Bioinformatics, 35(20), 2019, 4053–4062
doi: 10.1093/bioinformatics/btz175
Advance Access Publication Date: 14 March 2019
Original Paper

Systems biology

SodaPop: a forward simulation suite for the evolutionary dynamics of asexual populations on protein fitness landscapes

Louis Gauthier [1,2], Rémicia Di Franco[1,2,3] and Adrian W. R. Serohijos [1,2,*]

[1]Département de Biochimie and [2]Centre Robert-Cedergren en Bioinformatique et Génomique, Université de Montréal, Montréal, QC H3T 1J4, Canada and [3]Enseirb-Matmeca, Bordeaux Institute of Technology, Talence 33400, France

*To whom correspondence should be addressed.
Associate Editor: Alfonso Valencia

Received on May 28, 2018; revised on January 21, 2019; editorial decision on March 9, 2019; accepted on March 12, 2019

Abstract

Motivation: Protein evolution is determined by forces at multiple levels of biological organization. Random mutations have an immediate effect on the biophysical properties, structure and function of proteins. These same mutations also affect the fitness of the organism. However, the evolutionary fate of mutations, whether they succeed to fixation or are purged, also depends on population size and dynamics. There is an emerging interest, both theoretically and experimentally, to integrate these two factors in protein evolution. Although there are several tools available for simulating protein evolution, most of them focus on either the biophysical or the population-level determinants, but not both. Hence, there is a need for a publicly available computational tool to explore both the effects of protein biophysics and population dynamics on protein evolution.

Results: To address this need, we developed SodaPop, a computational suite to simulate protein evolution in the context of the population dynamics of asexual populations. SodaPop accepts as input several fitness landscapes based on protein biochemistry or other user-defined fitness functions. The user can also provide as input experimental fitness landscapes derived from deep mutational scanning approaches or theoretical landscapes derived from physical force field estimates. Here, we demonstrate the broad utility of SodaPop with different applications describing the interplay of selection for protein properties and population dynamics. SodaPop is designed such that population geneticists can explore the influence of protein biochemistry on patterns of genetic variation, and that biochemists and biophysicists can explore the role of population size and demography on protein evolution.

Availability and implementation: Source code and binaries are freely available at https://github.com/louisgt/SodaPop under the GNU GPLv3 license. The software is implemented in C++ and supported on Linux, Mac OS/X and Windows.

Contact: adrian.serohijos@umontreal.ca

Supplementary information: Supplementary data are available at *Bioinformatics* online.

1 Introduction

Protein coding sequence evolution broadly depends on two questions. First, how random mutations change protein structure, function, and consequently organismal fitness. Second, which of these random mutations eventually survive or are purged in evolution. The first question relates to the distribution of fitness effects of random mutations (DFE), which can be determined from the knowledge of the fitness landscape or genotype-phenotype relationship. The second question relates to the role of population size, dynamics and structure in determining a mutation's fixation probability. Unfortunately, these two questions seldom intersect—the first is traditionally asked by molecular biochemists and biophysicists, and the second is asked by evolutionary biologists and population geneticists. Despite the efforts to combine these two causalities in molecular evolution (Bershtein *et al.*, 2017; DePristo *et al.*, 2005; Echave and Wilke, 2017; Goldstein, 2011; Harms and Thornton, 2013; Liberles *et al.*, 2012; Silander *et al.*, 2007), there remains a divide between these disciplines, both in concepts and methods. Indeed, to date, there is no publicly available computational tool that integrates molecular biophysics and population genetics.

Currently available methods to perform forward evolutionary simulations differ in scale, scope and flexibility, but they are generally intended to investigate potential scenarios for whole genome evolution based on observed genetic variation among natural populations (Carvajal-Rodriguez, 2008; Hoban *et al.*, 2012). Several forward-based tools including *forqs* (Kessner and Novembre, 2014), *SLiM* (Messer, 2013), *ForwSim* (Padhukasahasram *et al.*, 2008), *GENOMEPOP* (Carvajal-Rodriguez, 2008), *FFPopSim* (Zanini and Neher, 2012), *QuantiNemo* (Neuenschwander *et al.*, 2008), *GeneEvolve* (Tahmasbi and Keller, 2017), *simuPOP* (Peng and Kimmel, 2005), *SFS_CODE* (Hernandez, 2008) and *fwdpp* (Thornton, 2014) implement genetic features such as chromosome types, linkage and recombination. However, those tools concern themselves with patterns of polymorphism and structural variation across chromosomes, rather than explicit coding sequence evolution. Thus, it is challenging to model the evolution of protein sequences. Other programs such as *OncoSimulR* (Diaz-Uriarte, 2017) model the evolution of large asexual populations with user-defined fitness landscapes but enforce strictly bi-allelic loci on limited sites and do not model DNA sequences explicitly. Altogether, these tools account for neither the biochemical nor biophysical features of specific gene products.

Another class of models in protein evolution are the methods in molecular phylogenetics. Embedded in these phylogenetic approaches is a quantitative model of protein evolution that describes the amino acid substitution rates (Rodrigue *et al.*, 2010). Substitution matrices contain information on the rate at which mutations can arise and the rate at which they can fix based on their estimated selective advantage (Yang and Nielsen, 2002). These transition matrices may also contain implicit information on the biophysics of proteins—for instance, the transition probabilities between amino acids of similar chemical properties are higher than those amino acids of different types. In some cases, these matrices can also include information on the tertiary structure of proteins (Halpern and Bruno, 1998; Lartillot and Philippe, 2004; Scherrer *et al.*, 2012). However, none of these models explicitly account for the contribution of population dynamics and structure in shaping sequence evolution.

Finally, there are the physics-based models of protein evolution. To investigate the role of biophysical properties and structure to protein evolution (Bloom *et al.*, 2007; Shakhnovich, 2006; Taverna and Goldstein, 2000), these models often rely on simplified representations of proteins that fold their sequences on a lattice to calculate biophysical properties. As such, they can estimate folding stability and protein-protein interactions. Nonetheless, most of these models for protein evolution are agnostic to the role of population size and dynamics. Although there are studies that investigated the interplay between population structure and protein biophysics (Rotem *et al.*, 2018; Serohijos *et al.*, 2013; Wylie and Shakhnovich, 2011), the computational tools for performing forward evolutionary simulations that account for both factors are not yet available for the community.

More broadly, this synthesis of population genetics and protein biophysics is important in the evolution of microbes and pathogens, such as the acquisition of antibiotic resistance and viral evolution. Several works have used this integrative approach to explore the interplay between different scales in evolutionary biology. For example, Rotem *et al.* showed that the evolution and dynamics of biophysical traits of an RNA virus subjected to a neutralizing antibody are strongly dependent on population size (Rotem *et al.*, 2018). Salverda *et al.* showed how the dynamics of TEM-1 β-lactamase adaptation is determined by the topography of the fitness landscape and by mutational supply (Salverda *et al.*, 2017). Finally, Heckmann *et al.* used a population-genetic model to show that enzyme kinetic parameter evolution in Escherichia coli is constrained by strong epistatic interactions (Heckmann *et al.*, 2018).

Here we introduce SodaPop, an efficient forward simulator of the evolutionary dynamics of asexual populations with explicit genomic sequences. SodaPop is written in an object-oriented programming (OOP) framework, where the effects of population structure and the biophysical effects of mutations can be explored simultaneously. The input is the spectrum of mutational effects on biochemical or biophysical properties, which can be derived from protein engineering methods (Jia *et al.*, 2015; Kumar *et al.*, 2006; Laimer *et al.*, 2015; Yin *et al.*, 2007) or from exhaustive mutagenesis experiments such as deep mutational scanning (DMS) (Bloom, 2014; Firnberg *et al.*, 2014; Fowler and Fields, 2014). SodaPop also allows flexibility in defining fitness functions from biochemical/biophysical models that describe the evolution of proteins. Finally, the OOP framework facilitates the integration of new attributes and parameters into the model, as well as the customization of input and output data.

