Microbes in Space: The Importance of Spatial Organization in Evolution

Katrina van Raay

A dissertation

submitted in partial fulfillment of the

requirements for the degree of

Doctor of Philosophy

University of Washington

2021

Reading Committee:

Benjamin Kerr, Chair

Maitreya Dunham

Janneke Hille Ris Lambers

Program Authorized to Offer Degree:

Biology

University of Washington

Abstract

Microbes in Space: The Importance of Spatial Organization in Evolution

Katrina van Raay

Chair of the Supervisory Committee:

Benjamin Kerr

Biology

My research uses a multi-prong approach to understand the ecological and evolutionary processes that drive and maintain diversity. Specifically, I examine how competition and spatial structure drive evolutionary change.

Part 1. A population under selection to improve one trait may evolve a sub-optimal state for a second trait due to tradeoffs and other evolutionary constraints. How the duration of evolution under conditions favoring the first trait affects the capacity of a population to adapt when conditions change to favor the second trait is an open question. We investigated this question using isolates from a lineage spanning 60,000 generations of the Long-Term Evolution Experiment (LTEE) with *Escherichia coli*. In the LTEE, cells have access to a shared pool of resources, and the bacteria have evolved increased competitive ability and a concomitant reduction in numerical yield. Here, we shifted the focus of selection to numerical yield by propagating LTEE-derived

populations in media-in-oil emulsions, in which cells grew in isolated patches with private resources. We found that the length of time evolving under shared resources (i.e., duration in the LTEE) did not affect the ability to re-evolve toward higher numerical yield. The evolution of numerical yield in emulsified populations commonly occurred through mutations in the phosphoenolpyruvate phosphotransferase system. These mutants exhibit slower uptake of glucose, which makes them poorer competitors for public resources. However, these mutants also produce smaller cells that release less carbon as overflow metabolites, which makes them more numerically productive when resources are private. Our results demonstrate that mutations that were not part of adaptation under one selective regime may enable access to ancestral phenotypes when selection changes to favor evolutionary reversion.

Part 2. Chemical warfare in the microbial world is ubiquitous. One kind of chemical weapon microbes employ is a proteinaceous toxin called a bacteriocin. The best-studied bacteriocins are the colicins, produced by and active against *Escherichia coli*. Many colicin systems encode suicidal lysis genes, such that the producing cell releases the toxin through cell lysis. Released colicin kills sensitive competing cells allowing immune clones of the producer to capitalize on the liberated resources. However, a major group of colicin systems lack this lysis gene, and it has been unclear how such toxins were released from the producing cell. I explore a newly discovered union between such a colicin system and a prophage (bacterial virus that has been incorporated into the bacterium's genome), where release of the colicin occurs via phage-encoded cell lysis. This union should be detrimental to susceptible cells, which would be hit twice: once by the colicin, once by the phage. Mathematical modeling shows that the success of this dual killing system depends on the prophage's ability to produce infectious virions in a well-mixed environment. If the prophage is cryptic (encoding lysis, but not producing infectious phage), then

this dual system is susceptible to invasion by “cheaters” that only possess the colicin system (and its immunity) but do not lyse. I use agent-based simulations to model this scenario in a spatially structured environment. Under spatial conditions, the social dilemma and is resolved when the population exhibits fluctuating rock-paper-scissors dynamics (where rock chases scissor, scissor chases paper, and paper chases rock), providing insights into what ecological conditions were necessary in order to maintain this union.

Table of Contents

Acknowledgements

I am very grateful for the people who supported and believed in me before and during the PhD experience.

Chapter 1

A return to form: Novel mutations keep ancestral traits accessible despite evolutionary history

Katrina van Raay[1,2], Sergey Stolyar[2,3], Jordana Sevigny[1,2], Jeremy A. Draghi[4], Richard E. Lenski[2,5], Christopher J. Marx[2,3,6,7], Benjamin Kerr[1,2], Luis Zaman[2,8]

1. Department of Biology, University of Washington, Seattle, WA;

2. BEACON Center for the Study of Evolution in Action, Michigan State University, East Lansing, MI;

3. Department of Biological Sciences, University of Idaho, Moscow, ID;

4. Department of Biological Sciences, Virginia Tech, Blacksburg, VA;

5. Department of Microbiology and Molecular Sciences; Michigan State University, East Lansing, MI;

6. Institute for Modeling Collaboration and Innovation, University of Idaho, Moscow, ID;

7. Institute for Bioinformatics and Evolutionary Studies, University of Idaho, Moscow, ID;

8. Department of Ecology and Evolutionary Biology, and Center for the Study of Complex Systems, University of Michigan, Ann Arbor, MI.

Introduction

The history of life has been inextricably shaped by evolutionary constraints. Given phenotypic trade-offs, an organism cannot simultaneously optimize two traits with respect to fitness (Agrawal, Conner, and Rasmann 2010; Stearns 1989). Thus, a population under strong selection to improve one trait may evolve a sub-optimal state for another trait. More generally, pleiotropy means that mutations in one gene affect multiple traits simultaneously (Pavličev and Cheverud 2015), and antagonistic pleiotropy occurs when mutations are beneficial for one trait but detrimental for another. While there is a rich literature on evolutionary tradeoffs (Beardmore et al. 2011; Gounand et al. 2016; Guillaume and Otto 2012) and antagonistic pleiotropy (Rose, 1982; Rose, 1985), there has been less attention given to how the evolutionary history of a lineage under selection for one trait affects its subsequent trajectory when selection shifts to favor a second trait that trades off with the first (but see Ostrowski, Ofria, & Lenski, 2015; Teotónio & Rose, 2000; Travisano & Lenski, 1996; Velicer, 1999).

To understand how previous adaptation might affect future evolution, let us imagine a population under long-term selection for one trait that trades off with a second trait (Figure 1a). For simplicity, we assume higher values of each trait correspond to higher fitness when under selection. Long-term improvement in the favored trait (trait 1 in Figure 1a) occurs along with concomitant decreases in the second trait (trait 2). If selection shifted to favoring trait 2, how would the duration of history evolving under selection for trait 1 affect future evolution (Figure 1b)? There are at least two plausible hypotheses about how that duration would influence the rate of adaptation. The "Mutational Access" hypothesis (Figure 1c) posits that populations that spent more time under selection for trait 1 have fewer available mutations that would improve fitness by increasing trait 2 (i.e., the "late descendant" compared to the "intermediate descendant" in Figure

1a). If so, then this would lead to a *lower* rate of adaptation for the late descendant than for the intermediate one when trait 2 is favored. This mutational limitation could result from the necessity of compensating for (or reverting) more mutational steps; alternatively, it might occur if mutations increasing trait 2 become increasingly deleterious over time due to the entrenchment of trait 1 (Schank and Wimsatt 2011; Shah, McCandlish, and Plotkin 2015). The "Strength of Selection" hypothesis (Figure 1d), by contrast, suggests that the intermediate and late descendants have equal access to mutations improving trait 2, but the selective benefits for the late descendant are greater due to the relatively larger changes in fitness upon improvement (Barrick et al. 2010; Chou et al. 2011). Under this hypothesis, the rate of adaptation is predicted to be *higher* for the late descendant.

If adaptation is possible under selection for trait 2, then we expect a return towards the progenitor's value of that trait. Does this phenotypic return occur by "rediscovering" the ancestral genetic and mechanistic underpinnings of the second trait (Levin et al. 1997; Pennings, Ogbunugafor, and Hershberg 2020)? This rediscovery could occur by reverting key mutations, or by otherwise returning to the regulatory, physiological, or metabolic states that enabled the progenitor's phenotype. Indeed, the mechanisms that underly a return to the ancestral state (e.g., Figures 1b-d) might differ depending on whether evolution is initiated with intermediate or late descendants.

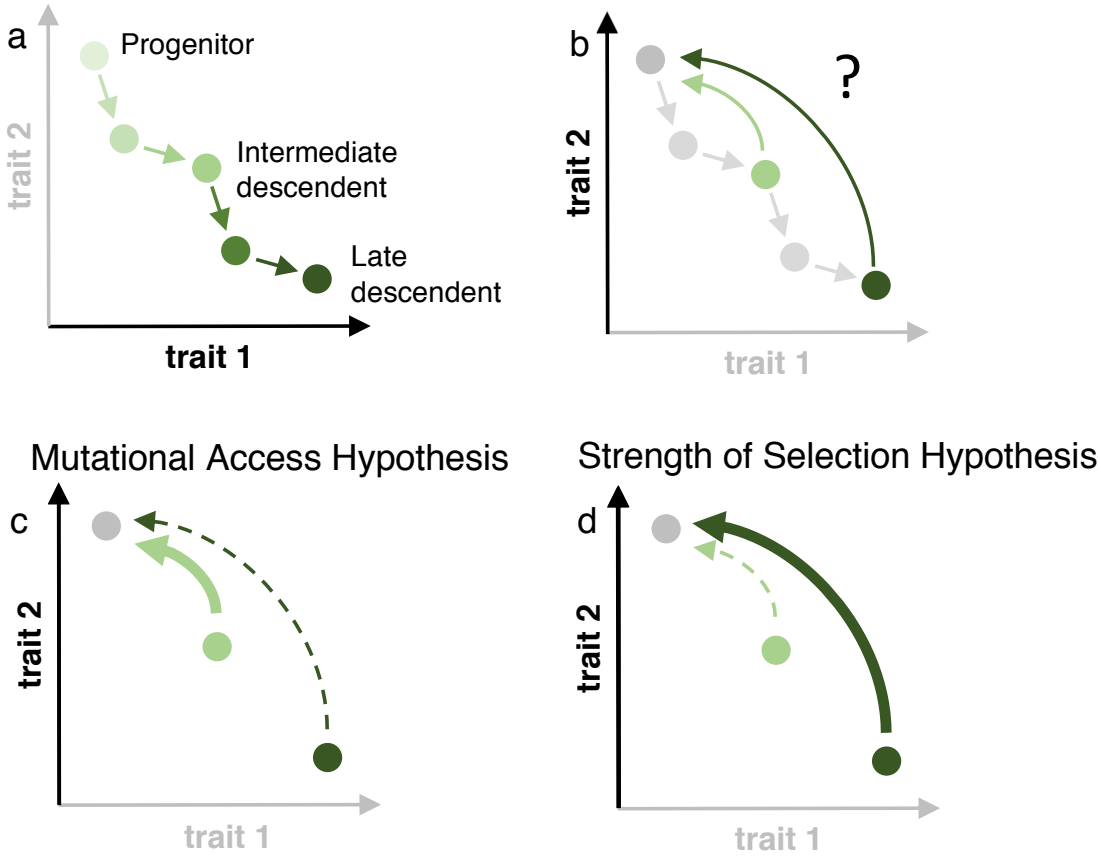

Figure 1: Evolutionary trajectories and alternative hypotheses. a) A population evolves under direct selection for one trait (trait 1, bolded axis). A higher value along either axis indicates an improvement in that trait under selection. Each circle represents a genotype in an evolutionary line of descent. We focus on three genotypes along an evolutionary trajectory: the progenitor, an intermediate descendent, and a late descendant. b) Selection is relaxed for trait 1 and strengthened for trait 2 (bolded axis). The curved arrow represents the transition of a population with descendent traits towards traits that resemble the progenitor when selection acts on trait 2. Whether, how fast, and by what mechanisms this transition occurs are the main questions addressed in this paper. c) Mutational access hypothesis. Intermediate (light green) and late (dark green) descendants are shown after selection is relaxed on trait 1 and strengthened on trait 2. The width of arrows pointing from each descendent towards the progenitor indicates the rate of change. The thick arrow from the intermediate descendent denotes a faster return to the progenitor phenotype than the dashed arrow from the late descendent. Under this hypothesis, the late descendent is either mutationally further from the progenitor state or the mutational options to revert to the progenitor phenotype are more costly due to entrenchment. d) Strength of selection hypothesis. Intermediate (light green) and late (dark green) descendants are shown when selection flips from favoring trait 1 to trait 2. The width of arrows pointing from each descendent towards the progenitor indicates the relative strength of selection. The thick arrow from the late descendent indicates stronger selection favoring a mutation that restores the ancestral phenotype, and thus a faster response, than for the same mutation in the intermediate descendant (dashed arrow). Under this hypothesis, both descendants have the same mutational access to various positions in phenotype space.

Here we present a realization of the thought experiment in Figure 1, in which the bacterium *Escherichia coli* faces a tradeoff between its competitive ability and yield. We test how the length

of time that populations have adapted in an environment that selects primarily for faster growth (making the bacteria better competitors) affects their future adaptation in a novel environment that favors increased yield. To do so, we take advantage of the Long-Term Evolution Experiment (LTEE), in which 12 replicate populations of *E. coli* have been evolving for over 70,000 generations in a relatively simple environment (Lenski 2017; Vasi, Travisano & Lenski 1994; Wiser, Ribeck, and Lenski 2013). Periodic samples from each line have been frozen in suspended animation, so that the "progenitor," "intermediate descendant," and "late descendant" are available by reviving samples from this living fossil record. The well-stirred, unstructured environment of the LTEE favors rapid growth, above all else (Vasi et al. 1994, Novak et al. 2006). Furthermore, there is no expectation of direct selection for either metabolic yield or numerical yield. In an unstructured environment, a mutant that consumes resources more slowly but efficiently is disadvantaged, because a faster-growing but less-efficient competitor can deplete the shared resources in the meantime. Instead, a mutant that grows faster has an advantage in an unstructured environment, even if it leads to a reduction in the population's numerical yield, metabolic yield, or both.

The actual outcomes in the LTEE are somewhat more complex. All the populations evolved faster growth, but their total biomass also increased (Lenski & Mongold 2000, Novak et al. 2006 , Marshall et al. in prep.). On the other hand, when directly measuring metabolic efficiency through central metabolism, the evolved populations are slightly (but significantly) less proficient at converting glucose into biomass (Harcombe et al. 2013). Moreover, evolved strains from the LTEE -- but not the ancestor -- are able to grow on peripheral metabolites, thus suggesting that loss of efficiency through central metabolism is more than compensated by the exploitation of alternative substrates (Leiby and Marx 2014). Although there would have been no advantage to

increased biomass if it led to slower growth, faster growth was evidently achieved, at least in part, by doing just that. This constellation of results can be understood by considering that, in the novel environment of the LTEE, the ancestral strain was likely far from the trade-off front between growth rate and total biomass, leaving room for improvements in both trats (Novak et al. 2006). At the same time, and perhaps counterintuitively, all of the LTEE lines evolved to produce much larger individual cells than the ancestor (Lenski & Travisano 1994, Vasi et al. 1994, Grant et al. 2021). The increase in average cell size more than offset the increased biomass and, as a consequence, the numerical yield of the LTEE populations—the number of the cells at the end of the growth cycle and just before the daily transfers—has declined substantially (Vasi et al. 1994, Wiser et al. 2013). In other words, the LTEE lineages exhibit an increase in competitive ability over their common ancestor (a progression up the rate axis, as in the evolution of trait 1 in Figure 1a), while simultaneously having decreased in their numerical yield (Harcombe et al., 2013; Wiser, Ribeck, and Lenski 2013) (a progression down the numerical yield axis, as in the evolution of trait 2 in Figure 1a).

Now consider what might happen if the bacteria that previously evolved in the LTEE moved to and evolved in a structured environment that was otherwise similar to the unstructured LTEE environment. By distributing a transplanted population into many isolated 'patches,' each seeded by a single cell, any resources saved by growing more efficiently would accrue disproportionately to cells with the same genotype, which would effectively eliminate resource competition. Thus, increased metabolic efficiency could evolve, even if it led to slower growth. Now imagine that the cells in these patches were periodically pooled, diluted distributed as single founders over a new set of empty patches. In that case, selection would favor increased numerical yield since each individual cell is a potential propagule able to establish a new subpopulation.

Increased numerical yield could be achieved by becoming more metabolically efficient, by producing smaller individual cells, or perhaps some combination of these changes. In this way, selection shifts to favor numerical yield over competitive ability for shared resources (accomplishing the transformation from Figure 1a to Figure 1b in our thought experiment).

In our study, we achieve this shift by propagating cells in water-in-oil emulsions comprised of millions of aqueous media-filled droplets surrounded by an oil phase (Figure 2) (Bachmann et al. 2013). If a population of cells is diluted sufficiently before creation of the emulsion, then each cell will usually be the sole occupant of a droplet. In such a case, barring mutation, there is no competition between genotypes for resources, as each cell (and its progeny) has access to a private resource supply demarcated by the droplet. If growth of isolated cells inside droplets proceeds for sufficient time to exhaust the substrate and is followed by demulsification, dilution, and redistribution into droplets across successive transfers (as in Figure 2a), then those genotypes that produce *more* cells (rather than the fastest-growing cells) will have an advantage. Indeed, prior experiments have demonstrated that various forms of isolation can favor numerical yield at the expense of competitive ability in several different biological systems (Bachmann et al. 2013; Eshelman et al. 2010; Kerr et al. 2006; van Tatenhove-Pel et al. 2021).

Here we focus on a single LTEE lineage in order to explore how previous selection for one trait (competitive ability) affects the tempo and mode of evolution under selection for a second trait (numerical yield) in the presence of a phenotypic tradeoff. We probe how the length of time adapting to the well-mixed LTEE regime affects the ability of a population to adapt under the emulsion conditions. We investigate whether increases in numerical yield come with associated decreases in competitive ability, and whether they are achieved through increases in metabolic efficiency. Because improvement in competitive ability in the LTEE was accompanied by an

increase in cell size (Grant et al. 2021; Novak et al. 2006), and because – all else being equal – a reduction in cell size increases numerical yield, we track this attribute in our evolutionary lines. We also determine the nature and number of mutations that mediate these phenotypic changes. Because changes in metabolism can affect both competitive ability and yield, we explore not only the morphological and genetic bases, but also the metabolic underpinnings, of the evolved traits when there are evolutionary reversions to the ancestral phenotypes.

Methods

Strains

All experiments were founded with *E. coli* B isolates that came from the Ara–5 lineage in the LTEE. The isolates used in our experiment came from 0 generations (the founder of the LTEE), 20,000 generations, and 60,000 generations of evolution in the LTEE (Table S1; hereafter called 0K, 20K, and 60K, respectively). The 0K isolate is the ultimate progenitor of the 20K and 60K isolates in the context of the LTEE. However, because we conduct evolution experiments in the emulsion system with these three strains, we will refer to these LTEE isolates as the "0K emulsion ancestor", the "20K emulsion ancestor", and the "60K emulsion ancestor".

Emulsion evolution experiment

We propagated three replicate cultures founded by each of the three LTEE isolates (the emulsion ancestors) under selection for increased numerical yield by using water-in-oil emulsions (Bachmann et al. 2013) for 100 transfers. This method (see below) creates millions of media-filled, picoliter-sized droplets surrounded by an oil phase, in which bacterial cells in one droplet are

isolated from those in others (Figure 2). Emulsions were set up in 1.5-ml Eppendorf conical microcentrifuge tubes, and they were incubated at 37 °C without shaking.

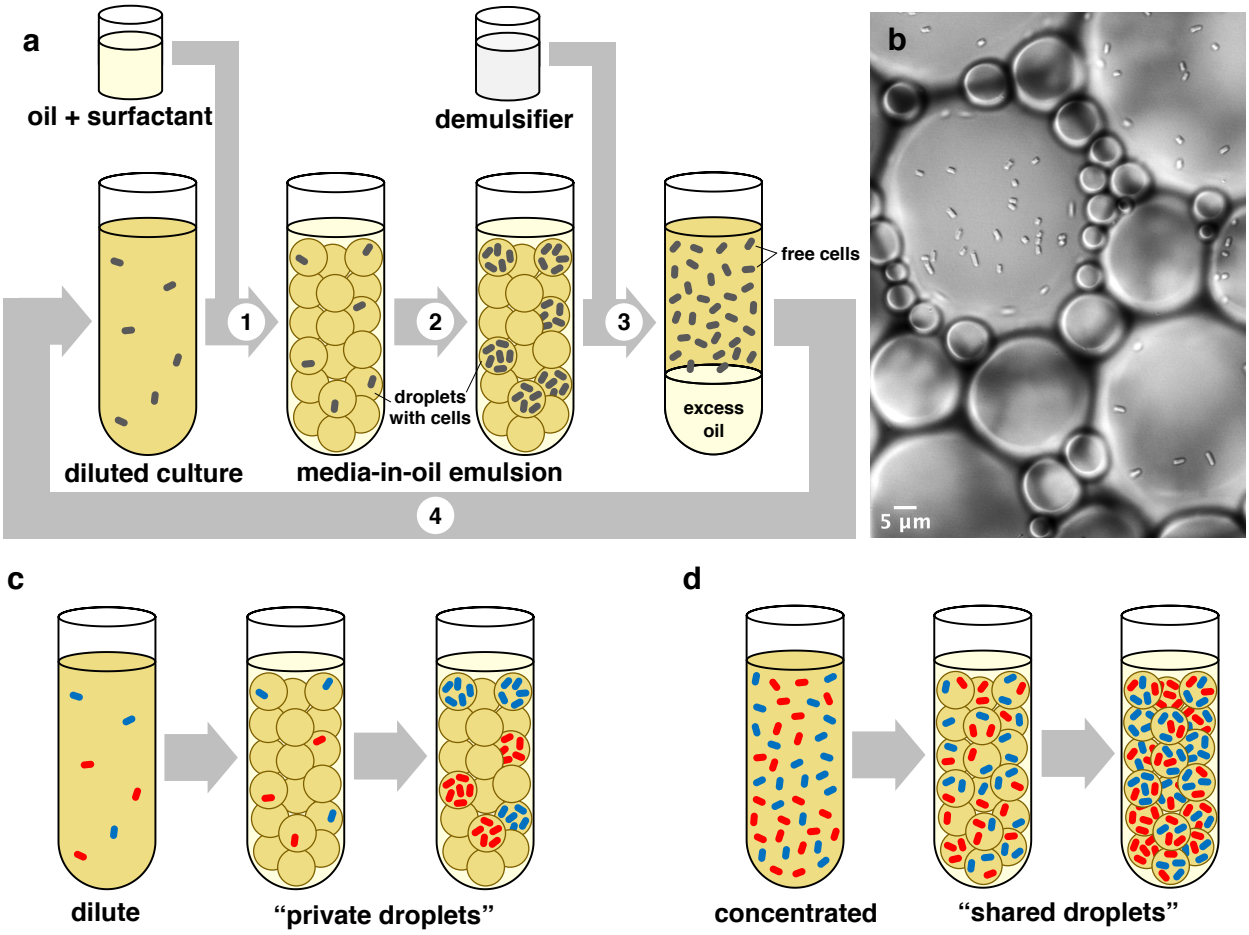

Figure 2: a) Schematic of transfer protocol: 1. A mixture of oil and surfactant is added to a diluted culture of bacteria in growth media and vortexed to form millions of droplets, 2. Bacteria in the emulsion are incubated to allow subpopulation growth within the droplets, 3. After overnight growth, a demulsifier is added to break the emulsion and resuspend the bacteria, 4. The free cells are diluted into fresh media and the cycle is repeated, **b)** Micrograph of bacteria growing in emulsion droplets, **c)** Schematic of inoculation and growth for private droplets, **d)** Schematic of inoculation and growth for shared droplets. For parts c and d, the two colors of bacterial cells represent distinct genotypes.

