

Engineering is evolution: a perspective on design processes to engineer biology

Received: 11 September 2023

Simeon D. Castle¹✉, Michiel Stock² & Thomas E. Gorochoowski^{1,3}✉

Accepted: 18 April 2024

Published online: 29 April 2024

 Check for updates

Careful consideration of how we approach design is crucial to all areas of biotechnology. However, choosing or developing an effective design methodology is not always easy as biology, unlike most areas of engineering, is able to adapt and evolve. Here, we put forward that design and evolution follow a similar cyclic process and therefore all design methods, including traditional design, directed evolution, and even random trial and error, exist within an evolutionary design spectrum. This contrasts with conventional views that often place these methods at odds and provides a valuable framework for unifying engineering approaches for challenging biological design problems.

Synthetic biology aims to apply engineering principles and the design process to create biological systems with new and desired functionalities¹. While it is seen by many as a young field that promises a lot, we have, in fact, been making with biology for millennia, most notably through agriculture and forestry, breeding, and fermentation of foods and beverages. However, it wasn't until the turn of the 21st century that we developed a formal engineering discipline around biology^{2–4}. There are parallels here with mechanical engineering and thermodynamics. Mechanical systems had been built for thousands of years. Still, it was only during the Industrial Revolution that mechanical engineering became a fully-fledged discipline, predating and driving the scientific development of thermodynamics upon which it now rests^{5,6}. Biological engineering today is in a similar position to where mechanical engineering was then, lacking the knowledge and models needed for more predictable design.

Over the last two decades, synthetic biology has made notable progress towards becoming a mature engineering discipline. However, we have also been humbled by how difficult it is to engineer biology. We are still limited to building relatively modest systems, mostly in model organisms, and success is far from guaranteed. This is partly due to technical challenges and our incomplete knowledge of biology. More fundamentally though, it stems from the complex nature of biosystems. Bioengineers deal with *living* systems with long evolutionary histories that grow, display agency, and have potential evolutionary futures. Existing bioengineering paradigms that do not acknowledge this fact will

always hold back our ability to engineer biology, regardless of how advanced our technology and biological knowledge may become. We, therefore, need a new kind of engineering that leans on different philosophical assumptions, where change, uncertainty, emergence, and complexity are built in.

A common trend across the field has been to apply classical engineering principles such as standardisation^{7–11}, decoupling^{12–17} and abstraction^{18,19} in an attempt to tame biological complexity. However, there has been a conspicuous lack of consideration as to *how* these principles should be applied and whether the fact that biology is an evolved and evolving substrate warrants a different approach. To consider more deeply how engineering principles and the design process should be applied to biology, we must first answer some fundamental questions about engineering and design. What precisely do we do when we engineer something? What is the design process, and how does it work? How does applying engineering principles for a given discipline relate to the substrate it engineers? We need to apply an engineering mindset to the design of the engineering discipline itself.

Biological engineering is meta-engineering

By evolving, biosystems produce and refine themselves, and they have a purpose. Just as a hammer is designed for and fulfils the purpose of driving nails, biological objects such as a limb or an enzyme have evolved to perform tasks at which they must succeed (e.g., enabling a crucial chemical conversion). Evolution separates biological systems

¹School of Biological Sciences, University of Bristol, Life Sciences Building, 24 Tyndall Avenue, Bristol, UK. ²KERMIT, Department of Data Analysis and Mathematical Modelling, Ghent University, Ghent, Belgium. ³BrisEngBio, School of Chemistry, University of Bristol, Cantock's Close, Bristol, UK.

✉ e-mail: s.castle@bristol.ac.uk; thomas.gorochoowski@bristol.ac.uk

from non-biological natural systems²⁰. Bioengineers do not design simple artefacts; they design systems that are themselves capable of something like design. It allows, and perhaps even requires, that bioengineers take a step back and operate at a higher level of abstraction. We need to become meta-engineers: engineers that engineer the engineering process itself.

An analogy can be made with photography. Before the camera was invented, creativity and skill in image-making were applied directly to the construction of the image itself through marks left on canvas or paper (e.g., where to apply brush strokes and which colours to use). Once the camera was invented, we no longer needed to create the images ourselves. We could instead construct image-making systems. Human skill and creativity were still very much present, but simply applied at a higher level of abstraction (e.g., considering the choice of subject, lens, shutter speed, and a multitude of other parameters). Rather than creating an image directly, the photographer carefully tunes a system and process to create an image the way they want. After the invention of photography, questions about image-making took on a new form. How do we make the best camera? How do we know which settings to use to achieve a desired result? What are the principles and rules for this new image-making art form?²¹

Engineering design as an evolutionary process

The most commonly taught framework for the design process is the German systematic method, which describes design as occurring in three stages: (i) functional, (ii) conceptual, and (iii) embodiment²². However, anyone who has carried out even a simple design project knows that it does not follow this simple sequential process. The three stages mingle and overlap and are cycled through many times (Fig. 1a). The design-build-test cycle (Fig. 1b), first proposed by management theorist Kim B. Clark²³, is popular amongst synthetic biologists and captures the iterative nature of the engineering method more closely. Still, the details of how each stage works and relates to each other are not always well defined. The idea that design is a process that needs no explanation is simply assumed.

In perhaps the most theoretically rigorous attempt at explaining the design process, Armand Hatchuel and Benoît Weil’s CK theory describes design as existing in two spaces: concept space, and knowledge space (Fig. 1c)²⁴. Design progresses by translating *concepts*, C (design ideas whose viability are unknown), into *knowledge*, K, in an iterative process consisting of four “operators” collectively named the design square. Concepts can generate new concepts through ideation (C-C operator), concepts can be converted into knowledge through

