944	0.0004067	0.0004451	251.351
10	1	10	0.9942158	0.00008595	0.9951800	0.0009689	0.0000864	398.730

Tab. 3.1.: Numerical simulations for the approximation method in Lemma 3.3.1 in the case of the Fisher–KPP PDEs with Neumann boundary conditions in (3.45) in Section 3.5.1.

d	T	N	Mean of the approx. method	Standard deviation of the approx. method	Reference value	L^1 -approx. error	Standard deviation of the error	Average runtime in seconds
1	$1/5$	10	0.9986540	0.00005417	0.9986178	0.0000402	0.0000507	64.206
2	$1/5$	10	0.9971986	0.00012122	0.9972858	0.0001229	0.0000738	86.086
5	$1/5$	10	0.9933020	0.00013637	0.9931724	0.0001308	0.0001369	136.330
10	$1/5$	10	0.9862864	0.00014875	0.9863772	0.0001446	0.0000848	206.904
1	$1/2$	10	0.9989920	0.00006920	0.9990462	0.0000722	0.0000443	82.898
2	$1/2$	10	0.9979800	0.00012102	0.9980528	0.0001119	0.0000755	115.407
5	$1/2$	10	0.9948250	0.00022405	0.9951104	0.0002868	0.0002252	190.090
10	$1/2$	10	0.9897806	0.00043646	0.9901634	0.0004260	0.0003927	295.212
1	1	10	0.9994678	0.00010809	0.9994910	0.0000812	0.0000642	104.738
2	1	10	0.9988340	0.00012236	0.9990816	0.0002478	0.0001225	148.529
5	1	10	0.9967888	0.00044382	0.9971944	0.0004067	0.0004451	251.351
10	1	10	0.9942158	0.00008595	0.9951800	0.0009689	0.0000864	398.730

Tab. 3.2.: Numerical simulations for the approximation method in Lemma 3.3.1 in the case of the Fisher–KPP PDEs with Neumann boundary conditions in (3.45) in Section 3.5.1.

3.5.2 Non-local competition PDEs

In this subsection we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the solutions of certain non-local competition PDEs (cf., e.g., Doebeli & Ispolatov [Doebeli2010], Berestycki et al. [Berestycki2009b], Perthame & Génieys [Perthame2007], and Génieys et al. [Genieys2006a]).

Assume Lemma 3.5.1, let $\mathfrak{s} = \frac{1}{10}$, $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $\mathbb{D} = \mathbb{R}^d$, $T \in \{1/5, 1/2, 1\}$, $N = 10$, $K_1 = K_2 = \dots = K_N = 1$, and $M_1 = M_2 = \dots = M_N = 400$, assume for every $n, m, j \in \mathbb{N}$, $\omega \in \Omega$ that $\xi^{n,m,j}(\omega) = (0, \dots, 0)$, assume for every $m \in \mathbb{N}$ that $\gamma_m = 10^{-2}$, and assume for every $s, t \in [0, T]$, $v, x, \mathbf{x}, z, \mathbf{z} \in \mathbb{R}^d$, $y, \mathbf{y} \in \mathbb{R}$, $A \in \mathcal{B}(\mathbb{R}^d)$ that $\nu_x(A) = \frac{1}{(\pi)^{d/2} \mathfrak{s}^d} \int_A \exp(-\mathfrak{s}^{-2} \|x - \mathbf{x}\|^2) d\mathbf{x}$, $g(x) = \exp(-\frac{1}{2} \|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \epsilon v$, $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = y(1 - \mathbf{y} \mathfrak{s}^d \pi^{d/2})$, and

$$H(t, s, x, v) = x + \mu(x)(t - s) + \sigma(x)v \quad (3.46)$$

(cf. (3.6) and (3.16)). The solution $u: [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}$ of the PDE in (3.41) then satisfies that for every $t \in [0, T]$, $x \in \mathbb{R}^d$ it holds that $u(0, x) = \exp(-\frac{1}{2} \|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + u(t, x) \left(1 - \int_{\mathbb{R}^d} u(t, \mathbf{x}) \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}\|^2}{\mathfrak{s}^2}\right) d\mathbf{x} \right). \quad (3.47)$$

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + u(t, x) \left(1 - \int_{\mathbb{R}^d} u(t, \mathbf{x}) \exp\left(-\frac{\|\mathbf{x}-\mathbf{x}\|^2}{\mathfrak{s}^2}\right) d\mathbf{x} \right). \quad (3.48)$$

In (3.48) the function u models the evolution of a population characterized by a set of d biological traits under the combined effects of selection, competition and mutation. The number $u(x, t) \in \mathbb{R}$ describes the number of individuals with traits $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ at time $t \in [0, T]$. In Table 3.3 we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the mean of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the standard deviation of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the L^1 -approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the uncorrected sample standard deviation of the approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, and the average runtime in seconds needed for calculating one realization of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$ based on 5 independent realizations (5 independent runs). The reference value, which is used as an approximation for the unknown value $u(T, 0, 0, \dots, 0)$ of the exact solution of (3.48), has been calculated through the multilevel Picard approximation method for non-local nonlinear PDEs in Lemma 3.4.1 (see Lemma 3.4.3).

