# Sequencing Guided Genetic Part Engineering

By

MATTHEW J. TARNOWSKI


School of Biological Sciences
UNIVERSITY OF BRISTOL

A dissertation submitted to the University of Bristol in accordance with the requirements of the degree of DOCTOR OF PHILOSOPHY in the Faculty of Life Sciences.

JUNE 2022

Word count: 49107

# Abstract

Microorganisms shape the living world through activities encoded in their DNA sequences. Synthetic biology often involves modifying the behaviour of microorganisms with designed DNA made up of genetic parts to achieve complex biological functionality. Large libraries of genetic designs can be constructed using DNA assembly and amplification, which could enable a broader understanding of how to design sequences encoding specific functions. However, methods to characterise sequence and function are often limited in scalability and scope since they can only characterise individual genetic parts. DNA and RNA sequencing open new possibilities for characterising entire DNA libraries. Here, we apply nanopore sequencing, a technology where DNA sequences are read as they pass through nanoscale pores (nanopores), to study assembled libraries of genetic parts. DNA sequencing shows that one-pot combinatorial DNA assembly by ligation reliably constructs libraries from multiple genetic parts and that library composition is significantly more uniform when DNA is amplified without protein expression. The libraries encode transcriptional terminator genetic parts that signal where transcribing RNA polymerases (RNAPs) should dissociate from DNA. We use terminators as 'transcriptional valves' to tune RNAP read-through and control transcript isoform abundance, offering a new mechanism to regulate genetic designs transcriptionally. We develop a method to characterise the *in vitro* transcription of valve libraries simultaneously at nucleotide resolution using nanopore direct RNA sequencing (dRNA-seq). This method reveals that upstream sequence context changes how much termination occurs, along with genetic design principles for tuning and insulating terminator function. Using this knowledge, we then engineer an array of CRISPR guide RNAs transcriptionally regulated by our valves. With DNA being the code of life, synthetic biology comes with a responsibility to the living planet. Therefore, a framework for responsible innovation is used to assess potential real-world impacts of the arrays. This work provides new avenues for studying DNA library composition, regulating transcription and innovating biotechnology responsibly, demonstrating the value of sequencing for exploring complex sequence-function landscapes.

# Dedication and acknowledgements

Dedicated to the unseen, the unknown and the unknowable.

I am thankful for the support of my friends, family, research group and collaborators both within and beyond the university.

This body of work was supported by the EPSRC/BBSRC Centre for Doctoral Training in Synthetic Biology grant EP/L016494/1 and BrisSynBio, a BBSRC/EPSRC Synthetic Biology Research Centre grant BB/L01386X/1.

## Author's declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: MATTHEW TARNOWSKI, DATE: 14/10/2022

# Table of Contents

# List of Tables

# List of Figures

CHAPTER 1

# INTRODUCTION

*"Who run the world? Girls"* – **Beyoncé**

A close look at a human being reveals billions of tiny organisms living on and inside of them. Being so small, these organisms are called microorganisms and many of them are single cells. A cell is a tiny droplet of water, encapsulated by a permeable membrane, containing millions of interacting molecules that act together to keep the cell alive. Molecules within a cell move and are exchanged with the outside environment to enable life. Nearly all biological life is made up of cells and we humans exist as many cells working together. In fact, a human being contains as many microbial cells as human cells [241]. These microorganisms cover our skin, our gut and virtually all parts of our body [8]. Even our "human" cells have interior components that are thought to have stemmed from individual microorganisms that evolved to live within other microbial cells in a process known as symbiosis [101, 228]. Despite being invisible to the naked eye, microorganisms are integral for our survival and help run the living world (**Figure 1.1 A**).

Even though we can't see them, microorganisms are everywhere, they are the invisible glue between all life. They are essential in the production of many foods and drinks like beer, bread, wine, yoghurt and cheese [132]. They can also be used to make medicines, fibres and fuels. To produce these products, microorganisms require some kind of energy. Since cellular life arose on this planet over 3 billion years ago [219], microorganisms have evolved to grow using many sources such as air [152], sunlight [273] and plants [53]. Microbes also invariably live together as large and often diverse communities [298]. Scientists frequently perform experiments to understand how microorganisms live and explore ways to modify them to alter the by-products that they produce.

Most living organisms, including microorganisms, follow instructions encoded in their DNA, sometimes referred to as the code of life. DNA is a little like a recipe book for the microorganism

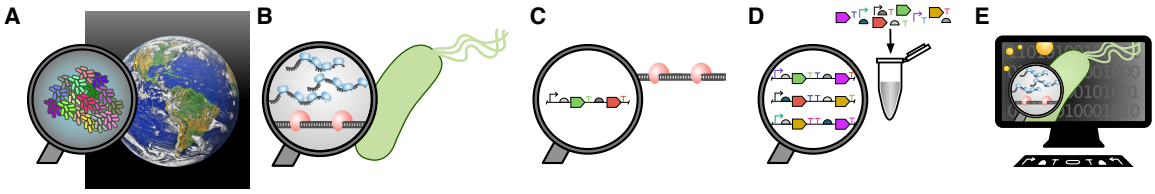



Figure 1.1: **Microbiology, molecular biology, synthetic biology (A)** Biological life on this planet arises from billions of microorganisms living in communities. **(B)** Each microorganism reads DNA (bottom) using transcription with RNA polymerases (red) to produce RNA (top) followed by translation into proteins using ribosomes (blue). **(C)**. Sequencing enables the study of the sequence and function of both RNA and DNA, revealing genetic parts with specific functions such as promoters (arrow), terminators (T), genes (block) and ribosome binding sites (semi-circle). **(D)** Synthetic biologists combinatorially assemble genetic parts into large libraries of genetic designs and study them with sequencing. **(E)** Sequencing datasets enable predictive design of genetic parts for engineering microorganism DNA and behaviour.

and is referred to as its "genome". It is written in a genetic code containing a four-letter alphabet (each letter being one of four possible nucleotides). Microorganisms follow the recipes written in their DNA sequence to create the molecules they need to live and this shapes their behaviour (**Figure 1.1 B**).

## 1.1 Synthetic biology

All aspects of this living biological planet arise from a vast number of interactions of molecules and organisms, each of which could be considered a bio-technology. Evolving from this milieu to play an important ecological role [185, 248], humans have always interacted with the living world. For millenia, humans have engineered microbiology by selecting particular groups of microorganisms for fermenting using favourable environmental conditions and maintaining these microbial cultures over long periods of time [132]. During this process the microorganisms adapt to their environment through mutations in their DNA [38]. In this way, the input that the microorganisms use to grow can be changed slowly, allowing the microorganisms to adapt [153].

In the past two decades, synthetic biologists have taken a different approach to engineering microorganisms: modifying their DNA directly. Recent innovations in DNA technologies brought about the ability to genetically engineer microorganisms [15]. DNA synthesis technology has enabled the chemical synthesis of defined DNA sequences, which can be ordered from a DNA synthesis company [144]. The longer a DNA sequence is, the more difficult and expensive it is to synthesise [144]. Therefore, complex sequences are often split up into multiple pieces, which are synthesised separately and then assembled (**Figure 1.1 D**) [41]. The sequences are often designed based upon sequences found in native microbial genomes using DNA sequencing [125].

DNA sequencing involves reading the sequence of DNA molecules [244]. A single microorganism often contains millions of nucleotides of DNA, so this is quite a feat. The combination of DNA synthesis and DNA sequencing makes design and characterisation of DNA sequences possible [72].

Synthetic biology is notoriously difficult to define. However a definition from a recent primer in synthetic biology will suffice: *"Synthetic Biology is an exciting and rapidly evolving interdisciplinary research field which aims to provide a systematic framework for the engineering of biological systems and cells at the genetic level."* [15]. Shortly after the emergence of synthetic biology as a concept, four challenges that limit the genetic engineering of biology were outlined: managing biological complexity, constructing and characterising synthetic biological systems, spontaneous physical variation of biological system behaviour, and evolution [68]. To deal with these challenges, three ideas from structural engineering were put forward: standardisation, decoupling and abstraction [68]. The registry of standard biological parts offers free access to standardised DNA sequences encoding basic biological functions that can be used to engineer many-component synthetic biological systems quickly and reliably [245]. These parts encode biological functions which are organised across levels of complexity using abstraction hierarchies from the level of DNA sequence to parts, to devices and ultimately systems [68]. This enables decoupling of complex biological problems into multiple simpler problems at the level of a single sequence, part, device or system that can be dealt with independently before combining to produce a functioning whole [68]. These are just a few of the many approaches that synthetic biologists use for engineering biology.

The "design-build-test-learn" (DBTL) cycle approach to engineering is commonly utilised to engineer synthetic biology [36]. DNA sequencing and DNA synthesis offer the tools for building and testing genetic designs. Using these tools iteratively allows sequencing to guide genetic engineering using the DBTL cycle. In order to engineer the DNA of microorganisms, synthetic biologists must understand how to read and write the genetic code [187], however, the design of functional DNA sequences from scratch is a challenge since the complexity of DNA sequences is astounding [168]. Therefore, synthetic biologists often undertake experiments to understand what the function of many different DNA sequences are in a single experiment to build up a picture of how sequence encodes function (**Figure 1.1 D**) [85]. This involves synthesising and assembling novel DNA sequences and characterising their functions en masse [187]. By doing so, they can begin to understand the genetic code of DNA, allowing them to understand how to create DNA sequences encoding specific functions (**Figure 1.1 E**) [277]. With this knowledge they seek to create genetic designs that change the behaviour of microorganisms, for example enabling cells to produce molecules they deem useful. This might include medicines, fuel, food and fibre [102].

## 1.2 Genetic parts

Voigt, 2006 – *"The genome contains commands dictating how cells eat, reproduce, communicate, move and interact with their environment. Cells can be programmed by introducing synthetic DNA containing new commands that instruct the cell to perform a set of artificial tasks in series or in parallel."* [277]

From the outset, synthetic biology approaches have sought to use standardised and parameterised "genetic parts" for genetic engineering [68, 277]. A standard biological part (or genetic part) has been defined as a genetically encoded object that performs a biological function and that has been engineered to meet specified design or performance requirements [35]. Genetic parts are most often engineered from examples in genomes [35] to fulfil a specific function which controls a cellular process. Ideally, genetic parts are built to be robust: capable of functioning in any cellular environment [277]. Such genetic parts can then be used to predictively engineer microorganisms [196], changing their behaviour [102]. There are many types of genetic part encoded in the DNA of microorganisms [68] and sequencing experiments continue to reveal how the sequence of specific genetic parts encodes function [32, 125, 196]. Well characterised genetic parts can then be combined to engineer more complex biological systems.

There are two key types of molecule which a cell produces from its DNA: RNA and protein [143]. Whilst proteins are sometimes described as the molecular machinery of the cell, RNA molecules are essential for their production and also carry out additional functions of their own [6]. The production of proteins relies on two processes: transcription and translation (**Figure 1.1 B**) [143]. Transcription involves making an "RNA" copy of the DNA sequence, an RNA molecule and it is an RNA polymerase that performs this process. Often the RNA molecule is then used as a template to produce a protein molecule. This occurs through a process of translation: the RNA sequence is translated in to one or more protein sequences by a ribosome.

Genetic parts can be used to control the processes of transcription and translation and in doing so, regulate which RNA molecules are produced from DNA sequences [276] and which proteins are made from those RNA molecules. The genetic part which initiates the process of transcription of DNA is referred to as a promoter [15] and the genetic part which terminates transcription of DNA is referred to as a terminator [15]. In this thesis we focus on characterising the function of terminator genetic parts.

## 1.3 Terminator genetic parts

The function of a terminator genetic part is to regulate where transcription should stop. At a particular position in the genome, transcription can be terminated to varying degrees ranging from none (0%) to full (100%) [47]. The degree to which termination occurs is referred to as the termination efficiency. The sequence of all genetic parts encodes their function; for terminators this is their capacity to stop transcription. Through studying terminator sequences and their

functions, principles to design transcriptional terminator sequences can be elucidated. Although terminators have an important role in controlling the transcription of DNA and are widely used in synthetic biology, there is limited characterisation data for many designs. Furthermore, whilst promoter strength is well known to vary, variation in termination efficiency has been somewhat overlooked and is currently in the spotlight. Lalanne *et al.* [149] and Sorek *et al.* [57] recently revealed that terminators in many microbes rarely completely stop the process of transcription. Instead, termination is normally incomplete and this feature is used to tune the amount of RNA polymerase (RNAP) which continues transcription past the gene. Terminators act a little more like valves, which, like a valve in a pipe, influence the flow of transcribing RNAPs along DNA. This is contrary to the prevailing view in synthetic biology which sees terminators as stop signs, prohibiting any transcription into nearby regions of the DNA [192].

## 1.4 Engineering genetic parts using sequencing

Sequencing technologies enable both DNA and RNA molecules to be read, referred to as DNA sequencing [244] and RNA sequencing [251], respectively. A sequencing read is generated containing partial or complete sequences of the DNA or RNA molecules studied. Whilst there are many advantages and disadvantages to each of the many sequencing technologies [244, 251], one important factor is the length of the sequencing reads that can be produced as this determines the kinds of studies that can be performed. Until recently Sanger sequencing and sequencing-by-synthesis (i.e. Illumina sequencing) have been the norm, yet both are limited to short reads (~1000 nucleotides and ~300 nucleotides respectively) [244]. Both involve the creation of a copy of the sequenced molecule where each nucleotide has a unique label; the sequence is "read" by measuring the sequence of labels, revealing the initial sequence. Both methods have high accuracy (>99.9%) and sequencing-by-synthesis has the benefit of being able to characterise many more reads than Sanger sequencing. However, sequencing-by-synthesis requires fragmentation of the DNA or RNA sample and therefore necessitates computational assembly of the resulting short sequencing reads using overlaps to generate a complete sequence. Both of these methods are also limited by the relatively short reads and indirect measurement of a copy of the sequence, which results in loss of epigenetic information. In the past decade, technologies have emerged that enable much longer sequencing reads to be recovered directly from DNA and RNA molecules [115].

Nanopore and Pacific Biosciences (PacBio) sequencing are both long read sequencing technologies which measure sequences directly and indirectly, respectively [115]. Nanopore sequencing is a relatively recent innovation, launched commercially for DNA and RNA in 2015 and 2018, respectively. It involves "reading" DNA or RNA molecules passing them through protein nanopores embedded in a membrane and measuring the disturbance in ionic current [94]. The benefit of nanopore sequencing is that it is a "long read" method capable of reading sequences of any length

[94]. This opens opportunities to study sequences that cannot be characterised using "short read" sequencing, which is limited to reading sequences up to length 300 nucleotides and necessitates fragmentation of the native DNA or RNA molecules in preparation for sequencing [94]. Whichever sequencing technology is used, sequencing experiments typically generate millions of reads and computational tools are important for processing the datasets that are generated [240] (with the exception of Sanger sequencing, which generates a single read per reaction). Through comparing the abundance of RNA to the DNA sequences using sequencing, scientists can elucidate genetic parts that control the production of RNA from DNA [125] and use them to engineer microorganisms [98].

Synthetic biologists use genetic parts and novel genes as components to engineer genetic constructs, which are often long sequences thousands of nucleotides in length. Such genetic constructs can then regulate the levels of several genes simultaneously [198]. Host microorganisms can use the proteins produced by the genes to metabolise a substrate into a product. More complicated constructs can be created that make computations: decisions based on stimuli from their internal or external environment [102, 198] and these are commonly referred to as genetic circuits. Genetic constructs are also generally put together on a circular self-replicating piece of DNA called a plasmid and each genetic construct is referred to as a genetic design (**Figure 1.1 D**). Genetic parts with predictable function are necessary for rationally engineering genetic designs. However, it is known that genetic part function can be affected by the surrounding genetic sequence [301], the function of surrounding genetic parts [37] and cellular context [125]. Therefore, synthetic biologists have devised methods to generate and test the function of large pools (referred to as libraries) of genetic designs.

Generally, DNA libraries are produced by DNA synthesis and assembly (**Figure 1.1 D**). A multitude of DNA assembly methods are available for the creation of libraries of combinations of genetic parts from recombinant and synthesised DNA [67]. In order to characterise a library of constructs evenly, each one must be present with a similar abundance. It is therefore important that the DNA assembly of the library results in a relatively uniform distribution of library members [139]. Through testing the function of all of the assembled part combinations, genetic part dependence upon genetic context can be better understood. Methods exist for characterising genetic parts individually [47, 198], however, these are slow and time and resource intensive. The ability to read DNA sequences (DNA sequencing) enables collections (referred to as "pools" or "libraries") of unique genetic parts to be characterised simultaneously [23, 65, 163]. This is referred to as multiplexed sequencing since all the genetic parts are sequenced together (**Figure 1.1 D**) and then sequencing reads belonging to each genetic part are identified using their unique sequences. Multiplexed sequencing of RNA is also possible, which enables the functions of pools of genetic parts to be measured too. The datasets from multiplexed sequencing assays can be used to predict the function of genetic parts from their sequence computationally (**Figure 1.1 E**), a prerequisite for predictive design of genetic circuits [23, 301].

## 1.5 Responsible research and innovation

Whilst the impacts of research and innovation cannot be predicted, they can be anticipated and reflected upon. This can be prescient, given that regulation often lags behind innovation [200]. In the past century, science and technology has been studied [182]. This led to the formation of ideas for guiding science and technology such as anticipatory governance, technology assessment and ethical, legal and social aspects of technologies [201]. These foundational ideas led to the field of responsible research and innovation (RRI) [201]. A framework for responsible research and innovation has been published and endorsed by the UK Engineering and Physical Sciences Research Council (EPSRC). The AREA framework encourages anticipating future impacts of a technology (A), reflecting upon them (R), widely engaging and deliberating over them (E) and acting (A) to influence the direction of the research [202].

This thesis has offered an opportunity to explore how to research and innovate responsibly in the context of synthetic biology. New genetic parts enable novel applications and uses of synthetic biology, which have impacts beyond the technology itself [211, 290]. However, the necessity for researchers and innovators to be accountable for the technologies they develop is contested [201, 259].

Tools are essential to facilitate responsible innovation. One such tool has been developed based on the work of Ivan Illich, who outlined "convivial" [117] as a technical term to designate a modern society of responsibly limited tools. Recently, Vetter *et. al* [274] transformed the ideas of Illich in to a matrix, the matrix of convivial technology (MCT). This matrix assesses four life-cycle levels of a technology in terms of five dimensions of living well.

With so many novel bio-technologies developed in the past century, asking how they can be developed responsibly, though hard to contemplate, could help shape a healthy living world for all. Therefore, this thesis culminates in two activities in responsible innovation. Firstly, the MCT is adapted to enable visual comparisons of technology assessments. This activity leads to anticipation of the impacts of technologies and it is used to assess the technology developed in this thesis. Then, an account of the personal research journey undertaken in this thesis is used to reflect upon the origins of the research and alternative approaches to innovation. Together, these activities facilitate discussion of the research and action based on the findings, thus initiating all aspects of the AREA framework for responsible innovation [202].

## 1.6 Motivation

At a high-level, the motivation of this thesis was to develop methods and genetic parts for engineering biotechnologies that are non-polluting, biodegradeable and renewable. The high-throughput experimental methods and large genetic design libraries that we designed could test multiple DBTL cycles simultaneously and generate datasets which could remove the need for DBTL cycles entirely by predicting part function from sequence. The important role of terminators

in transcriptional regulation is emerging [149] and lent itself to study using the newly available nanopore direct RNA sequencing technology. How terminators are affected by upstream sequence context has many unknowns, yet this region could be used to tune termination efficiency. Being the first technology to sequence RNA and DNA directly, nanopore sequencing was well-suited yet seldom used in characterising the large genetic designs used synthetic biology. Methods were available for constructing large and diverse libraries of genetic designs however their sequence composition and function were rarely scrutinised exhaustively. Nanopore sequencing could enable nucleotide resolution, going beyond the summary statistics often used to characterise genetic parts. Furthermore, it did not require fragmentation or amplification, a step in previous sequencing methods that leads to loss and bias of sequence information, respectively. *In vitro* synthetic biology is a growing area of research and presents a simple experimental setting for developing a new method, motivating us to engineer genetic parts for this context, with a commonly used and transferable biomolecule: T7 RNAP. A further aim was to demonstrate how the characterised terminators could be applied in synthetic biology, leading us to explore this in the context of CRISPR guide RNAs which are frequently expressed as arrays. When applied, genetic part arrays could form part of technologies used in socioeconomic systems and, intended or not, have the potential to form part of the interconnected biosystem on this living planet. Responsible innovation is rarely considered at this early stage of research and there is an absence of suitable tools for doing this. Such tools could guide later innovation and therefore, an effort was made to consider the potential impacts of newly developed biotechnology tools.

## 1.7 Thesis overview

This thesis begins with a literature review concerning the background underlying this work in Chapter 2. Then, in Chapter 3, we give details of the materials and methods used to undertake this research. In Chapter 4 we develop experimental and computational methods for assembly and multiplexed nanopore DNA sequencing of combinatorial libraries of genetic part sequences. We use these methods to show how DNA library amplification protocols affect DNA library composition and mutation. In Chapter 5, we adapt these methods to study the function of a library of sequences encoding terminators and upstream sequence context. We use nanopore direct RNA sequencing to generate read profiles where termination can be measured as a drop in sequencing reads. Some read profile features beyond termination events are evident and through studying them we optimise the experimental method and develop a model of one of the features to improve termination efficiency measurements. Following these improvements, we characterise termination of the full library at nucleotide resolution. This shows the importance of upstream sequence context in determining termination efficiency and that terminators can be used as "transcriptional valve" genetic parts capable of controlling the degree to which transcription is stopped at a certain point. In Chapter 6 we use our insights and methods to design further valve

libraries exploring first how terminators can be tuned and insulated. Then, how base-pairing and structural interactions between the core terminator and upstream sequence context can influence termination efficiency. We use some of the valves to engineer an array of consecutive RNA genetic parts (CRISPR guide RNAs) whose stoichiometry is controlled at the level of transcription. In Chapter 7, an RRI tool, the matrix of convivial technology (MCT), is used to assess different approaches to innovating valve arrays in terms of the potential wider impacts they may have. Finally, in Chapter 8, we draw conclusions on combinatorial DNA library assembly by ligation, terminator design, use of the MCT to evaluate synthetic biology tools, and the merits and limitations of using multiplexed nanopore sequencing for genetic part engineering. The thesis culminates with a discussion of future directions: engineering DNA library composition, *in vivo* characterisation, uses of transcriptional valves and responsible bio-engineering.

## 1.8 Publications and scientific contributions in this thesis

The primary scientific contributions of this thesis involved studying the *in vitro* termination of T7 RNA polymerase by libraries of terminators (including arrays of terminators) at nucleotide resolution using nanopore direct RNA sequencing. This revealed how terminator and upstream genetic sequence influences termination efficiency. This work is covered by Chapter 5 and 6 and has been published in [257]:

- Tarnowski, M. J. and Gorochowski, T. E. (**2022**). Massively parallel characterization of engineered transcript isoforms using direct RNA sequencing, *Nature Communications*, 13:1, 434.

Further scientific contributions in Chapter 4 show that ligation can be used to combinatorially assemble large DNA libraries from variants of three genetic parts and a plasmid backbone. This was confirmed using nanopore DNA sequencing, which also showed that DNA library amplification protocols alter DNA library composition.

Two perspectives were co-authored which influenced the thesis but do not form a core part of its content [102, 180]:

- Greco, V., Tarnowski, M. and Gorochowski, T. E. (**2019**). Living computers powered by biochemistry. *The Biochemist*, 41:3, 13-18.

- Manrubia, S. *et al.* (**2021**) From genotypes to organisms: State-of-the-art and perspectives of a cornerstone in evolutionary dynamics, *Physics of Life Reviews*, 38, 55-106.

CHAPTER 2

# BACKGROUND

## 2.1 Engineering biology

### 2.1.1 From microbiology to synthetic biology

This living planet has been engineering itself since microbiological life arose 3.5 billion years ago. For roughly three billion years, microorganisms carried this out, eventually giving rise to multicellular life around half a billion years ago. Microorganisms continue to live inside and on the surface of multicellular organisms to this day, participating to some degree in their activities. Many kinds of multicellular organism have lived here. One of the youngest, *Homo sapiens* has been engineering biology one way or another in the 300,000 years that it has been around [25, 261]. This started with selection of plants and animals by hunter-gatherers [177, 249] under an economy of the commons [26] and a philosophy of animism [112] and humans as part of nature [231]. Many humans around the world maintain such cultures [231]. The biotechnologies developed by this approach are agroecological: providing humans food whilst being part of the ecosystem [1]. Humans engineer the composition of microorganisms in the environment in many ways for example with fermentation [132] and the organic matter that they generate [90]. Thus, humans have been engineering biology for some time before the present [216].

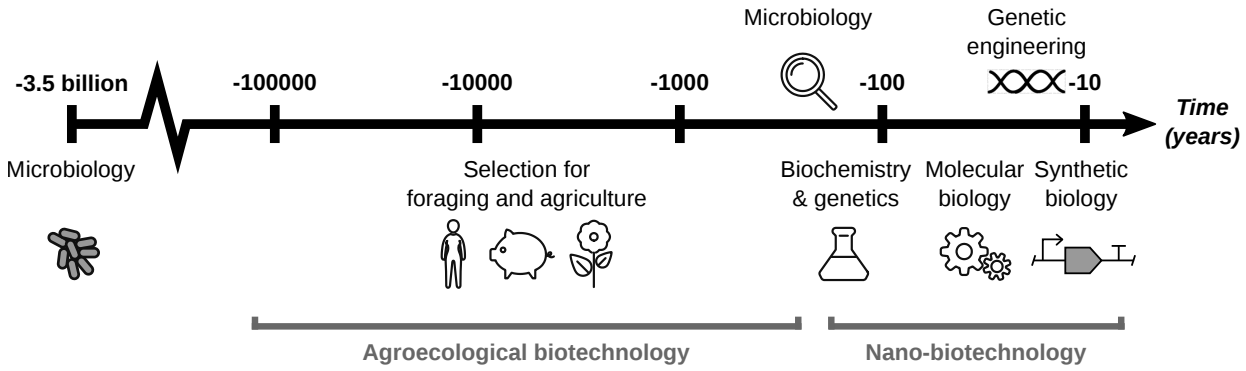



Figure 2.1: **A timeline of approaches to engineering biology.**

The approach to engineering biology both changes and is changed by the ways that humans organise themselves politically, socially, economically and ecologically. Humans have organised themselves in a multitude of ways [100]. Whilst it is an oversimplification, two approaches to engineering biotechnology have been common throughout the majority of human history: foraging and agriculture and they are not mutually exclusive [100]. However in recent history, changes to political, social and economic systems have led to different approaches to engineering biotechnology. Around 500 years ago, a capitalist economy began to dominate [162] along with a philosophy of humans separate from nature and seeking to control it [154]. In the relatively short time since then, some humans controlled and enslaved others, put oil, metals and other extractable resources to use at scale, concentrated power, and generally disrupted the living planet [162]. Separated from land-based cultures, knowledges and philosophies of integration, humans sought to understand and control nature using the scientific method and a philosophy of reductionism [154].

Microbiologists studied the invisible organisms living around them, leading them to discussions of their biochemistry and genetics at the beginning of the twentieth century [193]. It has been proposed that molecular biology is the fruit of the convergence of these two disciplines [193]. Genetics became a leading biological science due to its discoveries and influence on agronomy, leading to farmers and seed merchants contributing to the development of genetics research [193]. Similarly various factors played a role in shaping molecular biology, such as the migration of scientists, growing global communication needs, a focus on code-breaking and the linked birth of computing [193].

From molecular biology, genetic engineering and its successor, synthetic biology arose. In contrast to agroecological biotechnologies, these are nano-biotechnologies [93]. They operate at the nanoscale, often involving changing the genetics within cells themselves. All interactions within this living world could be considered biotechnology. However the Western culture [263] from which this PhD arises generally considers the term biotechnology to mean nano-biotechnology. Synthetic biology is a relatively new field that aims to design and genetically engineer organisms and molecules with specific functions [15]. It relies on recent research and innovation which has enabled humans to read and write sequences of DNA, the code of biological life.

### 2.1.2 Foundations of molecular biology, genetics and synthetic biology

Over the past 150 years, scientists have transformed our understanding of how cells work. Around the turn of the twentieth century, an understanding of genetics grew, beginning with studies of inheritance. This led to the definition of genotype and phenotype in 1909 [193]. Genotype refers to factors that are transmitted down generations; these factors are called genes. Phenotype refers to the totality of characters; many characters have several forms that can be easily distinguished [193]. Eventually it was understood that genes encode the function of proteins (otherwise known as enzymes) [143].

Whilst deoxy-ribonucleic acid (DNA) was first chemically studied in the 1930s it was not until the 1950s that its structure was characterised [193]. DNA is a polymer made from four nucleotides: adenine (A), guanosine (G), cytosine (C) and thymine (T) (**Figure 2.2 A**). Together these make up the genetic code. These nucleotides comprise a base (that is, a negatively charged chemical structure) attached to a phosphate backbone via a sugar chemical structure. Many bases are attached to the phosphate backbone in a specific order, giving rise to a sequence of nucleotides. Double-stranded DNA (dsDNA) is the most prevalent form of DNA and the two strands form a double-helix structure (**Figure 2.2 B**). This structure arises due to hydrogen bonds which can form between the nucleotides of the two strands. The two strands are "base-paired" when the hydrogen bonds have formed and only certain nucleotides can base-pair with one another. G and C can base-pair via three hydrogen bonds whilst A and T can base-pair via two hydrogen bonds (**Figure 2.2 A**). Whilst each hydrogen bond in dsDNA is weak and can easily be broken, collectively these bonds hold the dsDNA in a helical structure.

The structure of another nucleic acid was discovered shortly after: ribonucleic acid (RNA). Like DNA, RNA is made up of nucleotides, however, these are A, C, G and U (uridine). RNA strands cannot form a double-helical structure like DNA and instead the most common form of RNA in cells is single-stranded RNA (ssRNA) which is less stable than DNA. Being single-stranded, internal base-pairs can form within RNA molecules (**Figure 2.2 C**). These base-pairs cause the RNA sequence to fold into particular structures. These structures generally comprise "hairpin" structures, where nearby nucleotides base-pair. The base-paired nucleotides form a stem and at the top of the stem is a loop, containing some unpaired nucleotides. RNA structures

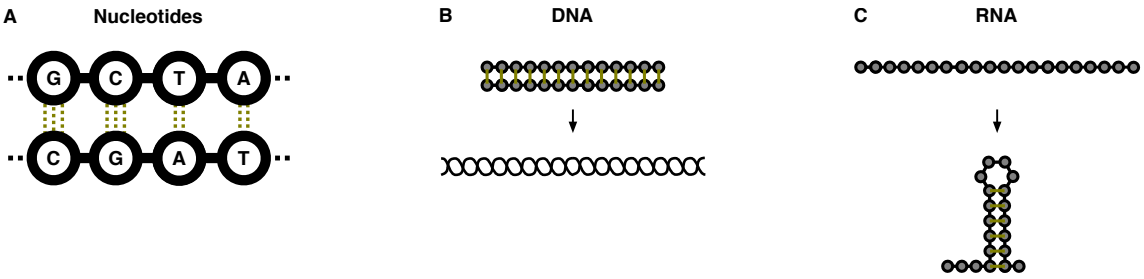



Figure 2.2: **Base-pairing of nucleotides leads to structures in DNA and RNA.**

can encode a variety of functions.

The relation between DNA, RNA and proteins was elucidated in the 1950s and is summarised by the "central dogma": "DNA makes RNA makes protein" (**Figure 2.3**) [193]. The process of making DNA from RNA is known as transcription and requires the action of an RNA polymerase. An RNA polymerase (RNAP) can bind DNA and makes a complementary copy of it in RNA, using ribonucleotides. This RNA molecule can then be translated into protein using a ribosome. A ribosome uses transfer-RNAs (tRNAs), which contain both an RNA-recognition element and an amino acid, to translate an RNA sequence into a sequence of amino acids. Amino acids are far more complex than nucleotides and most organisms use 20 different amino acids to make proteins. The sequence of amino acids folds into a complex structure, which carries out particular functions within the cell. The central dogma also proposes that RNA can be converted back to DNA in a process known as reverse-transcription (RT), using a reverse-transcriptase. The processes involved in the central dogma are now known to be connected, for example via transcription-translation coupling where the ribosome and the RNAP interact with one another [120]. Furthermore, whilst the RNAP transcribes, RNA is able to fold in a process known as co-transcriptional folding [188]. Along with DNA replication by DNA polymerase, these are the fundamental processes that genetic engineers and synthetic biologists use to engineer biology at the nanoscale.

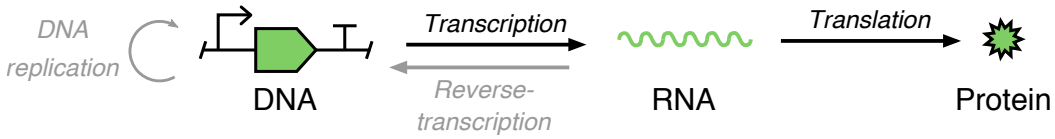



Figure 2.3: **The central dogma of molecular biology.**

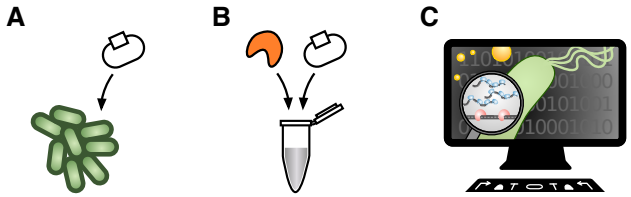



Figure 2.4: **Approaches to synthetic biology.** (**A**) *In vivo* synthetic biology. (**B**) *In vitro* synthetic biology. (**C**) *in silico* synthetic biology.

DNA sequences encode a variety of features. Those segments encoding proteins are referred to as "coding" regions. However, the cell must carefully tune the amount of each protein produced and this is achieved using "non-coding" regions. Non-coding regions are involved in regulating the production of DNA, RNA or protein. Various assays have been developed for characterising the non-coding and coding regions of DNA. It is through understanding how DNA sequence encodes function that nanoscale biotechnologies can be engineered.

Various tools have been developed for microbiology, molecular biology and genetic engineering. Particular strains of cells suited to different purposes have been developed. A frequent activity involves putting designed DNA inside of bacterial cells. This process is known as transformation and can be achieved through shocking the cells via heat or electricity.

Genetically engineered organisms and molecules are the foundations of *in vivo* and *in vitro* synthetic biology, respectively [55] (**Figure 2.4**). *In vivo* synthetic biology involves manipulation of cells directly, often using DNA to modify their behaviour. DNA is added to the host microorganism via transformation, where it is expressed. This process is known as cloning. Transformation often involves adding plasmid DNA: plasmids are circular DNA sequences frequently used by microorganisms to share DNA. Plasmids encode an origin of replication, where DNA replication machinery is recruited to make copies of a plasmid and maintain a number of plasmids within each cell as it replicates. Viruses can also be used as vectors to mediate expression of molecules within cells, such as the T7 bacteriophage [281]. *In vitro* synthetic biology on the other hand involves modification and combinations of biomolecules such as DNA, RNA and proteins in a test tube. *In vitro* synthetic biology is often referred to as "cell-free", where the processes of transcription and translation are facilitated using reactions in test tubes [84, 262]. Besides these approaches, *in silico* studies and experiments are becoming increasingly important for studying the large datasets generated by experiments simulated on the computer [76].

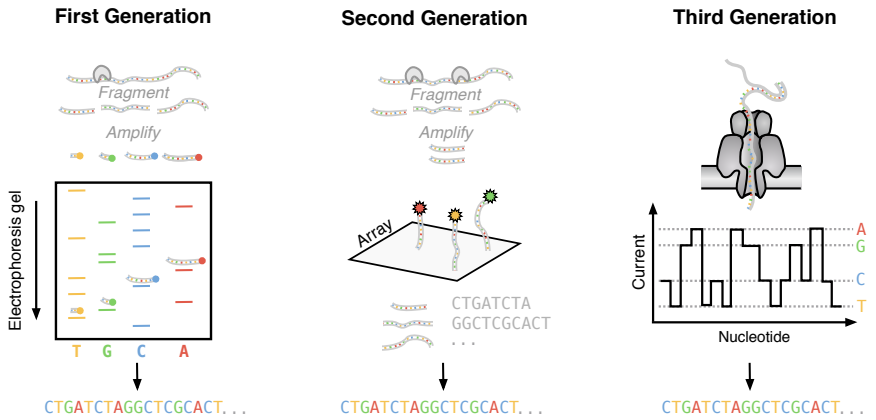
## 2.2 Sequencing

### 2.2.1 DNA sequencing

DNA was first reported 150 years ago [215] and just over a century later the first molecule of DNA was "read", in a process known as DNA sequencing (DNA-seq) [115]. DNA sequencing methods enable users to read the sequence of nucleotides encoded in DNA molecules and there are a variety of approaches (**Figure 2.5**). Since the first strand of DNA was sequenced, research and innovation has led to high-throughput methods capable of reading millions of DNA molecules in a single experiment [115]. However, at the outset, approaches to DNA sequencing were time-consuming (1 nucleotide per month [244]) and incremental advances have vastly increased its capabilities. Regardless of the DNA sequencing approach, the DNA sample must be prepared for sequencing by a particular technology. This process is referred to as sequencing library preparation and differs from DNA library assembly which involves assembling DNA molecules into a larger DNA molecule. After sequencing library preparation, the DNA can be sequenced using the appropriate sequencing technology, each of which has merits and limitations (**Figure 2.5**).

The first step-change in sequencing was around 1976 when two methods were developed that could sequence hundreds of nucleotides in one day [244]. One of these early DNA sequencing methods was Sanger sequencing [232] and it is frequently used to this day (**Figure 2.5**). This method involves using a primer that binds the DNA molecule to be read, the primer is then extended using DNA polymerase. Four extension reactions are completed, each with a small amount of a labelled chain-terminating nucleotide (A, C, T or G). Each reaction produces fragments of different lengths, which correspond to DNA molecules ending in that particular chain-terminating nucleotide. The sizes of the fragments from each reaction are measured by electrophoresis on polyacrylamide slab gels. This separates the DNA fragments by size, revealing the sequence of nucleotides it encodes. Each reaction is put into a different lane on the gel, which is put onto X-ray film. Imaging reveals a ladder from which the sequence can be read off by looking at the lane each fragment appears in and reading the fragments in order of size to infer the order of the bases (**Figure 2.5**) [244].

For the next decade, Sanger sequencing was further refined, resulting in machines that could sequence around 1000 nucleotides per day. Since genomes are large (frequently millions or billions of nucleotides long), an approach was developed to augment Sanger sequencing so that it was able to sequence such large DNA sequences. This is referred to as shotgun sequencing and it involves making copies of random regions across a genome, sequencing them and assembling the genome sequence using the overlaps between the sequencing reads [244]. At this point, the amount of genome-scale sequencing data grew exponentially and data repositories were created. This led to tools for searching through the sequencing data and a spirit of sharing data [244]. By 1986, nearly 10 million nucleotides were available. By 2001, a few academic genome centres generated up to 10 million nucleotides per day using automation, a draft sequence of the human




| ***Capability*** | *Sanger-sequencing* | *Sequencing-by-synthesis* | *Single-molecule-sequencing* |
|---|---|---|---|
| *Sequenceing reads* | Short read | Short read | Long read |
| *Max read length* | 1000 nt | 300 nt | > 1000000 nt |
| *Error-rate (per nt)* | < 0.1 % | < 0.1 % | 1-10 % |
| *Multiplexable* | No | Yes | Yes |
| *Full length* | No | No | Yes |
| *Portable* | No | No | Yes |
| *Direct (no fragmentation or amplification)* | No | No | Yes |

Figure 2.5: **Comparison of sequencing technologies.**

genome was produced and the cost of a 700 nucleotide sequencing read was approaching US$1. Sanger sequencing had come a long way through optimisation of wet lab protocols and software, however, the rate of improvement in throughput was slowing [244] and the next generation of sequencing technologies was emerging.

Today, billions of nucleotides can be sequenced in a single DNA sequencing experiment: what are the innovations that made this possible? Within a decade of the completion of the human genome project (2004), “next-generation” DNA sequencing (NGS) was commonplace and had all but superseded Sanger sequencing [244]. NGS technologies took a different approach to Sanger sequencing, the transformative change being multiplexing [244]. Instead of one sample per reaction, a complex library of DNA molecules is immobilised onto a 2D surface referred to as an array (**Figure 2.5**). A fluorescently labelled DNA molecule is synthesised from each DNA molecule, giving rise to the name for this technology: sequencing-by-synthesis. These technologies decreased the raw per-nucleotide cost by four orders of magnitude between 2007 and 2012 [244].

Sequencing-by-synthesis takes a completely different approach to the electrophoresis used in Sanger sequencing. Instead of bacterial cloning, *in vitro* amplification is used to generate copies of each DNA molecule to be sequenced. Instead of measuring fragment lengths, cycles of

biochemical reactions which emit fluorescent signals (one colour per nucleotide A, C, G and T) are used to reveal the sequence of nucleotides of each DNA molecule. The sequence of fluorescent signals at each point on the array reveals the DNA sequence of that DNA molecule. The purpose of the *in vitro* amplification is to amplify the signal produced at each point on the array and reduce errors. Since errors are cumulative as the DNA molecule is sequenced, genomes must still be fragmented as accuracy decreases with the length of the DNA molecule being sequenced. Presently, sequencing-by-synthesis technologies are limited to a maximum length of ~ 300 nucleotides, leading to another name for this approach: short-read-sequencing. In order to sequence long genomes, they are first fragmented, then sequenced, before the genome sequence is assembled using the overlaps between the short sequencing reads or even by comparison to a reference genome sequence [244]. More sequencing technologies have now been developed and whilst Sanger sequencing can be considered first-generation and sequencing-by-synthesis second generation, a third generation are also emerging [268].

The throughput of sequencing-by-synthesis is astounding, however, there are limitations. Fragmentation brings complications when the sequencing reads of fragments are assembled to recover the original DNA sequence: for repetitive regions or pooled DNA samples with near identical sequences it becomes impossible to figure out which sequencing read belongs to which part of the repetitive region or initial sample. Amplification also leads to copying errors, sequence-dependent biases and loss of epigenetic information. Furthermore, both of these processes add time and complexity to the process of sequencing [244]. By contrast, third generation DNA sequencing technologies involves real-time, full-length sequencing without fragmentation or amplification (**Figure 2.5**).

