Modeling emergence of *Wolbachia* toxin-antidote protein functions with an evolutionary algorithm

John Beckmann[1]*, Joe Gillespie[2] and Daniel Tauritz[3]

[1]Department of Entomology and Plant Pathology, Auburn University, Auburn, AL, United States, [2]Department of Microbiology and Immunology, School of Medicine, University of Maryland, Baltimore, Baltimore, MD, United States, [3]Department of Computer Science and Software Engineering, Auburn University, Auburn, AL, United States

Evolutionary algorithms (EAs) simulate Darwinian evolution and adeptly mimic natural evolution. Most EA applications in biology encode high levels of abstraction in top-down population ecology models. In contrast, our research merges protein alignment algorithms from bioinformatics into codon based EAs that simulate molecular protein string evolution from the bottom up. We apply our EA to reconcile a problem in the field of *Wolbachia* induced cytoplasmic incompatibility (CI). *Wolbachia* is a microbial endosymbiont that lives inside insect cells. CI is conditional insect sterility that operates as a toxin antidote (TA) system. Although, CI exhibits complex phenotypes not fully explained under a single discrete model. We instantiate in-silico genes that control CI, CI factors (*cifs*), as strings within the EA chromosome. We monitor the evolution of their enzymatic activity, binding, and cellular localization by applying selective pressure on their primary amino acid strings. Our model helps rationalize why two distinct mechanisms of CI induction might coexist in nature. We find that nuclear localization signals (NLS) and Type IV secretion system signals (T4SS) are of low complexity and evolve fast, whereas binding interactions have intermediate complexity, and enzymatic activity is the most complex. Our model predicts that as ancestral TA systems evolve into eukaryotic CI systems, the placement of NLS or T4SS signals can stochastically vary, imparting effects that might impact CI induction mechanics. Our model highlights how preconditions and sequence length can bias evolution of *cifs* toward one mechanism or another.

KEYWORDS

Wolbachia, cytoplasmic incompatibility, toxin antidote, evolutionary algorithm, reproductive parasitism, evolution, simulation, artificial intelligence

Introduction

TA systems typically involve two linked genes encoding a toxin and antidote (Yamaguchi et al., 2011). They skew Mendelian inheritance in their favor by addicting organisms to the presence of an antidote and killing offspring that do not inherit the TA module, via the toxin. Thus, they ensure inheritance in the next generation by post segregational killing. Ancestrally, TA systems might have arisen as selfish systems linked to the replication of prokaryotic plasmids (Rankin et al., 2012). How TA systems evolve is a chicken-egg paradox: a lone toxin is detrimental to host fitness and an antidote without a linked cognate toxin could be beneficial,

neutral, or detrimental, dependent on context. Prior models predict that TA systems assemble with selection on plasmids in contexts of genomic conflict or in situations where antidotes have beneficial functions in addition to toxin rescue (Rankin et al., 2012).

Wolbachia are bacteria that live inside insects (Hertig and Wolbach, 1924; Hertig, 1936; Werren et al., 2008). *Wolbachia* have the capability to conditionally sterilize mosquitos in a phenotype called CI (Laven, 1953, 1967a,b; Yen and Barr, 1971, 1973). CI is a unique biological instantiation of a TA system. The CI phenotype is useful to applied entomology (Laven, 1967b; Xi et al., 2005; Zheng et al., 2019). CI is currently applied as a biocontrol mechanism preventing the transmission of mosquito borne diseases across the world and on multiple continents in various applications. Mosquitos infected with *Wolbachia* exhibit reduced ability to transmit flaviviruses like Dengue and Zika (Teixeira et al., 2008; Ye et al., 2015). Ongoing attempts use the selective pressure of *cifs* to spread beneficial (probiotic) *Wolbachia* infections into wild mosquito populations to limit disease (Ross et al., 2022). At the molecular level, the beneficial spread of *Wolbachia* is linked to the function of *cif* TA genes.

Much of the evolutionary dynamics of CI has been well described at the population level. CI is common because it increases equilibrium frequencies and infection persistence, thereby increasing the chances of *Wolbachia* being transferred to new species hosts (Turelli et al., 2022). Yet in the insect, selection does not act to preserve or increase CI rates (Turelli, 1994). Importantly, evolutionary dynamics and selective pressures operating at the lowest molecular level and at the moment CI emerged in evolutionary history have never been described.

The genes that control this conditional sterility are two linked genes dubbed *cifs* that form complex TA systems (Beckmann and Fallon, 2013; Beckmann et al., 2017, 2019b). Uniquely, this TA system is viral, bacterial, and eukaryotic because it is encoded within/near WO-phages whose genomes reside in intracellular bacterial endosymbionts, which reside in insect hosts. The TA systems express extended phenotypes impacting the eukaryotic insect host. *Cifs* are uniquely positioned in that their evolutionary origin necessitates a functional jump from bacteria to eukaryotes. The *cif* TA system encodes a sperm delivered embryo killer toxin and a cognate rescuing antidote. If the insect host loses *Wolbachia*, remaining toxin sterilizes males, and these populations do not reproduce. Therefore, female insects keep *Wolbachia* because the antidotes are useful in the presence of toxins encountered in male sperm. Importantly, purifying selection does preserve the *cif* antidotes (Merçot and Poinsot, 1998; Meany et al., 2019; Driscoll et al., 2020); and on lower levels in the context of genomic conflict, selection can act to assemble the biochemical domains of toxin antidote systems (Rankin et al., 2012). Perhaps this is how the first CI system assembled. Though once assembled, selection on the insect level does not act to preserve the bacterial toxins which tend to pseudogenes and/or are replaced by subsequent invading *cif* systems (Martinez et al., 2020; Beckmann et al., 2021).

While molecular details on CI function are emerging, one problem is that rules governing induction of sterility via the *Wolbachia* TA system are debated. In general, the system behaves as a classical TA module, meaning one gene named *cifB* is inducer and its cognate partner *cifA* acts as antidote (Beckmann et al., 2019b). However, there remains unresolved nuance in the mechanism. Currently all data support the hypothesis that the first operon gene, *cifA*, is antidote (Beckmann et al., 2017; Shropshire et al., 2018). However, induction

of CI and the exact source of the toxicity appears more, or less, complex in various models. The two main models each have empirical evidence to support them. These models are the TA model (Poinsot et al., 2003; Beckmann et al., 2019a,b) and the 2 × 1 model (Shropshire et al., 2019; Shropshire and Bordenstein, 2019). The TA model is more parsimonious and significant evidence supports it in fruit flies, mosquitos, yeast models, and structural studies (Beckmann et al., 2017, 2019c; Bonneau et al., 2019; Adams et al., 2021; Xiao et al., 2021; Horard et al., 2022; Sun et al., 2022). In contrast, the 2×1 model posits that a single gene acts as rescue factor, but induction of sterility requires both *cifA* and *cifB* genes (LePage et al., 2017; Shropshire and Bordenstein, 2019).