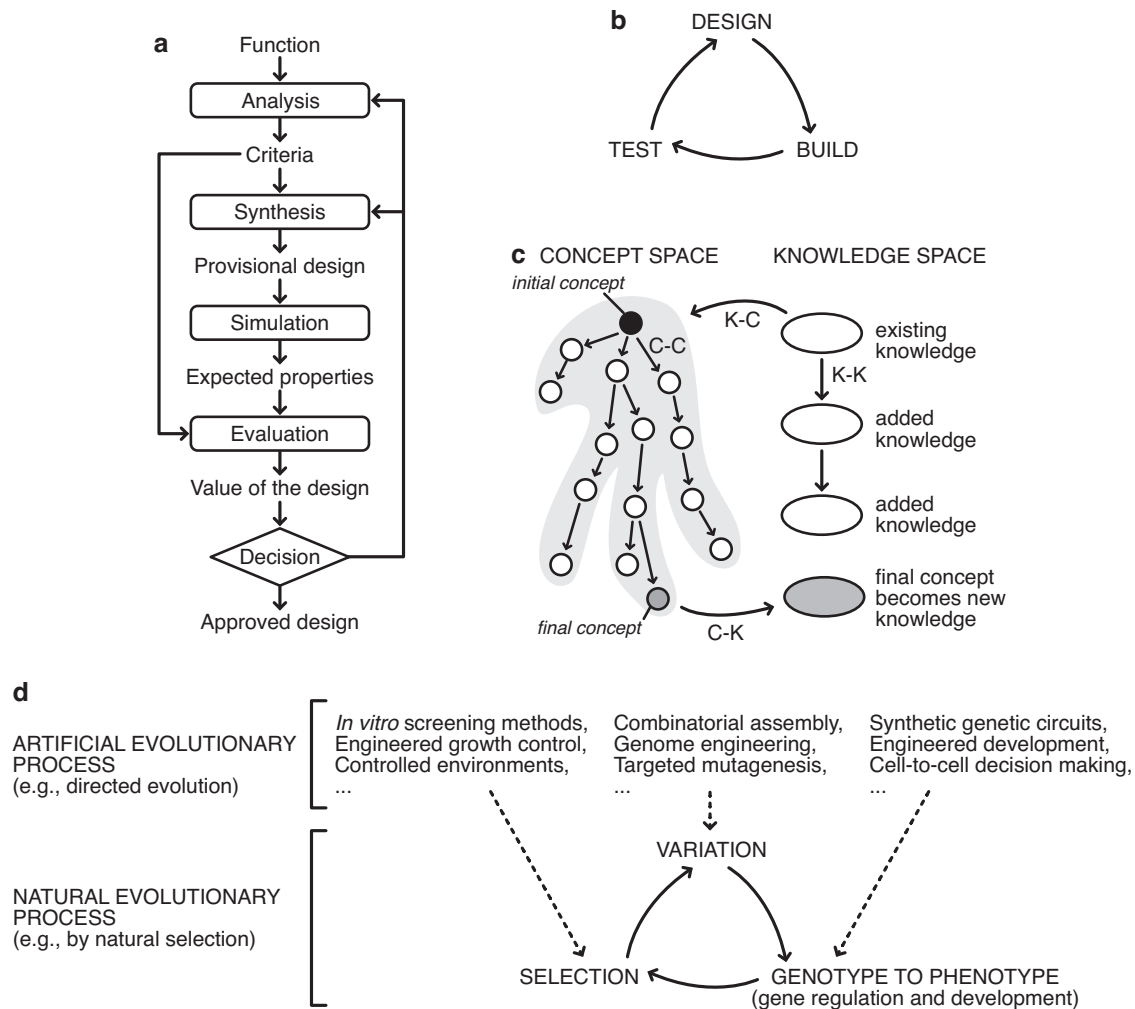


Fig. 1 | Schematics of various models of the design process and links to natural and artificial evolution. **a** The basic design cycle of Roozenberg and Eekels⁹. **b** The design-build-test cycle that is often referred to in synthetic biology. **c** In CK theory by Hatchuel and Weil⁸, design is a form of knowledge generation that takes place via various operations from concept space to knowledge space, in a cyclical process. **d** Connections between natural evolutionary processes (bottom) and artificial

evolutionary processes (top). While natural evolutionary processes harness the innate capabilities of biological populations to evolve through variation, their genotype to phenotype map, and selection, artificial evolutionary processes can modify these core elements using synthetic biology and other engineering tools. This enables the intent of the biological engineer to be embedded within the evolutionary process itself, even though natural evolution has no intent of its own.

prototyping (C-K operator), knowledge can feed new ideas (K-C operator), and knowledge can be expanded, e.g., through research (K-K operator). By separating the spaces based on logical status, CK theory can incorporate creativity via exploring the concept space while simultaneously expanding and incorporating domain knowledge.

Historian Walter Vincenti's book 'What engineers know and how they know it', is to our knowledge the only thorough attempt at placing the engineering process on a firm epistemological footing²⁵. He uses a wealth of case studies from aviation to demonstrate that engineering is its own form of knowledge generation and not simply applied science, concluding that the design process takes place through a cyclic process of idea generation and "vicarious testing"²⁵.

All these descriptions of the design process have a common core. They describe design as a cyclic iterative process where multiple concepts or ideas are either modified or recombined, and then produced physically (i.e., a prototype developed) or virtually through simulation or imagination to form candidate solutions that can be tested. The utility (ability to solve the design problem) of these variants is then assessed, and the best candidates are taken forward for further rounds of iteration. This process is directly analogous to biological evolution, where information about variant solutions is encoded in DNA as genotypes, expressed in the physical world via gene expression and development to produce phenotypes, and tested in the environment (Fig. 1d). Sufficiently functional solutions are then taken forward for future rounds of iteration by natural selection. It is therefore not surprising that biological evolution has successfully been applied to solve difficult engineering problems, both in silico with genetic algorithms commonly used to generate design solutions (e.g., novel satellite antenna shapes²⁶) and in the lab where directed evolution is widely used to develop and improve enzymes²⁷. Biological evolution is

a powerful design tool because biological evolution and design follow the same process.

More broadly, evolution is further evident at the macro scale of the entire technosphere. Technologies have been shown to progress through the modification and recombination of existing technologies that are then selected for by free market economics²⁸. Macro-evolutionary trends are also apparent in technology just as they are in biology: new "species" of technology can arise, technologies become extinct, and all technologies can ultimately be traced back through a lineage that starts with the first prehistoric tools²⁹.

The evolutionary design spectrum

We propose that all design approaches can be considered evolutionary: they combine some form of variation and selection over many iterations. This allows them to be characterised by the number of variants (the population size) that can be tested simultaneously (*throughput*) and how many design cycles/generations are needed to find a feasible solution (*time*). The product of these two numbers is the total number of variants that can be tested and is the *exploratory power* of the design approach. This will always pale in significance compared to the vast design space for any biological system more complex than a short peptide. Despite this tiny sampling, solutions can still be found in the design space for two reasons: (i) *exploration* – design approaches learn from previous iterations as they go, and (ii) *exploitation* – design approaches are constrained and guided by prior knowledge.

These two forms of learning vastly reduce the exploratory power needed for a design approach to find feasible solutions. Exploration is equivalent to the search performed by natural evolution as it roams the fitness landscape. Natural systems also exploit prior knowledge in the form of the unbroken lineage of their past; every generation of which has proven to be both heritable and adaptive. Exploitation is therefore linked to the evolution of evolvability³⁰. Through evolvability, biosystems exploit past evolution in developing body plans, symmetries, functional modules, and other mechanisms that constrain and bias evolution to increase the likelihood of adaptive change. Furthermore, mechanisms, such as learning and epigenetics, which pass information from one individual to the next, are of great importance in evolution^{31,32}. Leveraging either form of knowledge decreases the exploratory power needed by the design process and, thus, the time and throughput required to reach performant solutions. Different design processes leverage exploration and exploitation to different extents and have different practical limits on throughput and time. Though natural evolutionary processes have no *intent* (i.e., long-term goal), they do have a *purpose* to increase fitness. This is well illustrated by the Luria-Delbrück experiment, which suggested that adaptation is not due to deliberate genetic mutations in response to selective pressures but instead arises by selecting for random mutations already present in the evolving population³³. Evolution by natural selection, therefore, lacks the intent we would typically ascribe to the concept of design. However, in artificial evolutionary systems (e.g. directed evolution), the bioengineer can steer the underlying processes toward an intended goal because they control how variation and selection occur, as well as the genotype to phenotype map (Fig. 1d). They can indirectly install their intent through design of the evolutionary process itself.

Any design methodology can therefore be considered a point on a two-dimensional evolutionary design spectrum (Fig. 2) where one dimension is the throughput of the methodology (i.e., how many designs can be created and tested at once) and the other is generation count (i.e., the number of iterations, cycles, generations, that the design process is made up of). Methodologies with high throughputs and high numbers of design cycles have high exploratory power. Conversely, methodologies with lower throughputs and design cycles have lower power and require greater exploitation of prior knowledge and greater evolvability to produce successful designs.