3.5.3 Non-local sine-Gordon type PDEs

In this subsection we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the solutions of non-local sine-Gordon type PDEs (cf., e.g., Hairer & Shen [Hairer2016], Barone et al. [Barone1971], and Coleman [Coleman1994]).

d	T	N	Mean	Std. dev.	Ref. value	L^1 -error	Std. dev. error	avg. runtime (s)
1	$\frac{1}{5}$	10	1.1730854	0.00013316	1.1689300	0.0035549	0.0001139	12.473
2	$\frac{1}{5}$	10	1.2084412	0.00005216	1.2039100	0.0037637	0.0000433	11.820
5	$\frac{1}{5}$	10	1.2128370	0.00007562	1.2074180	0.0044881	0.0000626	11.549
10	$\frac{1}{5}$	10	1.2068802	0.00005007	1.2017500	0.0042689	0.0000417	11.833
1	$\frac{1}{2}$	10	1.4680348	0.00022779	1.4574020	0.0072957	0.0001563	11.675
2	$\frac{1}{2}$	10	1.5909570	0.00016869	1.5809500	0.0063297	0.0001067	11.627
5	$\frac{1}{2}$	10	1.6085470	0.00007118	1.5931920	0.0096379	0.0000447	11.706
10	$\frac{1}{2}$	10	1.5886930	0.00022610	1.5751840	0.0085761	0.0001435	11.599
1	1	10	2.0458830	0.00039746	2.0308460	0.0074043	0.0001957	11.486
2	1	10	2.4577706	0.00005648	2.4563720	0.0005694	0.0000230	11.607
5	1	10	2.5299212	0.00022210	2.5474100	0.0068653	0.0000872	11.605
10	1	10	2.4684364	0.00014073	2.4673420	0.0004436	0.0000570	11.689

Tab. 3.3.: Numerical simulations for the approximation method in Lemma 3.3.1 in the case of the non-local competition PDEs in (3.48) in Section 3.5.2.

Assume Lemma 3.5.1, let $s = \frac{1}{10}$, $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $\mathbb{D} = \mathbb{R}^d$, $T \in \{\frac{1}{5}, \frac{1}{2}, 1\}$, $N = 10$, $K_1 = K_2 = \dots = K_N = 1$, and $M_1 = M_2 = \dots = M_N = 400$, assume for every $n, m, j \in \mathbb{N}$, $\omega \in \Omega$ that $\xi^{n,m,j}(\omega) = (0, \dots, 0)$, assume for every $m \in \mathbb{N}$ that $\gamma_m = 10^{-3}$, and assume for every $s, t \in [0, T]$, $v, x, \mathbf{x}, z, \mathbf{z} \in \mathbb{R}^d$, $y, \mathbf{y} \in \mathbb{R}$, $A \in \mathcal{B}(\mathbb{R}^d)$ that $\nu_x(A) = \frac{1}{(\pi)^{d/2} s^d} \int_A \exp(-s^{-2} \|x - \mathbf{x}\|^2) d\mathbf{x}$, $g(x) = \exp(-\frac{1}{2} \|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \epsilon v$, $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = \sin(y) - \mathbf{y} \pi^{d/2} s^d$, and

$$H(t, s, x, v) = x + \mu(x)(t - s) + \sigma(x)v \quad (3.49)$$

(cf. (3.6) and (3.16)). The solution $u: [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}$ of the PDE in (3.41) then satisfies that for every $t \in [0, T]$, $x \in \mathbb{R}^d$ it holds that $u(0, x) = \exp(-\frac{1}{2} \|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + \sin(u(t, x)) - \int_{\mathbb{R}^d} u(t, \mathbf{x}) e^{-\frac{\|x-\mathbf{x}\|^2}{s^2}} d\mathbf{x}. \quad (3.50)$$

In Table 3.4 we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the mean of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the standard deviation of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the L^1 -approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the uncorrected sample standard deviation of the approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, and the average runtime in seconds needed for calculating one realization of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$ based on 5 independent realizations (5 independent runs). The reference value, which is used as an approximation for the unknown value $u(T, 0, 0, \dots, 0)$ of the exact solution of (3.50), has been calculated through the multilevel Picard approximation method for non-local nonlinear PDEs in Lemma 3.4.1 (see Lemma 3.4.4).

3.5.4 Replicator-mutator PDEs

In this subsection we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the solutions of certain replicator-mutator PDEs describing the

d	T	N	Mean	Std. dev.	Ref. value	L^1 -error	Std. dev. error	avg. runtime
1	$\frac{1}{5}$	10	1.1357852	0.00004031	1.1329060	0.0025414	0.0000356	11.403
2	$\frac{1}{5}$	10	1.1667370	0.00011471	1.1629940	0.0032184	0.0000986	11.351
5	$\frac{1}{5}$	10	1.1704630	0.00005095	1.1664700	0.0034231	0.0000437	11.137
10	$\frac{1}{5}$	10	1.1650314	0.00004317	1.1609480	0.0035173	0.0000372	11.179
1	$\frac{1}{2}$	10	1.3500632	0.00009184	1.3409300	0.0068111	0.0000685	11.118
2	$\frac{1}{2}$	10	1.4364596	0.00004400	1.4269380	0.0066727	0.0000308	11.215
5	$\frac{1}{2}$	10	1.4472808	0.00006241	1.4338880	0.0093402	0.0000435	11.199
10	$\frac{1}{2}$	10	1.4325428	0.00006902	1.4209860	0.0081329	0.0000486	11.174
1	1	10	1.7087326	0.00020741	1.6880340	0.0122620	0.0001229	11.070
2	1	10	1.8962484	0.00006563	1.8653440	0.0165677	0.0000352	11.076
5	1	10	1.9221128	0.00008683	1.8889600	0.0175508	0.0000460	11.082
10	1	10	1.8934832	0.00009063	1.8588340	0.0186403	0.0000488	11.148

Tab. 3.4.: Numerical simulations for the approximation method in Lemma 3.3.1 in the case of the non-local sine-Gordon PDEs in (3.50) in Section 3.5.3.

dynamics of a phenotype distribution $u(t, x)$ under the combined effects of selection and mutation (cf., e.g., Hamel et al. [Hamel2020]).