Our hypothesis is that both models coexist in nature as alternate variations of the broader TA theme. These variations might arise as CI evolves from a simple prokaryotic TA module into a eukaryotic CI system (see Figures 1A,B). To explain, induction of sperm sterility in a eukaryote via a prokaryotic TA module necessitates the evolution of additional functions beyond toxin and antidote. In support of this hypothesis, prior models predicted that beneficial functions in addition to antidote functionality are prerequisites for TA module emergence (Rankin et al., 2012). In our case, *Wolbachia* must first secrete the toxin out one of its Sec-independent secretion systems; for the remainder of this study, we implicate the Rickettsiales *vir* homolog (*rvh*) type IV secretion system (T4SS) for CI protein secretion. T4SS substrates require a signal sequence, usually found at the C-terminus. Once secreted, the toxin must localize into the nucleus via a nuclear localization signal (NLS). Thus in simple terms, for a CI system to evolve requires additions of secretion signals and nuclear localization signals. In TA systems, the two proteins bind each other. There is a likely possibility that binding of *cifA* to *cifB* occurs prior to secretion and thus one protein might drag the other through a given secretion system. Under this hypothesis, it is possible that T4SS and NLS sequences could evolve in either antidote or toxin genes in different insect hosts. If the *cifA* antidote acquires an NLS and T4SS signal but *cifB* has neither, this leads to additional complexity in the system necessitating cooperative induction of sterility by *cifA* and *cifB* (hence a 2×1). While most empirical work evidences a strict TA in four known orthologs (cid^{wPip}, cid^{wHa}, cin^{wNo}, and cin^{Ott}), there is indication of 2×1 in two systems (cid^{wMel} and cin^{wPip}). Our research did not focus on determining if one model was correct at the complete expense of the other, but rather seeks to understand evolutionary pressures and selective mechanisms that might bias evolution of one model over another. Understanding the precise molecular mechanisms underlying the *cif* TA system and its evolution contributes information to "fine-tune" *Wolbachia* based biocontrol. Once we have perfect knowledge for how the *cif* TA sterility is induced, we can design the most efficient and parsimonious transgene insertions to reconstruct sterility in transgenic mosquitos as a biotechnological tool (i.e., with 2 genes or 1).

It was our goal to gain insights on the molecular evolution of CI by modeling CI's emergence with an evolutionary algorithm. Using EAs to model natural evolution has been a productive application (Lenski et al., 2003; Messer, 2013; Haller and Messer, 2016; Haller and Messer, 2019). Modeling gene drives in mosquitos with EAs, machine learning, and computer simulation has provided insights that predicted efficacy of actual biocontrol tools (Champer et al., 2018, 2022; Li and Champer, 2023). Biological evolution can be modeled by EAs at different ecological levels. Various abstractions and assumptions are made by any given model. EAs are typically top-down ecological
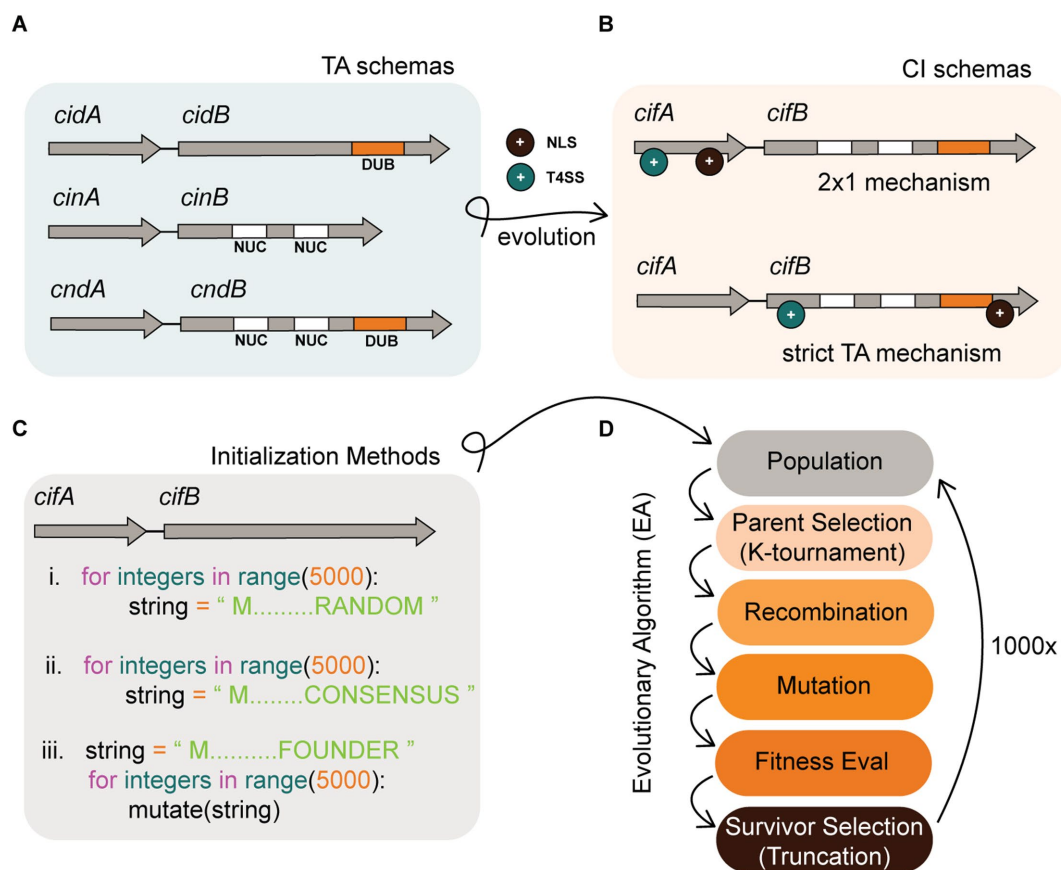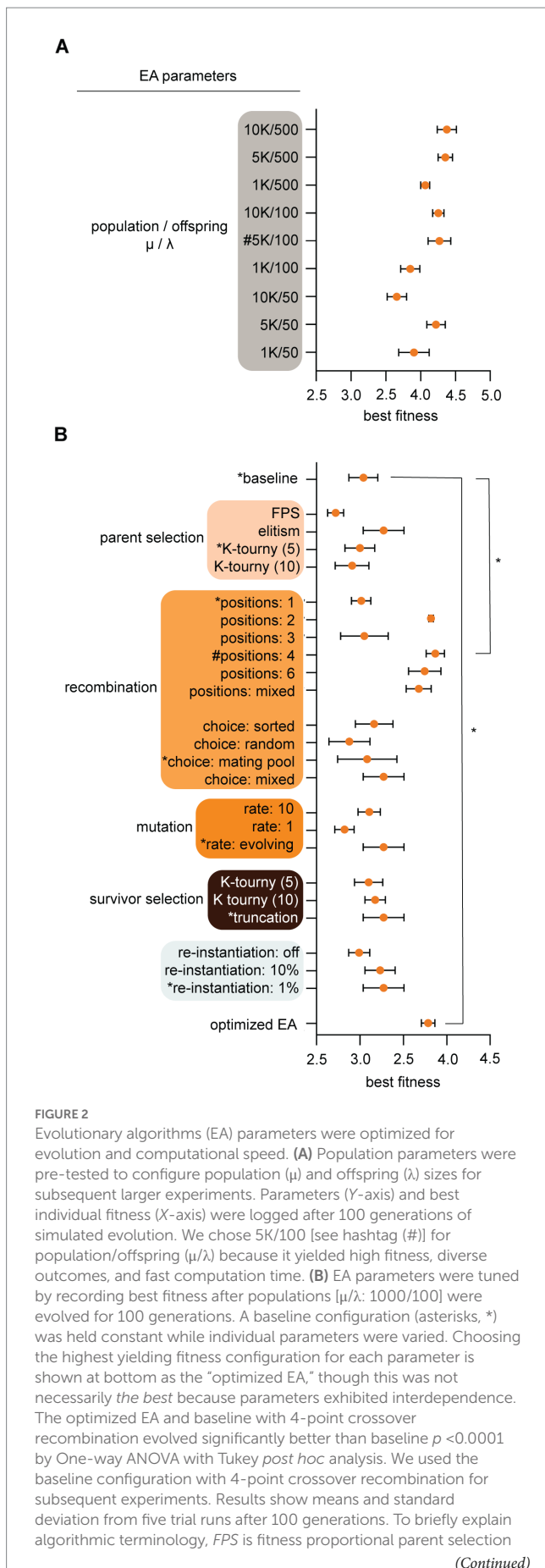
FIGURE 1