To the best of our knowledge, SodaPop is the first publicly available tool that explicitly combines the role of protein biophysics and population dynamics. The main program is implemented in C++ with a command line interface. We also provide scripts for analysis and visualization of the simulation results. Source code, binaries and documentation can be downloaded freely from https://github.com/louisgt/SodaPop under the GNU GPLv3 license. This software is portable on any POSIX-compliant operating system, including Linux and Mac OS/X, or on Windows using the Cygwin environment.

2 Materials and methods

2.1 Hierarchical and object-oriented description of an asexual population

We designed a hierarchical and object-oriented representation of an asexual population (Fig. 1A). A population of N cells is contained in a dynamic array. *Cell* is a data structure defined by the *Cell.h* class whose attributes include its fitness (the reproductive capacity and defined in greater detail below), its mutation rate, and the array of

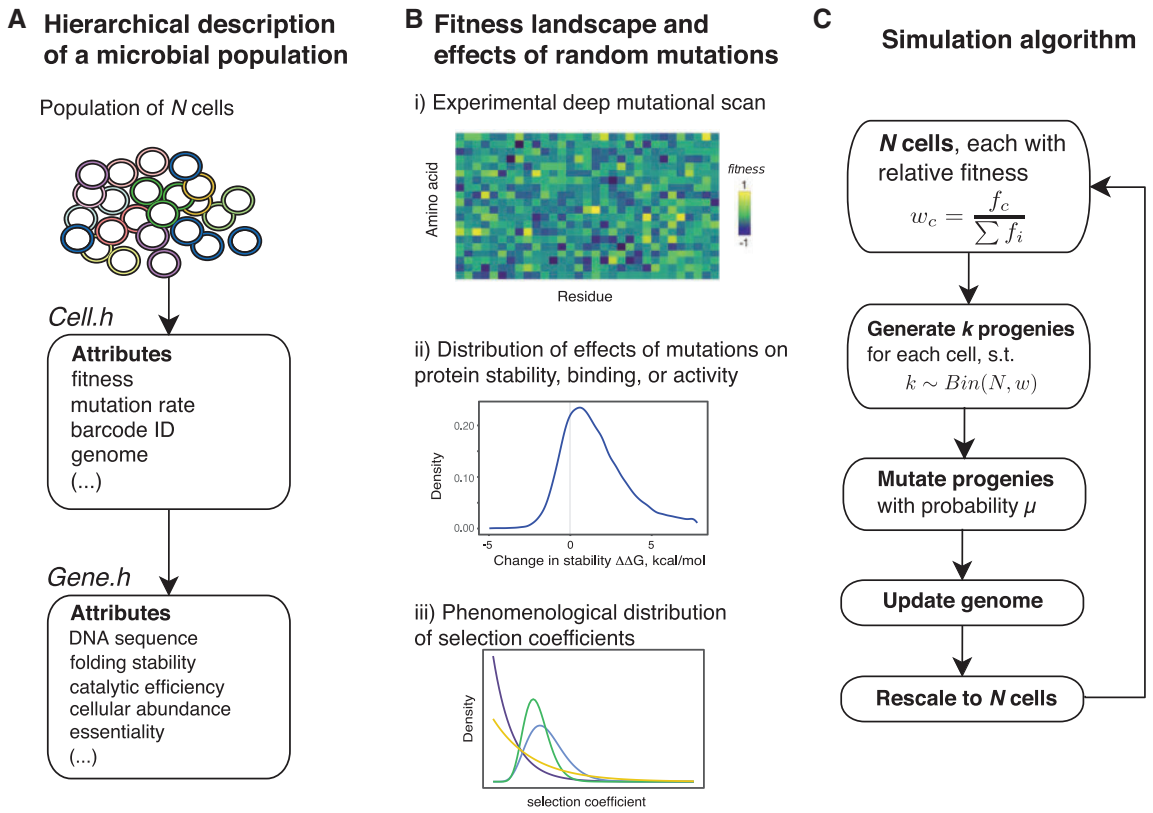

Fig. 1. Conceptual overview of SodaPop. (**a**) A model population consists of *N* distinct cells defined by attributes like fitness and mutation rate. Users can expand on these attributes in the class definition. Each cell's genome consists of a gene array, with each gene defined by attributes such as coding sequence, biochemical properties and abundance. (**b**) Users can choose from three different models of fitness landscapes. The DFE may take the form of deep mutational scanning substitution matrices for each protein, biophysical substitution matrices derived from computational tools or phenomenological distributions. (**c**) SodaPop's evolutionary algorithm. The pseudo-Wright-Fisher process iterates through every cell and draws a corresponding number of progeny for the next generation based on organismal relative fitness. These progeny are mutated with probability $L\mu$, where L is the genomic length and μ is the mutation rate. Once the whole parent population has reproduced, the new generation is rescaled to N cells

genes in the genome (Supplementary Fig. S1). Each cell is also assigned an ID (15-nt barcode), which can be used to track different lineages in the population. The *Cell* data structure can be extended to include other attributes and functions. Additionally, each Cell contains a vector of *Genes*, a data structure defined by the *Gene.h* class (Supplementary Fig. S2). Each *Gene* is characterized by its nucleotide sequence, thermodynamic folding stability, enzymatic efficiency, intracellular abundance and essentiality. Each instance of a gene also contains the variables N_a and N_s, which track the non-synonymous and synonymous substitutions, respectively. The *Gene.h* class is also extendable to include other features such as protein-protein interaction and information on binding partners (see Supplementary Manual, Section 8.2).

2.2 Evolutionary algorithm

SodaPop uses an adapted Wright-Fisher model with selection (Fisher, 1922; Wright, 1931) to evolve the population in discrete and non-overlapping generations (Fig. 1C). To propagate to the next generation, each cell (the parent) in the population is replicated to k progenies. The number k is drawn from a binomial distribution with N trials and a probability of success equal to the relative fitness of the parent w_c. This value is defined as the fitness of the parent cell normalized by the sum of the fitness of all cells in the population (Fig. 1C). The total number of progenies N could differ from the target population size N_e, thus to maintain a constant population size, it is scaled appropriately between generations. If $N < N_e$, $\Delta N = |N - N_e|$ randomly sampled cells from the offsprings are replicated. If $N > N_e$, ΔN randomly chosen progenies are removed. Population size can also be adjusted between generations to mimic population bottleneck experiments or expansion into an ecological niche (Barrick and Lenski, 2013; Ebert, 1998; Gullberg *et al.*, 2011).

During replication, the genome of each new progeny can acquire mutations at a rate $L\mu$, where μ is the mutation rate per base pair per generation and L is the length of the genome. Specifically, for each replicated cell, the number of mutations m is drawn from a binomial distribution with L trials and a probability of success equal to μ. All mutations have equal likelihood of occurring anywhere in the genome. Each mutation is thus randomly mapped to a specific site in a particular gene. Depending on the fitness landscape chosen or defined by the user (Fig. 1B), the effect of a mutation on the biophysical properties of the gene product and fitness can either come from experiments such as deep mutational scan, from physical force field calculations, or from a predefined distribution (Supplementary Manual, Chapter 5). The mutation model is also reversible, so that all effects on protein properties contributed by the previous allele are removed, to be replaced by those of the incoming substitution. The fitness landscape models are described in Section 2.3.

In the course of the simulation, SodaPop saves a snapshot of the population every T generations, as defined by the user. The snapshot of all the genomic sequences of cells in the population allows for the reconstruction of evolutionary trajectories and the calculation of

evolutionary rates. Additionally, this snapshot can be used as input for subsequent simulations. Users can also choose from multiple output formats according to their specific needs. For instance, a shorter output format only includes information at the level of cells, namely, barcodes and fitness values. A more complete format includes gene information for every cell, including DNA and amino acid sequences. This functionality allows users to tune both the level of detail required and the speed and memory usage of the program. For complete documentation on output formats, refer to the Supplementary Manual Section 3.4.