Ideally, DNA molecules would be sequenced accurately and directly, regardless of their length and there are now two technologies striving to do this [244]. Pacific Biosciences (PacBio) technology observes polymerase-mediated synthesis optically in real time. Like sequencing-by-synthesis, a DNA molecule is synthesised using fluorescently labelled nucleotides. The engineered polymerase used in PacBio sequencing is highly processive, with most read lengths between 10,000 and 100,000 nucleotides long. Error rates are around 10% and randomly distributed [244]. This can be reduced by taking approaches that re-read the same molecule several times. Another technology for single-molecule-sequencing is nanopore sequencing, which is used extensively in this thesis.

### 2.2.2 Nanopore sequencing

Nanopore sequencing involves a single-stranded DNA molecule passing through a narrow channel within a nanoscale protein pore (nanopore). The nanopore is embedded in an electrically resistant polymer membrane and a constant voltage is applied across the membrane to produce an ionic current through the nanopore in an electrolytic solution. Negatively charged single-stranded DNA molecules are driven through the nanopore from the negative to the positively charged

side. Since a current of ions also passes through each nanopore, these ions are displaced as the DNA molecules pass through the nanopore. This causes a change in ionic current as the DNA molecule pass through, that is dependent on the nucleotides that are present within the nanopore. By measuring the change in current, the sequence of nucleotides that are passing through the nanopore can be determined [282]. After the experiment, the current signal must be converted to the original sequence of nucleotides in a process known as "basecalling". Basecalling involves identifying (calling) each nucleotide from the measured changes in current. Various computational algorithms are used to do this [4]. It is a computationally demanding process that is best completed using a powerful server.

Nanopores alone cannot sequence DNA and this sequencing technology relies on several other innovations. This begins in the sequencing library preparation, where a sequencing adapter is added to each DNA sequence which facilitates the initial passage of the DNA molecules through the nanopores. Alone, sequences pass through nanopores too fast for the nucleotides to be measured. Therefore a special kind of protein is bound to the adaptor, referred to as a motor protein. The motor protein attaches to the adapter and regulates the speed at which the sequences pass through the nanopore (translocation). Tethering oligonucleotides with affinity for the polymer membrane guide the DNA to the vicinity of the nanopores by binding the adaptors [119]. The DNA molecule is ratcheted through the nanopore in a step-wise manner at a rate of tens to hundreds of nucleotides per second. The DNA molecules are translocated through the nanopore at a constant rate which is determined by the motor protein [282]. To increase the number of sequencing reads, an array of nanopores is utilised, which allows DNA molecules to pass through each nanopore simultaneously.

The development of nanopore sequencing began in the 1980s, when the concept of detecting changes in ion current arising from the four DNA nucleotides was proven [282]. The signal-to-noise ratio was improved by utilising the motor proteins which are processive enzymes that can control the movement of DNA through the nanopore [282]. The motor protein reduced fluctuations in the kinetics of the movement, improving data quality. After three decades of research, nanopore sequencing became a tool that could be utilised by any researcher. Oxford Nanopore Technologies (ONT) released the first commercially available nanopore sequencing device (the MinION) in 2015. Since then, ONT has continued to improve the motor protein and nanopore. The latest nanopores have two sensing regions (referred to as reader heads) which can lead to higher accuracy [282].

ONT has continually improved the average accuracy of sequencing reads, from 65% in 2015 to around 90% in 2021 [282]. Whilst a 10% error-rate may not seem ideal, sequencing each DNA molecule multiple times enables a consensus sequence to be generated, significantly improving the accuracy. There are also "2D" nanopore sequencing methods, which do this by ligating the two strands of the DNA duplex together using a hairpin adapter and sequencing both strands sequentially with the nanopore. "$1D^2$" methods also exist, where each strand is ligated separately

to an adapter and there is a high probability (> 60%) that they will be sequenced consecutively through the same nanopore. 2D and $1D^2$ methods lead to an accuracy of 94% and 95% respectively, however, presently only the "1D" method is supported where an adapter is ligated to each strand of the DNA duplex and they are sequenced independently [282]. Nonetheless, if multiple sequencing reads are collected for each DNA sequence, then a highly-accurate consensus sequence can be generated. Beyond these optimisations in the chemistry used in nanopore sequencing technology, accuracy has been improved by new base-calling algorithms. Given the richness of the information encoded in raw current traces, the accuracy of nanopore sequencing reads will likely continue to improve.

Whilst nanopore sequencing accuracy is low relative to sequencing-by-synthesis and Sanger sequencing, read length is far greater. Reads of up to two million nucleotides have been demonstrated. At this point, the length of the reads depends on the size of the molecules in the sequencing library and various approaches for extracting large DNA molecules have been developed [282]. However, DNA is often still fragmented by shearing using sonication, transposase cleavage or pipette extrusion. For nanopore sequencing, average read lengths have increased from 1,000s to 10,000s of nucleotides [282]. Small fragments can decrease sequencing yields because they have higher adapter ligation and translocation efficiency relative to long fragments. Therefore, methods for selecting DNA fragments based upon size are utilised [282]. Whilst entire genomes are not routinely read on single sequencing reads, nanopore sequencing read length is vastly superior compared to sequencing-by-synthesis, reducing the complexity when assembling a fixed sequence.

The throughput of nanopore sequencing has increased from 1,000s to millions of sequencing reads per experiment. ONT's MinION device is a handheld piece of hardware into which recyclable "flow cells" containing nanopore arrays are inserted; other devices have since been developed. The MinION flow cell contains 512 channels, each containing upto four nanopores (only one of which can be measured at any one time). All nanopores are embedded in a membrane and the change in current at each nanopore can be measured, allowing 512 molecules to be sequenced simultaneously [282]. Since the MinION, various other devices have been released from the Flongle, with 126 channels, to the GridION with four parallel MinION flow cells to the PromethION with upto 48 parallel flow cells and 3000 channels per flow cell [282]. The data produced by a single flow cell depends upon the number of active nanopores, the DNA translocation speed and the time that the device is run for. Current throughput from a MinION flow cell has increased to around 10 billion nucleotides (gigabases, Gb). The PromethION device can yield 100s of Gb.

The combination of long reads, direct sequencing of molecules, improving accuracy and high-throughput that nanopore sequencing enables is inspiring novel studies of natural and synthetic DNA. These possibilities will be explored thoroughly, however, first, two final unique advantages of nanopore sequencing should be explained. The first is that by virtue of sequencing molecules

directly, nanopore sequencing enables the study of epigenetic sequence modifications [282] though this is not the subject of this thesis and will not be discussed in detail. Instead, we focus on the ability to use nanopore sequencing to directly sequence RNA molecules. Due to the instability of RNA, a further step in sequencing library preparation is advocated, though not essential. This step involves reverse-transcription (RT) of a DNA strand, resulting in an RNA-DNA hybrid duplex which has greater stability. The adapter is then ligated to the RNA molecule, which is sequenced directly. The reverse-transcribed DNA molecule is only present for stability and is not sequenced.

Nanopore direct RNA sequencing (dRNA-seq) is lower-throughput and lower accuracy than nanopore DNA sequencing. Accuracy is around ~ 85% and it is increasing with time, as are the number of sequencing reads measured in a single experiment [282]. Around 1–3 Gb (1,000,000 reads) of data is generated per MinION flow cell of a dRNA-seq library. This is due in part to its relatively low sequencing speed (70 nt per second for RNA samples compared to up to 450 nt per second for DNA samples). Nonetheless, whilst RNA can be sequenced using sequencing-by-synthesis, nanopore dRNA-seq offers the capacity to study longer RNA molecules directly.

### 2.2.3 RNA sequencing

In 1965 the first RNA molecule was sequenced after a total of fifteen working years [244]. It was a further forty years until there was a step change in RNA sequencing [251]. Nowadays, a typical *in vivo* RNA sequencing experiment involves RNA extraction from cells, mRNA enrichment or ribosomal RNA (rRNA) depletion (since rRNA comprises the majority of cellular RNA and is often not the subject of a study), DNA synthesis by reverse transcriptase (depending on the sequencing technology) and preparation of an adaptor-ligated sequencing library. The resultant DNA library is then sequenced, resulting in 10 – 30 million sequencing reads for sequencing-by-synthesis [251]. Whilst nanopore dRNA-seq involves directly sequencing the RNA molecules, a reverse-transcription step is often used as the resulting DNA strand, whilst not sequenced, stabilises the RNA molecules. Sequencing adaptors are ligated to the RNA molecules, necessary for the initiation of sequencing. Whilst these adaptors are supplied to ligate to sequences containing adenosine homopolymers, they can be designed to ligate to a particular sequence. Currently, a nanopore dRNA-seq experiment results in around 1 million reads [251].

RNA sequencing has been applied in a variety of ways. Differential gene expression studies have revealed aspects of the transcriptome expressed from various genomes [251]. Other applications have looked into mRNA splicing [246], RNA structure [297], transcriptional dynamics [98], RNA-RNA [161] and RNA-protein interactions [10], the role of non-coding RNAs in gene expression [251] and studies of the RNA in single cells [24]. The majority of these studies have used sequencing-by-synthesis since nanopore dRNA-seq has only recently been developed.

In 2018, the first report of using nanopore dRNA-seq to sequence RNA molecules without fragmentation, RT or amplification was published [83]. This reduces any bias from these steps in

| Data | File format | Source | Processing |
|---|---|---|---|
| Nanopore read current | fast5 | Nanopore sequencer | guppy |
| Read sequences & quality | FASTQ | Guppy | seqtk |
| Read sequences | FASTA | seqtk | seqtk |
| BLASTN alignments | blastn (.out) | BLASTN | awk, pandas |
| Sequence alignment map | SAM | minimap2 | samtools, pysam |
| Read profile | depth (.d) | pysam | pysam |
| Summary statistics | csv | python | matplotlib |
| Genetic design features | gff | python | python |

Figure 2.6: **An overview of common bioinformatic tools and file formats.**

sequencing library preparation. Long reads of 1000 to 50000 nt have been measured, enabling full-length transcripts to be characterised. This simplifies analysis of transcriptomes. Furthermore, like nanopore DNA-seq, epigenetic modifications to RNA can also be detected. However, the relatively low-throughput brings challenges, as does the fact biases during sample and sequencing library preparation are not well understood [251]. Whilst full length RNA molecules can be sequenced in theory, the reality is that transcriptomes characterised by nanopore dRNA-seq routinely show patterns of read depth that have no biological explanation [63, 83, 105]. These patterns are likely method-related, arising from the degradation of RNA samples during sample and library preparation [30]. Despite these challenges, the ability to sequence long RNA molecules directly brings a multitude of new opportunities for studying RNA.

### 2.2.4 Bioinformatic tools for nanopore sequencing data

Sequencing experiments generate huge amounts of data, which must be processed computationally in order to answer research questions. Many bioinformatic tools are available for processing nanopore sequencing data and these have been extensively reviewed [4, 240, 282]. These tools enable the raw sequencing data (nanopore current traces) to be converted to various formats used to probe hypotheses (**Figure 2.6**). Each step can result in artefacts arising from data processing and so care must be taken to understand whether this is occurring during bioinformatic analyses. Nanopore sequencing data is produced in FAST5 format, which stores both metadata and read information. The FAST5 file of multidimensional data is organised in a nested manner which allows access of information in a piece-wise manner without the need for navigating the entire dataset [282]. Typically one FAST5 file contains the information for 4000 sequencing reads and a sequencing experiment generates hundreds to thousands of these files.

The first step in processing nanopore sequencing data is to convert the nanopore read current

to a DNA or RNA sequences [240, 282]. This involves the use of a basecalling algorithm which calls the sequence of bases (nucleotides) from the changes in current measured as the molecule passed through the nanopore. There are a selection basecalling algorithms available, with guppy being the most widely used due to its superior accuracy and speed [287]. The current version of guppy uses neural networks to directly translate the raw current data into a sequence [287].

A FASTQ file is produced after basecalling. The FASTQ format contains details for each sequencing read: reference (name) of the read, direction (5' to 3' or 3' to 5'), sequence and quality score. This quality score details the accuracy of each basecall, that is, the accuracy of each nucleotide in each read. A FASTA file is similar but contains only the sequencing read, with no indication of the read quality. Both FASTQ and FASTA files can be processed quickly with seqtk. Seqtk allows the information for subsets of sequencing reads to be pulled from these files, using the read references alone. Furthermore, tools for dealing with the large datasets are useful. The datasets are often organised as tables, making data table processing tools such as the GNU bash tool awk and the pandas library in Python useful.

Sequencing reads often need to be located within a reference genome in order to generate a consensus sequence. The basic local alignment search tool (BLAST) enables alignment of reads to a reference genome [258], resulting in a file containing all possible alignments. Each alignment has an alignment score and the location of the aligned region in the reference genome. BLAST was developed in the era of Sanger sequencing and was therefore designed for small numbers of highly accurate long reads; aligners are beginning to be developed for aligning error-prone nanopore sequencing reads [282]. For example, minimap2, which creates a sequence alignment map (SAM) file showing where sequencing reads map to genomes [156].

Following alignment of sequencing reads, further processing is often necessary to analyse sequence or function. Various tools exist to do this, for example the SAMtools suite (which includes the python library pySAM) [157] facilitates custom analyses of sequence alignment files, including generation of "depth" profiles which show the number of sequencing reads aligned to each position in the reference genome. After dRNA-seq, transcriptional read profiles can be generated that show how different regions of the sequence encode function. Statistics summarising the function encoded by particular sequences can be generated and visualised using Python with graph-plotting libraries such as matplotlib. The road from raw sequencing data to answering a research question often involves multiple steps, which requires the development and use of diverse and often bespoke information processing tools (**Figure 2.6**).

## 2.3 Characterising genetic parts

### 2.3.1 Types of genetic part

Genetic parts are the tools used for the majority of genetic engineering and synthetic biology [15, 277]. By combining many genetic parts, a genetic design can be created capable of modifying

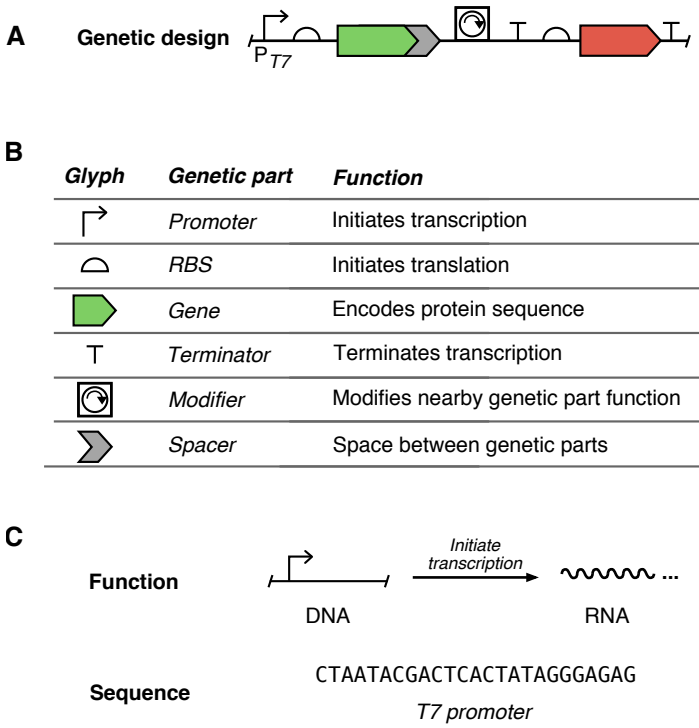



Figure 2.7: **Genetic designs and genetic parts.** (**A**) A genetic design. (**B**) Types of genetic part. (**C**) The function and sequence of a genetic part.

cellular behaviour (**Figure 2.7 A**). Gene expression involves two steps: transcription of DNA into RNA and translation of RNA into protein (**Figure 2.3 A**). Genetic parts used to control transcription include promoters and terminators, which initiate and terminate the process of transcription, respectively (**Figure 2.7 B**). The RBS initiates translation of the RNA transcript, leading to production of the protein sequence encoded in the gene.

Genetic parts are encoded in nucleic acid sequences. For instance, the T7 promoter (from the T7 phage genome) is a sequence that is approximately 25 nucleotides long. This sequence initiates transcription by T7 RNA polymerase by allowing the polymerase to bind to its sequence (**Figure 2.7 C**). Changing the sequence of the genetic part is likely to alter its function. Since a vast number of variants are possible, even for short sequences, synthetic biologists often characterise large libraries of variants of genetic parts [180]. This is useful for generating a diversity of genetic parts that can be used to avoid homologous recombination, which can occur if genetic parts are repeatedly used in a genetic design (typically contiguous sequences of 25 nt or more should be avoided [81]). Whilst it is convenient to define genetic parts that give rise to specific functions, it isn't always clear cut as nearby genetic parts and sequences influence one another [37]. Therefore, it is also important to characterise combinations of genetic parts to

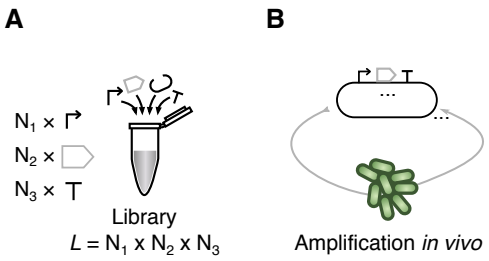



Figure 2.8: **Combinatorial DNA assembly.** (**A**) A DNA library of size $L$ is made by assembling variants of three types of genetic part into a plasmid backbone. (**B**) Amplification of the DNA library in cells *in vivo*.

understand how their sequences and functions interact.

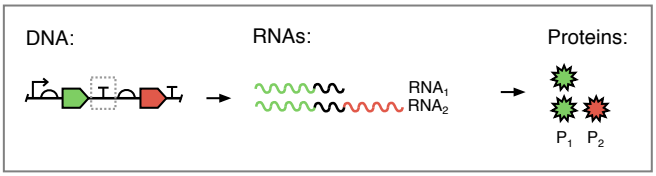
### 2.3.2 Creating genetic parts

In order to characterise genetic designs, they must be defined and made. One approach is to identify and characterise genetic parts from genomes [125]. Short oligonucleotides of DNA encoding genetic parts can then be created either by copying from a genome using PCR or direct chemical synthesis [7]. Libraries of genetic designs can then be built by assembling these oligonucleotides using one of the variety of methods that are available for assembling pieces of DNA [41, 67], or by making slight variations of a particular design using a mutagenesis approach [124]. The resultant DNA library is often amplified *in vivo* prior to characterisation in order to increase the amount of DNA sample (**Figure 2.8 B**). Synthetic biology often requires large DNA libraries which test diverse genetic designs. In order to assemble such libraries from few pieces of DNA, combinatorial DNA assembly can be used (**Figure 2.8 A**) [124, 197]. This method involves designing and making sets of genetic part variants that, when combined, will join together to create many genetic designs, each with a unique combination of genetic parts [144, 187]. Combinatorial DNA assembly lends itself to optimisation of genetic designs without prior knowledge of optimal genetic parts [197]. Once a DNA library has been created, genetic designs with the desired activity can be identified by selecting those with the desired function, allowing optimisation without prior knowledge. However this does not reveal the activity of all the genetic designs in the library. Measuring the function of genetic designs in a DNA library using sequencing can generate a large dataset, which can be used to predict function from sequence.

### 2.3.3 Characterising genetic parts

A major bottleneck when developing new genetic parts is the time and effort needed to characterize libraries of parts to understand their design principles and build predictive models of their function. Various methods have been used to characterise genetic parts, each with their own merits and limitations (**Figure 2.9**) Historically, electrophoresis gels were used to study the transcriptional function of genetic parts [123, 175, 238]. RNA is prepared either from an *in vitro* or *in vivo* experiment and the proportions of RNA are quantified by the intensity of the associated band on the gel. More recently, chip-based capillary electrophoresis [179] has been used. However, this method is time and resource consuming, since each genetic part must be measured in an individual experiment.

A




B

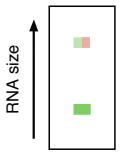
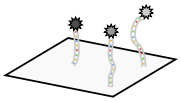
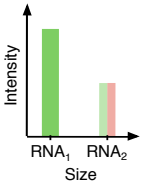
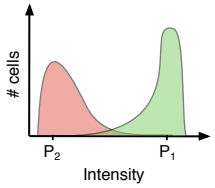
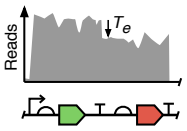
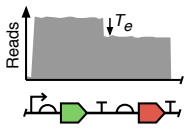
| *Capability* | Electrophoresis Gel | Flow cytometry | Sequencing-by-synthesis | Direct RNA sequencing |
|---|---|---|---|---|
| *Technology* |  |  |  |  |
| *Readout* |  |  |  |  |
| *$T_e$ calculation* | $T_e = \frac{RNA_1 - RNA_2}{RNA_1}$ | $T_e = 1 - \frac{P_2 / P_1}{R_2 / R_1}$ | $T_e = \frac{RNA_1 - RNA_2}{RNA_1}$ | $T_e = \frac{RNA_1 - RNA_2}{RNA_1}$ |
| *Multiplexable* | No | No | Yes | Yes |
| *nt resolution* | No | No | Sometimes | Yes |
| *Long designs* | Yes | Yes | No | Yes |
| *In vitro* | Yes | No | Yes | Yes |
| *In vivo* | Yes | Yes | Yes | Yes |

Figure 2.9: **A comparison of approaches to characterising terminators.** (**A**) A genetic design encoding a terminator surrounded by two genes results in the production of two different RNA transcripts, which produce different fluorescent proteins. (**B**) Comparing experimental approaches to terminator characterisation.

Flow cytometry is an *in vivo* approach that can be used to measure genetic part function. It involves rapid measurement of multiple parameters of single cells. The fluorescence and size or shape of the cells can be studied and cells can even be sorted based upon these parameters using fluorescence activated cell sorting (FACS). The function of the genetic part is coupled to a change in fluorescence signal. Flow cytometry results in an overview of the distribution of genetic part function across a cell population. Samples cannot be be multiplexed and genetic parts have to be tested separately, limiting throughput to hundreds of genetic parts in a single run.

Sequencing methods can be used to characterise both the sequence (genotype) and the function (phenotype) of genetic parts (**Figure 2.7 C**). DNA sequencing is suitable for characterising the genetic part sequence. Depending upon the function of the genetic part, either DNA or RNA

sequencing could be used as a means of measurement [163]. Transcriptional genetic parts are characterised using RNA sequencing. Sequencing can enable "multiplexed" characterisation of entire pooled libraries of genetic designs when each sequencing read contains a unique barcode sequence (**Figure 2.10**). The pool of sequencing reads that are generated in a sequencing experiment can then be demultiplexed using these unique barcodes. The barcode sequence is used to identify the genetic design that a sequencing read belongs to, enabling many genetic designs to be sequenced at the same time, in parallel [48]. By using this approach, sequencing studies can be used to study how sequence encodes function [85]. In order to understand the function of each genetic part sequence, both the barcode and the function of interest must be encoded in the DNA or RNA sequence, respectively. After sequencing, bioinformatic methods are used to separate the sequencing reads based upon the barcodes (**Figure 2.10**) and further analyses are completed to calculate the function of each genetic part. In this way, sequencing reads can be used to map genetic construct sequence to function (genotype to phenotype) [180]. This reduces the cost, time and resources required to sequence each genetic design in the library.

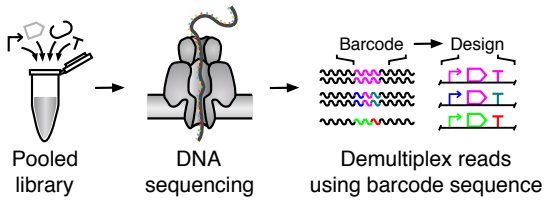



Figure 2.10: **Multiplexed sequencing of DNA libraries.**

Genetic design libraries used in synthetic biology are often incompatible with multiplexed sequencing-by-synthesis characterisation for several reasons. Firstly, every genetic design in the library has the same plasmid backbone. After DNA fragmentation in sequencing-by-synthesis library preparation, many fragmented DNA molecules contain only the plasmid backbone sequence. After sequencing, these fragments cannot be matched to their genetic design since they do not contain the barcode sequence. The sequencing-by-synthesis experiment could be targeted (using a primer) to a site in the design which bears the barcode sequence during amplification. However this would rule out characterisation of the entire genetic design, or, characterisation of long combinations of genetic parts, since only the few hundred nucleotides adjacent to the primer would be characterised. Amplification steps are often biased [52, 189, 199] and result in loss of epigenetic information. In nanopore sequencing, demultiplexing can either be completed using basecalled sequences [258] or raw nanopore current traces [288], which include information about epigenetic markers. Secondly, the genetic designs that synthetic biologists use are often thousands of nucleotides in length [198] making nanopore sequencing methods appropriate [282] since they are capable of collecting long sequencing reads. In contrast, sequencing-by-synthesis is

restricted to read lengths of 300 nucleotides or less (**Figure 2.5**). Therefore nanopore sequencing is well suited to the types of DNA libraries constructed in synthetic biology since each sequencing read should encode an entire genetic design sequence.

It is important to distinguish this multiplex sequencing approach from multiplexing in sequencing-by-synthesis. In sequencing-by-synthesis, multiplexing involves sequencing many fragmented sequencing reads from the same sample simultaneously on an array. In contrast, multiplexed analysis of genetic designs involves pooling the genetic designs in a single sample, which is run in a single sequencing experiment without fragmentation before demultiplexing the sequencing reads (**Figure 2.10**). Multiplexed sequencing methods have been used to characterise many types of genetic parts. Promoters [145], ribosome binding sites [32] and terminators [116] have all been characterised with targeted sequencing-by-synthesis. Nanopore sequencing is beginning to be used to characterise combinations of genetic parts [65]. Using the long reads to characterise combinations of genetic parts can reveal how genetic parts affect one another [37].

## 2.4 Transcriptional termination

Nanopore sequencing can be used study RNA molecules directly, enabling studies of genetic parts for transcription, such as terminators. The process of transcription regulates the production of RNA from DNA (**Figure 2.3**) [15]. Transcription begins with the recruigtment of RNA polymerase (RNAP) at a promoter (transcription initiation), resulting in the formation of an elongation complex (EC) (**Figure 2.11**) which proceeds to make an RNA copy of the DNA template (transcription elongation). The elongation complex comprises RNAP, DNA template and nascent RNA. In order to form, around ~ 10 nucleotides (nt) unwind in the DNA and in this region, the two strands of DNA are not base-paired. Instead, the template DNA strand is base-paired with the nascent RNA molecule, forming a DNA-RNA hybrid (**Figure 2.11**) [209]. When the EC reaches a terminator and termination occurs, the EC dissociates from the DNA template and the nascent RNA [192].

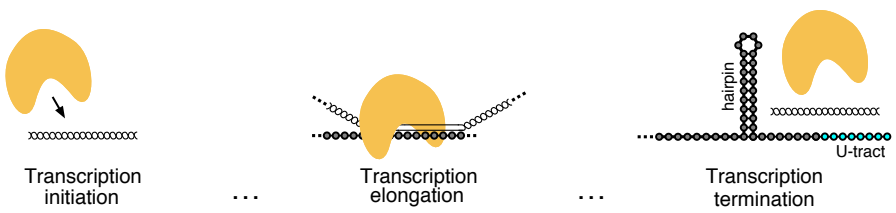



Figure 2.11: **Key steps in transcription.**

Terminators are genetic parts (**Figure 2.7**) that regulate a key process in transcription

(**Figure 2.11**). However terminators do not guarantee termination of transcription [149]. Instead, when a transcribing RNAP reaches a terminator, it can either terminate transcription or read through. The termination efficiency of a terminator can be calculated by comparing the frequency of termination events to the frequency of read-through events. Terminators give rise to multiple RNA molecules, referred to as transcript isoforms. Transcript isoforms are commonly used by eukaryotes to diversify the RNA products produced by a single gene through the subsequent processing of a transcript by splicing machinery [246]. Although such machinery is not generally present in prokaryotes, there is growing realization that these organisms also generate transcript isoforms by utilizing incomplete transcriptional termination [57, 149, 254]. In this case, two transcript isoforms are possible: one ending at the terminator (if termination succeeds), and the other reading through (if termination fails). Given that each transcript isoform will eventually be translated into a protein molecule, terminators can influence the proteins that a cell makes. Whilst promoters and ribosomes have regularly been investigated using multiplexed sequencing experiments [32, 125, 145], terminators have not and present an opportunity for increasing the tools that synthetic biologists can use to engineer biology.

In bacteria there are two types of sequence-encoded terminators: intrinsic terminators and Rho-dependent terminators. Rho-dependent terminators require the RNA helicase Rho to terminate [209]. Rho is a ring protein and facilitates termination by binding to the nascent RNA transcript and threading RNA 5' to 3' through the centre of the ring using the power of ATP to translocate. Once the RNA has passed through the ring, Rho dissociates RNAP from RNA and template DNA [209]. Rho binding sites are ~ 80 nucleotides in length with a high cytosine content and little secondary structure. They are referred to as Rho utilisation sites (*rut*), lie in RNA upstream of the point of termination and have few common features, making their prediction challenging [209]. Transcription can also be terminated by Mfd, an ATP-dependent DNA translocase, that can bind simultaneously to DNA and a stalled RNAP, removing it from DNA and simultaneously recruit DNA repair machinery [226]. In contrast to Mfd- and Rho-dependent terminators, intrinsic terminators can cause the elongation complex to dissociate without any auxiliary molecules, they also have common sequence features [209] and their mechanism of termination is relatively well understood [192]. Therefore, intrinsic terminators present a better opportunity for engineering genetic parts and are focused upon here on in.

### 2.4.1 Mechanism of intrinsic termination

Studies of intrinsic termination have revealed that it is a kinetic process guided by the sequence and structure of the RNA molecule [192, 209]. Generally, a particular sequence is required to induce a pause in the elongation complex and a particular RNA structure is required to induce physical dissociation of the EC [192] (**Figure 2.11**).

In terms of sequence, a series of consecutive uridine residues (referred to as the U-tract) must be transcribed to cause the RNAP to pause, which halts addition of nucleotides to the

growing RNA molecule [192]. The pause is thought to occur via a reduction in the stability of the DNA-RNA hybrid [108] since A:U base-pairs have only two hydrogen bonds whereas G:C have three hydrogen bonds. Studies of other transcriptional pauses associated with intrinsic terminators suggest that the sequences of the upstream RNA, the nucleotides in the elongation complex and the downstream DNA also affect the duration of the pause [192]. For example, the importance of an elemental pause sequence (EPS) has been proposed. The EPS is a G-C base-pair at the site where the RNA-DNA hybrid is attempting to unwind (approximately 10 nt upstream of the point of termination) and it disfavours hybrid unwinding, leading to a pause. This was revealed by single-molecule and sequencing analysis of RNAP elongation [226]. In any case, the U-tract is a found in intrinsic terminators across bacteria [190] and plays an important role in termination [47].

In terms of structure, a G-C rich "hairpin" structure must form just upstream of the U-tract [192]. Being G-C rich, the hairpin structure often contains the EPS at the base of the downstream hairpin stem. Whilst studies have shown that the hairpin is not as predictive of termination efficiency as the U-tract, it certainly has some influence [33, 47].

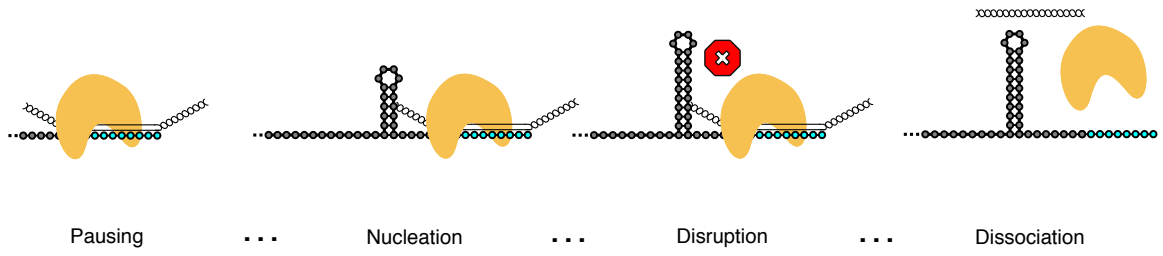



Figure 2.12: **Mechanism of intrinsic termination.**

The mechanism of intrinsic termination proceeds in four stages (**Figure 2.12**) [192]. Firstly, the elongating RNAP pauses upon reaching a U-tract. This pause allows the hairpin structure to begin to form (nucleation). The structure continues to form until it disrupts the RNAP, leading to dissociation of the RNA, DNA and RNAP. This disruption is the greatest energy barrier and is essentially irreversible [192]. However the structural changes that lead to dissociation of the elongation complex and how these vary between intrinsic terminators with different sequences are at best partially understood [192]. Whilst previous studies have characterised sets of hundreds of individual terminators [33, 47], new capabilities in RNA sequencing open possibilities for characterising libraries of terminators [116].

### 2.4.2 Characterising transcriptional terminators

Generally, empirical methods for characterising terminators rely upon characterising either the transcript isoforms or proteins upstream and downstream of the terminator (**Figure 2.13**). Termination efficiency is measured by studying the difference in expression of RNA or protein before and after the terminator. Methods are also available to computationally search for terminator sequences in genomes [190].

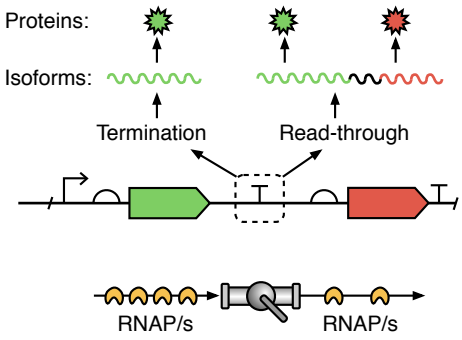



Figure 2.13: **Regulation of transcription by terminators.**

Nowadays a common *in vivo* approach to measuring termination efficiency is a fluorescence assay in which two different fluorescent reporter proteins have the terminator to be tested placed between them (**Figure 2.9**). By comparing the ratio of fluorescence for each reporter with and without the terminator present, it is possible to indirectly quantify the fraction of transcriptional read through and the termination efficiency of the terminator [33, 47, 168]. However, since flow cytometry cannot be multiplexed this approach cannot be scaled up to characterising thousands of genetic designs or long arrays of genetic parts. Furthermore, a challenge with all *in vivo* approaches to characterising terminators is that they often need to make assumptions about the stability of the cellular environment and properties of the transcripts and proteins produced. These may not always hold: variations in mRNA stability [111], the occlusion of adjacent ribosome binding sites due to terminator mRNA secondary structures, the impact of a terminator on translational coupling of neighbouring genes [47], and transcription-translation coupling [168] could all potentially play a role and affect the accuracy of the measurements made. Such differences may explain why *in vitro* and *in vivo* termination efficiencies have not been found to correlate well [66]. Nonetheless, insights into *in vitro* termination [179] have proven useful, enabling identification of novel terminators which have subsequently been used both *in vitro* [239] and *in vivo* [160].

More recently, sequencing-by-synthesis of RNA has been used to characterise terminators in genomes [57] and pools of terminators *in vivo* in high-throughput in a single experiment

(**Figure 2.9**) [116]. This necessitates removal of ribosomal RNA (referred to as rRNA depletion), which can lead to substantial biases in coverage during sequencing library preparation [148]. However this can be overcome with correction of read profiles and enables a more detailed and direct measurement of termination at a nucleotide resolution by allowing for transcriptional profiles capturing RNAP flux along the DNA to be inferred [98]. Drops in RNAP flux within these profiles could be used to measure termination efficiencies of large libraries.

Nanopore direct RNA sequencing enables multiplexed characterisation of libraries of terminators since each sequencing read can be matched to a specific terminator sequence in the library. Furthermore, fragmentation, reverse-transcription and amplification are not required, meaning that the read profiles generated should not be biased. Finally, nanopore sequencing is capable of reading the sequence of entire genetic designs on single sequencing reads (1,000s of nt), which can enable consecutive terminators to be characterised with nucleotide resolution. dRNA-seq has been used to measure terminators in genomes [105] but not libraries of terminators.

### 2.4.3 Engineering intrinsic terminators

Principles for designing and engineering intrinsic terminators have been elucidated by studying terminators in genomes along with closely related sequences [62]. The function of natural terminators can be highly dependent upon context in ways that are not well understood, which has led to efforts to engineer terminator genetic parts with predictable function [192]. A "hybrid engineering" approach has commonly been taken to do this: the essential features of the terminator sequence (such as the U-tract and hairpin) are recombined to create hybrid terminators [62]. However, the breadth of terminators available is much smaller than for other genetic parts such as promoters and relatively few terminators are in common use [62].

In the past decade various studies have begun to expand the variety of terminators available and elucidate principles for engineering them using the availability more high-throughput characterisation methods [33, 47, 116]. In 2013, Chen *et al.* characterised a library of 582 intrinsic terminators *in vivo* using flow cytometry, measuring termination efficiency [47] and identifying 39 "strong" terminators which reduced downstream expression by >50-fold. This set included 227 natural (from *E. coli*) and 265 synthetic terminators which permute important sequence features [47]. Of the natural terminators, 87 had a termination efficiency greater than 90%. Sequence features necessary for termination were studied using the data from the set of natural terminators. The strongest contributor to termination efficiency was the U-tract [47]. The probability of observing a U declined from near unity to the probability expected from a random distribution of nucleotides along the U-tract away from the terminator hairpin [47]. Strong terminators had a weak binding free energy between the U-tract and the template DNA as expected since they facilitate transcription termination through providing weak base-pairing at this region which leads to EC dissociation following RNA hairpin formation.

Correlations of termination efficiency with hairpin characteristics varied: the correlation with

hairpin loop closure was strongest; there was a weak correlation with hairpin folding and no correlation with loop or stem length [47]. The low energy for loop closure observed for strong terminators suggests a kinetic mechanism favouring rapid loop closure, consistent with the observation that tetraloops which increase stability and rate of folding are favoured in genomes [269] and with the observation that termination efficiency is inversely correlated with stem mismatches [47]. Among strong terminators, GC content was only elevated at the base of the stem and not near the loop [47], consistent with a theory that the free energy released from base-pairing in this region has the highest contribution to the ratcheting of the U-tract off the DNA [142, 150]. An A-rich tract was often found upstream of the hairpin in strong terminators [47] and this is thought to be a region required for bidirectional terminators, which can terminate RNAPs in both directions. The A-tract could also extend the hairpin stem and the energy arising from stem extension (which can be used to ratchet the U-tract from the DNA) was found to correlate with termination efficiency [47]. However, when characterised in the opposite direction, few of the moderate strength terminators were terminated equivalently in both directions [47].

Chen *et al.* also designed synthetic terminator libraries to test how each sequence feature contributed to termination efficiency [47]. Two preliminary libraries designed with synthetic DNA that was not derived from a natural source genome did not yield any strong terminators [47]. A third library which systematically varied terminator features using sequences gleaned from strong *E. coli* terminators did, however [47]. Three strong terminators were used as scaffolds and sequence features from other strong terminators were swapped into the scaffolds [47]. For most sequence features, scaffolds reacted differently to changes that were made showing that there was no optimal sequence for each feature and that sequence features depend upon the local sequence context. Synthetic terminators could have a weak termination efficiency even when all of the sequence features originated from strong terminators [47]. Substituting in a U-tract containing only U's improved the strength of 2 of 3 scaffolds but for the third scaffold, it decreased termination efficiency sixfold [47]. With the exception of this U-tract change, the optimal sequence for each feature was that of the native terminator at that position [47]. Changing the A-tract of the scaffolds had the smallest effect; the best hairpin loop and stem were a stable hairpin (GAAA) and the longest 8 base-pair stem respectively [47]. One of the scaffolds was both one of the strongest terminators and the most sensitive to sequence changes and it is this one which is predicted to form a pseudoknot immediately after the U-tract; comparisons revealed that changes to sequence features which disrupt the pseudoknot weaken this terminator [47]. The native terminator for this scaffold is from upstream of the *pheA* gene and is involved in attenuation (premature termination). Pseudoknots are known to cause ribosomal roadblocks which can decrease translational coupling and this could account for the higher than expected termination efficiency that was measured [47].

Chen *et al.* developed a biophysical model of how the sequence of a terminator affects its strength by predicting the probability that an RNAP will dissociate upon reaching the terminator

[47]. A two-step process was assumed where the hairpin nucleates and then the U-tract is ratcheted from the RNAP; termination occurs when these steps occur faster than the RNAP moving through the terminator [47, 209, 278]. A kinetic model was derived with fitted parameters and used to predict termination efficiency for all 582 terminators, resulting in an $R^2$ value of 0.40 [47]. The experimental results and model reveal that termination efficiency prediction is challenging and must factor in the effects of sequence context.

In the same year as Chen *et al.*, Cambray *et al.* used a similar flow cytometry approach to characterise 51 natural and synthetic terminators with more of a focus on understanding sequence context [33]. These terminators encoded termination efficiencies across an ~800-fold range in *E. coli*. Co-transcriptional RNA folding simulations indicated that structures extending beyond the core terminator hairpin are likely to increase termination [33]. The data was used to develop a linear sequence-function model to estimate termination efficiencies ($R^2$ = 0.45) which struggled to accurately predict termination efficiency for terminators where extended RNA structures were formed [33]. The multiple linear regression model related measured termination efficiency to up to 12 sequence features thought to impact termination [33]. Proximal sequence interactions were tested for 11 terminators by constructing minimal terminator motifs encoding only the hairpin and U-tract. 9 of these reduced termination, with read-through increasing by between 2 and 20-fold; in one case termination increased 20-fold due to removal of an attenuating extended RNA structure [33]. A later study investigated how sequence context can affect termination [159], with a focus on transcription-translation coupling. Li *et al.* showed that termination of tR2 in *E. coli* increased as the distance between the gene and the terminator increased [159] and that the terminator could also terminate effectively in the first 100 nt of the gene coding sequence but not in the latter part of the coding region [159]. This reflects the coupling of the ribosome and RNAP and the authors suggest that a translating ribosome can repress transcription termination [159].