Background infographic. **(A)** Schemas of Wolbachia TA modules in a more ancestral prokaryotic form. To evolve into CI systems, ancestral TA modules must add more complex features including an NLS (+ black circle) and a T4SS (+ cyan circle). **(B)** CI system schemas might evolve into two descriptive models which include the 2 × 1 and strict TA model. The location where NLS or T4SS features evolve could impact the mechanistic induction of CI. A CI schema where both NLS and T4SS features co-occur in cifA alone is predicted to require both cifA and cifB for induction. In contrast, if these features co-occur in cifB, then cifB would be sufficient for induction of CI and behave as a strict TA module. **(C)** *In silico* simulation of this evolution requires an initial instantiation of a population of TA strings. Our experiments tested three distinct methods of instantiation that include (i) instantiating random strings, (ii) instantiating semi-random strings comporting to conserved cif consensus sequence, and (iii) instantiating a single individual and deriving an entire population by mutagenesis of that founder. **(D)** After instantiation of the population, it evolves under the selective pressure of a fitness function and follows discrete generations. Our algorithm selects parents by *K*-tournament and distributes these individuals into a mating pool. Offspring are generated by recombination of parents wherein two strings swap discrete sub-strings to create a new child. After recombination, child strings are mutated. Fitness of the TA is then evaluated, and survivors are selected based upon truncation survivor selection. In truncation, the population is sorted and the lowest fitness individuals that fall below a threshold are culled such that population numbers remain at the carrying capacity. The algorithm terminates after 1,000× generations.

models modeling populations of organisms. Top down EAs model gene flow of beneficial or deleterious traits. Within populations, each organism can be assigned a fitness value. Organisms and their genes can then mate, recombine, mutate, and die. These EA implementations tend toward Wright-Fisher models and often obey set rules. These models are useful for questions on evolutionary theory and adeptly model gene drives and selective sweeps, etc. However, the actual coded implementations are often abstract and difficult to translate into the evolution of amino acid sequence.

Popular bottom-up EA frameworks that modeled evolution upwards toward complexity are also abstract because they implemented computer assembly functions like "push" and "pop" as analogies of protein and metabolic pathways (Lenski et al., 2003; Adami and LaBar, 2015). These studies have demonstrated that in bottom-up simulations, simple functions can give rise to more complex functions (like add and multiply) through evolution;

however, these are abstract analogies, not actual DNA code. There is a gap in implementations of bottom-up biological models. Bottom-up implementations could implement DNA code as the starting point and model how code changes. A bottom-up implementation should instantiate the lowest levels of selection on actual genes (Dawkins, 1976) and test the lowest level of function which is protein translations of that code. The *in silico* genes could be mutated and recombined as actual DNA molecules and fitness can determined by bioinformatic algorithms comparing string sequence similarity to proteins of known function. Our coded framework presented herein is novel in this respect.

Evolutionary algorithms are perfect for studying protein string evolution because the search space of protein strings is vast (considering 20 possible amino acids and strings in lengths of thousands $= 20^{1000}$ unique strings). Research implementing codon based EAs is in early stages (Loose et al., 2006; Wnętrzak et al., 2018;

Yoshida et al., 2018; Boone et al., 2021). For example, a few studies tested machine learning guided mutations and used EAs to design novel antimicrobial peptides (AMPs) (Loose et al., 2006; Yoshida et al., 2018; Boone et al., 2021). These researchers guided evolution from known AMP strings rather than evolving novel de-novo protein strings. These studies provide some support for the concept of using sequence similarity as a proxy for fitness. Here we use sequence similarity to *cif* consensus sequences as fitness proxy to process simulations orders of magnitude faster than possible with bioassay. To gain a better understanding of *cif* evolution, we encoded *cif* TA genes directly as chromosomes within a population based EA and observed the evolution of their strings (see Figures 1C,D).

Materials and methods

EA design

An overlapping generations ($\mu + \lambda$) EA was coded in Python 3[1] where population size (μ) was 5,000 individuals and offspring (λ) was 100. Other variations were tested (see Figure 2A). Within the EA, code classes included an EA class (running the EA simulation functions, main methods, and data logging capacities), a TA class (housing the chromosome instantiations), and a main driver. The driver receives input from an editable JSON configuration file. All configuration files and outputs were saved and stored for reproducibility. The random seed is configurable for reproducibility.

EA class

EA algorithms are stochastic in nature. Evolutionary trajectories can proceed down different routes or converge. Thus, our main EA experiments consisted of 30 runs each (Figures 3, 4). Main methods within the EA class included class resets (to reset logs and class variables after each run); methods for population instantiation. In-Silico simulation of this evolution requires an initial instantiation

FIGURE 2

Evolutionary algorithms (EA) parameters were optimized for evolution and computational speed. **(A)** Population parameters were pre-tested to configure population (μ) and offspring (λ) sizes for subsequent larger experiments. Parameters (*Y*-axis) and best individual fitness (*X*-axis) were logged after 100 generations of simulated evolution. We chose 5K/100 [see hashtag (#)] for population/offspring (μ/λ) because it yielded high fitness, diverse outcomes, and fast computation time. **(B)** EA parameters were tuned by recording best fitness after populations [μ/λ: 1000/100] were evolved for 100 generations. A baseline configuration (asterisks, *) was held constant while individual parameters were varied. Choosing the highest yielding fitness configuration for each parameter is shown at bottom as the "optimized EA," though this was not necessarily *the best* because parameters exhibited interdependence. The optimized EA and baseline with 4-point crossover recombination evolved significantly better than baseline *p* <0.0001 by One-way ANOVA with Tukey *post hoc* analysis. We used the baseline configuration with 4-point crossover recombination for subsequent experiments. Results show means and standard deviation from five trial runs after 100 generations. To briefly explain algorithmic terminology, *FPS* is fitness proportional parent selection

(Continued)

---

1   https://www.python.org/

FIGURE 3

Output data from three large evolution experiments. Columns (i-iii) show results from three different instantiation methods described in Figure 1. Rows show average fitness of all TAs within a population versus generations **(A,F,K)**. Average sub-feature fitness of all TAs within a population versus generations **(B,G,L)**; Diversity index versus generations, where lower numbers indicate more similarity in string sequence and therefore loss of diversity **(C,H,M)**; Average T4SS site location of the population (black line) versus generations **(D,I,N)**; Average NLS site location (black line) of the population versus generations **(E,J,O)**. Scoring for T4SS and NLS sites is as follows: a score of 0 indicates that the site evolved in the antidote gene (*cifA*) and a 1 indicates that the site evolved in the toxin gene (*cifB*); therefore a score of 0.5 means that half the population had the site in *cifA* and the other half had the site in *cifB*. Mean values are plotted with black or colored lines. Standard deviation is marked in gray lines. Bias above 0.5 indicates preferential evolution of TA and below 0.5, 2×1. Panels **(D,I,N)** are all significantly different from each other at termination, (*p* <0.05) by one-way ANOVA with Tukey *post hoc* analysis. Panels **(E,O)** were significantly different by the same.