Assume Lemma 3.5.1, let $\mathfrak{m}_1, \mathfrak{m}_2, \dots, \mathfrak{m}_d, \mathfrak{s}_1, \mathfrak{s}_2, \dots, \mathfrak{s}_d, \mathfrak{u}_1, \mathfrak{u}_2, \dots, \mathfrak{u}_d \in \mathbb{R}$ satisfy for every $k \in \{1, 2, \dots, d\}$ that $\mathfrak{m}_k = \frac{1}{10}$, $\mathfrak{s}_k = \frac{1}{4\pi}$, and $\mathfrak{u}_k = 0$, assume that $d \in \{1, 2, 5, 10\}$, $\mathbb{D} = \mathbb{R}^d$, $T \in \{1/5, 1/2, 1\}$, $K_1 = K_2 = \dots = K_N = 1$, and $M_1 = M_2 = \dots = M_N = 8000$, assume that $\xi^{n,m,j}, n, m, j \in \mathbb{N}$ are independent $\mathcal{U}_{[-1/4, 1/4]^d}$ -distributed random variables, assume for every $m \in \mathbb{N}$ that $\gamma_m = 10^{-3}$, let $a: \mathbb{R}^d \rightarrow \mathbb{R}$ satisfy for every $x \in \mathbb{R}^d$ that $a(x) = -\frac{1}{2}\|x\|^2$, and assume for every $s, t \in [0, T]$, $x = (x_1, \dots, x_d)$, $\mathbf{x} \in D$, $v = (v_1, \dots, v_d)$, $z, \mathbf{z} \in \mathbb{R}^d$, $y, \mathbf{y} \in \mathbb{R}$ $A \in \mathcal{B}(\mathbb{R}^d)$ that $\nu_x(A) = 2^d \int_A dx$, $g(x) = (2\pi)^{-d/2} [\prod_{i=1}^d |\mathfrak{s}_i|^{-1/2}] \exp(-\sum_{i=1}^d \frac{(x_i - \mathfrak{u}_i)^2}{2\mathfrak{s}_i})$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = (\mathfrak{m}_1 v_1, \dots, \mathfrak{m}_d v_d)$, $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = y(a(x) - \mathbf{y}a(\mathbf{x}))$, and

$$H(t, s, x, v) = x + \mu(x)(t - s) + \sigma(x)v \quad (3.51)$$

(cf. (3.6) and (3.16)). The solution $u: [0, T] \times \mathbb{D} \rightarrow \mathbb{R}$ of the PDE in (3.41) then satisfies that for every $t \in [0, T]$, $x = (x_1, \dots, x_d) \in \mathbb{D}$ it holds that $u(0, x) = (2\pi)^{-d/2} [\prod_{i=1}^d |\mathfrak{s}_i|^{-1/2}] \exp(-\sum_{i=1}^d \frac{(x_i - \mathfrak{u}_i)^2}{2\mathfrak{s}_i})$ and

$$(\frac{\partial}{\partial t} u)(t, x) = u(t, x) \left(a(x) - \int_D u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) + \sum_{i=1}^d \frac{1}{2} |\mathfrak{m}_i|^2 (\frac{\partial^2}{\partial x_i^2} u)(t, x). \quad (3.52)$$

$$(\frac{\partial}{\partial t} u)(t, x) = u(t, x) \left(a(x) - \int_D u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) + \sum_{i=1}^d \frac{1}{2} |\mathfrak{m}_i|^2 (\frac{\partial^2}{\partial x_i^2} u)(t, x). \quad (3.53)$$

$$(\frac{\partial}{\partial t} u)(t, x) = u(t, x) \left(a(x) - \int_D u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) + \sum_{i=1}^d \frac{1}{2} |\mathfrak{m}_i|^2 (\frac{\partial^2}{\partial x_i^2} u)(t, x). \quad (3.54)$$

In (3.54) the function u models the evolution of the phenotype distribution of a population composed of a set of d biological traits under the combined effects of selection and mutation. The number $u(x, t) \in \mathbb{R}$ describes the number of individuals with traits $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ at time $t \in [0, T]$. The function a models a quadratic Malthusian fitness function. In

d	T	N	Mean	Std. dev.	Ref. value	L^1 -error	Std. dev. error	avg. runtime (s)
1	$1/5$	10	1.4199752	0.01310404	1.4078422	0.0099902	0.0073974	78.462
2	$1/5$	10	2.0293278	0.01901140	1.9820198	0.0238686	0.0095919	78.887
5	$1/5$	10	5.9160758	0.21986155	5.5305710	0.0697043	0.0397539	78.492
10	$1/5$	10	37.0632950	5.28388539	30.5872152	0.2117251	0.1727482	79.046
1	$1/2$	10	1.4336860	0.03447971	1.3988110	0.0249319	0.0246493	77.868
2	$1/2$	10	2.0762260	0.07197504	1.9566722	0.0611006	0.0367844	78.332
5	$1/2$	10	6.2846914	0.38290314	5.3554406	0.1735153	0.0714980	79.065
10	$1/2$	10	49.0773484	15.07067499	28.6807436	0.7111603	0.5254632	79.804
1	1	10	1.4309828	0.02890686	1.3850483	0.0331645	0.0208707	77.757
2	1	10	2.0891042	0.09289536	1.9183588	0.0890060	0.0484244	78.636
5	1	10	6.4315024	0.25624012	5.0971171	0.2617922	0.0502716	78.983
10	1	10	43.2048600	5.00051040	25.9806024	0.6629661	0.1924709	79.459

Tab. 3.5.: Numerical simulations for the approximation method in Lemma 3.3.1 in the case of the replicator-mutator PDEs in (3.54) in Section 3.5.4.