SodaPop is built upon memory-efficient data structures and a fast algorithm to achieve high computational performance and to minimize the general trade-off between flexibility and runtime (Carvajal-Rodriguez, 2008). The program can readily execute simulations in the order of 10^6 individually defined cells with runtimes clocking under a few hours on an ordinary four-core desktop computer with 8 or 16GB RAM.

2.3 Fitness landscapes

SodaPop allows users to choose from several fitness landscapes based on protein biochemistry. These fitness landscapes have been used to model and explain the rates of protein evolution (Bloom *et al.*, 2007; Serohijos *et al.*, 2012; Taverna and Goldstein, 2002), polymorphisms in protein coding regions (Serohijos and Shakhnovich, 2014a, b), epistasis (Bershtein *et al.*, 2006; Bloom *et al.*, 2007) and the log-normal distribution of protein evolutionary rates in genomes (Wolf *et al.*, 2009).

The first landscape makes the biological assumption that the fitness of the organism is proportional to the number of proteins in the cell that are folded to their native 3D structure (Taverna and Goldstein, 2002; Wylie and Shakhnovich, 2011). Using a two-state model of protein folding, the fraction of proteins in the native state (P_{nat}) is given by (Privalov and Khechinashvili, 1974)

$$P_{(\text{nat},i)} = \frac{1}{1 + e^{(\beta \Delta G_i)}}, \tag{1}$$

where $\beta = 1/k_B T$, $k_B T = 0.593$ kcal/mol, and ΔG_i is the folding free energy of the protein. If gene i has an abundance A^i in cell, fitness as a function of the total number of proteins that are folded in the cells is

$$\text{fitness}_{\text{folded}} = \sum_i (A_i \cdot P_{(\text{nat},i)}), \tag{2}$$

where the sum is over all the protein coding genes in the genome.

The second landscape assumes that fitness is proportional to metabolic flux, which is true for essential metabolic enzymes. Assuming that the proteins are enzymes in a linear metabolic pathway (Serohijos and Shakhnovich, 2014b)

$$\text{fitness}_{\text{flux}} = \frac{a_0}{\sum_i (e_i \cdot A_i \cdot P_{(\text{nat},i)})^{-1}}, \tag{3}$$

where e_i is the enzyme efficiency and a_0 is a normalizing factor that reflects the concentration of input metabolites to the pathway (see Supplementary Manual Section 7.2).

The third landscape is based on the assumption that fitness is inversely proportional to the total number of misfolded proteins in a cell (Drummond and Wilke, 2008). Misfolded proteins form aggregates that could be toxic to the cell (Bucciantini *et al.*, 2002; Stefani and Dobson, 2003). $(1 - P_{\text{nat}})$ is the probability for a protein to be misfolded, thus, fitness due to protein misfolding can be modeled as

$$\text{fitness}_{\text{toxicity}} = \exp(-c \sum_i A_i (1 - P_{(\text{nat},i)})), \tag{4}$$

where A_i is the protein abundance and c is the fitness cost per misfolded protein (Geiler-Samerotte *et al.*, 2011). When initiating a simulation with SodaPop, users can choose the fitness function by using the appropriate index in the command-line call to the program. The two fitness landscapes defined by Equations (3) and (4) may be combined to explore the simultaneous effects of metabolic flux and toxicity due to misfolding. Users can also customize their own fitness landscape by inserting a new function in the *PolyCell.h* source file (Supplementary Manual Section 8.1 and Supplementary Fig. S3). For instance, to investigate the evolution of protein-protein interactions, one could define a new function based on the assumption that proteins must satisfy the requirement of being folded and bound to a binding partner to be functional (Heo *et al.*, 2011; Manhart and Morozov, 2015; Zhang *et al.*, 2008). This particular fitness function has been used to investigate the dynamics of viral escape against a neutralizing antibody by perturbing its interaction with the Fab domain of the antibody complex (Cheron *et al.*, 2016; Rotem *et al.*, 2018).

2.4 Fitness effects of mutations

When a random mutation occurs, it may change the biochemical and biophysical properties of a protein and, in turn, the fitness of the cell. SodaPop has three approaches to model the fitness effects of mutations (Fig. 1b).

2.4.1 Mutational effects derived from a distribution

The effects of random mutations on the folding stability of globular proteins can be characterized as Gaussian distribution with mean $\mu = 0.6$ kcal/mol and standard deviation $\sigma = 0.9$ kcal/mol (Tokuriki *et al.*, 2007). These results arose from both comprehensive mutagenesis of several proteins and then estimating the effects of mutations on stability using physical force fields (Tokuriki *et al.*, 2007). These distributions are also in agreement with >5000 stability measurements of purified proteins (Kumar *et al.*, 2006). Thus, the user can define the DFE as a two-parameter distribution of the form $N(\mu, \sigma)$ or $\Gamma(\alpha, \beta)$ in the command-line (Supplementary Manual, Section 5.1). Instead of drawing $\Delta\Delta G$ values from a Gaussian, a Gamma distribution can be used to draw selection coefficients (Eyre-Walker *et al.*, 2006; Nielsen and Yang, 2003; Tamuri *et al.*, 2012) and calculate the corresponding fitness.

2.4.2 Mutational effects derived from physical force fields

Users can also provide as input a matrix that describes the change in folding stability ($\Delta\Delta G_{\text{folding}}$) or activity ($k_{\text{cat}}/K_M$) for all possible one-away substitutions at all residues. These quantities can be provided by the user (using parameter *–i*) as look-up tables that are accessed at runtime. The entries in these tables are derived from computational protein engineering tools such as Rosetta (Das and Baker, 2008), Eris (Yin *et al.*, 2007) or FoldX (Guerois *et al.*, 2002) prior to the evolutionary simulation. Updating the protein folding stability or activity also updates the fitness using either Equations (2)–(4).

2.4.3 Mutational effects derived from experiment

The user can also provide as input fitness effects from experimental DMS approaches (Araya and Fowler, 2011). DMS combines mutational library generation, selection, and high-throughput sequencing to assay the fitness of up to 95% of possible one-away mutations to

a protein. To date, over a dozen systematic and exhaustive DMS assays have been conducted in various proteins to determine their local fitness landscape (Fowler and Fields, 2014; Wrenbeck *et al.*, 2017a). New computational tools can leverage the information in these experimental datasets to predict mutational effects for full proteomes (Gray *et al.*, 2018), allowing the creation of comprehensive substitution matrices for thousands of proteins. Users can input these matrices using the parameter $-i$ (Supplementary Manual, Section 3.3).

Indels are another major source of innovation in protein evolution. Although the current version of SodaPop does not explicitly handle them, some types of indels can already be modeled (see Section 8.3 of the Supplementary Manual).

2.5 Analysis of simulation data

The SodaPop package includes shell and R scripts for visualizing the population dynamics and the time-series of the fitness and biophysical properties of proteins. These scripts also allow for the tracking of lineages and clonal structure and the calculation of protein evolutionary rates. A detailed list and description of outputs and analyses can be found in Chapter 6 of the User Manual. Since the program registers explicit DNA and protein sequences for all the cells in the population, a single simulation can generate thousands to millions of sequences per saved generation. These sequences can be used as starting points for high-resolution methods in molecular evolution and phylogenetics. For example, the sequences can be used to create multiple sequence alignments and apply standard molecular evolution analyses, such as the McDonald-Kreitman test or Tajima's *D*.

3 Results

We describe four applications of SodaPop to demonstrate its broad scope and flexibility.

3.1 Application I: population dynamics on fitness landscapes based on protein biochemistry

To show how SodaPop can be used to explore population dynamics and protein properties on biochemical fitness landscapes, we simulated a population of $N_e = 10^4$ cells on the fitness landscape defined by Equation (3). Each cell contains ten genes in the folate biosynthesis pathway of *Escherichia coli*. Shown in Figure 2A and B are the mean population fitness and the average stability of the ten genes over the course of the simulation. At a finer temporal resolution, we can trace the segregation and eventual fixation of arising mutations and the effects of clonal interference (Fig. 2C; Supplementary Fig. S4).