FIGURE 4
Parameters that control bias in the location of NLS and T4SS signals were tested. **(A)** We were able to control starting diversity by changing population size (μ). Small populations of 1,000 had significantly less diversity than populations of 20,000 by Mann–Whitney U test. In turn, these changes significantly altered the bias of the signal during the course of evolution, but not the final result [see dotted line vs. solid line in **(B,C)**; T4SS location and NLS location respectively]. The most significant terminal impact on bias of signal location was when we increased the average length of the toxins (see dashed line in **B,C**). Panels **(B,C)** are T4SS and NLS sites is as above. Mean values are plotted with black or colored lines. In **(B)** terminal conditions of doubling toxin length to μ=20k were significantly different for T4SS signal locations (p <0.05) by unpaired t-test with Welch's correction. These experiments all were performed under the third [(iii) founder] method of instantiation.

of a population of TA strings. Our experiments tested three distinct methods of instantiation that include (i) instantiating random strings, (ii) instantiating semi-random strings comporting to conserved *cif* consensus sequence, and (iii) instantiating a single individual and deriving an entire population by mutagenesis of that founder; sorting functions that sorted TA populations based on fitness; parent selection methods (which finally used K-tournament of $K = 5$ after preliminary testing; see Figure 2B). *K*-tournament selection runs multiple fitness tournaments among a few individuals chosen randomly from the population. Winners of a tournament with the best fitness are sent to the mating pool array to be selected for recombination. In experiments recombination used 4-point crossover recombination, but we tested other modes (Figure 2B). Mutation methods utilized an algorithm that randomly locates DNA base pairs and flips to a random choice of A, T, G, or C. Mutation also encoded insertion and deletion functions with randomly sized indels. The mutation method evaluates fitness by calling the fitness evaluation method from the TA class (see below). After fitness evaluation, survivors were selected via truncation

survivor selection. Other survivor selection regimes were tested (Figure 2B). Truncation sorts the population and culls the lowest fitness individuals in a number equivalent to the number of offspring added per generation. Thus, carrying capacity remains constant at $\mu = 5,000$. Data logging functions were encoded, for example, calculateAverageFitness(), which tallies an average TA fitness. A termination condition method was coded but not used in final experiments. Logs were recorded in output files and saved. We tracked 15 quantifiable observations: (1) highestTAFitness_HTF, (2) avgBindingFitness_ABF, (3) avgDUBFitness_ADF, (4) avgNucFitness_ANF, (5) avgTAfitness_ATF, (6) avgToxinLength_ATL, (7) avgToxinAALength_ATAL, (8) avgAntidoteLength_AAL, (9) avgAntidoteAALength_AAAL, (10) avgTAMutationRate_ATMR, (11) avgNLSSITELocation_NLSL, (12) avgTypeIVSITELocation_TYPL, (13) avgNLSFitness_ANLSF, (14) avgT4SSFitness_AT4F, (15) diversityIndex_DI.

TA class

TA class individuals were instantiated with chromosomes encoding the string toxin and string antidote in DNA code. TAs additionally hold class variables including a nuclease score (measuring how well the toxin schema matches a known *cin* toxin consensus sequence) and a deubiquitylating (DUB) score (measuring how well the toxin schema matches a known *cid* toxin consensus sequence) (Gillespie et al., 2018). They also hold a NLS score which is determined by presence or absence of a "KRAR" string (Rossi et al., 1993) and a T4SS score determined by presence or absence of a "R-X(7)-R-X-R-X-R" string (Vergunst et al., 2005). All functional domains including nuclease domain, deubiquitylating domain, NLS, and T4SS signals are detected through a pairwise alignment algorithm and can be given partial scores if parts of the sequence are present. Pairwise alignment is built into the EA by importation of the Biopython[2] module's pairwise2 method. "Biopython is a set of freely available tools for biological computation written in Python by an international team of developers." The pairwise2 method is called with a −1 gap penalty, a −0.1 gap extension penalty, and a false condition so that end gaps are not penalized. A binding score (measuring how well the pair bind each other) is determined by our own algorithm. This algorithm is based on a sliding window that slides two strings together in comparison to find and tally a score of the best matching residue configurations. Precisely 11 charged residues are known to underlie *cifA* and *cifB* binding (Xiao et al., 2021). Therefore, if a sliding window detects an alignment of K with D, a score would be increased by 1 and the process continues. Repelling charges are penalized by −1. A total matched binding sequence should not exceed 11 binding residues in accordance with crystal structure data (Xiao et al., 2021). Class methods within the TA class include standard "getters" and "setters" [i.e., setSchemata() which instantiates the toxin strings], a translation method that translates the DNA code into proteins, a coded number parser to facilitate binding evaluations with integers rather than strings (to speed up computation), sub component fitness evaluation methods, and a "to string" reporting method.

---

2   https://biopython.org/

Main class

The main driver simply imports and stores the JSON configuration files. It instantiates the EA class. Finally, it initiates the simulation.

Calculating fitness

The fitness function for an individual TA pair is defined as the sum of its binding score, nuclease score, deubiquitylating score, NLS score, and T4SS signal score. Each sub-component fitness can maximally be 1 and therefore the max fitness of a perfect TA is 5. To elaborate precisely on how sequence similarity is used as a proxy for fitness, we describe the situation for DUB fitness. The DUB domain is a catalytic sequence of amino acids that conforms to a schema. The DUB schema in *cifs* is precisely, "HWVTLVI---------YY-DSL--------I---L-----D---------QQ-DG---CG----EN," where dashes (−) are interchangeable spaces (*do not cares*) and letters are requirements of specific amino acids in specific positions. A perfect alignment score of 1 for a *cif* DUB would match this schema. Anything not conforming to the schema is penalized by the alignment algorithm for gaps and mismatches. The schema for the nuclease domain is as follows, "DL-LL-R---------PIIIELK--------------------DLVL---------PIGLELK." These two consensus schemas were originally derived directly from compilation of diverse CI and CI-like toxins (Gillespie et al., 2018). Schemas for NLS and T4SS signals are also pulled from literature and listed on the preceding TA Class description. Thus, by using sequence similarity to conserved schemas and the binding algorithm (described above) we can sum elements for a perfect TA fitness score of 5. Parsimony pressure is applied if a TA genome exceeds a threshold of 4,500 DNA base pairs (this is an estimate of average *cif* TA size) and pressure increases corresponding to the length of the additional extraneous code. Parsimony pressure thus acts to minimize the coding length of TA pairs and accurately reflects selective pressures inducing reduction of *Wolbachia* genomes. *In toto*, a final fitness score involves the sum of the five functional component scores with a penalty function subtracting a coefficient parsimony penalty based on sequence length. All code is publicly available for inspection and reuse on github.

Experimental setup

The EA evolves populations of TAs and evaluates their fitness. Simulations were initialized via three distinct methods described in Figure 1C. How the simulation is initiated impacts the levels of inherent diversity in the starter population. Methods i–iii decrease in starting diversity from most to least, respectively. After 1,000 generations the simulation is terminated, and data collected. Data collected is given above and was graphed in Graphpad's Prism software. Experiments were conducted with 30 runs each.

Statistical analysis

For experiments generating multiple comparisons like optimizing the EA (Figures 2, 3) we employed one-way ANOVA with Tukey *post*

hoc analysis using Graphpad Prism software. We compared values present at the final generation at termination of the simulation. *p*-values were considered significant if less than the standard 0.05. In Figure 4, terminal data were compared using unpaired two-tailed *t*-test with Welch's correction.

Results and discussion

Validating and tuning the EA's fitness function

One prerequisite of implementing an EA is an ability to evaluate fitness of individuals. A protein's function and thus its fitness is encoded in primary structure (amino acid strings). Protein function can be predicted by comparing strings to others with known function. Therefore, we use sequence similarity to *cif* domains as a proxy of fitness and thereby apply selective pressure. In our EA an individual in-silico *cif* is constituted of the two DNA genes and their translated protein strings. Many individual TA pairs are instantiated within populations. The EA mutates and recombines them exactly as DNA can mutate and recombine. Fitness of individual TA pairs is modelled as a sum of (1) how well a toxin can kill a cell (based on sequence similarity to known killer toxin domains from *cins* and *cids*) and (2) how well the antidote binds its partner toxin (modeled as matching charged residues within cognate TA pairs). Additionally, we add (3) NLS and (4) T4SS signal domains as additional summed components of fitness. We then quantified where NLS and T4SS signals evolved during simulations (in *cifA* or *cifB*) and tracked biased emergence of 2×1 versus TA.