Sequencing studies of terminators in genomes [57] have recently inspired a study of a large library of terminators assembled using DNA assembly [116]. This terminator library focused upon studying variation at the base of the core terminator hairpin. DNA assembly using ligation of two sets of semi-randomised DNA oligonucleotides was used to create a library of terminators with a theoretical size of upto 16,384 and sequencing (using sequencing-by-synthesis) characterisation revealed over 10,000 unique terminator sequences varying in termination strength by over 1,000-fold [116]. The hairpin scaffold was a 14 base-pair (bp) fully paired hairpin with a strong GAAA loop and an 8 bp randomised region at the base of the stem followed by two strong (G-C or C-G) base-pairs to close the stem and an 8 nt U-tract and A-tract downstream and upstream respectively [116]. For this library there was a weak correlation between terminator hairpin free energy and termination efficiency and no correlation with GC content. The strongest terminators showed a high frequency of 5'-GG...CC-3' as the base-pairs closing the stem which was noted by Cambray *et al.* too [33]. The next base-pair within the stem was frequently 5'-T...A-3'. This study highlights the capacity of sequencing to characterise much larger libraries of terminators than

was possible with flow cytometry. It also shows that the insights into principles for terminator engineering are restricted by the design of the DNA library.

Intrinsic terminators can be used to engineer transcription in a variety of ways. These can be summarised in four categories: a stop sign to halt transcription, a switch to terminate in response to a stimulus and as a valve to tune transcription of two genes, or arrays of genes (**Figure 2.14**). The design of terminator genetic parts in synthetic biology has often focused on increasing their termination efficiency with the aim of generating stop signs for transcription [33, 47, 66, 116, 179, 192]. Characterising libraries of terminators reveals how strong terminators can be engineered [47] and offers a diversity of sequences needed to avoid genetic design mutation by homologous recombination [47]. The strong terminators can be used in place of double terminators, which are often used to invoke strong termination and insulate transcription of consecutive genetic parts from each other [198] and the host genome [206]. This prevents "leaky" transcription from occurring, whereby RNAP read through terminators into adjacent genetic parts [133]. By insulating transcriptional units from one another, protein stoichiometry could be predicted from promoters and ribosome binding sites alone.

Terminator motifs are commonly used to regulate transcription in more complex ways [6, 250]. The terminator can be used as a switch, either allowing or preventing transcription depending upon the presence of a molecular stimuli. This is often in response to a ligand, with the terminator hairpin either forming or collapsing in the presence of the ligand [6]. This inspired the engineering of small transcription activating RNAs (STARs) [45]. STARs are designed RNA sequences that bind to the nascent RNA sequence encoding the terminator hairpin and prevent it from forming and this allows transcriptional flux to be controlled [6].

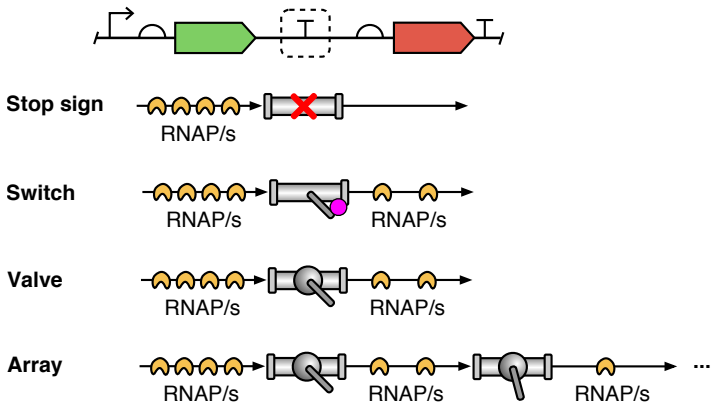



Figure 2.14: **Terminators can be engineered in a variety of ways.**

Recent RNA sequencing studies have revealed that microorganisms use terminators as valves

to control transcription of neighbouring genes [57, 149]. Terminators rarely completely terminate transcription and instead, the terminator can be considered a valve, which regulates the flow of RNAP and therefore the ratio of the transcript isoforms that arise. Engineering transcriptional valves from terminators is unprecedented yet could be used to manipulate transcript isoform ratios. In one case, RNA hairpins (without a U-tract) were used in combination with RNase sites as tunable intergenic regions capable of tuning the transcript isoforms (and therefore protein stoichiometries) arising from consecutive genes [210]. An ability to design transcript isoforms using terminators could open new avenues to control the stoichiometry of multi-gene expression purely at the level of transcription. This regulatory approach could impose less burden [27, 43, 95] than other more commonly used methods, like operons (insulated expression of several genes each with their own promoter and terminator), as not all RNAPs would need to synthesize full length multi-gene transcripts and thus could be freed more quickly for other tasks.

Arrays are commonly used in nature and in biotechnology, for example to regulate production of CRISPR guide RNAs, a terminator is included after each guide RNA [34, 181, 183, 223, 243]. By treating terminators as valves, they could be used to enable differential expression of the components of the array. After each genetic part, another valve would be used to reduce the transcription of the following genetic parts. This could be extended to control ratios of other RNA genetic parts such as small RNA triggers for toehold switches [98], STARs [45]) where translation into protein does not occur. The transcribed RNA parts would have to be excised from the resultant transcript isoforms, using an RNA processing enzyme such as Csy4 [110] or Cas12a [34]. Arrays of RNA parts present opportunities for *in vitro* synthetic biology such as building transcriptional regulatory networks [234] or inexpensive biosensors [127].

## 2.5 Responsible biotechnology research and innovation (RRI)

### 2.5.1 History of RRI

In parallel to the developments in molecular bioscience, the field of science and technology studies (STS) has emerged [182]. STS scholars have shown that innovating technology is not a neutral and apolitical processes but reflects the values, ideologies, and world-views of the society in which they emerge (**Figure 2.15**) [203, 263, 290]. Paths of technological change are enabled by socio-economic conditions, interests and history [203]. Multiple paths of technological change are possible and they often coexist [151]. However over time, one path may become dominant and *naturalised*, creating the illusion that it is the only way of doing things despite being the result of convergent interests, asymmetric power relationships, domination and violence [203]. Knowing that science and technology arises from society, STS scholars began to reflect upon how it can be guided responsibly. Ideas arose that include anticipatory governance, technology assessment and ethical, legal and social aspects of technologies [201]. These foundational ideas led to the field of responsible research and innovation (RRI) [201].

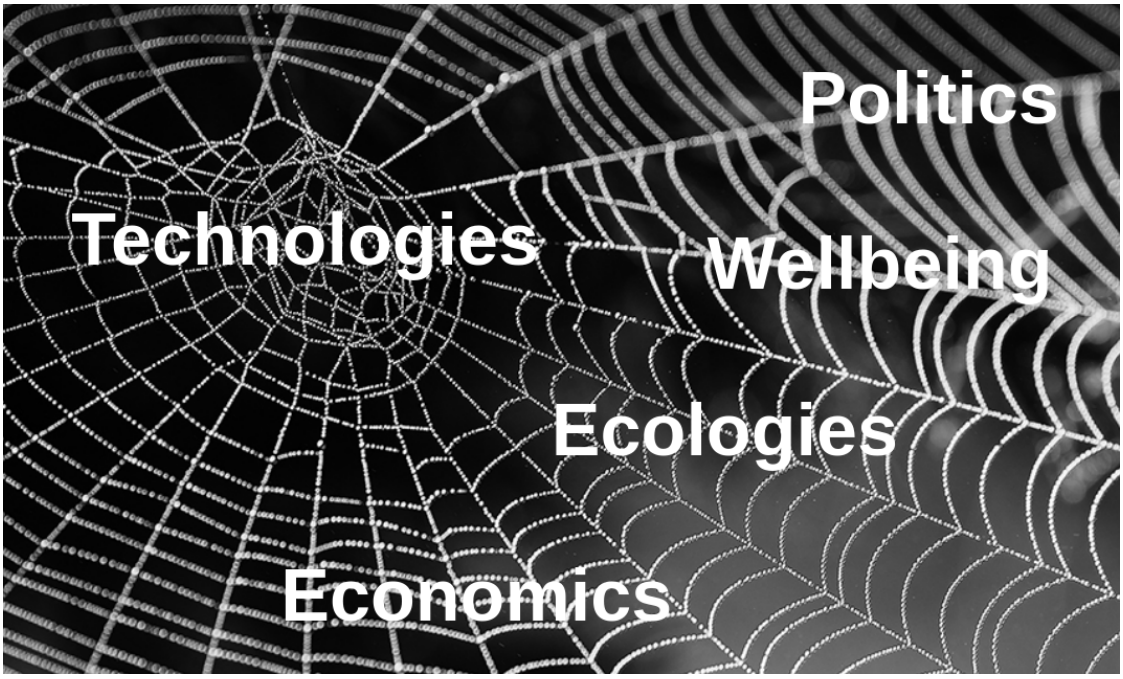
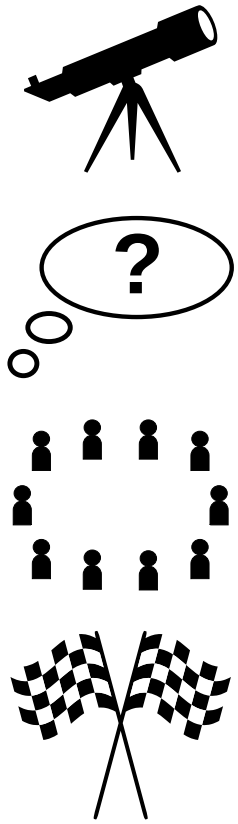



Figure 2.15: **Technologies shape the living world.** Technologies arise from and also influence politics [290], ecologies [225], economies [138], well-being and more.

### 2.5.2 A framework for researching and innovating responsibly

Over time, approaches to research and innovate responsibly were developed and the UK engineering and physical sciences research council (EPSRC) now encourage the use of the AREA framework for responsible research and innovation (**Figure 2.16**). The AREA framework encourages researchers to undertake four key activities throughout the research and innovation process: anticipate (A), reflect (R), engage (E) and act (A) [202]. Anticipation involves describing and analysing the potential impacts of the research, which may or may not be intended. These could be economic, social or environmental impacts. Reflection is a broad activity that encompasses many considerations. The purposes, framing and motivations of the research are considered as well as the implications of the research. Uncertainties, assumptions, areas of ignorance and questions are also considered as well as dilemmas and social transformations that may arise. Engaging involves opening up the visions, impacts and questioning from the anticipatory and reflective activities to broader deliberation, dialogue, engagement and debate in an inclusive way. Finally, acting involves using the anticipate, reflect and engage processes to influence the direction and trajectory of the research and innovation process. Taken together, this can seem overwhelming for researchers, making it important to have more tangible tools they can use to facilitate each activity.



**Anticipate:**

*describing and analysing the impacts, intended or otherwise, (for example economic, social, environmental) that might arise*

**Reflect:**

*reflecting on the purposes of, motivations for and potential implications of the research, and the associated uncertainties, areas of ignorance, assumptions, framings, questions, dilemmas and social transformations these may bring*

**Engage:**

*opening up such visions, impacts and questioning to broader deliberation, dialogue, engagement and debate in an inclusive way*

**Act:**

*using these processes to influence the direction and trajectory of the research and innovation process itself*

Figure 2.16: **The AREA framework for responsible innovation followed by the EPSRC.**

### 2.5.3 Tools for RRI

Common tools for responsible innovation include public engagement activities, facilitated discussions and training. Public engagement and societal outreach are common means of responsible innovation. Historically this has involved a "public understanding of science" approach, where scientists and experts are involved in activities designed to inform, communicate and disseminate information [204]. However this is shifting to a more deliberative approach involving two-way dialogue with publics and stakeholders grounded in the notions of co-creation and co-production of knowledge [204]. This shift reflects an acknowledgement and need for a plurality of epistemological approaches which have been disregarded in the past [61].

Knowledge co-creation is one means of responsible research however it is not always achievable and tools to engage alongside the research process may be preferable. In this case, arts-based approaches involving creative activities enable anticipating and reflecting on research, meanwhile engaging more widely [204]. This can involve creating artwork such as a painting, a play,

song or poem. Still, these practices are time-consuming and may not be suitable for time and resource-pressured researchers [91].

Guiding principles for engineering technologies may offer different ways of approaching technology development too. Ivan Illich outlined a philosophy of "convivial" technology [117] for designing technologies that consider the interdependence between people and between technology and humans [274]. The ideas of Illich have been transformed in to a matrix, the matrix of convivial technology (MCT) [274]. This matrix assesses four life-cycle levels of a technology (materials, production, use, infrastructure) in terms of five dimensions of living well: relatedness, access, adaptability, bio-interaction and appropriateness (**Figure 2.17**). The dimensions of conviviality were selected based upon qualitative research of innovators developing low impact technologies.

The MCT [274] is a tool with potential to facilitate the anticipatory and reflective activities in the framework for responsible research. It is a matrix developed based on the philosophy of convivial technology [117] that can be used to assess the ability to live well with a technology. Whilst it is purely qualitative and subjective, it provides an opportunity for researchers and innovators to reflect upon their research and anticipate the impacts that it may have.

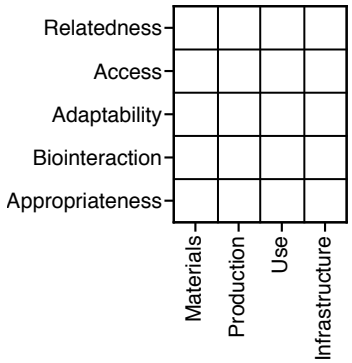



Figure 2.17: **The matrix of convivial technology** The matrix of convivial technology can be used to assess the conviviality of four life-cycle levels of a technology in terms of five dimensions.

The MCT comprises five dimensions of conviviality, assessed for each of the four life-cycle levels of a technology, resulting in a matrix comprising 20 fields (**Figure 2.17**) [274]. Assessing each field requires the inquirer to select the most appropriate descriptors from a list of options, which guides the assessor in answering an overarching question related to each dimension of conviviality. The first dimension of conviviality is relatedness and is summarised by the following question: what does it bring about between people? The next is access: who can produce / use it, where and how? The third is adaptability: how independent and linkable is it? Then there is bio-interaction: how does it interact with living organisms? Finally, appropriateness: what is the relation between input and output considering the context? The four life-cycle levels

are summarised as follows. Materials: the harvesting, processing and disposal of raw matter. Production: assembling raw materials and pre-products. Use: procuring the task it was built for. Infrastructure: needed environment for using. By compiling all twenty fields, the assessor can use the MCT to answer the overarching questions for each life-cycle level of the technology and reflect on their perception of the ability to live well with the technology.

A final tool for reflecting on research is accounting the journey of the researcher. This is commonly shared in humanities theses. This activity involves reflecting upon the context and motivations of the research project, the experiences whilst undertaking the research and how these experiences changed the researcher's perception of the research. Sharing responsible research activities such as the MCT or research journey in theses may lead to wider discussions, as encouraged by the AREA framework for responsible research. In turn, this may lead to action to change the direction and trajectory of the research.

## 2.6 Summary

Engineering biology on planet earth has a long history. Taking a very literal view, one may consider that microorganisms have been engineering biology for billions of years. A similarly wide view of biotechnology as biological technology indicates that like all other organisms, humans have been developing biotechnology since day one. Initially a philosophy of integration and commons economy led them to develop agroecological biotechnologies where there was no line between agriculture and "natural" ecology. For 100,000s of years this was done using foraging and agriculture, selecting plants and animals suited to their lifestyle. If the total time that planet earth has existed for were 24 hours, in the last second some humans began following a human-centric and capitalist philosophy and shortly thereafter developed tools for engineering biology at the nano, sub-cellular level. These tools are DNA sequencing, synthesis and assembly and have resulted from research in the fields of microbiology, molecular biology, genetic engineering and synthetic biology. This has led to the characterisation of genetic parts, for engineering biology at the genetic level.

Technologies for sequencing have improved dramatically over the past fifty years in terms of throughput and length of sequencing reads. The latest generation of sequencing technology involves sequencing single molecules directly without the fragmentation and amplification required by its predecessors. Advantages include long sequencing reads, less bias and no loss of epigenetic information. Whilst accuracy is reduced, it is improving with time as the technology is developed. Nanopore sequencing measures single DNA or RNA molecules as they pass through a nanoscale protein pore (nanopore) and can sequence reads up to millions of nucleotides.

Synthetic biologists are finding sequencing assays useful for multiplexed characterisation of libraries of genetic parts in a single experiment. Combinatorial DNA assembly offers a useful approach for building large libraries that are distinguishable by nanopore sequencing despite

its relatively high error-rate. Nanopore sequencing is appropriate for characterising the long genetic constructs that synthetic biologists use and the can enable study of the interactions between genetic parts. Methods for multiplexed sequencing of sequences and their functions using nanopores are only beginning to be developed and can generate datasets of the function of large libraries of genetic parts, which can be used to predict function from sequence.

The design of novel genetic part DNA sequences is crucial to synthetic biology. Functional sequences of DNA in genomes can offer a source of inspiration, especially DNA sequences encoding regulatory function which are important for controlling protein production. Key processes in protein production are transcription and translation, which are regulated by genetic parts: promoters, terminators and ribosomes. Recent transcriptional studies of organisms have highlighted that the function of terminators is more complex than initially proposed [149]. Terminators rarely completely stop transcription and instead filter the amount of RNA polymerases passing a point, resulting in multiple transcript isoforms. This can be used to regulate the ratios of neighbouring genes using transcript isoforms. Though pigeon-holed as stop signs for some time, terminators are ripe for engineering. Since both their sequence and function is expressed in RNA, RNA sequencing is suitable for high-throughput characterisation of libraries of terminator genetic parts. Generating these libraries (Chapter 4) and characterising their function using nanopore sequencing based methods (Chapters 5 and 6) will be the focus of this thesis, with one further chapter considering how the biotechnology developed here can be further innovated in a responsible manner (Chapter 7).

CHAPTER 3

# MATERIALS AND METHODS

## 3.1 Core molecular biology protocols

### 3.1.1 Cell Strains

All cloning was performed using *Escherichia coli* strains DH10-$\beta$ ($F^-$ endA1 *glnV44 thi-1 recA1 relA1 gyrA96 deoR nupG purB20 $\phi$80dlacZ$\Delta$M15 $\Delta$(lacZYA–argF)U169, hsdR17($r_K^- m_K^+$), $\lambda^-$)* (C3019I, New England Biolabs), or BL21-DE3 (*fhuA2 [lon] ompT gal ($\lambda$ DE3) [dcm] $\Delta$hsdS ($\lambda$ DE3 = $\lambda$ sBamHIo $\Delta$EcoRI-B int::(lacI::PlacUV5::T7 gene1) i21 $\Delta$nin5*) (Thermo Fisher Scientific, OneShot$^{TM}$ BL21(DE3) Chemically Competent E. coli, C600003) where specified.

### 3.1.2 Buffers

TE buffer (10 mM tris(hydroxymethyl)aminomethane (Sigma-Aldrich, 252859), 0.1 mM ethylenediaminetetraacetic acid (Sigma-Aldrich, 20-158), pH 8.0) was used to solvate oligonucleotides. Annealing buffer (10 mM tris(hydroxymethyl)aminomethane (Sigma-Aldrich, 252859), 50 mM sodium chloride (Sigma-Aldrich, 20-158), 1 mM ethylenediaminetetraacetic acid (Sigma-Aldrich, S7653), pH 7.5-8.0) was used to dilute oligonucleotides prior to annealing them.

### 3.1.3 Media

To prepare media, first water was ultra-purified using a water purifier (Merck-Millipore, Milli-Q Integral ultrapure water type1). For lysogeny Broth (LB Miller), LB Broth (Miller) (Sigma-Aldrich, L3522) was added, which contains tryptone (10 g/L), NaCl (10 g/L) and yeast extract (5 g/L). For LB Broth Miller with agar, LB Broth with agar (Miller) (Sigma-Aldrich, L3147) was added which contains agar (15 g/L), tryptone (10 g/L), NaCl (10 g/L) and yeast extract (5 g/L). Solids were

dissolved in the water by vigorous shaking and then the solution was sterilised in an autoclave for 15 minutes at 121°C. After cooling to below 55°C, relevant antibiotics were added to the media solution to enable selection for resistance markers in plasmids. LB Broth (Miller) and LB Broth with agar (Miller) were used as liquid and solid media for culturing cells respectively.

### 3.1.4 Antibiotic stocks

Antibiotic selection was performed using 100 $\mu$g/mL of ampicillin. To make ampicillin stocks, ampicillin sodium salt (Sigma-Aldrich, A9518-25G) was dissolved in ultra-pure water and then filter-sterilised through 0.22 $\mu$M sterile polyethersulfone (PES) syringe filters (Star Lab, E4780-1226). 1000X (100 mg/mL) ampicillin stocks were made and stored at -20°C. For use, stocks were thawed and diluted to working concentration (100 $\mu$g/mL).

### 3.1.5 Glycerol stocks

Glycerol stocks of bacteria strains expressing plasmids were made by gently mixing overnight liquid cell culture with 60% (v/v) glycerol in water, to give a solution of cells with a final concentration of 20% glycerol, which was placed in a -80°C freezer.

### 3.1.6 Conditions for growing cells

Cells grown on agar plates were grown for 14 hours at 37°C. Cells grown in liquid media were grown in culture tubes at 250 RPM and 37°C in an orbital shaking incubator (Stuart, SI500) for 14 hours.

### 3.1.7 Gel electrophoresis and DNA extraction

Gel electrophoresis was used to check the quality of plasmids during DNA assembly, which is described later. A gel was prepared by adding agarose (0.27 g) (Lonza, 982-100) to TAE buffer (30 mL) containing GelGreen Nucleic Acid Gel Stain (Biotium, 41005) diluted to 1X concentration from a 10,000X concentration stock. The mixture was dissolved by microwaving at 800 W for 30 seconds, twice, with 2 seconds of mixing in between. The liquid agarose mixture was poured into a 20mL gel casting tray with an appropriate well comb inserted. The gel was left to set for 30 minutes at room temperature.

DNA samples were prepared by diluting the appropriate mass of each sample with nuclease-free water to give a sample of final volume 5 $\mu$L, containing 10-100 ng of DNA. 1 $\mu$L of 6 X purple gel loading dye without SDS (B7025S, New England Biolabs) was added to each DNA sample. The set gel in its casting tray was moved into the gel box (Bio-Rad, Mini-Sub Cell GT). The well comb was removed from the gel and the gel box was filled with TAE buffer until its level was just above the gel surface. The DNA samples were loaded into the wells and a ladder was added to a separate lane (either New England Biolabs 1 kb DNA Ladder, N3232S, or New England

Biolabs 1 kb Plus DNA Ladder, N0559S). Gels were run at 80 V for 30 to 100 minutes with a constant 80 mA current (Bio-Rad, PowerPac Basic). Gels were visualised and photographed on a gel documentation system (UVP, BioDoc-It). Where gel extractions were performed, the Monarch DNA Gel Extraction Kit (T1020S, New England Biolabs) was used with the standard protocol, eluting in nuclease-free water (6 $\mu$L).

### 3.1.8 Transformation of cells

Unless otherwise stated, the following protocol was used for transforming cells with plasmid DNA. A tube of chemically competent cells was thawed on ice for 10 minutes. Thawed cells (50 $\mu$L) were pipetted into a microfuge tube (size 1.5 mL). Plasmid DNA (1-5 $\mu$L with a mass of 1pg – 100 ng) was pipetted into the thawed cells. Cells were mixed by flicking the tube 5 times. The mixture was incubated on ice for 30 minutes. Then cells were placed in a dry heating block (Thermo Fisher Scientific, 88870004) at 42 °C for 30 seconds (for DH10-$\beta$ cells; 10 seconds for BL21 (DE3) cells) and incubated on ice for 5 minutes. SOC outgrowth medium (450 $\mu$L, B9035S, New England Biolabs) warmed to room temperature was added to the cells. The resulting mixture was incubated at 37 °C and 1250 RPM for 60 minutes in a microtitre plate shaker incubator (Stuart, SI505). Transformed cells (100 $\mu$L) were pipetted onto an agar selection plate and spread over the agar. The plate with cells was incubated overnight at 37 °C.

### 3.1.9 Plasmid stock preparation

To prepare purified plasmid DNA, after overnight growth of cells expressing the plasmid on agar plates, cells were picked from single colonies and used to inoculate 5 mL of LB (Miller) media with the relevant antibiotic. Cells were grown in liquid culture for 14 hours at 250 RPM at 37°C in an orbital shaking incubator (Stuart, SI500). Then cells (4 mL) were centrifuged for 10 minutes at 4000 $g$. The supernatant was discarded and plasmid DNA was extracted from the remaining pellet of cells using the Monarch Plasmid Miniprep kit (T1010S, New England Biolabs) with the standard protocol, eluting in nuclease-free water (30 $\mu$L).

### 3.1.10 Sanger sequencing

Where plasmids needed sequence verification, the Eurofins Genomics service was used. To prepare samples, purified plasmid DNA was diluted with nuclease-free water to prepare a 15 $\mu$L sample in a safe-lock tube (size 1.5 mL) with a concentration between 50 to 100 ng/$\mu$L. The relevant primer to sequence the intended region of the plasmid was added (2 $\mu$L of a 10 $\mu$M primer stock).

### 3.1.11 Measuring DNA and RNA concentration

To measure the concentration of DNA and RNA samples, 1 $\mu$L of the sample was added to the NanoPhotometer N60 (Implen) and the concentration was recorded in ng/$\mu$L. Where specified the

Qubit fluorometer was used to measure concentrations with the Qubit High Sensitivity or Broad Range kit, by following the standard protocol.

## 3.2 Pooled combinatorial assembly of DNA libraries

### 3.2.1 Plasmid mutagenesis

The pGR plasmid backbone [47] (Addgene plasmid 46002) was modified for use in combinatorial assembly by first mutating the *gfp* stop codons from 'TAATAA' to 'TTAGCA' using Q5 mutagenesis (E0554S, New England Biolabs). Primers were designed with 5'-ends annealing back-to-back using the online design software NEBaseChanger$^{TM}$. The forward primer CTATACAAATtagcAGAATTCACTAGTAGCGGCCG (mutated nucleotides in lower case) and reverse primer TTCATCCATGCCATGTGTAATC were used.

The standard protocol was followed with an annealing temperature of 61 °C for 30 seconds, an elongation time of 30 seconds per kilobase and no Kinase, Ligase and DpnI treatment. The sequence of the mutated product was checked by Sanger sequencing (primer GGTCCTTCTTGAGTTTGTAAC) and its length was checked by gel electrophoresis. The resulting plasmid was referred to as pGRm (mutated pGR).

### 3.2.2 Changing the plasmid promoter

The *araC* gene and $P_{BAD}$ promoter of the mutated pGR plasmid was replaced with a consensus T7 promoter sequence (pT7). Promoter substitution used the restriction enzymes AatII (10 units; R0117S, New England Biolabs) and NheI-HF (10 units; R3131S, New England Biolabs) in 1X CutSmart buffer (B7204S, New England Biolabs) and nuclease-free water (final volume 50 $\mu$L) at 37 °C for 30 min; 80 °C for 20 minutes, followed by adding annealed promoter oligonucleotides (forward sequence: CTAATACGACTCACTATAGGGAGAG and reverse sequence: CTAGCTCTCCCTATAGTGAGTCGTATTAGACGT, both provided in 5'-3' orientation) to vector DNA (0.020 pmol) at a 3:1 molar ratio with T4 DNA ligase (400 units; M0202S, New England Biolabs), T4 DNA ligase buffer and nuclease-free water to a final volume of 20 $\mu$L for 30 minutes at room temperature and then 65 °C for 10 minutes. The sequence of the resulting plasmid DNA was checked by Sanger sequencing (primer AAAGGGAATAAGGGCGACACGG) and its length was checked by gel electrophoresis. The resulting plasmid was referred to as pGRmT7 (mutated pGR with T7 promoter) (**Appendix A.3**).

### 3.2.3 Annealing DNA duplexes using oligonucleotide pairs

Oligonucleotides were ordered from Integrated DNA Technologies (25 nmol, dry lyophilized solid, standard desalting). All oligonucleotide tubes were centrifuged for 30 seconds and then diluted to 100 $\mu$M in TE buffer by adding 10 times TE buffer (in $\mu$L) than there were nM of the

particular oligonucleotide (for example adding 300 $\mu$L TE buffer to a tube containing 30 nM of an oligonucleotide).

Oligonucleotides were designed in pairs, with each pair containing a complementary forward and reverse oligonucleotide that could be annealed to form a DNA duplex. Each oligonucleotide was designed such that after annealing, a short overhang of between 1 and 5 nucleotides remained at its 5'-end. This short overhang was later used in the DNA assembly process to ligate DNA duplexes with complementary overhangs.

To anneal the complementary forward and reverse oligonucleotides, an equal amount of each (2 $\mu$L, 100 $\mu$M) was added to 46 $\mu$L annealing buffer. The mixture was heated to 95 °C for 5 minutes and slowly cooled to room temperature over the course of one hour.

### 3.2.4 Pooling DNA duplexes

Annealed DNA duplexes were pooled since there was little risk of of unintentional annealing of non-pairs occurring between DNA duplexes (whereas that could be the case if pooling unannealed oligonucleotides). For each DNA library, care was taken that the duplex DNA was pooled at the correct stoichiometry to ensure sufficient abundance of each duplex DNA to make all combinations. This was achieved by adding the same total volume of variants for each part in the assembly. The pool was then diluted to a final concentration of 1 pmol/$\mu$L with nuclease-free water.

### 3.2.5 Phosphorylation of pooled duplex DNA

The pooled duplex DNA (20 $\mu$L) was phosphorylated using T4 polynucleotide kinase (10 units; M0201S, New England Biolabs) in 10X T4 DNA ligase buffer (2 $\mu$L) at 37 °C for 30 min; 65 °C for 20 minutes. This is a process that phosphorylates the 5'-ends of the DNA duplexes and it is essential for subsequent assembly by ligation of DNA duplexes.

### 3.2.6 Combinatorial assembly of DNA libraries

The engineered plasmid backbone DNA (1 $\mu$g) was digested using EcoRI-HF (20 units; R3101S, New England Biolabs) and SpeI-HF (20 units; R3133S, New England Biolabs) in 10X CutSmart buffer (5 $\mu$L) and nuclease-free water (35 $\mu$L) at 37 °C for 4 hr and then 80 °C for 20 minutes. The digested plasmid was then subjected to gel electrophoresis and extracted from the gel. Digested plasmid backbone (50 fmol) was used for pooled ligation based combinatorial assembly by adding 5-fold excess of the phosphorylated duplex DNA pool (that is, sufficient to assemble 250 fmol of each genetic design), nuclease-free water (40 $\mu$L), 10X T4 DNA ligase buffer (5 $\mu$L) and T4 DNA ligase (320 units; M0202S, New England Biolabs) and incubating at room temperature for 3 hr and then 65 oC for 10 minutes.

All libraries constructed in this work are described in **Appendix A.2**). Whilst a full plasmid is assembled, the combinatorial assembly of DNA duplexes occurs at one particular region.

Following assembly, this assembled region is referred to as the intrinsic barcode of the plasmid since it contains a unique combination of assembled DNA duplexes, which can be used to identify sequencing reads arising from it. The intrinsic barcode also encodes a unique intrinsic transcriptional terminator comprising the core terminator hairpin and upstream sequence (referred to as a "modifier" and a "spacer"), whose function is later measured using *in vitro* transcription followed by direct RNA sequencing. The intrinsic barcode is assembled into the backbone from DNA duplexes encoding the spacer, modifier and core terminator sequence in that order (5'-3') .There are several variants of each of these parts, which are assembled combinatorially into the plasmid to make the DNA library.

## 3.3 Pooled amplification of DNA libraries by transformation

### 3.3.1 DNA library amplification in liquid culture

For amplification in liquid culture, *E. coli* strain DH10-$\beta$ cells were used except for one case where *E. coli* strain BL21 (DE3) cells were used. The cells were transformed with the assembled DNA library (3 $\mu$L, equivalent to 5 ng). Instead of adding the cells to plates, cells (100 $\mu$L) were added to LB media (10 mL) containing ampicillin (1X) in a culture tube and incubated for 14 hours with shaking at 250 RPM at 37 °C. A glycerol stock of each cell culture was taken and the remaining cell culture (9 mL) was centrifuged at 4000 X $g$ for 10 minutes and the DNA was extracted using the standard plasmid miniprep protocol (T1010L, New England Biolabs).

### 3.3.2 DNA library amplification on agar plates

For amplification on agar plates, the assembled DNA library (3 $\mu$L, equivalent to 5 ng) was added to each of 10 aliquots of *E. coli* strain DH10-$\beta$ cells (45 $\mu$L each). The transformation protocol was followed for each aliquot. Each aliquot was then added to one 1.5 L rectangular glass tray (Pyrex) containing LB agar with ampicillin (1X) and cells were grown overnight at 37 °C. Following this, colonies were scraped from each agar plate.

The colonies from each plate were scraped into a culture tube. DNA was extracted by following the plasmid miniprep protocol (T1010L, New England Biolabs) with the following modifications. 4-fold of each miniprep reagent was used for the cells scraped from each plate, with thorough mixing and addition to a single miniprep column. At the end of the protocol, the DNA was eluted in nuclease-free water (30 $\mu$L). The DNA from each plate was combined to make the final amplified DNA library sample.

## 3.4 Verification of assembled DNA libraries using nanopore DNA sequencing

### 3.4.1 Nanopore DNA sequencing

DNA from amplified DNA libraries (50 to 400 ng per sample) was prepared for DNA sequencing using the rapid barcoding kit following the standard protocol (SQK-RBK004, Oxford Nanopore Technologies). DNA samples were sequenced for 48 hr on FLO-MIN106 flow cells. Generated FAST5 files were basecalled using guppy version 3.1.5 [287] with default settings and the configuration file dna_r9.4.1_450bps_fast.cfg, resulting in FASTQ files. High accuracy mode was not used as it was too time consuming for the high numbers of sequencing reads generated using nanopore DNA sequencing.

### 3.4.2 Demultiplexing nanopore DNA sequencing reads

Following basecalling, the FASTQ files containing sequencing reads were combined into a single file using the cat command in GNU bash. BLASTN requires the sequencing data in FASTA format in a BLASTN database. Therefore, the FASTQ file was converted to FASTA format and the resulting file was used to make a BLASTN database of reference sequences. BLASTN version 2.2.31 [258] was used to align sequencing reads to reference sequences using BLASTN parameters selected based upon simulated nanopore sequencing data: -outfmt 6 -gapopen 5 -gapextend 2 -reward 2 -penalty -3 -evalue 1 -word size 4 -max target seqs 1000000 -max hsps 1. Reads were aligned to the intrinsic barcode sequence rather than to the whole plasmid sequence. Since the designed DNA libraries were large, the GNU bash command parallel was used to run multiple queries of the sequencing reads database at the same time.

Custom Python and GNU bash scripts were then used for demultiplexing, the process of matching each sequencing read to an intrinsic barcode based upon the best alignment (the alignment with maximum bitscore) (**Appendix A.3**). Sequencing reads with no alignment to a intrinsic barcode, or alignments to multiple intrinsic barcodes with the same maximum bitscore were excluded from further analysis. Part and intrinsic barcode frequencies were calculated relative to the total number of sequencing reads assigned to intrinsic barcodes.

### 3.4.3 Generating consensus sequences

We used demultiplexed sequencing reads to generate a consensus sequence for each intrinsic barcode and assess DNA assembly fidelity. Demultiplexed sequencing reads for each intrinsic barcode were aligned to the plasmid encoding the intrinsic barcode using minimap2 version 2.17 [156] with the command map-ont and arguments: -ax. Racon version 1.4 [270] with parameters: -m 8 -x -6 -g -8 -w 500, was used to polish the plasmid sequence and refine the consensus sequence produced. The polished and reference sequences were then aligned using Multiple Alignment

using Fast Fourier Transform (MAFFT) with parameters: –localpair –maxiterate 1000 [131]. Finally, a custom Python script was used to count the average number of single nucleotide polymorphisms (SNPs) per intrinsic barcode (**Appendix A.5**).

## 3.5 Pooled in vitro transcription and direct RNA sequencing

### 3.5.1 Plasmid linearisation

For *in vitro* transcription, it is important that the DNA template is linear, to prevent continuous transcription around the circular plasmid, which would result in the generation of long heterogeneous RNA transcripts because of the high processivity of T7 RNA polymerase. Therefore DNA from the pooled library (1 $\mu$g) was linearised using AatII (10 units) in 10X CutSmart buffer (5 $\mu$L) and nuclease-free water (40 $\mu$L) at 37 °C for 30 minutes and then 80 °C for 20 minutes. The linearised product was purified using the Monarch PCR Cleanup kit (T1030L, New England Biolabs) and eluted in nuclease-free water (12 $\mu$L).

### 3.5.2 *In vitro* transcription

In vitro transcription was completed using the HiScribe T7 High Yield RNA Synthesis Kit (E2040S, New England Biolabs). The following reagents were combined: T7 RNA Polymerase mix (2 $\mu$L), adenosine triphosphate (2 $\mu$L), guanosine triphosphate (2 $\mu$L), cytidine triphosphate (2 $\mu$L), uridine triphosphate (2 $\mu$L), kit reaction buffer (2 $\mu$L) and the linearised DNA pool (250 ng). To this mixture, we added the RNA calibration strand (0.5 $\mu$L, from SQK-RNA002, Oxford Nanopore Technologies) as a control RNA that would not be transcribed but would show any RNA degradation arising from experimental conditions. This mixture was incubated at 37 °C for 35 minutes. Synthesized RNA was diluted 20-fold in nuclease-free water and purified using a Zymo Clean and Concentrate kit (R1013, Zymo Research). The concentration was measured using the Qubit Broad Range kit.

### 3.5.3 Polyadenylation of transcripts

The purified RNA was then poly-adenylated. Care was taken to ensure that the reagents were fresh and had not been freeze-thawed more than once as their quality decreases significantly with use. *E. coli* Poly(A) Polymerase (10 units; M0276S, New England Biolabs) was added to the purified RNA (10 $\mu$g), with 10X reaction buffer (2 $\mu$L), RNase inhibitor murine (0.5 $\mu$L; M0314S, New England Biolabs) and 2 $\mu$L adenosine triphosphate, ATP (10 mM) at 37 °C for 30 minutes. The reaction was stopped by proceeding to RNA purification with elution in 15 $\mu$L nuclease-free water (R1013, Zymo Research). The concentration was measured using the Qubit Broad Range kit.

### 3.5.4 Nanopore direct RNA sequencing

RNA sequencing libraries were prepared from the polyadenylated RNA (1 $\mu$g) using the direct RNA sequencing kit following the standard protocol (SQK-RNA002, Oxford Nanopore Technologies) with the flow cell priming kit EXP-FLP002 (Oxford Nanopore Technologies). This involves a reverse-transcription step. The reverse-transcribed strand is not sequenced, instead it is made to increase the stability of the RNA strands during the sequencing experiment. After preparing the RNA sequencing library, its concentration was measured using the Qubit High Sensitivity kit and normally amounted to around 200 ng in total. RNA sequencing libraries were sequenced for 48 hours on FLO-MIN106 flow cells. This produced data in FAST5 format, which was basecalled using guppy version 3.1.5 in high-accuracy mode [287] with default settings and the configuration file rna_r9.4.1_70bps_hac.

## 3.6 Computational demultiplexing and termination analysis pipeline

### 3.6.1 Demultiplexing nanopore RNA sequencing reads

The same demultiplexing protocol was followed as for nanopore DNA sequencing reads except that after basecalling, sequencing data from two *in vitro* transcription reactions was pooled. The sequencing reads were pooled since RNA sequencing results in fewer sequencing reads than DNA sequencing due to RNA degradation. The higher the number of sequencing reads per design, the higher the accuracy of the measurement of termination efficiency.

### 3.6.2 Read profile generation

Demultiplexed reads for each intrinsic barcode were collated into a single file using the function subseq from seqtk (available at https://github.com/lh3/seqtk), this process was also parallelised using GNU parallel. Collated reads were then mapped to a plasmid sequence encoding the appropriate intrinsic barcode using minimap2 version 2.17 [156] to generate a sequence alignment (SAM) file.

RNA degradation caused some RNAs to be truncated within the intrinsic barcode, meaning that RNA sequencing reads were sometimes incorrectly demultiplexed, warranting correction. This occurred when insufficient intrinsic barcode was present in the sequencing read to match it to a particular reference sequence. Therefore we used pySAM [157] to remove RNA sequencing reads which had been demultiplexed yet did not contain a full intrinsic barcode sequence. To do this, the SAM file was refined by removing sequencing reads that did not contain a full intrinsic barcode sequence and ended between the start and 20 nt into the final assembled part in the intrinsic barcode as defined by a general feature format (GFF) file that had been made for each plasmid sequence, which specified the location of the assembled parts within the intrinsic barcode.

### 3.6.3 Calculating termination efficiencies

Termination efficiency ($T_e$) values are calculated from the read depth either side of the valve.

$$T_e = R(x_s)–R(x_e)]/R(x_s) \qquad (1)$$

Where $R(x)$ is the read depth at position x in the genetic design, and $x_s$ and $x_e$ are the start and end nucleotide position of the intrinsic barcode, respectively. $T_e$ values were further corrected using a model which accounted for RNA truncation (**Section 5.4**).

## 3.7 Designing arrays of gRNAs regulated by transcriptional valves

Valve designs used in the arrays consisted of particular sets of spacer, modifier and core-terminator sequences. The CRISPR guide RNA (gRNA) sequences were selected from the CRISPRlator construct designed by Santos-Moreno *et al.* [233]. Since the array would result in multiple guides per transcript, the RNA transcripts would have to be processed following transcription to separate them and make them functional. The same strategy used by Santos-Moreno *et al.*was used: inclusion of Csy4 recognition sites around each gRNA.

Due to limitations on the number of repetitive sequences that can be effectively synthesised, measures were taken to reduce sequence homology within each array. Csy4 recognition sites were shortened to 15 nt, which includes all but the 3'-cytidine of the shortest functional recognition site to be characterized [110]. The 3'-cytidine was omitted as it falls outside of the hairpin and the Csy4 recognition site used by Santos-Moreno *et al.* omitted it yet it was still recognized by the enzyme.

Instead of using the same CRISPR handle for each gRNA we selected a unique handle for each gRNA from the non-repetitive examples in Reis *et al.* [223]. Three handles with dissimilar sequences and high functionality were used. These handles are compatible with dCas9sp. While gRNAs are often transcribed with a terminator sequence, this was omitted as valves were used to regulate transcription instead.