3.2 Application II: simulating population dynamics with barcoded cells for lineage-tracking and competition assays

The ability to track the lineages of cells and frequency of clones in an evolving population is crucial because it enables estimating the selective advantage of mutations, the extent of clonal interference and the establishment time of adaptive mutations. Tracking the population dynamics typically makes use of neutral genetic or fluorescent markers (Hegreness *et al.*, 2006; Illingworth and Mustonen, 2012; Moura de Sousa *et al.*, 2013; Pinkel, 2007; Zhang *et al.*, 2012). More recently, through the introduction of randomized and unique barcodes into the chromosomes, it is now possible to track the population dynamics and lineages at the resolution of single cells [Blundell and Levy (2014); Gerrits *et al.* (2010); Levy *et al.* (2015); Venkataram *et al.* (2016)]. Nonetheless, because the resulting dynamics can be complex, these experimental results are complemented by simulations that could provide null expectations. SodaPop allows for this functionality. By assigning a unique barcode identifier to each cell, one can trace the lineage and history of cells in the simulation. As an example, we performed a simulation of laboratory evolution of $N_e = 10^4$ cells, each assigned a 15-nt barcode (Fig. 3). To mimic standing genetic variation, different fitness values can be assigned to the barcodes (Supplementary Manual Section 4.2). Figure 3A is a lineage density plot showing the relative share of each barcode in the population and Figure 3B shows the frequency of each barcode.

3.3 Application III: simulating coding sequence evolution under the constraints of both population dynamics and selection for folding stability

Next, we demonstrate how SodaPop can be used to perform simulations of protein sequence evolution that account for both effects of selection for biophysical properties, such as folding stability, and population dynamics. Specifically, we show the performance of SodaPop in recapitulating the amino acid conservation among orthologs. As a model system, we chose aminodeoxychorismate synthase (pabB gene in *Escherichia coli*), an enzyme in the folate biosynthesis pathway.

To generate simulated orthologs, we performed evolutionary simulations of 10^5 cells with *E. coli* pabB for 10^6 generations, under selection for folding stability (Eq. 3). The effects of random mutations on fitness are derived from a 452 residue $\times$ 20 amino acid matrix that contains values for changes in folding stability, $\Delta\Delta G = \Delta G_{\text{mut}} - \Delta G_{\text{wildtype}}$. The $\Delta\Delta G$ values are estimated using a physical force field (Yin *et al.*, 2007) and using the pabB 3D structure from Protein Data Bank (PDB '1K0E'). We first performed an equilibration simulation where an initially monoclonal population was evolved for 10^5 generations to reach mutation-selection balance. Then, to mimic divergence from a common ancestor, we used the endpoint of the equilibration simulation as the starting population for 250 independent evolutionary simulations. These divergent simulations are performed for 10^6 generations, thereby ensuring that the distribution of pairwise sequence identities for simulated sequences matches that of the extant orthologs of *pabB*.

To compare the simulated sequences with extant orthologs of pabB, we retrieved the top hits of a protein sequence search in OrthoDB for bacteria (Waterhouse *et al.*, 2013). We excluded sequences with an absolute length difference of more than 20 bp. We then aligned the 657 remaining sequences with *hmmalign* (http://hmmer.org/) to the corresponding HMM of Pfam domains in pabB (PF04715 and PF00425) and trimmed the flanking gaps from the alignment. The amino acid conservation of simulated and orthologous pabB sequences from multiple sequence alignment is shown in Supplementary Figures S5 and S6, respectively.

We used Shannon entropy (Shannon, 1948) as a measure of per-site conservation for each residue z

$$S_z = -\sum_{i=1}^{N} p_i \ln(p_i), \tag{5}$$

where p_i is the observed frequency of amino acid i in that specific site. A higher entropy indicates that a residue's identity is more frequently substituted based on the multiple sequence alignment. Conversely, as the entropy approaches zero, that residue's identity is generally conserved throughout evolution. The site-specific entropies of the simulated pabB sequences are significantly correlated

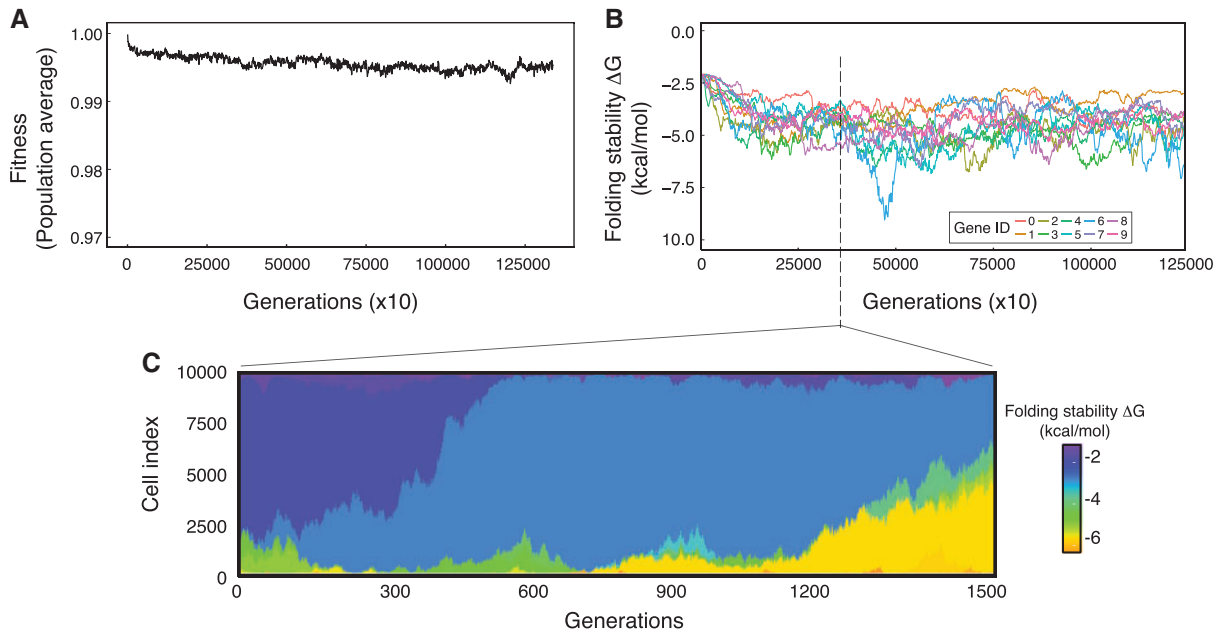

Fig. 2. Application I: population dynamics on a fitness landscape based on metabolic flux (Eq. 4). We modelled the evolution of a population of $N_e = 10^4$ cells with ten genes from the folate biosynthesis pathway. (**A**) Fitness of the population evolving towards mutation-selection balance. (**B**) Average folding stability of the ten genes over the course of the simulation. Each gene is coloured differently and referenced by a numeric identifier. (**C**) Muller plot of the folding stability dynamics of gene 1 for a window of 1.5×10^4 generations. Cells are indexed along the *y* axis and grouped according to kinship. Genetically identical clones are colored according to their folding stability