After initial design (Figure 1), coding, and parameter optimization (Figure 2), we determined that the EA evolved efficiently and observed that population sizes of 5,000 individuals with offspring sizes of 100 individuals were optimal because they yielded high fitness, diverse outcomes, and fast computation time (Figure 2A). These assumptions have flaws (for example nature is not an algorithm that optimizes parameters to speed up evolution; discussed below), but these settings served as a starting point. Next, we tested different algorithmic methodologies for parent selection, recombination, mutation, survivor selection, and a "re-instantiation" method immigrating 10, 1%, or 0% de-novo individuals (described in methods). Results show means and standard deviation from five trial runs after 100 generations. Our goal was to determine optimal algorithms for maximizing *cif* fitness within simulation time periods. After observing EA behavior, we determined to use a "baseline" configuration of K-tournament selection where $K = 5$ for parent selection. Selected parents are transitioned to the mating pool where mating only occurs between individuals within that selective sub-population. Mating of TA parents is implemented with 4-point crossover recombination with a self-adaptive mutation rate to generate offspring TAs. The self-adaptive mutation rate is encoded within an individual's chromosome and can change if higher or lower mutation rates contribute to better fitness. Offspring TAs are loaded back into the main population ($\mu + \lambda$) and compete for survival via truncation, which culls the lowest fitness individuals. Subsequent experiments used these conditions unless otherwise specified.

Tracking evolution of cif domains shows that NLS and T4SS signals are quick to evolve

We performed three large experiments based upon three methods of instantiating populations. Our intention was to determine if starting preconditions biased preferential evolution of NLS or T4SS signals in *cifA* versus *cifB*. Any bias might indicate conditions under which 2×1 or strict TA mechanisms would arise from the evolutionary process. In these first tests, the EA successfully evolved and evaluated the fitness of TA modules. In all experiments fitness of individuals gradually increased toward 5 (Figures 3A,F,K). The rising fitness values over the course of the simulation indicates that our code is selecting for progressive *cif* assembly. All three simulations show a start at low average TA fitness which improves as more successful TAs evolve and overtake the population.

We tracked each sub-component of fitness including nuclease, DUB, NLS, and T4SS signal evolution (Figures 3B,G,L). As each component assembles in the evolving *cifs*, their corresponding fitness values move upwards to 1. The speed at which each domain reaches 1 indicates how difficult it is to evolve. These data also indicate our code works correctly as fitness of each sub-component increases with each generation towards a max score of 1. Importantly these data also indicate the inherent complexity of each sub-component and clearly show that NLS and T4SS signals are relatively quick to evolve in simulations (Figure 3; green and orange lines respectively). Binding is of intermediate complexity and arises slower (Figure 3; yellow lines). Nuclease and DUB catalytic domains are slow to evolve and do not completely reach perfect consensus sequences within the timeline of the evolutionary experiment (Figure 3; black and purple lines respectively). These data are in concordance with the given complexity of the domains. For example, the NLS is only 4 residues ("KRAR") whereas max binding fitness requires 11 matching residues in both toxin and antidote, and consensus sequences of catalytic domains must match 23 conserved residues within their schemas.

The five components' relative evolvability (or inherent speed of their evolution) indicates that CI systems might frequently lose, replace, adapt, and move NLS and T4SS signals, whereas binding and catalytic domains are more likely to remain conserved in-place due to difficulty of evolving them in the first place. If they are destroyed, they cannot quickly be replaced, whereas NLS and T4SS signals might be "fungible." We note that a full spectrum of T4SS signals has yet to be identified and this simulation only implements one example (Atmakuri et al., 2003; Nagai et al., 2005; Schulein et al., 2005; Vergunst et al., 2005; Huang et al., 2011).

To facilitate data interpretation, we point out that Figures 3A–C, F–H,K–M are simply controls that demonstrate that the simulations are programmed correctly. The results leading to conclusions about 2×1 vs. TA systems are contained in Figures 3D,E,I,J,N,O. These data plot the proportion of T4SS signals found in cifA/cifB (Figures 3D,I,N) and the proportion of NLS signals found in cifA/cifB (Figures 3E,J,O) for the populations under the three different simulations. In the plots, a score of 0 (below the midline and into the blue) indicates that the site evolved in the antidote gene (*cifA*) and implies 2×1 function has evolved whereas a score of 1 (above the midline and into the salmon) indicates that the site evolved in the toxin gene (*cifB*) implying a TA function has evolved; a score of 0.5 means that half the population had these sites in *cifA* and the other half had the site in *cifB*.

In many simulations, the algorithm terminates with both (2×1 and TA) models co-existing. These data indicate that both strict TA and 2×1 systems could co-exist and might even inter-convert between mechanisms on evolutionary time scales with drift, mutation, and recombination. These in-silico observations are congruent with empirical literature demonstrating both systems are apparently extant (Beckmann et al., 2017; LePage et al., 2017; Shropshire et al., 2018; Shropshire and Bordenstein, 2019; Adams et al., 2021; Sun et al., 2022). Cautiously, we note that these observations are premised on assumptions that there must be some conditions selecting for the evolution of CI; these ecological conditions are not yet completely defined (Turelli, 1994; Martinez et al., 2020; Beckmann et al., 2021) yet must exist under some context that gives rise to CI and *cifs*; perhaps amongst discrete spatial limitations and genomic competitions (Rankin et al., 2012). Importantly, our model simply justifies how multiple CI mechanisms might evolve to coexist on the amino acid level.

Parameters of simulations bias evolution of TA versus 2×1

When we measured where NLS and T4SS signals evolved (in *cifA* or *cifB*) under three different starting conditions [method (i) random, (ii) consensus, and (iii) founder; see Figures 1C,D] we detected biases in the evolutionary trajectory of one model over another (Figures 3D,I, N,E,J,O). After random instantiation (method i.) both NLS and T4SS signals' scores were slightly less than 0.5 indicating a slight preference for evolution of those sequences in *cifA* genes (Figures 3D,E). After semi-random instantiation (method ii.) there was strong bias to evolve the T4SS within *cifA* genes indicating a bias toward 2×1 (Figure 3I). Only in the third method did both NLS and T4SS signals preferentially evolve in the *cifB* gene, thereby indicating bias toward TA mechanisms (Figures 3N,O). Each method showed statistically different termination conditions for T4SS locations with all *p*-values <0.05. Method i significantly differed from method iii with respect to termination condition of NLS signal. Importantly, these results indicate that our model can detect significant evolutionary bias toward one mechanism over another and that preconditions at the start of evolution can bias the evolutionary trajectory toward either 2×1 or TA mechanisms.

We next sought to understand the conditions that drove biased evolution of one mechanism over another. Interestingly, some Culex mosquito populations maintain Wolbachia populations that contain multiple diversifying *cid* systems (Altinli et al., 2018; Bonneau et al., 2019). To monitor *cif* genetic diversity within the populations we tracked a diversity index, which was determined by randomly sampling 10 toxins from the population each generation and calculating the average similarity of those 10 toxins' amino acid strings. In populations where individual TAs fix and overtake the population, diversity decreases to zero (Figures 3C,H,M). After 1,000 generations, most populations are overtaken by one or a few TAs of high fitness. In method (iii), which resulted in biased evolution of strict TA systems, the diversity index was lowest (see Figure 3M). We tested whether diversity directly drove bias by altering the relative levels of genetic diversity within the population. We controlled this by simply changing population size (μ). Smaller populations carried less diversity (Figure 4A). The relative diversity did alter the course and

path of evolution, but not the outcome, which always converged on TA mechanisms (Figures 4B,C). Thus, population diversity did not drive the bias toward either TA or 2×1. We next tested whether length of the toxin protein impacted the outcome. When we doubled the size of the average size of the toxins, we significantly raised the bias of the model toward the TA model (Figure 4). We discuss the theoretical impacts of these observations below. Importantly, these observations demonstrate that we have successfully encoded an EA that evolves and tracks *cif* amino acid evolution.