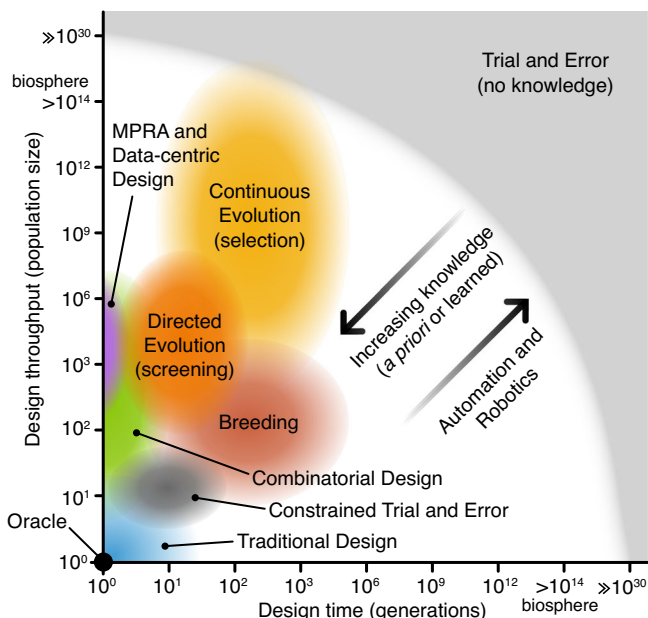


Fig. 2 | The evolutionary design spectrum. Design approaches in the bottom left have low numbers of variants (population sizes) and design cycles (generations). These require significant prior knowledge for successful design with an oracle able to create a perfect design in a single attempt. Design approaches at the top right make little use of prior knowledge and learning, require less evolvability, and their search is less constrained. Trial and error (with no prior knowledge), falls at this extreme, where hyper-astronomical scales are required to design non-trivial systems. Between these extremes, you have all other design methods. In all these cases, prior knowledge can be used to constrain the design space and learned knowledge can dynamically guide the process on the fly. Furthermore, automation and robotics can help to enhance evolutionary design processes, increasing their design throughput and the number of generations that can be feasibly run.

A design methodology must have sufficient specific evolvability to generate phenotypic functions that meet design requirements within the practical constraints of resources and time. This means that for design success, efforts must be made to either maximise evolvability or exploratory power (or both). To improve specific evolvability, exploitation of prior knowledge, ad-hoc exploration, and engineering principles like modularity and hierarchy can be used to constrain the evolutionary landscape to improve the efficiency of search. Take, for example, the simple design problem of creating a synthetic promoter with a given specification (e.g., a desired transcription initiation rate). This could be designed using a variety of methods:

- **Trial and error** (hyper-astronomical population size and number of design cycles): Our starting material is a pool of random sequences, there is no prior knowledge about promoters, and no knowledge is carried from one generation to the next. Finding the optimal design requires random trial and error through the entire search space. Even for this small genetic part, the design space is in the order of 10^{60} . We would therefore need hyper-astronomical population sizes and timescales for all but the simplest systems. However, if we instead merely need to find a minimally satisfactory solution, there are some cases where a design space is sufficiently populated for this strategy to work. For example, 10% of randomly selected DNA sequences of length 100 nucleotides have been shown to display some promoter activity³⁴.
- **Constrained trial and error** (moderate population size and number of design cycles): When employing a trial-and-error type approach, it is common for significant amounts of prior knowledge about the system to be used to cut down the search space to a size that can be reasonably assessed through a feasible number of design cycles. In promoter design, this would relate to a core template sequence that is known to have some basal level of activity, with design variants exploring mutations in a small region known to affect function (e.g., RNA polymerase binding affinity). While this makes trial and error a feasible approach, the diversity of the designs is highly limited and knowledge from one design cycle does not help inform the next.
- **Continuous evolution with selection** (large population size, large number of design cycles): If a way of coupling utility to survival can be found, e.g., by placing the promoter design candidates upstream of an essential gene in the chosen organism, then continuous evolution is possible^{35–37}. This means population size is limited by what can be cultured, not what can be screened, and generation counts are limited by how long the experiment can be run. Thus, hundreds or even thousands of generations are feasible for fast-growing organisms like *Escherichia coli*³⁸.
- **Directed evolution via screening** (moderate population size, moderate number of design cycles): Alternatively, we could use prior knowledge to select an imperfect promoter as a starting point and learn from each generation by choosing improved candidates. For example, the promoter can be coupled to a fluorescent reporter gene and screened via cytometry³⁹. This way, we could find satisfactory designs with moderate population sizes (e.g., hundreds or thousands).
- **Massively parallel reporter assays** (moderate/large population size, single design cycle): Special cases of evolutionary design methods are those with a generation count of one and often a highly constrained design space. Though no continuous learning is possible, design solutions can be found if prior knowledge of the systems can be combined with sufficient throughput. So-called massively parallel reporter assays (MPRAs) often involve fluorescence-activated cell sorting (FACS) and DNA sequencing to reconstruct the evolutionary landscape. In this example, a library of potential promoter designs could be used to drive the expression of a fluorescence reporter gene⁴⁰. Other forms of MPRA use barcoded RNA sequencing to measure transcription rates of variants in a library directly⁴¹. In principle, these approaches can be expanded into multiple generations by repeating the process, though this may be costly and labour-intensive.
- **Combinatorial design** (small/moderate population size, single design cycle): When designing larger or more complex systems, working at single nucleotide resolution may be inefficient. Prior knowledge and abstraction can be used to work with genetic units larger than single nucleotides. For example, if various domains of a promoter are known (e.g., transcription factor binding domains, sigma factor binding domains, TATA box), then these domains can be abstracted into parts that can serve as “genetic units” to be assembled in a combinatorial fashion and the resultant library screened for the best candidates. This strategy depends on abstraction to create complex forms of genetic recombination that allow for search to take place in a highly constrained design space.
- **Traditional design** (small population size, small number of design cycles): The traditional design process leverages prior knowledge of the design problem built up through many generations – both formal codified knowledge (e.g., models), tacit knowledge, and intuition to constrain the design space. Design therefore takes place needing only small numbers of variant concepts and iterations. For example, a traditional or rational design approach may involve using existing knowledge of natural functional elements within promoters and mathematical modelling to construct a synthetic promoter that may be iterated through multiple rounds of testing and improvement⁴².
- **Data-centric design** (small to large population size, small number of design cycles): The data-centric design approach can be seen as a variant of the traditional design that often employs high-throughput assays (e.g., MPRAs). Typically, a predictive model is initially trained on existing data sets to estimate how well a new variant is expected to work. Models range from a basic linear regression to more complex artificial neural networks that, for the promoter example, would take a DNA sequence as input and return the expected promoter strength. Datasets used for training could contain naturally occurring variants or synthetically designed and synthesised variants that have been characterised in the lab. If searching for a specific promoter strength, the model can also be used to suggest variants that are known to improve the model (e.g., exploring novel areas of design space) in a process called active learning⁴³. This type of design process is often coupled with Design of Experiment (DoE) or Bayesian Optimization methods to optimally select new variants to build and test for subsequent design cycles to maximise the information learnt about the relationship between factors of a design/experiment^{44–46}. Furthermore, the application of liquid handling robotics allows for far greater numbers of design cycles at the cost of smaller population sizes⁴⁷.
- **Oracle** (population of one, single design cycle): If it were possible to have complete knowledge of a system before beginning the design process, we could select the ideal design without any testing. Such an oracle could perform the design process in a single generation with a population size of one, immediately producing an optimal promoter base-for-base.