In order to select for increased numerical yield, we diluted the bacterial culture before setting up the emulsions, such that 90% of occupied droplets would contain only 1 cell at the

beginning of each transfer cycle. We subsequently showed that occupied droplets had an average of 1.12 cells (see Supplementary section *Calculating Droplet Statistics*). Bacteria were allowed to grow and divide in droplets for 24 hours, after which the droplets were broken with 1*H*,1*H*,2*H*,2*H*-perfluoro-1-octanol, releasing all the cells into a common pool. A fraction of the cells from this pool were immediately used to initiate the next transfer (see Figure 2a). Because of the level of dilution before emulsification, lineages from this treatment have evolved in "private droplets" (see Figure 2c). This treatment was continued for 100 transfers.

To control for any effects of growing in emulsions, we also propagated three replicate cultures founded by each LTEE isolate for 100 transfers under higher starting-density conditions, in which each occupied droplet had almost 3 cells on average (see Supplement). We predicted that we would not observe evolution towards increased numerical yield in this "shared-droplet" treatment (see Figure 2d).

During the evolution experiment, the cell density of freshly broken emulsions was measured with a FilterMax F5 Multi-Mode Microplate Reader (Molecular Devices) after 24 hours. Briefly, we pipetted 150 µl of overnight culture from a broken emulsion into a flat-bottom 96-well microtiter plate, and we recorded an OD_{595} reading. This reading was then used to estimate cell density using a calibration curve giving the relationship between OD_{595} and CFU/ml, which was established in prior experiments. Cultures were then diluted to 2×10^6 CFU/ml for the private droplet treatment and 5×10^7 CFU/ml for the shared droplet treatment. Emulsions were created by adding 200 µl of a mixture of 9 parts Novec 7500 HFE to 1 part Pico-surf (5% (w/w)) to 300 µl of diluted culture in the growth medium DM1000 in 1.5 ml Eppendorf tubes, and vortexing at maximum speed for 2 minutes. DM1000 contains 1000 mg/l of glucose, and it was used instead of the DM25 (25 mg/l glucose) used in the LTEE because DM1000 exhibited clearer differences

between the 0K and 60K samples in terms of numerical yield and competitive ability than DM25 in pilot experiments.

Single-colony isolates were picked from each population at the end of our evolution experiment (after transfer 100) and frozen at -80 °C as 20% v/v glycerol stocks. We refer to these evolved isolates as "emulsion descendants". Two loci previously shown to distinguish the 0K, 20K, and 60K ancestors were Sanger sequenced for each isolate to confirm the intended derivation of the isolates. We detected within-experiment contamination via Sanger and subsequent whole-genome sequencing in 4 of the 18 emulsion descendants, so these populations were dropped from our analysis. The complete genomes of the remaining 14 strains (8 private droplet, 6 shared droplet) were sequenced and analyzed, and the 6 isolates of private-droplet descendants that had mutations were used for further analysis. For the shared droplet treatment, one descendent was randomly chosen from each lineage to be assessed in the phenotypic assays (3 isolates).

Competition assays

Using the frozen samples of the emulsion ancestors (0K, 20K, and 60K) and their emulsion descendants from the private and shared droplet treatments, we assessed the relative competitive abilities of all of our strains. The emulsion descendants competed against a common marked strain (equivalent to our 0K emulsion ancestor) in shared droplet conditions. We used a neutral marker (a point mutation in the *araA* gene) to distinguish the focal and common competitor strains on tetrazolium-arabinose indicator agar plates.

Competitions ran for 3 transfer cycles in emulsions. Competitions were initiated with a 1:1 ratio of the focal strain (denoted F) to the common competitor (denoted C). The competitive ability

of the focal strain relative to the common competitor ($w(F,C)$) was assessed by calculating a ratio of their realized Malthusian growth rates (Lenski et al. 1991):

$$w(F,C) = \frac{\ln \frac{F_{\text{final}}}{F_{\text{initial}}} + \sum_{i=2}^{3} \ln z_i}{\ln \frac{C_{\text{final}}}{C_{\text{initial}}} + \sum_{i=2}^{3} \ln z_i},$$

where X_{initial} and X_{final} are the initial (beginning of the first transfer) and final (end of the third transfer) densities, respectively, of strain X (with $X \in \{F, C\}$) and z_i is the factor by which the population is diluted in order to initiate the i^{th} transfer.

Genome Sequencing

Single-colony isolates from 14 evolved lines and their 3 ancestors were grown in DM1000 and frozen as 20% v/v glycerol stocks at -80 °C. For genome sequencing, cells from these frozen stocks were reanimated in 5 ml of LB (lysogeny broth). Genomic DNA was extracted using a Qiagen DNEasy kit with 1 ml of overnight culture, following the manufacture's protocol. Library preparation and sequencing for 16 strains was done at the Microbial Genome Sequencing Center (MiGS) at the University of Pittsburgh. One strain (the 0K emulsion ancestor) was sequenced at the University of Washington on Illumina's NextSeq platform and prepped with Illumina Nextera barcodes. Genome sequences had an average of 131X coverage.

Mutations (predicted mutations, unassigned missing coverage, and unassigned new junctions) were called using breseq (Version 0.33.2) (Deatherage 2014) with default parameters. All mutations were confirmed using Integrative Genomics Viewer (Robinson et al. 2011), and four point mutations identified in strains that differed phenotypically from their ancestors in cell size or numerical yield were Sanger sequenced for confirmation (Table 3).

Numerical yield and cell size assays

Populations of each emulsion ancestor and descendant were grown under the shared-droplet conditions over a standard 24-hour growth cycle. We used the shared-droplet emulsions instead of the well-stirred conditions to control for the microenvironmental effects of growing in an emulsion (e.g., oxygen depletion during growth). The final cell number in the 24-hour sample was used as our measure of numerical yield. Cell size was measured as the median cell volume (μm^3) of several thousand cells using a Beckman Coulter MSE 4 instrument.

Metabolic analysis

Supernatants from the yield assays were filtered and stored at -80 °C prior to metabolic analysis. After thawing, samples were analyzed using High Performance Liquid Chromatography (HPLC) on a 20A chromatographic system (Shimadzu) equipped with diode array and refractive index detectors. Samples were eluted with 6 mM H_2SO_4 at a flow rate of 0.6 ml/min from an Aminex HPX 87H organic acid analysis column (300 by 7.8 mm) (BioRad). We recorded the areas for all noticeable peaks associated with detected metabolites (glucose, citrate, acetate, formate, and lactate), and these values were converted to metabolite concentrations (in mM) using organic acid and sugar standards (from BioRad or made in-house). Citrate, an iron chelator present in DM medium, is neither produced nor consumed by these bacterial strains, and it has been shown not to affect the growth of the Ara–5 lineage (Leiby, Harcombe, and Marx 2012). Thus, citrate was used as an internal control.

Analysis of metabolic data

The rates of production for each metabolite are difficult to compare because of differences in the growth rates of the strains. To minimize this potentially confounding factor, we performed linear regression of the concentration of each metabolite against the remaining concentration of glucose. The slope of this relationship estimates the ratio of metabolite produced for each unit of glucose consumed, assuming an approximate equilibrium between the uptake of glucose and the release of metabolites into the medium. As glucose levels drop, however, there is increased uncertainty in the glucose measurements; moreover, as the glucose becomes scarce, the cells may begin to consume the excreted metabolites, which would bias our estimates of the production rates. To avoid these problems, we excluded time-points in the linear regressions for which the mean glucose concentration was below 0.8 mM. We chose this threshold because, at lower concentrations, growth curves showed clear signs of departure from exponential growth. We also excluded the measurements made at t=0 and, instead, we constrained the x-intercept of the regressions to reflect the known initial concentrations in the media (i.e., 6 mM glucose and none of the overflow metabolites). To estimate how quickly the glucose was consumed, we performed regressions of glucose concentrations against time, using the same criteria for data exclusion. This approach provided a measure of growth rate that was insensitive to differences in cell size across strains.

Results & Discussion

Private-droplet treatment selected for high numerical yield

Our private droplet treatment was meant to minimize local competition between genotypes, and we predicted that it would favor increased numerical yield. To assess whether the emulsion-

evolved populations underwent changes in numerical yield, we measured their cell densities, along with those of their ancestors, at the end of 24 hours of growth in the emulsion system. All 6 private droplet descendants had significantly increased numerical yield (cell number) compared with their ancestors (Figure 3, Table S2; BK 433: $p < 0.05$; BK 431, BK 437, BK 438, BK 441, BK 442: $p < 0.01$, unpaired two-tailed t-tests). Our shared-droplet treatment served as control for the private-droplet treatment, allowing local competition between genotypes, while controlling for growth in the emulsion environment. Two of the three shared-droplet descendants had increased numerical yield and one had decreased yield; however, the gains were less than those in the private-droplet treatment, and all three changes were non-significant. Combining the 0K, 20K, and 60K lines, the private-droplet descendants were significantly more productive than the shared-droplet descendants ($p=0.039$, paired one-tailed t-test; see Supplementary section *Statistical Analysis*). The two evolutionary treatments had different effective population sizes. However, the weaker numerical-yield response in the shared-droplet treatment relative to the private-droplet treatment cannot be explained by the lack of mutational opportunity, because the number of cell divisions per transfer is actually greater in the shared-droplet treatment (see Supplementary section *Calculating Number of Cell Divisions in Droplets)*. Taken together, these data confirm our prediction that the private-droplet treatment exerts stronger selection for increased numerical yield than does the shared-droplet treatment.

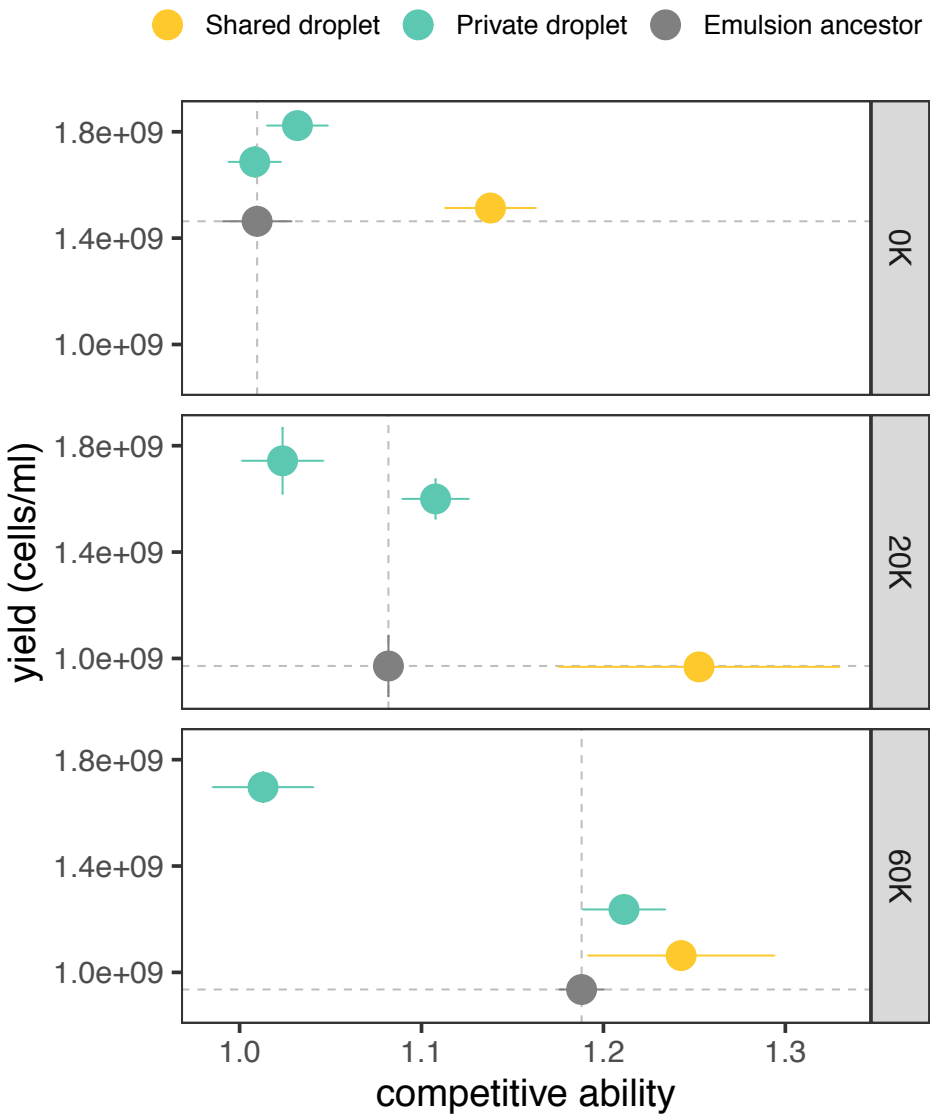

Figure 3: Numerical yield (cell density after the standard 24-hour growth cycle) compared to relative competitive ability, both measured in the emulsion system. Each panel shows strains originating from a single emulsion ancestor (in grey, at right) from the Ara–5 lineage of the LTEE (0, 20,000, and 60,000 generations). Colors indicate emulsion evolution treatments (labels at top). The coordinate positions of each point are the averages of three replicate assays for each phenotypic trait. Error bars show SEM; when they are not visible along either axis, the corresponding SEM is smaller than the symbol itself. Dashed lines show the mean competitive ability and numerical yield of the relevant ancestor.

In particular, the shared-droplet results demonstrate that the increase in numerical yield in the private-droplet treatment was not simply an evolutionary response to propagation under the emulsion conditions. Instead, these results imply that the initial density within a droplet affects the evolutionary outcome. By reducing competition between genotypes within the private droplets, selection has favored numerical yield. In contrast, we would expect the shared-droplet treatment to shift selection towards increased competitive ability.

Shared droplet treatment favored competitive ability

Competitive ability increased in all three shared-droplet descendants that we tested (Figure 3). However, the change was significant relative to the corresponding emulsion ancestor only in the case of the line derived from the 0K ancestor (Table S2). Competitive ability also increased in three of the six private-droplet descendants, but none of these increases were significant, and the magnitudes of the increases were smaller than those observed in the shared-droplet treatment. Competitive ability decreased in the other three private-droplet descendants, and the decline was significant in one case (BK 441: $p < 0.01$, unpaired two-tailed t-test). Combining the results from the 0K, 20K, and 60K lines, the shared-droplet descendants were significantly more competitive than the private-droplet descendants ($p=0.038$, paired one-tailed t-test; see Supplementary section *Statistical Analysis*). Overall, these results support our prediction that starting each droplet with multiple cells favors the evolution of increased competitive ability.

Tradeoff between competitive ability and numerical yield

If competitive ability and numerical yield strictly traded off, then competitive ability would decrease in the private-droplet treatment as numerical yield increased, and numerical yield would

decrease in the shared-droplet treatment as competitive ability increased. However, we did not see such a strict tradeoff. Although the private-droplet descendants had larger gains in numerical yield, some shared-droplet descendants also experienced moderate gains in numerical yield. Similarly, although the shared-droplet descendants exhibited larger gains in competitive ability, some private-droplet descendants also showed moderate improvements in their competitive ability. Thus, these two traits do not strictly tradeoff with one another, and instead there seems to be some misalignment, in which larger gains in one trait sometimes occur with more modest gains in the other trait.

To develop this notion of misalignment further, we can contrast the case of a strict tradeoff, or pure antagonistic pleiotropy (Figure 4a), with two different scenarios of synergistic pleiotropy. The first synergistic scenario involves strict alignment, such that a mutation that is most beneficial for trait 1 is also the most beneficial for trait 2 (Figure 4b). In essence, this scenario is the opposite of strict antagonistic pleiotropy. The second form of synergism involves misalignment. In this scenario, all mutations simultaneously improve both traits, but the mutations that are the strongest for trait 1 are the weakest for trait 2, and *vice versa* (Figure 4c). Misalignment occurs when the relative effect of a mutation on one trait is the opposite of its relative effect on another trait. Therefore, antagonistic pleiotropy (Figure 4a) is also a case of misalignment, although the term is broader than the traditional definition of "antagonism" (as illustrated in Figure 4c). In the Supporting Information, we develop a statistical test for misalignment. Using this test, we find that the increases in numerical yield in the private-droplet treatment came with concomitant weaker performance in competitive ability, while the improvements in competitive ability in the shared-droplet treatment came with weaker performance in numerical yield. Despite the absence of a strict tradeoff, these traits exhibited misalignment in our experiment. When two traits are misaligned,

independent evolution in environments that favor one trait or the other can produce a pattern that supports a canonical tradeoff curve when the descendants are pooled across environments and the ancestors are ignored (i.e., considering only the colored points, while ignoring the grey points, in Figures 3 and 4c).

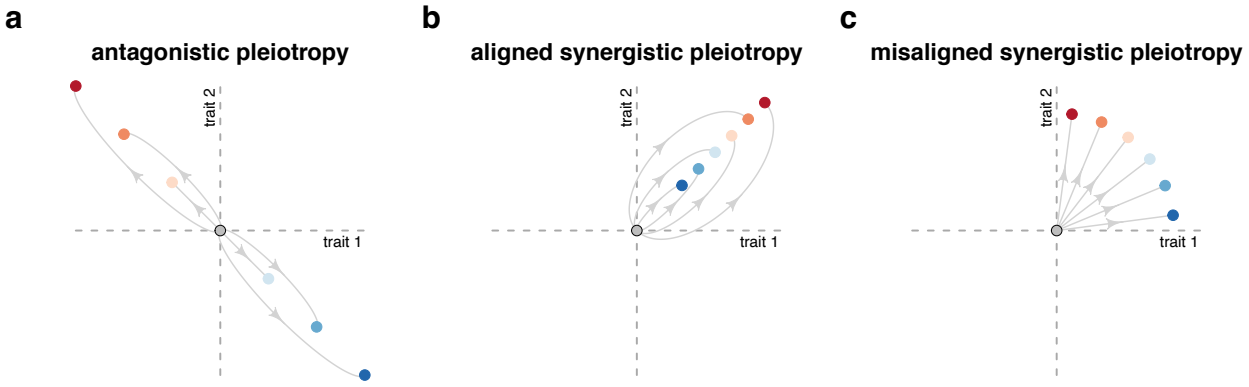

Figure 4: Three scenarios for pleiotropy with respect to two traits. The grey circle is the focal, ancestral genotype and the colored circles show the phenotypes of evolved genotypes. Here, fitness improvement with regard to a trait is taken to mean a *positive* change relative to the ancestor. (a) Antagonistic pleiotropy under a "strict tradeoff" situation, in which an improvement in one trait leads to a decline in the other, and the magnitude of enhancement in one dimension correlates with the magnitude of deterioration in the other. (b) Aligned synergistic pleiotropy describes a situation where the magnitude of improvement in one trait correlates with the magnitude of improvement in the other. (c) Misaligned synergistic pleiotropy describes a situation in which evolution improves both traits, but those genotypes with the strongest improvement in one trait have the weakest improvement in the other.

Private-droplet treatment favored smaller cells

While performing microscopy to assess droplet diameter (Supplemental Figures S8 & S9), we noticed variation in cell size across the emulsion treatments. The cells that had evolved in private droplets appeared smaller than cells that had evolved in shared droplets. We therefore decided to investigate this trait systematically. Figure 5 shows the relationship between cell size and numerical yield.

As the LTEE populations evolved under the well-mixed conditions that favored increased competitive ability, both cell size and total biomass increased in all 12 populations (Grant et al.,

2021, Lenski and Mongold, 2000). This increase in cell size can be seen in our data for the three time points that we sampled from the Ara–5 population, by comparing the values of the three grey points along the x-axis in Figure 5. For the emulsion evolved lines, the median cell volume decreased in all of the isolates from the private-droplet treatment, and the changes were significant in three cases (BK 437, BK 438, BK 441: $p < 0.01$, unpaired, two-tailed t-test). Using an emulsion protocol similar to our private-droplet treatment, Bachman *et al*. (2013) also reported observing an evolved decrease in cell size. They noted that propagation in an emulsion system could favor such a change because any mutant that distributed the same total amount of biomass into a greater number of smaller cells would be able to colonize more droplets each transfer. Our shared-droplet treatment provides an appropriate control for this inference. Indeed, the median cell size actually increased slightly in all three replicates of this treatment, and the change was significant in one case (BK 443: $p < 0.01$, unpaired, two-tailed t-test). The opposing directional changes in our two treatments indicates that neither the emulsion environment nor the transfer protocol (pooling all droplets and transferring only a fraction of cells to the next growth cycle, as shown in Figure 2a) was responsible for the consistently smaller cells seen in the private-droplet descendants. Instead, these findings support the hypothesis that smaller cells evolved in response to the reduced inter-genotype competition in the private-droplet treatment. When the 0K, 20K, and 60K lines are pooled, the difference in cell size between the private-droplet and shared-droplet descendants is marginally significant ($p=0.051$, paired one-tailed t-test; see Supplementary section *Statistical Analysis)*.