At the start and end of each array was a short 15–25 nt randomly generated sequence to allow for PCR amplification and serve as a buffer for restriction enzyme cleavage. Finally, around each gRNA-handle and each valve we included unique single-cutter restriction sites to facilitate modification of the arrays. The complete array sequences were ordered as double-stranded gBlocks$^{TM}$ gene fragments (Integrated DNA Technologies). Four arrays were created: pVGA010, pVGA011, pVGA012, pVGA013 (**Appendix A.3**).

## 3.8 In vitro transcription and dRNA-seq of arrays

The entire sequence of each array was ordered, with no need for DNA assembly. DNA arrived freeze-dried in tubes and was solubilized as follows. The tubes were solvated in nuclease-free

water (8 $\mu$L), briefly vortexed, incubated at 50 °C for 15 minutes, cleaned up (Monarch DNA cleanup kit, T1030L, New England Biolabs) and eluted in nuclease-free water (10 $\mu$L). After measuring the concentration by nanodrop, all four arrays were pooled (62.5 ng of each) along with the RNA calibration strand (0.5 $\mu$L) and transcribed using the HiScribe$^{TM}$ T7 In Vitro Transcription Kit protocol and reagents (E2030, New England Biolabs). The mixture was mixed thoroughly by tapping and incubated at 37 °C for 35 minutes.

10 $\mu$L of reaction product was dilute 5-fold with nuclease-free water and purified using the Zymo Clean and Concentrate 25 kit (RCC-25, Zymo Research), eluting in nuclease-free water (25 $\mu$L). Following RNA quantification (Nanodrop), 10 $\mu$g of this RNA was diluted to a total volume of 13.5 $\mu$L with nuclease-free water and polyadenylated using *E. coli* Poly(A) Polymerase (10 units), with RNase inhibitor murine (0.5 $\mu$L), 10X *E. coli* Poly(A) Polymerase Reaction Buffer (2 $\mu$L) and ATP (10 mM, 2 $\mu$L). The reaction was incubated at 37 °C for 35 minutes and stopped by purification with the Zymo Clean and Concentrate 25 kit. Following nanodrop quantification, polyadenylated RNA transcripts (1 $\mu$g) were prepared for sequencing using the nanopore kit SQK-RNA002, flow cell priming kit EXP-FLP002 (Oxford Nanopore Technologies) and sequenced in duplicate on FLO-MIN106D (Oxford Nanopore Technologies) flow cells for 48 hours.

## 3.9 Computational tools and genetic design visualization

Computational analyses were executed using Python version 3.5, Ubuntu 16.04.7 LTS (xenial) and GNU bash version 4.3.48(1)-release (x86_64-pc-linux-gnu), example scripts are included (**Appendix A.5**). All genetic diagrams are shown using Synthetic Biology Open Language Visual (SBOL Visual) notation [14]. SBOL Visual diagrams were generated using the DNAplotlib Python package version 1.0 [64] which were then annotated and composed with Inkscape or OmniGraffle version 7.9.2.

## 3.10 Generating non-structural RNA sequences

All regions to provide padding in modifiers were designed using RNAInverse [302] which can generate RNA sequences with a specific RNA structure. Structures are specified with brackets representing base-paired nucleotides and dots representing unpaired nucleotides. RNAInverse would not accept a query structure with no base-pairing so a query structure that the tool would accept was submitted: (.....) and then edited. The output sequences were trimmed by one nucleotide at both ends and any sequences with a non-zero folding energy, a restriction enzyme recognition site for EcoRI, SpeI or AatII, or a site with 4 or more adjacent identical nucleotides were removed. The remaining sequences were used in the design of modifier sequences.

## 3.11 Co-transcriptional folding simulations

The RNA-seq read profiles of genetic designs elucidated the where the nascent RNA sequences were commonly terminated. To investigate the RNA secondary structures at these points, we simulated co-transcriptional folding [86] of a relevant portion of the terminated sequence. We removed the final 8 nucleotides of the terminated sequence, which have been shown to base-pair with the DNA template in the T7 RNAP transcription elongation complex [255]. We assumed a previously reported transcription rate of 333 nt/s [291] for T7 RNA polymerase. The co-transcriptional folding simulation gives an ensemble of possible RNA structures for the query sequence and we studied only the structures with the lowest folding energy.

## 3.12 Library coverage calculation

For the DNA library amplified on agar plates, we estimated library coverage using the approach presented by Patrick *et al.* [207] to calculate the expected number of distinct sequences in a library chosen at random from a set of sequence variants. Given a pooled library containing $L$ sequences, and a set of $V$ equiprobable variants, let $v_i$ be one of the possible variants. Since the variants are equiprobable, the mean number of occurrences of $v_i$ in $L$ is

$$\lambda = L/V \qquad (2)$$

For $\lambda << L$ (i.e., $V >> 1$), the actual number of occurrences of $v_i$ in $L$ is essentially independent of the number of occurrences of any other variant $v_j$ where $j \neq i$, and therefore well-approximated by a Poisson distribution

$$P(x) = \frac{e^{-\lambda}\lambda^x}{x!} \qquad (3)$$

where $P(x)$ gives the probability that $v_i$ occurs exactly $x$ times in the library. The probability that $v_i$ occurs at least once is given by $1 - P(0) = 1 - e^{-\lambda} = 1 - e^{-L/V}$. Therefore, the number of distinct variants expected in the library is given by

$$C \approx V(1 - e^{-L/V}) \qquad (4)$$

and the fractional completeness of the library is

$$F = \frac{C}{V} \approx 1 - e^{-L/V} \qquad (5)$$

The library size required for fractional completeness F is therefore

$$L \approx -V \ln(1 - F) \qquad (6)$$

For the library L3 studied in this thesis, V = 1183 variants and a fractional completeness of $F > 1 - \frac{1}{1183} = 0.99915$ was required to ensure with high probability the representation of all variants in the library. This necessitates a library size of at least $L \approx -V \ln(1 - 0.99915) = 8364$. To achieve this, we performed a transformation protocol that used 10 large Pyrex glass trays with approximately 50,000 transformants per tray, resulting in $L \approx 500000$.

## 3.13 Compiling the matrix of convivial technology

The matrix of convivial technology consists of a matrix of twenty fields which allow evaluation of the five dimensions of conviviality (relatedness, access, adaptability, bio-interaction and appropriateness) for each of the four life-cycle level of a technology (materials, production, use, infrastructure) [274]. Each field has several metrics which the assessor uses to evaluate the technology. Each metric is a choice between a non-convivial descriptor and a convivial descriptor.

The matrix of convivial technology was adapted for visualisation as follows. For each field, the choice of descriptors were used to calculate a value which approximates whether the technology is considered convivial, non-convivial, or both. To do this, each technology was assigned a value for each choice of descriptors according to which descriptor it was considered to be. If the technology was assessed to be at the convivial end of the spectrum, this metric was assigned a value of +1; if it was assessed to be at the non-convivial end, this metric was assigned a value of -1. If it was assessed to be both, the metric was assigned a value of 0, or if not considered to be assessable, assigned a value of "NA".

Once all choice of descriptors had been assigned a value, the modal value was calculated for each of the 20 fields. This was assumed to approximate the conviviality of the technology for this field as perceived by the assessor. The values for each field was used to colour the matrix of convivial technology as follows: red for -1 values, orange for values of 0 and green for values of +1. Since we assessed the uses arising from different approaches to innovation, only the "use" life-cycle level was assessed. The assessment of the conviviality of the use was evaluated for each approach to innovation. In all studies, once the matrix was compiled, the Python library matplotlib was used to visualise it. Fields are coloured by perceived conviviality, as measured by the assessor (red: non-convivial, green: convivial, yellow: both). In this study, the assessor was the innovator.

CHAPTER 4

# CHARACTERISING COMBINATORIAL GENETIC PART LIBRARIES

## 4.1 Introduction

Understanding how DNA sequences encode function offers a basis for understanding and even writing the code of life. Vast numbers of DNA sequence variants are possible and studying all of these is often impossible: creating all DNA sequence variants up to length 79 nucleotides would require a mass greater than the earth [168]. Genetic sequences encoding a specific function vary from 10s to millions of bases. Nonetheless, since the discovery of DNA, attempts have been made to understand the language of genetics [72]. A major goal in the field of synthetic biology is understanding how to engineer genetic parts. To this end, synthetic biologists frequently assemble large numbers of DNA sequences (**Figure 4.1 A**). In the past, methods limited genetic engineers to making small "libraries" of DNA sequences constructed and characterised individually [47]. Multiplexed methods are now available to assemble much larger DNA libraries and characterise the function of all sequences in a single experiment (**Figure 4.1 C**), providing datasets to enable prediction of function from sequence [33] (**Figure 4.1 D**).

One low cost method to assemble DNA libraries involves *in vitro* ligation of synthesised oligonucleotides into longer pieces of DNA (**Figure 4.1 A**). This is one of three main approaches to *in vitro* combinatorial DNA assembly: ligation-based, PCR-based and restriction enzyme-based [50]. Ligation can be used to assemble oligonucleotides combinatorially, enabling multiple genetic parts to be assembled into a genetic design. The result is different combinations of functional genetic parts (e.g. protein coding genes or regulatory elements like promoters, ribosome binding sites and terminators) [85]. This is important for studying how genetic parts behave in different genetic contexts. The function of a genetic part often depends upon the genetic context, for example nearby sequences [37]. By comparing the functions of similar genetic designs, an understanding of what affects the function of genetic parts can be reached. Combinatorial DNA assembly allows

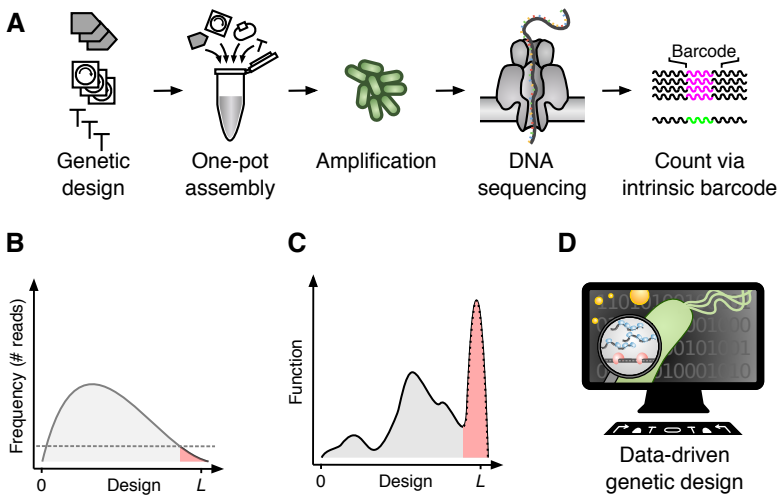



Figure 4.1: **Combinatorial DNA assembly and multiplexed sequencing enables genetic design** (**A**) A DNA library is combinatorially assembled from genetic part variants, amplified; after nanopore DNA sequencing, sequencing reads are demultiplexed using barcodes. (**B**) An example DNA library composition (with $L$ designs). A number of designs fall below the threshold frequency (red shading) (**C**). The function of low frequency designs cannot be characterised (red shading). (**D**) Sequence-function mapping enables predictive design of genetic parts

for inexpensive generation of diverse DNA libraries necessary to attain an overview of the variety of functions encoded across the vast possibilities of sequence space.

DNA assembly is often followed by some form of amplification to create many copies of each design in a library. This is because after DNA assembly there is only a small amount of the DNA due to the small quantities of oligonucleotides used. By inserting the genetic parts into a plasmid backbone, replication and expression of the plasmid and the parts it contains becomes possible *in vivo*. The assembled DNA library can be amplified *in vivo* in a host microorganism or *in vitro* using PCR. Normally when DNA libraries are amplified *in vivo*, they are not expressed. That is, their genes are not transcribed or translated into RNA or proteins; the purpose is only to replicate (amplify) the DNA. The different genetic designs in the library may elicit different growth rates, which could bias the composition of genetic designs in the library. Amplifying the library DNA without expression can avoid such biases, however, in order to study the function of libraries of genetic parts *in vivo* the genetic designs must be expressed. This may cause changes to the library composition and even mutations to genetic designs, both of which can complicate characterisation of the constituent genetic designs.

Despite the capability to create large DNA libraries, the sequences produced are rarely characterised. Mutations arising during DNA assembly and amplification, result in sequences differing from the intended genetic designs. This can lead to an incorrect mapping of sequence to function. In the era of Sanger-sequencing, which cannot be performed using pooled ("multiplexed")

samples, instead of full library characterisation a few assembled DNA constructs are typically characterised and statistical analysis used to estimate genetic design and mutational frequency across a library [78]. This approaches gives only a snapshot of the DNA library composition, which is problematic for combinatorially assembled libraries, which are not assembled uniformly and can result in a much lower frequency of some genetic designs in the library (**Figure 4.1 B**) [166]. Whilst DNA libraries are frequently used to measure the function of each design using multiplexed DNA or RNA sequencing, the composition of the library is seldom measured [145]. Non-uniform library composition can make measurement of function challenging and lead to incomplete library characterisation (**Figure 4.1 C**).

Recently, there have been some efforts to characterise libraries of genetic designs. Pooled sequencing of DNA libraries facilitated complete characterisation of the sequences of 96 plasmids [82]. A subset of 24 constructed using restriction enzyme-based DNA assembly by ligation contained 15 with indels or SNPs [82]. Thus, DNA sequencing (DNA-seq) can reveal mutations to members of DNA libraries that may affect their function. Genetic part libraries present unique challenges for characterisation by sequencing-by-synthesis. Firstly, only regions containing certain genetic parts are varied, meaning that there are regions within the constituent genetic designs that are homologous to all other genetic designs in the library. This means that it is impossible to match (demultiplex) short sequencing reads to their original plasmid as they could belong to any member of the DNA library. The short reads produced by sequencing-by-synthesis are up to 300 nucleotides whilst genetic designs can be 1000s of nucleotides long, meaning that the short reads have to be assembled by finding those with overlapping sequences to reconstruct the original plasmid sequence. For similar reasons, sequencing-by-synthesis is not suitable for assembling plasmid sequences which contain repetitive regions.

Sequencing methods such as nanopore sequencing are now capable of characterising a whole genetic design sequence with a single sequencing read, overcoming these limitations. If each design contains a unique sequence (a barcode), assigning reads to designs (demultiplexing) is straightforward and read assembly is not necessary (**Figure 4.1 A**). From a single nanopore sequencing run, millions of sequencing reads can be collected, each representing a different genetic design. This makes the approach appropriate for characterising large DNA libraries. Nanopore sequencing is beginning to be used to thoroughly characterise individual [51] and combinatorially assembled DNA libraries [166]. This presents a unique opportunity to investigate how DNA assembly and amplification methods affect DNA library composition.

In this chapter we demonstrate the utility of nanopore DNA sequencing for characterising the composition and sequences of entire DNA libraries. The DNA libraries are assembled using a simple combinatorial assembly by ligation and amplified *in vivo* with and without expression of RNA and proteins (Section 4.2). In Section 4.3 we develop a computational analysis pipeline for assigning nanopore sequencing reads to designs and approaches to design sequence barcodes. Then, in Section 4.4 we use sequencing based characterisation to study the composition of

DNA assemblies. In Section 4.5 we explore insights nanopore sequencing can offer in to the mutations of the assembled DNA libraries. Our findings show that library composition changes significantly when the encoded RNA and proteins are expressed during *in vivo* amplification: design frequencies change and design sequences mutate. Finally, in Section 4.6 we discuss the advantages and limitations of using nanopore DNA sequencing to characterise libraries of genetic parts. This demonstrates how nanopore sequencing can be used to guide the engineering of libraries of genetic parts.

## 4.2 DNA library assembly, design and amplification

### 4.2.1 Assembly of combinatorial DNA libraries by ligation

We devised a simple method to assemble a library of genetic designs containing variants of genetic parts at low cost (**Figure 4.2**). Variants are generated by insertion of short sequences which comprise genetic parts (**Figure 4.2 A**) in to a plasmid backbone (**Figure 4.2 B**). The plasmid backbone is generated by cutting out a region from the plasmid using restriction enzymes. DNA ligase is then used to ligate short DNA duplexes made from oligonucleotides. The oligonucleotides are designed, synthesised and annealed prior to ligation into the plasmid backbone (**Methods**) [50]. We used the DNA assembly process to incorporate three adjacent DNA duplexes into the plasmid backbone simultaneously in a specific order (**Figure 4.2 B**). Rather than designing a single DNA duplex to fill the entire excised gap and re-ligate the plasmid backbone, we designed a sequence of three genetic parts, each with unique 4-nucleotide overhangs at their ends (**Figure 4.2 A**). These ends were chosen to minimise the chance of incorrect assembly [213]. Overhangs of adjacent parts were complementary meaning that part order was specified and also that different part variants could be designed (with the same overhangs) to insert at a particular position. This DNA assembly by ligation successfully constructed libraries of genetic designs with variation of several genetic parts.

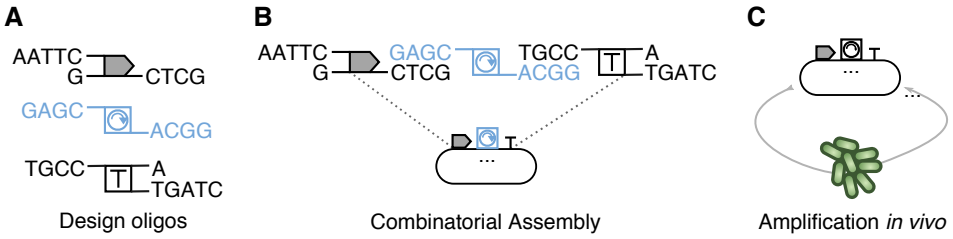



Figure 4.2: **Assembly of a combinatorial DNA library** (**A**) Design of oligonucleotides ( "oligos") for combinatorial assembly of a DNA library. Annealed pairs of oligos are shown with overhangs required for ligation. The spacer, modifier (blue) and terminator genetic parts are shown (top to bottom). (**B**) Combinatorial assembly of the genetic parts occurs by ligation of complementary overhangs. Every combination of spacer, modifier and terminator is created in a single reaction.

This combinatorial DNA assembly method could construct large libraries inexpensively that would enable the influence of nearby sequence context upon genetic parts to be studied. Libraries were designed such that they could be made from DNA sequences that could be inexpensively synthesised as single strand DNA oligonucleotide. This limits the genetic parts to a unique sequence of length 13 to 60 nucleotides (including an overhang of length 4 nucleotides at each end). Complementary forward and reverse oligonucleotides were designed with overhangs at the 5'-end. The oligonucleotides pairs were annealed to create DNA duplexes. The DNA duplexes were pooled in the correct stoichiometry and phosphorylated. This design enabled the variants of each of the genetic parts to be used as the building blocks in the DNA assembly process and combinatorially ligated to form the genetic designs.

### 4.2.2 DNA library design

We designed a DNA assembly that would integrate combinations of genetic parts into the pGR plasmid that would give rise to genetic designs with some degree of transcriptional termination. The pGR plasmid consists of a promoter followed by two genes with regulatory elements (ribosome binding site and terminator before and after the gene respectively) [47]. Two restriction enzymes were used to cut at two points, excising a region between the first and the second gene. The genetic parts were ligated into the gap that had been created between the two genes (**Figure 4.2 B**). The genetic parts were inserted at this position since it would allow termination to be measured by fluorescence cytometry as well RNA sequencing, if required. We designed several libraries for DNA assembly. The DNA libraries were designed such that each assembled plasmid would contain a transcriptional terminator and two upstream genetic parts encoding genetic sequence context. Various terminators were selected from a previous study [47]. A negative control terminator was also selected, containing a random non-coding sequence generated by R2oDNA designer [40], further verified to not contain a hairpin in the mRNA secondary structure

using the Vienna RNAfold tool [104].

The genetic sequence immediately upstream of a terminator-hairpin also influences termination [33, 159] and we reasoned that this region could be used to fine-tune termination efficiency. We therefore included "modifier" parts in our library design. Modifiers were designed containing motifs designed to interact with canonical regions of a terminator hairpin sequence such as possible U- and A-tracts within a terminator using complementary homopolymers of adenine or uracil, respectively [47]. A further modifier was designed to encode a small RNA secondary structure with the goal of affecting RNA structure formation near the terminator part ("S").

It has been shown that inert random sequences can insulate genetic parts, provided they are long enough [39, 159]. Insulating terminators from upstream genetic context could improve the robustness of a genetic part's performance when used in different genetic contexts. Thus, we decided to include a selection of modifiers of different lengths where each was a random non-coding sequence generated by R2oDNA designer [40]. Our first library design contained 18 modifiers based upon three sub-sequences (randomly generated) incorporating different motifs or lengths of a random sequence (**Figure 4.3 B**). A second library design contained 13 modifiers, none of which were based upon the same sub-sequence for reasons that we will come to explain.

To assess the robustness of the termination efficiency of each combination of modifier and core terminator to local upstream genetic context, our library also included "spacer" elements. These allowed us to see how termination efficiency varies when used in combination with other components (e.g. coding regions). Using the NullSeq tool [165], we generated random and genetically diverse 33 bp long spacers with a nucleotide composition similar to coding regions of E. coli that could be placed at the 5' end of the modifier. 5 spacers were used in L1 and 7 spacers were used in L2. Taken together, our spacers, modifiers and core terminators could be combinatorially assembled to create large libraries of unique designs able to regulate transcription and provide valuable information regarding the design of transcriptional terminators. Two libraries were assembled: L1, consisting of 540 designs and L2, consisting of 1183 designs (**Figure 4.3**).

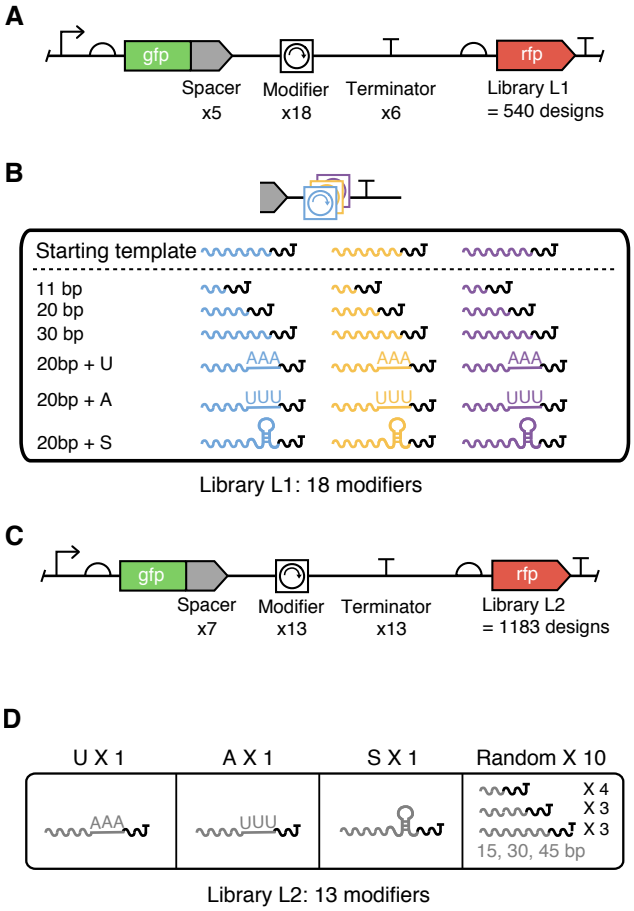



Figure 4.3: **Combinatorial DNA library designs.** (**A**) The initial library (L1) of intrinsic barcodes used to optimise sequence demultiplexing consisted of 5 spacers, 18 modifiers and 6 terminators, resulting in 540 unique designs. For part sequences see Appendix. (**B**) Modifiers for L1 were based upon 3 random starting template sequences, represented by different coloured subsequences. From each template sequence 6 variants were made, each containing different proportions of the template sequence indicated by the number of base pairs: 11 bp sub-sequence, 20 bp sub-sequence, full 30 bp sequence, and several 20 bp sub-sequences with different motifs ( “A”, “U” and “S”). (**C**) The final library (L2) was designed to assemble 1183 designs from 7 spacers, 13 modifiers and 13 terminators. (**D**) Modifiers for the final library were designed such that they did not share any sub-sequences.

Libraries were designed such that each assembled design would fulfil two roles: a barcode (**Figure 4.1 A**) and a genetic part whose function could be measured (**Figure 4.1 C**). Barcodes are essential for studying DNA libraries as they enable DNA sequencing reads to be matched to genetic designs (a process referred to as demultiplexing) (**Figure 4.1 A**). For our library designs, barcodes were part of the plasmid sequence and encoded a specific function, therefore we refer to them as intrinsic barcodes. Intrinsic barcode sequences must be designed to ensure that

sequencing reads can be demultiplexed. L1 (**Figure 4.3 B**) included some quite closely related genetic parts (as little as one nucleotide difference) which meant that the resulting sequencing reads could not be demultiplexed. For this reason, we made the second library of 1183 designs containing genetic parts with dissimilar sequences (**Figure 4.3 D**). By characterising each library with nanopore DNA sequencing in turn, we elucidated design principles which enable intrinsic barcode sequences to be demultiplexed after nanopore DNA sequencing (Section 4.3). Libraries must contain intrinsic barcodes sufficiently dissimilar from one another to be distinguishable despite the 5-15% sequencing error-rate of nanopore sequencing and must be small enough such that they can be characterised in a single minion flow cell sequencing run. We now describe different methods used to amplify the DNA library L2.

### 4.2.3 DNA amplification

After *in vitro* combinatorial assembly, DNA libraries are commonly amplified *in vivo* (**Figure 4.4 A**). This involves transformation of cells, spreading of these cells on to large petri dishes and overnight growth. The bacterial colonies that grow are then scraped. This combines the bacteria containing the amplified plasmids in to a single pool, from which plasmid DNA can be extracted. We could find no published explanation of the necessity of this scraping protocol. An anecdotal explanation was that individual clones could grow without competition. Since this protocol is important for pooled sequencing experiments, we decided to investigate whether a less time and resource consuming approach could be taken. We reasoned that DNA libraries could also be amplified in liquid culture if there was no selection between variants. Owing to the similar length and nucleotide content of the designs in our library, there was likely to be little selection during replication of the DNA (without expression of RNA or proteins), therefore we amplified the library in liquid culture.

To test the function of genetic designs, they must be expressed in a host organism. This may lead to selection of the designs with the least burden and even evolution of designs to reduce burden via mutation. To observe the effect of expression on library composition we amplified the DNA assembly using four conditions (**Figure 4.4 B**). P0: no amplification, P1: amplification of the DNA library in liquid culture (N = 2 replicates), P2: amplification of the DNA library in liquid culture meanwhile expressing the RNA and proteins that it encodes (N = 2 replicates) and P3: amplification of the DNA library on agar plates with the scraping protocol. For protocols P1, P2 and P3, cells were grown for 14 hours at 37°C. A cell strain (DH10-$\beta$) that would not express the RNA or proteins of the genetic designs was used for P1 and P3. The DNA library could not be expressed in this strain since it lacked the appropriate RNA polymerase for the T7 promoter in the plasmid backbone. For P2 we amplified the DNA library in a strain (BL21 DE3) which contained T7 RNA polymerase and therefore would express the RNA and proteins of the DNA library. BL21 DE3 cells express T7 RNA polymerase without induction, so no inducer was added. For P3, DH10-$\beta$ cells were transformed with the pooled library, grown overnight and




| | P0 | P1 | P2 | P3 |
|---|---|---|---|---|
| DNA | - | + | + | + |
| RNA | - | - | + | - |
| protein | - | - | + | - |
| growth | - | liquid | liquid | solid |
| N | 1 | 2 | 2 | 1 |

Figure 4.4: **Amplification of a combinatorial DNA library** Details of DNA library amplification protocols: molecules expressed during amplification (DNA, RNA, protein), growth medium and number of repeats (N).

approximately 500,000 colonies (>400-fold library coverage) selected from plates via scraping before their pooled DNA was extracted. Such a high fold-coverage ensured representation of each design in the sample (**Methods**) [207]. Next, we describe the optimisation of the experimental and computational methods for studying the DNA libraries.

## 4.3 Optimising nanopore characterisation of genetic part libraries

### 4.3.1 Nanopore DNA sequencing for multiplexed DNA library characterisation

At the outset of this thesis, nanopore sequencing was not in routine use for studying combinatorially assembled DNA libraries. There are many protocols for preparing DNA sequencing libraries for nanopore sequencing; the simplest and fastest protocol is the rapid barcoding kit (SQK-RBK004) (**Figure 4.5 A**). 400 ng plasmid DNA is recommended, which can be made up of up to 12 samples, which are sequenced simultaneously. This equates to 40 ng per sample, equivalent to 16 fmol, approximately 1 billion plasmid molecules. A total of 1-5 million sequencing reads are acquired, which equates to reading approximately 0.01 % of the available molecules. Our experiments showed that the rapid barcoding kit could successfully sequence plasmid DNA libraries. We characterised each library with at least 100,000 sequencing reads; approximately 100-fold more sequencing reads than designs.

The library preparation time for the rapid barcoding kit is approximately 10 minutes and involves a key molecular step: “tagmentation” of plasmid DNA (**Figure 4.1**). Tagmentation involves cleaving and adding a barcode to the circular plasmid DNA molecules. At the same time as cleaving the plasmids, the transposome complex adds barcoded transposase adapters. A transposome complex with a different barcoded (the barcode is a unique genetic sequence) transposase adapter is added to each sample. This enables multiple samples to be pooled after

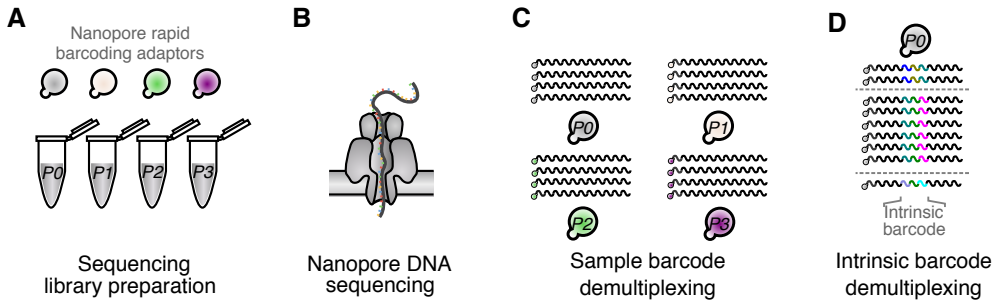



Figure 4.5: **Characterising multiple DNA libraries with nanopore DNA sequencing** (**A**) A different rapid barcoding adaptor is added to each DNA library. (**B**) DNA sequences are measured as the pooled, barcoded, libraries pass through the nanopores. (**C**) After basecalling, the sequencing reads are demultiplexed using the library barcodes. (**D**) A second demultiplexing step uses the intrinsic barcodes (coloured) to count sequencing reads representing each design in each library.

this initial barcoding step and run in the same sequencing experiment. Note that this barcode is different to the intrinsic barcode in our assembled genetic designs; it is used to identify which sample the sequencing read belongs to. This sample barcode enabled us to characterise multiple preparations of the same DNA assembly (**Figure 4.2 B**) in a single DNA sequencing experiment. Sequencing reads for each sample are demultiplexed using the adapter barcode sequences (**Figure 4.5 C**). Since each sample contains an entire library of genetic designs, a subsequent demultiplexing step is required to match genetic designs to their sequences (**Figure 4.5 D**) and we optimise a method to do this in the next section.

### 4.3.2 Developing a computational pipeline for demultiplexing high-error sequencing reads using intrinsic barcodes

To optimise a computational pipeline (**Figure 4.6**) for demultiplexing and ensure that our library design could be accurately characterised, we simulated the error-ridden reads that would be obtained from nanopore DNA-seq. In our simulations, errors took the form of random nucleotide substitutions that occurred at a 15% substitution frequency. While other types of error found in nanopore sequencing reads such as insertions, deletions and elevated error rates at homopolymers were not included in our model, we found that our simulations were able to identify key parameters and criteria for designing parts with a sufficient dissimilarity for effective demultiplexing.

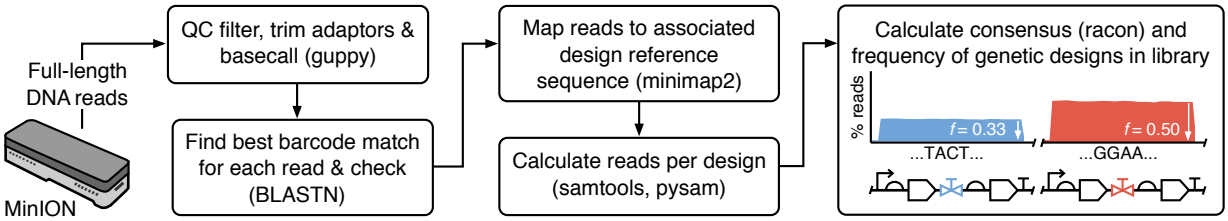



Figure 4.6: **Bioinformatic pipeline used to demultiplex DNA sequencing reads** Key computational tools shown in parentheses

To demultiplex the DNA-seq reads, the BLAST-nucleotide (BLASTN tool) was used to find all possible alignments between a read and the library of designs, with the best matching design being chosen [2, 158]. Due to the 5-15% per-base error-rate of nanopore sequencing, BLASTN was selected as a means of identifying barcodes that were robust to errors within the reads; trials using minimap2 alone to demultiplex reads recurrently failed. Whilst nanopore does characterise long reads, not all of them are the full length of the plasmid. We designed the computational pipeline to accommodate direct RNA sequencing (used in chapters 5 and 6) where approximately 30% of reads are fragmented and may not contain the barcode sequence. Therefore, BLASTN alignments were completed using the barcodes rather than the full length plasmid sequence, to prevent fragmented sequencing reads which don't contain a design from being allocated non-uniformly to designs, which would lead to read profiles that may not be representative of the actual design sequence.

Optimising the BLASTN parameters is crucial for accurate characterisation and so computational analyses were performed where a designed library (540 designs, **Figure 4.3 B**) was used to systematically explore the role of each BLASTN parameter. Each design had a set of simulated reads generated. These reads were then pooled for all the designs and attempts made to demultiplex reads to each design. This allowed us to generate an optimised set of parameters that allows each design to be accurately identified (**Methods**).

The important BLASTN parameters for demultiplexing sequencing reads highlighted were: word_size, which was reduced from the default (11) to the minimum (4) and the reward for a nucleotide match was raised from the default (1) to 2. In one case, aligning to the sequence containing one of the terminator sequences (T6) with the 4 nucleotide overhang at the 5' end and 7 nucleotides at the 3' end, changing these two parameters reduced the false negative rate from 96% to 5%. The BLASTN command used the parameters: -outfmt 6 -gapopen 5 -gapextend 2 -reward 2 -penalty -3 -evalue 1 -word_size 4 -max_target_seqs 1000000 -max_hsps 1. An e-value threshold of 1 was selected such that only alignments where less than one hit of a similar score could be expected to be seen by chance (from a database of that size) would be given.

The final computational demultiplexing and analysis pipeline (**Figure 4.6**) involved aligning the barcode sequences of all designs against all reads using BLASTN with optimised parameters and then associating each read with the design that had the best alignment score (**Methods**).

Reads with multiple best alignments did occur, albeit infrequently (<1%), and these reads were omitted from analyses. Reads for each design were then mapped to the appropriate plasmid reference sequence and design-specific read depth profiles generated. To reduce the false assignment of barcodes entirely we had to design barcodes such that they could not be mistaken for one another.

### 4.3.3 Design criteria for intrinsic barcodes

The role of the intrinsic barcode is to facilitate multiplexed sequencing of a pooled library by enabling sequencing reads to be matched to the correct design using the design sequence alone [48]. Intrinsic barcodes were designed to enable demultiplexing of sequencing reads from both nanopore DNA-seq and dRNA-seq datasets. In this chapter we focus on characterising the sequence of the genetic parts encoded in the intrinsic barcode; in chapters 5 and 6 we study the function of the genetic parts which the design encodes.

Due to the 85-95 % per-base accuracy of nanopore sequencing, reads arising from intrinsic barcodes which are highly similar are difficult to distinguish. Demultiplexing trials highlighted that in our first library (L1) design (**Figure 4.3 A**) BLASTN often aligned the wrong design to a simulated sequencing read by matching only part of the design barcode to the design. For example aligning to spacer and modifier whilst aligning to a different (or no) terminator. Initially we used an example intrinsic barcode (S1-M1A-T2) and used BLASTN to identify simulated sequencing reads to which this aligned. Genetic part T2 differed from one of the other parts (T3) by only a single nucleotide and our simulations indicated that it would be impossible to distinguish sequencing reads for these terminators behind the noise of the high error reads which had one error per 10 nucleotides on average.

Thus it became apparent that the library design was a key factor determining the ease with which sequencing reads could be demultiplexed. The set of modifiers in the initial library design L1 proved too similar to demultiplex the simulated error-prone nanopore sequencing reads using BLASTN. This was because whilst the modifiers were unique, they shared identical sub-sequences based upon the three random template sequences (**Figure 4.3 B**). Therefore designs had regions within the modifier which were identical to part of the modifier sequence in other designs in the library. Whilst this would be useful for making direct comparisons of the effect of modifier length where sequence context did not change, it proved challenging for sequence demultiplexing. Using the simulated data, BLASTN parameters could not be found which correctly demultiplexed the modifiers in the initial library.

Given the difficulty demultiplexing simulated reads for the initial library using the intrinsic barcodes, attempts were made to match reads to the different elements of the design (spacer, modifier, terminator) separately. To take this approach, parts had to be extended into neighbouring sequence by for example appending 4-base pair ligation overhangs (or further still, into the neighbouring backbone sequence where possible) to allow BLASTN the minimum query sequence

it required to identify alignments. Without extension, significantly fewer reads than expected are aligned. Using the full intrinsic barcode as an alignment query was selected for our final computational analysis pipeline since no sequence extension was required in this case.

Nonetheless, these attempts to align to specific parts were key to find the limits of BLASTN query sequences (which must be > 30 nucleotides) and optimise BLASTN parameters. They confirmed that the complications arose from the similarity of the sequences of the modifier variants. Identical sequences could be avoided by looking at only the spacer or terminators (except T2 and T3) and in these cases sequences could be accurately demultiplexed. Given these challenges, for the final library design, L2 (**Figure 4.3 C**) each part was selected such that it did not have significant sequence in common with any other part.

Informed by these findings we identified a set of parts with limited BLASTN alignments to one another or the plasmid backbone (threshold of < 10 nucleotides) using iterative all-all BLASTN alignments. This involved aligning the set of parts for a particular part type against one another using BLASTN (e-value threshold: $10^{-}10$ and refining the set until there were no alignments between parts. Further all-all alignments of all parts to be assembled were completed using BLASTN (e-value threshold: 0.2) and the set was refined such that sequences did not have a stretch of > 10 nucleotides identical to any other part, whilst still fulfilling the initial criteria. This ensured that nanopore sequencing reads arising from the final library (**Figure 4.3 D**) of 1183 designs could be demultiplexed using BLASTN.

Nanopore DNA-seq characterisation of the initial library design revealed that intrinsic barcodes must be sufficiently distinct to enable them to be distinguished despite the nanopore error rate. Our simple strategy to achieve part dissimilarity ensured design of a library, L2, where no challenges were encountered during demultiplexing of pooled nanopore DNA-seq characterisation.

## 4.4 Nanopore DNA sequencing characterisation of entire DNA libraries

Nanopore DNA-seq was able to reveal the composition of our redesigned DNA library (L2) under different amplification protocols. The results showed that designs in the combinatorial DNA library are successfully assembled *in vitro*. Furthermore, some amplification methods produced DNA libraries with sufficient uniformity for characterisation in nanopore sequencing assays, whilst others did not (**Figure 4.7 A**). Nanopore DNA-seq can reveal both the library composition (this Section) and the occurrence of mutations during DNA assembly (next Section). Both of these properties are important for interpreting the results of multiplexed sequencing based assays, yet they are rarely investigated.

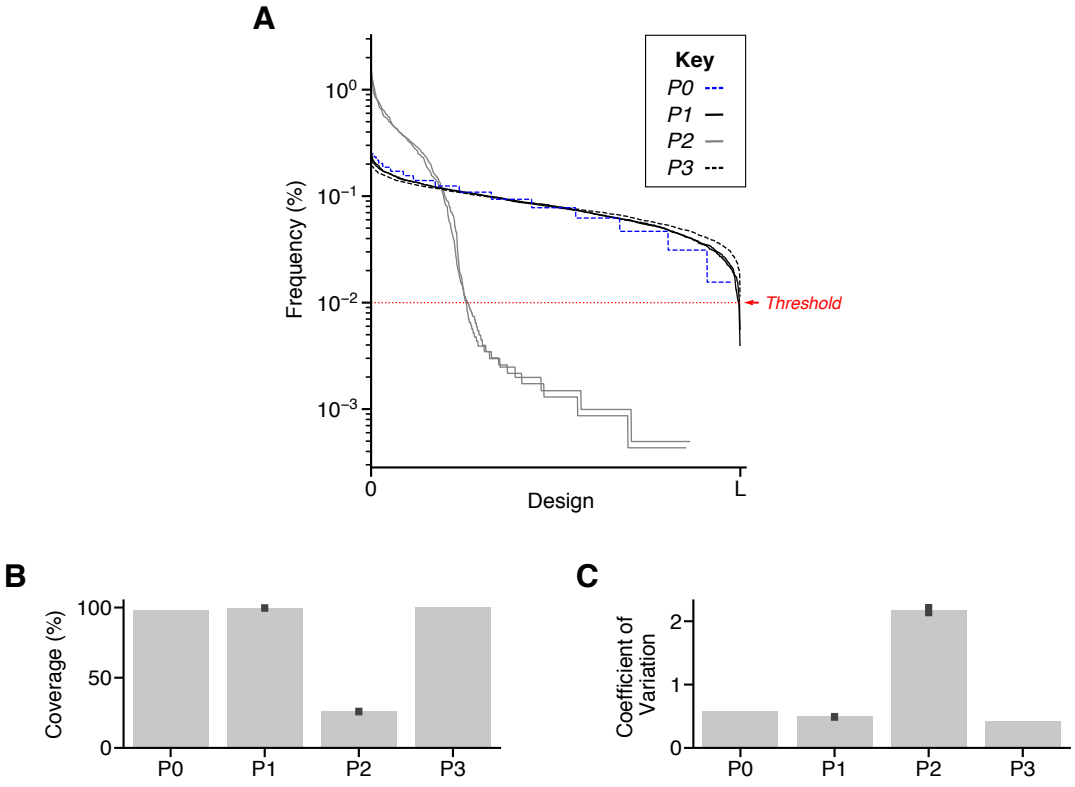



Figure 4.7: **Analysis of library composition using nanopore DNA sequencing data.** (**A**) Library compositions in terms of frequency of each design (%) for libraries P0 (dotted blue line), P1 (solid black line), P2 (solid grey line), P3 (dotted black line). Multiple replicates are shown for library preparations where measured. The red dotted line indicates the threshold frequency for characterisation in a nanopore sequencing assay (100 reads per design per million sequencing reads). Each distribution is ranked in order of frequency. (**B**) Percentage of designs in each DNA library preparation that reach the characterisation threshold. Replicate measurements are summarised with an error bar of length maximum deviation from the median value. (**C**) Coefficient of variation for each DNA library preparation.