Choosing a design approach

As outlined above, different design methods have different practical limits for population size and generation count. This means they leverage exploration and exploitation to differing extents and thus have varying degrees of exploratory power. In addition, there are practical limits to population size and generation count depending on the type of system being designed and the nature of the design problem. For example, a methodology for designing a complex biosystem

such as a multi-component genetic circuit may have limited exploratory power due to practical constraints on throughput and time. Conversely, a simple system such as a small protein whose function can be selected for with a continuous growth assay may be carried out in a large population for many generations quite readily. The design process chosen must, therefore, match the constraints of the design problem. As exploratory power reduces, specific evolvability must increase. This can be done through leveraging prior knowledge or applying known engineering principles. However, there may be other ways of improving specific evolvability for specific designs problems, such as by engineering genetic variation⁴⁸.

If evolutionary design approaches are to be used to their full potential for engineering biology, methods to generate and characterise libraries of greater genetic diversity will need to be developed. These libraries should have structural and functional variety and go beyond nucleotide variation or the simple interchanging of parts. What form of synthetic variation is most important and how artificial genetic units should be engineered and selected to maximise the evolvability of resultant populations depends on the size and complexity of the genetic system to be evolved and on technical limitations in terms of generating and reproducing populations, and measuring, screening or selecting the functions of individual genotypes. Different types of genetic systems across scales require different evolutionary design approaches, falling into different regions of the evolutionary design spectrum.

Generally, a more complex biosystem will have greater technical limits on population size and generation times. Therefore, as the complexity of a biosystem increases, the appropriate evolutionary design approach shifts to a point between traditional design and trial and error in the evolutionary design spectrum. Complex systems are difficult for the human mind to design via traditional approaches (bottom left, Fig. 2), but useful results are also unlikely to occur if trial and error is used due to the vastness of the design landscape. The application of deep learning approaches that can exploit high-dimensional gradients inferred from data to guide a design process (i.e., data-centric design) may be able to help in these situations.

In the following sections, we look at how the design methodologies described above can be applied in different ways to engineer biological systems across scales.

Designing genetic regulatory elements and proteins

Regulatory elements like promoters, ribosome binding sites (RBSs) and terminators have relatively small genotypes making them amenable to direct oligo synthesis or nucleotide mutagenesis of a starting template via error-prone PCR. Activity levels can be coupled to fluorescence to enable selection via FACS⁴⁹, or expression of an antibiotic resistance gene⁵⁰ or DNA polymerase⁵¹ to affect the reproductive success of the underlying design. This allows for enormous population sizes, where even a single generation may yield good results. Nanopore-based direct-RNA sequencing has also been used to monitor the performance of thousands of transcriptional regulators effectively from large mixed pools of diverse combinatorically assembled designs in a single experiment⁵². Beyond screening, there has been some success in exploiting our knowledge of the modular structure of these biological parts and key motifs within them (e.g., the crucial role of -10 and -35 sites in *Escherichia coli* promoters). However, in virtually all examples to date, current models cannot accurately predict the phenotype from genotype, resulting in the need for nucleotide-level mutations of moderately performant designs to achieve a desired functionality^{53,54}.

Proteins have been extensively designed using directed evolution. Sequence lengths range from peptides of a few amino acids, which can potentially be explored exhaustively⁵⁵, but are typically on the order of -1-3 kb long when encoded in DNA. At this scale, single nucleotide

mutations are an effective design strategy, especially if focused on specific target domains or regions. One of the most widely used in vitro methodologies for this task is phage-assisted continuous evolution (PACE)³⁷, which enables rapid cycles of evolution by having the gene of interest encoded within a bacteriophage and linking its activity to the fitness of phage replication. Continuous in vivo evolution platforms like OrthoRep⁵⁶ or eMutaT7⁵⁷ are also ideally suited to this task. However, for complex, multi-domain proteins, the addition of structured recombination considering domain boundaries can significantly improve evolvability, as beneficial mutations across different modules can be effectively combined^{58,59}. Screening can be performed using in vitro methods with large populations⁶⁰ and a handful of design cycles are typically sufficient to find improved designs.

There is also growing interest in leveraging the power of automation to increase throughput and, thus, exploratory power⁶¹. Integration of advanced liquid handling robotics and high-throughput assays with existing continuous directed evolution platforms like PACE offers the ability to greatly expand the number of evolutionary trajectories that can be simultaneously explored⁶²⁻⁶⁴. These provide valuable information regarding the reproducibility of these design processes and help to improve overall reliability. The use of feedback control within these systems can also help to accelerate evolution by reshaping fitness landscapes dynamically over time to support effective adaptation of the population⁶².

Outside the lab, emerging machine learning methods for protein structure prediction allow for more traditional and data-centric design approaches to be applied. These can supplement, or in some simple cases, surpass the need for further directed evolution⁶⁵⁻⁶⁷. However, designing complex molecular dynamics and unstructured regions remains challenging for computational models. In these cases, computationally efficient coarse-grained models able to capture qualitative features of protein dynamics can be used to provide semi-functional starting points⁶⁸, and high-throughput automated directed evolution applied for further refinement of the designs.

Designing multi-gene circuits and pathways

Genetic circuits and metabolic pathways are often composed of numerous proteins and regulatory elements that together enable more complex functionalities. Apart from the DNA encoding them being much longer (typically tens of kilobases long), their multi-component structure means that hierarchy begins to play a role. Specifically, variation can occur at the nucleotide level, or modifications can be made at higher levels of organisation (e.g., the swapping of a transcription factor or entire promoter) enabling larger jumps in the genotype space.

To engineer genetic circuits, the most common approach to date has been through model-guided design to find suitable regulatory topologies that implement a desired logic, and then the application of trial and error to vary the components used and introduction of mutations in key regulatory elements to finally tune gene expression levels such that an overall function is optimised. Introducing insulating elements like self-cleaving ribozymes into sub-circuits (e.g., simple logic gates) has enabled traditional design approaches to push forward the complexity of the circuits we can build. However, limits are soon reached, making the design of functional genetic circuits composed of more than ten regulators a challenge. A further difficulty in employing approaches like directed or continuous evolution is the multi-state nature of these systems, where multiple inputs and outputs exist. Unlike simple functional screens that can select designs based on a single output measurement (e.g., fluorescence or enzymatic activity), genetic circuits often require a specific input-output response. These are difficult to screen because of the combinatorial explosion in the number of possible states that must be tested as the number of inputs and outputs grows. Some attempts have been made to overcome this

for simple cases (e.g., a single input and output) by removing the need to carry out cell sorting and using changes in growth rate followed by deep sequencing⁵⁰, but the population sizes will always be fundamentally limited, highlighting the need for traditional model-guided or data-centric design approaches in this area.

In contrast, the engineering of metabolic pathways often employs a different approach because maximising a single output metabolite is the goal. In this context, the topology of the pathway is typically fixed (i.e., a specific set of chemical conversions is required), making it well suited to a combinatorial design approach. Sets of enzymes for each step in the pathway can be combined in different ways and genetic regulatory parts with varying activity levels are used to create populations of designs where the set of required enzymes are expressed at differing levels. Because screening can be costly to perform in high throughput (although more feasible than for multi-state genetic circuits), the combinatorial approach can be guided by DoE methods to ensure the most information is learnt after each design cycle⁴⁵.

An interesting alternative to combinatorial design for metabolic pathways is the use of *in vitro* SCRaMbLE (Synthetic Chromosome Rearrangement and Modification by *loxP*-mediated Evolution). In this context, the genes encoding each enzyme of the pathway are considered as individual units that can be copied, deleted and their orientation within the pathway flipped by the SCRaMbLE system. This allows for the rapid generation of libraries of metabolic pathway variants with diverse expression levels for each unit⁶⁹. SCRaMbLE could also be applied to gene regulatory networks. However, the enormous design space for most genetic circuits (far exceeding those of metabolic pathways) means that the random SCRaMbLE-based search may not always be effective unless further constrained.