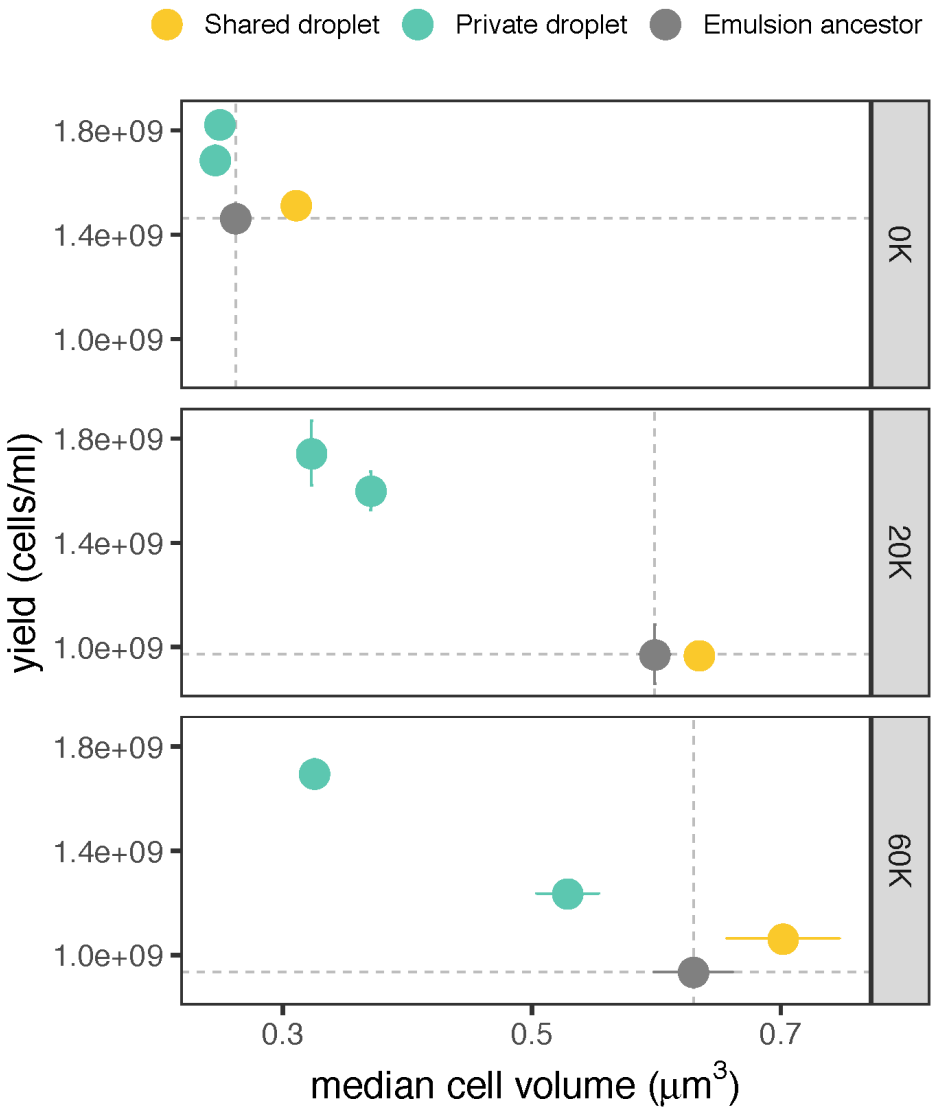

Figure 5: Median cell volume compared to numerical yield. Each panel shows strains originating from a single emulsion ancestor (in grey, at right) from the Ara−5 lineage of the LTEE (0, 20,000, and 60,000 generations). Colors indicate emulsion evolution treatments (labels at top). The coordinate positions of each point show the average of three replicate assays for each phenotypic trait. Error bars show SEM; when they are not visible along either axis, the corresponding SEM is smaller than the symbol itself. Dashed lines show the mean numerical yield and cell volume of the relevant ancestor.

Genome sequencing reveals mutations in phosphoenolpyruvate phosphotransferase system

Given the evolution of smaller cells in the private-droplet treatment, we wondered if these populations were reverting to the ancestral state of the LTEE. After all, the progenitor of the LTEE (the 0K ancestor in our emulsion experiment) had very small cells. In particular, would the same genes and metabolic pathways that changed in the LTEE and led to larger cells in that experiment undergo reversion? Or would novel genetic and metabolic changes lead to a similar phenotypic endpoint? Also, would these changes depend on the particular LTEE-derived ancestor used in the emulsion experiment (e.g., 20K versus 60K)? To find out, we sequenced the complete genomes of 14 emulsion-evolved descendants and their three corresponding ancestors.

This whole-genome sequencing revealed a total number of 13 mutations that distinguish the evolved descendants from their ancestors (Table S3). We saw no cases of identical mutations in multiple descendants, nor was any gene mutated in descendants that evolved in the two different treatments. However, three of the six descendants from the private-droplet treatment had point mutations in genes encoding proteins in the phosphoenolpyruvate (PEP) phosphotransferase system (PTS) (Table 1), which is involved in glucose uptake (Carmona et al. 2015; Escalante et al. 2012; Nam et al. 2001; New et al. 2014; Notley-Mcrobb, Death, and Ferenci 2006; Xia et al. 2017). All three of these descendants also had increased numerical yield and reduced cell size (Table S2), and all three were derived from the 20K and 60K emulsion ancestors. Of the three private-droplet descendants without mutations in PTS genes, two derived from the 0K ancestor, which already had high numerical yield in the emulsion conditions. The third private-droplet descendant without mutations in PTS genes derived from the 60K ancestor, and it exhibited much smaller changes in numerical yield and cell size than its counterpart with a PTS-associated mutation. The changes in numerical yield and cell size were also small for both of the private-

droplet descendants derived from the 0K ancestor, suggesting that the high-yield phenotype of the LTEE progenitor was largely maintained. Bachmann et al. (2013) also found a point mutation in a PTS gene in one strain that evolved increased numerical yield in emulsions, suggesting that PTS might have some role in modulating numerical yield.

The PTS-associated mutations occurred in *ptsH*, which encodes a phosphocarrier protein; *ptsG*, which encodes a subunit of the glucose transporter; and *crp*, which encodes a global transcriptional regulator that affects some steps in glucose uptake (Kimata et al. 1997). Given that these mutations include an early stop codon and non-conservative changes in amino acids, it is likely that some or all of them cause reductions or even losses of function. Mutations in *pykF,* which encodes pyruvate kinase, were among the earliest and most repeatable genetic changes in the LTEE (Woods et al. 2006; Tenaillon et al. 2016; Peng et al. 2018). These mutations presumably reduce the conversion of PEP to pyruvate in central metabolism, which would leave more PEP available to power the transport of PTS-dependent sugars. Indeed, distinct patterns of pleiotropy were observed in the LTEE for various substrates that differed in terms of whether their uptake requires the PTS (Travisano and Lenski 1996). Thus, although the mutations selected in the private droplet-treatment were not in the same genes as those responsible for phenotypic changes in the LTEE, they appear to affect the same system.

STRAIN (BK#)	LTEE GEN	POSITION	REF	ALT	ANNOTATION	GENE	DESCRIPTION
437	20K	2462747	A	C	*86Y (TAA→TAC)	*ptsH*	phosphocarrier protein Hpr
437	20K	3414187	G	A	E55K (GAA→AAA)	*crp*	cAMP-activated global transcriptional regulator CRP
438	20K	1173797	G	A	G454S (GGT→AGT)	*ptsG*	PTS glucose transporter subunit IIBC
441	60K	2462629	T	C	L47P (CTG→CCG)	*ptsH*	phosphocarrier protein Hpr

Table 1: Mutations in private-droplet descendants associated with PTS.

Slower glucose uptake and decreased overflow metabolism in private-droplet descendants

Perhaps the private-droplet descendants achieved their higher numerical yield and smaller cell size in a physiologically similar way to the LTEE progenitor, despite genetic changes that did not merely reverse those that occurred during the LTEE. Because the mutations we found affected glucose transport, we decided to examine their metabolic profiles. Specifically, we sought to determine whether the private-droplet descendants had evolved toward a metabolic state, at least in terms of overflow metabolites, that resembled the LTEE progenitor.

We examined the concentrations of both glucose and various excreted metabolites in spent medium as the various bacterial strains grew in emulsions (Figure 6a). We observed that the glucose concentration for the 20K and 60K private-droplet descendants decreased more slowly than for the corresponding shared-droplet descendants, their ancestors, and the 0K descendants of either the shared or private droplets (Figure 6a, glucose panel). The slower glucose consumption was most pronounced in the private-droplet descendants that had mutations in PTS, indicating that these mutations simultaneously reduced the rate of glucose utilization while increasing numerical yield. These mutations thus appear to reverse the faster glucose-fueled growth and lower numerical yield achieved in the LTEE.

The primary metabolite that is excreted by *E. coli* growing on glucose, and by the LTEE lines in particular, is acetate (Harcombe et al. 2013). Acetate is a byproduct of both glycolysis and fermentation, which is a less efficient form of metabolism than respiration (Szenk, Dill, and de Graff 2017). We saw differences in acetate production between the private-droplet descendants from 20K and 60K and their emulsion ancestors. Specifically, the descendants produced less acetate, which suggests they were more efficient at glucose metabolism under these conditions

than their ancestor. In contrast, some of the shared-droplet descendants accumulated higher levels of acetate (Fig. 6a). In both cases, the acetate concentration eventually declines as it is consumed by the cells after they have depleted the available glucose.

In addition to acetate, other organic acids such as formate and lactate can be generated by *E. coli* during growth on glucose. For all of the isolates in our study, either formate or lactate was generated, but not both. For the 0K lineages, formate was produced by all types (ancestor, private-droplet descendants, and shared-droplet descendant). For the 20K and 60K lineages, only the shared-droplet descendant produced formate, whereas lactate was produced by the 20K and 60K ancestors and their private-droplet descendants.

Differences between strains in the accumulated concentrations of excreted metabolites might reflect different production rates, but they might also be influenced by differences in growth rate and cell size, as well as subsequent consumption of those products. We therefore standardized these measurements by computing the rate of production of each excreted metabolite per unit of glucose consumed, and by excluding time points after the remaining glucose fell below a threshold concentration (see *Methods—Analysis of metabolic data*). After so doing, Figure 6b shows a positive relationship between the initial rate of glucose utilization and the fraction of this carbon that is "wasted" as excreted metabolites in the 20K and 60K emulsion lines. Carbon waste was calculated as a sum of the concentrations across all excreted metabolites concentrations, weighted by the number of carbon atoms in each metabolite. The data indicate that the shared-droplet descendants use the glucose rapidly, but in a relatively wasteful way, whereas the private-droplet descendants consume glucose more slowly and efficiently.

We used these standardized rates (Table S4) to perform a principal components analysis (PCA; Figure 6c) in order to visualize the differences between the emulsion ancestors and their

descendants. The metabolic changes between the 0K ancestor and its descendants were modest, but the changes in the 20K and 60K lines were much greater. For the 20K and 60K ancestors, the metabolic shifts in the private-droplet treatment were distinct from those in the shared-droplet treatment, while the overall patterns of change were nearly identical for the 20K and 60K lines. Thus, in addition to PTS-related mutations conferring high numerical yield for three of the four private-droplet descendants of these two ancestors, the magnitude and nature of the metabolic changes appear similar. One private-droplet descendant (derived from the 60K ancestor) exhibited a smaller decrease in cell size and a more moderate increase in numerical yield (Figure 5). This isolate also did not have mutations associated with PTS, and it differed somewhat less in its metabolic profile from its ancestor. On balance, therefore, we lack clear support for either the Mutational Access Hypothesis or the Strength of Selection Hypothesis (Figure 1). Nonetheless, it is clear that just one or a few novel genetic changes can restore, at least approximately, the ancestral phenotypes of slow growth, small cells, and high numerical yield, even after tens of thousands of generations that led to faster growth, larger cells, and lower numerical yield.

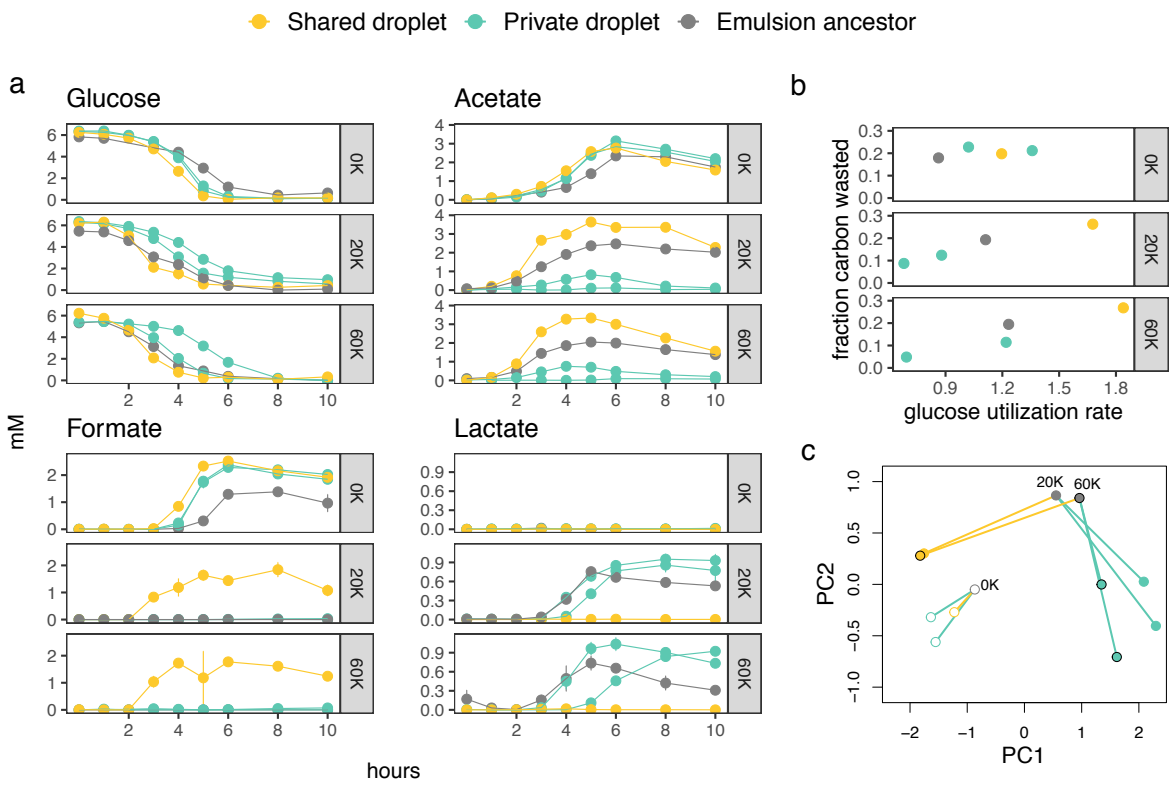

Figure 6: Glucose consumption and metabolite production. a) The concentrations of glucose, acetate, formate and lactate over 10 hours. Each trajectory shows one strain, and each point along a trajectory is the average of up to three replicates per strain. Colors indicate emulsion evolution treatments (labels at top). b) Glucose utilization versus the fraction of carbon wasted. c) Principal components analysis (PCA) based on the strain-specific rates of excreted metabolites per unit glucose consumed. The 0K strains are shown as empty points, the 20K strains as filled points without borders, and the 60K strains as filled points with borders. Lines, colored by treatment, connect emulsion ancestors (grey points) to their descendants.

Summary

Three of the four private-droplet descendants of the 20K and 60K ancestors achieved or even surpassed the numerical yield of the ultimate LTEE progenitor (the 0K emulsion ancestor). When compared to their ancestors, these high-yield descendants showed several parallel changes: smaller cells, more efficient glucose metabolism, mutations in genes encoding the PTS, and lower levels of acetate production. None of these changes occurred in the descendants that evolved from the same ancestors under the shared-droplet treatment. While the increased numerical yield and smaller cell size appeared to be reversions to the phenotypic state of the ultimate progenitor, the

genetic and metabolic underpinnings of the descendants' phenotypes were distinct from those of the ultimate progenitor. Specifically, mutations in genes encoding the PTS do not distinguish the 0K, 20K, and 60K isolates from the LTEE, and the metabolic profiles of the 20K and 60K private-droplet descendants are distinct from that of the 0K emulsion ancestor (Fig. 6c). Therefore, these data indicate a novel mechanistic basis for phenotypic reversion in this system, and one that is at least partly shared between the 20K and 60K emulsion lines.

Dollo's law of irreversibility states that "An organism never returns exactly to a former state, even if it finds itself placed in conditions of existence identical to those in which it has previously lived. But by virtue of the indestructibility of the past [...] it always keeps some trace of the intermediate stages through which it has passed" (Gould 1970). Our experiment certainly did not return these organisms to their ancestral conditions. However, the private-droplet treatment was designed to place populations under selection for an ancestral phenotype, namely higher yield. Even as the private-droplet lineages derived from later generations of the LTEE recovered the high numerical yield of the LTEE ancestor, their phenotypic "return" did not involve a straightforward reversal of the steps in their evolutionary history. Rather, the mutational targets and metabolic patterns underlying their return were distinct from those evolutionary paths explored in the LTEE. By contrast, when under the same selection for increased numerical yield, the private-droplet descendants of the 0K emulsion ancestor exhibited genetic and metabolic changes distinct from the 20K and 60K emulsion lines, suggesting that changes during the first 20,000 generations of the LTEE had altered the potential for subsequent adaptation to the private-droplet regime Such contingency is a recurring feature of adaptation in biological systems (Blount et al. 2008; Card et al. 2019). However, we found no evidence of contingency in the later generations, as both the intermediate (20K) and late (60K) isolates from the LTEE readily recovered their ancestral form,

seemingly by taking similar steps. Therefore, past evolution need not invariably alter future evolution, even when the future involves a return to the past.

Acknowledgments. We thank Carrie Glenney and Brian Connelly for their help in developing the methods for the emulsion experiments, Taka Shimazu for his invaluable help in the lab, Katie Dickinson, Sonia Singhal, Peter Conlin, Hannah Jordt, Olivia Kosterliz, Emelyne Cunnington, and Herwig Bachmann for valuable discussions. This work was supported, in part, by the BEACON Center for the Study of Evolution in Action (NSF cooperative agreement DBI-0939454) and a grant from the National Science Foundation (DEB-1951307).

References

Agrawal, Anurag A., Jeffrey K. Conner, and Sergio Rasmann. 2010. "Tradeoffs and Negative Correlationsin Evolutionary Ecology BT - Evolution Since Darwin." *Evolution Since Darwin* (10):243–68.

Bachmann, H., M. Fischlechner, I. Rabbers, N. Barfa, F. Branco dos Santos, D. Molenaar, and B. Teusink. 2013. "Availability of Public Goods Shapes the Evolution of Competing Metabolic Strategies." *Proceedings of the National Academy of Sciences* 110(35):14302–7. doi: 10.1073/pnas.1308523110.

Barrick, J. E., Kauth, M. R., Strelioff, C. C., & Lenski, R. E. (2010). Escherichia coli rpoB mutants have increased evolvability in proportion to their fitness defects. *Molecular Biology and Evolution*, *27*(6), 1338–1347. https://doi.org/10.1093/molbev/msq024

Beardmore, R. E., Gudelj, I., Lipson, D. A., & Hurst, L. D. (2011). Metabolic trade-offs and the maintenance of the fittest and the flattest. *Nature*, *472*(7343), 342–346. https://doi.org/10.1038/nature09905

Blount, Z. D., C. Z. Borland, and R. E. Lenski. 2008. Historical contingency and the evolution of a key innovation in an experimental population of *Escherichia coli*. *Proceedings of the National Academy of Sciences, USA* 105: 7899-7906.

Card, K. J., T. LaBar, J. B. Gomez, and R. E. Lenski. 2019. Historical contingency in the evolution of antibiotic resistance after decades of relaxed selection. *PLOS Biology* 17: e3000397.

Carmona, Susy Beatriz, Fabián Moreno, Francisco Bolívar, Guillermo Gosset, and Adelfo Escalante. 2015. "Inactivation of the PTS as a Strategy to Engineer the Production of Aromatic Metabolites in Escherichia Coli." *Journal of Molecular Microbiology and Biotechnology* 25(2–3):195–208. doi: 10.1159/000380854.

Chou, H. H., Chiu, H. C., Delaney, N. F., Segrè, D., & Marx, C. J. (2011). Diminishing returns epistasis among beneficial mutations decelerates adaptation. *Science*, *332*(6034), 1190–1192. https://doi.org/10.1126/science.1203799

Deatherage D.E., Barrick J. E. 2014. "Identification of Mutations in Laboratory-Evolved Microbes from Next-Generation Sequencing Data Using Breseq. In: Sun L., Shou W. (Eds) Engineering and Analyzing Multicellular Systems. Methods in Molecular Biology (Methods and Protocols)." 1151:1–22. doi: 10.1007/978-1-4939-0554-6.

Escalante, Adelfo, Ania Salinas Cervantes, Guillermo Gosset, and Francisco Bolívar. 2012. "Current Knowledge of the Escherichia Coli Phosphoenolpyruvate-Carbohydrate Phosphotransferase System: Peculiarities of Regulation and Impact on Growth and Product Formation." *Applied Microbiology and Biotechnology* 94(6):1483–94. doi: 10.1007/s00253-012-4101-5.