To elaborate on the question of which instantiation method is more biologically realistic, we suggest the following thoughts. One assumption is that a prokaryotic TA model preceded the evolution of CI. There is evidence for this in the fact that *cif* operons have been observed within plasmids of *Rickettsia* (a sister lineage of *Wolbachia*), lending plausibility to the hypothesis that CI emerged from an ancestral prokaryotic TA plasmid selection system (Gillespie et al., 2018). If this hypothesis is correct, method (i) random instantiation, is not biologically relevant as it begins selection on all four components simultaneously from completely random sequences. In contrast, method (ii) assumes a prokaryotic TA system already exists and reasonably comports with some known toxin consensus sequences then selects for the addition of NLS and T4SS signals in the jump from prokaryotic TA to eukaryotic CI. This method is biologically relevant only for the first emergence of CI's evolution in deep evolutionary history. In that situation and according to the data observed here, this model shows preference for the evolution of 2×1 systems, but not to the complete exclusion of TA systems. The third instantiation method which started a population based on an individual founding sequence, method (iii), exhibited strong bias of NLS and T4SS assembly in *cifB* genes, indicating strong preference for a strict TA functionality. Notably, this model had the least diversity within its population and likely reflects more accurately the actual evolution in *Wolbachia* systems where an insect is colonized by a founder strain and diversity is only encountered in sporadic co-infections that only occur rarely in evolutionary history, but are likely the source for CI gene evolution if phages exchange genes during coinfection. Therefore, our analysis can explain the observed bias in favor of strict TA functionality by about 60% of studied *cif* orthologs; notably in our simulation method (iii) the bias was also about 60%. To be cautious, however, we note that only ~6 ortholog TA pairs have been studied in detail and it remains to be seen whether the observed frequency of TA or 2×1 functionalities is some relic of sample bias. After emergence of TAs, our model's data predicts they flux periodically from 2×1 to strict TA. Future studies can utilize this framework to determine more conditions that give rise to 2×1 versus TA systems.

Increasing length of toxin biases evolution toward TA mechanisms

Instantiation method (iii), where a population is generated by a founder, more accurately reflect the day-to-day evolution of *Wolbachia* organisms in their hosts. In each insect, the *Wolbachia* encountered will be entirely derived from the ancestor of that infection and therefore recombination with sequences of radically different *cifs* is unlikely, though not impossible due to mobility of WO phage viruses and infrequent co-infections. Method (iii) most accurately reflects these conditions and in this model, there was strong bias toward the

evolution of strict TA functionality (Figure 3). This suggests that over time, most (~3/4) CI systems should end up in a state of strict TA functionality with some variation induced by ongoing flux of NLS and T4SS signals. One of the key factors seemingly controlling this evolution is simply the length of the corresponding antidote and toxin (Figure 4). Because the NLS and T4SS signals are of low complexity and evolve quickly, they should stochastically arise more often in what is the longer gene of the pair. Of all syntenic *cif* operons, the length of the toxin is always longer than the length of the antidote. This also indicates a simple bias toward strict TA if NLS and T4SS signals simply drift into the larger ORF. Biology is complex, yet factors having the biggest role in these mechanistic biases might be as simple as gene length. However, sequence length does not explain everything about the model. In data from method (ii), where the strongest bias toward 2×1 was observed, the toxins and antidotes on average are the same size. Therefore, size does not account for all the forces driving bias in either direction.

Conclusion, future directions, and limitations

The hypotheses and take-homes from our model are thus: (1) CI might evolve from less complex prokaryotic TA systems (Figure 1). (2) TA systems can convert to CI systems by the addition of at minimum NLS and T4SS signals (Figure 1) though these domains may not be completely sufficient. (3) Where NLS and T4SS signals evolve (in *cifA* or *cifB*) is predicted to be the determinant of 2×1 or strict TA mechanics (Figure 1). (4) In cases where CI and *cifs* diverge from a single founder, method (iii), the evolution is biased toward strict TA systems, but not at full exclusion of 2×1 systems (Figures 3N,O). (5) In our model, sequence length can predispose bias of signal evolution in a location; simply meaning that if B genes are longer than A genes, it is more likely that NLS and T4SS signals will arise inside them first. Finally, (6) Codon-based EAs can be applied in a bottom-up approach to address questions related to the evolution of protein strings.

In future experiments we plan to utilize this framework to test additional sequences of NLS and T4SS signals. Importantly, the signals we used are not the only ones that exist in nature. There can be cryptic and/or bi-partite combinatorial sequence motifs that contribute to secretion and localization (Schulein et al., 2005). To add to the nuance, our algorithm does not account for redundant sequences. For example, it does not quantify if additional NLS or T4SS signals evolve elsewhere, beyond the first. It would be interesting to re-program the system to measure and tally if multiple NLS and T4SS signals are evolving and where they are. One prediction our model makes is that because NLS and T4SS signals are of low complexity, there may be multiple redundant signals within the same gene. In future experiments we will look for this.

Our model makes many assumptions. One assumption we made, to begin analysis somewhere, is that the parameters causing the fastest evolution of *cifs* in simulations were apt to simulate the natural evolutionary dynamics of these TA modules. However, evolution within the natural organism might not be so ideal. Therefore, favoring the most efficient methodologies and parameters to evolve high fitness quickly might be incongruent with nature. Although we grounded the evolution of the EA in real biology using actual *cif*

sequences and known binding features. The benefit of our coded framework is that it can be modified to test and address future criticisms and hypotheses. For example, while we have only implemented the 2×1 and strict TA mechanisms, if ever a possibility of a third mechanism is observed or postulated, we can add that possibility to the code base.

Finally, our model encodes and models evolution of the most primal or basal level of CI (the amino acids). It is not an ecological model assessing TA allele fixation in populations. It would be inappropriate to directly compare our model with prior ecological models (Turelli, 1994; Rankin et al., 2012); although our model could be imported into those models as a foundation. The natural evolution of selfish TA elements involves multiple levels of evolutionary dynamics. For example, *cif* systems exist within WO-phages that exist within *Wolbachia* bacteria that live within insect hosts that live within populations. *Cifs* impact evolution and population dynamics on all these levels. Future models might incorporate our codon-based EA as a subcomponent of a larger multi-competitive EA framework. Such a program might provide vast insights into the complex evolutionary dynamics inherent to *Wolbachia* biology and make predictions about actual CI gene function.

Data availability statement

The original contributions presented in the study are included in the article/supplementary material, further inquiries can be directed to the corresponding author.