Designing whole cells

Attempting to design entire cells is yet another step up in terms of complexity, with the need to simultaneously coordinate many different cellular processes to maintain viability and fitness. In this setting, a few mechanistic ‘whole cell’ models of well-studied organisms have been built that recapitulate general cellular phenotypes and can be used to support traditional model-based design approaches at this scale^{70–72}. These models have guided host modifications for improved metabolic engineering⁷³ and even been used to generate minimised genomes *in silico*⁷⁴. While these models are sometimes useful, their accuracy is severely lacking due to our limited knowledge of how cells work and missing parameters. It therefore remains to be seen how well predictions from these models generalise outside of the conditions in which they were built. The difficulty in building such models also highlights the need to embrace evolutionary design approaches that rely less on prior knowledge and actively search a design space for desired features.

Beyond computational approaches, long-term experimental evolution studies have shown that novel traits can emerge through natural variation and selection. However, the timescales for discovering these traits can be incredibly slow, requiring thousands of generations⁷⁵. The emerging field of synthetic genomics⁷⁶ offers routes to accelerate this process and overcome our lack of knowledge when designing at a cellular scale. Efforts to synthesise and refactor entire genomes opens opportunities to unravel the core principles underlying genome structure and operation^{77–79}. It also allows for the creation of genomes with new capabilities. For example, the Sc2.0 project is resynthesizing and refactoring the *Saccharomyces cerevisiae* genome to extract and consolidate typically unstable elements (e.g., tRNAs⁸⁰), and has introduced genome-wide modifications to enable SCRaMbLE’ing of genome architecture for accelerated *in vivo* generation of genomic diversity⁸¹. This system has already demonstrated its ability to evolve strains optimised to produce desired chemicals⁸² and has provided insights into the incredible plasticity of the *S. cerevisiae* genome⁸³.

Designing microbial communities and ecosystems

Many biological processes, such as the nitrogen cycle, arise via the collective action of multiple species. Microbial communities are composed of many different types of organisms that compete, cooperate and tolerate each other. Microbiome engineering⁸⁴ is concerned with finding the community that can perform a desired function, such as degrading a pollutant⁸⁵, increasing digestive health/capabilities^{86,87} or making smart materials⁸⁸ and developing methods to stabilise its existence within the broader environment in which it will be deployed (e.g., the human gut).

Artificial selection (a form of directed evolution) can be used to evolve communities directly⁸⁹. The selection takes place at the population level rather than the individual organism. Because it works with a population of populations, the collective is often referred to as a meta-population – an important concept in ecology^{90,91}. Population selection was pioneered in selecting microbial communities in the soil to increase plant biomass⁸⁹ and has since been combined with mathematical models to guide the selection of promising co-cultures^{92,93}. A challenge with the approach is that while microorganisms are, in principle, free-living and can be combined at will, this does not mean there is an incentive for functional associations to form. Another issue is handling the dynamics of the communities. For example, when a contributing species goes extinct or fast-growing organisms take over⁹⁴.

To enable the engineering of stable communities, autotrophic dependencies can be exploited and engineered⁹⁵ or other forms of dependency and interaction used^{96,97}. In this context, traditional model-based design approaches have proven useful in sufficiently capturing the dynamics of the well-defined communities built in the lab. Deploying these communities into real-world environments, though, remains a significant challenge due to a lack of knowledge regarding the interactions present within complex and diverse native communities that might impact the function of engineered communities and their ability to stably associate. To overcome this, physical separation via encapsulation of engineered microbes has been used to lessen competitive effects and safeguard against unwanted release^{98,99}. Looking forward, a mixture of traditional design and evolutionary approaches will likely be required to develop sufficiently robust systems to handle the complexity of real-world microbial communities and ecosystems.

Towards the effective design of engineered biology

Engineering biology is vastly different from other engineering disciplines. This is not only because living matter has the inherent capacity to evolve but also because human-built ‘machines’ (e.g., cars, computers, etc.) are typically organised differently than biological systems¹⁰⁰. Both machines and biological organisms contain hierarchical levels. However, machines are designed so that every level is predictable and can be understood independently from other levels. For example, formulating rocket fuel requires distinct expertise from designing a rocket’s navigation system, and while both are required to build a rocket, they can be considered independent during design. In contrast, living systems have organisational levels that are highly interconnected, allowing small mutations to affect broad phenotypes in complex ways. Furthermore, it is common for higher levels or organisation to influence lower ones via downward causation, and human-built machines typically show a hardware-software duality, whereas biological components have blurred lines of information storage and processing (e.g., RNA being a genetic information carrier and a catalytic entity in its own right)¹⁰¹. Traditional design works with blueprints, which are typically a one-to-one mapping to the artefact built, while living systems have highly non-linear genotype-phenotype maps that are executed via developmental processes and influenced by environmental factors. These properties mean that engineering frameworks based on

reductionism are challenging to apply¹⁰². In contrast, evolution is naturally adept at handling these properties because it acts holistically on the system within its wider environment.

These difficulties mean that adopting a single design methodology for all bioengineering tasks is likely impossible. Instead, combining more traditional design cycles that use model-based design with high levels of abstraction, may allow a targeted population of designs to be generated upon which more classical evolutionary design approaches can be used to tune and refine partially functional or suboptimal designs¹⁰³. Such hybrid methods are already being applied to minimal genome design, helping to accelerate the generation of a minimal cell with reasonable fitness¹⁰⁴. Beyond combining design methods, installing new abilities for cells to design themselves through guided mutations⁵⁷ and rearrangements¹⁰⁵ of their genetic material could further accelerate the design process (e.g., systems like SCRaMBLE). This would work with biology's innate capacity to evolve, while also allowing us to constrain the paths evolution can take such that prior knowledge can bolster the overall exploratory power of the system. Living systems also tend to evolve to become more evolvable, e.g., by making them more robust or modular¹⁰⁶. For example, certain mutations can stabilize a protein and make it more susceptible to beneficial mutations^{107,108}. Hence, it may be possible to engineer such systems so that they are easier to improve.

Another fundamental challenge that is typical when engineering biology is the definition of a successful design. In most engineering fields, the goal is to maximise some desired utility function. However, with biology, there is always a trade-off in the *utility* of the design (i.e., how well it performs the task we require) and the *fitness* of the resultant biological system (i.e., its reproductive rate). We previously developed the 'evotype' as a framework for capturing the evolutionary capacity of a biological design and explicitly highlighting this trade-off⁴⁸. Stated differently, when designing with living matter, we do not only have to consider how well it works, but we also need to consider how it will adapt and evolve during use. Note that this concept integrates the top-down objective (intent) of the bioengineer in the utility with the bottom-up evolutionary aspects captured by fitness.

While we can never fully escape evolution when designing and deploying engineered biological systems, its impact can sometimes be heavily reduced. Recent studies have demonstrated that cooperative assembly of transcription factors can confer regulatory specificity and long-term genetic circuit stability¹⁰⁹, and the development of highly characterised and insulated landing pads in genomes can both increase the predictability of genetic parts and reduce selective pressures to mutate or remove these foreign elements once installed¹⁷. In a similar way, dynamic regulation of the burden a genetic circuit imposes on a host through feedback control have also proven successful in extending the functional lifetime of synthetic genetic circuitry and enabling improved production yields¹¹⁰. This effort to reduce burden mirrors how computer software needs to be optimized to best exploit underlying hardware. For example, mobile apps are often coded to best utilize lower-power processor instructions and minimize continuously running background computations to conserve battery life. However, the burden in these settings manifests differently. While burden in biological systems can lead to mutations that change the functionality of the genetic circuit, the burden of code running on an electronic computer mainly affects performance and resource usage. In living systems, the "software" component influences how the "hardware" evolves.