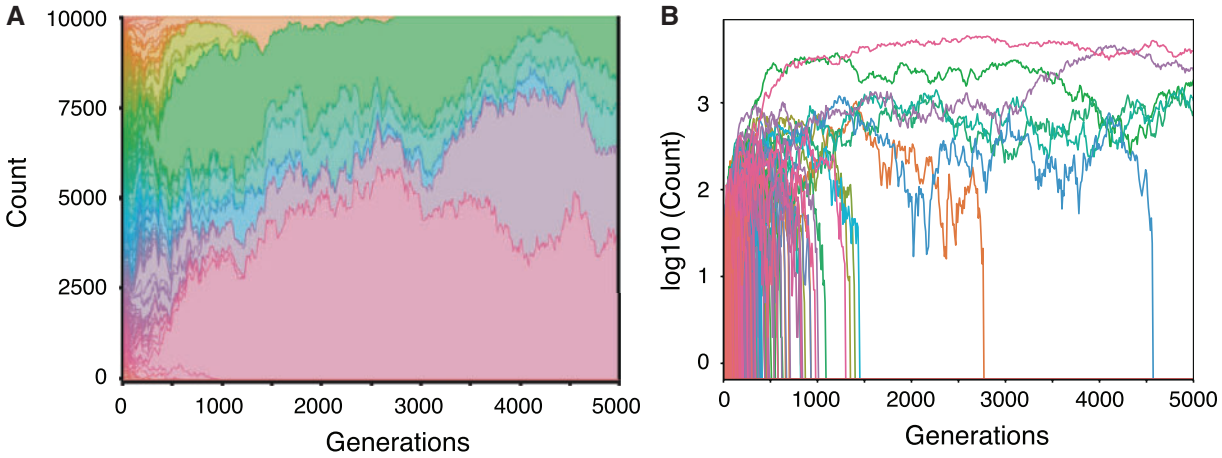

Fig. 3. Application II: evolution under selection for folding stability (Eq. 2). (**A**) SodaPop tracks the segregation of lineages concurrently using 15-nt barcodes. Each cell was assigned a unique barcode at time zero. Each colour represents the relative share of each barcode in the population at any time point. The information in the left panel is represented in (**B**) as the logarithm of the number of cells sharing the same barcode

with corresponding extant orthologs (Fig. 4A). Moreover, the magnitude of this correlation is dependent on population size (Fig. 4B), demonstrating the importance of population dynamics in shaping protein sequence evolution. It is also expected that core residues are more conserved than solvent exposed residues (Ramsey *et al.*, 2011). Indeed, the extent of the correlation between entropy and surface-accessibility of residues (relative solvent accessibility; RSA) in real sequences of pabB (Fig. 4C) is also recapitulated in simulation (Fig. 4D). Lastly, the strength of this correlation is also dependent on population size (Fig. 4E).

Next, we explored if simulations with SodaPop could also capture the co-evolution between sites and whether this result is dependent on N_e. Here, co-evolution is calculated by mutual information (MI) using MISTIC (Simonetti *et al.*, 2013). Indeed, increasing the stringency of selection due to larger N_e increases the number of co-evolving sites and the strength co-evolution between those sites (Fig. 5A). Interestingly, population dynamics seems to influence the rank-ordered distribution of cumulative site co-evolution (CMI) with increasing N_e (Fig. 5B). In protein engineering, site co-evolution is now being used to predict *de novo* 3D structure of proteins and protein complexes (Marks *et al.*, 2012). Being able to explore how this property is influenced by population dynamics will be of practical importance to the community.

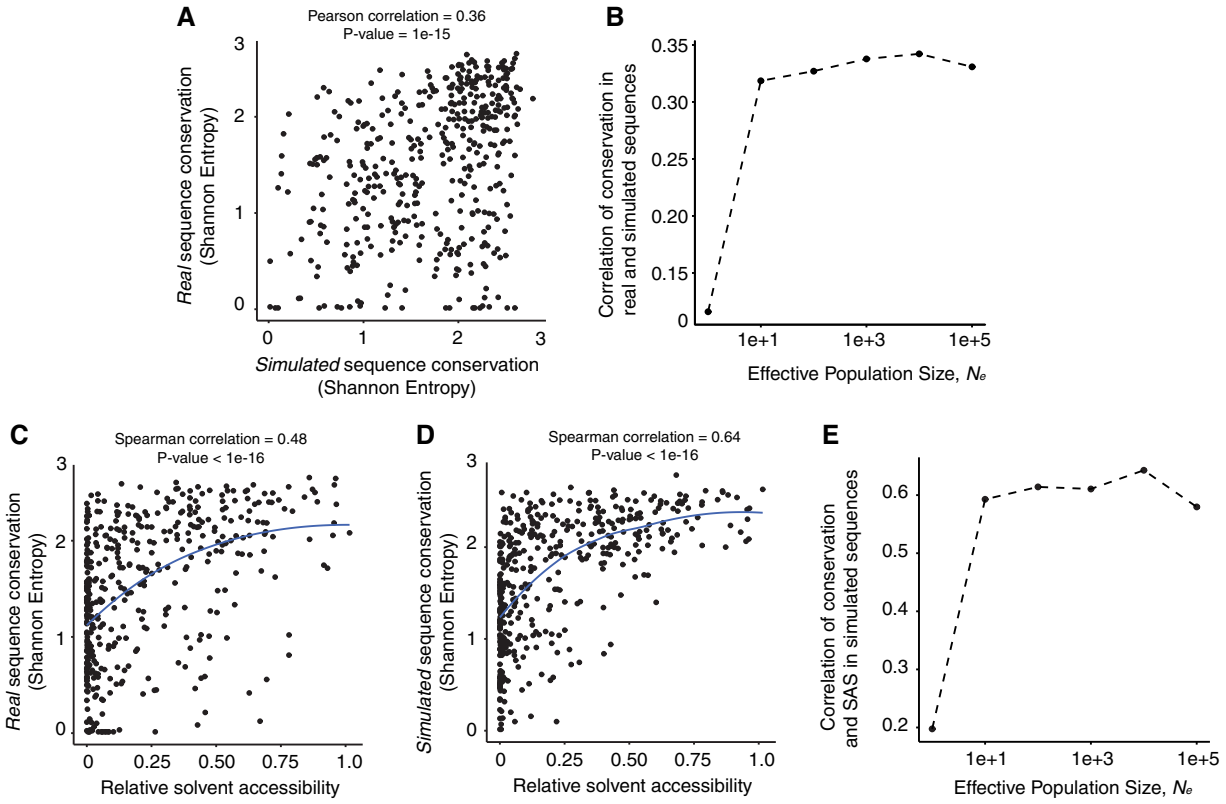

Fig. 4. Application III: coding sequence evolution under the constraints of both population dynamics and selection for folding stability. (**A**) Comparison of Shannon entropy per site between real *pabB* orthologs and sequences simulated with SodaPop ($N_e = 10^5$ cells). (**B**) Dependence on population size (N_e) of the sequence conservation patterns recapitulated by SodaPop using a biophysical fitness landscape. (**C and D**) Simulated sequences recapitulate the strength of the correlation between relative solvent accessibility and the conservation of real sequences. Curves are Lowess fits using all data points (smoothing parameter $\alpha = 1$). (**E**) Dependence on N_e of the correlation between per-site conservation and surface accessibility