Eshelman, C. M., Vouk, R., Stewart, J. L., Halsne, E., Lindsey, H. a, Schneider, S., … Kerr, B. (2010). Unrestricted migration favours virulent pathogens in experimental metapopulations: evolutionary genetics of a rapacious life history. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, *365*, 2503–2513. https://doi.org/10.1098/rstb.2010.0066

Gould, S. J. (1970). Dollo on Dollo ' s Law : Irreversibility and the Status of Evolutionary Laws Author (s): Stephen Jay Gould Source : Journal of the History of Biology , Vol . 3 , No . 2 (Autumn , 1970), pp . 189-212 Published by : Springer Stable URL : http://www.jsto. *J Hist Biol.*, *3*(2), 189–212.

Gounand, I., Daufresne, T., Gravel, D., Bouvier, C., Bouvier, T., Combe, M., … Mouquet, N. (2016). Size evolution in microorganisms masks trade-offs predicted by the growth rate hypothesis. *Proceedings of the Royal Society B: Biological Sciences*, *283*(1845), 20162272. https://doi.org/10.1098/rspb.2016.2272

Grant, N. A., Abdel Magid, A., Franklin, J., Dufour, Y., & Lenski, R. E. (2021). Changes in Cell Size and Shape During 50,000 Generations of Experimental Evolution with Escherichia coli. *Journal of Bacteriology*, (March). https://doi.org/10.1128/jb.00469-20

Guillaume, F., & Otto, S. P. (2012). Gene functional trade-offs and the evolution of pleiotropy. *Genetics*, *192*(4), 1389–1409. https://doi.org/10.1534/genetics.112.143214

Harcombe, W. R., Delaney, N. F., Leiby, N., Klitgord, N., & Marx, C. J. (2013). The Ability of Flux Balance Analysis to Predict Evolution of Central Metabolism Scales with the Initial Distance to the Optimum. *PLoS Computational Biology*, *9*(6). https://doi.org/10.1371/journal.pcbi.1003091

Kerr, B., Neuhauser, C., Bohannan, B. J. M., & Dean, A. M. (2006). Local migration promotes competitive restraint in a host-pathogen "tragedy of the commons." *Nature*, *442*(7098), 75–78. https://doi.org/10.1038/nature04864

Kimata, K., Takahashi, H., Inada, T., Postma, P., & Aiba, H. (1997). cAMP receptor protein-cAMP plays a crucial role in glucose-lactose diauxie by activating the major glucose transporter gene in Escherichia coli. *Proceedings of the National Academy of Sciences of the United States of America*, *94*(24), 12914–12919. https://doi.org/10.1073/pnas.94.24.12914

Leiby, N., Harcombe, W. R., & Marx, C. J. (2012). Multiple long-term, experimentally-evolved populations of Escherichia coli acquire dependence upon citrate as an iron chelator for optimal growth on glucose. *BMC Evolutionary Biology*, *12*(1). https://doi.org/10.1186/1471-2148-12-151

Leiby, N., & Marx, C. J. (2014). Metabolic Erosion Primarily Through Mutation Accumulation, and Not Tradeoffs, Drives Limited Evolution of Substrate Specificity in Escherichia coli. *PLoS Biology*, *12*(2). https://doi.org/10.1371/journal.pbio.1001789

Lenski Richard E. 2017. "Experimental Evolution and the Dynamics of Adaptation and Genome Evolution in Microbial Populations." *The ISME Journal* 1–14. doi: 10.1038/ismej.2017.69.

Lenski, Richard E., and Michael Travisano. 1994. "Dynamics of Adaptation and Diversification: A 10,000-Generation Experiment with Bacterial Populations." *Proceedings of the National Academy of Sciences of the United States of America* 91(15):6808–14. doi: 10.1073/pnas.91.15.6808.

Levin, B. R., Lipsitch, M., Perrot, V., Schrag, S., Antia, R., Simonsen, L., … Stewart, F. M. (1997). The population genetics of antibiotic resistance. *Clinical Infectious Diseases : An Official Publication of the Infectious Diseases Society of America*, *24 Suppl 1*, S9-16. Retrieved from http://www.ncbi.nlm.nih.gov/pubmed/8994776

Manhart, M., Adkar, B. V., & Shakhnovich, E. I. (2016). *Tradeoffs between microbial growth phases lead to frequency-dependent and non-transitive selection*. https://doi.org/http://dx.doi.org/10.1101/096453

Nam, Tae Wook, Seung Hyon Cho, Dongwoo Shin, Ja Hee Kim, Jin Young Jeong, Joon Hee Lee, Jung Hye Roe, Alan Peterkofsky, Sa Ouk Kang, Sangryeol Ryu, and Yeong Jae Seok. 2001. "The Escherichia Coli Glucose Transporter Enzyme IICBGlc Recruits the Global Repressor Mlc." *EMBO Journal* 20(3):491–98. doi: 10.1093/emboj/20.3.491.

New, Aaron M., Bram Cerulus, Sander K. Govers, Gemma Perez-Samper, Bo Zhu, Sarah Boogmans, Joao B. Xavier, and Kevin J. Verstrepen. 2014. “Different Levels of Catabolite Repression Optimize Growth in Stable and Variable Environments.” *PLoS Biology* 12(1):17–20. doi: 10.1371/journal.pbio.1001764.

Notley-mcrobb, Lucinda, Alison Death, and Thomas Ferenci. 2006. “The Relationship between External Glucose Concentration and CAMP Levels Inside.” *Biochemical Journal* (1997):1909–18. doi: 10.1099/00221287-143-6-1909.

Novak, Maja, Thomas Pfeiffer, Richard E. Lenski, Uwe Sauer, and Sebastian Bonhoeffer. 2006. “Experimental Tests for an Evolutionary Trade-off between Growth Rate and Yield in E. Coli.” *The American Naturalist* 168(2):242–51. doi: 10.1086/506527.

Pavličev, M., & Cheverud, J. M. (2015). Constraints Evolve: Context Dependency of Gene Effects Allows Evolution of Pleiotropy. *Annual Review of Ecology, Evolution, and Systematics*, *46*(1), 413–434. https://doi.org/10.1146/annurev-ecolsys-120213-091721

Peng, F., Widmann, S., Wünsche, A., Duan, K., Donovan, K. A., Dobson, R. C. J., … Cooper, T. F. (2018). Effects of Beneficial Mutations in pykF Gene Vary over Time and across Replicate Populations in a Long-Term Experiment with Bacteria. *Molecular Biology and Evolution*, *35*(1), 202–210. https://doi.org/10.1093/molbev/msx279

Pennings, P. S., Ogbunugafor, C. B., & Hershberg, R. (2020). Reversion is most likely under high mutation supply, when compensatory mutations don’t fully restore fitness costs. *BioRxiv*. https://doi.org/10.1101/2020.12.28.424568

Pfeiffer, Thomas, Stefan Schuster, and Sebastian Bonhoeffer. 2001. “Cooperation and Competition in the Evolution of ATP-Producing Pathways.” 292(April).

Robinson, James T., Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, and Jill P. Mesirov. 2011. “Integrative Genomics Viewer.” *Nature Biotechnology* 29(1):24–26. doi: 10.1038/nbt0111-24.

Rose, M. (1982). *Antagonistic pleiotropy, dominance, and genetic variation*. 48*(2501).

Rose, M. R. (1985). Life history evolution with antagonistic pleiotropy and overlapping generations. *Theoretical Population Biology*, *28*(3), 342–358. https://doi.org/10.1016/0040-5809(85)90034-6

Schank, J., & Wimsatt, W. (2011). *Generative Entrenchment and Evolution Author (s): Jeffrey C . Schank and William C . Wimsatt Source : PSA : Proceedings of the Biennial Meeting of the Philosophy of Science Association , Vol . 1986 , Volume Two : Symposia and Invited Papers (1986), pp. 1986*(1986), 33–60.

Shah, P., McCandlish, D. M., & Plotkin, J. B. (2015). Contingency and entrenchment in protein evolution under purifying selection. *Proceedings of the National Academy of Sciences of the United States of America*, *112*(25), E3226–E3235. https://doi.org/10.1073/pnas.1412933112

Stearns, S. C. 1989. “Trade-Offs in Life-History Evolution Published by: British Ecological Society Stable URL: Https://Www.Jstor.Org/Stable/2389364 Trade-Offs in Life-History Evolution.” *Functional Ecology* 3(3):259–68.

Stevenson, K., McVey, A. F., Clark, I. B. N., Swain, P. S., & Pilizota, T. (2016). General calibration of microbial growth in microplate readers. *Scientific Reports*, *6*(December), 4–10. https://doi.org/10.1038/srep38828

Szenk, Mariola, Ken A. Dill, and Adam M. R. de Graff. 2017. “Why Do Fast-Growing Bacteria Enter Overflow Metabolism? Testing the Membrane Real Estate Hypothesis.” *Cell Systems* 5(2):95–104. doi: 10.1016/j.cels.2017.06.005.

Tenaillon, O., J. E. Barrick, N. Ribeck, D. E. Deatherage, J. L. Blanchard, A. Dasgupta, G. C. Wu, S. Wielgoss, S. Cruveiller, C. Médigue, D. Schneider, and R. E. Lenski. 2016. Tempo and mode of genome evolution in a 50,000-generation experiment. *Nature* 536: 165-170.

Teotónio, H., & Rose, M. R. (2000). Variation in the reversibility of evolution. *Nature*, *408*(6811), 463–466. https://doi.org/10.1038/35044070

Travisano, Michael, and Richard E. Lenski. 1996. "Long-Term Experimental Evolution in Escherichia Coli. IV. Targets of Selection and the Specificity of Adaptation." *Genetics* 143(1):15–26. doi: 10.1007/s000360050052.

van Tatenhove-Pel, R. J., Zwering, E., Boreel, D. F., Falk, M., van Heerden, J. H., Kes, M. B. M. J., … Bachmann, H. (2021). Serial propagation in water-in-oil emulsions selects for Saccharomyces cerevisiae strains with a reduced cell size or an increased biomass yield on glucose. *Metabolic Engineering*, *64*(November 2020), 1–14. https://doi.org/10.1016/j.ymben.2020.12.005

Vasi, F., Travisano, M., & Lenski, R. E. (1994). *Long-Term Experimental Evolution in Escherichia coli . II . Changes in Life-History Traits During Adaptation to a Seasonal Environment Author (s): Farida Vasi , Michael Travisano and Richard E . Lenski Published by : The University of Chicago Press for*. *144*(3), 432–456.

Velicer, G. J. (1999). Pleiotropic effects of adaptation to a single carbon source for growth on alternative substrates. *Applied and Environmental Microbiology*, *65*(1), 264–269. https://doi.org/10.1128/aem.65.1.264-269.1999

Wiser, Michael J., Noah Ribeck, and Richard E. Lenski. 2013. "Long-Term Dynamics of Adaptation in Asexual Populations." *Science (New York, N.Y.)* 342(6164):1364–67. doi: 10.1126/science.1243357.

Woods, R., D. Schneider, C. L. Winkworth, M. A. Riley, and R. E. Lenski. 2006. Tests of parallel molecular evolution in a long-term experiment with *Escherichia coli*. *Proceedings of the National Academy of Sciences, USA* 103:9107-9112.

Xia, Tian, Neeraj Sriram, Sarah A. Lee, Ronni Altman, Jeffrey L. Urbauer, Elliot Altman, and Mark A. Eiteman. 2017. "Glucose Consumption in Carbohydrate Mixtures by Phosphotransferase-System Mutants of Escherichia Coli." *Microbiology (United Kingdom)* 163(6):866–77. doi: 10.1099/mic.0.000480.

Supplement for

A return to form: Novel mutations keep ancestral traits accessible despite evolutionary history

Katrina van Raay, Sergey Stolyar, Jordana Sevigny, Jeremy A. Draghi, Richard E. Lenski, Christopher J. Marx, Benjamin Kerr, and Luis Zaman

Page **Contents**

Supplemental Experimental Methods

Microscopy for droplet size

Emulsion droplets were imaged using an inverted water-immersion Olympus confocal microscope. Images were made by pipetting 40 µl of emulsion onto a glass slide with a glass coverslip on the bottom and part of the slide carved out to make a cavity. Images were analyzed in FIJI (version 2.0.0-rc-69/1.52n), and the maximum width of 238 droplets was measured (in µm) using the line tool.

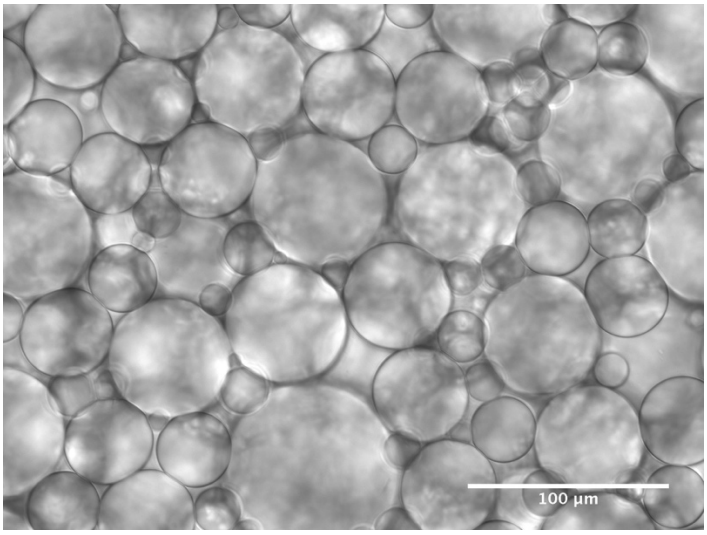

Figure S1: An example image of emulsion droplets using confocal microscopy.

Calculating droplet statistics

Droplets in the emulsions were not uniform in size (see Fig. S1). We can use the distribution of droplet volumes to estimate how many cells, on average, were in an occupied droplet for each of our emulsion treatments (shared or private).

We can describe the distribution of droplet volumes with a probability density function $p(v)$. Suppose that the density of cells before emulsification is given by λ (CFU/ml). The probability that a droplet has c cells, $\delta(c)$, is:

$$\delta(c) = \int_0^\infty \frac{e^{-\lambda v}(\lambda v)^c}{c!} p(v)dv.$$

The mean number of cells per droplet would then be:

$$\mu = \sum_{c=0}^{\infty} c\delta(c).$$

This summation can be reframed as follows:

$$\mu = \sum_{c=1}^{\infty} \int_0^\infty \frac{e^{-\lambda v}(\lambda v)^c}{(c-1)!} p(v)dv,$$

$$\mu = \int_0^\infty p(v)dv \left\{ \sum_{c=1}^{\infty} \frac{e^{-\lambda v}(\lambda v)^c}{(c-1)!} \right\},$$

$$\mu = \int_0^\infty p(v)dv \left\{ \lambda v \sum_{c=1}^{\infty} \frac{e^{-\lambda v}(\lambda v)^{c-1}}{(c-1)!} \right\},$$

$$\mu = \int_0^\infty p(v)dv \left\{ \lambda v \sum_{i=0}^{\infty} \frac{e^{-\lambda v}(\lambda v)^i}{i!} \right\},$$

$$\mu = \int_0^\infty \lambda v p(v)dv,$$

$$\mu = \lambda \bar{v},$$

where $\bar{v}$ is the average volume of a droplet.

If we condition on the presence of at least one cell in the droplet, we can recompute the mean number of cells, which we call μ_{occupied}. In that case, we obtain:

$$\mu_{\text{occupied}} = \frac{\sum_{c=1}^{\infty} c\delta(c)}{\sum_{c=1}^{\infty} \delta(c)},$$

$$\mu_{\text{occupied}} = \frac{\mu}{\int_0^{\infty} p(v)dv \left\{\sum_{c=1}^{\infty} \frac{e^{-\lambda v}(\lambda v)^c}{c!}\right\}},$$

$$\mu_{\text{occupied}} = \frac{\mu}{\int_0^{\infty} p(v)dv \left\{-e^{-\lambda v} + \sum_{c=0}^{\infty} \frac{e^{-\lambda v}(\lambda v)^c}{c!}\right\}},$$

$$\mu_{\text{occupied}} = \frac{\mu}{\int_0^{\infty} p(v)dv \{1 - e^{-\lambda v}\}},$$

$$\mu_{\text{occupied}} = \frac{\int_0^{\infty} \lambda v p(v)dv}{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv}.$$

Finally, the fraction of occupied droplets that have only a single cell, ρ_{single}, is given by:

$$\rho_{\text{single}} = 1 - \frac{\sum_{c=2}^{\infty} \delta(c)}{\sum_{c=1}^{\infty} \delta(c)},$$

$$\rho_{\text{single}} = 1 - \frac{\int_0^{\infty} p(v)dv \left\{\sum_{c=2}^{\infty} \frac{e^{-\lambda v}(\lambda v)^c}{c!}\right\}}{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv},$$

$$\rho_{\text{single}} = 1 - \frac{\int_0^{\infty} p(v)dv \left\{-e^{-\lambda v} - e^{-\lambda v}\lambda v + \sum_{c=0}^{\infty} \frac{e^{-\lambda v}(\lambda v)^c}{c!}\right\}}{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv},$$

$$\rho_{\text{single}} = 1 - \frac{\int_0^{\infty} p(v)dv \left\{1 - e^{-\lambda v} - e^{-\lambda v}\lambda v\right\}}{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv},$$

$$\rho_{\text{single}} = 1 - \frac{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv - \int_0^{\infty} e^{-\lambda v}\lambda v p(v)dv}{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv},$$

$$\rho_{\text{single}} = \frac{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv - 1 + \int_0^{\infty} e^{-\lambda v} p(v)dv + \int_0^{\infty} e^{-\lambda v}\lambda v p(v)dv}{1 - \int_0^{\infty} e^{-\lambda v} p(v)dv},$$

$$\rho_{\text{single}} = \frac{\int_0^\infty e^{-\lambda v}\lambda v p(v)dv}{1 - \int_0^\infty e^{-\lambda v}p(v)dv}.$$

If we can determine the values of the following integrals

$$\mu = \int_0^\infty \lambda v p(v)dv,$$

$$\alpha = \int_0^\infty e^{-\lambda v}p(v)dv,$$

$$\beta = \int_0^\infty e^{-\lambda v}\lambda v p(v)dv,$$

then we can compute $\mu_{\text{occupied}} = \frac{\mu}{1-\alpha}$ and $\rho_{\text{single}} = \frac{\beta}{1-\alpha}$.

We have an empirically determined distribution of droplet diameters, measured in μm. Figure S2 shows a distribution of droplet diameters measured on 238 droplets, with the droplets binned in 10-μm intervals. Suppose we consider one of these intervals, ranging from diameter D_{low} to D_{high}. Letting the volume of a droplet with diameter D be given by $V(D)$, the average droplet volume in the focal interval will be:

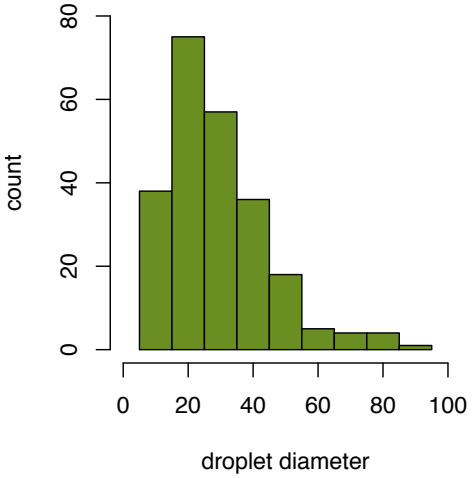

Figure S2: The distribution of droplet diameters from microscopic analysis.

$$V(D_{\text{low}}, D_{\text{high}}) = \frac{\int_{D_{\text{low}}}^{D_{\text{high}}} V(D)dD}{D_{\text{high}} - D_{\text{low}}}.$$

Because $V(D) = \frac{\pi}{6}D^3$,

$$V(D_{\text{low}}, D_{\text{high}}) = \left(\frac{\pi}{6}\right)\frac{\int_{D_{\text{low}}}^{D_{\text{high}}} D^3 dD}{D_{\text{high}} - D_{\text{low}}},$$

$$V(D_{\text{low}}, D_{\text{high}}) = \left(\frac{\pi}{6}\right)\frac{\frac{{D_{\text{high}}}^4}{4} - \frac{{D_{\text{low}}}^4}{4}}{D_{\text{high}} - D_{\text{low}}},$$

$$V(D_{\text{low}}, D_{\text{high}}) = \left(\frac{\pi}{24}\right)\frac{\left({D_{\text{high}}}^2 - {D_{\text{low}}}^2\right)\left({D_{\text{high}}}^2 + {D_{\text{low}}}^2\right)}{D_{\text{high}} - D_{\text{low}}},$$

$$V(D_{\text{low}}, D_{\text{high}}) = \left(\frac{\pi}{24}\right)\frac{\left(D_{\text{high}} - D_{\text{low}}\right)\left(D_{\text{high}} + D_{\text{low}}\right)\left({D_{\text{high}}}^2 + {D_{\text{low}}}^2\right)}{D_{\text{high}} - D_{\text{low}}},$$

$$V(D_{\text{low}}, D_{\text{high}}) = \left(\frac{\pi}{24}\right)\left(D_{\text{high}} + D_{\text{low}}\right)\left({D_{\text{high}}}^2 + {D_{\text{low}}}^2\right),$$

If the droplet count for our focal bin is n and the total number of droplets is N, then we will say the probability of falling in the interval is

$$P(D_{\text{low}}, D_{\text{high}}) = \frac{n}{N}$$

So, let us designate the diameter boundaries of the bins as D_0, D_1, D_2, D_3, … D_{B}. We can then use the following approximations:

$$\mu = \int_0^\infty \lambda v p(v) dv \approx \sum_{i=0}^{\text{B}-1} \lambda\{V(D_i, D_{i+1})\}\{P(D_i, D_{i+1})\},$$

$$\alpha = \int_0^\infty e^{-\lambda v} p(v) dv \approx \sum_{i=0}^{\text{B}-1} e^{-\lambda\{V(D_i, D_{i+1})\}}\{P(D_i, D_{i+1})\},$$

$$\beta = \int_0^\infty e^{-\lambda v} \lambda v p(v) dv \approx \sum_{i=0}^{\text{B}-1} e^{-\lambda\{V(D_i, D_{i+1})\}} \lambda\{V(D_i, D_{i+1})\}\{P(D_i, D_{i+1})\}.$$

There is one remaining issue, namely the units. The standard units for cell density are CFU/ml, whereas the volume unit for droplets here is μm^3. However, because there are 10^{12} μm^3 per milliliter, we simply divide the cubic-micron volume by 10^{12} in the integrals above.