We found that only the libraries generated without expression of RNA or protein could be fully characterised in our nanopore DNA sequencing studies (**Figure 4.7 B**). To calculate a measure of library coverage, we calculated the proportion of designs in each library that exceeded a threshold frequency of 100 reads per million. This threshold was selected based upon the minimum number of reads required (100) to characterise a sequence or function and the minimum number of reads per nanopore sequencing experiment (1 million reads can be expected from a typical sequencing run on a single flow cell). This indicated that over 99% of designs in libraries P0, P1 and P3 could be characterised in a sequencing assay compared to approximately 30% of designs in library P2, where RNA and proteins were expressed. The coefficient of variation offers a useful summary statistic for comparing the distribution of sequencing reads amongst

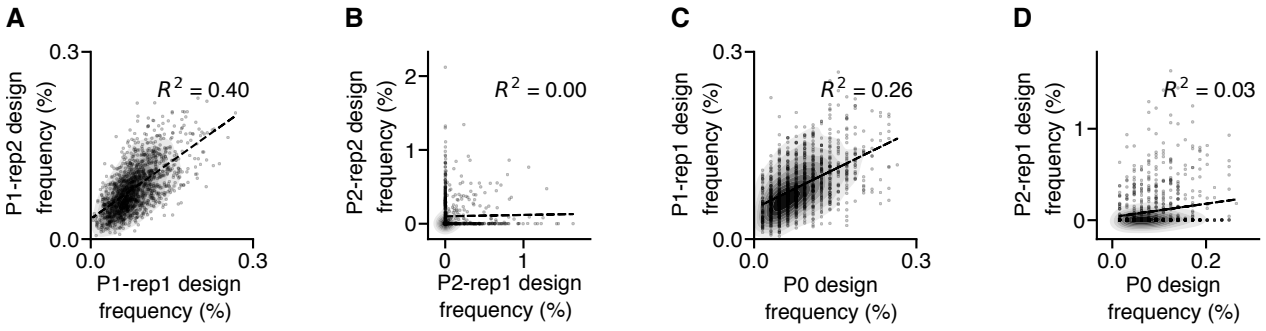



Figure 4.8: **Comparing design frequencies between DNA libraries prepared by different amplification protocols.** (**A**) Comparing design frequencies between replicates of P1. (**B**) Comparing design frequencies between replicates of P2. (**C**) Comparing design frequencies between P1 (replicate 1) and P0. (**D**) Comparing design frequencies between P2(replicate 1) and P0.

designs (**Figure 4.7 C**). It is calculated by dividing the standard deviation by the mean for design percentage frequencies. This summarises the variation of design frequencies in a single number. These results demonstrate that design frequencies vary far more when there is expression of RNA and protein from the plasmid. This likely arises since cells expressing certain genetic designs have a selective advantage and become more frequent within the population of cells from which library DNA is extracted.

The library distribution was consistent between experimental replicates (**Figure 4.7 A**). However, the frequency of each design was only consistent for replicates of P1 (**Figure 4.8 A**). For P2, where the library contained only 30% of the intended designs, each replicate contained a different set of designs (**Figure 4.8 B**). The frequency of designs before and after amplification of the DNA library is correlated for P1 (**Figure 4.8 C**), indicating that the DNA assembly method determines the composition of the library in this case. This suggests that for P2, where genetic design frequency cannot be predicted by the frequency of the design before amplification (**Figure 4.8 D**), the library distribution may a consequence of chance rather than fitness. Further investigation into the correlation between the most frequent designs and their predicted function could be completed once the designs have been characterised *in vivo*.

We looked beyond design frequency to the frequency of the different parts in the library (**Figure 4.9**). Median part frequencies matched the ratios expected from equiprobable assembly (7 spacers would have 14% reads each, 13 modifiers: 8% and 13 terminator: 8%), except for short modifiers 15 bp long which were under-represented, even before amplification. This may be due to reduced assembly efficiency or fewer BLASTN alignments for shorter parts. Libraries amplified with RNA and protein expression showed greater variation in part frequency. This could be because certain parts presented a selective advantage caused by regulation of the expression of the upstream GFP gene or downstream RFP gene (**Figure 4.3 C**). This could explain the

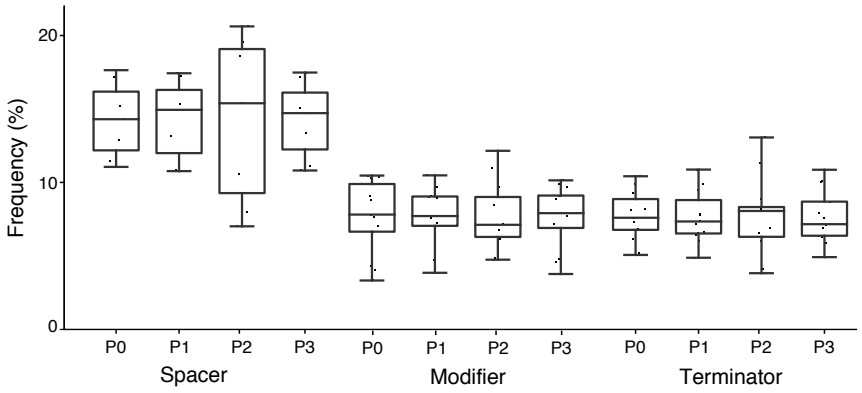



Figure 4.9: **Part frequencies for each DNA library preparation.** Box-plot of frequency (%) of the 7 spacer variants, 13 modifier variants (%) and 13 terminator variants amongst demultiplexed sequencing reads for one replicate of each DNA library preparation protocol. Boxes represent inter-quartile range at outer edges and median within; whiskers indicate the range.

relatively large distribution of spacer frequencies for P2, since the spacer was designed to be translated as part of the GFP coding sequence.

To summarise, amplifying DNA assemblies without RNA or protein expression resulted in significantly better DNA library coverage (**Figure 4.7 B**). Furthermore, amplification of DNA libraries in liquid culture can be adequate for preparing libraries for characterisation in high-throughput sequencing assays. On the other hand, amplifying the DNA library in a host where genetic designs are expressed gave DNA libraries with very uneven coverage. Due to our library design, the amount of RNA and protein expressed by each genetic design will differ and the designs exerting the least burden on host cells are likely to become more abundant. Therefore the composition of library designs is biased during amplification with expression and should be avoided. Whilst abundances of individual designs vary significantly (**Figure 4.8**), abundances of different genetic parts used in the combinatorial assembly vary less (**Figure 4.9**) and this could become important if representative rather than full library characterisation is required. Our studies highlight how simple changes to protocols for amplifying DNA libraries can significantly affect their composition.

The ability to demultiplex sequencing reads also varies depending upon the amplification protocol (**Figure 4.10**). Before amplification, 94% of sequencing reads do not map to any intrinsic barcode. These reads likely represent unassembled DNA, since the DNA assembly is completed with the inserted duplex DNA in 3-fold excess and plasmid DNA had not yet been purified from these assembly components. Even for plasmid DNA purified after amplification, there are reads that are significantly shorter than the full length plasmid and do not align to a design

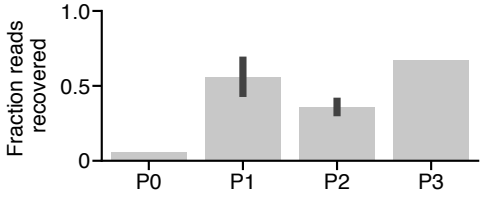



Figure 4.10: **Fraction of sequencing reads with an alignment to a design for each DNA library.**

(approximately 30%) because no intrinsic barcode could be assigned. However, we found that the fraction of demultiplexed sequencing reads approximately halved when the libraries were grown with expression of RNA and protein. This could be due to an elevated mutation rate arising from the expression of multiple proteins and RNAs from the plasmid. Therefore we searched for mutations in the sequencing reads, elucidating features of sequences within the DNA libraries that would otherwise go unnoticed.

## 4.5 Nanopore sequencing reveals mutations in DNA libraries

Mutations are unavoidable when working with DNA sequences that are copied by living cells. Whilst BLASTN and minimap2 successfully aligned and mapped our reads to the intrinsic barcodes of our designs, there could still be mutations in the sequencing reads. In the following sections we assess several types of mutation. This gives an insight into the benefits of using nanopore DNA-seq to verify the integrity of DNA libraries.

### 4.5.1 Insight 1: single nucleotide resolution of DNA assembly

We begin by looking for single nucleotide changes to individual intrinsic barcodes (**Figure 4.11 A**). These are also known as single nucleotide polymorphisms (SNPs), which are a feature of all genomes [173]. To identify SNPs, we generated a consensus sequence for each design with more than 10 reads from the nanopore DNA-seq data (**Methods**) for each assembly protocol (except for P0, which had too few reads to calculate consensus sequences since this DNA was not amplified). After aligning the consensus to the reference sequence we could compare the two sequences and identify nucleotide changes within the design. Since some designs were longer than others, the gap in SNP counts at the end of shorter designs was filled with zeros to allow SNP counts for all consensus sequences to be collated. Our analysis indicates a much higher abundance of SNPs in the libraries where the plasmid was expressed during amplification.

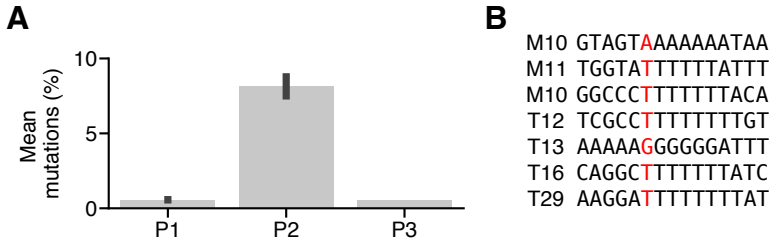



Figure 4.11: **Analysis of single nucleotide polymorphisms (SNPs) in assembled DNA libraries.** (**A**) Mean intrinsic barcode nucleotides with SNP for each DNA library preparation. Replicate measurements are summarised with an error bar of length maximum deviation from the median value. (**B**) False positive SNP are called at homopolymers in two modifier (M) and five terminator (T) genetic parts. The SNP position is indicated in bold red and the 5 nt upstream and 9 nt downstream are shown.

Whilst we found a mean of 1% SNPs per design for P1 and P3 (**Figure 4.11 A**), further investigation indicated that these SNPs were likely to be false positives (**Figure 4.11 B**). These SNPs only occurred at nucleotides before homopolymer regions. We hypothesise that these SNPs are an artefact of the base-calling algorithm [130], which often struggles to define the length of homopolymers accurately. Nanopore base-calling is error-prone at homopolymers because it is difficult for the algorithm to calculate the number of consecutive nucleotides from the electrical signal measured during the sequencing process.

In our case it appears that the number of nucleotides in homopolymers within the design was consistently underestimated, resulting in a consensus missing one nucleotide of the homopolymer. After consensus calling and alignment our computational pipeline therefore predicts a deletion just before the homopolymer. We found no SNPs in the regions where the oligonucleotides are ligated to one another or the plasmid backbone. Using the same analysis, of the consensus sequences generated for P1, only one had one SNP predicted in an equivalent (100 bp) length region of the GFP encoding sequence. This suggests that these DNA assembly and amplification methods proceeded with minimal errors. To corroborate this assessment of ligation fidelity we confirmed the sequence of several individual library members using Sanger sequencing.

P2 showed many more SNPs amongst the DNA library. The number of SNPs inversely correlated with the number of reads for a design. It could be that designs with large SNPs had fewer reads assigned since the mutation they contained decreased identification of sequencing reads to the intrinsic barcode sequence. The large number of mutations combined with the low library coverage render this approach inappropriate for generating defined DNA libraries for characterisation.

The computational pipelines that we have developed offers a starting point for studying

where mutations occur. Large consecutive SNPs were mostly insertions or deletions rather than substitutions. We calculated the frequency of SNPs across the plasmid sequence for consensus sequences generated for designs from libraries assembled with protein expression. Positions with false positive SNPs arising from basecalling issues had an indel in approximately 40% of cases. However looking beyond these outliers reveals crests and troughs in the profile of mutation frequency (**Figure 4.12**). These trends were reproduced across independent biological replicates. Mutations were elevated at particular regions within the plasmid: especially within regulatory regions.

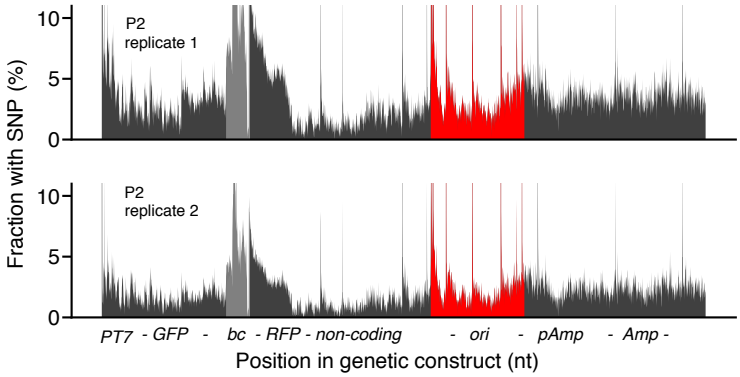



Figure 4.12: **Analysis of single nucleotide polymorphisms (SNPs) in a genetic design.** Percentage SNP is shown for each nucleotide of the plasmid. The genetic design is annotated as a track underneath, beginning at the promoter $P_{T7}$. The intrinsic barcode and origin of replication regions are highlighted in light grey and red respectively. The y-axis maximum is 10 as any outliers with more than 10 % SNPs arise due to basecalling errors at homopolymers. Data calculated from both replicates of P2 are shown.

### 4.5.2 Insight 2: insertion sequences

Insertion sequences (ISs) have been defined as DNA sequences < 2500 nucleotides with cryptic function that are commonly inserted into DNA sequences at multiple sites in a target molecule [178]. ISs have a simple genetic organisation generally encoding no function other than those required for their mobility. An IS typically comprises DNA sequences at each end which are required for DNA recombination and surround one or two genes encoding transposases, which recognise and process the recombination sequences. ISs have been categorised into various families and the members of each family have a similar consensus sequence. ISs are frequently involved with accessory functions, that is, facilitating the sharing of genetic information between bacteria via plasmids. However many form an integral part of the chromosomes of their host and

participate in chromosome rearrangements and plasmid integration [178]. Due to their role in mediating host-plasmid interaction we decided to study whether any ISs were integrated into the genetic designs in the DNA library.

We searched for a short 100 nt sequence within representatives of 7 IS families (IS1A, IS2, IS3, IS4, IS30, IS150, IS186) amongst the DNA-seq reads from each sample. We studied only alignments with alignment length > 80 nucleotides and percentage identity > 80%. IS sequences were found infrequently (up to 2 per 1000 sequencing reads) for all samples (**Figure 4.13 A**). Searching for a scrambled IS sequence yielded no alignments and searching for a different 100 nt sequence representing each family returned the same BLASTN hits indicating that these BLASTN hits are bonafide. The majority of IS sequencing reads are filtered out during the alignment, mapping and consensus generation process (**Figure 4.13 B**). ISs do not explain the indels causing the SNPs for P2. The SNP indels are smaller insertions and deletions (10s of nt rather than 1,000s of nt). To summarise, whilst ISs are present in the DNA library preparations, they are unlikely to affect results when mapping to a known reference sequence as sequencing reads containing them are filtered out through this process.

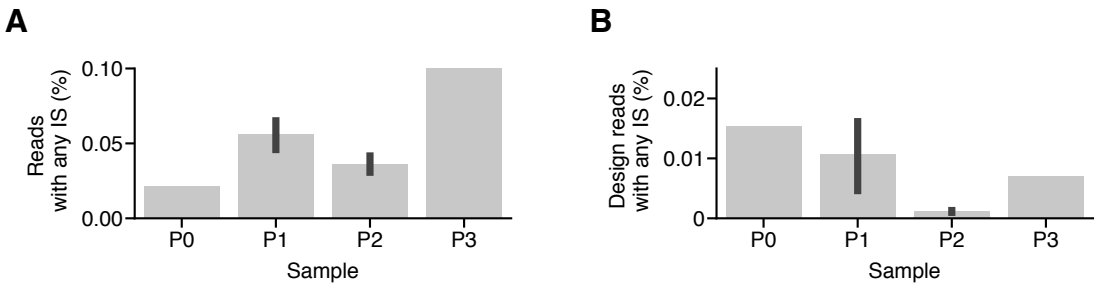



Figure 4.13: **Analysis of insertion sequences (ISs) within DNA sequencing reads.** (**A**) Percentage of sequencing reads with an IS alignment for each DNA assembly method. (**B**) Percentage of demultiplexed sequencing reads with an IS alignment for each DNA assembly method. Replicate measurements are summarised with an error bar of length maximum deviation from the median value.

### 4.5.3 Insight 3: dimer sequences

We noticed a small number of sequencing reads were double the length of the plasmid backbone. We studied the number of sequencing reads (before alignment) that were monomers or dimers. We set "monomer" and "dimer" length windows of 3500-4000 and 7200-7700 nucleotides and calculated their ratios. For P1 and P3, there were around 1% plasmid "dimer" reads (**Figure 4.14**). A similar ratio was observed for the *in vitro* sample (P0). This is not surprising, since the template

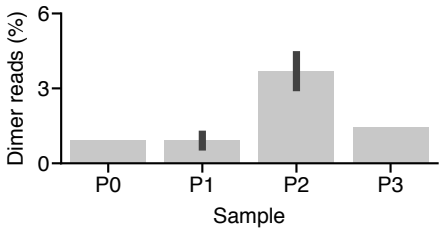



Figure 4.14: **Frequency of genetic design dimers** Percentage of sequencing reads which are dimers for each DNA assembly method. Replicate measurements are summarised with an error bar of length maximum deviation from the median value.

plasmid used in *in vitro* DNA assembly had itself been prepared by amplification *in vivo* and therefore would likely contain both ISs and dimers. However P2 resulted in around 3-fold more sequencing reads for dimers (approximately one dimer per thirty monomers). This indicates that dimers could arise as a consequence of the amplification host and not during the DNA assembly or sequencing library preparation. Dimer genetic constructs could pass through alignment and mapping protocols and therefore change function *in vivo*. Measurements of dimers should be filtered out using sequencing read length when analysing sequencing experiment results.

## 4.6 Discussion

*In vitro* combinatorial DNA assembly by ligation correctly assembled complete DNA libraries from variants of 3 different genetic parts and a plasmid backbone. Though the molecular assembly method leaves "scars" where the DNA is ligated, it is a low-cost, quick and pooled approach which can assemble genetic constructs where modular parts (genes and regulatory elements) are combinatorially co-varied. The DNA assembly approach that we use costs £0.30 per 150 nucleotide design for our assembly. However depending upon the number of parts and part-variants used in the assembly, this could fall. This approach is limited to part lengths of 60 nucleotides or less at present (including overhangs for ligation), which is adequate for many regulatory genetic parts (promoter, RBS, terminator). Synthesising longer segments of DNA is significantly more costly. Oligonucleotides are easy to design, inexpensive and can be ordered as pairs and annealed or ordered as duplexes. This DNA assembly approach could easily be extended to create genetic designs with variation at multiple sites across a plasmid.

Compared to sequencing-by-synthesis, nanopore sequencing has the advantage that each sequencing read corresponds to a single genetic design. But how much does it cost? For the DNA libraries studied here ~100,000 sequencing was sufficient to characterise each library assembled and amplified without expression. This equates to an average sequencing depth of ~85 sequencing reads per design. The sequencing depth required will depend upon the template length, number of designs and library distribution for a given library. This was around 4000 nt, 1183 designs and

percentage frequency CV of  0.5 for this case. For such a library, on a single flow cell, between 10 and 50 DNA assemblies could be characterised at this sequencing depth. That equates to up to 60,000 designs. Even with libraries with such a low CV, more complete coverage would be obtained by running multiple small libraries rather than making and characterising one large library.

This would involve running multiple sequencing library preparations on the same flow cell, which would not be difficult given the speed of the rapid barcoding kit. A single flow cell (in 2021) costs £720, and the kit costs £525, with enough reagents for 6 runs, with up to 12 samples per run: adequate for all 50 proposed assemblies. Therefore, the cost per sequencing library (including the fixed cost of the flow cell) comes to between £25 and £125, without factoring in DNA assembly or bioinformatic analyses. This equates to a sequence-verification cost per design of between £0.02 and £0.10, making this a cost-effective approach compared to Sanger-sequencing which costs approximately £5 per 1000 nucleotides of each design characterised.

Our results indicate that DNA assemblies prepared for sequencing assays using liquid culture amplification give adequate library coverage. This is a much faster and less resource-intensive method compared to the literature precedent of amplifying individual colonies on agar plates. We found that plasmid amplification improved sequencing read recovery: only 6% of sequencing reads aligned to a design for the unamplified *in vitro* sample. Growth in liquid culture (P1) gives adequate library coverage but when encoded RNA and proteins are expressed (P2), poor library coverage is observed. Our library contained plasmids of similar length (+/- 50 nucleotides) and nucleotide content, though if the same method was applied to a library where these parameters varied, amplification could be biased. Changes to library composition should be considered in when characterising the function of DNA libraries *in vivo*, where they must be expressed. *In vivo* characterisation is normally carried out on cells growing in the exponential phase, where there is no competition for resources. For P2, amplification in cells for 14 hours may have ended with cells in the lag phase, where competition for resources could have been exacerbated relative to the exponential phase. Library bias is unlikely to occur for *in vitro* sequencing assays measuring function. Of P1 and P3, which both had satisfactory library distributions, P3 produced significantly more DNA using the scraping protocol and a marginally more uniform library distribution. Therefore it was selected for use in assays to measure the function of the genetic designs (chapters 5 and 6).

There remains room for improving the DNA assembly and amplification protocol to reduce bias in library distributions. Whilst the library distribution was slightly more uniform for the scraping protocol (P3), this protocol is significantly more time and resource consuming and there were still up to 3-fold differences in frequency of designs. This appears to arise during *in vitro* DNA assembly (**Figure 4.8**). During outgrowth on agar in the scraping protocol, colony sizes were visibly different yet did not influence the distribution of design frequencies. We anticipate that DNA assembly of libraries of genetic constructs with even coverage is an emerging

challenge for synthetic biologists. Different DNA assembly protocols may result in different DNA library distributions. Whilst not relevant for our study, amplification *in vitro* [28] or in emulsions [114, 135, 247] could reduce bias during amplification. Nanopore DNA-seq offers a rapid approach to assess and optimise different methods for preparing libraries.

A subsequent experiment studied the concentration of DNA transformed in to the cells, showing that this is also important for determining library coverage. We followed protocol P1 with a modification: 4-fold higher concentration of transformed DNA per cell. P1 gives a library which could be characterised with dRNA-seq on one flow cell. However after this protocol modification only approximately 50% of the library was characterised with dRNA-seq on one flow cell with a similar number of sequencing reads. Therefore for a new DNA library, the transformation protocol may need calibration in order to generate uniform DNA libraries in liquid culture. A balance is needed between enough DNA to allow transformation and not to much DNA, which may to prevent fast outgrowth and competition between library members. Outgrowth time is also likely to be important: sufficiently long to amplify the DNA but not too long such that cells begin to compete for resources. We found that a minority (6%) of sequencing reads for the *in vitro* DNA assembly contained a successfully assembled intrinsic barcode. This shows a limitation of preparing DNA assemblies *in vitro* without purification: short oligonucleotides are used in excess in the assembly and if the plasmid is not purified before sequencing, they will be sequenced. These short sequences are enriched in sequencing reads since, being in excess and also much shorter, they are sequenced more frequently and also faster than any plasmid molecules. Therefore, amplification seems advisable for enriching and purifying successfully assembled plasmids. However, depending upon the amount and purity of the DNA library required, alternative approaches to *in vivo* amplification could be considered. For example, using the *in vitro* assembled DNA library directly, or amplification of linear or circular genetic design templates *in vitro* using PCR. Despite the small number of sequencing reads for designs, *in vitro* assembled DNA did have good library coverage and CV.

Our intrinsic barcode strategy successfully enables demultiplexing of sequencing reads using barcodes encoding a functional genetic part. We found that we had to design intrinsic barcodes to be sufficiently dissimilar to distinguish using error-prone nanopore sequencing. Our barcode design criteria are stringent (no two parts can have > 10 consecutive identical nucleotides) and parameters could likely be relaxed. These criteria limit the similarity of designs and therefore the kinds of libraries that can be characterised with nanopore sequencing, favouring those produced with combinatorial DNA assembly and ruling out those produced through saturation mutagenesis. These limitations to library composition will decrease as nanopore sequencing accuracy increases, which in turn facilitates demultiplexing of more similar sequencing reads.

An alternative approach to demultiplex sequencing reads is to use unique molecular identifiers (UMIs) [130] which serve only as a barcode and do not have a function designed in to them. These non-functional barcodes are added to designs before amplification of the design-UMI complex.

The UMIs are used to demultiplex the sequencing reads. The sequence or function measured in the sequencing based assay must contain the UMI, therefore efforts must be made such that the UMI does not affect function: challenging and perhaps impossible, especially in transcriptomics where the template DNA sequence is split in to multiple transcript isoforms. Using UMIs does have the advantage of allowing more similar genetic parts to be studied within the same library since demultiplexing relies on distinguishing the UMI and not the sequence of the genetic parts. Contemporary genetic design takes a modular approach built up from distinct genetic parts, lending itself to nanopore characterisation since each genetic part (gene / promoter / ribosome binding site / terminator) could be considered an intrinsic barcode. For nanopore DNA sequencing, an intact read of the plasmid sequence represents the entire genetic design sequence, making it more amenable to demultiplexing than sequencing-by-synthesis where barcodes often become split across multiple sequencing reads due to the necessary fragmentation step in sequencing library preparation.

Our studies indicate that sequence mutations seldom occur during DNA assembly and amplification without expression. We observe that designs with large mutations (ISs) are filtered out during the bioinformatic analysis pipelines. This is useful when using sequencing to characterise sequence or function as it means that results will not be influenced by these mutant sequences. In the absence of a selection pressure, our DNA assembly method resulted in very few SNPs within the designs, confirming the fidelity of our assembly method. Amplification in cells expressing the plasmid resulted in a selection pressure giving rise to mutations that cannot be accounted for by IS and a higher frequency of mutant dimer sequences. Nanopore DNA-seq enabled us to choose a suitable, low-mutation, DNA assembly and amplification protocol.

Mutations of genetic designs could be useful for diversifying DNA libraries. The frequency of mutations arising after amplification with expression were measured across the plasmid sequence (**Figure 4.11**). This showed an elevation within regulatory regions. This may be useful for identifying parts of a genetic design which incur a burden and are selected against or are amenable to adaptation by neutral drift. However we are likely to have only captured a glimpse of the diversity of mutated sequences.For P2, two thirds of sequencing reads are not aligned to designs (for P1, it is only one third). This pool of sequencing reads likely holds an abundance of mutational diversity. Mutational analysis could be applied to understand the location of mutations, highlighting points within genetic designs prone to adaptation by biological variation.

Whilst computational pipelines that filter out mutants are useful for characterising the intended genetic designs in DNA libraries, further investigations could be undertaken to understand what the mutated sequences are. This could be done by searching for recurring sequences (haplotypes) within the unaligned reads. These haplotypes would include designs with IS, indels, and many other kinds of mutation. This goes beyond the questions about assembling uniform DNA libraries that we set out to answer. Nonetheless, it is important for understanding changes to the library when characterising it *in vivo* with high-throughput sequencing of cells sorted by their

function. These mutants would also show how a library could be diversified by using adaptation mechanisms encoded in the host. In summary, our nanopore DNA-seq study of combinatorially assembled DNA libraries raises as many questions as answers.

CHAPTER 5

# CHARACTERISING TRANSCRIPTIONAL TERMINATORS USING DIRECT RNA SEQUENCING

## 5.1 Introduction

The ability to precisely control gene expression enables modification of the behaviour of living cells. With this goal in mind, synthetic biologists have predominantly focused on developing genetic parts to regulate initiation rates of transcription and translation for genes of interest [196]. However, endogenous gene regulation is often multifaceted, employing diverse mechanisms that affect the stability and processing of DNA, RNA and proteins to create complex regulatory programs [22]. Whilst the promoter controls transcription initiation, terminators control where transcription should end and have been less well explored as parts to regulate complex genetic circuits (**Figure 5.1 A**).

Terminators rarely cause complete termination of transcription [149]. Instead, they terminate with some probability. A terminator can therefore give rise to two possible transcripts (transcript isoforms), one where transcription ended at the terminator and one where the transcribing RNA polymerase reads through the terminator. The ratio of these transcript isoforms can be used to calculate a measure of the probability of termination, referred to as the termination efficiency ($T_e$). In cells, transcription initiation at a promoter followed by incomplete transcription termination can generate diverse transcript isoforms [254]. When designing terminators, a focus is often placed on increasing termination efficiency [47, 192]. This helps insulate the expression of transcriptional units from each other [198] and the host genome [206]. Engineering terminators with a variety of efficiencies could open new avenues to control the stoichiometry of transcript isoforms and thus the genes which they encode.

Development of new genetic parts is often time consuming due to the need to build them and

characterise their function. For transcriptional terminators a common approach is to use an *in vivo* fluorescence assay in which two different fluorescent reporter proteins have the terminator to be tested placed between them [33, 47, 168]. However, various factors beyond the process of termination can affect the measured termination efficiency. For example, terminator structure is thought to play a role in maintaining mRNA stability, preventing it from degradation by RNA hydrolase [111]. He *et al.* studied three terminators in *E. coli* in terms of transcription shut-down degree and upstream mRNA protection capacity, finding that they contribute almost equally to the overall effect of the terminator, which they refer to as the apparent termination efficiency [111]. This may explain why *in vitro* and *in vivo* termination efficiencies have not been found to correlate well [66] (since the cellular proteins which degrade RNA are not present *in vitro*). Nonetheless, insights into termination *in vitro* could prove useful for the recent increase in development of *in vitro* synthetic biology approaches [55, 234]. These approaches to characterising terminators are limited in throughput and scalability, since designs often need to be tested separately and testing cannot be multiplexed. This limits our ability to study large genetic design spaces and understand how to engineer terminators.

More recently, methods employing RNA sequencing (RNA-seq) have been used to provide high-throughput measurement of termination efficiency [116]. Pooled libraries of 1,000s of terminators can be characterised simultaneously. First, a library of genetic designs, each encoding a different terminator, is designed and then constructed using DNA assembly [41]. Then transcription of the DNA library *in vivo* is used to produce mRNA transcripts from the genetic designs, which are sequenced [116, 276]. The result is a pool of sequencing reads representing all of the transcript isoforms produced from each genetic design. The final step is separating the sequencing reads belonging to each design. In order to do this, at the library design stage a unique sequence is selected for each terminator. These unique sequences are referred to as barcodes since each one is different. This enables the sequencing reads arising from each terminator to be identified, using the unique barcode sequence. The process of separating sequencing reads is referred to as demultiplexing. Demultiplexed sequencing reads can then be used to characterise the termination efficiency of each genetic design in the library. RNA-seq is capable of characterising the transcript isoforms produced by terminators [251] and the precise location at which termination events occur by measuring transcriptional profiles capturing RNA polymerase (RNAP) flux along the DNA.

A challenge when using sequencing to measure transcriptional terminators is that for sequencing-by-synthesis methods, a fragmentation step is required during sequencing library preparation [96, 98, 116]. This limits the types of libraries of genetic designs that can be characterised. DNA libraries with large regions of homologous plasmid backbone cannot be characterised since fragmented reads containing only plasmid (and not barcode) sequence cannot be assigned to a particular design. Targeted short read sequencing could potentially be used to overcome this issue by targeting the design barcodes but would suffer from biases present during the

reverse transcription (RT) step and subsequent PCR amplification [52, 189, 199]. Furthermore, depending upon the library design, many specific primers would have to be designed for the library. Nanopore-based direct RNA sequencing (dRNA-seq) does not require fragmentation, RT or PCR and can measure full-length reads of transcript sequences [83], which should always contain the barcode required for demultiplexing. Thus nanopore dRNA-seq is suitable for capturing transcriptional termination at nucleotide resolution.

In this chapter we develop a method to characterise the function of entire pooled libraries of intrinsic terminators using nanopore dRNA-seq. In Section 5.2 we outline the experimental and computational methods used. Then in Section 5.3 we discuss several features of the sequencing read profiles. We then create a model to correct for one of these features (Section 5.4) and compare the sequence (DNA-seq) and function (dRNA-seq) measurements (Section 5.5). Finally, we characterise transcription termination of an entire library of 1183 terminator genetic designs from Chapter 4 (Section 5.6 and 5.7). This shows that the library has a variety of termination efficiencies and reveals the termination profile of each genetic design at nucleotide resolution. We conclude with a discussion of this chapter (Section 5.8). Our methodology and experimental findings offer a novel means to control RNAP flux and transcription in genetic circuits and demonstrate how long-read sequencing can improve our understanding of large genetic design spaces.

## 5.2 Characterising terminators using direct RNA sequencing

We set out to develop a method to characterise pooled libraries of terminators. Crucially, each transcript isoform produced at a terminator encodes its associated terminator sequence either at the 3'-end, if termination was successful, or within the body of the transcript, if transcriptional read-through had occurred (**Figure 5.1 A**). Each genetic design in this library contains a different unique sequence which is made up of three genetic parts: a spacer, a modifier and a core terminator **Figure 5.1 B**). This unique design sequence acts as an "intrinsic barcode" that is present in every read. This allows for individual reads to be attributed to a particular design without the need to separate and barcode each before preparing the sequencing library. All 1183 pooled designs of library L2 were transcribed *in vitro* with T7 RNA polymerase. The RNA transcripts that are produced were then sequenced **Figure 5.1 C**) and the data demultiplexed to simultaneously produce separate read depth profiles for each design. A read depth profile represents the number of sequencing reads at each nucleotide of a design (**Figure 5.1 D**). This simple and fast method can be used to assemble and characterise large libraries of transcriptional genetic parts in under a week.

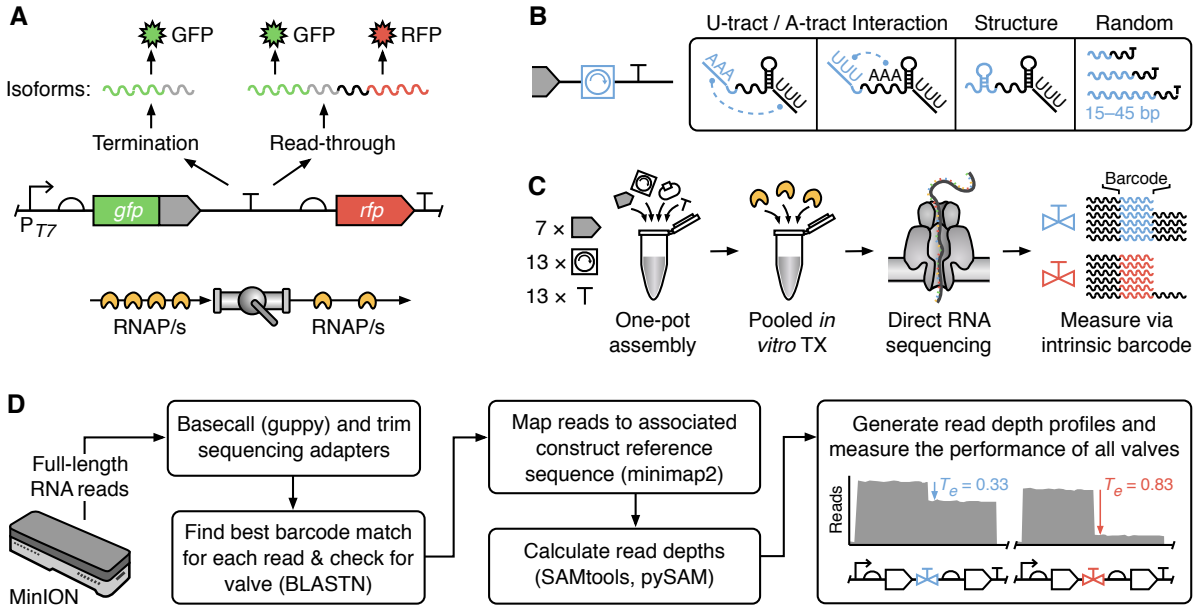



Figure 5.1: **Characterisation of transcript isoforms using nanopore direct RNA sequencing.** (**A**) Schematic of the genetic construct used to characterise transcriptional terminators. (**B**) Our modular transcriptional terminators comprise a "spacer" (grey), "core terminator" (T) and "modifier" sequence (blue) used to tune termination efficiency. Various modifiers were designed to interact with the U- and A-tract of the core terminator, form small secondary structures in the RNA, and act as different length inert spacing elements. (**C**) The steps involved in the assembly of the modular transcriptional terminator library and its pooled characterisation using nanopore-based direct RNA sequencing. (**D**) Analysis pipeline used to generate design specific read depth profiles and calculate termination efficiencies ($T_e$) from pooled direct RNA sequencing data. Key computational tools shown in parentheses.

The nanopore DNA-seq read demultiplexing protocol we developed in Chapter 4 was used for demultiplexing nanopore dRNA-seq reads (**Figure 5.1 D**). The protocol was augmented to enable read profiles to be generated from sequencing reads. The entire computational pipeline involves basecalling sequencing reads, matching reads to designs with the best match to the intrinsic barcode using BLASTN, mapping the matched reads to the reference sequence for that design using minimap2, calculating read depth profiles across the reference using SAMtools and pySAM and finally, measuring termination efficiency (**Methods**). By comparing the ratio of read depths for the two transcript isoforms for a design (i.e., read depth directly before and after the terminator) a termination efficiency can be calculated (**Figure 5.1 D**). This workflow enables characterisation of entire libraries of terminators in a single dRNA-seq experiment.

Some erroneous demultiplexed reads had to be filtered out after generating the read profiles: those where no core terminator sequence was present after alignment and mapping. This step became necessary for dRNA-seq because the RNA transcripts were often fragmented, a common feature of RNA-seq data. A fragmented read missing part of the intrinsic barcode could be

mistaken with many other designs due to the combinatorial assembly approach and therefore was removed.

The dRNA-seq reads from replicate nanopore dRNA-seq runs were pooled to increase read depth for all designs, which results in increased accuracy when measuring termination efficiency. The distribution of design frequencies included some designs with relatively few sequencing reads, making pooling of the datasets necessary. This set of pooled sequencing reads was then demultiplexed and a read depth profile was created for each design. A termination efficiency ($T_e$) was calculated for each read depth profile (**Methods**) and further corrected (**Section 5.4**).

## 5.3 dRNA-seq sequencing read profile features

To begin with, the transcription profiles for the negative control, T33, where no termination is expected were reviewed. Several key features were present within the generated transcriptional profiles. First, we noticed that dRNA-seq reads often had 6 nt of their 5' sequence truncated (**Figure 5.2 A**), which could make it difficult to determine precise transcription start sites. As dRNA-seq progresses from the 3' to 5' end of an RNA molecule, this short region likely corresponds to the point where the motor protein that ratchets the RNA molecule through the pore reaches the 5'-end and releases the molecule, causing an increased error rate or removal of the short sequence still contained within the pore.

Second, we found that all dRNA-seq read depth profiles showed a steady decreases in read depth. Since we had mapped sequencing reads to the terminator intrinsic barcode, the read depth is highest at the barcode and this steady decrease occurs in both directions away from the barcode (**Figure 5.2 A**). Such a feature is found in all nanopore dRNA-seq studies to date covering RNA samples from many different organisms [63, 83]. It is thought to arise due to fragmentation of full-length RNA molecules (e.g., by shearing caused during pipetting) and/or premature abortion during sequencing resulting in truncated reads. In contrast, only small drops were observed for nanopore DNA sequencing of the constructs (**Figure 5.2 A**), possibly due to the greater stability of the molecule [157].

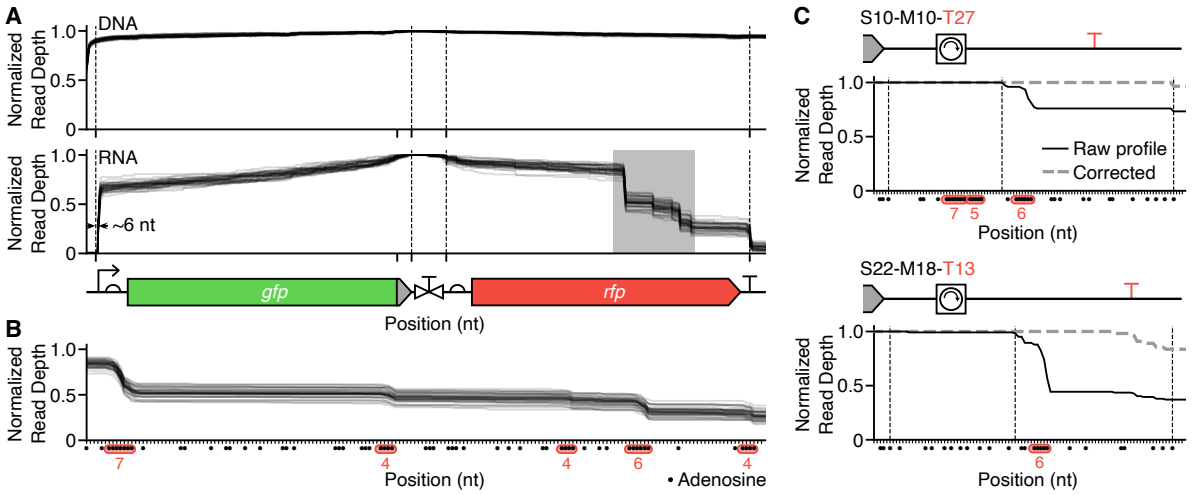



Figure 5.2: **Features of direct RNA sequencing read profiles with inefficient polyadenylation.** (**A**) Normalised sequencing read depth profiles from nanopore-based DNA-seq and dRNA-seq for 42 designs containing the same core terminator T33 (non-terminating control) and modifiers of length 30 nucleotides. Vertical dotted lines denote transcript and terminator boundaries. Plasmid map illustrated beneath, to scale. Grey shaded region is expanded in the panel below. (**B**) Expanded region from panel A showing dRNA-seq read depth profiles with dots corresponding to adenosine nucleotides. Adenosine homopolymers >3 nt in length are highlighted in red and their lengths are shown below. (**C**) Corrected (dashed grey lines) and raw (solid black lines) dRNA-seq read depth profiles. Two different designs where the core terminator contains an adenosine homopolymer within the terminator sequence. Vertical dotted lines indicate spacer-modifier and modifier-terminator boundaries.