Author contributions

JB conceived of, conducted the experiments, and wrote the first draft manuscript. DT conceived of experiments and edited the manuscript. JG edited the manuscript. All authors contributed to the article and approved the submitted version.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

References

Adami, C., and LaBar, T. (2015). From entropy to information: biased typewriters and the origin of life. *ArXiv*. Available at: https://arxiv.org/abs/1506.06988

Adams, K. L., Abernathy, D. G., Willett, B. C., Selland, E. K., Itoe, M. A., and Catteruccia, F. (2021). *Wolbachia* cifB induces cytoplasmic incompatibility in the malaria mosquito vector. *Nat. Microbiol.* 6, 1575–1582. doi: 10.1038/s41564-021-00998-6

Altinli, M., Gunay, F., Alten, B., Weill, M., and Sicard, M. (2018). *Wolbachia* diversity and cytoplasmic incompatibility patterns in *Culex pipiens* populations in Turkey. *Parasit. Vectors* 11:198. doi: 10.1186/s13071-018-2777-9

Atmakuri, K., Ding, Z., and Christie, P. J. (2003). VirE2, a type IV secretion substrate, interacts with the VirD4 transfer protein at cell poles of *Agrobacterium tumefaciens*. *Mol. Microbiol.* 49, 1699–1713. doi: 10.1046/j.1365-2958.2003.03669.x

Beckmann, J. F., Bonneau, M., Chen, H., Hochstrasser, M., Poinsot, D., Merçot, H., et al. (2019b). The toxin-antidote model of cytoplasmic incompatibility: genetics and evolutionary implications. *Trends Genet.* 35, 175–185. doi: 10.1016/j.tig. 2018.12.004

Beckmann, J. F., Bonneau, M., Chen, H., Hochstrasser, M., Poinsot, D., Merçot, H., et al. (2019a). Caution does not preclude predictive and testable models of cytoplasmic incompatibility: a reply to Shropshire et al. *Trends Genet.* 35, 399–400. doi: 10.1016/j. tig.2019.03.002

Beckmann, J. F., and Fallon, A. M. (2013). Detection of the *Wolbachia* protein WPIP0282 in mosquito spermathecae: implications for cytoplasmic incompatibility. *Insect Biochem. Mol. Biol.* 43, 867–878. doi: 10.1016/j.ibmb.2013.07.002

Beckmann, J. F., Ronau, J. A., and Hochstrasser, M. (2017). A *Wolbachia* deubiquitylating enzyme induces cytoplasmic incompatibility. *Nat. Microbiol.* 2:17007. doi: 10.1038/nmicrobiol.2017.7

Beckmann, J. F., Sharma, G. D., Mendez, L., Chen, H., and Hochstrasser, M. (2019c). The Wolbachia cytoplasmic incompatibility enzyme CidB targets nuclear import and protamine-histone exchange factors. *eLife* 8:e50026. doi: 10.7554/eLife. 50026

Beckmann, J. F., Van Vaerenberghe, K., Akwa, D. E., and Cooper, B. S. (2021). A single mutation weakens symbiont-induced reproductive manipulation through reductions in deubiquitylation efficiency. *Proc. Natl. Acad. Sci.* 118:e2113271118. doi: 10.1073/ pnas.2113271118

Bonneau, M., Caputo, B., Ligier, A., Caparros, R., Unal, S., Perriat-Sanguinet, M., et al. (2019). Variation in Wolbachia cidB gene, but not cidA, is associated with cytoplasmic incompatibility mod phenotype diversity in *Culex pipiens*. *Mol. Ecol.* 28, 4725–4736. doi: 10.1111/mec.15252

Boone, K., Wisdom, C., Camarda, K., Spencer, P., and Tamerler, C. (2021). Combining genetic algorithm with machine learning strategies for designing potent antimicrobial peptides. *BMC Bioinform.* 22:239. doi: 10.1186/s12859-021-04156-x

Champer, J., Liu, J., Oh, S. Y., Reeves, R., Luthra, A., Oakes, N., et al. (2018). Reducing resistance allele formation in CRISPR gene drive. *Proc. Natl. Acad. Sci.* 115, 5522–5527. doi: 10.1073/pnas.1720354115

Champer, S. E., Oakes, N., Sharma, R., García-Díaz, P., Champer, J., and Messer, P. W. (2022). Modeling CRISPR gene drives for suppression of invasive rodents using a supervised machine learning framework. *PLoS Comput. Biol.* 17:e1009660. doi: 10.1371/ journal.pcbi.1009660

Dawkins, R. (1976). *The selfish gene.* New York: Oxford University Press.

Driscoll, T. P., Verhoeve, V. I., Brockway, C., Shrewsberry, D. L., Plumer, M., Sevdalis, S. E., et al. (2020). Evolution of Wolbachia mutualism and reproductive

parasitism: insight from two novel strains that co-infect cat fleas. *PeerJ* 8:e10646. doi: 10.7717/peerj.10646

Gillespie, J. J., Driscoll, T. P., Verhoeve, V. I., Rahman, M. S., Macaluso, K. R., and Azad, A. F. (2018). A tangled web: origins of reproductive parasitism. *Genome Biol. Evol.* 10, 2292–2309. doi: 10.1093/gbe/evy159

Haller, B. C., and Messer, P. W. (2016). SLiM 2: flexible, interactive forward genetic simulations. *Mol. Biol. Evol.* 34, 230–240. doi: 10.1093/molbev/msw211

Haller, B. C., and Messer, P. W. (2019). SLiM 3: forward genetic simulations beyond the Wright–fisher model. *Mol. Biol. Evol.* 36, 632–637. doi: 10.1093/molbev/msy228

Hertig, M. (1936). The Rickettsia, *Wolbachia pipientis* (gen. Et sp.n.) and associated inclusions of the mosquito, *Culex pipiens. Parasitology* 28, 453–486. doi: 10.1017/S0031182000022666

Hertig, M., and Wolbach, S. B. (1924). Studies on Rickettsia-like micro-organisms in insects. *J Med Res* 44, 329–374.7.

Horard, B., Terretaz, K., Gosselin-Grenet, A. S., Sobry, H., Sicard, M., Landmann, F., et al. (2022). Paternal transmission of the *Wolbachia* CidB toxin underlies cytoplasmic incompatibility. *Curr. Biol.* 32, 1319–1331.e5. doi: 10.1016/j.cub.2022.01.052

Huang, L., Boyd, D., Amyot, W. M., Hempstead, A. D., Luo, Z. Q., O'Connor, T. J., et al. (2011). The E block motif is associated with *Legionella pneumophila* translocated substrates. *Cell. Microbiol.* 13, 227–245. doi: 10.1111/j.1462-5822.2010.01531.x

Laven, H. (1953). Reciprocally differentiable crossing of mosquitoes (Culicidae) and its significance for plasmatic heredity. *Z. Indukt. Abstamm. Vererbungsl.* 85, 118–136.

Laven, H. (1967a). *Chapter 7: speciation and evolution in Culex pipiens*, vol. 251 (Elselvier: Academic Press), 251–275.

Laven, H. (1967b). Eradication of *Culex Pipiens* Fatigans through cytoplasmic incompatibility. *Nature* 216, 383–384. doi: 10.1038/216383a0

Lenski, R. E., Ofria, C., Pennock, R. T., and Adami, C. (2003). The evolutionary origin of complex features. *Nature* 423, 139–144. doi: 10.1038/nature01568

LePage, D. P., Metcalf, J. A., Bordenstein, S. R., On, J., Perlmutter, J. I., Shropshire, J. D., et al. (2017). Prophage WO genes recapitulate and enhance Wolbachia-induced cytoplasmic incompatibility. *Nature* 543, 243–247. doi: 10.1038/nature21391

Li, J., and Champer, J. (2023). Harnessing Wolbachia cytoplasmic incompatibility alleles for confined gene drive: a modeling study. *PLoS Genet.* 19:e1010591. doi: 10.1371/journal.pgen.1010591

Loose, C., Jensen, K., Rigoutsos, I., and Stephanopoulos, G. (2006). A linguistic model for the rational design of antimicrobial peptides. *Nature* 443, 867–869. doi: 10.1038/nature05233

Martinez, J., Klasson, L., Welch, J. J., and Jiggins, F. M. (2020). Life and death of selfish genes: comparative genomics reveals the dynamic evolution of cytoplasmic incompatibility. *Mol. Biol. Evol.* 38, 2–15. doi: 10.1093/molbev/msaa209