For the private-droplet treatment, the cell density was 2 x 10^6 CFU/ml and the statistics are:

$$\mu = 0.05 \text{ cells/droplet}$$
$$\mu_{\text{occupied}} = 1.12 \text{ cells/droplet}$$
$$\rho_{\text{single}} = 0.90$$

For the shared-droplet treatment, the cell density was 5 x 10^7 CFU/ml and the statistics are:

$$\mu = 1.33 \text{ cells/droplet}$$
$$\mu_{\text{occupied}} = 2.99 \text{ cells/droplet}$$
$$\rho_{\text{single}} = 0.45$$

Therefore, in the private-droplet treatment, 90% of all occupied droplets have only a single cell, with an average of 1.12 cells per occupied droplet. In the shared droplet treatment, by contrast, 45% of the occupied droplets have a single cell, with an average of almost 3 cells per occupied droplet.

Calculating number of cell divisions in droplets

Here, we focus on the number of cell divisions in the emulsion system if the initial cell density is λ_i (CFU/ml). Suppose also that, at stationary phase, any droplet that was initialized with one or more cells will reach a final density of λ_f (where $\lambda_f > \lambda_i$). Focusing on one milliliter of volume, the initial number of cells is $c_i = \lambda_i$. The final number of cells (c_f) will not generally be λ_f, however, because some droplets are not occupied. The total number of cell divisions is simply $d = c_f - c_i$, because every cell produced requires a cell division. The number of generations, measured as population doublings, is then given by $g = \log_2(c_f/c_i)$.

So, to find the number of cell divisions, we focus on finding c_f. For a droplet with volume v (measured in ml), the probability that it gets colonized by one or more cells (given the initial density λ_i) is

$$\chi(v) = 1 - e^{-\lambda_i v}$$

We assume that all colonized droplets reach a final density of λ_f. Further, we assume that every colonized droplet is initiated at a cell density less than λ_f, such that no cell death occurs before reaching the final density. These assumptions are reasonable for the initial densities and droplet-size distribution of our experiment. The final number of cells in a single milliliter is

$$c_f = \frac{\int_0^\infty \chi(v)\lambda_f v p(v) dv}{\int_0^\infty v p(v) dv}$$

Noting $\bar{v} = \int_0^\infty v p(v) dv$, we can simplify this expression

$$c_f = \frac{\lambda_f \int_0^\infty (1 - e^{-\lambda_i v}) v p(v) dv}{\bar{v}} = \frac{\lambda_f \bar{v} - \lambda_f \int_0^\infty e^{-\lambda_i v} v p(v) dv}{\bar{v}}$$

Letting $\beta = \int_0^\infty e^{-\lambda_i v} \lambda_i v p(v) dv$, we have:

$$c_f = \lambda_f - \frac{\lambda_f \beta}{\lambda_i \bar{v}}$$

Therefore, the number of cell divisions is

$$d = \lambda_f \left(1 - \frac{\beta}{\lambda_i \bar{v}}\right) - \lambda_i$$

Using terminology from the previous section, we obtain

$$\bar{v} = \int_0^\infty v p(v) dv \approx \sum_{i=0}^{\mathrm{B}-1} \{V(D_i, D_{i+1})\}\{P(D_i, D_{i+1})\}$$

$$\beta = \int_0^\infty e^{-\lambda_i v} \lambda_i v p(v) dv \approx \sum_{i=0}^{\mathrm{B}-1} e^{-\lambda_i \{V(D_i, D_{i+1})\}} \lambda_i \{V(D_i, D_{i+1})\}\{P(D_i, D_{i+1})\}$$

Using the droplet distribution from Figure S2, in Figure S3 we graph the logarithm (base 10) of the number of cell divisions for a range of initial cell densities (from 10^6 to 10^8 CFU/ml) and three final cell densities (5×10^8, 1×10^9, and 2.5×10^9 CFU/ml). These final densities are in the range of the final density in our experimental system. As the initial cell density rises, two effects occur. First, the number of droplets that are colonized increases. Second, the number of cells per occupied droplet increases. The first factor leads to more cell divisions, whereas the second factor leads to fewer divisions. As the initial cell density is increased from very low values, the first effect dominates the second, and the number of cell divisions increases. However, as the initial cell density continues to increase towards the final cell density, the first effect becomes greatly muted (as nearly all the droplets are occupied) and the second effect dominates, so that the number of cell divisions decreases (this effect can be seen in the curve for the 5×10^8 final density). Figure S3 also shows that, for each of the final cell densities, the shared-droplet treatment has more cell divisions than the private-droplet treatment.

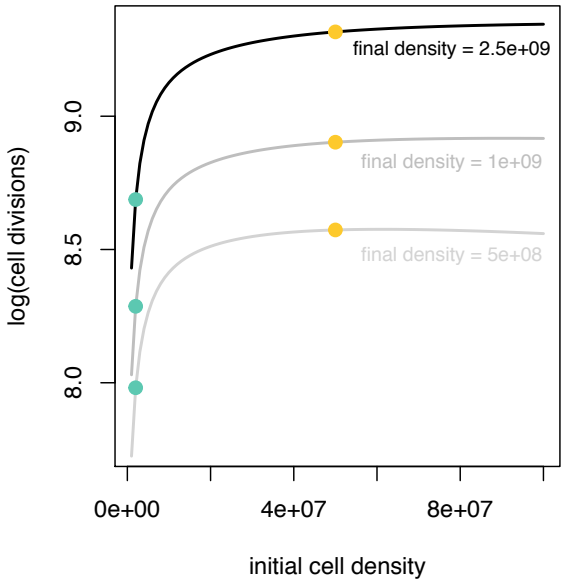

Figure S3: The logarithm of the number of cell divisions (per ml) as a function of initial cell density (CFU/ml) for three different final cell densities (CFU/ml). The private-droplet and shared-droplet treatments are defined by their initial densities, and they are indicated by teal and yellow points, respectively.

Calibrating OD readings and cell counts

In order to facilitate calculating the dilutions to perform in our experimental transfers, we built a JavaScript calculator that converted each OD_{595} reading to a cell-density estimate. We used that

estimate to calculate the appropriate dilutions and volume of media to add for each population. This tool is available online: https://kerrlab.github.io/Project_ET/

To calibrate the OD to CFU relationship, we initially performed three replicate mock competitions between REL606 and REL11638 in emulsions. We used this mixture to capture some of the variation in OD values between clones starting from different evolutionary timepoints in our experiments. After overnight growth, the emulsions were broken and a dilution series was prepared on a 96-well plate. The OD values of these dilutions were measured, and multiple dilutions were plated to obtain CFU counts. With known dilution series and CFUs, we had a large range of OD values and CFU counts. Using a second-degree polynomial fit to these data, the calculator estimated the CFU count from an entered OD value. The calculator was further refined after collecting additional data with corresponding CFU and OD values.

Optical density and changes in cell size

Optical density (OD) is a measurement of the amount of light absorbed by a sample. As cells in a culture multiply, the culture becomes more turbid and its optical density increases. Cell size can also affect OD. Generally, for the same density of cells, a culture of smaller cells will have lower OD (Stevenson et al. 2016). Because we used OD to compute the dilutions for transfers in our evolution experiment (see previous section), and because we did not measure cell size over the course of the experiment, evolutionary changes in cell size might inadvertently affect our dilution scheme. In the private-droplet treatment cells evolved smaller size, such that our conversion of OD to cell density would have led to an underestimate. This underestimation would have led to a higher initial cell density each transfer, and it would thereby have *reduced*

the selection for numerical yield. Thus, our finding of increased numerical yield in the private-droplet treatment is conservative, having occurred despite the weakened selection. Similarly, the slight increases in cell size in the shared-droplet treatment would have led to a lower initial cell density each transfer, thereby reducing selection for competitive ability, and again making the inferences conservative.

Focal strains

We used six evolved isolates from the private-droplet treatment that had acquired one or more mutations during the experiment (as determined by whole-genome sequencing) in the cell size, numerical yield, and metabolic assays. We excluded two private-droplet isolates that did not acquire mutations or show changes in cell size or numerical yield from additional assays and analyses. For the shared-droplet treatment, we chose one isolate at random from the three replicates for each emulsion ancestor (0K, 20K and 60K) for further analysis.

Growth and cell size assays

To assess population growth, we revived frozen isolates in 5 ml of DM1000 and grew them overnight at 37 °C with shaking. We added 5 µl of each overnight culture to 10 ml of Isoton and ran samples through a 30-µm aperture on a MSE4 Beckman Coulter Counter to estimate cell density. The cultures were then diluted to 5 x 10^7 cells/ml, and 30 emulsions were set up for each isolate to allow 3 replicates at 10 timepoints each. The emulsion populations were allowed to grow without shaking at 37 °C, and they were then destructively sampled at 0, 1, 2, 3, 4, 5, 6, 8, 10, and 24 hours. The cell density and median cell volume (μm^3) were recorded at each timepoint using the Coulter Counter (Figure S4).

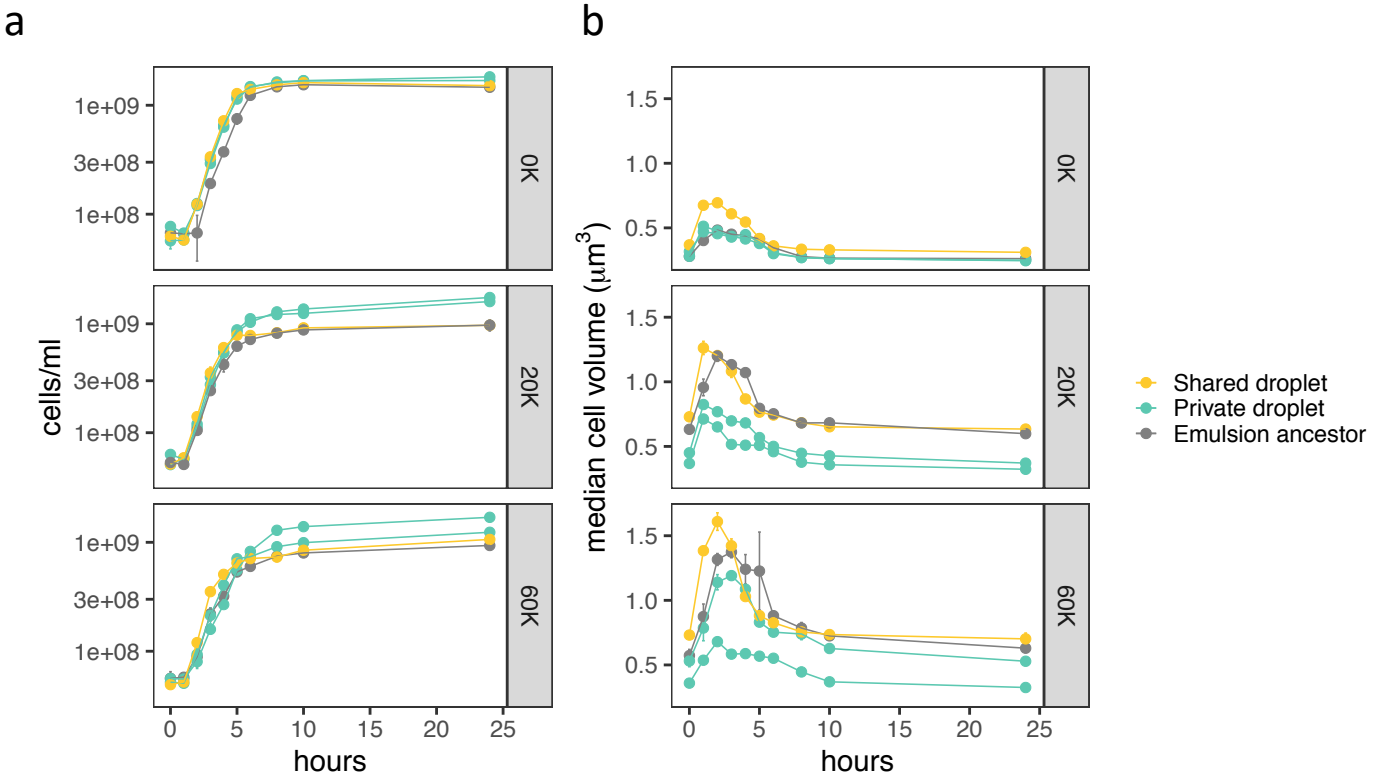

Figure S4: Population growth and median cell volume. **a)** Cell density over a 24-hour transfer cycle. **b)** Median cell volume over the same 24-hour cycle. In both panels, each curve shows one strain colored according to the labels (at right). Each point is the average of three replicates, and error bars are SEM.

Statistical and Quantitative Methods

Statistical analysis

All statistical analyses were performed using R (Version 3.6.0). Means and standard error of the mean were calculated for competitive fitness, cell size, and cell number for each isolate at each timepoint, usually based on three replicate samples. For each of the above traits, we ran unpaired, two-tailed t-tests comparing the emulsion-evolved strains with their emulsion ancestors (for 0K, 20K, and 60K lineages).

To compare the sets of descendants from the private-droplet and shared-droplet treatments, we used the mean values for each isolate across all technical replicates. We additionally took the average value for the two private-droplet lines derived from each emulsion ancestor (0K, 20K, and 60K). Pairing each private-droplet average with the corresponding shared-droplet value based on the common emulsion ancestor, we performed one-tailed, paired t-tests on the mean values of competitive fitness, cell size, and numerical yield. We used one-tailed tests here because our alternative hypotheses were directional.

Misalignment analysis

We consider a general case in which experimental evolution occurred in two treatments ($\boldsymbol{B}$ and $\boldsymbol{C}$) and we are tracking two phenotypic traits (x and y) in multiple lines for each treatment. Treatment $\boldsymbol{B}$ selects for increased values of trait x, while treatment $\boldsymbol{C}$ selects for increased values of trait y.

We assume that the vector $\vec{a} = \langle x_a, y_a \rangle$ describes the phenotype of the common ancestor of all the evolved lines in both treatments. In line ℓ of the $\boldsymbol{B}$ treatment, the evolved phenotype is denoted $\vec{b}_\ell = \langle x_{b_\ell}, y_{b_\ell} \rangle$. Similarly, in line ℓ of the $\boldsymbol{C}$ treatment, the evolved phenotype is denoted $\vec{c}_\ell = \langle x_{c_\ell}, y_{c_\ell} \rangle$. We start by rewriting all evolved phenotypes as deviations from their common ancestor. Specifically, line ℓ of the $\boldsymbol{B}$ and $\boldsymbol{C}$ treatments are expressed as:

$$\tilde{b}_\ell = \vec{b}_\ell - \vec{a} = \langle x_{b_\ell} - x_a, y_{b_\ell} - y_a \rangle = \langle \tilde{x}_{b_\ell}, \tilde{y}_{b_\ell} \rangle,$$
$$\tilde{c}_\ell = \vec{c}_\ell - \vec{a} = \langle x_{c_\ell} - x_a, y_{c_\ell} - y_a \rangle = \langle \tilde{x}_{c_\ell}, \tilde{y}_{c_\ell} \rangle,$$

respectively. Suppose there are m and n lines in treatments $\boldsymbol{B}$ and $\boldsymbol{C}$, respectively. We denote the maximal deviations in the two dimensions as:

$$\tilde{x}_{\max} = \max\{\tilde{x}_{b_1}, \tilde{x}_{b_2}, \ldots \tilde{x}_{b_m}, \tilde{x}_{c_1}, \tilde{x}_{c_2}, \ldots \tilde{x}_{c_n}\},$$
$$\tilde{y}_{\max} = \max\{\tilde{y}_{b_1}, \tilde{y}_{b_2}, \ldots \tilde{y}_{b_m}, \tilde{y}_{c_1}, \tilde{y}_{c_2}, \ldots \tilde{y}_{c_n}\}.$$

We then express the evolved phenotypic deviations as proportions of these maximal deviations. Specifically, line ℓ of the $\boldsymbol{B}$ and $\boldsymbol{C}$ treatments, are expressed as:

$$\hat{b}_\ell = \langle \frac{\tilde{x}_{b_\ell}}{\tilde{x}_{\max}}, \frac{\tilde{y}_{b_\ell}}{\tilde{y}_{\max}} \rangle = \langle \hat{x}_{b_\ell}, \hat{y}_{b_\ell} \rangle,$$
$$\hat{c}_\ell = \langle \frac{\tilde{x}_{c_\ell}}{\tilde{x}_{\max}}, \frac{\tilde{y}_{c_\ell}}{\tilde{y}_{\max}} \rangle = \langle \hat{x}_{c_\ell}, \hat{y}_{c_\ell} \rangle.$$

Consider the example in Figure S5a, where there are 3 lines of treatment $\boldsymbol{B}$ and 6 lines of treatment $\boldsymbol{C}$. Generally, we see that the treatment $\boldsymbol{B}$ lines have high $\hat{x}$ values and low $\hat{y}$ values, while the treatment $\boldsymbol{C}$ lines have high $\hat{y}$ values and low $\hat{x}$ values. Of course, the high values are expected because the two sets of lines were selected for increases in those dimensions. It is the low values that are our focus here.

Figure S5b shows one statistic to summarize the tendency for relatively high gains in one dimension to correspond to relatively low gains (or even losses) in another dimension. Essentially, we draw a "summary vector" taking the average angle of all the vectors in a treatment (thick lines), and the angular distance between the two summary vectors gives a misalignment statistic that we call θ^* (in pink). We note that when high gains in one dimension lead to low gains (or losses) in the other dimension, this statistic will be larger.

In Figure S6, we collect all the $\hat{x}$ values (Fig. S6a) and all the $\hat{y}$ values (Fig. S6b) of the lines from both treatments, and we plot them along the corresponding axes.

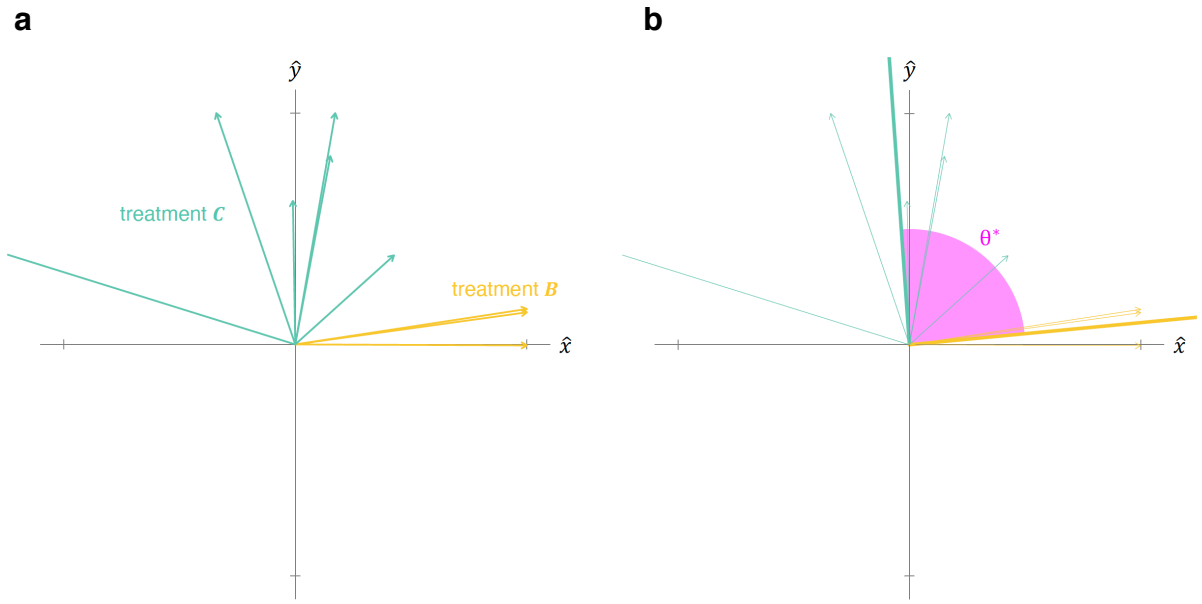

Figure S5: (a) The rescaled phenotypic deviations of evolved lines from two treatments (3 lines from treatment B and 6 lines from treatment C) are shown as vectors, with the origin representing the common ancestral reference. The notches on the abscissa and ordinate correspond to values of 1 and -1. One of the vectors for treatment C has a very low x-value (which is allowed), and thus we do not show its arrow head. **(b)** The summary vectors (thick lines) take the average angle of all the vectors for the replicate lines in a treatment. The angular distance between the two summary vectors (θ^*, in pink) is our misalignment statistic.