Figure 3.2 we use the machine learning-based approximation method in Lemma 3.5.1 to approximately plot $[-1/4, 1/4] \ni x \mapsto \mathbb{V}_n^{1,0}(\theta, (x, 0, 0, \dots, 0)) \in \mathbb{R}$ as an approximation of $[-1/4, 1/4] \ni x \mapsto u(t_n, (x, 0, 0, \dots, 0)) \in \mathbb{R}$ with $d = 5, n \in \{0, 1, 2, 3\}, N = 10, T = 1/2$. In Table 3.5 we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the mean of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the standard deviation of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the L^1 -approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the uncorrected sample standard deviation of the approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, and the average runtime in seconds needed for calculating one realization of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$ based on 5 independent realizations (5 independent runs). The value $u(T, 0, 0, \dots, 0)$ of the exact solution of (3.54) has been calculated by means of Lemma 3.5.2 below.

Lemma 3.5.2. Let $d \in \mathbb{N}, u_1, u_2, \dots, u_d \in \mathbb{R}, m_1, m_2, \dots, m_d, s_1, s_2, \dots, s_d \in (0, \infty)$, let $a: \mathbb{R}^d \rightarrow \mathbb{R}$ satisfy for every $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ that $a(x) = -\frac{1}{2} \sum_{i=1}^d |x_i|^2$, for every $i \in \{1, 2, \dots, d\}$ let $\mathfrak{S}_i: [0, \infty) \rightarrow (0, \infty)$ and $\mathfrak{U}_i: [0, \infty) \rightarrow \mathbb{R}$ satisfy for every $t \in [0, \infty)$ that

$$\mathfrak{S}_i(t) = m_i \left[\frac{m_i \sinh(m_i t) + s_i \cosh(m_i t)}{m_i \cosh(m_i t) + s_i \sinh(m_i t)} \right] \quad \text{and} \quad \mathfrak{U}_i(t) = \frac{m_i u_i}{m_i \cosh(m_i t) + s_i \sinh(m_i t)}, \quad (3.55)$$

and let $u: [0, \infty) \times \mathbb{R}^d \rightarrow \mathbb{R}$ satisfy for every $t \in [0, \infty), x = (x_1, \dots, x_d) \in \mathbb{R}^d$ that

$$u(t, x) = (2\pi)^{-d/2} \left[\prod_{i=1}^d |\mathfrak{S}_i(t)|^{-1/2} \right] \exp \left(- \sum_{i=1}^d \frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)} \right). \quad (3.56)$$

Then

(i) it holds that $u \in C^{1,2}([0, \infty) \times \mathbb{R}^d, \mathbb{R})$,

(ii) it holds for every $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ that $u(0, x) = (2\pi)^{-d/2} [\prod_{i=1}^d |\mathfrak{S}_i|^{-1/2}] \exp(-\sum_{i=1}^d \frac{(x_i - u_i)^2}{2s_i})$, and

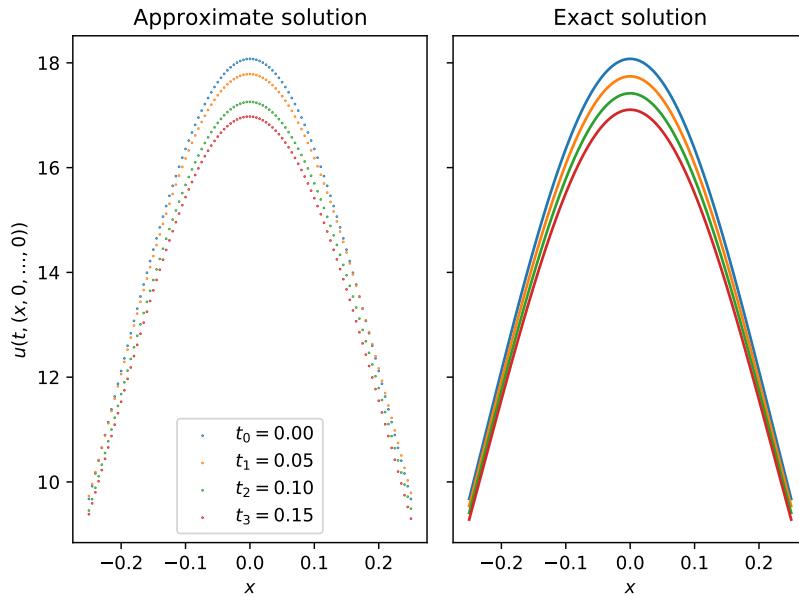


Fig. 3.2.: Approximate plot of $[-1/4, 1/4] \ni x \rightarrow \mathbb{V}_n^{1,0}(\theta, (x, 0, 0, \dots, 0)) \in \mathbb{R}$ as an approximation of $[-1/4, 1/4] \ni x \rightarrow u(t_n, (x, 0, 0, \dots, 0)) \in \mathbb{R}$ in the case of the replicator-mutator PDEs in (3.54) in Section 3.5.4 with $d = 5$, $n \in \{0, 1, 2, 3\}$, $N = 10$, $T = 1/2$. On the left plot we approximately represent the machine learning approximation $[-1/4, 1/4] \ni x \rightarrow \mathbb{V}_n^{1,0}(\theta, (x, 0, 0, \dots, 0)) \in \mathbb{R}$ and on the right plot we approximately plot the exact solution $[-1/4, 1/4] \ni x \rightarrow u(t_n, (x, 0, 0, \dots, 0)) \in \mathbb{R}$ in (3.56) in the case of the replicator-mutator PDEs in (3.54).