The small proportion of sequencing reads representing RNA fragmented within the barcodes used for mapping leads to a minority of erroneous read mappings. This occurs where the sequencing read matches only part of the barcode and it is impossible to accurately align that read to a particular combinatorial design. We removed sequencing reads arising from these mapping artefacts by selecting only reads with alignment across the spacer, modifier and the first 20 nt of the terminator. The model we outline later corrects termination efficiency after the removal of these reads. Termination of T7 RNAP requires both a hairpin structure and U-tract and while both of these elements are found alone in different modifiers, they are not found together except within the core terminator parts, making drops caused by termination highly unlikely outside of this part. We confirmed this by visually checking the raw read profiles, finding no drops within the spacer or modifier region except for those caused by the mis-alignments that this step seeks to filter out.

A third observation was that in the case of inefficient polyadenylation, significant drops in read depth were seen outside of the core terminators and predominantly at short poly-A sequences >3 nt in length (**Figure 5.2 B**). When preparing RNA for dRNA-seq a poly-A tail is required for ligation of sequencing adapters to the 3'-end of the RNA molecules. As *in vitro*

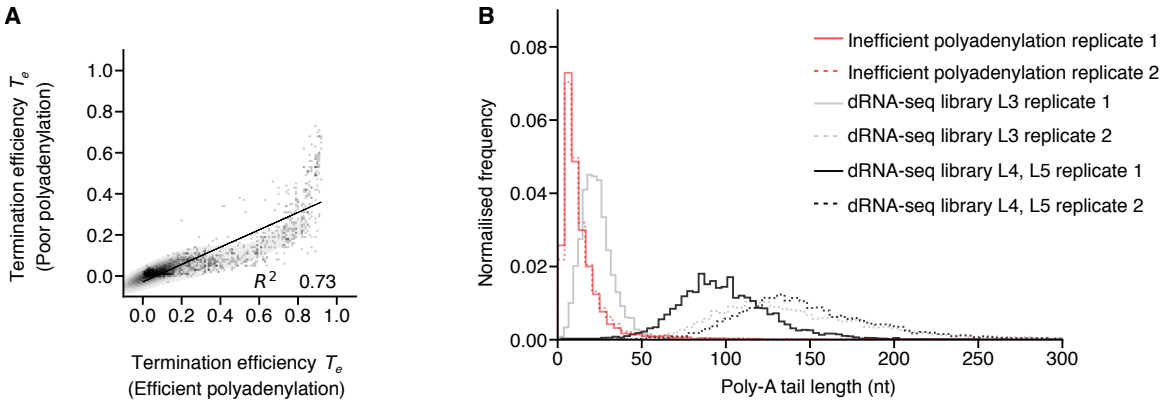



Figure 5.3: **Investigating polyadenylation efficiencies.** (**A**) Comparison of termination efficiency with and without efficient polyadenylation during sequencing library preparation (library L2). (**B**) Histograms showing the varying lengths of RNA poly-A tail lengths for several sequencing libraries prepared. RNA from L4 and L5 (see Chapter 6) were pooled and prepared together as a single sequencing library. See Appendix for library compositions.

transcription of our constructs will not produce transcripts of this form, we used *E. coli* poly(A) polymerase to polyadenylate all the RNAs produced (**Methods**). Analysis of the dRNA-seq data showed <10 nt poly-A tails were present, which were shorter than other dRNA-seq runs we had previously performed (**Figure 5.3 B**).

We hypothesised that inefficient polyadenylation allows for fragmented RNAs with a short poly-A end to become enriched during sequencing and thus causes notable drops at these points within a construct that do not correspond to termination events. Our subsequent dRNA-seq runs with efficient polyadenylation do not show these drops in read depth at adenosine homopolymer regions. For runs with inefficient polyadenylation, we could partially correct read profiles for designs containing parts with poly-A regions in their template strand (i.e., M10, T13 and T27) by retaining only mapped reads which do not terminate at a poly-A motif outside the terminator hairpin (**Figure 5.2 C**). However, even with this correction, $T_e$ measurements were significantly affected for all designs (**Figure 5.3 A**) and therefore we repeated the experiments with efficient polyadenylation. The polyadenylation step varied between replicates; using fresh *E. coli* Poly(A) polymerase enzyme that had not been freeze-thawed ensured that all RNAs were polyadenylated (**Figure 5.3 B**) and this approach was taken for the data presented hereon in.

## 5.4 Modelling direct RNA sequencing

To validate the hypothesised causes of RNA fragmentation and explore their possible impact on $T_e$ measurements, we developed a mathematical model and used data from an RNA Control Strand (RNA CS) that is externally "spiked-in" to each dRNA-seq experiment for quality control

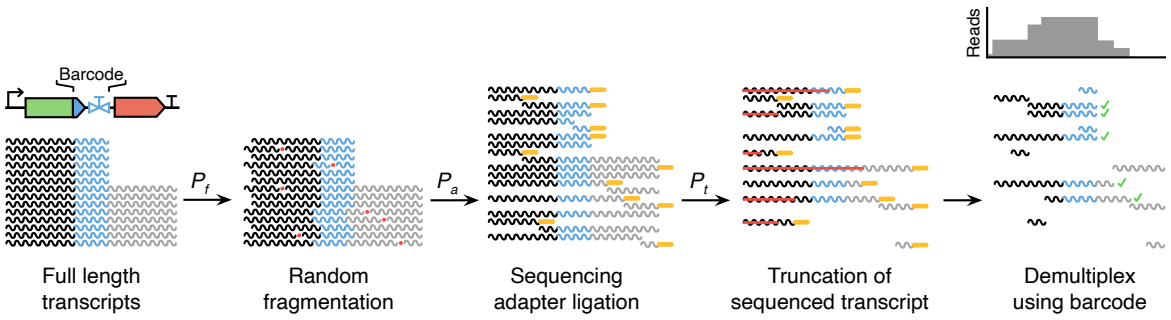



Figure 5.4: **Overview of the direct RNA sequencing model.** Reads are denoted by squiggles that are colour coded to show core regions (e.g., blue region is the intrinsic barcode). Red dots show points of random fragmentation, orange oblongs represent sequencing adapters attached to only the 3'-end of an RNA molecule, and green ticks denote reads that contain a complete barcode sequenced and which are used to generate a read depth profile. $P_f$, $P_a$, and $P_t$ are probabilities that reads are selected for each of the modification steps (i.e., random fragmentation, adapter ligation, and truncation, respectively).

assessments. Because the RNA CS is a single fixed length sequence, we could use it to test how different amounts of fragmentation or sequencing abortion affect the shape of the read depth profile recovered. We found that experimental data could be well described by a simple model with three probabilistic processes: fragmentation before ligation of sequencing adapters, successful adaptor ligation, and sequencing read truncation (**Figure 5.4**). Sequencing read truncation could be caused by RNA fragmentation (after adapter ligation) and/or early abortion of the sequencing process. We developed a simple probabilistic model to capture the key processes impacting the reads recovered from a direct RNA sequencing (dRNA-seq) run.

We begin by assuming that all starting RNA transcripts correspond to either an isoform that terminates at the terminator or at an appropriate point downstream of the terminator. First, reads are chosen with probability $P_f$ to become fragmented at a random location along their length. This step captures the inevitable fragmentation that occurs when extracting and purifying an RNA sample. Next, sequencing adapters are attached to full length transcripts and fragmented RNAs with probability $P_a$ and only molecules with an adapter attached are taken forward for sequencing. Sequenced molecules are then chosen with probability $P_t$ for truncation at a random position along the sequence. This step captures possible further fragmentation of the RNA during sequencing library preparation whereby only the fragment containing the adapter is sequenced, or possible truncation of reads due to premature termination during the sequencing of a molecule. In both cases, this significantly reduces the information captured per read and renders many reads impossible to demultiplex when truncation occurs downstream of the intrinsic barcode. Finally, we filter out any that do not contain a complete terminator design (i.e., intrinsic barcode). Reads without a full barcode cannot be uniquely identified and so the reads are removed during the demultiplexing step. Reads that make it through these steps are

then used to generate a read depth profile.

To demonstrate the model's ability to capture read depth profiles generated from real sequencing data, we made use of sequencing reads for the RNA Control Strand. The RNA CS is a single known sequence unlike any other in our library and only consists of full-length RNA molecules. Fitting our model to dRNA-seq data from the two biological replicates, we found that parameter values of $P_f$ = 0.1, $P_a$ = 0.66 to 0.90 (depending upon the sequencing run) and $P_t$ = 0.45 enabled a close fit for all sequencing runs, with only minor deviations at 5' and 3' ends of the RNA CS sequence (**Figure 5.5 A**). We also assumed the presence of an intrinsic barcode in the centre of the RNA CS sequence and found that our model could also accurately predict RNA CS read depth profiles recovered after demultiplexing of the real dRNA-seq data (**Figure 5.5 B**). This suggests that the read distribution that is generated by the model closely fits that recovered from sequencing.

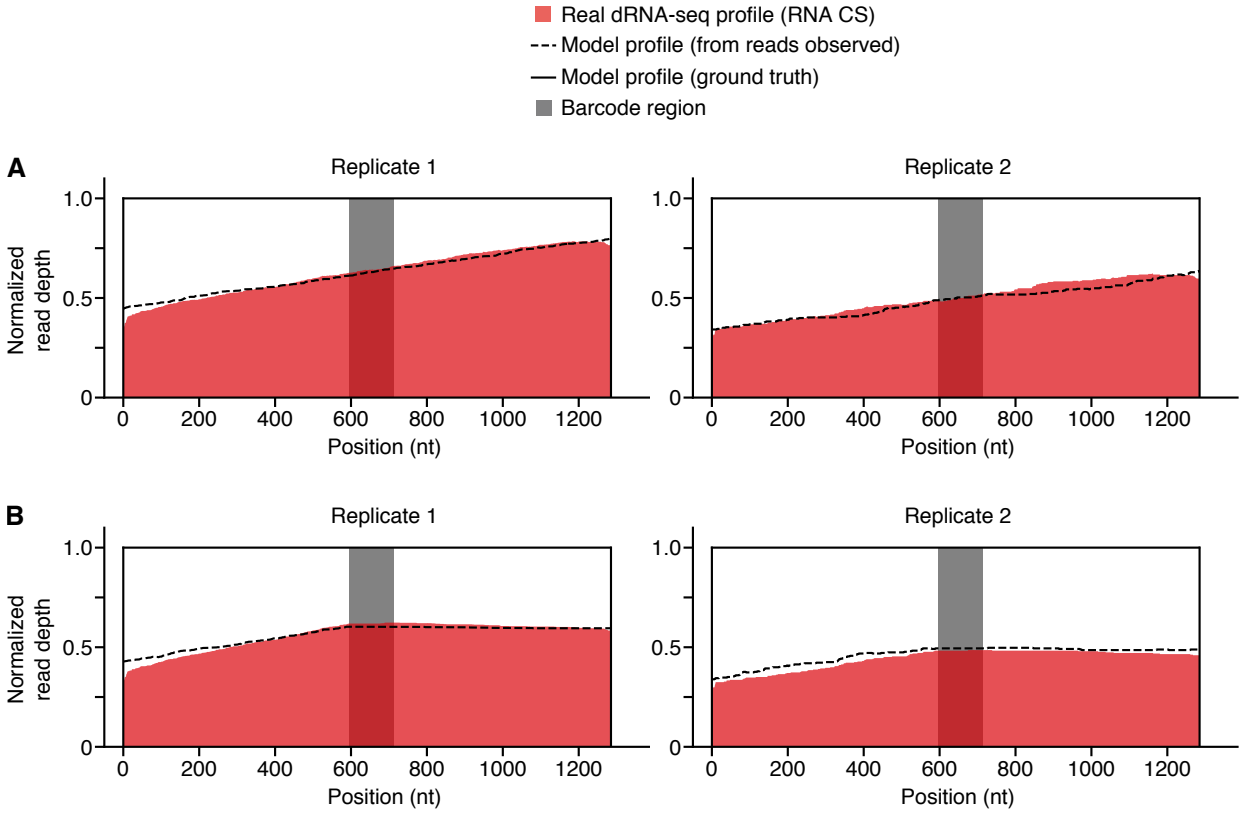



Figure 5.5: **Fitting model to direct RNA sequencing data.** (**A**) Read depth profiles shown for all reads mapping to the RNA CS sequence for two dRNA-seq biological replicates (filled red) and fitted dRNA-seq model used to simulate the processing of the total number of reads with a BLASTN alignment to the RNA CS sequence, where $P_f$ = 0.1, $P_a$ = 0.87, $P_t$ = 0.45 (dashed black line for observed profile, solid black line for the model ground truth). (**B**) Read depth profiles for reads that map to the grey "intrinsic barcode" for the real dRNA-seq data (filled red) and fitted model (dashed black line for observed profile, solid black line for the model ground truth). The termination efficiency for the RNA CS "intrinsic barcode" is zero.

Finally, to assess how well the observed read depth profiles matched the ground truth, we used the model with parameters fitting to the real dRNA-seq data for RNA CS to simulate the sequencing process on synthetically generated transcripts for a hypothetical set of terminators with termination efficiencies varying between 0 and 1. By comparing the actual termination efficiency of each hypothetical terminator with the observed termination efficiency measured from the generated read depth profiles, we found a slight over estimation in $T_e$ (**Figure 5.6**). To ensure this didn't bias our measurements for the data from the real terminators, this deviation was corrected for by subtracting the calculated error from the observed termination efficiency seen in the model simulations, to give a final $T_e$ value. Though $P_t$ varied between sequencing runs, the error correction for any given $T_e$ value was found to be consistent across sequencing runs (±1% deviation).

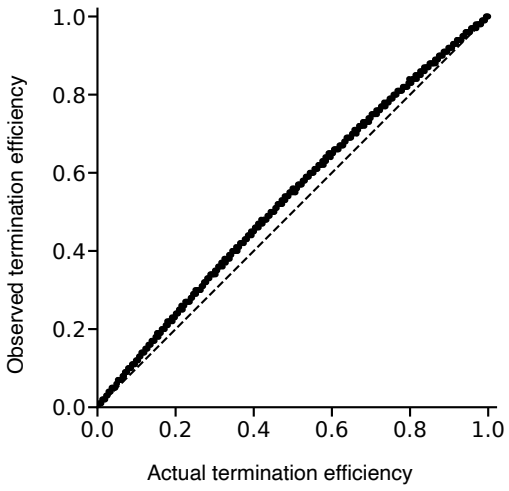



Figure 5.6: **Deviation between observed and actual termination efficiencies.** Each point denotes a model simulation based on 100,000 simulated reads for transcriptional terminators with varying termination efficiencies and parameter values of $P_f$ = 0.1, $P_a$ = 0.87 and $P_t$ = 0.45. Dashed line shows y = x.

While read profiles for RNA CS have a significant decrease only from the 3'-end to 5'-end (**Figure 5.5**), profiles for designs decrease in both directions away from the barcode (**Figure 5.2 A**). It is not clear why this is the case since the RNA CS sequence was included in the *in vitro* transcription reaction and therefore was exposed to the same experimental conditions as our designs. It could reflect increased degradation of *in vitro* transcribed RNA, or a gradual drop-off of T7 RNA polymerase during the process of transcription, both of which would not affect the measured termination efficiencies.

## 5.5 Comparing dRNA-seq and DNA-seq read profiles

We compared the number of dRNA-seq and DNA-seq reads matched to each design. The composition of the dRNA-seq reads was less evenly distributed compared to the DNA-seq reads (**Figure 5.7 A**). After removing erroneous alignments, we found that transcript abundances were weakly correlated with DNA construct frequencies ($R^2$ = 0.22), with strong terminators over-represented in the dRNA-seq data (**Figure 5.7 C**).

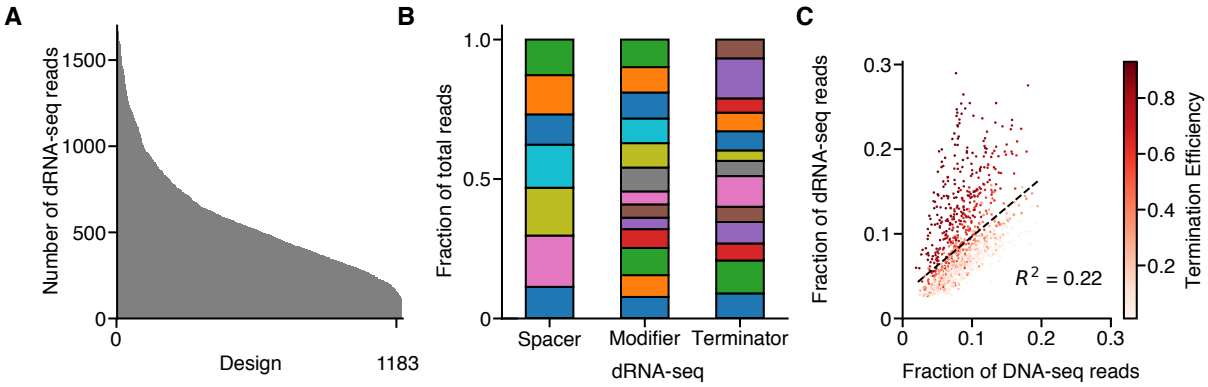



Figure 5.7: **Analysis of dRNA-seq reads of transcribed RNAs.** (**A**) Number of dRNA-seq reads for each design, ordered by number of reads. (**B**) Frequency of each part in the dRNA-seq data. (**C**) Comparison of fraction of dRNA-seq reads and DNA-seq reads representing each design, coloured by termination efficiency. Part and design frequencies were calculated relative to the total number of annotated sequencing reads.

Whilst dRNA-seq and DNA-seq abundance correlated, some designs were more abundant in in the dRNA-seq data (**Figure 5.7 C**). The relative proportion of each spacer, modifier and terminator amongst assigned dRNA-seq reads is shown in (**Figure 5.7 B**). dRNA-seq read abundance correlated with terminator strength and strong terminators were over-represented (**Figure 5.7 C**). Our model predicts this pattern and corrects the effect that this has on termination efficiency (**Figure 5.8**). The model indicates that this arises due to truncation of the sequencing reads (caused by RNA degradation after adaptor ligation or by premature abortion of the sequencing process). For weak terminators, transcripts extend beyond the terminator, therefore there is truncation prior to the intrinsic barcode. When this occurs, the sequencing read is not mapped to any design and the number of reads mapping to a design decreases as the proportion of non-terminated transcripts increases.

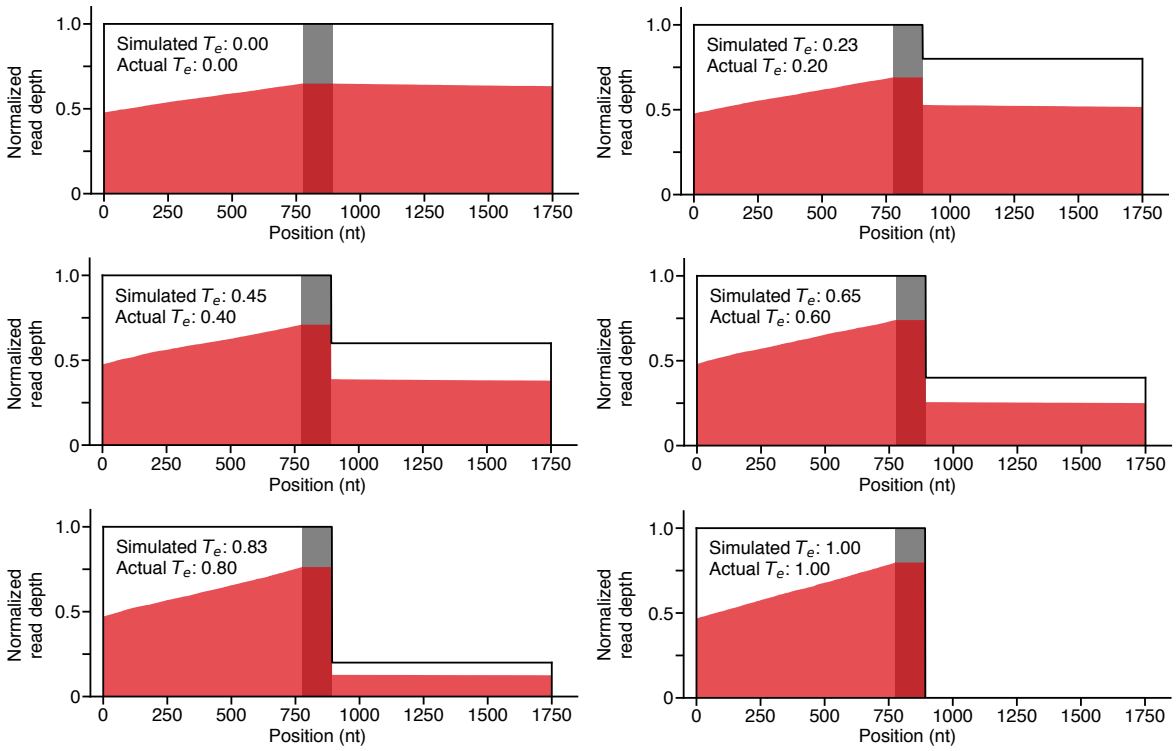



Figure 5.8: **Simulated termination profiles.** Each plot denotes a model simulation based on 10,000 simulated reads for transcriptional terminators with varying termination efficiencies and parameter values of $P_f$ = 0.1, $P_a$ = 0.87 and $P_t$ = 0.45. Solid line shows non-fragmented (actual) transcript profile, red shading shows simulated read profile, shaded area shows design location.

In future experiments, this effect could be minimised by using DNA templates which end shortly after the genetic part, leaving little room for read truncation prior to barcode sequencing. This effect highlights that whilst nanopore sequencing characterises RNAs directly, their representativeness of the actual transcriptional profile depends upon the integrity of RNA after sequencing library preparation and may be affected by premature sequencing termination.

## 5.6 Characterising transcription termination at nucleotide resolution

*In vitro* transcription using T7 RNA polymerase of the entire DNA library L2 followed by dRNA-seq enabled us to rapidly assay the performance of each design simultaneously. The outlined analysis of the generated read depth profiles revealed several key features in line with other dRNA-seq studies [105]. The mathematical model we developed allowed us to correct for unwanted deviations between actual and measured read profiles (**Figure 5.6**). A good reproducibility was observed for $T_e$ values between replicates ($R^2$ = 0.99) and for terminators shared across

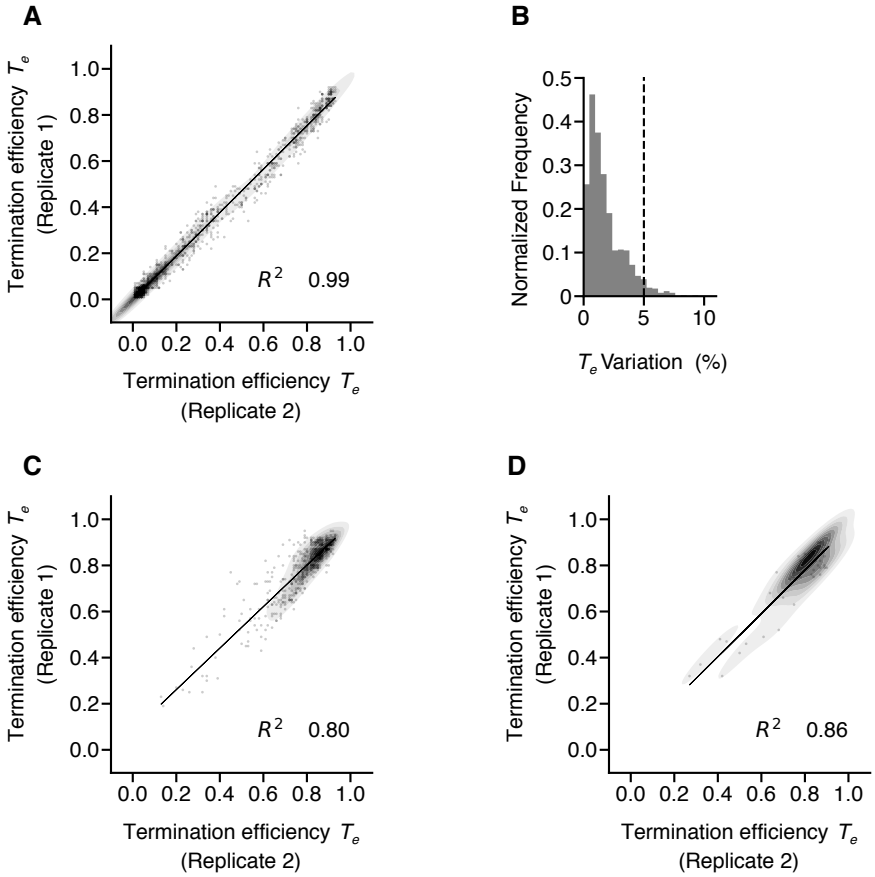



Figure 5.9: **Comparison of termination efficiencies across experimental replicates.** (**A**) Comparison of termination efficiency between experimental replicates of the same library (L2). (**B**). Histogram of absolute difference in termination efficiency measured, for experimental replicates of the same library (L2). (**C**) Comparison of termination efficiency between experimental replicates of the same library (L3). (**D**) Comparison of termination efficiency of constructs shared between two different libraries (L2 and L3). Each point represents a single transcriptional terminator design and dotted line shows the linear regression. $R^2$ is the square of the Pearson correlation coefficient. See Appendix for library compositions.

separately assembled libraries with different part compositions (**Figure 5.9**). This resulted in 98% of designs having a difference of <5% in $T_e$ across the experimental replicates of our initial library (**Figure 5.9 B**).

A valuable feature of part characterisation by RNA-seq is the ability to extract nucleotide resolution insights from the read depth profiles. To enable comparisons between our designs where total numbers of reads for each varied, we generated profiles normalised by the read depth at the start of the terminator such that drops due to termination corresponded to a fractional change (**Figure 5.10 A**). We also calculated Δ-values corresponding to the change in normalised read depth at each nucleotide position with respect to the previous nucleotide, enabling us

to pinpoint and compare changes more easily. We found that the maximum Δ-value for each design was proportional to its $T_e$ and amounted to approximately 40% of the total $T_e$ value. Each terminator maintained a predominant termination pattern (as shown by its Δ-profile), which varied in amplitude depending on the upstream modifier and spacer (**Figure 5.10 A**). The ability to observe these nucleotide resolution changes demonstrates a unique benefit of the pooled nanopore dRNA-seq approach over previous methods to characterise terminators.

The termination pattern is an important phenotype and we found that termination does not occur at a single nucleotide location; for each of our core terminators it occurred over several nucleotides. This could enable stochastic fluctuations in the cell. While the $T_e$ of a terminator did vary across genetic contexts, in general the termination pattern remained consistent. These patterns revealed that the rate of termination often fluctuates nucleotide by nucleotide, resulting in multiple drops in the Δ-profiles and therefore multiple transcript isoforms.

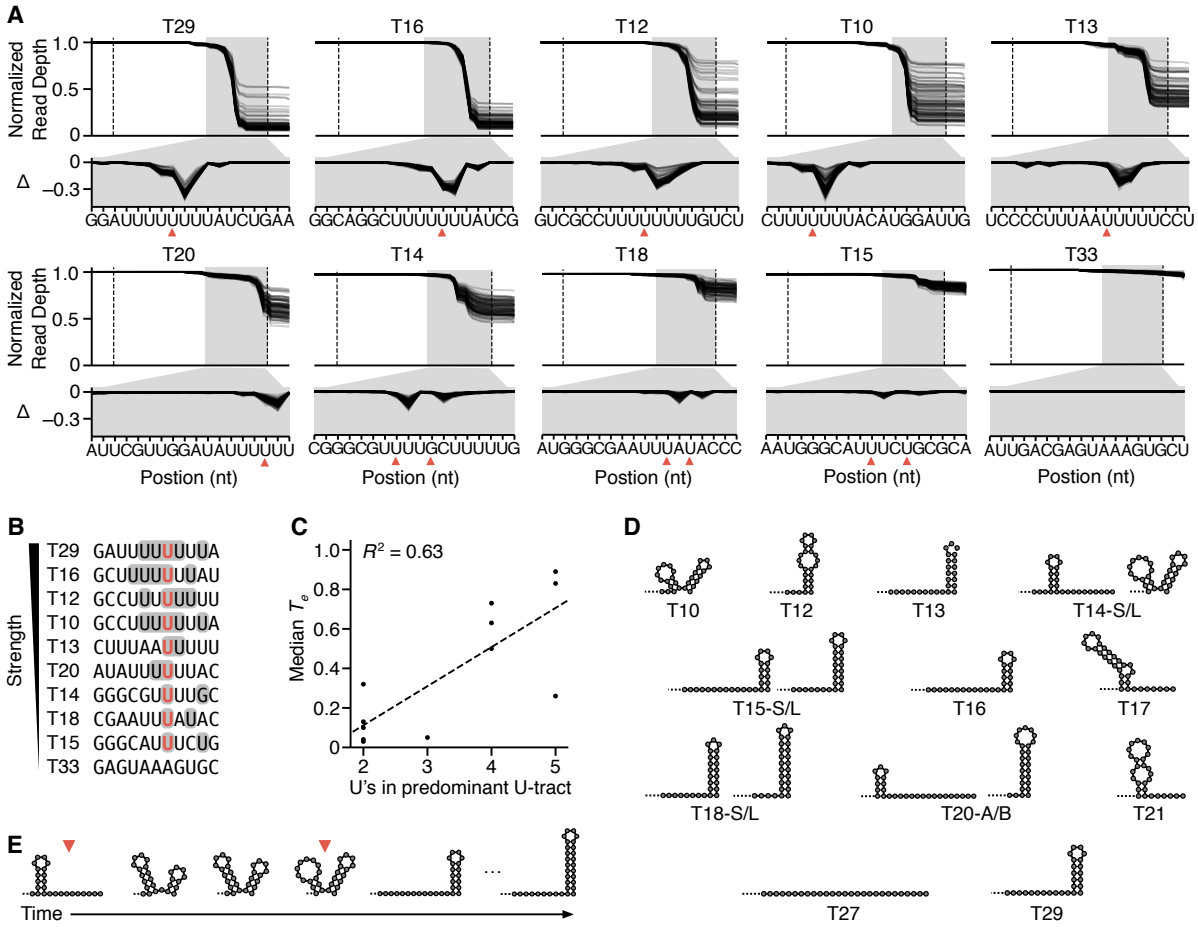



Figure 5.10: **Nucleotide resolution read depth profiles reveal terminator phenotypes.** (**A**) Normalised dRNA-seq read depth profiles for functioning terminators and non-terminator control (T33). Data for T17, T21 and T27 not shown as no termination observed. Each line corresponds to a design. Dotted lines denote the start and end of the core terminator genetic part. Red triangle indicates final nucleotide of the dominant point(s) of termination. Grey shaded region is expanded in the lower panel to show read depth changes (Δ). (**B**) Termination locations for functioning terminators. Many terminators terminate at more than one position and these points are indicated with shading. The most common point of termination is coloured red. (**C**) Median termination efficiency ($T_e$) for designs containing each core terminator compared to the number of U residues in the U-tract, which is the 8 nt sequence upstream of the point of termination. $R^2$ is the square of the Pearson correlation coefficient. (**D**) Secondary structure predictions of transcripts using a co-transcriptional folding simulation at the measured point of termination for each terminator (for inactive terminators, the sequence upstream of the U-tract with maximal number of U residues was used). Two structures are shown for T20; T20-A is formed first, and T20-B lies over a large energy threshold. For weak terminators, the structure prediction for both the shorter (S) and the longer (L) transcripts produced are shown. (**E**) Co-transcriptional mRNA secondary structures predicted for T14 as it is transcribed, with dominant points of termination indicated.

As expected, drops in read depth for each terminator occurred within the corresponding U-tract (**Figure 5.10 B**). We found that termination was possible with as few as 3 U's, but that maximum drops in the profiles occurred at a similar point (after 5 or 6 U's) for the stronger core terminators. This likely captures a position where a combination of optimal T7 RNAP pausing and weakened stability of the transcription elongation complex leads to formation of a hairpin within the core terminator that is sufficient to effectively facilitate termination. The number of U's in the U-tract at the point of maximal termination showed a correlation with $T_e$, although there were some outliers (T20, T14; **Figure 5.10 C**). This matches a previous finding for *E. coli* RNAP termination [47]. However, termination was found to always reach a peak with a U-tract that comprises fewer than the maximum possible number of U's in the U-tract.

Using this data, it was possible to predict RNA secondary structures at the various points of termination and assess their potential influence. To do this we simulated co-transcriptional folding [86] of the terminated sequence at the point of maximal termination (**Figure 5.10 D**), assuming a previously reported T7 RNAP transcription rate of 333 nt/s [86] (**Methods**). We removed the final 8 nt of the terminated transcript isoform since this area has been shown to base-pair with the DNA template in the T7 RNAP transcription elongation complex [255]. The co-transcriptional folding algorithm often gives multiple folding states for the RNA molecule and for simplicity, we chose to study the structure with the lowest folding energy. The strongest terminators were predicted to form terminating hairpins at their 3'-end, proximal to the elongating T7 RNAP. To the contrary, T20 was found to get locked in a secondary structure involving a hairpin ending 16 nt upstream of the T7 RNAP and this could be the reason that it remained a weak terminator despite having a long U-tract. The simulation predicted that in order to reach a terminating hairpin close to the U-tract, T20 would have to surpass a large energy barrier.

There were three inactive terminators (T17, T21, T27) and the negative control (T33). Each of these could have a maximum of 3 U's in the U-tract and none were predicted to form a hairpin proximal to the U-tract, which is likely the cause of their inactivity. This showed that T27, the reverse oriented phage terminator, was not able to efficiently terminate T7 RNAP bidirectionally. The three weakest active terminators (T14, T18 and T15) all had only 3 U's in their U-tract, terminated at multiple points separated by one or more nucleotides, and were predicted to form a hairpin proximal to the U-tract (**Figure 5.10 D**). It is not clear why T14 gives stronger termination than T15 and T18 though one element that might increase $T_e$ is the unique double hairpin structure that can form in the core terminator at the later points of termination. The second peak in termination for T14 coincides with the last point at which this double hairpin is predicted to exist (**Figure 5.10 E**). T15 and T18 on the other hand are predicted to form a single long hairpin structure.

An ability to engineer the precise point(s) of termination and therefore dominant 3'-UTR sequences may be important in deciding gene stoichiometries *in vivo* as it could potentially affect mRNA degradation rates, however, this is contested [44, 186]. We reviewed the possible

transcripts produced by the characterised terminators and showed that the final nucleotide of transcript isoforms can be either A, C, T or G. Frequently the dominant transcript terminated within a stretch of U's though less frequently observed transcripts were found to terminate immediately after a stretch of U's. Further investigation revealed that in a minority of cases, upstream sequence could tune the major point of termination. For example, some modifiers showed different stoichiometries of two types of transcript produced by T13.

## 5.7 General termination properties

$T_e$ was calculated from the transcriptional profiles and compared between genetic designs. Overall, $T_e$ varied from 0 to 0.94 across the library with core terminators displaying varying levels of $T_e$ and sensitivity to different modifier and spacer parts (**Figure 5.11**). Grouping designs by their core terminator showed that each had a unique median $T_e$ and that the variability differed between terminators (**Figure 5.11**). We found that some designs showed little to no termination (T17, T21, T27 and the negative control, T33 all had $T_e < 0.05$). The remaining nine terminators displayed a range of median termination efficiencies for T7 RNAP, which was heavily influenced by upstream sequence context. The variety of $T_e$ values would allow for a wide range of transcript isoform stoichiometries to be produced, from 11:1 to 1:1. We undertook an analysis of other general biophysical parameters that may play a role in termination: GC content and minimum free folding energy. However, none of these were correlated with measured $T_e$ (**Figure A.1**).

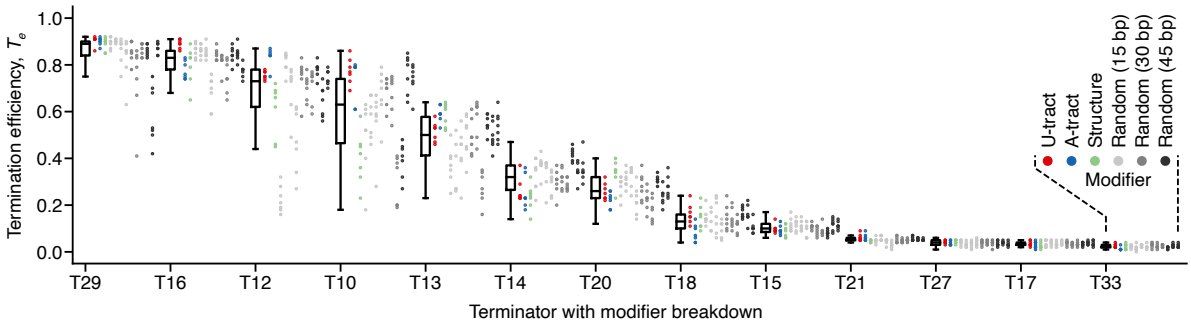



Figure 5.11: **Characterisation of a T7 RNA polymerase transcriptional library** Termination efficiency ($T_e$) for every design in the library. Each point denotes the $T_e$ valve for a unique genetic construct color coded by the modifier present. Points are grouped by core terminator with a box plot summarising the data for all associated constructs. Boxes represent inter-quartile range (IQR) at outer edges and median within; whiskers indicate 1.5 x IQR. Random sequence modifier parts: (left–right) M10–M22.

## 5.8 Discussion

By developing a nanopore-based dRNA-seq characterisation method (**Figure 5.1**), we were able to simultaneously measure the termination efficiency of an entire mixed pool of 1183 unique transcriptional terminators as well as provide nucleotide resolution insights into precisely where termination occurred for each (**Figure 5.10**). We found that all terminators produced multiple transcript isoforms, whose ratio could be tuned with upstream sequences. Such detail is lost with more typical fluorescence-based assays [33, 47], but is essential for developing the low-level biophysical models of genetic parts that can support predictive bio-design workflows [89, 146, 265].

Although dRNA-seq opens up new avenues for high-throughput characterisation of genetic part libraries, some challenges remain. The most prominent of these is ensuring the read depth profiles accurately represent the transcript variants present. Here, we show how some unwanted features caused during the preparation of a sequencing library can be effectively corrected for using a simple mathematical model (**Figure 5.4**). Improvements in the ability to sequence full length transcripts would lead to more usable sequencing reads and more representative read profiles, making it a valuable direction for future research. Improving nanopore sequencing accuracy will also be essential to distinguish designs with high similarity. Read accuracy for nanopore-based dRNA-seq is, at present, lower than for commonplace Illumina-based short-read RNA-seq (median read accuracy of 80–90% versus >99.9%, respectively [63]). This method relies on each design having a sufficiently different sequence for each read to be accurately mapped. Although this gap is closing with improvements to basecallers and sequencing chemistries, DNA library distributions, barcodes and analysis pipelines need to be carefully tuned and validated to ensure accurate demultiplexing of sequencing reads. Small differences in RNA sequence can be important during RNA processing, for example point mutants, which can determine co-transcriptional folding pathways which in turn can mediate co-transcriptional RNA processing [297]. Improvements to the method could facilitate the comprehensive exploration of large genetic design landscapes [42] and generation of algorithms to predict function from sequence [89].

Whilst this method could be used to characterise variants at multiple positions in genetic circuits, there are limitations. Drops in transcription at terminators across a genetic circuit mean that the whole genotype is not encoded on each of the transcribed RNAs. For example, a promoter followed by two terminators will produce some transcripts where the sequence of the final terminator is not encoded. Transcribed RNA with the same sequence for multiple genetic circuits would be impossible to demultiplex. To ensure demultiplexing of transcripts, each genotype would need a unique set of intrinsic barcodes or a unique molecular identifier (UMI) near to the promoter. This way, each individual barcode or UMI corresponds to only one transcriptional start site and each transcript could be matched to the DNA from which it was transcribed. This limitation means that libraries combinatorially assembled at multiple positions using our simple assembly method could not be demultiplexed unless tagged with a UMI. Instead

genetic circuits would have to be assembled or synthesised individually, increasing the cost. This poses challenges for understanding the effects of combinations of terminators upon one another. Libraries must be carefully designed to ensure that all transcripts can be distinguished if using dRNA-seq to characterise the encoded genetic parts.

RNA fragmentation meant that many sequencing reads did not contain an intrinsic barcode and therefore were useless. Fragmentation causes a significant reduction in the number of reads mapping to an intrinsic barcode (read recall of only 20% of the total reads with alignment to a barcode for dRNA-seq, compared to 70% for nanopore DNA-seq, where there is little fragmentation). Therefore, improvements in experimental protocols to reduce fragmentation/truncation or the incorporation of methods to enrich barcode containing reads (e.g., using "read until" technologies or sequence-specific dRNA-seq) could both improve the accuracy of $T_e$ calculations and increase the size of the libraries that can be assessed using a single sequencing run.

Consecutive intrinsic barcodes could increase or decrease read recall depending upon the library design. If there was a requirement that a read must contain all intrinsic barcodes in the genetic circuit then read recall would likely decrease further, by a further 80%, making only 4% of reads usable. However if each intrinsic barcode at each terminator position was unique to one genetic circuit in the library, read recall could be improved as there are more opportunities for fragmented RNAs to be matched to a design. Sequence-specific primers could also be used to target reads to the region of interest. This would necessitate a barcode which comprises *5' – UMI 1 – primer site 1 – terminator – primer site 2 – 3'*, allowing transcripts from all designs to be targeted with the same two primer sites. This strategy is amenable to both short-read and long-read dRNAseq and DNA-seq (after reverse-transcribing the RNA in to DNA and possibly amplifying it with PCR [105]) though it would suffer from bias from the molecular processes involved [52, 189, 199]. This shows how library and assay design must be compatible and selected on a case-by-case basis.