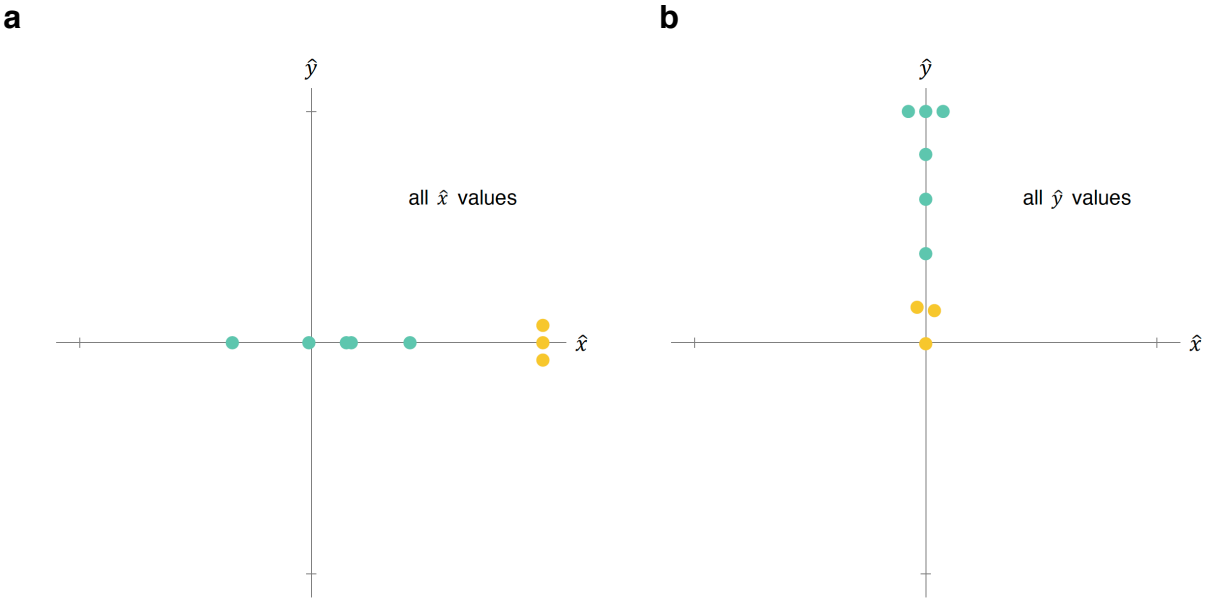

Figure S6: (a) Using the case illustrated in Fig. S5a, the $\hat{x}$ values are shown, with similar or equal $\hat{x}$ values separated to keep track of all values. We note that one $\hat{x}$ value (corresponding to the vector from Fig. S5a with a very low x-value) is not shown. **(b)** Using the same case, all $\hat{y}$ values are shown, again with similar or equal $\hat{y}$ values separated for visualization.

Consider the following null hypothesis: when there is not selection for the trait along the abscissa, all of the possible $\hat{x}$ values in Fig. S6a are equally likely; and when there is not selection for the trait along the ordinate, all of the possible $\hat{y}$ values in Fig. S6b are equally likely. We can then use a bootstrapping approach to gauge whether our actual statistic (θ^*) is more extreme than angles that we would expect under this null hypothesis. To do so, we fix the $\hat{x}$ values of the treatment $\boldsymbol{B}$ lines and resample with replacement three $\hat{y}$ values from Fig. S6b. Similarly, we fix the $\hat{y}$ values of the treatment $\boldsymbol{C}$ lines and resample with replacement six $\hat{x}$ values from Fig. S6a. A θ statistic can then be computed on the vectors resulting from the resampling. We performed the resampling and calculation 10,000 times. Figure S7 shows 25 of these iterations, where blue indicates a value of θ smaller than our actual statistic θ^* (Fig. S5b) and red indicates a value of θ larger than θ^*.

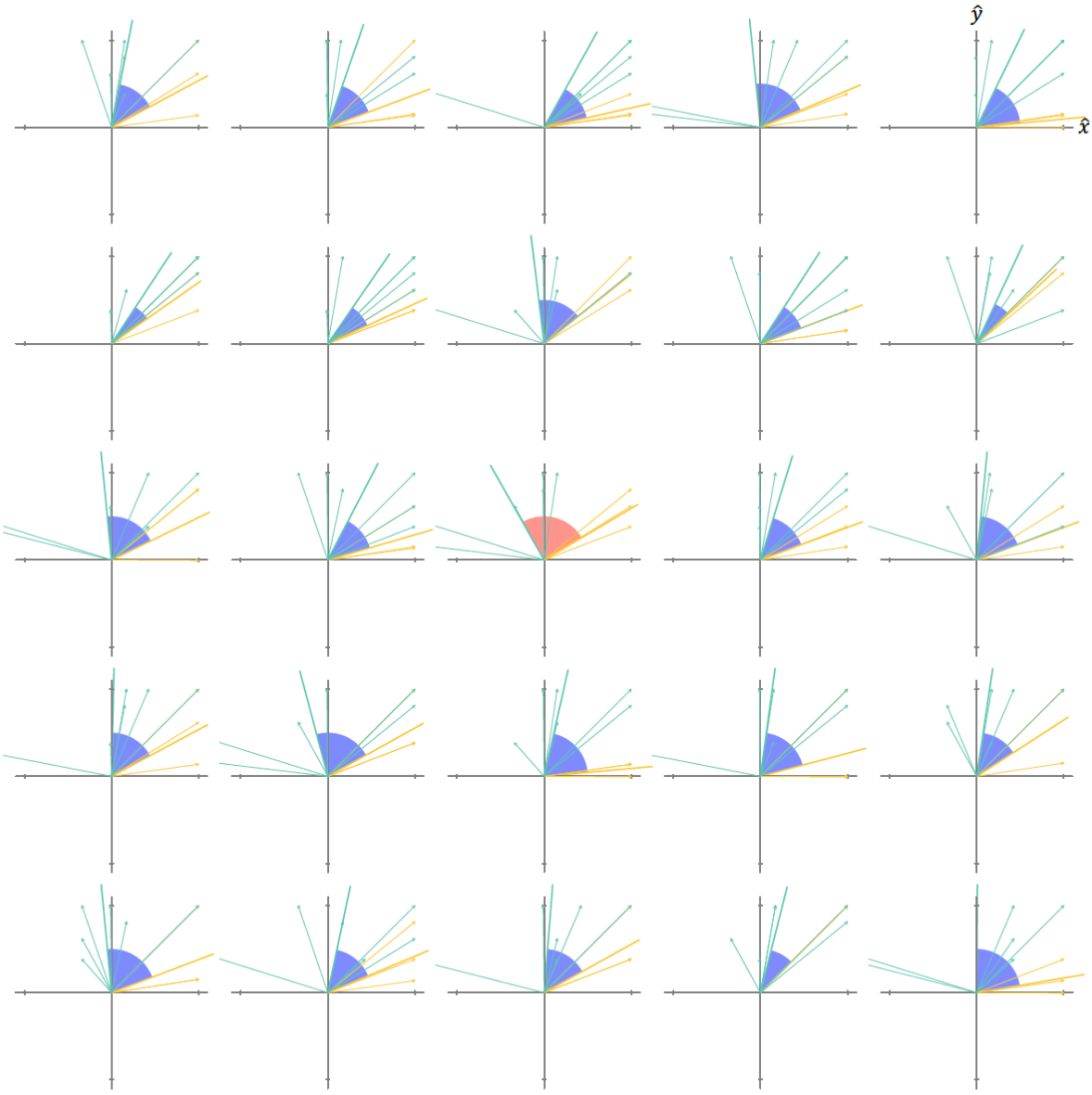

Figure S7: The angle between summary vectors (θ) is shown for 25 iterations of a bootstrapping scheme, in which treatment $\boldsymbol{B}$ vectors resample $\hat{y}$ values and treatment $\boldsymbol{C}$ vectors resample $\hat{x}$ values. When this resampled angle is smaller than θ^*, it is colored in blue; when larger, it is colored in red.

For our case study here, we see that the θ from resampled values is more extreme than the actual statistic θ^* only 3.3% of the time (Figure S8). Thus, by convention, we would reject the null hypothesis. In this case, we would say that the responses to selection in the two treatments is significantly misaligned.

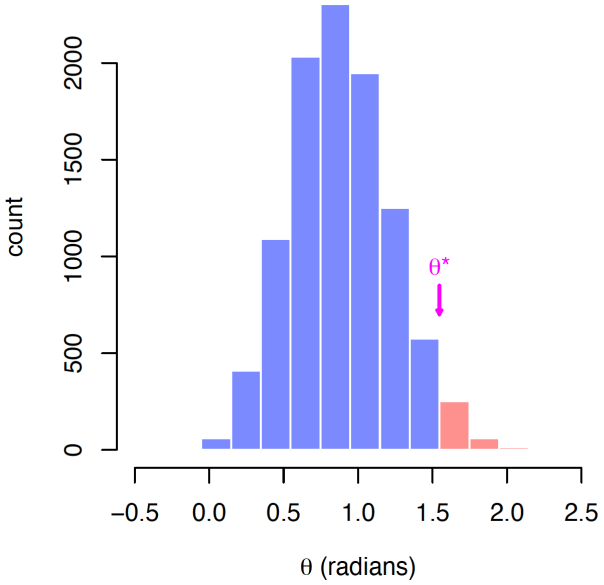

Figure S8: The distribution of angles resulting from our bootstrapping procedure. Only 3.3% of the angles from resampled data are more extreme than our actual misalignment statistic (θ^*). The red portion of the distribution corresponds to larger angles than the actual statistic, and the blue corresponds to smaller angles.

Our case study uses the actual data from our emulsion evolution experiments. However, there are three caveats that must be made clear. First, for each emulsion ancestor (0K, 20K, and 60K) the evolved competitive fitness (abscissa) and numerical yield (ordinate) in the shared-droplet and private-droplet treatments were rescaled relative to the phenotypic values of their common ancestor. Thus, the 0K descendants were rescaled relative to the 0K ancestor, the 20K descendants relative to the 20K ancestor, and the 60K descendants relative to the 60K ancestor. Second, all of the rescaled data were pooled for our misalignment analysis (including six private-droplet lines and three shared-droplet lines). Third, we note that these data are from only part of the experiment. The only private-droplet lines that we analyzed were those exhibiting a mutation called by our first pass of genomic sequencing, and each shared-droplet descendant line was randomly picked from the three shared-droplet descendant lines for each emulsion ancestor. With these caveats in mind, the analysis is consistent with a misalignment between competitive ability and numerical yield.

List of Strains and Supplemental Results

STRAIN (BK #)	LTEE SOURCE GENERATION	LTEE ID	TREATMENT (THIS STUDY)	PHENOTYPIC ASSAYS RUN?
427	0K	REL 606	None	Yes
429	20K	REL 8597	None	Yes
430	60K	REL 11700	None	Yes
431	0K	NA	Private droplet	Yes
432	0K	NA	Private droplet	No
433	0K	NA	Private droplet	Yes
437	20K	NA	Private droplet	Yes
438	20K	NA	Private droplet	Yes
439	20K	NA	Private droplet	No
441	60K	NA	Private droplet	Yes
442	60K	NA	Private droplet	Yes
443	0K	NA	Shared droplet	Yes
445	20K	NA	Shared droplet	Yes
446	20K	NA	Shared droplet	No
447	60K	NA	Shared droplet	No
448	60K	NA	Shared droplet	Yes
449	60K	NA	Shared droplet	No

Table S1: Strains used and generated in this study.

STRAIN (BK #)	LTEE GEN.	EVOL-UTION	COMPETITIVE ABILITY	Δ COMPETITIVE ABILITY	YIELD (CELLS/ ML)	Δ YIELD (CELLS/ML)	CELL VOLUME (μM^3)	Δ CELL VOLUME (μM^3)
431	0K	Private droplet	1.03	0.02	1.82E+09	3.57E+08**	0.249	-0.013
433	0K	Private droplet	1.01	-0.001	1.69E+09	2.23E+08*	0.246	-0.016
437	20K	Private droplet	1.02	-0.06	1.74E+09	7.72E+08**	0.323	-0.276**
438	20K	Private droplet	1.11	0.03	1.60E+09	6.28E+08**	0.371	-0.228**
441	60K	Private droplet	1.01	-0.18**	1.70E+09	7.48E+08**	0.325	-0.304**
442	60K	Private droplet	1.21	0.02	1.24E+09	2.88E+08**	0.529	-0.101**
443	0K	Shared droplet	1.14	0.13*	1.51E+09	5.00E+07	0.311	0.049**
445	20K	Shared droplet	1.25	0.17	9.68E+08	-3.33E+06	0.634	0.036
448	60K	Shared droplet	1.24	0.05	1.06E+09	1.15E+08*	0.702	0.072

Table S2: Results of unpaired, two-tailed t-tests comparing each evolved strain with its corresponding LTEE ancestor. Significance levels are denoted by single asterisk (* $p < 0.05$) and double asterisk (** $p < 0.01$).

STRAIN (BK #)	LTEE GEN.	GENOME POSITION	REF.	ALT.	ANNOTATION	GENE	DESCRIPTION
431	0K	558049	G	A	G234E (GGA→GAA)	*ECB_RS02700* →	hypothetical protein
431	0K	1352719	G	T	Q57K (CAA→AAA)	*ECB_RS06750* ←	peptide ABC transporter permease SapC
433	0K	1316697	T	A	H229L (CAC→CTC)	ECB_RS06560 ←	bifunctional indole-3-glycerol-phosphate synthase TrpC/phosphoribosylanthranilate isomerase TrpF
433	0K	4417026	C	A	intergenic (-44/+64)	*ECB_RS21720* ← / ← *ECB_RS21725*	iron-sulfur cluster repair protein YtfE/DMT family transporter
433	0K	4616552	G	T	V342L (GTG→TTG)	*ECB_RS22740* →	multifunctional transcriptional regulator/nicotinamide-nucleotide adenylyltransferase/ribosylnicotinamide kinase NadR
443	0K	3761276	G	T	G174C (GGT→TGT)	*ECB_RS18560* →	bifunctional GTP diphosphokinase/guanosine-3',5'-bis pyrophosphate 3'-pyrophosphohydrolase
437	20K	2462747	A	C	*86Y (TAA→TAC)	*ECB_RS12210* →	phosphocarrier protein Hpr
437	20K	3414187	G	A	E55K (GAA→AAA)	*ECB_RS16980* →	cAMP-activated global transcriptional regulator CRP
438	20K	1173797	G	A	G454S (GGT→AGT)	*ECB_RS05820* →	PTS glucose transporter subunit IIBC
441	60K	2462629	T	C	L47P (CTG→CCG)	*ECB_RS12210* →	phosphocarrier protein Hpr
442	60K	2009534	G	A	R91C (CGC→TGC)	*ECB_RS10145* ←	DUF496 family protein
447	60K	970689	A	ATGTA	pseudogene (829/3153 nt)	*ECB_RS04795* ←	formate acetyltransferase
448	60K	610497	G	A	A32A (GCG→GCA)	*ECB_RS02935* →	enterobactin biosynthesis bifunctional isochorismatase/aryl carrier protein EntB

Table S3: List of mutations that distinguish evolved strains from their corresponding ancestors.

STRAIN (BK #)	LTEE GEN.	EVOLUTION TREATMENT	ACETATE	LACTATE	FORMATE
427	0K	Emulsion ancestor	0.458	0.001	0.157
429	20K	Emulsion ancestor	0.454	0.083	0
430	60K	Emulsion ancestor	0.409	0.116	0
431	0K	Private droplet	0.475	0.002	0.312
433	0K	Private droplet	0.530	0.002	0.299
437	20K	Private droplet	0.017	0.161	0.007
438	20K	Private droplet	0.126	0.164	0
441	60K	Private droplet	0.017	0.084	0.004
442	60K	Private droplet	0.193	0.100	0.000
443	0K	Shared droplet	0.475	0.001	0.233
445	20K	Shared droplet	0.657	0.002	0.254
448	60K	Shared droplet	0.664	0.004	0.269

Table S4: Standardized rates of production of three excreted metabolites per unit of glucose consumed.

Chapter 2

Combining multiple killing systems within a single cell

Katrina van Raay, Joshua Nahum, and Benjamin Kerr

Portions of this chapter are modified from a published version in Environmental Microbiology

Introduction

The natural world is filled with instances of competition between organisms. One extreme form of competition, *allelopathy*, involves the production of chemicals that harm or kill competitors. Such chemical warfare is ubiquitous in the microbial world. Nearly every bacterial lineage studied to date contains strains that produce proteinaceous anti-microbials called bacteriocins (Riley & Wertz 2002a). The best-understood types are the *colicins*, produced by and active against *Escherichia coli* and its relatives (Cascales 2007). For one group of these colicins (group A), allelopathy is encoded in a three-gene operon housed on a plasmid (Fig. 1). One gene encodes the toxin (colicin), a second gene encodes an immunity protein, which binds and neutralizes the toxin, and a third gene encodes a lysis protein. Under stressful conditions, a fraction of plasmid-bearing (colicinogenic) cells turn on the operon, produce toxin, and lyse, releasing the toxin to the external environment. Cells that lack the plasmid die upon exposure to the toxin. In cells that contain the plasmid, the immunity protein, which is constitutively expressed, prevents cell death (Riley & Wertz 2002b, Cascales *et al.* 2007). Thus, active producers kill sensitive

competitors, allowing their latent immune clones to utilize liberated resources (Riley & Gordon 1999, Kerr 2007).

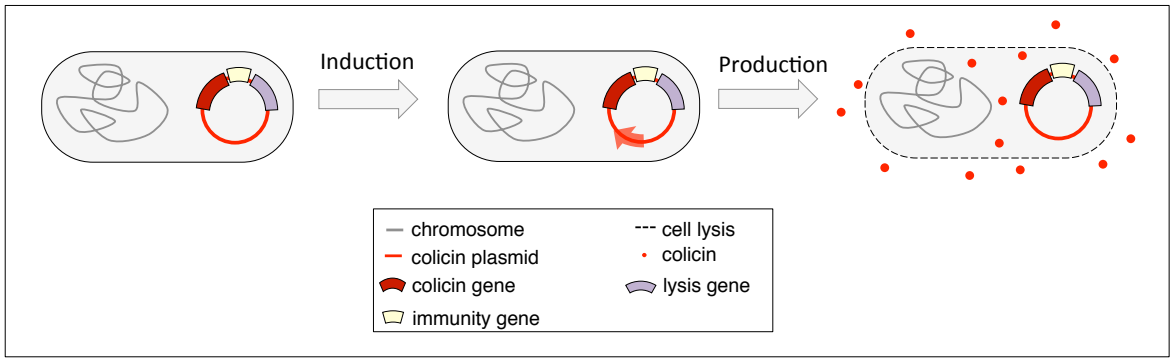

Figure 1. Schematic of a bacterial cell with a group A colicin system. The group A colicin system encodes three genes on the colicin plasmid (red circle): the toxin producing gene (colicin) shown in red on the colicin plasmid, the immunity gene (shown in yellow), and the lysis gene (shown in purple), which lyses the bacterial cell and releases colicin into the environment. Under stress, a portion of colicinogenic in a population undergo induction and begin colicin synthesis.

However, there exists another group of colicin systems (group B) that has only the toxin and immunity genes and appears to lack the lysis gene (van der Wal *et al.* 1995, Cascales *et al.* 2007). It is a mystery how these colicins get out of their host cells. In an exciting study, Nedialkova *et al.* (2016) show that *prophages* may offer a route of escape for the group B colicin ColIb. Prophages are bacterial viruses that have been incorporated into the bacterial genome (bacteria carrying prophages are called lysogens) (Campbell 2003). Prophages are common in bacteria, comprising up to 20% of bacterial genomes (Bondy-Denomy & Davidson 2014). Similar to colicin plasmids, prophages can be vertically transmitted within bacterial lineages without induction. However, under certain circumstances, the prophage can be induced, resulting in the production of progeny phage and lysis of the host cell. Interestingly, some of the same conditions (those activating the SOS response) that induce prophages also induce colicins. In this way, prophage systems may offer an escape route for group B colicins.

Nedialkova and colleagues demonstrate the plausibility of this scenario through both additive and subtractive engineering. Specifically, the authors transformed the ColIb plasmid into a prophage-free *E. coli* strain. This transformant produced colicins, but these toxins were not released from the producing cell. The authors then transduced the ColIb *E. coli* with a temperate phage, 933W, and observed the release of the colicins. The focal bacterium of this study, *Salmonella enterica* serovar Typhimurium, contains four intact prophages and the ColIb plasmid. With the prophages intact, colicin is released. However, when one of the phages, ST64B, is deleted, the release of colicin is dramatically reduced.

For both *E. coli* and *S. enterica*, the lysis proteins of the phage appear to be critical for colicin release. Transduction with a lysis-deficient 933W phage is insufficient for colicin release from *E. coli*. Similarly, the loss of the lysis protein of ST64B can reduce colicin release from *S. enterica*. Using reporter assays, the authors additionally show that the timing of toxin production precedes the timing of phage-encoded lysis. Collectively, their experiments provide evidence that phage systems can complement the release deficiency of group B colicins.

Further, the authors demonstrate that phage-mediated release is imperative for successful allelopathic effect in a community context: colicinogenic cells harboring the prophage dramatically outcompete colicin sensitive strains in co-cultures. However, colicinogenic cells without the prophage (or lacking phage lysis genes) have reduced competitive ability. Thus, the combination of two killing systems (phage and colicin) can have ecological consequences with profound effects on community structure.

Mathematical Model

Once the colicin and prophage come together in a cell, why is the union successful? By modifying a mathematical model of Brown *et al.* (2006), we explore the consequences of bringing a prophage and group B colicin together in a bacterial community (see Box 1, Fig. 2).