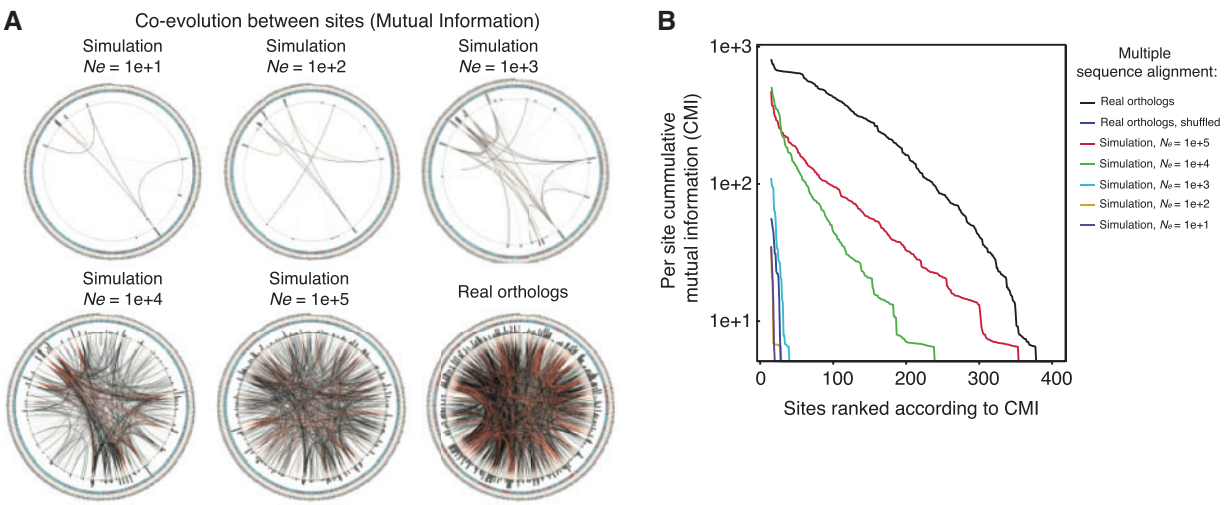

Fig. 5. Application III: co-evolution between sites and the effects of population dynamics. (**A**) Mutual information (MI) in the multiple alignments of sequences generated from simulation under different population sizes (N_e). The outer circle indicates the site in the multiple sequence alignment and the amino acid identity of the sequence at the beginning of simulation. Pairs of sites with MI greater than 6.5 are connected by edges, with the top 5% colored red. Mutual information is calculated from MSAs using the MISTIC web-server (Simonetti *et al.*, 2013). (**B**) Positions are rank-ordered according to their cumulative MI (CMI) for multiple sequence alignments from real orthologs or simulations under different population sizes

3.4 Application IV: simulating sequence evolution with selection for both folding stability and catalytic activity

Mutations affect not only protein stability, but also enzyme activity. For some residues and structural motifs, the trade-off between activity and stability can constrain evolutionary paths (Meiering *et al.*, 1992). Thus, we compared two models of enzyme evolution with SodaPop: (1) a model with effects on protein stability alone and (2) a model with effects on enzyme activity in addition to

Table 1. Application IV: Pearson correlation coefficient between simulated *amiE* sequences and biological orthologs

	Pearson correlation	*P*-value
Selection for folding	0.33	3.85×10^{-10}
Selection for folding and activity	0.43	2.2×10^{-16}

stability. For the first model, we used FoldX's PositionScan command (Guerois *et al.*, 2002) to get the landscape of effects on folding stability of a protein. For the second model, we took advantage of a recent deep-mutational scanning assay comprising >96.3% of all possible one-away non-synonymous substitutions for hydrolysis activity of the amiE amidase (Wrenbeck *et al.*, 2017b). We regressed the previously calculated effects on folding from the DMS data to extract the contribution of each mutation to enzyme activity (Supplementary Methods). We find that supplementing the effects on protein stability with effects on the catalytic activity increases the recapitulation of sequence conservation in natural orthologs in comparison with sequences from simulation based on folding stability alone (Table 1, Supplementary Fig. S7).

3.5 Performance and runtime

We benchmarked SodaPop for six population sizes spanning five orders of magnitude with cells containing ten genes (total genome size 6.5 kbp). All simulations were run on a standard iMac desktop with a 3.2GHz Intel Core i5 processor and 16GB RAM. Figure 6 shows that runtime is quasi-monomial with respect to population size, with an exponent of 1.1 (Supplementary Fig. S8). Simulating up to a million cells for long time periods is entirely tractable using standard desktop computers. We limited our desktop benchmarking to $N = 10^6$ cells, as higher orders of magnitude introduce a shift in performance due to a RAM bottleneck. The simulation of populations with higher orders of magnitude requires a larger amount of memory than the current standard in commercial desktop computers. However, these larger population sizes can be simulated on high-performance computing clusters where memory allocation is not limiting.

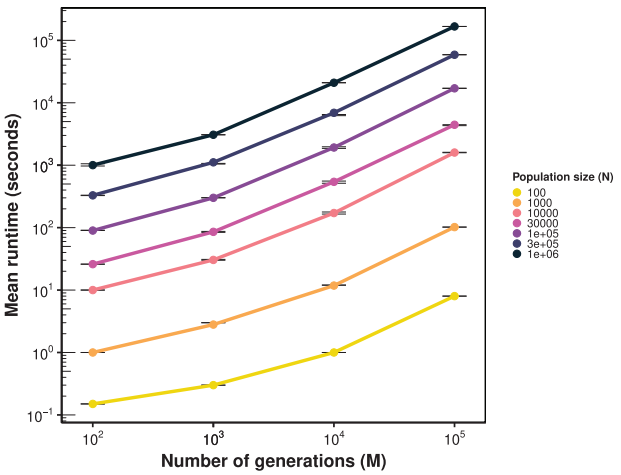

Fig. 6. Benchmarking of SodaPop for different population sizes. The runtime of SodaPop is shown for varying population sizes and simulation length. The time step for each test case was set to 0.01*N*. Each data point represents the average runtime over 100 simulations for a particular condition. Error bars represent standard error of the mean (SEM)

4 Discussion and conclusion

We have created SodaPop, a fast and scalable tool for evolutionary simulations based on biochemical and biophysical fitness landscapes. Considering the need to address questions at the interface of molecular evolution and population genetics, and with most of the current computational methods unable to account for explicit clonal dynamics, we believe SodaPop provides a comprehensive and extensible framework that can encompass a wide array of evolutionary scenarios. Briefly, we showed that our program can be used, among other things, to simulate evolution under selection for specific biophysical properties, to provide null expectations for lineage-tracking experiments of laboratory evolution, and to model coding sequence evolution and co-evolution. To broaden these possibilities and achieve greater predictive power, we intend to implement future models of fitness landscapes, such as a systems-level description of the cell that could integrate omics data—transcription level, gene regulation networks and protein-protein interactions. We also aim to expand the scope of the program to other types of genetic changes by integrating features of recombination and indels to SodaPop. In the case of recombination, the first and simplest approach is to assume an ad hoc distribution of effects, as done in other forward simulation algorithms. This approach is appropriate for modeling whole genome recombination events (inter-genic). However, this does not account for the consequence of recombination in specific genes (intra-genic) and on their biophysical properties, which is the overaching motivation for SodaPop. The second approach is to phenomenologically model the fitness effect of recombination and its dependence on recombination sites within proteins. Random recombination of homologous sequences found that crossover sites in the middle of the protein are more deleterious because it likely impacts the packing of the core (Romero and Arnold, 2009). The third approach is to have a sequence-based model of recombination and to calculate the biophysical properties of the recombinant sequences. Considering that for a given evolutionary run, at least 10^6 sequences need to be folded (but few of which will be selected), on-the-fly calculation of folding stability for a sequence is possible only in lattice models of protein folding (Shakhnovich and Gutin, 1993; Voigt *et al.*, 2002), barring the associated computational cost.

In the case of indels, their effects in protein evolution are more nuanced and depend on structural and biophysical considerations. For example, random indels occurring in reverse turns, loops, or surfaces are considered to be less deleterious than those in beta sheets, helices, or protein cores (Benner *et al.*, 1993; Hsing and Cherkasov, 2008; Pascarella and Argos, 1992). A detailed model of indels will require explicit folding simulation of proteins for each mutation, which can also be performed using lattice models of proteins.

The object-oriented design of SodaPop will facilitate these future developments. In a technical perspective, there are several features that could improve the performance and practical use of SodaPop. First, modeling mutation history for each cell using a linked list of mutations instead of explicit sequence change could reduce the memory required to store cell objects by a significant factor without incurring any information loss. Second, circumventing the command-line application with a graphical user interface (GUI) wrapper will facilitate user interaction in creating input files, setting up simulations and choosing the appropriate parameters. Third, implementing multi-threading options in the main program loop and in the analysis pipeline will allow the program to run on multiple processors in parallel, which can significantly improve runtime on high-performance computing clusters. These features are currently under development for future versions.