The focus of this work was to assess the function of transcriptional terminators *in vitro*. This allowed us to avoid other confounding factors that would be difficult to control for *in vivo* (e.g., RNA degradation [111]). However this is a limitation as the function of the genetic parts that we characterised is likely to differ *in vivo*. Therefore, a detailed assessment of how well these results hold or correlate to *in vivo* measurements would offer an important future direction. Our study could also be carried out in high-throughput using other sequencing based approaches that combine fluorescence activated cell sorting (FACS) and subsequent sequencing (Sort-seq) [43, 89] or targeted approaches based on the pull-down of specific RNAs. Using the DNA sequencing method presented in Chapter 4 is likely the best approach for measurement after FACS. Since FACS would involve growing cells with expression of the genetic circuits, long read sequencing offers the benefit of being able to identify the mutations that arise.

Whilst long read DNA-seq could enable genetic circuits with multiple terminators to be characterised with its improved read recall, it would be limited by the use of unique fluorescent

reporters. There is no panacea and the right method should be selected based on its suitability for taking the required measurement. The method presented in this chapter was appropriate for the task at hand because it could characterise RNA transcripts directly, demultiplex sequencing reads from combinatorial libraries and offer nucleotide resolution of termination. In Chapter 6 we use our insights into termination to design and characterise multiple libraries of terminators and elucidate design principles for tuning the flow of RNAP using terminators.

CHAPTER 6

# ENGINEERING TRANSCRIPTIONAL VALVES

## 6.1 Introduction

Recent findings have shown that transcriptional terminators rarely completely terminate transcription [149]. In genomes, incomplete termination in combination with internal promoters and RNase cleavage sites is used to diversify transcript isoforms [254]. Rather than an end point of a transcript, terminators can be considered as "valves" able to regulate the RNA polymerase (RNAP) flux passing through a point in DNA and thus the ratio of transcript isoforms that occur, this was shown by the variety of termination efficiencies measured in Chapter 5, which offers a new approach for synthetic biologists to regulate arrays of consecutive genes at the level of transcription.

An understanding of how a terminator sequence influences this function is required in order to use them to predictively design transcript isoforms. Genetic parts are known to vary with nearby genetic context [37]. This is the case for promoters, another type of transcriptional genetic part and it has been shown that short sequences can be used to insulate the promoter from nearby sequence context [39]. Ribosome binding sites on the other hand are best insulated by cleaving the upstream promoter sequence [167]. Insulating genetic parts from sequence context is important as it means that nearby sequences can be varied without impacting the function of genetic parts. This means that an insulated genetic part could be used to predictively regulate any gene. In the case of terminators, removal of the native genetic context generally reduces termination efficiency [33] though there is little understanding of how to design insulators for terminators. Since genetic context can change termination efficiency, it could be used to tune termination efficiency as well as insulate terminators. The mechanism of intrinsic terminators relies upon the formation of a base-paired hairpin sequence during co-transcriptional folding of

the nascent RNA molecule.

Sequencing can be used to test the function of genetic parts and in doing so it can guide the process of engineering genetic parts. Furthermore, multiplexed sequencing can accelerate the design-build-test-learn cycle approach to engineering, frequently used in synthetic biology, by enabling multiple sets of genetic parts to be tested in a single experiment [89]. The function of intrinsic terminators can be measured using nanopore direct RNA sequencing. High-throughput characterisation of terminators can quickly elucidate how genetic context influences termination. An understanding of termination of polymerases widely used in biotechnology (such as T7 RNAP) means that the parts developed can be used broadly across organisms in the future and also in *in vitro* cell-free systems [55].

In this chapter, we iteratively design and characterise several large libraries of transcriptional valves for T7 RNAP using the nanopore direct RNA sequencing (dRNA-seq) method developed in Chapter 5. Using this data, we are able to infer design principles which indicate how genetic context can be used to tune termination efficiency or insulate a terminator's performance from local variable sequences. In Section 6.2 we discuss the design of the valves encoded in library L2 assembled in Chapter 4 in detail. In Section 6.3 we study the capacity of designs in library L2 to tune and insulate valves. In Section 6.4 and 6.5 we study valve library L3 which explores base-pairing and structural interactions between the modifier and terminator to tune and insulate terminators respectively. In Section 6.6 we study library L4 which consists of core terminators with various modifications. In Section 6.7 we design, build and test an array of RNA parts (CRISPR guide RNAs) regulated by engineered valves *in vitro* and finally in Section 6.8 we discuss the insights in to transcriptional valves revealed by these results. In summary, we characterise and engineer transcriptional valves which can control RNAP flux and regulate arrays of genetic parts using transcript isoforms.

## 6.2 Designing transcriptional valves

To demonstrate how transcriptional valves might be built, we attempted to construct proof-of-concept designs for T7 RNAP. T7 RNAP was selected due to its broad use in synthetic biology, which stems from the fact that it is a single-subunit RNAP with high processivity, making it ideal for both *in vitro* use [234] as well as an orthogonal transcription system *in vivo* [164, 281]. While diverse terminators are available for the native E. coli RNAP [33, 47], for T7 RNAP only a single terminator exists in the T7 phage genome [121] and only a few alternatives have been characterised [172, 176, 179]. Furthermore, while termination of RNAP in model microorganisms like E. coli and S. cerevisiae has been extensively studied [212], few intrinsic terminators beyond those in the T7 phage genome have been characterised.

Terminator features, such as hairpin structure and U-tract composition, can strongly influence termination efficiency [47, 126]. Therefore, as a basis for an initial library of transcriptional

valves, we chose 13 different intrinsic terminators to act as "core terminator" elements (T). We began by selecting the single terminator from the T7 phage genome (T27) [121], which has previously been characterised *in vitro* [179]. To test its possible bidirectionality, a feature that terminators for other RNAPs have been shown to exhibit [47], it was included in a reverse orientation in our designs [179]. Beyond the native T7 phage terminator, E. coli terminators present another source of these parts and have been shown to terminate T7 RNAP *in vitro* [46, 179]. Therefore, 11 intrinsic rho-independent terminators were selected from the E. coli genome spanning a wide range of termination efficiencies *in vivo* [47] along with a negative control terminator (T33).

The sequence upstream of a terminator-hairpin also influences termination [33, 159] and can tune the termination efficiency of a valve. 13 different "Modifier" parts (M) were characterised in our transcriptional valve design designed to interact with canonical regions of a terminator hairpin sequence (**Figure 6.1 A**). Modifiers M10 and M11 were designed to interact with possible U- and A-tracts within a terminator by containing complementary homopolymers of adenine or uracil, respectively [47]. A further modifier M12 containing a small RNA secondary structure was designed with the goal of affecting RNA structure formation near the terminator part. Beyond tuning termination efficiency with RNA interactions and structures, it has been shown that inert random sequences can play an insulating role, provided they are long enough [39, 159]. Insulating terminators from upstream genetic context could improve their robustness when used in different genetic contexts. Upstream genetic context of intrinsic terminators influences termination in a distant-dependent manner [33, 159]. Thus, we decided to include a selection of modifiers of different lengths (M13–M16: 15 bp, M17–M19: 30 bp and M20–M22: 45 bp) (**Figure 6.1 B**) where each was a random non-coding sequence.

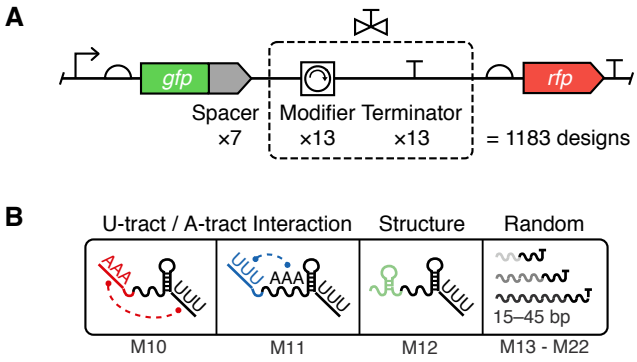



Figure 6.1: **Structure of the transcriptional valve library L2. (A)** Combinatorial library design. **(B)** Modifier variants.

To assess the robustness of each valves' termination efficiency to local upstream genetic context, our library also included "spacer" elements (S) (**Figure 6.1**). These did not form part of the transcriptional valve, but instead allowed us to see how a particular valve might behave when used in combination with other components (e.g., coding regions). Using the NullSeq tool [165], we generated 7 random and genetically diverse 33 bp long spacers with a nucleotide composition similar to coding regions of E. coli that could be placed at the 5' end of a valve. Each spacer had a stop codon "TAA" at its 3'-end, though this was not utilised in our *in vitro* transcription assay. The designed spacers, modifiers and core terminators could be combinatorially assembled to create a library, L2, of 1183 unique designs able to regulate RNAP flux and provide valuable information regarding the design principles of transcriptional valves.

## 6.3 Tuning and insulating transcriptional valves

### 6.3.1 Tuning the strength of transcriptional valves

We were interested to know if patterns within the initial library design might offer insight into the capacity of each core terminator to be tuned or insulated by the upstream modifier sequence. For example, designs displaying a wide range of $T_e$ values for the same core terminator would indicate that the terminator is highly tunable, while a small range of $T_e$ for a design used with differing spacer elements would suggest that it is able to insulate its function from upstream sequence context.

It is known that local sequence context can be used to effectively alter the function of many types of genetic part [97, 99, 167, 196, 206]. We therefore designed modifiers in our initial library with the aim of being able to tune the strength of a valve. In Chapter 5 we showed that changes in upstream genetic context (both spacer and particularly modifier sequences) could significantly influence termination strength, allowing $T_e$ to be varied over a range of up to 0.68. The ability to tune each core terminator varied, with the $T_e$ of T10 being most tunable and T16 being least. The capacity to tune terminator strength could arise from the diversity of co-transcriptional structures that form proximal to the U-tract when interacting with the modifier (**Figure 5.10 D**). Therefore, to create a library of highly tuned transcriptional valves, it is important to ensure the core terminators are themselves tunable.

Large variability in the magnitude of tuning was seen across the different valves we tested suggesting that sequence specific features play a key role in modulating the precise termination efficiency. Spacers were found to not have such a systematic effect. Nonetheless, some valves were highly influenced by spacers, emphasising the importance of upstream sequence in the region 100 nt upstream of the point of termination. For designs grouped by spacer, the median percentage deviation from the median $T_e$ of the valve they contained was found to be less than 5% and this is reflected by a median deviation across all spacers of 0% (**Figure 6.2 A**, Spacer). Therefore tuning of termination efficiency is best achieved by varying sequence context close to

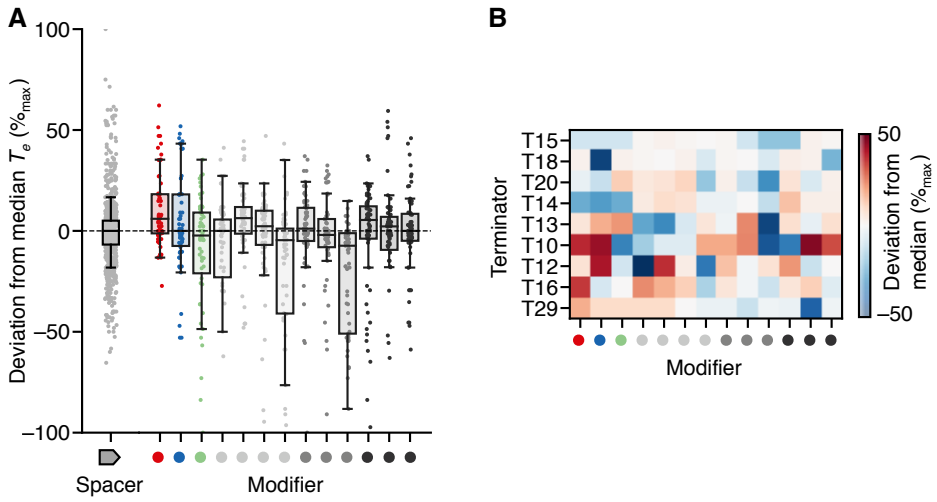



Figure 6.2: **Investigating the ability to tune terminators using upstream sequence** (**A**) Percentage deviation in $T_e$ (as a percentage of maximum possible deviation) for each construct from the median of all constructs containing the same core terminator. Each point corresponds to a single construct and points are grouped by modifier. Only data for active terminators (median $T_e > 0.05$; T29, T16, T12, T10, T13, T14, T20, T18, T15) are shown. Boxes represent inter-quartile range (IQR) at outer edges and median within; whiskers indicate 1.5 x IQR. (**B**) Terminator and modifier breakdown of the percentage deviation in $T_e$ (as a percentage of maximum possible deviation) for active terminators. Random sequence modifier parts in all plots: (left–right) M10–M22, colour-coded as for Figure 6.1.

the core terminator part: the modifier region in our case.

In general, each modifier tuned each terminator in a different way. However, some modifiers were found to have a similar tuning effect across many different core terminators (**Figure 6.2 A, B**). Some had a generally positive influence (e.g., M14, M20) or negative influence (e.g., M16, M19). Furthermore, the U- and A-tract interactors generally exerted opposite tuning effects on stronger terminators and weaker terminators, tuning them up and down in strength, respectively. Therefore, when tuning a T7 RNAP terminator, while bespoke modifiers are likely required, our library offers some starting points for sequence features that are likely to have a desired effect.

### 6.3.2 Insulating transcriptional valves from local genetic context

Our library specifically included random non-coding modifiers of varying length to assess the insulating effects for transcriptional valves. In general, we found that an increase in the length of these modifiers led to a reduction in $T_e$ variability when an identical valve design was used across numerous genetic contexts (i.e., upstream spacer sequences; (**Figure 6.3 A**). This matches findings for bacterial promoters and terminators where longer upstream insulating sequences resulted in more predictable gene expression [39, 159]. However it is the first time this has been

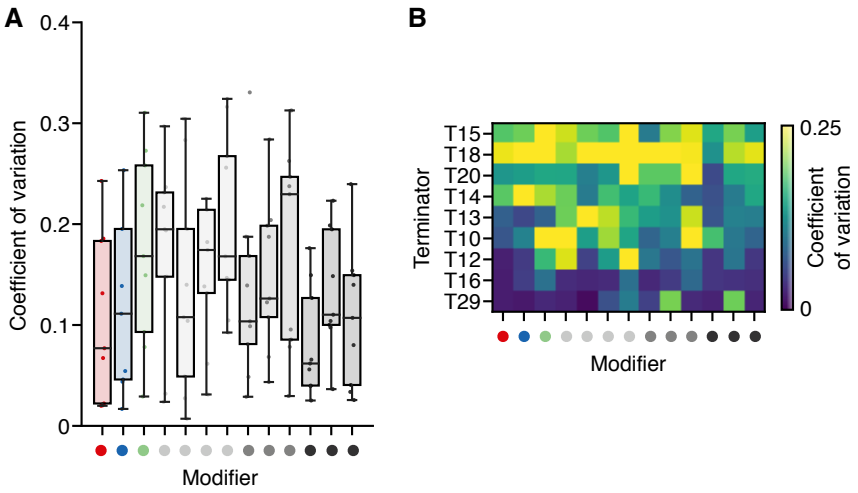



Figure 6.3: **Investigating the ability to insulate terminators using upstream sequence** (**A**) Terminator and modifier breakdown of the CV of $T_e$ values across spacer variants for active terminators. Boxes represent inter-quartile range (IQR) at outer edges and median within; whiskers indicate 1.5 x IQR. (**B**) Coefficient of variation (CV) of $T_e$ values across spacer variants for active terminators grouped by modifier. Random sequence modifier parts in all plots: (left–right) M10–M22, colour-coded as for Figure 6.1.

shown with T7 RNAP and *in vitro*. Notably, these effects were also terminator specific, with some core terminators showing more predictable behaviour across modifiers (e.g., T16) than others (e.g., T10) (**Figure 6.3 B**). This suggests that some terminators are better suited to tuning T7 RNAP *in vitro*, while others are better placed to maintain a consistent termination efficiency.

## 6.4 Exploring modifier-terminator base-pairing

It was evident from this initial library that the general modifiers comprising random sequences we had used limited our ability to understand the role of key interactions between the modifier and terminator parts because no terminator specific interactions had been designed. To rectify this, a further library was built to understand the effect of the modifier region upstream of the core terminator in a more comprehensive way (**Figure 6.4 A**). Informed by our findings that longer modifiers were better insulators of $T_e$, we designed all new modifiers as length 45 nt to enhance the robustness of the valves function across different genetic contexts. Co-transcriptional folding simulations had highlighted that the sequences we had designed to interact with the U-tract and A-tract were insufficient. These modifiers seldom formed structures that would influence termination by virtue of base-pairing because the A-tract is often short (<6 consecutive A's) and at the point of termination, the U-tract is sequestered by the RNAP. Therefore, we designed modifier sequences that would target specific sequences within three strong core terminators (T29, T16, T10).

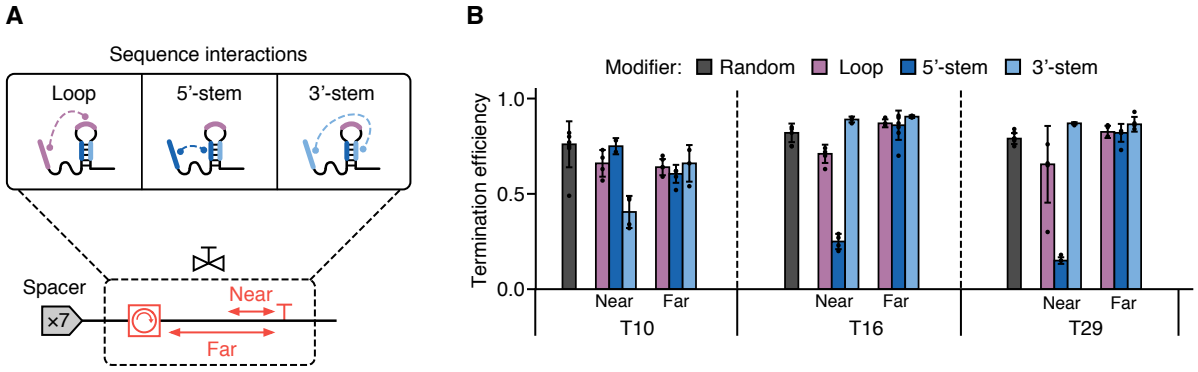



Figure 6.4: **Engineering modifiers that tune core terminators** (**A**) Overview of the modifier library based on sequence interactions used to explore tuning of core terminator function (part of library L3). A "near" and "far" modifier variant was designed for each motif (except for the random option). 8 nt sequences were designed to be complementary to regions of the core terminator covering the loop (purple), 5'-stem (dark blue), 3'-stem (light blue). **B** Median termination efficiency ($T_e$) for each valve designed to interact via sequence, grouped by modifier (coloured) and by terminator (T10, T16, T29). Error bars denote the standard deviation in all plots.

Motifs containing an 8 nt reverse complement sequence of three different regions of the core terminators were designed into modifiers. Gaps were filled with non-structural RNA sequences (**Methods**). While ideally we would have used identical padding sequences, we instead chose padding sequences with identical RNA secondary structure (i.e., no predicted structure) as these unique sequences would help ensure accurate read demultiplexing after errors accumulated during nanopore basecalling. These motifs targeted the 5'-stem, loop and 3'-stem regions of the terminator hairpin (**Figure 6.4 A**). Two variants of each motif were designed to explore the distance dependence of the engineered motifs: "near" which was incorporated into the modifier at its 3'-end, and "far" which was incorporated at the 5'-end of the modifier ( 70 nt from the U-tract).

Characterisation of this library revealed that base-pairing can significantly reduce termination efficiency in a distance-dependent manner (**Figure 6.4 B**). The motifs designed to base-pair with the loop consistently caused a reduction (13–16%) in $T_e$. Loop-modifier interaction during termination could alter the core terminator hairpin during its formation, at the point of termination, or both, affecting the number of termination events. The largest effect on $T_e$ was caused by motifs designed to interact with the stem of the terminator. We hypothesise that this is because core terminator stem base-pairing is essential for hairpin formation and even drives ratcheting of the U-tract off the DNA template, whereas the loop can base-pair with an interacting motif at the same time as the completing hairpin. Therefore, a motif that can base-pair with the stem could outcompete the core terminator hairpin.

For the two strongest terminators (T29, T16) the 5'-stem interactor caused a large drop in $T_e$, while the 3'-stem interactor did not. In the case of T16 this is likely because 5 of the 8 targeted nucleotides are predicted to be concealed within the T7 RNAP (2 nt for T29 and 3 nt for T10),

where they cannot base-pair at the point of termination since they are in the U-tract. This effect on $T_e$ was greater than any other drop caused by a modifier tested so far. As with previous modifiers, T10 behaved differently to T16 and T29. The 3'-stem interactor had a large effect on termination, while the 5'-stem interactor did not. This could be a consequence of the extra native sequence context between the hairpin and the motifs (5 nt and 3 nt more than T29 and T16, respectively) resulting in a location in which the motif can base-pair with the 3'-stem. However, it may also arise from the inherent tunability of T10. In either case, the effect on $T_e$ of motifs that base-pair with terminators diminishes when they are placed far from the terminator.

## 6.5 Exploring structural interactions

We next investigated whether structure could insulate terminator function by designing and testing a library of modifiers containing secondary structure motifs (**Figure 6.5 A**). To explore the effect of upstream structure on termination, we designed three short (stem length 3 nt) and three long (stem length 6 nt) hairpins. The sets of short and long hairpins contained one of three loops (UUCG, GAAA, GAGA) known to facilitate strong hairpin formation [16]. Furthermore, modifiers were designed with all of these secondary structures at both the 5'-end and 3'-end to test the distance-dependence of secondary structure influence on termination efficiency. Again, gaps were filled with non-structural RNA sequences (**Methods**). A variety of other more complex RNA secondary structures are known and one modifier containing an "elbow" (the TAR element) and one containing a pseudoknot [303] were also designed to see any role these might play in modulating termination.

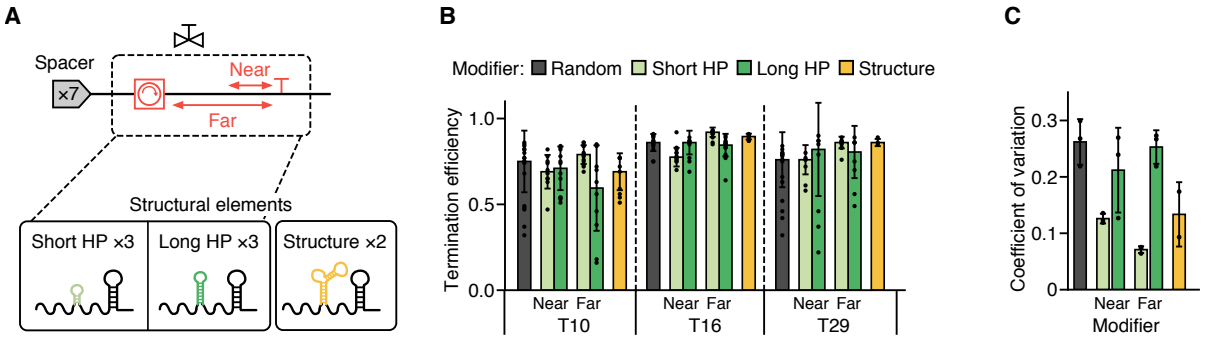



Figure 6.5: **Engineering modifiers that insulate core terminators** (**A**) Overview of the modifier library based on structural elements used to explore insulation of core terminator function (part of library L3). A "near" and "far" modifier variant was designed for each motif (except for the random and structure options). For structural elements, 3 short hairpins (light green), 3 long hairpins (dark green) and 2 further RNA structures (orange) were designed. **B** Median $T_e$ for each designed structural element. (C) Coefficient of variation of $T_e$ values for the structural elements. CV is calculated across spacers for each terminator and grouped by structural element. Error bars denote the standard deviation in all plots.

After assembling and characterising this new library, we were able to confirm that RNA secondary structure upstream of terminators affected the robustness of terminator function with T7 RNAP (**Figure 6.5 B**). We found that short hairpin structures and complex RNA structures were the best insulators of terminator function (**Figure 6.5 C**), while long hairpin structures made termination efficiency more sensitive to upstream genetic context. The rigid requirements of short hairpin formation mean that they are likely rarely influenced by base-pairing with upstream structure. This would explain why they are good insulators since they offer dependable upstream secondary structure that does not base-pair or interact structurally with the terminator hairpin. In contrast, since long hairpins can form with a variety of stem lengths they could influence and be influenced by neighbouring sequences. The resultant diversity of secondary structures that can then arise upstream of the terminator hairpin would mean that these modifiers significantly affect $T_e$ and therefore act as poor insulators, as seen in our results. Since we only tested strong terminators in library L3, conclusions cannot be drawn on the capacity of base-pairing and structural sequence motifs to increase $T_e$. Nonetheless, these results revealed motifs that could alter the $T_e$ or robustness of terminator function and therefore should be avoided when designing genetic circuits that involve uncharacterised gene-terminator combinations.

## 6.6 Understanding core terminator design principles

Our results had shown that many of the relationships observed were terminator dependent, and so a final library (L4) was designed and tested to investigate variations of the core terminator part that had the greatest influence on $T_e$ (**Figure 6.6 A**). We constructed a library of designs including U-tract enrichment to include up to 8 consecutive U residues down-stream of the core terminator hairpin; context switch where the 19–25 nt immediately upstream of the native terminator hairpin (before the modifier) was changed to a random sequence; U-tract switch to consecutive A, C, or G residues and finally other source organisms used to find diverse terminator sequences. This set of core terminators was assembled and tested in just one upstream genetic context (the GFP gene; no spacers or modifiers) and so our analysis is focused on understanding the influence of sequence context proximal to the core terminator hairpin.

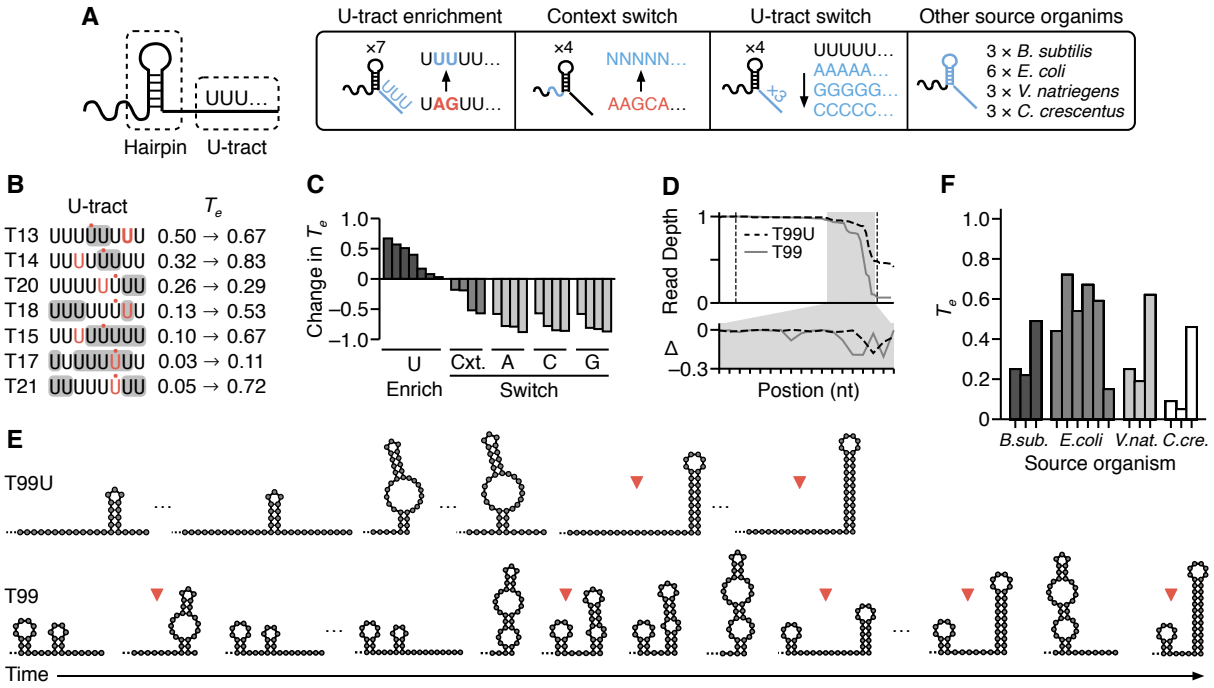



Figure 6.6: **Exploring design features of the core terminator.** (**A**) Overview of modifications made to core terminators. (**B**) Effect of U-tract enrichment on most common termination position and change in termination efficiency ($T_e$). Substitution of nucleotides for U are indicated with grey shading. Termination position before and after substitution are indicated in bold red text and red dot respectively. (**C**) Change in $T_e$ for U-tract enrichment (U Enrich, dark grey), context switch (Cxt, grey) and U-tract switches (A, C, G, light grey). (**D**) Normalised dRNA-seq read depth profiles for the T7 phage T-theta terminator (T99, black dashed line) and a variant (T99U, grey solid line). Dotted lines denote the start and end of the core terminator. Grey shaded region is expanded in the lower panel to show read depth changes (Δ). (**E**) Secondary structures predicted by co-transcriptional folding simulation of terminator formation (left to right) for T99U and T99. Positions where measured termination occurs are indicated with red triangles. (**F**) Measured $T_e$ of T7 RNAP for terminators sourced from diverse organisms.

Analysis showed that the $T_e$ of weak core terminators used in our initial library could be increased by increasing the number of U's in the U-tract (**Figure 6.6 B**). Our initial results indicated that a U-tract of at least 5 U's consistently resulted in termination (**Figure 5.10 C**). Therefore, we re-engineered weaker core terminators so that they contained a U-tract of length 8 nt. This increased $T_e$ to varying extents (**Figure 6.6 B**). Despite each of these newly designed terminators having the possibility to terminate at a U-tract complete with 8 U's, the dominant transcript isoform had only 5 or 6 U's in each case. Nonetheless, the increase in $T_e$ showed some correlation with the number of extra U's in the dominant U-tract (**Figure 6.7**). For active terminators, the largest increases in termination occurred when additional U's increased the number of U's in the dominant transcript isoform to 5 or 6 (T14, T15, T18). U's added downstream of the point of termination had little effect (T20), or when they did, did not increase the number of

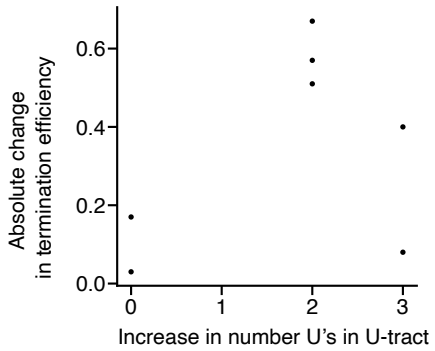



Figure 6.7: **Effect of U-tract changes on termination efficiency.** Scatter plot showing how the absolute change in termination efficiency increases with an increasing number of U's in the U-tract. Each point corresponds to an individual terminator.

U's in the U-tract (T13). Finally, in two cases (T17 and T21), termination of inactive terminators was found to be rescued.

The location of termination and thus the specific transcript produced consistently changed following these modifications (**Figure 6.6 B**). For these designs, transcript isoforms became either 1–2 nt shorter or longer. The location of termination and termination prior to a complete "UUUUUUUU" U-tract may arise since the extra U's were not added immediately upstream of the dominant point of termination. Instead, they were added at positions that would give a sequence of 8 U's with the minimal number of single nucleotide substitutions. Therefore, to generate strong terminators from a template sequence, our results suggest first characterising the dominant transcript isoform and then increasing the number of U's within its U-tract.

To complement the modifier library, the same core terminators (T10, T16, T29) were used as a basis for various U-tracts containing no U's. Variants of each of these core terminators with an 8 nt tract of A, C or G were designed. To ensure their transcripts could be distinguished after dRNA-seq we put unique non-structural RNA sequences as barcodes upstream of the core terminators (**Methods**). These variants were inspired by data showing that T7 RNAP can slip and terminate at sites of 8 consecutive A's *in vitro* [191]. However, we found this not to be the case for T29, T16, T10 or the T7 phage T-theta terminator (**Figure 6.6 C**). The poly-C tract showed very weak termination ($T_e < 0.1$). For our previous designs, some native sequence context immediately upstream of the terminator hairpin (and before the modifier) is retained. Changing this decreased $T_e$ for all the terminators studied (**Figure 6.6 C**), indicating that to maintain $T_e$, there is an optimal position upstream of core terminators to add modifiers.

Characterisation of the T7 phage terminator (T-theta) revealed wide diversity in the points of termination (**Figure 6.6 D**). We found T-theta to be strong (median $T_e = 0.82$) and tunable. The progression of minimum free energy structures predicted to form as each nucleotide is transcribed suggests that a variety of structures form approaching the point of maximum

termination (**Figure 6.6 E**). This is likely to account for the ability for termination to occur at various positions. Changing the native context immediately upstream of the core terminator hairpin decreased $T_e$ by 59 %. This was despite a change to the "GC" at the end of the U-tract to "UU". Furthermore, these modifications to the terminator changed the distribution of transcript isoforms significantly, resulting in a single peak of termination (**Figure 6.6 D**). This native context variant changes the co-transcriptional structures predicted in the build-up to the point of maximal termination, preventing a hairpin immediately upstream of the U-tract from forming for the first two possible transcript isoforms. These insights into how upstream genetic context influences terminator hairpin structures are potentially important for not only tuning $T_e$, but also transcript isoform abundances.

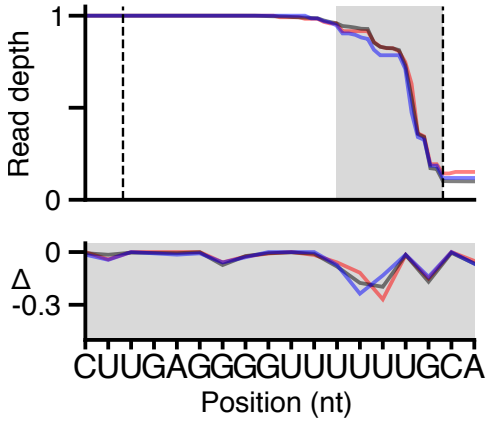



Figure 6.8: **Termination profiles of valve M81-T99 with different spacers.** Normalised dRNA-seq read depth profiles for the T7 phage T-theta terminator (T99) with modifier M81 and three different upstream spacers (S10 in red, S19 in black and S20 in blue). Dotted lines denote the start and end of the core terminator. Grey shaded region is expanded in the lower panel to show read depth changes (Δ).

T-theta was also tested with a variety of upstream modifiers designed to base-pair with the core terminator or form secondary structures. Of these, one short hairpin near to the core terminator (M81) made small changes to the ratios of transcript isoforms depending on the upstream genetic context (**Figure 6.8**). Co-transcriptional simulations of this valve indicated that a variety of secondary structures can form immediately upstream of the terminating hairpin, which can extend and therefore be influenced by sequences further upstream (**Figure 6.6 E**). The effect of versatile secondary structures upstream of the valve could potentially influence transcript isoforms as well as $T_e$. The ability for T-theta to form a complex mixture of transcript isoforms whose ratio can be tuned by upstream sequence could arise from co-evolution of this terminator with the T7 RNA polymerase. This would result in a high capacity for tuning both transcription (via $T_e$) and mRNA stability (via RNA degradation) following mutation of the core terminator or upstream sequence.

Finally, strong terminators highlighted by previous studies were also characterised (**Figure 6.6 F**). These comprised a set of three strong core terminators from four different bacteria characterised by Lalanne *et al.* [149], along with 3 further *E. coli* terminators with long U-tracts [47]. At least one example of a terminator with $T_e > 0.5$ was present in the selection for each organism. This selection sought to expand the options for engineering strong T7 RNAP valves. While each of these terminators have evolved to function in different cellular contexts, we found that they behave similarly with T7 RNAP *in vitro*: termination invariably occurred in a region with multiple U's in the U-tract. These results highlight that terminators sourced from many organisms can terminate T7 RNAP and provide yet more options for core terminator parts when designing valves.

## 6.7 Controlling expression stoichiometry of a CRISPR guide RNA array

The ability for our valves to control the stoichiometry of transcript isoforms makes them ideally suited for multiplexed regulation of RNA-based parts. To demonstrate how this might be achieved, we chose to focus on the expression of a CRISPR-Cas9 guide RNA (gRNA) array. While gRNAs have been co-expressed as arrays [34, 169, 181, 223, 243], few efforts have been made to rationally regulate the relative levels of gRNAs within an array. This could be important for implementing complex patterns of gene activation or repression. Promoters of varying strength have been used to achieve a similar goal [79]. However, promoters do not couple gRNA stoichiometries to one another in the same way as can be achieved by using transcriptional valves and are sensitive to noise and genetic context that can affect each promoter independently.

To test whether the characterised valves could be used to predictively regulate transcript isoform abundance we designed four arrays regulated by valves. These arrays were designed such that they could be characterised in a multiplexed assay. The experimental and computational methods developed in chapter 5 enabled characterisation of the transcriptional function encoded in all arrays in a single sequencing run. Each array contained the same three gRNAs (complete with handles, a short sequence that binds the Cas9 protein [223]) separated by two unique valves (**Figure 6.9 A; Methods**). A set of valves were selected from the initial valve library to give a range of gRNA expression stoichiometries and yet have distinguishable intrinsic barcode sequences. We pooled the arrays, used T7 RNAP to transcribe the pool *in vitro* and then performed dRNA-seq characterisation to calculate the ratios of expressed gRNAs from each array.

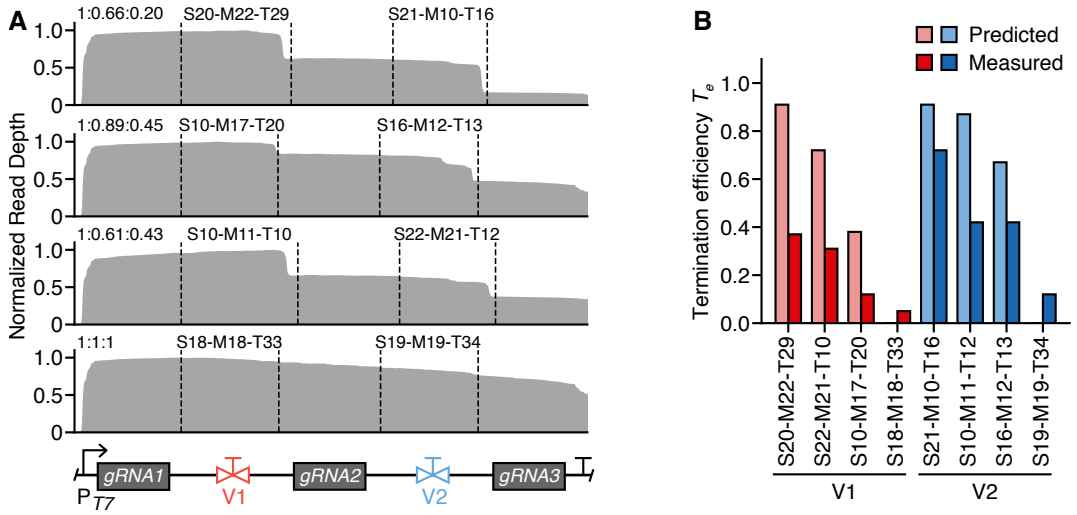



Figure 6.9: **Using transcriptional valves to regulate an array of CRISPR sgRNAs.** (**A**) Normalised dRNA-seq read depth profile for each array. Dotted lines denote the start and end of each valve. General array design is shown at the bottom, with the $P_{T7}$ promoter followed by 3 guide RNAs (gRNAs) separated by two transcriptional valves (V1, V2). Resulting gRNA stoichiometries are indicated in top left of each profile (gRNA1:gRNA2:gRNA3) and valve designs are shown above their respective parts. (**B**) Comparison of predicted (lighter shading) and measured (darker shading) termination efficiency ($T_e$) for each valve in the arrays. Bars are colored by valve position.

We found that as expected each design produced different stoichiometries of the gRNAs (**Figure 6.9 A**). We calculated predicted ratios based on the characterisation of the valve library and compared those to the measured ratios from the arrays (**Figure 6.9 B**). While valves ranked the same in terms of $T_e$, the absolute termination observed was significantly lower in the array, with a decrease that correlated with proximity to the promoter. This feature has been previously observed [295], though to our knowledge the cause is not fully understood. One hypothesis is that proximity to the promoter has been predicted to increase transcriptional read through of protein "roadblocks" by virtue of an increased force from RNAP traffic, which is cumulative [70, 71], and a similar effect could be occurring in our case.

To test this hypothesis further, we characterised the small library (L4) of 45 terminators using varying concentrations of T7 RNAP for the *in vitro* transcription reactions to vary the RNAP traffic present on the DNA (1X and 0.2X concentrations). We found a strong correlation ($R^2$ = 0.93) in the $T_e$ values. However, some valves did show significant decreases in $T_e$ for the higher concentration of T7 RNAP (**Figure 6.10**). Therefore, the systematic reduction in $T_e$ for the CRISPR gRNA arrays, may be a result of the much closer position of the promoter to terminator in these constructs. Nevertheless, characterisation of the arrays demonstrate the ability to use transcriptional valves as a means of multiplexed regulation of RNA-based parts.

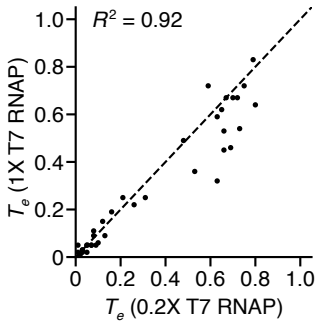



Figure 6.10: **Valve behaviour at different T7 RNA polymerase (RNAP) concentrations.** Comparison of $T_e$ measurements for *in vitro* transcription reactions performed with varying concentrations of T7 RNAP. Each point corresponds to a transcriptional valve. $R^2$ is the square of the Pearson correlation coefficient.

## 6.8 Discussion

In this chapter, we have shown how transcriptional terminators can be considered as "valves" to regulate the flow of RNAP along DNA and control the ratio of transcript isoforms produced.