(iii) it holds for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ that

$$(\frac{\partial}{\partial t} u)(t, x) = u(t, x) \left(a(x) - \int_{\mathbb{R}^d} u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) + \sum_{i=1}^d \frac{1}{2} (\mathfrak{m}_i)^2 (\frac{\partial^2}{\partial x_i^2} u)(t, x). \quad (3.57)$$

Proof of Lemma 3.5.2. First, note that the fact for every $i \in \{1, 2, \dots, d\}$ it holds that $\mathfrak{S}_i \in C^\infty([0, \infty), (0, \infty))$, the fact for every $i \in \{1, 2, \dots, d\}$ it holds that $\mathfrak{U}_i \in C^\infty([0, \infty), \mathbb{R})$, and (3.56) establish item ((i)). Moreover, observe that the fact for every $i \in \{1, 2, \dots, d\}$ it holds that $\mathfrak{S}_i(0) = \mathfrak{s}_i$, the fact for every $i \in \{1, 2, \dots, d\}$ it holds that $\mathfrak{U}_i(0) = \mathfrak{u}_i$, and (3.55) establish item ((ii)). Next note that (3.56) ensures that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$u(t, x) = \prod_{i=1}^d \left[(2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)}\right) \right]. \quad (3.58)$$

The product rule hence implies that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\begin{aligned} & (\frac{\partial}{\partial t} u)(t, x) \\ &= \frac{\partial}{\partial t} \left(\prod_{i=1}^d \left[(2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)}\right) \right] \right) \\ &= \sum_{i=1}^d \left[\left[\prod_{j \in \{1, \dots, d\} \setminus i} \left((2\pi \mathfrak{S}_j(t))^{-1/2} \exp\left(-\frac{(x_j - \mathfrak{U}_j(t))^2}{2\mathfrak{S}_j(t)}\right) \right) \right] \right. \\ &\quad \cdot \left. \left[\frac{\partial}{\partial t} \left((2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)}\right) \right) \right] \right]. \end{aligned} \quad (3.59)$$

The chain rule and (3.58) hence ensure that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\begin{aligned}
& (\frac{\partial}{\partial t} u)(t, x) \\
&= \sum_{i=1}^d \left[\left[\prod_{j \in \{1, \dots, d\} \setminus i} \left((2\pi \mathfrak{S}_j(t))^{-1/2} \exp \left(-\frac{(x_j - \mathfrak{U}_j(t))^2}{2\mathfrak{S}_j(t)} \right) \right) \right] \right. \\
&\quad \cdot \left[\left(\frac{\partial}{\partial t} \left((2\pi \mathfrak{S}_i(t))^{-1/2} \right) \right) \left(\exp \left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)} \right) \right) \right. \\
&\quad \left. + \left((2\pi \mathfrak{S}_i(t))^{-1/2} \right) \left(\frac{\partial}{\partial t} \left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)} \right) \right) \left(\exp \left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)} \right) \right) \right] \left. \right] \\
&= \sum_{i=1}^d \left[\left[\prod_{j \in \{1, \dots, d\} \setminus i} \left((2\pi \mathfrak{S}_j(t))^{-1/2} \exp \left(-\frac{(x_j - \mathfrak{U}_j(t))^2}{2\mathfrak{S}_j(t)} \right) \right) \right] \right. \\
&\quad \cdot \left[- \left((2\pi \mathfrak{S}_i(t))^{-1/2} \right) \frac{(\frac{\partial}{\partial t} \mathfrak{S}_i)(t)}{2\mathfrak{S}_i(t)} \exp \left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)} \right) \right. \\
&\quad + \left((2\pi \mathfrak{S}_i(t))^{-1/2} \right) \left(\frac{2(\frac{\partial}{\partial t} \mathfrak{U}_i)(t)(x_i - \mathfrak{U}_i(t))}{2\mathfrak{S}_i(t)} \right. \\
&\quad \left. \left. + \frac{(x_i - \mathfrak{U}_i(t))^2(\frac{\partial}{\partial t} \mathfrak{S}_i)(t)}{2|\mathfrak{S}_i(t)|^2} \right) \exp \left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)} \right) \right] \left. \right] \\
&= u(t, x) \left[\sum_{i=1}^d \left(\frac{-(\frac{\partial}{\partial t} \mathfrak{S}_i)(t)}{2\mathfrak{S}_i(t)} + \frac{2\mathfrak{S}_i(t)(\frac{\partial}{\partial t} \mathfrak{U}_i)(t)(x_i - \mathfrak{U}_i(t)) + (x_i - \mathfrak{U}_i(t))^2(\frac{\partial}{\partial t} \mathfrak{S}_i)(t)}{2|\mathfrak{S}_i(t)|^2} \right) \right]. \tag{3.60}
\end{aligned}$$