In conclusion, evolutionary theory is doubly important when engineering biology: not only is the substrate of synthetic biology a result of evolution, engineering itself can be seen as a form of evolution. It turns out that evolution must be understood as the foundation of the engineering method, as well as the creator and driving force of the living substrate itself. Though biological evolution and

technological evolution may seem very different, they are fundamentally the same process. Recognition of this opens the potential for a new way of thinking about how to engineer biology and effectively "design" in the context of living systems.

References

1. Endy, D. Foundations for engineering biology. *Nature* **438**, 449–453 (2005).
2. Gardner, T. S., Cantor, C. R. & Collins, J. J. Construction of a genetic toggle switch in *Escherichia coli*. *Nature* **403**, 339–342 (2000).
3. Elowitz, M. B. & Leibler, S. A synthetic oscillatory network of transcriptional regulators. *Nature* **403**, 335–338 (2000).
4. Becskei, A. & Serrano, L. Engineering stability in gene networks by autoregulation. *Nature* **405**, 590–593 (2000).
5. Carnot, S., Thurston, R. H., Carnot, H. & Kelvin, W. T. B. *Reflections on the Motive Power of Heat and on Machines Fitted to Develop That Power*. (J. Wiley, 1890).
6. Hunt, B. J. *Pursuing Power and Light: Technology and Physics from James Watt to Albert Einstein*. (Johns Hopkins University Press, 2010).
7. Canton, B., Labno, A. & Endy, D. Refinement and standardization of synthetic biological parts and devices. *Nat. Biotechnol.* **26**, 787 (2008).
8. Buecherl, L. et al. Synthetic biology open language (SBOL) version 3.1.0. *J. Integr. Bioinform.* **20**, 20220058 (2023).
9. Baig, H. et al. Synthetic biology open language visual (SBOL visual) version 3.0. *J. Integr. Bioinform.* **18**, 20210013 (2021).
10. Casini, A., Storch, M., Baldwin, G. S. & Ellis, T. Bricks and blueprints: methods and standards for DNA assembly. *Nat. Rev. Mol. Cell Biol.* **16**, 568–576 (2015).
11. Bayer, C. N., Rennig, M., Ehrmann, A. K. & Nørholm, M. H. H. A standardized genome architecture for bacterial synthetic biology (SEGA). *Nat. Commun.* **12**, 5876 (2021).
12. Del Vecchio, D., Ninfa, A. J. & Sontag, E. D. Modular cell biology: retroactivity and insulation. *Mol. Syst. Biol.* **4**, 161 (2008).
13. Jones, R. D. et al. An endoribonuclease-based feedforward controller for decoupling resource-limited genetic modules in mammalian cells. *Nat. Commun.* **11**, 5690 (2020).
14. Shopera, T., He, L., Oyetunde, T., Tang, Y. J. & Moon, T. S. Decoupling Resource-Coupled Gene Expression in Living Cells. *ACS Synth. Biol.* **6**, 1596–1604 (2017).
15. Lou, C., Stanton, B., Chen, Y.-J., Munsy, B. & Voigt, C. A. Ribozyme-based insulator parts buffer synthetic circuits from genetic context. *Nat. Biotechnol.* **30**, 1137 (2012).
16. Qi, L., Haurwitz, R. E., Shao, W., Doudna, J. A. & Arkin, A. P. RNA processing enables predictable programming of gene expression. *Nat. Biotechnol.* **30**, 1002 (2012).
17. Park, Y., Espah Borujeni, A., Gorochowski, T. E., Shin, J. & Voigt, C. A. Precision design of stable genetic circuits carried in highly-insulated *E. coli* genomic landing pads. *Mol. Syst. Biol.* **16**, e9584 (2020).
18. Nielsen, A. A. K. et al. Genetic circuit design automation. *Science* **352**, aac7341 (2016).
19. Aldulijan, I. et al. Functional Synthetic Biology. *Synth. Biol.* **8**, ysad006 (2023).
20. Hartwell, L. H., Hopfield, J. J., Leibler, S. & Murray, A. W. From molecular to modular cell biology. *Nature* **402**, C47–C52 (1999).
21. Berger, J. *Ways of Seeing*. (Penguin UK, 2008).
22. Wallace, K. M. & Blessing, L. T. M. Observations on Some German Contributions to Engineering Design In Memory of Professor Wolfgang Beitz. *Res. Eng. Des.* **12**, 2–7 (2000).
23. Wheelwright, S. C. & Clark, K. B. Competing through development capability in a manufacturing-based organization. *Bus. Horiz.* **35**, 29–43 (1992).