Acknowledgements

We would like to thank members of the Serohijos Lab for testing the program and contributing valuable questions and ideas.

Funding

A.W.R.S. acknowledges a grant from (Natural Sciences and Engineering Research Council) and start-up funds from Université de Montréal. L.G. is a recipient of a doctoral fellowship from Université de Montréal's Faculté des études supérieures et postdoctorales. R.D.F. was supported by ENSEIRB-MATMECA Bordeaux.

Conflict of Interest: none declared.

References

Araya,C.L. and Fowler,D.M. (2011) Deep mutational scanning: assessing protein function on a massive scale. *Trends Biotechnol.*, **29**, 435–442.

Barrick,J.E. and Lenski,R.E. (2013) Genome dynamics during experimental evolution. *Nat. Rev. Genet.*, **14**, 827–839.

Benner,S. *et al.* (1993) Empirical and structural models for insertions and deletions in the divergent evolution of proteins. *J. Mol. Biol.*, **229**, 1065–1082.

Bershtein,S. *et al.* (2006) Robustness-epistasis link shapes the fitness landscape of a randomly drifting protein. *Nature*, **444**, 929–932.

Bershtein,S. *et al.* (2017) Bridging the physical scales in evolutionary biology: from protein sequence space to fitness of organisms and populations. *Curr. Opin. Struct. Biol.*, **42**, 31–40.

Bloom,J.D. (2014) An experimentally determined evolutionary model dramatically improves phylogenetic fit. *Mol. Biol. Evol.*, **31**, 1956–1978.

Bloom,J.D. *et al.* (2007) Thermodynamics of neutral protein evolution. *Genetics*, **175**, 255–266.

Blundell,J.R. and Levy,S.F. (2014) Beyond genome sequencing: lineage tracking with barcodes to study the dynamics of evolution, infection, and cancer. *Genomics*, **104**, 417–430.

Bucciantini,M. *et al.* (2002) Inherent toxicity of aggregates implies a common mechanism for protein misfolding diseases. *Nature*, **416**, 507–511.

Carvajal-Rodriguez,A. (2008) Simulation of genomes: a review. *Curr. Genom.*, **9**, 155–159.

Cheron,N. *et al.* (2016) Evolutionary dynamics of viral escape under antibodies stress: a biophysical model. *Protein Sci.*, **25**, 1332–1340.

Das,R. and Baker,D. (2008) Macromolecular modeling with rosetta. *Annu. Rev. Biochem.*, **77**, 363–382. PMID: 18410248.

DePristo,M.A. *et al.* (2005) Missense meanderings in sequence space: a biophysical view of protein evolution. *Nat. Rev. Genet.*, **6**, 678–687.

Diaz-Uriarte,R. (2017) Oncosimulr: genetic simulation with arbitrary epistasis and mutator genes in asexual populations. *Bioinformatics*, **33**, 1898–1899.

Drummond,D.A. and Wilke,C.O. (2008) Mistranslation-induced protein misfolding as a dominant constraint on coding-sequence evolution. *Cell*, **134**, 341–352.

Ebert,D. (1998) Experimental evolution of parasites. *Science*, **282**, 1432–1435.

Echave,J. and Wilke,C.O. (2017) Biophysical models of protein evolution: understanding the patterns of evolutionary sequence divergence. *Annu. Rev. Biophys.*, **46**, 85–103.

Eyre-Walker,A. *et al.* (2006) The distribution of fitness effects of new deleterious amino acid mutations in humans. *Genetics*, **173**, 891–900.

Firnberg,E. *et al.* (2014) A comprehensive, high-resolution map of a gene's fitness landscape. *Mol. Biol. Evol.*, **31**, 1581–1592.

Fisher,R.A. (1990) On the dominance ratio. 1922. *Bull. Math. Biol.*, **52**, 297–318.

Fowler,D.M. and Fields,S. (2014) Deep mutational scanning: a new style of protein science. *Nat. Methods*, **11**, 801–807.

Geiler-Samerotte,K.A. *et al.* (2011) Misfolded proteins impose a dosage-dependent fitness cost and trigger a cytosolic unfolded protein response in yeast. *Proc. Natl. Acad. Sci. USA*, **108**, 680–685.

Gerrits,A. *et al.* (2010) Cellular barcoding tool for clonal analysis in the hematopoietic system. *Blood*, **115**, 2610–2618.

Goldstein,R.A. (2011) The evolution and evolutionary consequences of marginal thermostability in proteins. *Proteins*, **79**, 1396–1407.

Gray,V.E. *et al.* (2018) Quantitative missense variant effect prediction using large-scale mutagenesis data. *Cell Systems*, **6**, 116–124.

Guerois,R. *et al.* (2002) Predicting changes in the stability of proteins and protein complexes: a study of more than 1000 mutations. *J. Mol. Biol.*, **320**, 369–387.

Gullberg,E. *et al.* (2011) Selection of resistant bacteria at very low antibiotic concentrations. *PLoS Pathog*, **7**, e1002158.

Halpern,A.L. and Bruno,W.J. (1998) Evolutionary distances for protein-coding sequences: modeling site-specific residue frequencies. *Mol. Biol. Evol.*, **15**, 910–917.

Harms,M.J. and Thornton,J.W. (2013) Evolutionary biochemistry: revealing the historical and physical causes of protein properties. *Nat. Rev. Genet.*, **14**, 559–571.

Heckmann,D. *et al.* (2018) Modeling genome-wide enzyme evolution predicts strong epistasis underlying catalytic turnover rates. *Nat. Commun.*, **9**, 5270.

Hegreness,M. *et al.* (2006) An equivalence principle for the incorporation of favorable mutations in asexual populations. *Science*, **311**, 1615–1617.

Heo,M. *et al.* (2011) Topology of protein interaction network shapes protein abundances and strengths of their functional and nonspecific interactions. *Proc. Natl. Acad. Sci. USA*, **108**, 4258–4263.

Hernandez,R.D. (2008) A flexible forward simulator for populations subject to selection and demography. *Bioinformatics*, **24**, 2786–2787.

Hoban,S. *et al.* (2012) Computer simulations: tools for population and evolutionary genetics. *Nat. Rev. Genet.*, **13**, 110 EP.

Hsing,M. and Cherkasov,A. (2008) Indel pdb: a database of structural insertions and deletions derived from sequence alignments of closely related proteins. *BMC Bioinformatics*, **9**, 293.

Illingworth,C.J.R. and Mustonen,V. (2012) A method to infer positive selection from marker dynamics in an asexual population. *Bioinformatics*, **28**, 831–837.

Jia,L. *et al.* (2015) Structure based thermostability prediction models for protein single point mutations with machine learning tools. *PLoS One*, **10**, e0138022.

Kessner,D. and Novembre,J. (2014) Forqs: forward-in-time simulation of recombination, quantitative traits and selection. *Bioinformatics*, **30**, 576–577.

Kumar,M.D. *et al.* (2006) Protherm and pronit: thermodynamic databases for proteins and protein-nucleic acid interactions. *Nucleic Acids Res.*, **34**, D204–D206.

Laimer,J. *et al.* (2015) Maestro—multi agent stability prediction upon point mutations. *Bmc Bioinformatics*, **16**, 116.

Lartillot,N. and Philippe,H. (2004) A Bayesian mixture model for across-site heterogeneities in the amino-acid replacement process. *Mol. Biol. Evol.*, **21**, 1095–1109.

Levy,S.F. *et al.* (2015) Quantitative evolutionary dynamics using high-resolution lineage tracking. *Nature*, **519**, 181–186.

Liberles,D.A. *et al.* (2012) The interface of protein structure, protein biophysics, and molecular evolution. *Protein Sci.*, **21**, 769–785.

Manhart,M. and Morozov,A.V. (2015) Protein folding and binding can emerge as evolutionary spandrels through structural coupling. *Proc. Natl. Acad. Sci. USA*, **112**, 1797–1802.