Moreover, observe that (3.55), the chain rule, and the product rule ensure that for every $i \in \{1, \dots, d\}$, $t \in [0, \infty)$ it holds that

$$\begin{aligned}
(\frac{\partial}{\partial t} \mathfrak{U}_i)(t) &= \frac{\partial}{\partial t} \left(\frac{\mathfrak{m}_i \mathfrak{u}_i}{\mathfrak{m}_i \cosh(\mathfrak{m}_i t) + \mathfrak{s}_i \sinh(\mathfrak{m}_i t)} \right) \\
&= -|\mathfrak{m}_i|^2 \mathfrak{u}_i \left[\frac{\mathfrak{m}_i \sinh(\mathfrak{m}_i t) + \mathfrak{s}_i \cosh(\mathfrak{m}_i t)}{[\mathfrak{m}_i \cosh(\mathfrak{m}_i t) + \mathfrak{s}_i \sinh(\mathfrak{m}_i t)]^2} \right] \\
&= -\mathfrak{S}_i(t) \mathfrak{U}_i(t)
\end{aligned} \tag{3.61}$$

and

$$\begin{aligned}
(\frac{\partial}{\partial t} \mathfrak{S}_i)(t) &= \frac{\partial}{\partial t} \left(\mathfrak{m}_i \left[\frac{\mathfrak{m}_i \sinh(\mathfrak{m}_i t) + \mathfrak{s}_i \cosh(\mathfrak{m}_i t)}{\mathfrak{m}_i \cosh(\mathfrak{m}_i t) + \mathfrak{s}_i \sinh(\mathfrak{m}_i t)} \right] \right) \\
&= |\mathfrak{m}_i|^2 \left[\frac{\mathfrak{m}_i \cosh(\mathfrak{m}_i t) + \mathfrak{s}_i \sinh(\mathfrak{m}_i t)}{\mathfrak{m}_i \cosh(\mathfrak{m}_i t) + \mathfrak{s}_i \sinh(\mathfrak{m}_i t)} \right] - |\mathfrak{m}_i|^2 \left[\frac{\mathfrak{m}_i \sinh(\mathfrak{m}_i t) + \mathfrak{s}_i \cosh(\mathfrak{m}_i t)}{\mathfrak{m}_i \cosh(\mathfrak{m}_i t) + \mathfrak{s}_i \sinh(\mathfrak{m}_i t)} \right]^2 \\
&= |\mathfrak{m}_i|^2 - |\mathfrak{S}_i(t)|^2.
\end{aligned} \tag{3.62}$$

Combining this with (3.60) yields that for every $i \in \{1, \dots, d\}$, $t \in [0, \infty)$ it holds that

$$\begin{aligned}
& (\frac{\partial}{\partial t} u)(t, x) \\
&= \frac{u(t, x)}{2} \left[\sum_{i=1}^d \left[\frac{-[|\mathfrak{m}_i|^2 - |\mathfrak{S}_i(t)|^2]}{\mathfrak{S}_i(t)} \right. \right. \\
&\quad \left. \left. + \frac{2|\mathfrak{S}_i(t)|^2 \mathfrak{U}_i(t) (\mathfrak{U}_i(t) - x_i) + (x_i - \mathfrak{U}_i(t))^2 (|\mathfrak{m}_i|^2 - |\mathfrak{S}_i(t)|^2)}{|\mathfrak{S}_i(t)|^2} \right] \right] \\
&= \frac{u(t, x)}{2} \left[\sum_{i=1}^d \left[|\mathfrak{m}_i|^2 \left(\left(\frac{x_i - \mathfrak{U}_i(t)}{\mathfrak{S}_i(t)} \right)^2 - \frac{1}{\mathfrak{S}_i(t)} \right) \right. \right. \\
&\quad \left. \left. + \mathfrak{S}_i(t) + 2(|\mathfrak{U}_i(t)|^2 - \mathfrak{U}_i(t) x_i) - (|x_i|^2 - 2\mathfrak{U}_i(t) x_i + |\mathfrak{U}_i(t)|^2) \right] \right] \\
&= \frac{u(t, x)}{2} \left[\sum_{i=1}^d \left[|\mathfrak{m}_i|^2 \left(\left(\frac{x_i - \mathfrak{U}_i(t)}{\mathfrak{S}_i(t)} \right)^2 - \frac{1}{\mathfrak{S}_i(t)} \right) + \mathfrak{S}_i(t) + |\mathfrak{U}_i(t)|^2 - |x_i|^2 \right] \right]. \tag{3.63}
\end{aligned}$$

Moreover, note that (3.58) and the product rule demonstrate that for every $i \in \{1, \dots, d\}$, $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\begin{aligned}
& (\frac{\partial}{\partial x_i} u)(t, x) \\
&= \frac{\partial}{\partial x_i} \left[\prod_{j=1}^d \left[(2\pi \mathfrak{S}_j(t))^{-1/2} \exp \left(-\frac{(x_j - \mathfrak{U}_j(t))^2}{2\mathfrak{S}_j(t)} \right) \right] \right] \\
&= \left[\frac{\partial}{\partial x_i} \left[(2\pi \mathfrak{S}_i(t))^{-1/2} \exp \left(-\frac{(x_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)} \right) \right] \right] \\
&\quad \cdot \left[\prod_{j \in \{1, \dots, d\} \setminus i} \left[(2\pi \mathfrak{S}_j(t))^{-1/2} \exp \left(-\frac{(x_j - \mathfrak{U}_j(t))^2}{2\mathfrak{S}_j(t)} \right) \right] \right] \\
&= -u(t, x) \left(\frac{x_i - \mathfrak{U}_i(t)}{\mathfrak{S}_i(t)} \right) = u(t, x) \left(\frac{\mathfrak{U}_i(t) - x_i}{\mathfrak{S}_i(t)} \right). \tag{3.64}
\end{aligned}$$