24. Hatchuel, A. & Weil, B. A new approach of innovative design: an introduction to CK theory. *Proc. of ICED 03, 14th Int. Conf. Eng. Des.*, DS31_1794FPC (2003).
25. Vincenti, W. G. *What Engineers Know and How They Know It: Analytical Studies from Aeronautical History*. (Johns Hopkins University Press, 1990). **Critically examines the nature of engineering knowledge, its evolution, and its impact on societal development, providing valuable insights into the essence of engineering design and how it differs from scientific discovery.**
26. Hornby, G., Globus, A., Linden, D. S. & Lohn, J. D. Automated Antenna Design with Evolutionary Algorithms. *AIAA Space*, 20060024675 (2006).
27. Arnold, F. H. Directed Evolution: Bringing New Chemistry to Life. *Angew. Chem. Int. Ed.* **57**, 4143–4148 (2018). **Provides an overview of directed evolution and its capability to create enzymes that can catalyse new to nature reactions by the Nobel prize winning scientist that pioneered the approach.**
28. Arthur, W. B. *The Nature of Technology: What It Is and How It Evolves*. (Free Press, 2009).
29. Sahal, D. *Patterns of Technological Innovation*. (Addison-Wesley Publishing Company, Advanced Book Program/World Science Division, 1981).
30. Wagner, G. P. & Altenberg, L. Perspective: Complex Adaptations and the Evolution of Evolvability. *Evolution* **50**, 967–976 (1996).
31. Ashe, A., Colot, V. & Oldroyd, B. P. How does epigenetics influence the course of evolution? *Philos. Trans. R. Soc. B Biol. Sci.* **376**, 20200111 (2021).
32. Jablonka, E. & Lamb, M. J. *Inheritance Systems and the Extended Evolutionary Synthesis*. (Cambridge University Press, Cambridge, 2020).
33. Luria, S. E. & Delbrück, M. Mutations of bacteria from virus sensitivity to virus resistance. *Genetics* **28**, 491–511 (1943).
34. Yona, A. H., Alm, E. J. & Gore, J. Random sequences rapidly evolve into de novo promoters. *Nat. Commun.* **9**, 1530 (2018).
35. Morrison, M. S., Podracky, C. J. & Liu, D. R. The developing toolkit of continuous directed evolution. *Nat. Chem. Biol.* **16**, 610–619 (2020).
36. Molina, R. S. et al. In vivo hypermutation and continuous evolution. *Nat. Rev. Methods Primer* **2**, 36 (2022).
37. Esvelt, K. M., Carlson, J. C. & Liu, D. R. A system for the continuous directed evolution of biomolecules. *Nature* **472**, 499–503 (2011).
38. Barrick, J. E. et al. Genome evolution and adaptation in a long-term experiment with *Escherichia coli*. *Nature* **461**, 1243–1247 (2009).
39. Nevoigt, E. et al. Engineering promoter regulation. *Biotechnol. Bioeng.* **96**, 550–558 (2007).
40. de Boer, C. G. et al. Deciphering eukaryotic gene-regulatory logic with 100 million random promoters. *Nat. Biotechnol.* **38**, 56–65 (2020). **The largest massively parallel reporter assay (MPRA) carried out to date, which helped to develop new models to support the rational design of complex gene regulatory elements.**
41. White, M. A. Understanding how cis-regulatory function is encoded in DNA sequence using massively parallel reporter assays and designed sequences. *Recent Adv. Funct. Assays Transcr. Enhanc.* **106**, 165–170 (2015).
42. Cai, Y.-M. et al. Rational design of minimal synthetic promoters for plants. *Nucleic Acids Res.* **48**, 11845–11856 (2020).
43. Pandi, A. et al. A versatile active learning workflow for optimization of genetic and metabolic networks. *Nat. Commun.* **13**, 3876 (2022).
44. Gilman, J., Walls, L., Bandiera, L. & Menolascina, F. Statistical Design of Experiments for Synthetic Biology. *ACS Synth. Biol.* **10**, 1–18 (2021).
45. Woodruff, L. B. A. et al. Registry in a tube: multiplexed pools of retrievable parts for genetic design space exploration. *Nucleic Acids Res.* **45**, 1553–1565 (2017).
46. Xu, P., Rizzoni, E. A., Sul, S.-Y. & Stephanopoulos, G. Improving Metabolic Pathway Efficiency by Statistical Model-Based Multivariate Regulatory Metabolic Engineering. *ACS Synth. Biol.* **6**, 148–158 (2017).
47. Carbonell, P., Radivojevic, T. & Garcia Martín, H. Opportunities at the Intersection of Synthetic Biology, Machine Learning, and Automation. *ACS Synth. Biol.* **8**, 1474–1477 (2019).
48. Castle, S. D., Grierson, C. S. & Gorochoowski, T. E. Towards an engineering theory of evolution. *Nat. Commun.* **12**, 3326 (2021). **Introduces the concept of the evotype to capture the evolutionary capacity of engineered biology with the goal of ultimately harnessing evolution for biological design.**
49. Cambray, G., Guimaraes, J. C. & Arkin, A. P. Evaluation of 244,000 synthetic sequences reveals design principles to optimize translation in *Escherichia coli*. *Nat. Biotechnol.* **36**, 1005–1015 (2018).
50. Tack, D. S. et al. The genotype-phenotype landscape of an allosteric protein. *Mol. Syst. Biol.* **17**, e10179 (2021).
51. Meyer, A. J., Segall-Shapiro, T. H., Glassey, E., Zhang, J. & Voigt, C. A. *Escherichia coli* “Marionette” strains with 12 highly optimized small-molecule sensors. *Nat. Chem. Biol.* **15**, 196–204 (2019).
52. Tarnowski, M. J. & Gorochoowski, T. E. Massively parallel characterization of engineered transcript isoforms using direct RNA sequencing. *Nat. Commun.* **13**, 434 (2022).
53. Chen, Y.-J. et al. Characterization of 582 natural and synthetic terminators and quantification of their design constraints. *Nat. Methods* **10**, 659 (2013).
54. Stanton, B. C. et al. Genomic mining of prokaryotic repressors for orthogonal logic gates. *Nat. Chem. Biol.* **10**, 99–105 (2014).
55. Bendixsen, D. P., Collet, J., Østman, B. & Hayden, E. J. Genotype network intersections promote evolutionary innovation. *PLoS Biol.* **17**, e3000300 (2019).
56. Ravikumar, A., Arzumanyan, G. A., Obadi, M. K. A., Javanpour, A. A. & Liu, C. C. Scalable, Continuous Evolution of Genes at Mutation Rates above Genomic Error Thresholds. *Cell* **175**, 1946–1957.e13 (2018).
57. Park, H. & Kim, S. Gene-specific mutagenesis enables rapid continuous evolution of enzymes in vivo. *Nucleic Acids Res.* **49**, e32–e32 (2021).
58. Voigt, C. A., Kauffman, S. & Wang, Z.-G. *Rational evolutionary design: The theory of in vitro protein evolution*. in *Advances in Protein Chemistry* vol. 55 79–160 (Academic Press, 2001).
59. Voigt, C. A., Mayo, S. L., Arnold, F. H. & Wang, Z.-G. Computational method to reduce the search space for directed protein evolution. *Proc. Natl. Acad. Sci.* **98**, 3778–3783 (2001).
60. Porebski, B. T. et al. Rapid discovery of high-affinity antibodies via massively parallel sequencing, ribosome display and affinity screening. *Nat. Biomed. Eng.* **8**, 214–232 (2023). **Demonstrates how existing high-throughput DNA sequencing technologies can be repurposed to enable the massively parallel screening of anti-body-antigen interactions opening routes to accelerated directed evolution utilising vast population sizes.**
61. Romero, P. A., Tran, T. M. & Abate, A. R. Dissecting enzyme function with microfluidic-based deep mutational scanning. *Proc. Natl. Acad. Sci.* **112**, 7159–7164 (2015).
62. DeBenedictis, E. A. et al. Systematic molecular evolution enables robust biomolecule discovery. *Nat. Methods* **19**, 55–64 (2022). **Combines phage assisted continuous evolution (PACE) with high-throughput robotics to enable more systematic and comprehensive evolution of new biological parts and systems.**
63. Zhong, Z. et al. Automated Continuous Evolution of Proteins in Vivo. *ACS Synth. Biol.* **9**, 1270–1276 (2020).
64. Huang, T. P. et al. High-throughput continuous evolution of compact Cas9 variants targeting single-nucleotide-pyrimidine PAMs. *Nat. Biotechnol.* **41**, 96–107 (2023).