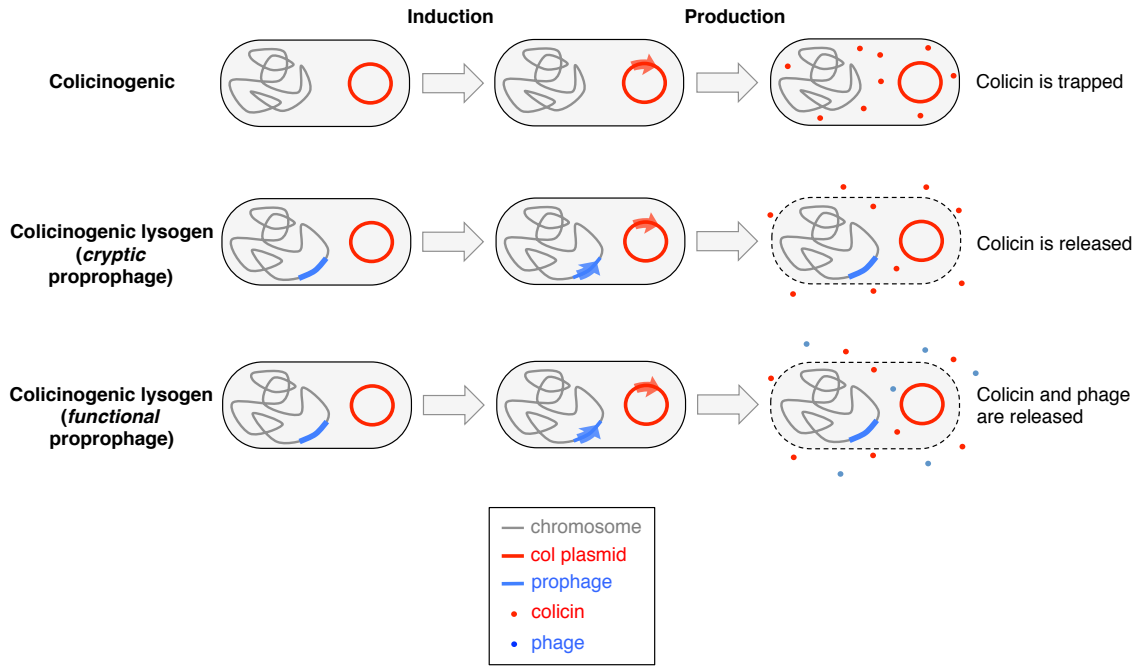

Figure 2: Schematic representation of the combination of phage and bacteriocin systems. Top. *Colicinogenic bacterium* holds a group B colicin plasmid but lacks a lysis gene. Colicin production is induced by the SOS response, but any colicins produced are trapped within the cell. **Middle.** *Colicinogenic lysogen bacterium with cryptic prophage* contains both a group B colicin plasmid and a prophage that cannot produce virions. Even though this cell is not able to produce infectious virions upon induction, the lysis gene of the prophage is functional. The timing of lysis expression is such that the cell lyses after colicins have been produced, releasing colicins into the environment. **Bottom.** *Colicinogenic lysogen bacterium with functional prophage* containing both a group B colicin plasmid and a virion-producing prophage. Under induction, colicins and phage are produced, and released into the environment after cell lysis.

Box 1: *Mathematical model to explore community dynamics of colicin and phage systems*

Nedialkova *et al.* address community-level consequences of the association of phage and bacteriocin systems through pairwise competition experiments. In this Box, we further explore potential dynamics for this community by adjusting and extending a mathematical model developed by Brown *et al*. 2006. Using a system of differential equations, we track the density of sensitive bacteria (S), lysogens (L), colicinogenic bacteria (C), colicinogenic lysogens (X), free phage (P), and free bacteriocin (B):

$$\dot{S} = r_S\left(1 - \frac{N}{K}\right)S - a_P PS - a_B BS$$

$$\dot{L} = \left\{r_L\left(1 - \frac{N}{K}\right) - p\right\}L + la_P PS - a_B BL$$

$$\dot{C} = \left\{r_C\left(1 - \frac{N}{K}\right) - \alpha p\right\}C - a_P PC$$

$$\dot{X} = \left\{r_X\left(1 - \frac{N}{K}\right) - p\right\}X + la_P PC$$

$$\dot{P} = (y_P - 1)(1 - l)a_P P(S + C) + y_P p(L + X) - u_P P$$

$$\dot{B} = y_B p(\alpha C + X) - u_B B$$

The growth rate of bacterial strain i is given by r_i. The total density of bacteria is N. The carrying capacity of the system is K. The rate of attack of phage and bacteriocin to vulnerable cells is a_P and a_B, respectively. The probability that a vulnerable cell infected with a phage becomes a lysogen is l. The rate that a lysogen produces and releases phage is p. The factor α measures the reduction in bacteriocin release for a colicinogenic strain; when there is no lysis gene for such a strain, $\alpha = 0$. The yield of phage and bacteriocin from a cell producing and releasing these particles is y_P and y_B, respectively. The rate of decay of phage and bacteriocin is u_P and u_B,

respectively. We use the parameter values from Brown *et al*. 2006 with the following modifications: $r_L = 1.0$, $r_C = 1.0$, $r_X = 0.9$, $y_B = 1000$, $u_B = 4.4 \times 10^{-2}$, and $\alpha = 0$.

Similar to the results of Nedialkova *et al.*, when the colicinogenic strain does not release bacteriocins, it does not dominate the sensitive strain in a two-strain community. However, if the bacteriocin and phage systems are combined, the colicinogenic lysogen can displace all other bacterial strains (see Fig. B1A). It is important to note that lysogens produce infectious phage in this model. Due to production of phage that either kill or convert the sensitive bacteria, the lysogen can exclude the sensitive strain when these two strains are alone. When the two killing systems come together, the bacteriocin system effectively "piggybacks" on the success of the phage system and the colicinogenic lysogen dominates despite slower growth.

In the system studied by Nedialkova *et al*. the lysogen does not produce infectious ST64B phage. Thus, we reconsider our system without the free phage component. With this change, sensitive cells outcompete colicinogenic cells when the two strains are alone; and sensitive cells similarly displace lysogens in isolation. However, when all four bacterial strains are present, the (formerly ineffective) colicinogenic strain ends up dominating. The colicinogenic strain can be seen as a kind of cheater, failing to release costly bacteriocins, but benefiting from the colicinogenic lysogen that produces and releases the bacteriocin that kill the sensitive and lysogenic strains. An important point here is that in a well-mixed system, the colicinogenic lysogen strain the authors describe would be susceptible to displacement by a release-deficient strain.

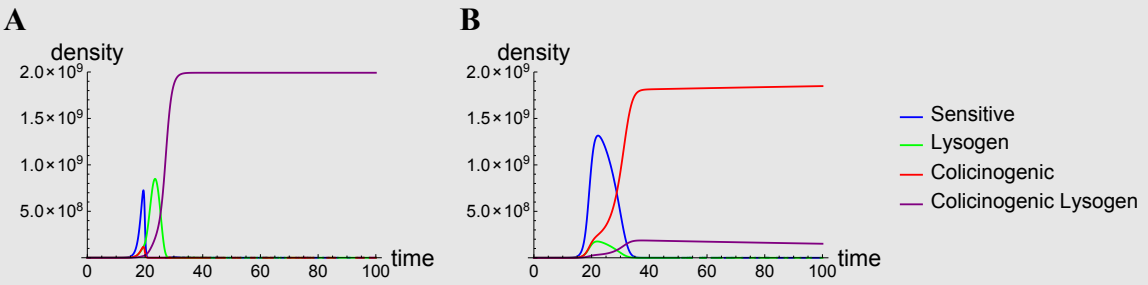

Figure B1: **Community dynamics predicted by a simple mathematical model**. **(A)** When lysogens produce infectious phage, the colicinogenic lysogen is predicted to dominate the community. **(B)** When lysogens do not produce infectious phage, the community is dominated by the colicinogenic strain, which benefits from the bacteriocin released by the colicinogenic lysogen, but avoids the costs (including suicide) associated with the colicinogenic lysogen.

Our model shows that when the lysogen produces infectious virions, the colicin system can piggyback to high frequency along with the prophage system (Fig B1A). In this scenario, vulnerable cells could be hit with both colicins and phage. Thus, if these two killing systems can come together into the same cell, there is potential for this "colicinogenic lysogen" to drive itself to dominance in the community.

Discussion

Prophage can become dysfunctional in bacteria through selection or neutral drift, such that the lysogen no longer encodes for infectious virions; such prophages are referred to as 'cryptic' (Bobay *et al.* 2014, Casjens 2003). In the system described by Nedialkova and colleagues, the prophage ST64B is one such cryptic prophage, where a single frame shift mutation prevents infectious virions from being produced but maintains the lysis gene (Figueroa-Bossi & Bossi 2004). How does a strain containing both a cryptic prophage (with a functional lysis gene) and a colicin system fare in a bacterial community?

It turns out that crypsis is the Achilles' heel of the colicinogenic lysogen, which is now dominated by strains that produce colicins but have no prophage (Fig. B1B). The colicinogenic-only strain (Fig. 1 top panel) benefits by having immunity to the colicin, but does not incur the cost of killing itself. Here, the colicinogenic strain is a kind of "cheater," taking advantage of the bacteriocins released by the colicinogenic lysogen with a cryptic prophage (Fig. 1 middle panel) while avoiding the cost of releasing these "public goods." This cheater does not allow the colicinogenic lysogen to rise to high frequency. This result motivates a follow-up question: Are there circumstances where a colicinogenic lysogen with a cryptic prophage can dominate strains that are only colicinogenic?

One possibility is that the benefits conferred by carrying the prophage outweigh the cost associated with suicide (particularly if suicide is infrequent). Cryptic prophages can provide other benefits to their hosts, including immunity to infection by other phage (coinfection), resistance to antibiotics, and genes for virulence factors (Brüssow *et al.* 2004, Tucker 2004, Wang *et al.* 2010, Fortier & Sekulovic 2013). However, if the prophage themselves are beneficial, a different kind of cheater then becomes possible: a strain that drops the phage lysis gene but retains the advantageous components of the prophage.

Both kinds of cheaters may be kept in check by another feature of microbial life: spatially limited dispersal and interaction. While the model in the Box assumes a well-mixed community (as in a well-shaken flask), natural communities of microbes are often structured spatially (e.g., in biofilms). Given such spatial structure, only a bacterium that *releases* bacteriocin allows its clones to expand into the territory held by sensitive neighbors (Chao & Levin 1981; Kerr 2007). Due to spatially restricted dispersal of cells and limited diffusion of toxins, the potential for a cheater to free ride off the releasing strain is diminished. Thus, spatial structure may account for the rise of

bacteria holding both a colicin plasmid and a cryptic prophage encoding cell lysis, as in the system described by Nedialkova *et al.* In the Appendix, I explore this idea further using a spatially explicit agent-based model.

Summary and Prospects

Given the ubiquity of both prophages and bacteriocins in bacterial populations, it seems likely that these (and other) killing systems come together with reasonable frequency. Such a union could complement one or multiple dysfunctional partners or synergistically magnify the effects of both. Indeed there are cases where group A colicins have come together with group B colicins in a single cell (Gordon & O'Brien 2006). Similarly, multiple prophage systems might join forces within a cell. The focal strain of *S. enterica* in the Nedialkova *et al.* study has, in addition to the cryptic prophage ST64B, three functional prophages. In a population of bacteria, the timing of virion release of two of these functional prophages is offset such that there are two waves of virion release: one subset of lysogens releases virions after the other (Bossi *et al.* 2003). This allows virions from the second prophage to infect and kill competitor cells that have been lysogenized by virions from the first, resulting in a second chance to decrease the population of competing cells. Here, two prophages together enhance the ability of the strain to kill competitors.

Nedialkova and colleagues show that a dysfunctional prophage can complement a dysfunctional colicin in a bacterium and result in a more effective killing system. The strain additionally benefits by housing productive prophages that increase its virulence and allelopathic impact. What other fascinating unions remain to be discovered? What are their effects at the cell, population, and community level? Certainly there are many exciting avenues for exploration, such

as investigating the mechanisms behind different killing systems, their interactions with each other, and their ecological and evolutionary consequences.

Appendix: Agent-based Model of a Spatially Structured Community

The partnership of a group B colicin system that cannot release colicins into the external environment and a prophage that can lyse the cell is an exciting discovery. While this discovery provides a solution to the mystery of how group B colicins can function as allelopathic agents, it also generates a new mystery: how the union of cryptic prophage and group B colicin plasmids persists. This new mystery is rooted in the fact that the colicinogenic lysogen with cryptic prophage is a public goods producer that can be subverted by a strain that only possesses the colicin plasmid and does not lyse.

One scenario that might enable cells with both the cryptic prophage and group B colicin plasmid to thrive in the presence of cells with only the group B colicin plasmid would be if benefits and resources could be localized. This can be realized through the addition of spatial structure, where recipients that benefit from the presence of a public good such as a colicin will likely be related to the cell that produced them. Here, we develop an agent-based model that incorporated spatial structure to investigate the conditions in which a bacterial species carrying a group B colicin plasmid and cryptic prophage might be stably maintained.

We modeled three-player bacterial communities in a 100×100 square lattice world, where each position contained a single cell of one of the bacterial strains (*Sensitive (S)*, *Colicinogenic (C)*, *Colicinogenic-Lysogenic* with cryptic prophage (*CL-)*, or was empty; see Table 1 for Strains). All individuals shared base growth and death rates, and suicidal organisms (*CL-*) experienced an increased death rate. Individuals that produce colicins paid a cost through decreased growth rates

(see Table 2 for parameters and their ranges). When an organism died, its position in the lattice became empty. Empty positions could become occupied when a neighboring cell reproduced into it. If a cell committed suicide and lysed, colicin would be released to kill susceptible cells in the immediate vicinity. Parameter sweeps were performed for varying growth, death, and suicide rates for each of the strains. Spatial structure was manipulated by varying the neighborhood, where a neighborhood is the set of surrounding positions where a cell undergoing division could "disperse" one of its daughters into the focal position. This could be very small (the nearest neighbors), very large (the whole lattice—which would approximate a well-mixed system), or somewhere in between.

STRAIN	**IMMUNE TO COLICIN**	**IMMUNE TO PHAGE**	**RELEASES COLICIN**	**RELEASES PHAGE**	**COMMITS SUICIDE**
SENSITIVE (*S*)	N	N	N	N	N
COLICINOGENIC (*C*)	Y	N	N	N	N
COLICINOGENIC-LYSOGENIC WITH CRYPTIC PROPHAGE (*CL-*)	Y	Y	Y	N	Y

Table 1: Strains and strain properties.

PARAMETER DESCRIPTION	VALUE
Cost for carrying colicin	0.01 – 0.15
Cost for carrying phage	0.01 – 0.15
Probability of induction	0.0037
Colicin persistence	10 updates
Length of world	100 positions
Seed proportions	0 – 1
Nearest neighbors	4 – 24 positions

Table 2: Parameters for agent-based model. Costs for carrying the colicin and carrying the phage were incorporated by reducing a cell's growth rate by the amount specified.

When simulations of three-player communities were seeded in equal proportions (0.25 of each player and 0.25 of the positions empty) and when the cost for carrying the colicin was greater than the cost for carrying phage, CL- could be maintained in a community with Sensitive cells (S) and Colicinogenic cells (C) (Fig. 3), despite carrying a greater cost than either S or C cells.

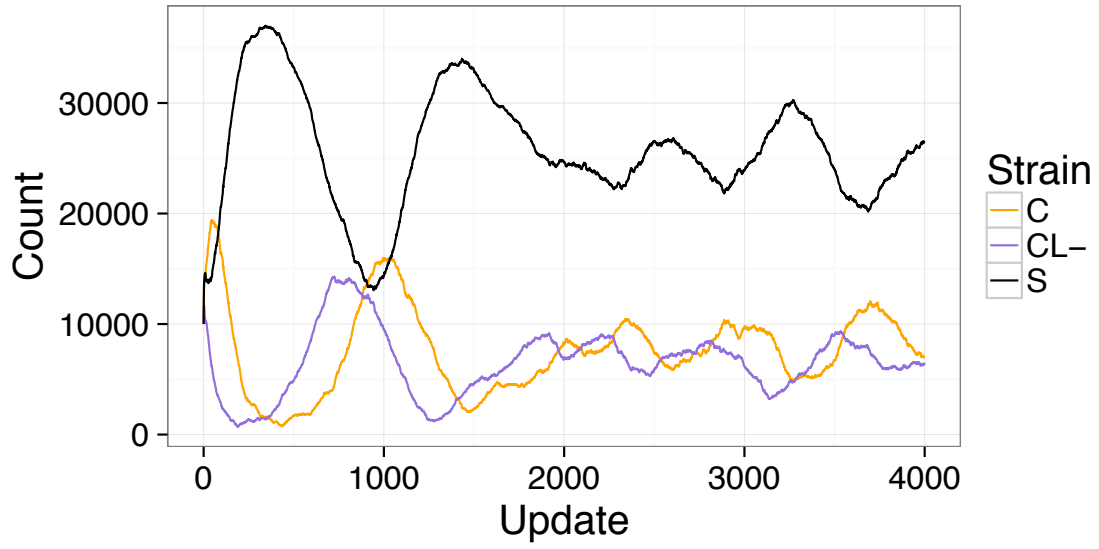

Figure 3: Results of a three-player community made up of Colicinogenic (C), Colicinogenic-Lysogenic with cryptic prophage (CL-), and Sensitive (S) bacteria.

While S cells were able to rise to a higher frequency than either C or CL-, patches of C and CL- were able to persist in the community. In fact, the pattern of rise and fall of all the cells was cyclical. Each peak in S density is followed by a peak in CL- density, which is followed by a peak in C density, and then the cycle renews.

The success of the S type was due, in part, to avoiding costs associated with carrying the colicin or phage. However, since CL- was able to release colicin and kill S, it was able to move into space previously occupied by S (which is why CL- peaks in Figure 3 follow S peaks). Because C was able to grow in the presence of colicin, but avoided the costs of releasing them, C was able to displace CL- when competing for common space (which is why C peaks in Figure 3 follow CL- peaks). Finally, because S avoided costs associated with the colicin plasmid, and because C did not release colicin, S was able to displace C when competing for common space (S peaks in Figure 3 follow C peaks). A snapshot of the community after 2,000 updates shows the patchiness of S, C, and CL- (Fig. 4). Patches of C chased CL- patches, CL- patches chased S patches, and S patches chased C patches. This dynamic resembles the non-transitive game rock-paper-scissors, where rock beats scissor, scissor beats paper, and paper beats rock (Nahum et al. 2011).

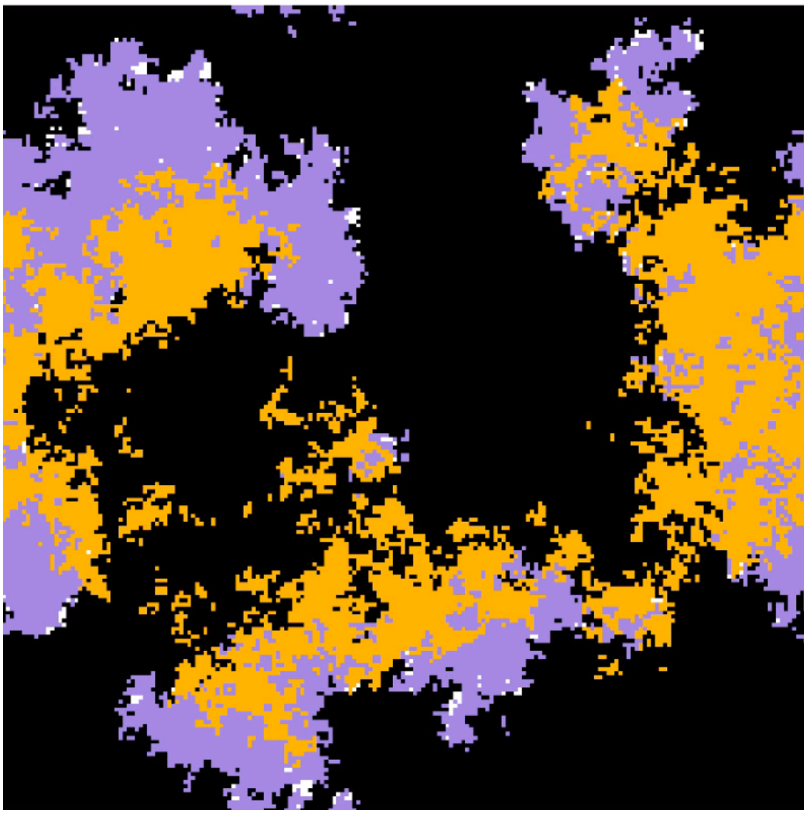

Figure 4: Snapshot of a world that was seeded with equal proportions of S (black), C (orange), and CL- (purple). After about 2000 updates S is in the highest abundance, but patches of C and CL- are present.

When we removed spatial structure from the model and allowed organisms to reproduce into any position, and not just a neighboring position, we were not able to maintain CL- (Fig. 5). Interestingly, under these conditions C was not able to persist either. This result differs from the well-mixed conditions of our mathematical model, suggesting that neither model fully captures all of the elements affecting the population dynamics of these strains. Regardless, these results demonstrate the importance spatial structure can have on ecological and evolutionary dynamics and offer a possible explanation for how CL- persisted and is present in natural bacterial populations today.

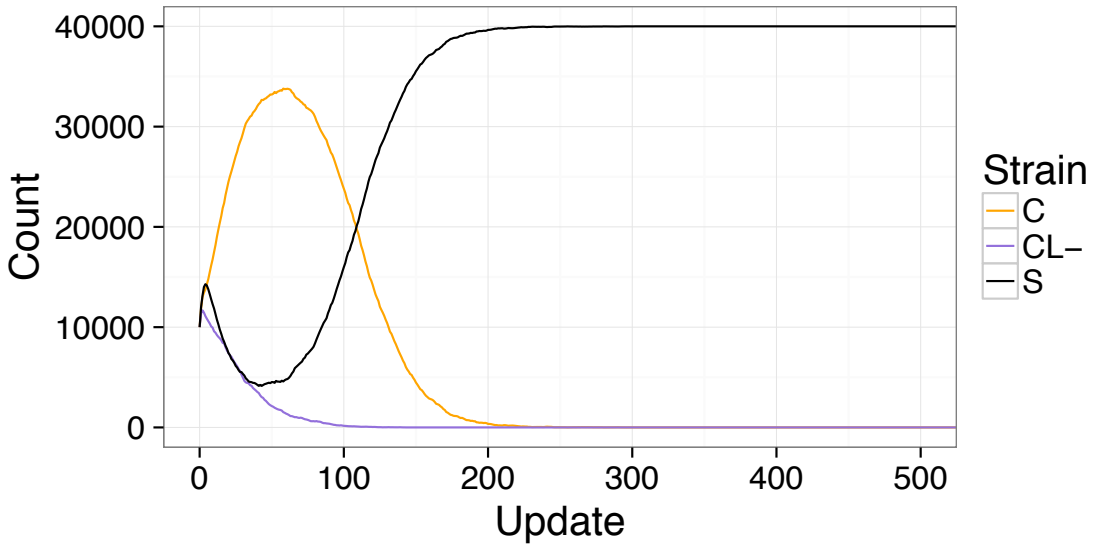

Figure 5: Results of the same three player community as in Fig. 3, but with structure removed. In the well-mixed agent-based model, bacteria were allowed to reproduce into *any* square. CL- is not able to be maintained when just this feature is changed, and C dies out as well.