The product rule and (3.58) therefore assure that for every $i \in \{1, \dots, d\}$, $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\begin{aligned}
& (\frac{\partial^2}{\partial x_i^2} u)(t, x) = \frac{\partial}{\partial x_i} \left(u(t, x) \left(\frac{\mathfrak{U}_i(t) - x_i}{\mathfrak{S}_i(t)} \right) \right) \\
&= u(t, x) \left[\left(\frac{\mathfrak{U}_i(t) - x_i}{\mathfrak{S}_i(t)} \right)^2 - \frac{1}{\mathfrak{S}_i(t)} \right] = u(t, x) \left[\left(\frac{x_i - \mathfrak{U}_i(t)}{\mathfrak{S}_i(t)} \right)^2 - \frac{1}{\mathfrak{S}_i(t)} \right]. \tag{3.65}
\end{aligned}$$

Hence, we obtain that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\left[\sum_{i=1}^d \left[\left(\frac{1}{2} |\mathfrak{m}_i|^2 \right) (\frac{\partial^2}{\partial x_i^2} u)(t, x) \right] \right] = \frac{u(t, x)}{2} \left[\sum_{i=1}^d \left[|\mathfrak{m}_i|^2 \left(\left(\frac{x_i - \mathfrak{U}_i(t)}{\mathfrak{S}_i(t)} \right)^2 - \frac{1}{\mathfrak{S}_i(t)} \right) \right] \right]. \tag{3.66}$$

Next observe that (3.58) ensures that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\begin{aligned}
& u(t, x) \left(a(x) - \int_{\mathbb{R}^d} u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) \\
&= u(t, x) \left(\left[-\frac{1}{2} \right] \left[\sum_{i=1}^d |x_i|^2 \right] - \int_{\mathbb{R}^d} \left[-\frac{1}{2} \right] \left[\sum_{i=1}^d |\mathbf{x}_i|^2 \right] u(t, \mathbf{x}) d\mathbf{x} \right) \\
&= \frac{u(t, x)}{2} \left(- \left[\sum_{i=1}^d |x_i|^2 \right] \right. \\
&\quad \left. + \sum_{i=1}^d \left[\int_{\mathbb{R}} \left(|\mathbf{x}_i|^2 (2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{(\mathbf{x}_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)}\right) \right) d\mathbf{x}_i \right. \right. \\
&\quad \cdot \left. \left. \cdot \left(\prod_{j \in \{1, \dots, d\} \setminus i} \int_{\mathbb{R}} \left((2\pi \mathfrak{S}_j(t))^{-1/2} \exp\left(-\frac{(\mathbf{x}_j - \mathfrak{U}_j(t))^2}{2\mathfrak{S}_j(t)}\right) \right) d\mathbf{x}_j \right) \right] \right). \tag{3.67}
\end{aligned}$$

This and the fact for every $i \in \{1, \dots, d\}$, $t \in [0, \infty)$ it holds that

$$\int_{\mathbb{R}} \left((2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{(x - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)}\right) \right) dx = 1 \tag{3.68}$$

imply that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\begin{aligned}
& u(t, x) \left(a(x) - \int_{\mathbb{R}^d} u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) \\
&= \frac{u(t, x)}{2} \left(\sum_{i=1}^d \left[-|x_i|^2 + \int_{\mathbb{R}} \left(|\mathbf{x}_i|^2 (2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{(\mathbf{x}_i - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)}\right) \right) d\mathbf{x}_i \right] \right). \tag{3.69}
\end{aligned}$$

Next observe that the integral transformation theorem demonstrates that for every $i \in \{1, \dots, d\}$, $t \in [0, \infty)$ it holds that

$$\begin{aligned}
& \int_{\mathbb{R}} \left(x^2 \left[(2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{(x - \mathfrak{U}_i(t))^2}{2\mathfrak{S}_i(t)}\right) \right] \right) dx \\
&= \int_{\mathbb{R}} \left((x + \mathfrak{U}_i(t))^2 \left[(2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{x^2}{2\mathfrak{S}_i(t)}\right) \right] \right) dx \\
&= \int_{\mathbb{R}} \left(x^2 \left[(2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{x^2}{2\mathfrak{S}_i(t)}\right) \right] \right) dx \\
&\quad + \int_{\mathbb{R}} \left(|\mathfrak{U}_i(t)|^2 \left[(2\pi \mathfrak{S}_i(t))^{-1/2} \exp\left(-\frac{x^2}{2\mathfrak{S}_i(t)}\right) \right] \right) dx \\
&= \mathfrak{S}_i(t) + |\mathfrak{U}_i(t)|^2. \tag{3.70}
\end{aligned}$$

Combining this with (3.69) ensures that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$u(t, x) \left(a(x) - \int_{\mathbb{R}^d} u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) = \frac{u(t, x)}{2} \left[\sum_{i=1}^d (\mathfrak{S}_i(t) + |\mathfrak{U}_i(t)|^2 - |x_i|^2) \right]. \tag{3.71}$$

This and (3.66) demonstrate that for every $t \in [0, \infty)$, $x = (x_1, \dots, x_d) \in \mathbb{R}^d$ it holds that

$$\begin{aligned} & u(t, x) \left(a(x) - \int_D u(t, \mathbf{x}) a(\mathbf{x}) d\mathbf{x} \right) + \sum_{i=1}^d \frac{1}{2} |\mathfrak{m}_i|^2 (\frac{\partial^2}{\partial x_i^2} u)(t, x) \\ &= \frac{u(t, x)}{2} \left[\sum_{i=1}^d \left[|\mathfrak{m}_i|^2 \left(\left(\frac{x_i - \mathfrak{U}_i(t)}{\mathfrak{S}_i(t)} \right)^2 - \frac{1}{\mathfrak{S}_i(t)} \right) + \mathfrak{S}_i(t) + |\mathfrak{U}_i(t)|^2 - |x_i|^2 \right] \right]. \end{aligned} \quad (3.72)$$

Combining this with (3.63) establishes item ((iii)). This completes the proof of Lemma 3.5.2. \square

3.5.5 Allen–Cahn PDEs with conservation of mass

In this subsection we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the solutions of certain Allen–Cahn PDEs with cubic nonlinearity, conservation of mass and no-flux boundary conditions (cf., e.g., Rubinstein & Sternberg [RUBINSTEIN1992]).