Meany, M. K., Conner, W. R., Richter, S. V., Bailey, J. A., Turelli, M., and Cooper, B. S. (2019). Loss of cytoplasmic incompatibility and minimal fecundity effects explain relatively low Wolbachia frequencies in *Drosophila mauritiana. Evolution* 73, 1278–1295. doi: 10.1111/evo.13745

Merçot, H., and Poinsot, D. (1998). And discovered on Mount Kilimanjaro. *Nature* 391:853. doi: 10.1038/36021

Messer, P. W. (2013). SLiM: simulating evolution with selection and linkage. *Genetics* 194, 1037–1039. doi: 10.1534/genetics.113.152181

Nagai, H., Cambronne, E. D., Kagan, J. C., Amor, J. C., Kahn, R. A., and Roy, C. R. (2005). A C-terminal translocation signal required for dot/Icm-dependent delivery of the Legionella RalF protein to host cells. *Proc. Natl. Acad. Sci.* 102, 826–831. doi: 10.1073/pnas.0406239101

Poinsot, D., Charlat, S., and Mercot, H. (2003). On the mechanism of Wolbachia-induced cytoplasmic incompatibility: confronting the models with the facts. *BioEssays* 25, 259–265. doi: 10.1002/bies.10234

Rankin, D. J., Turner, L. A., Heinemann, J. A., and Brown, S. P. (2012). The coevolution of toxin and antitoxin genes drives the dynamics of bacterial addiction complexes and intragenomic conflict. *Proc. Biol. Sci.* 279, 3706–3715. doi: 10.1098/rspb.2012.0942

Ross, P. A., Robinson, K. L., Yang, Q., Callahan, A. G., Schmidt, T. L., Axford, J. K., et al. (2022). A decade of stability for wMel Wolbachia in natural *Aedes aegypti* populations. *PLoS Pathog.* 18:e1010256. doi: 10.1371/journal.ppat.1010256

Rossi, L., Hohn, B., and Tinland, B. (1993). The VirD2 protein of *Agrobacterium tumefaciens* carries nuclear localization signals important for transfer of T-DNA to plant. *Mol Gen Genet* 239, 345–353. doi: 10.1007/bf00276932

Schulein, R., Guye, P., Rhomberg, T. A., Schmid, M. C., Schröder, G., Vergunst, A. C., et al. (2005). A bipartite signal mediates the transfer of type IV secretion substrates of *Bartonella henselae* into human cells. *Proc. Natl. Acad. Sci. U. S. A.* 102, 856–861. doi: 10.1073/pnas.0406796102

Shropshire, J. D., and Bordenstein, S. R. (2019). Two-by-one model of cytoplasmic incompatibility: synthetic recapitulation by transgenic expression of cifA and cifB in *Drosophila. PLoS Genet.* 15:e1008221. doi: 10.1371/journal.pgen.1008221

Shropshire, J. D., Leigh, B., Bordenstein, S. R., Duplouy, A., Riegler, M., Brownlie, J. C., et al. (2019). Models and nomenclature for cytoplasmic incompatibility: caution over premature conclusions - a response to Beckmann et al. *Trends Genet.* 35, 397–399. doi: 10.1016/j.tig.2019.03.004

Shropshire, J. D., On, J., Layton, E. M., Zhou, H., and Bordenstein, S. R. (2018). One prophage WO gene rescues cytoplasmic incompatibility in *Drosophila melanogaster. Proc. Natl. Acad. Sci.* 115, 4987–4991. doi: 10.1073/pnas.1800650115

Sun, G., Zhang, M., Chen, H., and Hochstrasser, M. (2022). The CinB nuclease from wNo *Wolbachia* is sufficient for induction of cytoplasmic incompatibility in *Drosophila. MBio* 13, e03177–e03121. doi: 10.1128/mbio.03177-21

Teixeira, L., Ferreira, Á., and Ashburner, M. (2008). The bacterial symbiont Wolbachia induces resistance to RNA viral infections in *Drosophila melanogaster. PLoS Biol.* 6:e1000002. doi: 10.1371/journal.pbio.1000002

Turelli, M. (1994). Evolution of incompatibility-inducing microbes and their hosts. *Evolution* 48, 1500–1513. doi: 10.1111/j.1558-5646.1994.tb02192.x

Turelli, M., Katznelson, A., and Ginsberg, P. S. (2022). Why *Wolbachia*−induced cytoplasmic incompatibility is so common. *Proc. Natl. Acad. Sci.* 119:e2211637119. doi: 10.1073/pnas.2211637119

Vergunst, A. C., van Lier, M. C. M., den Dulk-Ras, A., Grosse Stüve, T. A., Ouwehand, A., and Hooykaas, P. J. J. (2005). Positive charge is an important feature of the C-terminal transport signal of the VirB/D4-translocated proteins of Agrobacterium. *Proc. Natl. Acad. Sci.* 102, 832–837. doi: 10.1073/pnas.0406241102

Werren, J. H., Baldo, L., and Clark, M. E. (2008). Wolbachia: master manipulators of invertebrate biology. *Nat. Rev. Microbiol.* 6, 741–751. doi: 10.1038/nrmicro1969

Wnętrzak, M., Błażej, P., Mackiewicz, D., and Mackiewicz, P. (2018). The optimality of the standard genetic code assessed by an eight-objective evolutionary algorithm. *BMC Evol. Biol.* 18:192. doi: 10.1186/s12862-018-1304-0

Xi, Z., Khoo, C. C., and Dobson, S. L. (2005). Wolbachia establishment and invasion in an *Aedes aegypti* laboratory population. *Science* 310, 326–328. doi: 10.1126/science.1117607

Xiao, Y., Chen, H., Wang, H., Zhang, M., Chen, X., Berk, J. M., et al. (2021). Structural and mechanistic insights into the complexes formed by *Wolbachia* cytoplasmic incompatibility factors. *Proc. Natl. Acad. Sci.* 118:e2107699118. doi: 10.1073/pnas.2107699118

Yamaguchi, Y., Park, J. H., and Inouye, M. (2011). Toxin-antitoxin systems in bacteria and archaea. *Annu. Rev. Genet.* 45, 61–79. doi: 10.1146/annurev-genet-110410-132412

Ye, Y. H., Carrasco, A. M., Frentiu, F. D., Chenoweth, S. F., Beebe, N. W., van den Hurk, A. F., et al. (2015). Wolbachia reduces the transmission potential of dengue-infected *Aedes aegypti. PLoS Negl. Trop. Dis.* 9:e0003894. doi: 10.1371/journal.pntd.0003894

Yen, J. H., and Barr, A. R. (1971). New hypothesis of the cause of cytoplasmic incompatibility in *Culex pipiens* L. *Nature* 232, 657–658. doi: 10.1038/232657a0

Yen, J. H., and Barr, A. R. (1973). The etiological agent of cytoplasmic incompatibility in *Culex pipiens. J. Invertebr. Pathol.* 22, 242–250. doi: 10.1016/0022-2011(73)90141-9

Yoshida, M., Hinkley, T., Tsuda, S., Abul-Haija, Y. M., McBurney, R. T., Kulikov, V., et al. (2018). Using evolutionary algorithms and machine learning to explore sequence space for the discovery of antimicrobial peptides. *Chem* 4, 533–543. doi: 10.1016/j.chempr.2018.01.005

Zheng, X., Zhang, D., Li, Y., Yang, C., Wu, Y., Liang, X., et al. (2019). Incompatible and sterile insect techniques combined eliminate mosquitoes. *Nature* 572, 56–61. doi: 10.1038/s41586-019-1407-9