Appendix 2: Code for Agent-Based Model

To run this model, 3 python scripts were written: ‘cell.py’, ‘world.py’, and ‘data_processing.py’. For reference, the scripts are included below, separated by file. The scripts were originally written to include many types of cells, but in this thesis we only discuss results from communities that included S, C, CL-, and E cells.

Filename = ‘cell.py’

```
BACTERIA_TYPES = {"C+", "L+", "L-", "C+L+", "C+L-", "S", "E", "C-", "C-L+", "C-L-"}


COST_FOR_CARRYING_COLICIN =  X

COST_FOR_CARRYING_PHAGE =  Y


class Bacteria:
    """
    A class that represent the bacterial genotypes in our model
    (C+, L+, L-,  CL+, C+L-, S, E, C-, C-L+, C-L-)
    """
    def __init__(self, bacteria_type):
        assert bacteria_type in BACTERIA_TYPES
        self.bacteria_type = bacteria_type


    def __str__(self):
        return self.bacteria_type
```

```
    def is_immune_to_colicin(self):
        return "C" in self.bacteria_type

#this was already in the code, but I'd like to go over it- I'm not sure what
# can be "L"
    def is_immune_to_phage(self):
        return "L" in self.bacteria_type

    def can_release_colicin(self):
        return self.bacteria_type in {"C+L+", "C+L-", "C-L+", "C-L-", "C+"}

    def can_release_phage(self):
        return "L+" in self.bacteria_type

    def can_lyse(self):
        return self.can_release_colicin() or self.can_release_phage() or self.bacteria_type == "L-"

    def can_become_lysogen(self):
        return self.bacteria_type in {"C+", "C-", "S"}

    def can_become_Lminus(self):
        return self.bacteria_type in {"L+", "C+L+", "C-L+"}
```

```
def convert_to_Lminus(self):
    assert self.can_become_Lminus()
    if self.bacteria_type == "L+":
        self.bacteria_type = "L-"
    if self.bacteria_type == "C+L+":
        self.bacteria_type = "C+L-"
    if self.bacteria_type == "C-L+":
        self.bacteria_type = "C-L-"


def convert_to_lysogen(self):
    assert self.can_become_lysogen()
    if self.bacteria_type == "S":
        self.bacteria_type = "L+"
    if self.bacteria_type == "C+":
        self.bacteria_type = "C+L+"
    if self.bacteria_type == "C-":
        self.bacteria_type = "C-L+"


def replication_rate(self):
    """
    returns the probability of sucessful replication.
    E = 0
```

```
        S = 1
        C+ = 1 - COST_FOR_CARRYING_COLICIN
        C- = 1 - COST_FOR_CARRYING_COLICIN
        ...
        """
        if self.bacteria_type == "E":
            return 0
        if self.bacteria_type == "S":
            return 1
        rate = 1
        if "C+" in self.bacteria_type:
            rate = rate - COST_FOR_CARRYING_COLICIN
        if "C-" in self.bacteria_type:
            rate = rate - COST_FOR_CARRYING_COLICIN
        if "L-" in self.bacteria_type:
            rate = rate - COST_FOR_CARRYING_PHAGE
        if "L+" in self.bacteria_type:
            rate = rate - COST_FOR_CARRYING_PHAGE
        return rate


class Cell:
    """
```

```
A cell represents a position in a grid. The cell can have various things in
it, including colicin, and or phage, and or bacteria, or be empty.
"""
def __init__(self, has_colicin=0, has_phage=0, bacteria_type=Bacteria("E")):
    self.has_colicin = has_colicin
    self.has_phage = has_phage
    self.bacteria_type = bacteria_type

def __str__(self):
    return "Cell(has_colicin={}, has_phage={}, bacteria_type={})".format(
        self.has_colicin, self.has_phage,
        self.bacteria_type)

def __repr__(self):
    return str(self)
```

Filename = ‘world.py’

```
from cell import Bacteria, Cell

import random


from data_processing import PROB_OF_LYSOGEN_DYSFUNCTION,
PROB_OF_INDUCTION, PROB_OF_LYSOGEN_CONVERSION,
COLICIN_PERSISTENCE, PHAGE_PERSISTENCE, COST_FOR_CARRYING_COLICIN,
COST_FOR_CARRYING_PHAGE, PROB_OF_LYSOGEN_DYSFUNCTION




def seed_world(world, seed_proportions):

    total_proportion = sum(seed_proportions.values())

    assert total_proportion <= 1, "Seed proportions must tally to less than or equal 1"




    number_of_positions = len(world.cells)

    new_cells = []

    for bacteria_type_str, proportion in seed_proportions.items():

        num_cells_of_type = int(number_of_positions * proportion)

        for _ in range(num_cells_of_type):

            bacteria_type = Bacteria(bacteria_type_str)

            new_cells.append(Cell(bacteria_type=bacteria_type))


    while len(new_cells) < number_of_positions:
```

```
            new_cells.append(Cell())
        random.shuffle(new_cells)
        world.cells = new_cells


class World:
    """
    A grid of cells.
    """
    def __init__(self, dimension_x_length, dimension_y_length, cells=[]):
        self.dimension_x_length = dimension_x_length
        self.dimension_y_length = dimension_y_length
        assert self.dimension_x_length >= 1
        assert self.dimension_y_length >= 1
        self.cells = cells[:]
        total_cell_count = self.dimension_x_length * self.dimension_y_length
        assert len(self.cells) <= total_cell_count
        while len(self.cells) < total_cell_count:
            self.cells.append(Cell())

    def get_all_coordinates_in_world(self):
        """
        Go through all coordinates by row, starting with (0,0), (1,0), (2,0)
        """
```

```
    results = []
    for y in range(self.dimension_y_length):
        for x in range(self.dimension_x_length):
            coord = (x,y)
            results.append(coord)
    return results



def advance_one_generation(self, is_structured):
    """
    Colicin releasers relase colicin.
    Lysogens with functional phage release phage.
    Cells without immunity to colicin or phage will die if either is present.
    Remove free colicin and free phage.
    All alive bacteria replicate once.
    """
    self.prob_induce_lysis_of_all_cells(PROB_OF_INDUCTION)
    self.prob_lysogen_conversion(PROB_OF_LYSOGEN_CONVERSION)
    self.kill_non_immunes()
    self.degrade_colicin_and_phage()
    self.prob_lysogen_dysfunction(PROB_OF_LYSOGEN_DYSFUNCTION)
    self.replicate(is_structured)
```

```
def prob_lysogen_conversion(self, prob_of_lysogen_conversion):
    all_coords = self.get_all_coordinates_in_world()
    for coord in all_coords:
        x,y = coord
        cell = self.get_cell(x,y)
        bacteria = cell.bacteria_type
        if cell.has_phage > 0 and bacteria.can_become_lysogen():
            should_convert = random.random() < prob_of_lysogen_conversion
            if should_convert:
                bacteria.convert_to_lysogen()


def prob_lysogen_dysfunction(self, prob_of_lysogen_dysfunction):
    all_coords = self.get_all_coordinates_in_world()
    for coord in all_coords:
        x,y = coord
        cell = self.get_cell(x,y)
        bacteria = cell.bacteria_type
        if bacteria.can_become_Lminus():
            should_convert = random.random() < prob_of_lysogen_dysfunction
            if should_convert:
                bacteria.convert_to_Lminus()
```

```
def induce_lysis(self, x, y):
    """
    Induces lysis and releases colicin and/or phage into nearest neighbors
    set by radius for bacterial cells that can lyse ( anything with L )
    """
    cell = self.get_cell(x,y)
    bac = cell.bacteria_type
    release_phage = bac.can_release_phage()
    release_colicin = bac.can_release_colicin()

    nearest_neighbors = self.get_nearest_neighbors_of_cell(x, y, radius=2)
    for neighbor in nearest_neighbors:
        if release_phage:
            cell.has_phage = PHAGE_PERSISTENCE
            neighbor.has_phage = PHAGE_PERSISTENCE
        if release_colicin:
            cell.has_colicin = COLICIN_PERSISTENCE
            neighbor.has_colicin = COLICIN_PERSISTENCE
    if bac.can_lyse():
        cell.bacteria_type = Bacteria("E")
```

```
def prob_induce_lysis_of_all_cells(self, induction_rate):
    """
    for each cell in the world, the cell will be induced some proportion
    of the time randomly. If an induced cell can produce colicin or phage,
    such will be added to its neighbors.
    """
    all_coords = self.get_all_coordinates_in_world()
    for coord in all_coords:
        x,y = coord
        induced = random.random() < induction_rate
        if induced:
            self.induce_lysis(x, y)




def kill_non_immunes(self):
    """
    kills bacterial cells that are sensitive to colicin or phage if colicin
    or phage is present in cell.
    """
    all_coords = self.get_all_coordinates_in_world()
    for coord in all_coords:
        x,y = coord
```

```
        cell = self.get_cell(x,y)
        bac = cell.bacteria_type
        is_immune_to_colicin = bac.is_immune_to_colicin()
        if not is_immune_to_colicin and cell.has_colicin:
            cell.bacteria_type = Bacteria("E")
        is_immune_to_phage = bac.is_immune_to_phage()
        if not is_immune_to_phage and cell.has_phage:
            cell.bacteria_type = Bacteria("E")

def degrade_colicin_and_phage(self):
    """
    removes all colicin and phage from cells
    """
    all_coords = self.get_all_coordinates_in_world()
    for coord in all_coords:
        x,y = coord
        cell = self.get_cell(x,y)
        if cell.has_colicin:
            cell.has_colicin -= 1
        if cell.has_phage:
            cell.has_phage -= 1

def replicate(self, is_structured):
```

```
    """
    All non E bacteria will replicate randomly into a neighbor.
    """
    all_coords = self.get_all_coordinates_in_world()
    random.shuffle(all_coords)
    for coord in all_coords:
        x,y = coord
        cell = self.get_cell(x,y)
        bac = cell.bacteria_type
        replication_prob = bac.replication_rate()
        if random.random() < replication_prob:
            if is_structured:
                four_neighbors = self.get_nearest_four_neighbors_of_cell(x,y)
            else:
                four_neighbors = self.get_random_neighbors_of_cell(x, y, 4)
            unlucky = random.choice(four_neighbors)
            unlucky.bacteria_type = Bacteria(bac.bacteria_type)



def check_bounds(self, x, y):
    """
    checks the bounds of the world
    """
```

```
    assert x >= 0
    assert x < self.dimension_x_length
    assert y >= 0
    assert y < self.dimension_y_length

def __str__(self):
    """
    returns a string
    """
    return "World(dimension_x_length={}, dimension_y_length={}, cells={})".format(
    self.dimension_x_length, self.dimension_y_length, self.cells)

def get_cell(self, x, y):
    """
    this will return a cell associated with the coordinates x,y. indexed by 0.
    """
    self.check_bounds(x,y)
    index = x + y * self.dimension_x_length
    return self.cells[index]

def get_nearest_four_neighbors_of_cell(self, x, y):
    """
    gets nearest four neighbors of a focal cell: in a grid this is the cell
```

```
    immediately above, below, to the right, and left of a focal cell.
    """
    self.check_bounds(x,y)
    result = []
    result.append(self.get_wrapped_cell(x, y-1))
    result.append(self.get_wrapped_cell(x+1, y))
    result.append(self.get_wrapped_cell(x, y+1))
    result.append(self.get_wrapped_cell(x-1, y))
    return result

def get_nearest_neighbors_of_cell(self, x, y, radius):
    """
    Gets all the cells within a radius of the focal cell.
    Excludes the focal cells.

    Radius of 0 is nothing.
    Radius of 1 is the nearest 8 neighbors.
    Radius of 2 is the nearest 24 neighbors.
    """
    self.check_bounds(x,y)
    x_range = list(range(x - radius, x + radius + 1))
    y_range = list(range(y - radius, y + radius + 1))
    result = []
```

```
    for x_coord in x_range:
        for y_coord in y_range:
            if x_coord == x and y_coord == y:
                continue
            result.append(self.get_wrapped_cell(x_coord, y_coord))
    return result




def get_random_neighbors_of_cell(self, x, y, number_of_neighbors):
    """
    gets random cells from world not including focal cell.
    for well-mixed environment.
    """
    self.check_bounds(x,y)
    focal_coordinate = (x,y)
    coordinates = set()
    while len(coordinates) < number_of_neighbors:
        rand_x = random.randrange(self.dimension_x_length)
        rand_y = random.randrange(self.dimension_y_length)
        rand_coord = (rand_x, rand_y)
        if rand_coord == focal_coordinate:
            continue
```

```
        coordinates.add(rand_coord)
    results = []
    for coord in coordinates:
        x, y = coord
        rand_cell = self.get_cell(x, y)
        results.append(rand_cell)
    return results



def get_wrapped_cell(self, x, y):
    """
    If given a coordinate outside the bounds, this will return the cell after
    wrapping around the boundaries.
    """
    wrapped_x = x % self.dimension_x_length
    wrapped_y = y % self.dimension_y_length
    return self.get_cell(wrapped_x, wrapped_y)
```

Filename = 'data_processing.py'

```
import world
import csv
from cell import Bacteria, Cell, BACTERIA_TYPES
from collections import Counter
from multiprocessing import Pool
import os

DATA_DIRECTORY = setwd

def output_counts_to_csv(file_name, counters):
    """
    writes data from simulation to csv file format
    """
    with open(file_name, 'w') as csvfile:
        columns = ['Update', 'Strain', 'Count']

        writer = csv.DictWriter(csvfile, fieldnames=columns)

        writer.writeheader()

        for update, counter in enumerate(counters):
            for bac_type in sorted(list(BACTERIA_TYPES)):
```

```
            if bac_type not in counter:
                counter[bac_type] = 0

            writer.writerow({"Update": update, "Strain":bac_type, "Count":counter[bac_type]})



def print_tally(counter):
    """
    tallies all of the strains in the world and outputs it.
    """
    ordered_bac_types = sorted(list(BACTERIA_TYPES))
    for bac_type in ordered_bac_types:
        line = "{}: {}".format(bac_type, counter[bac_type])
        print(line)


def count_bacteria_types(world):
    """
    takes a world and returns a counter of the bacteria types.
    """

    bac_strs = []
    for cell in world.cells:
```

```
        bac = cell.bacteria_type
        bac_str = bac.bacteria_type
        bac_strs.append(bac_str)
    counter = Counter(bac_strs)
    return counter


def map_world_to_csv(world, file_name):
    """
    creates map of world (location and bacterial cell type)
    """
    x_dimension = world.dimension_x_length
    y_dimension = world.dimension_y_length



    lines = []
    for y in range(y_dimension):
        line = []
        for x in range(x_dimension):
            cell = world.get_cell(x,y)
            bacteria_type_str = cell.bacteria_type.bacteria_type
            line.append(bacteria_type_str)
        lines.append(line)
```

```
    with open(file_name, "w") as file_handle:
        writer = csv.writer(file_handle)
        writer.writerows(lines)


def run_replicate(is_structured, number_of_generations, length_of_world,
    seed_proportions, data_path):
    """
    seed_proportions can take in any bacteria type and value between 0 and 1
    """
    os.makedirs(data_path)
    abundance_path = data_path + "abundance.csv"

    earth = world.World(length_of_world, length_of_world,[])
    world.seed_world(earth, seed_proportions)


    counters = []
    counter = count_bacteria_types(earth)
    counters.append(counter)
    for gen_num in range(number_of_generations):
        earth.advance_one_generation(is_structured)
        counter = count_bacteria_types(earth)
```

```
        counters.append(counter)
        map_filename = "Map_{}.csv".format(gen_num)
        map_filename = data_path + map_filename
        map_world_to_csv(earth, map_filename)

    output_counts_to_csv(abundance_path, counters)
    return earth

def run_job(rep_num):
    rep_dir = "date_0_{}/".format(rep_num)
    data_path = DATA_DIRECTORY + rep_dir
    earth = run_replicate(is_structured=True, number_of_generations=2000,
        length_of_world=100, seed_proportions={ "C-": X,  "S": Y, "C-L-": Z },
        data_path=data_path)

if __name__ == '__main__':
    num_of_reps = 4
    num_of_workers = 2
    jobs = list(range(num_of_reps))
    with Pool(num_of_workers) as p:
        p.map(run_job, jobs)
```

References

Bobay, L.-M., Touchon, M., & Rocha, E. P. C. (2014). Pervasive domestication of defective prophages by bacteria. *Proceedings of the National Academy of Sciences*, *111*(33), 12127–12132. doi:10.1073/pnas.1405336111

Bondy-Denomy, J., & Davidson, A. R. (2014). When a virus is not a parasite: the beneficial effects of prophages on bacterial fitness. *Journal of Microbiology*, *52*(3), 235–242. doi:10.1007/s12275-014-4083-3

Bossi, L., Fuentes, J. A., Mora, G., & Figueroa-Bossi, N. (2003). Prophage contribution to bacterial population dynamics. Journal of Bacteriology, 185(21), 6467–71. doi:10.1128/JB.185.21.6467

Brown, S. P., Le Chat, L., De Paepe, M., & Taddei, F. (2006). Ecology of Microbial Invasions: Amplification Allows Virus Carriers to Invade More Rapidly When Rare. *Current Biology*, *16*(20), 2048–2052. doi:10.1016/j.cub.2006.08.089

Brüssow, H., Canchaya, C., Hardt, W., & Bru, H. (2004). Phages and the Evolution of Bacterial Pathogens : from Genomic Rearrangements to Lysogenic Conversion Phages and the Evolution of Bacterial Pathogens : from Genomic Rearrangements to Lysogenic Conversion. *Microbiology and Molecular Biology Reviews*, *68*(3), 560–602. doi:10.1128/MMBR.68.3.560

Campbell, A. (2003). The future of bacteriophage biology. *Nature Reviews. Genetics*, *4*(6), 471–477. doi:10.1038/nrg1089

Cascales, E., Buchanan, S. K., Duche, D., Kleanthous, C., Lloubes, R., Postle, K., … Cavard, D. (2007). Colicin Biology. *Microbiology and Molecular Biology Reviews*, *71*(1), 158–229. doi:10.1128/MMBR.00036-06

Casjens, S. (2003). Prophages and bacterial genomics: What have we learned so far? *Molecular Microbiology*, *49*(2), 277–300. doi:10.1046/j.1365-2958.2003.03580.x

Chao, L., & Levin, B. R. (1981). Structured habitats and the evolution of anticompetitor toxins in bacteria. *Proceedings of the National Academy of Sciences of the United States of America*, *78*(10), 6324–6328. doi:10.1073/pnas.78.10.6324

Figueroa-Bossi, N., & Bossi, L. (2004). Resuscitation of a defective prophage in Salmonella cocultures. *J Bacteriol*, *186*(12), 4038–4041. doi:10.1128/JB.186.12.4038-4041.2004

Fortier, L.-C., & Sekulovic, O. (2013). Importance of prophages to evolution and virulence of bacterial pathogens. *Virulence*, *4*(5), 354–65. doi:10.4161/viru.24498

Kerr, B. 2007. "The ecological and evolutionary dynamics of model bacteriocin communities." *Bacteriocins: Ecology and Evolution*. Ed. M.A. Riley and M. A. Chavan. Heidelberg: Springer, 111-134

Nahum, J. R., Harding, B. N., & Kerr, B. (2011). Evolution of restraint in a structured rock-paper-scissors community. *Proceedings of the National Academy of Sciences of the United States of America*, *108 Suppl*, 10831–10838. https://doi.org/10.1073/pnas.1100296108

Nedialkova, L. P., Sidstedt, M., Koeppel, M. B., Spriewald, S., Ring, D., Gerlach, R. G., Bossi, L. and Stecher, B. (2015) Temperate phages promote colicin-dependent fitness of *Salmonella enterica* serovar Typhimurium. Environ Microbiol. doi: 10.1111/1462-2920.13077

Riley, M. A., & Gordon, D. M. (1999). The ecological role of bacteriocins in bacterial competition. *Trends in Microbiology*, *7*(3), 129–33. doi:10.1016/S0966-842X(99)01459-6

Riley, M. A., & Wertz, J. E. (2002a). Bacteriocin diversity: Ecological and evolutionary perspectives. *Biochimie*, *84*(5-6), 357–364. doi:10.1016/S0300-9084(02)01421-9

Riley, M. A., & Wertz, J. E. (2002b). Bacteriocins: Evolution, Ecology, and Application. *Annual Review of Microbiology*, *56*(1), 117–137. doi:10.1146/annurev.micro.56.012302.161024

Tucker, C. P., & Heuzenroeder, M. W. (2004). ST64B is a defective bacteriophage in Salmonella enterica serovar Typhimurium DT64 that encodes a functional immunity region capable of mediating phage-type conversion. *International Journal of Medical Microbiology*, *294*(1), 59–63. doi:10.1016/j.ijmm.2003.12.001

van der Wal, F. J., Luirink, J., & Oudega, B. (1995). Bacteriocin release proteins: Mode of action, structure, and biotechnological application. *FEMS Microbiology Reviews*, *17*, 381–399. doi:10.1016/0168-6445(95)00022-4

Wang, X., Kim, Y., Ma, Q., Hong, S. H., Pokusaeva, K., Sturino, J. M., & Wood, T. K. (2010). Cryptic prophages help bacteria cope with adverse environments. *Nature Communications*, *1*(9), 147. doi:10.1038/ncomms1146