65. Jumper, J. et al. Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589 (2021).
66. Baek, M. et al. Accurate prediction of protein structures and interactions using a three-track neural network. *Science* **373**, 871–876 (2021).
67. Watson, J. L. et al. De novo design of protein structure and function with RFdiffusion. *Nature* **620**, 1089–1100 (2023). **One of the first demonstrations of de novo protein design from simple molecular specifications using a diffusion-based machine learning model.**
68. Joshi, S. Y. & Deshmukh, S. A. A review of advancements in coarse-grained molecular dynamics simulations. *Mol. Simul.* **47**, 786–803 (2020).
69. Liu, W. et al. Rapid pathway prototyping and engineering using in vitro and in vivo synthetic genome SCRaMbLE-in methods. *Nat. Commun.* **9**, 1936 (2018).
70. Karr, J. R. et al. A Whole-Cell Computational Model Predicts Phenotype from Genotype. *Cell* **150**, 389–401 (2012).
71. Ahn-Horst, T. A., Mille, L. S., Sun, G., Morrison, J. H. & Covert, M. W. An expanded whole-cell model of *E. coli* links cellular physiology with mechanisms of growth rate control. *npj Syst. Biol. Appl.* **8**, 30 (2022).
72. Thornburg, Z. R. et al. Fundamental behaviors emerge from simulations of a living minimal cell. *Cell* **185**, 345–360.e28 (2022).
73. Elseman, I. E. et al. Whole-cell modeling in yeast predicts compartment-specific proteome constraints that drive metabolic strategies. *Nat. Commun.* **13**, 801 (2022).
74. Rees-Garbutt, J. et al. Designing minimal genomes using whole-cell models. *Nat. Commun.* **11**, 836 (2020).
75. Lenski, R. E. & Travisano, M. Dynamics of adaptation and diversification: a 10,000-generation experiment with bacterial populations. *Proc. Natl Acad. Sci.* **91**, 6808–6814 (1994).
76. Ellis, T. What is synthetic genomics anyway? *Biochemist* **41**, 6–9 (2019).
77. Pinglay, S. et al. Synthetic regulatory reconstitution reveals principles of mammalian Hox cluster regulation. *Science* **377**, eabk2820 (2022).
78. Zhao, Y. et al. Debugging and consolidating multiple synthetic chromosomes reveals combinatorial genetic interactions. *Cell* **186**, 5220–5236.e16 (2023).
79. Foo, J. L. et al. Establishing chromosomal design-build-test-learn through a synthetic chromosome and its combinatorial reconfiguration. *Cell Genomics* **3**, 100435 (2023).
80. Schindler, D. et al. Design, construction, and functional characterization of a tRNA neochromosome in yeast. *Cell* **186**, 5237–5253.e22 (2023).
81. Dymond, J. & Boeke, J. The *Saccharomyces cerevisiae* SCRaMbLE system and genome minimization. *Bioeng. Bugs* **3**, 168–171 (2012).
82. Blount, B. A. et al. Rapid host strain improvement by in vivo rearrangement of a synthetic yeast chromosome. *Nat. Commun.* **9**, 1932 (2018). **Uses the SCRaMbLE system to generate controlled genotypic variation in *Saccharomyces cerevisiae* that is beneficial to a range of biological engineering needs (violacein and penicillin biosynthesis and xylose utilisation).**
83. Williams, T. C. et al. Parallel laboratory evolution and rational debugging reveal genomic plasticity to *S. cerevisiae* synthetic chromosome XIV defects. *Cell Genomics* **3**, 100379 (2023).
84. Sheth, R. U., Cabral, V., Chen, S. P. & Wang, H. H. Manipulating Bacterial Communities by in situ Microbiome Engineering. *Trends Genet.* **32**, 189–200 (2016).
85. Siles, J. A. & Margesin, R. Insights into microbial communities mediating the bioremediation of hydrocarbon-contaminated soil from an Alpine former military site. *Appl. Microbiol. Biotechnol.* **102**, 4409–4421 (2018).
86. Bai, X. et al. Engineering the gut microbiome. *Nat. Rev. Bioeng.* **1**, 665–679 (2023).
87. Sun, Q. et al. Enhancing nutritional niche and host defenses by modifying the gut microbiome. *Mol. Syst. Biol.* **18**, e9933 (2022).
88. Gilbert, C. et al. Living materials with programmable functionalities grown from engineered microbial co-cultures. *Nat. Mater.* **20**, 691–700 (2021).
89. Swenson, W., Wilson, D. S. & Elias, R. Artificial ecosystem selection. *Proc. Natl. Acad. Sci.* **97**, 9110–9114 (2000).
90. Hanski, I. Metapopulation dynamics. *Nature* **396**, 41–49 (1998).
91. Levins, R. Some Demographic and Genetic Consequences of Environmental Heterogeneity for Biological Control. *Bull. Entomol. Soc. Am.* **15**, 237–240 (1969).
92. Klitgord, N. & Segrè, D. Environments that Induce Synthetic Microbial Ecosystems. *PLoS Comput. Biol.* **6**, e1001002 (2010).
93. Xie, L. & Shou, W. Steering ecological-evolutionary dynamics to improve artificial selection of microbial communities. *Nat. Commun.* **12**, 6799 (2021). **Shows how microbial communities can potentially be designed through the application of carefully crafted selective pressures, scaling up biological engineering beyond single cells.**
94. Arias-Sánchez, F. I., Vessman, B. & Mitri, S. Artificially selecting microbial communities: If we can breed dogs, why not microbiomes? *PLoS Biol.* **17**, e3000356 (2019).
95. Aulakh, S. K. et al. Spontaneously established syntrophic yeast communities improve bioproduction. *Nat. Chem. Biol.* **19**, 951–961 (2023).
96. Fedorec, A. J. H., Karkaria, B. D., Sulu, M. & Barnes, C. P. Single strain control of microbial consortia. *Nat. Commun.* **12**, 1977 (2021).
97. Karkaria, B. D., Fedorec, A. J. H. & Barnes, C. P. Automated design of synthetic microbial communities. *Nat. Commun.* **12**, 672 (2021).
98. Inda-Webb, M. E. et al. Sub-1.4 cm³ capsule for detecting labile inflammatory biomarkers in situ. *Nature* **620**, 386–392 (2023).
99. Tang, T.-C. et al. Hydrogel-based biocontainment of bacteria for continuous sensing and computation. *Nat. Chem. Biol.* **17**, 724–731 (2021).
100. Bongard, J. & Levin, M. Living Things Are Not (20th Century) Machines: Updating Mechanism Metaphors in Light of the Modern Science of Machine Behavior. *Front. Ecol. Evol.* **9**, 650726 (2021).
101. Bongard, J. & Levin, M. There's Plenty of Room Right Here: Biological Systems as Evolved, Overloaded, Multi-Scale Machines. *Biomimetics* **8**, 110 (2023).
102. Nicholson, D. J. Is the cell really a machine? *J. Theor. Biol.* **477**, 108–126 (2019).
103. Abil, Z. & Danelon, C. Roadmap to Building a Cell: An Evolutionary Approach. *Front. Bioeng. Biotechnol.* **8**, 927 (2020).
104. Moger-Reischer, R. Z. et al. Evolution of a minimal cell. *Nature* **620**, 122–127 (2023). **Recovers the fitness of an engineered cell with a rationally minimised genome using experimental evolution, demonstrating the power of combining traditional and evolutionary design methodologies.**
105. Richardson, S. M. et al. Design of a synthetic yeast genome. *Science* **355**, 1040–1044 (2017).
106. Pigliucci, M. Is evolvability evolvable? *Nat. Rev. Genet.* **9**, 75–82 (2008).
107. Wagner, A. Evolvability-enhancing mutations in the fitness landscapes of an RNA and a protein. *Nat. Commun.* **14**, 3624 (2023).
108. Ogbunugafor, C. B. Mutations that enhance evolvability may open doors to faster adaptation. *Nat. Commun.* **14**, 6310 (2023).
109. Bragdon, M. D. J. et al. Cooperative assembly confers regulatory specificity and long-term genetic circuit stability. *Cell* **186**, 3810–3825.e18 (2023).

110. Ceroni, F. et al. Burden-driven feedback control of gene expression. *Nat. Methods* **15**, 387–393 (2018).

Acknowledgements

We would like to thank Claire Grierson for insightful discussions on this topic that helped to shape this perspective. S.D.C. was supported by the EPSRC/BBSRC Centre for Doctoral Training in Synthetic Biology grant EP/L016494/1. M.S. was supported by an FWO travel grant (V408123N). T.E.G. was supported by BrisEngBio, a UKRI-funded Engineering Biology Research Centre grant BB/W013959/1, and a Royal Society University Research Fellowship grant URF/R/221008. The funders had no role in study design, data collection and analysis, and the decision to publish or prepare the manuscript.

Author contributions

S.D.C. conceived the work and wrote the first draft of the manuscript. T.E.G. and M.S. provided input and edited the manuscript. T.E.G. supervised the work.

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to Simeon D. Castle or Thomas E. Gorochofski.

Peer review information *Nature Communications* thanks Harrison Steel, and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at <http://www.nature.com/reprints>

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

© The Author(s) 2024