Assume Lemma 3.5.1, let $\epsilon = \frac{1}{10}$, assume that $d \in \{1, 2, 5, 10\}$, $\mathbb{D} = [0, 1]^d$, $T \in \{1/5, 1/2, 1\}$, $N = 10$, $K_1 = K_2 = \dots = K_N = 1$, and $M_1 = M_2 = \dots = M_N = 400$, assume that $\xi^{n,m,j}, n, m, j \in \mathbb{N}$ are independent \mathcal{U}_D -distributed random variables, assume for every $m \in \mathbb{N}$ that $\gamma_m = 10^{-2}$, and assume for every $s, t \in [0, T]$, $v, x, \mathbf{x}, z, \mathbf{z} \in \mathbb{R}^d$, $y, \mathbf{y} \in \mathbb{R}$, $A \in \mathcal{B}(D)$ that $\nu_x(A) = \int_A d\mathbf{x}$, $g(x) = \exp(-\frac{1}{2}\|x\|^2)$, $\mu(x) = (0, \dots, 0)$, $\sigma(x)v = \sqrt{2}v$, $f(t, x, \mathbf{x}, y, \mathbf{y}, z, \mathbf{z}) = y - y^3 - (\mathbf{y} - \mathbf{y}^3)$, and

$$H(t, s, x, v) = R(x, x + \mu(x)(t - s) + \sigma(x)v) \quad (3.73)$$

(cf. (3.6) and (3.16)). The solution $u: [0, T] \times \mathbb{R}^d \rightarrow \mathbb{R}$ of the PDE in (3.41) then satisfies that for every $t \in [0, T]$, $x \in \mathbb{R}^d$ it holds that $u(0, x) = \exp(-\frac{1}{2}\|x\|^2)$ and

$$(\frac{\partial}{\partial t} u)(t, x) = \frac{\epsilon^2}{2} (\Delta_x u)(t, x) + u(t, x) - [u(t, x)]^3 - \int_{[0,1]^d} (u(t, \mathbf{x}) - [u(t, \mathbf{x})]^3) d\mathbf{x}. \quad (3.74)$$

In Table 3.6 we use the machine learning-based approximation method in Lemma 3.5.1 to approximately calculate the mean of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the standard deviation of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the L^1 -approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, the uncorrected sample standard deviation of the approximation error associated to $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$, and the average runtime in seconds needed for calculating one realization of $\mathbb{V}_N^{1,0}(\Theta_M^N, x)$ based on 5 independent realizations (5 independent runs). The reference value, which is used as an approximation for the unknown value $u(T, 0, 0, \dots, 0)$ of the exact solution of (3.74), has been calculated through the multilevel Picard approximation method for non-local nonlinear PDEs in Lemma 3.4.1 (see Lemma 3.4.6).

d	T	N	Mean	Std. dev.	Ref. value	L^1 -error	Std. dev. error	avg. runtime
1	$\frac{1}{5}$	10	1.1357852	0.00004031	1.1329060	0.0025414	0.0000356	11.403
2	$\frac{1}{5}$	10	1.1667370	0.00011471	1.1629940	0.0032184	0.0000986	11.351
5	$\frac{1}{5}$	10	1.1704630	0.00005095	1.1664700	0.0034231	0.0000437	11.137
10	$\frac{1}{5}$	10	1.1650314	0.00004317	1.1609480	0.0035173	0.0000372	11.179
1	$\frac{1}{2}$	10	1.3500632	0.00009184	1.3409300	0.0068111	0.0000685	11.118
2	$\frac{1}{2}$	10	1.4364596	0.00004400	1.4269380	0.0066727	0.0000308	11.215
5	$\frac{1}{2}$	10	1.4472808	0.00006241	1.4338880	0.0093402	0.0000435	11.199
10	$\frac{1}{2}$	10	1.4325428	0.00006902	1.4209860	0.0081329	0.0000486	11.174
1	1	10	1.7087326	0.00020741	1.6880340	0.0122620	0.0001229	11.070
2	1	10	1.8962484	0.00006563	1.8653440	0.0165677	0.0000352	11.076
5	1	10	1.9221128	0.00008683	1.8889600	0.0175508	0.0000460	11.082
10	1	10	1.8934832	0.00009063	1.8588340	0.0186403	0.0000488	11.148

Tab. 3.6.: Numerical simulations for the approximation method in Lemma 3.3.1 in the case of the Allen–Cahn PDEs with conservation of mass in (3.74) in Section 3.5.5.

4

HighDimPDE.jl: A Julia
package for solving high
dimensional, non-local,
non-linear PDEs

5

Interpretable Machine Learning for forecasting dynamical processes in ecosystems at different ecological states

Part III

Briding eco-evolutionary models and data

6

Econobiology: quantifying interactions and evolution in economic systems

Discussion

7.1 Conclusion

Colophon

This thesis was typeset with L^AT_EX 2_<. It uses the *Clean Thesis* style developed by Ricardo Langner. The design of the *Clean Thesis* style is inspired by user guide documents from Apple Inc.

Download the *Clean Thesis* style at <http://cleanthesis.der-ric.de/>.

Declaration

You can put your declaration here, to declare that you have completed your work solely and only with the help of the references you mentioned.

Zürich, March 18, 2022

Victor Boussange