Marks,D.S. *et al.* (2012) Protein structure prediction from sequence variation. *Nat. Biotechnol.*, **30**, 1072.

Meiering,E.M. *et al.* (1992) Effect of active site residues in barnase on activity and stability. *J. Mol. Biol.*, **225**, 585–589.

Messer,P.W. (2013) Slim: simulating evolution with selection and linkage. *Genetics*, **194**, 1037–1039.

Moura de Sousa,J.A. *et al.* (2013) An abc method for estimating the rate and distribution of effects of beneficial mutations. *Genome Biol. Evol.*, **5**, 794–806.

Neuenschwander,S. *et al.* (2008) Quantinemo: an individual-based program to simulate quantitative traits with explicit genetic architecture in a dynamic metapopulation. *Bioinformatics*, **24**, 1552–1553.

Nielsen,R. and Yang,Z. (2003) Estimating the distribution of selection coefficients from phylogenetic data with applications to mitochondrial and viral dna. *Mol. Biol. Evol.*, **20**, 1231–1239.

Padhukasahasram,B. *et al*. (2008) Exploring population genetic models with recombination using efficient forward-time simulations. *Genetics*, **178**, 2417–2427.

Pascarella,S. and Argos,P. (1992) Analysis of insertions/deletions in protein structures. *J. Mol. Biol*., **224**, 461–471.

Peng,B. and Kimmel,M. (2005) Simupop: a forward-time population genetics simulation environment. *Bioinformatics*, **21**, 3686–3687.

Pinkel,D. (2007) Analytical description of mutational effects in competing asexual populations. *Genetics*, **177**, 2135–2149.

Privalov,P. and Khechinashvili,N. (1974) A thermodynamic approach to the problem of stabilization of globular protein structure: a calorimetric study. *J. Mol. Biol*., **86**, 665–684.

Ramsey,D.C. *et al*. (2011) The relationship between relative solvent accessibility and evolutionary rate in protein evolution. *Genetics*, **188**, 479–488.

Rodrigue,N. *et al*. (2010) Mutation-selection models of coding sequence evolution with site-heterogeneous amino acid fitness profiles. *Proc. Natl. Acad. Sci. USA*, **107**, 4629–4634.

Romero,P.A. and Arnold,F.H. (2009) Exploring protein fitness landscapes by directed evolution. *Nat. Rev. Mol. Cell Biol*., **10**, 866–876.

Rotem,A. *et al*. (2018) Evolution on the biophysical fitness landscape of an rna virus. *Mol. Biol. Evol*., **35**, 2390–2400.

Salverda,M.L.M. *et al*. (2017) Adaptive benefits from small mutation supplies in an antibiotic resistance enzyme. *Proc. Natl. Acad. Sci. USA*, **114**, 12773–12778.

Scherrer,M.P. *et al*. (2012) Modeling coding-sequence evolution within the context of residue solvent accessibility. *BMC Evol. Biol*., **12**, 179.

Serohijos,A.W. and Shakhnovich,E.I. (2014) Contribution of selection for protein folding stability in shaping the patterns of polymorphisms in coding regions. *Mol. Biol. Evol*., **31**, 165–176.

Serohijos,A.W. and Shakhnovich,E.I. (2014) Merging molecular mechanism and evolution: theory and computation at the interface of biophysics and evolutionary population genetics. *Curr. Opin. Struct. Biol*., **26**, 84–91.

Serohijos,A.W. *et al*. (2012) Protein biophysics explains why highly abundant proteins evolve slowly. *Cell Rep*., **2**, 249–256.

Serohijos,A.W. *et al*. (2013) Highly abundant proteins favor more stable 3d structures in yeast. *Biophys. J*, **104**, L1–L3.

Shakhnovich,E. (2006) Protein folding thermodynamics and dynamics: where physics, chemistry, and biology meet. *Chem Rev*, **106**, 1559–1588.

Shakhnovich,E.I. and Gutin,A.M. (1993) Engineering of stable and fast-folding sequences of model proteins. *Proc. Natl. Acad. Sci. USA*, **90**, 7195–7199.

Shannon,C.E. (1948) A mathematical theory of communication. *Bell Syst. Tech. J*., **27**, 379–423.

Silander,O.K. *et al*. (2007) Understanding the evolutionary fate of finite populations: the dynamics of mutational effects. *PLoS Biol*., **5**, e94.

Simonetti,F.L. *et al*. (2013) Mistic: mutual information server to infer coevolution. *Nucleic Acids Res*., **41**, W8–W14.

Stefani,M. and Dobson,C.M. (2003) Protein aggregation and aggregate toxicity: new insights into protein folding, misfolding diseases and biological evolution. *J. Mol. Med. (Berl)*, **81**, 678–699.

Tahmasbi,R. and Keller,M.C. (2017) Geneevolve: a fast and memory efficient forward-time simulator of realistic whole-genome sequence and snp data. *Bioinformatics*, **33**, 294–296.

Tamuri,A.U. *et al*. (2012) Estimating the distribution of selection coefficients from phylogenetic data using sitewise mutation-selection models. *Genetics*, **190**, 1101–1115.

Taverna,D.M. and Goldstein,R.A. (2000) The distribution of structures in evolving protein populations. *Biopolymers*, **53**, 1–8.

Taverna,D.M. and Goldstein,R.A. (2002) Why are proteins marginally stable? *Proteins*, **46**, 105–109.

Thornton,K.R. (2014) A c++ template library for efficient forward-time population genetic simulation of large populations. *Genetics*, **198**, 157–166.

Tokuriki,N. *et al*. (2007) The stability effects of protein mutations appear to be universally distributed. *J. Mol. Biol*., **369**, 1318–1332.

Venkataram,S. *et al*. (2016) Development of a comprehensive genotype-to-fitness map of adaptation-driving mutations in yeast. *Cell*, **166**, 1585–1596.

Voigt,C.A. *et al*. (2002) Protein building blocks preserved by recombination. *Nat. Struct. Biol*., **9**, 553–558.

Waterhouse,R.M. *et al*. (2013) Orthodb: a hierarchical catalog of animal, fungal and bacterial orthologs. *Nucleic Acids Res*., **41**, D358–D365.

Wolf,Y.I. *et al*. (2009) The universal distribution of evolutionary rates of genes and distinct characteristics of eukaryotic genes of different apparent ages. *Proc. Natl. Acad. Sci. USA*, **106**, 7273–7280.

Wrenbeck,E.E. *et al*. (2017) Deep sequencing methods for protein engineering and design. *Curr. Opin. Struct. Biol*., **45**, 36–44.

Wrenbeck,E.E. *et al*. (2017) Single-mutation fitness landscapes for an enzyme on multiple substrates reveal specificity is globally encoded. *Nat. Commun*., **8**, 15695.

Wright,S. (1931) Evolution in mendelian populations. *Genetics*, **16**, 97–159.

Wylie,C.S. and Shakhnovich,E.I. (2011) A biophysical protein folding model accounts for most mutational fitness effects in viruses. *Proc. Natl. Acad. Sci. USA*, **108**, 9916–9921.

Yang,Z. and Nielsen,R. (2002) Codon-substitution models for detecting molecular adaptation at individual sites along specific lineages. *Mol. Biol. Evol*., **19**, 908–917.

Yin,S. *et al*. (2007) Eris: an automated estimator of protein stability. *Nat. Methods*, **4**, 466–467.

Zanini,F. and Neher,R.A. (2012) Ffpopsim: an efficient forward simulation package for the evolution of large populations. *Bioinformatics*, **28**, 3332–3333.

Zhang,J. *et al*. (2008) Constraints imposed by non-functional protein-protein interactions on gene expression and proteome size. *Mol. Syst. Biol*., **4**, 210.

Zhang,W. *et al*. (2012) Estimation of the rate and effect of new beneficial mutations in asexual populations. *Theor. Population Biol*., **81**, 168–178.