While rich, high-content characterisation data can normally only be produced for a small set of terminators [73, 96, 103, 206], the approach presented here circumvents this common limitation and allows us to more systematically explore the genetic design space of a large pooled library and extract several design principles. We show how local sequence context (i.e., modifier sub-sequences) can be used to tune termination efficiency, while the inclusion of sufficiently long insulating sequences (45 nt) at the 5'-end of a core terminator reduce changes in $T_e$ when the same valve is used in conjunction with different upstream spacer sequences (**Figure 6.3 A**). Furthermore, the successful use of terminators from divergent bacteria to control the viral T7 RNAP suggests that a similar characterisation approach could be used to rapidly develop libraries of transcriptional valves for RNAPs from other organisms [5, 292].

Iterative design-build-test-learn cycles using rapid, combinatorial DNA assembly and *in vitro* dRNA-seq enabled us to construct two targeted libraries covering a further 600 designs to investigate properties of sequences near to the terminator that influence $T_e$ (**Figure 6.4, Figure 6.5 and Figure 6.7**). Structure of the sequence upstream of the terminator strongly influences $T_e$ via interference with terminator hairpin formation. Short hairpins within the modifier sequence can insulate terminators by stabilising upstream RNA structure though this ability diminishes with hairpin size (**Figure 6.7 C**). Downstream of the terminator hairpin, in the U-tract, the sequence determines whether RNAP pauses are sufficiently long to allow for hairpin formation and transcript dissociation. Here, an abundance of U residues was found to be essential, and their composition determined the precise points of termination. Further libraries could be characterised using our method, focussing on the effect of sequences downstream of

the terminator, other genetic parts on termination or even genetic parts with entirely different functions [303].

Our newly designed valves behaved similarly at differing T7 RNAP concentrations (simulating varying transcription initiation rates) and could regulate ratios of CRISPR gRNAs by expressing them in an array interspersed with valves (**Figure 6.9 A**). In the arrays, the valves consistently showed a reduction in termination efficiency, in a distance-dependent manner (**Figure 6.9 B**). Furthermore, testing at a decreased T7 RNAP concentration resulted in an increase in termination efficiency for some valves. Taken together, these results suggest that increased T7 RNAP traffic, whether caused by absolute T7 RNAP concentration or proximity to the promoter, may cause terminator read through and a decrease in termination *in vitro*. Therefore, for predictive use of valves, RNAP traffic should be taken into consideration and could offer a mechanism for dynamic control of circuit behaviours in response to cellular processes engineered to regulate RNAP concentration or upstream sequence length [141].

This work views transcriptional terminators in a new light. Not merely as a hard end point when producing a transcript, but as a means to tune and orchestrate one of the many flows (e.g., transcription and translation) that underpin the synthesis of proteins from DNA [254]. Nature is known to regulate gene expression at multiple levels and through numerous processes to create complex regulatory programs [14]. Transcriptional valves offer bioengineers a new perspective on how multi-gene regulation can be implemented at a purely transcriptional level and a means to implement more diverse information flows in genetic circuitry.

CHAPTER 7

# RESPONSIBLE SYNTHETIC BIOLOGY RESEARCH AND INNOVATION

## 7.1 Introduction

Developing a new technology is similar to dropping a pebble into a pond: early decisions determine the resulting ripples. This is because technologies organise the living world by virtue of the politics embedded within them [290]. Energy production is a clear example of how technologies can enable different organisational structures of people and resources: nuclear and solar power can lead to centralised and decentralised politics, respectively [290]. Whilst easy to overlook, technologies influence people [290], economies [138], ecologies [225] and are in turn influenced by them (**Figure 2.15**).

Deliberation at the early stages of research or technology development could have beneficial effects for future users. Regulation of new technologies generally lags behind innovation [200], however, technologists could apply self-regulation from the outset. The UK Engineering and Physical Sciences Research Council (EPSRC) endorses the AREA framework (**Figure 2.16**) for responsible innovation. This framework encourages four activities: Anticipating impacts (social, economic, environmental or otherwise), Reflecting upon the research, Engaging broader deliberation inclusively, and Acting to influence the direction of the research and innovation [202]. In this chapter the potential consequences of the technology developed in this PhD are anticipated and reflected upon, this may in turn lead to broader deliberation and action.

Does engineering genetic parts carry a responsibility? Engineered DNA and microorganisms may have impacts upon all living things given that they can end up within microbiomes in ecosystems [205]. van Bruggen *et al.* argue that *"the health of all organisms in an ecosystem are interconnected and mediated through the cycling of subsets of microbial communities from the environment (in particular the soil) to plants, animals and humans, and back into the*

*environment"* [267]. They define health as the absence of disease and suggest that in the case of microbiomes, health arises from microbiome stability, resilience, diversity, connectedness and integrity of nutrient cycles and energy flows [267]. Whilst the potential impacts of engineered microorganisms and DNA are not considered, the ability of DNA to travel through microbiomes via horizontal gene transfer [279] makes this possible.

Given the speed of developments in synthetic biology research and innovation [187], regulation will likely lag behind [200]. The diversity of technologies developed in synthetic biology means that generalisations of the potential impacts or how they can be innovated responsibly cannot be made, meaning that each should be considered individually. The responsibility of researchers to consider how to research and innovate responsibly is contested, with some arguing for [201] and against [259]. In any case, biological scientists, engineers and technologists are learning how to innovate responsibly [204] though appropriate and accessible concepts and tools to do so are not always available.

Philosophers of technology propose that the concept of "conviviality" can be used to guide the development of tools [117, 274]. But what does that mean? In the book Tools for Conviviality, Ivan Illich stated *"After many doubts, and against the advice of friends whom I respect, I have chosen "convivial" as a technical term to designate a modern society of responsibly limited tools."* [117]. An underlying concept is that of "appropriate technology", which proposes use of locally adapted materials and technologies that can be built, maintained and repaired without foreign experts [274]. Whilst this concept omits details of who owns such technologies (local enterprises, global companies or the people using them), there are commonalities: decentralisation, some autonomy from hierarchical infrastructures, scalability and technologies which don't harm the environment [274].

An important consideration for notions of both convivial and appropriate technology is the social system that constructs and is constructed by the technology [274]. Crucially, in convivialist conceptions, people are seen as inherently interwoven in social networks and driven by complex motivations [92, 274]. Illich, 1973 – *"I consider conviviality to be individual freedom realised in personal interdependence and, as such, an intrinsic ethical value. I believe that, in any society, as conviviality is reduced below a certain level, no amount of industrial productivity can effectively satisfy the needs it creates among society's members."* [117]. The concept of convivial technology considers the interdependence between people and between technology and humans, Vetter, 2016 – *"reflecting the social construction of technology as well as the technological construction of human behaviour. This makes it possible to talk about "convivial technologies", a term that Ivan Illich did not use."* [274]. The concept of convivial technology focuses on the ideas that Illich raised – the need for creativity and autonomy for convivial tools [117, 274].

Social systems vary and Vetter puts forward convivial technology as a conceptual framework for technologies suitable for degrowth societies [274]. Kallis, 2014 – *"Degrowth signifies a society with a smaller metabolism, but more importantly, a society with a metabolism that has a different*

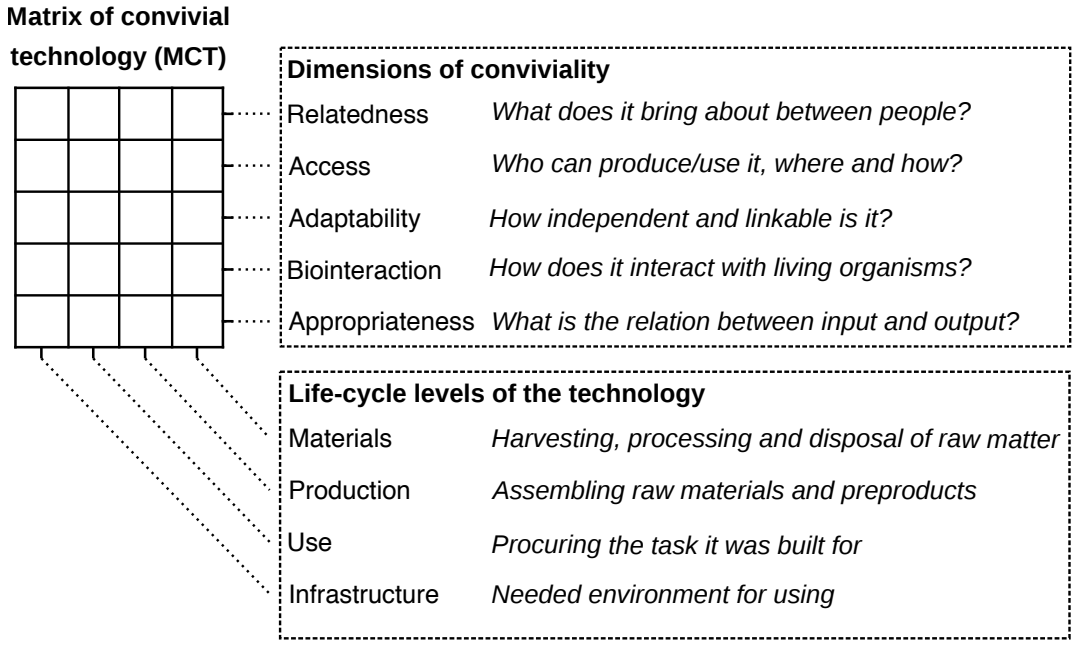



Figure 7.1: **The matrix of convivial technology** The matrix of convivial technology can be used to assess four life-cycle levels of a technology in terms of five dimensions of conviviality.

*structure and serves new functions. Degrowth does not call for doing less of the same. The objective is not to make an elephant leaner, but to turn an elephant into a snail. In a degrowth society everything will be different: different activities, different forms and uses of energy, different relations, different gender roles, different allocations of time between paid and non-paid work, different relations with the non-human world."* [54]. Degrowth theory opposes the idea of using green growth to solve ecological and social problems, since decoupling economic growth and material metabolism is not observed in reality [235, 274]. The concept of convivial technology offers a framework for developing technologies for a differently organised society [274].

Discussions of new technologies can become stuck in simple dichotomies between techno-pessimism and techno-optimism [134]. Recently, Vetter *et. al* published the matrix of convivial technology (MCT), a tool that was developed to discuss the ethical values of a given technology and to make explicit the impacts of the decisions taken when producing or using a technology (**Figure 7.1**) [274]. The MCT can help to move beyond this by structuring discussions of the various impacts of new technologies [274]. The MCT is a subjective and qualitative tool for assessing a technology during various stages of its life-cycle. Four life-cycle levels (materials, production, use, infrastructure) of a technology are considered in terms of five dimensions: relatedness, access, adaptability, bio-interaction and appropriateness, all of which are defined in **Figure 7.1**.

| Approach | IP approach | Context |
|---|---|---|
| IP1 | Patented | *in vivo* |
| IP2 | Patented | *in vitro* |
| OS1 | Open-source | *in vivo* |
| OS2 | Open-source | *in vitro* |

Table 7.1: **Four approaches to innovating synthetic biology technologies.** The four approaches arise from two decisions innovators can make during technology development. The first being whether the intellectual property (IP) is patented or shared using an open-source (OS) approach. The second being whether the technology is developed for an *in vitro* or *in vivo* context

In this chapter, I use the MCT to assess potential uses enabled by one of the technologies developed in this thesis, the CRISPR-valve arrays, in order to undertake the anticipatory and reflective activities endorsed by the AREA framework for responsible innovation [202]. The purpose of this chapter is to share a model that synthetic biologists and technologists in general can use to assess the technologies that they develop. My contribution to knowledge is adapting the MCT to enable easy comparison of different technologies via quantification and visualisation of the matrix. The MCT relies on a subjective assessment of the technology and therefore the results presented here are biased by my point of view (which is elaborated upon in the research journey Section 7.6). With this in mind, I have use personal pronouns throughout this chapter.

Decisions during the development of new biotechnologies influence how a technology is developed and can determine its impacts [12, 290]. Firstly, the choice of how to share the knowledge, via patenting intellectual property (IP) or via an open-source (OS) approach, which relates to the openness or permeability of the companies boundaries and in turn user-engagement during technology development [12]. A patent protects an invention by excluding actors other than patent holders from reproducing or using the invention unless its license is shared, sold or expires [224]. In the case of synthetic biology it has been pointed out that there is a diverse ecology of the open and the proprietary ranging from gene patenting to open source genetic parts [31]. Whilst open source allows distributed innovation, funding can be challenging: the benefits of the innovation are not reserved for the innovator and alternative strategies to secure funding must be taken [286]. A second choice made during biotechnology innovation is whether it is developed for *in vitro* or *in vivo* use. When combined, these suggest four possible approaches (**Table 7.1**). Countless other decisions would be made during biotechnology innovation however these two are focused upon to provide a case study for assessing technology innovation.

This chapter uses the MCT to reflect on possible outcomes and impacts arising from taking four different approaches to innovation (**Table 7.1**) of the CRISPR-valve arrays developed in this thesis. It begins by outlining the potential uses of the technology when it is developed with each approach to innovation (Section 7.2). Then, the MCT is adapted for quantification and visualisation (Section 7.3) and used to assess the different uses (Section 7.4). The chapter

concludes with a discussion of the merits and limitations of this activity (Section 7.5) and a further reflective activity describing my personal research journey (Section 7.6). Taken together, the activities presented in this chapter are examples of applying the AREA framework for responsible innovation [202] to scientific research. The origins and impacts of the scientific research are anticipated and reflected upon and a wider discussion of these impacts and actions that can be taken to change them are considered.

## 7.2 Potential uses of CRISPR-valve Arrays

The valve array biotechnology developed in this thesis is a versatile tool that enables arrays of genetic parts such as CRISPR guide RNAs (gRNAs) to be regulated using engineered transcriptional valves. CRISPR-valve arrays are a platform technology with many possible applications. These range from control of gene expression to genome engineering and are made possible due to the variety of *Cas* proteins and their associated functions that gRNAs can be designed for [34]. Various *Cas* proteins can be used for gene detection [74], meanwhile other *Cas* proteins can be used to knockdown particular genes [275]. Furthermore, any such technology can be developed for one of many uses since the gRNAs can be modified to target any gene. Therefore, they present a useful case study for considering how approaches to innovation affect development of technology towards uses. The market (customers) and uses would vary depending upon the strategy used to translate it for use outside of the lab and academia. The customers and uses would, in turn, influence how the technology evolved [12, 138].

In order to complete the MCT for the CRISPR-valve arrays, I found that a thorough understanding of the technology at each life-cycle level (**Figure 7.1**) is required. The product would be bespoke CRISPR-valve arrays suited for a particular user. By my estimation, all but one life-cycle level (the use) would be similar for all the approaches to innovation. CRISPR-valve arrays are encoded in DNA and the easiest way to make arrays is using synthetic DNA which is designed and ordered on a case-by-case basis. At the materials life-cycle level, no matter the approach to innovation in terms of IP or use *in vitro* or *in vivo*, synthetic DNA, reagents for DNA assembly, growth media and cells for DNA amplification would be used. Similarly, in these approaches to innovation, the production life-cycle level would likely always involve DNA assembly, amplification and testing using common wet lab procedures. Whilst glassware is used, disposable plastic-ware use is often substantial [264] to save time spent washing up. In all cases, the infrastructure required to design and make the technology would be office space for computational design and experimental labs or warehouses for production.

This means that the materials, production and infrastructure life-cycle levels (**Figure 7.1**) are likely to be similar in all approaches specified in **Table 7.1**. These life-cycle levels rely upon existing chemical industry supply chains, for example the mainstay of oligonucleotide synthesis has been the phosphoramidite method, which relies upon chemical building blocks derived from

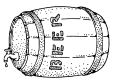
| ***Predicted use:*** | **Industrial production** | **Biosurveillance** | **Community production** | **Citizen science** |
|---|---|---|---|---|
| |  |  |  |  |
| ***Approach*** | *IP1* | *IP2* | *OS1* | *OS2* |
| *IP approach* | Patented | Patented | Open source | Open source |
| *Context* | *in vivo* | *in vitro* | *in vivo* | *in vitro* |
| *Application* | Gene regulation | Gene detection | Gene regulation | Gene detection |
| *Funding* | VC buys shares | VC buys shares | Capped return | Capped return |
| *User* | Industry | Industry | Community | Community |

Figure 7.2: **Reflecting on Different Approaches to CRISPR-valve Array Innovation** The funding, ownership, user and uses arising from each of the four approaches to innovation are predicted

mining oil and minerals, for 40 years [7]. The process of extracting the oil and mineral materials for these life-cycle levels often gives rise to pollution [75]. The impact of the materials varies depending on the material and how it was extracted, for example open-cast mining and shaft mining have very different environmental impacts [75]. This reveals the detail at which the impacts of supply chains need to be considered to make accurate assessments regarding the conviviality of a particular technology. However, since the objective of this chapter is to use the MCT to compare different approaches to innovation of the CRISPR-valve arrays and these life-cycle levels are the same for each approach, they are not assessed.

The life cycle level that would change depending upon the approach to innovation is the use. Therefore the MCT is only used to assess the conviviality of one of the life-cycle levels ("use") for each approach. CRISPR-valve arrays, like most synthetic biology tools are versatile and can be applied in many circumstances to enable different uses. Depending upon which approach is taken to innovation, different uses can be enabled through the development of a use of the technology. I set out to estimate a potential use based on the approach to innovation. In order to do this I predicted the likely outcomes of each approach in terms of funding, ownership, user and use (**Figure 7.2**).

The funding, user and use are some parameters that could vary depending upon the approach to innovation. Governments rationalise patent systems that protect intellectual property (IP) as a mechanism to correct market failure and to incentivise investment in research and development [224], This mechanism presumes that free knowledge will result in sub-par or no financial returns to its creators, leading to under-investment, under-productive markets and poorer economic and

social outcomes [224]. IP incentivises investment in research by exchanging investment in return for shares in a legal entity that owns IP, in expectation of future returns in the form of dividend payments. Since open-source approaches rely on sharing information, alternative forms of financing open-source research may be sought. Many funding options are available, such as investment as loans with capped returns, or grants obtained via crowdfunding.

Patenting IP can lead to a monopoly on use of technologies by those that hold the IP or able to pay for it [106]. Due to the high costs of establishing a patent, I predict that patented uses of the CRISPR-valve arrays would lead to use by large industrial companies which are able to pay high licencing fees. In the case of *in vivo* CRISPR-valve arrays, this may facilitate centralised industrial production at the scale of 10,000s of litres. However, if the IP is shared via an open-source (OS) approach [272], any user can utilise the technology. This would enable citizen scientists or community groups to use the technology, provided they have the means to understand and access it. Whilst anyone can use a technology after a patent expires (often after several decades), infrastructure requirements may limit who can apply the technology.

The other parameter I varied was development for an *in vitro* or *in vivo* context. The former lends itself to uses as a biosensor whilst the latter lends itself to use in producing goods (such as food, fibre, fuel, medicine). Production at industry scale is favoured when *in vivo* technologies are patented (IP1) whereas production at a smaller scale could enable distributed use of the technology by communities in bioreactors such as re-purposed microbreweries (OS1). The CRISPR-valve arrays would have to be engineered to suit the scale of use [3] since micro-environments are created as bioreactors are scaled up. This illustrates how, like other technologies, synthetic biology technologies are political [290].

The high costs of establishing patents may lead to patented *in vitro* CRISPR-valve arrays (IP2) being used by industry. CRISPR-valve arrays could be used to make biosensors, since various *Cas* proteins have been repurposed for gene detection [74]. The main customers of start-ups using biosensing technologies are in the sectors of infrastructure, renewable energy, marine, financial services, extractives, conservation, the water sector, nature-based solutions, nature positive supply chains and research in the case of Nature Metrics (www.naturemetrics.co.uk). Meanwhile, Biota (www.biota.com) currently advertise the following applications: decarbonisation, industrial asset integrity, environmental monitoring and energy (which involves fossil fuel extraction surveys). Whilst these start-ups apply biosensing to a variety of sectors, extractive industries are one customer for both companies. These extractive industries use biosensors as tools for bio-surveillance, that is, to detect genes, microorganisms or organisms in the environment and therefore biosensing may be used to legitimise mineral or oil extraction. Of the many uses arising from development of arrays using approach IP2, it is the use by extractive-industry customers that I consider, in order to assess a use that is likely to be non-convivial. In contrast, open source *in vitro* CRISPR-valve arrays (OS2) would likely be used by citizen scientists and academics for developing biosensors to study the environment. *In vitro* bio-production is another possible use,

though the scales required (100s to 1,000s of litres) currently have little precedence [220, 262].

Through tailoring the technology to different customers, the technology can facilitate a particular politics; a particular way that people and materials are organised. Technologies that lead to centralised production lead to monopolies and concentrations of wealth and power that lead to inequality [211]. This indicates that the seed that leads to centralisation or decentralisation of technologies may be planted when the innovator opts for patenting the technology and a shareholder business and fundraising model.

This assessment indicates that innovation strategies change not only the technology [12], but also the politics supporting and supported by the technology. The IP and funding strategy alters the target customers and uses of the technology (**Figure 7.2**). This demonstrates how versatile biotechnologies such as the CRISPR-valve arrays are capable of being developed to enable a variety of uses. If patented, the choice of how the biotechnology is developed is made by the scientists, funders and users. In the next section, I trial a tool for assessing whether the different possible uses of the CRISPR-valve arrays are convivial.

## 7.3 Adapting the Matrix of Convivial Technology

Having identified a possible use enabled by each approach to innovation, the next step would be to assess each use using the MCT and compare the results. The MCT allows someone to assess the use of a technology by selecting appropriate descriptors (**Figure 7.3**) for each element of the matrix. These descriptors are the result of coding analysis of data curated using ethnographic methods which encompass historical research of sources relating to alternative technologies, participatory observation in degrowth-oriented groups that develop or adapt “grass-roots technologies”, related online media, and narrative interviews about their motivation, the process of technology development and their ethical assumptions about technology [274]. The result is a series of antagonistic terms for each element in the matrix, which is used since it proved to be a way in which participants could easily compile the matrix [274]. The MCT advises that participants make crosses on the line in between the opposites in each field, showing to which side the chosen technology leans more [274]. After assessing the uses of the CRISPR-valve array technology using the MCT, comparing multiple assessments using the qualitative descriptors was not easy to do by eye. Therefore I set out to develop a way that the comparison of multiple MCT assessments could be made visually.

| Dimension: | Non-convivial | ↔ | Convivial | IP1 | IP2 | OS1 | OS2 |
|---|---|---|---|---|---|---|---|
| Relatedness<br>*What does the use bring about between people?* | Fosters competition | ↔ | Supports trust | | | | |
| | Fosters individual advantage | ↔ | Supports community | | | | |
| | Prefigured use only | ↔ | Allows creativity | | | | |
| | One solution fits all | ↔ | Respects local traditions | | | | |
| | Discourages care | ↔ | Simplifies care | | | | |
| | Uglifying | ↔ | Creates beauty | | | | |
| | Creates senselessness | ↔ | Creates art | | | | |
| | Alienating from own body | ↔ | Useful body enhancement | | | | |
| | Heteronomy | ↔ | Self-determination | | | | |
| | Compulsory | ↔ | Voluntarily | | | | |
| Access<br>*Who can produce / use it, where and how?* | Usable by an elite | ↔ | Usable by anyone | | | | |
| | Investor-controlled | ↔ | Open | | | | |
| | Cost intensive | ↔ | Low cost | | | | |
| | Need of foreign expert | ↔ | Use of local knowledge | | | | |
| | Not able to fulfill needs | ↔ | Fulfilling basic needs | | | | |
| | Abstract | ↔ | Comprehensible | | | | |
| | Repugnant | ↔ | Attractive | | | | |
| | Enforces cultural restraints | ↔ | Transforms cultural restraints | | | | |
| Adaptability<br>*how independent and linkable is the use?* | Fixed once finished | ↔ | Permanently changeable | | | | |
| | Isolated | ↔ | Interoperable | | | | |
| | Size fixed | ↔ | Scalable | | | | |
| | One-dimensional | ↔ | Multi-functional | | | | |
| | infrastructure needed | ↔ | Independent use possible | | | | |
| | Repairable by experts | ↔ | Repairable by skilled | | | | |
| | Close survey needed | ↔ | Uses self-regulation | | | | |
| | Monolithic | ↔ | Interchangeable | | | | |
| | One solution fits all | ↔ | Encourages diversity | | | | |
| | One piece | ↔ | Modular | | | | |
| Bio-interaction<br>*how does the use interact with living organisms?* | Illness/death | ↔ | Supports health | | | | |
| | Deteriorating soil | ↔ | Improving soil | | | | |
| | Water-polluting | ↔ | Improving water quality | | | | |
| | Air-polluting | ↔ | Supports clean air | | | | |
| | Violent | ↔ | Nonviolent | | | | |
| | Hazardous potential | ↔ | Safety proven and tested | | | | |
| | Toxic waste | ↔ | Biodegradeable | | | | |
| | Suppreses organic processes | ↔ | Allows co-productivities | | | | |
| Appropriateness<br>*what is the relation between input and output considering the context?* | Encourages waste | ↔ | Sustains sufficiency | | | | |
| | New | ↔ | Re-used | | | | |
| | Nondurable | ↔ | Durable | | | | |
| | Against local settings | ↔ | Uses local settings | | | | |
| | Needs painful time | ↔ | Allows joyful time | | | | |
| | Fossil energy | ↔ | Renewable energy | | | | |
| | Creates waste | ↔ | Byproducts are used | | | | |

Figure 7.3: **Assessment of the use life-cycle level of the matrix of convivial technology.** For each dimension of conviviality (relatedness, access, adaptability, bio-interaction and appropriateness), various antagonistic terms and an overarching question are specified. The assessor selects which term (if any) of each pair best describes the use of the technology. My assessment of the use arising from each approach to innovation is included (red: non-convivial, green: convivial, yellow: both, grey: not applicable).

To compare the assessed conviviality between the different uses of the CRISPR-valve arrays I developed a simple adaptation of the method such that the qualitative output of the MCT could be summarised numerically and then visualised. For each dimension, the assessments were converted to a list of numbers. Where a convivial or non-convivial descriptor had been selected, +1 or -1 was assigned, respectively. If both descriptors were deemed to be true then a value of 0 was assigned; if neither were considered to be true, a value of "NA" was assigned. The list of numbers for a particular dimension was summarised by calculating the mean value and rounding it. This summary value for each dimension could be 1, 0 or -1 and it is intended to approximate the conviviality. The summary values were used to colour the matrix of convivial technology as follows: red for -1 (non-convivial) values, orange for values of 0 and green for values of +1 (convivial). The perceived conviviality of each use could then be compared side-by-side (**Figure 7.4**).

## 7.4 Using the Matrix of Convivial Technology to Assess Uses Enabled by CRISPR-valve Arrays

Though the MCT can be used to assess four life-cycle levels, I focused only on the use level, since all other levels are the same for each approach to innovation. I assessed the anticipated use specified in **Figure 7.2** using the sets of descriptors outlined in the MCT (**Figure 7.3**). For each use, each dimension of the MCT was summarised as convivial, non-convivial or both and these summary values were compared (**Figure 7.4**).

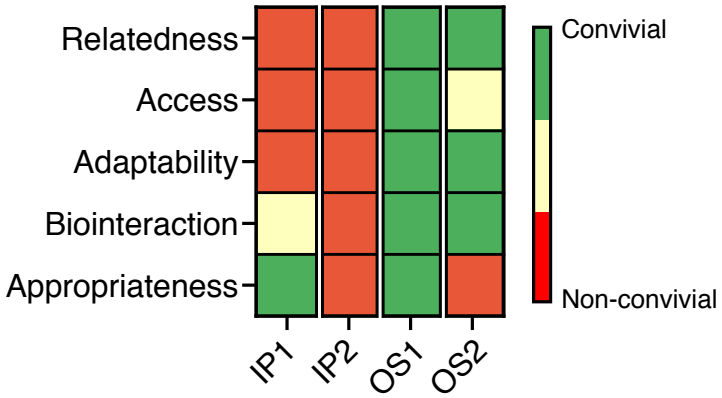



Figure 7.4: **Using the matrix of convivial technology to assess potential uses of CRISPR-valve arrays.** The uses arising from four approaches (x-axis) to translating the CRISPR-valve array technology developed in this thesis are assessed in terms of five dimensions of conviviality (y-axis). The uses assessed are industrial bio-production (IP1), bioprospecting (IP2), community bio-production (OS1) and citizen science (OS2). Fields are coloured by perceived conviviality, as measured by the innovator (red: non-convivial, green: convivial, yellow: both).

This approach enabled me to reflect upon the conviviality of different uses that could be enabled by the CRISPR-valve arrays. Whilst the detailed nature of the MCT is important, I found it impossible to holistically compare assessments of different uses. Therefore, I found the method to visualise the MCT useful.

I perceived uses arising from a patented approach that serves the extractive industry as non-convivial in the relatedness, access and adaptability dimensions (**Figure 7.4**). These dimensions are concerned with the ability to relate to, access and adapt the technology respectively. Exemplary descriptors that summarise my view of patented uses for these dimensions are “fosters individual advantage”, “usable by an elite” and “one solution fits all” respectively. To the contrary, I perceived uses arising from an open-source innovation approach as generally convivial for these three dimensions and example descriptors are “allows creativity”, “open” and “permanently changeable”, for the dimensions in the same order. The ambiguity of these descriptors gives an idea of how the MCT assessment really depends upon how the person who is filling in the MCT perceives the use of the technology.

For patented uses of CRISPR-valve arrays, I assessed bio-interaction as either both non-convivial and convivial, or non-convivial depending on the intended context (*in vivo*, IP1 or *in vitro*, IP2) respectively. For *in vivo* industrial bio-production this is because there is likely a reliance upon monoculture crops due to the scale of the operation. Such crops often necessitate land management with chemical-derived fertilisers and pesticides which disrupt microbiomes [266] and rely upon mining practices which cause pollution [75].

Patented *in vitro* development of the CRISPR-valve array for use in identifying and justifying use of sites for mineral or oil extraction is non-convivial in any dimension according to my assessment. The non-convivial descriptors for this dimension are illness/death, deteriorating soil, water-polluting, air-polluting, violent, hazardous potential, toxic waste and suppresses organic processes, all of which could be associated with mining. This demonstrates how a technology that may appear benign could have unanticipated consequences that can be harmful. In this case, the benign biosensor technology may be used to enable extractive industries to continue harmful practices meanwhile "green-washing" what is being done [60]. Of course, it could be argued the other way, that bioprospecting reduces environmental damage by highlighting areas of biodiversity. Fitting technologies into existing extractive and polluting industry and funding models comes with the risk of unintended harmful consequences arising from the existing system. However if we allow the open source, collaborative potential of novel technologies to influence society, the outcomes could be very different, allowing the new technology to lead to and influence societal system change.

Uses arising from an open-source approach are convivial in four dimensions: relatedness, access, adaptability and bio-interaction. The access and appropriateness of these uses would require and depend upon good information provision and customer engagement. Appropriateness and bio-interaction necessitates biodegradable design of any *in vitro* biosensor rather than using disposable plastics. With disposable plastics being the current norm for citizen science biosensors (for example, lateral flow devices), the appropriateness of this use is deemed non-convivial. The predicted use of *in vivo* open source CRISPR-valve arrays is community bio-production. I perceive distributed bio-production as convivial in all aspects as it can enable users to make the things that they need from local, biodegradable resources. This visual comparison of the conviviality of different potential uses enabled by CRISPR-valve arrays allowed me to see clearly how I perceive the different possible uses of this biotechnology.

## 7.5 Discussion

In this chapter I have used the matrix of convivial technology (MCT) to evaluate four possible uses that could be enabled by CRISPR-valve arrays. I found that the MCT offered a way to assess the wider impacts of a technology in detail. This exercise also inspired me to imagine and learn about different ways that the technology developed in this thesis could be translated for use outside of academia. I found that the first step for completing the MCT involved understanding and defining the four life-cycle levels of the technology (**Figure 7.2**) and that three of the life-cycle levels (materials, production and infrastructure) were the same regardless of the approach to innovation and were non-convivial owing to the pollution that they caused. Sustainable materials would need to be utilised in these lifecycle levels to make them convivial, making alternatives to chemical synthesis of DNA, such as cellular production of defined DNA sequences, important.

To consider the use of the technology arising from each approach to innovation, I had to define aspects of the business model (**Figure 7.2**). In doing this, the anticipated customers and use of the technology became apparent. Two aspects of a business model that influence how the business develops a technology are openness (the ability of users and competitors to share the technology) and user-engagement [12], which both relate to knowledge sharing. An open-source approach to technology development prioritises these two parameters whereas a patented, IP, model does not. By evaluating the anticipated use and resulting uses using the MCT, the wider impacts of technologies arising from different approaches to innovation can be understood (**Figure 7.4**). However, the uses that I have proposed are somewhat arbitrary and at a minimum, they provide examples that can be compared using the MCT and can provoke discussion. It would be interesting to interview a wider group of synthetic biologists and technology users to see what uses of CRISPR-valve arrays they anticipate.

The MCT helped me to critically reflect on the effects that this technology could have on the world and on people's lives. This activity indicated that the conviviality of a technology could vary depending upon the approach used to translate it for use beyond academia. The business model changes the funders, objectives, incentives and customers and dictates what the technology is designed to do [12]. The conviviality of the uses enabled by the technology varies significantly depending on what the customers use the technology for. Whilst this only revealed how I perceive these uses, by going through the process, I became more aware of potential wider consequences of the technology that were not intended. Thus, as well as confirming someone's perception of a technology, the MCT could also influence how technologists see the effects of the technology on the world, leading them to make more convivial decisions during technology development. It could be useful to see how different stakeholders in a technology (innovators, users, funders) perceive its conviviality.

The MCT enabled me to trace a process of events that may lead to development of non-convivial technologies. This process starts with IP strategy which changes the approach to funding, targeted users, the anticipated uses and therefore how the technology is developed. This indicates a self-fulfilling cycle at play in technology translation. Venture-capitalists (VCs) set the norm of technology start-ups by funding and guiding them. This norm often relies on a for-profit approach to business where VCs fund businesses in return for shares in the company, with the expectation of dividends or selling shares at an inflated value at a later date. Though not always the case, shareholders often seek large returns, leading to a business strategy which relies upon IP and targeting customers and users in industry, where wealth is concentrated. The resulting economic inequality allows VCs to continue to set the norm of how to set up a start-up and the cycle continues. This economic inequality also leads to social and environmental instability [242]. Training and opportunities in alternative business management approaches with a philosophy of developing convivial technologies could offer a different approach [61]. Support and investment that is not purely economically driven could catalyse changes in the way that businesses are set

up [221].

Recent technologies such as social media and online marketplaces have brought great opportunities for interconnection as well as great challenges, such as the concentration of wealth. In Bristol and beyond, technology start-ups generally follow the same approach to business as those in the internet boom ("dotcom bubble"). VC training and funding leads innovators to create a for-profit start-up company, generate intellectual property and commercial products. This often leads to companies guided by shareholders which prioritise profits over conviviality [61]. Economic inequality is perpetuated [211] and leads to social and environmental instability [242]. Approaches to innovation exist which prioritise goals beyond economic growth and different relations to technology such as those based on an ethic and a practice of care [203] or a tool for reparative governance [88] are possible. A non-profit start-up incubator "non-profit ventures" seeking to implement a post-growth economy encourages three core tenets: bootstrapping, flat growth and non-extraction [217]. Bootstrapping involves starting small, lean and independent; flat growth focuses on consistent long-term income and non-extraction ensures there are no exits, the financial value stays in the company and dividends are only for charity. Companies that share dividends in this way may mitigate environmental degradation caused by poverty and improve well-being [17]. Enterprise and technology could offer a way of transitioning to technologies, social systems and businesses for a healthy living planet. Reflective activities such as the matrix of convivial technology can guide entrepreneurs and funders in considering the wider effects of the systems that they create.

This chapter had two aims: adapting a tool for comparing the conviviality of applied technologies and using it to reflect upon potential uses that could be enabled by CRISPR-valve arrays. Technologies such as those developed in this thesis shape our lived experience. At this stage, the CRISPR-valve array is more of an abstract idea than a technology. There are many possible ways of applying it in the "real-world" as well as many approaches to get there. I used the MCT [274] as a tool to evaluate uses arising from different approaches to developing this technology.

My research journey during this PhD involved learning and attempting to practice responsible research and innovation (RRI). The AREA framework for responsible innovation (**Figure 2.16**) encourages four ongoing objectives: anticipate, reflect, engage and act [202]. Therefore I conclude this chapter on responsible innovation with an activity that fulfils some of these objectives: sharing the personal "researcher's journey", with a focus on RRI. This leads me to further anticipate the impacts of the research that might arise and reflect on the purposes, motivations and potential implications of the research. This thesis is the beginning of opening up such impacts to broader deliberation and debate (engage) and influencing the direction of the research and innovation process itself (act).

## 7.6 Research journey towards responsible innovation

Human endeavours are born of and influenced by the world around them. A PhD is influenced by the experiences of the researcher. Scientific research is often considered objective, though the experiences of researchers [18] and the cultures they participate in direct their studies [230], influencing funding and research priorities, methods and outputs [13]. Science does not happen cut off in the laboratory, but exists as part of society and culture. Scientific research and facts are influenced and mediated by culture, just as science offers ideas and concepts – as well as technologies – that get incorporated into and influence the direction and workings of society. To acknowledge and understand these influences, a personal "researcher's journey" section is sometimes included in humanities theses [58, 174]. I choose to follow this approach. This section explores my wider research journey beyond the scientific experiments and analyses I have done: an inquiry into how to research and innovate responsibly.

At the outset of the PhD I was both enthusiastic and wary of the promise of designing genetically engineered microorganisms to produce the things I used daily from plants. Modifying the genetic code of life sounded like something to do cautiously and with care. Organising a student synthetic biology association early in the PhD showed me that I was not alone in my naive, uncritical enthusiasm for synthetic biology. Whilst unsure of what to make of synthetic biology, the degree was an exciting opportunity and would help me to understand it in greater detail. Initially I was guided by the objectives laid out at the foundation of synthetic biology: simplifying genetics to a set of separate parts [15]. This led to my thesis focusing on developing methods to create novel genetic parts: more nuts and bolts for engineering biology. Since then, training courses, discussions and reading on responsible research and innovation (RRI) highlighted reflection and reflexivity (the capacity to adapt research direction in response to new knowledge) as a core process in undertaking research responsibly [202]. This guided me to reflect on the responsibility of my own research.

I began to reflect upon how responsible some of the norms of synthetic biology are. The first being the normative innovation strategy of rapidly growing start-up companies with venture-capitalist funding. Whilst this does lead to wealth creation, it also leads to wealth concentration [29]. The norms of technology innovation in the 21st century are leading to increased wealth inequality [227] which can lead to environmental degradation directly [225] and indirectly [242]. This growth model generally relies on the generation of intellectual property, another norm of synthetic biology: patenting genetic parts and engineered organisms. The first patent examiner of the US patent system, Thomas Jefferson, voiced concerns over monopolies on intellectual property, comparing ideas to fire, which can be shared without any loss of illumination: Jefferson, 1813 – *"Its peculiar character, too, is that no one possesses the less, because every other possesses the whole of it. He who receives an idea from me, receives instruction himself without lessening mine; as he who lights his taper at mine, receives light without darkening me."* [122]. Similarly, living biological organisms could be considered open-source: a single flower often produces 1,000s

of seeds and a single food or drink fermentation culture can be shared countless times.

A third norm arising from this approach to growth and innovation is ownership and therefore governance by investor shareholders with primary motives of profits rather than employee or environmental well-being. To reach rapid growth and large profits, the normal vision for synthetic biology production is centralised production [290], which can displace livelihoods [208, 260]. To enable this business model, inputs must be available in bulk, requiring crop monocultures, which has non-convivial environmental consequences [11, 113, 171]. This perpetuates [109, 194] extractive economies where low-value materials are bought and value is added in a centralised production facility before sale, concentrating wealth. All of these norms arise from an outlook of separation from nature and from one another. The outcome is technologies and systems that are fragile and stuck due to power concentration, inaccessibility [117], technological lock-in [80]. This prevents adaptation of technologies which could offer resilience as climate change continues [21]. Whilst many synthetic biology norms seem irresponsible, they are not the only way.

| Norm | Approach to | Alternatives |
|---|---|---|
| Exponential | **Growth** | Post-growth |
| Patenting | **Knowledge** | Open-source |
| Private | **Ownership** | Distributed |
| Centralised | **Production** | Decentralised |
| Monoculture | **Input** | Polyculture |
| Extractive | **Economy** | Equitable |
| Separated | **View of nature** | Connected |
| Fragile | **Outcome** | Resilient |

Figure 7.5: **Synthetic biology research and innovation norms and alternatives**

Having identified some of the impacts of synthetic biology under the normative innovation paradigm, I sought alternatives (**Figure 7.5**). In terms of an approach to growth, post-growth innovation [203] incubators [217] offer alternative philosophies of business which advocate slowly establishing the business using a bootstrapping approach that results in retaining power within the enterprise and leads to equitable non-extractive economies [217]. In terms of sharing knowledge, open-source [272, 289] and copyleft [195] approaches lead to more accessible and adaptable technologies. There are models for more distributed ownership and governance [218] structures which still lead to highly valued enterprises [59]. A steward ownership model (rather than shareholder ownership) means that the business is owned by a trust, whose board comprises people with an interest in the companies values rather than solely the economic prospects of the company [218]. The scale of production embedded within technologies is a choice which dictates how people and materials can be organised [290]. This could enable the use of diverse inputs and biodiverse agroecological ecologies that rely upon polycultures [214]. Human cultures have many outlooks [61] and that of humans as part of a connected living world is emerging once more [185] and can guide responsible research [136]. These alternative approaches to innovation could lead to adaptable, distributed technologies which offer sustainability and resilience [77].

In this chapter I have applied the AREA framework (**Figure 2.16**) [202] for responsible innovation to my scientific research. I began by anticipating the complex impacts that different approaches to innovation can have (**Figure 7.4**). Then I reflected on the roots of this PhD (Section 7.6) and alternative approaches to guide synthetic biology research and innovation (**Figure 7.5**). Whilst engaging audiences on these matters is possible, it is not something that I have had chance to do frequently within this PhD. As a whole, this chapter is the beginning of engaging wider audiences to deliberate approaches to synthetic biology innovation so that we can decide how to act to influence its trajectory.

CHAPTER 8

# CONCLUSIONS

Synthetic biologists seek to engineer microorganisms, yet are often faced with a limited set of genetic parts to tune cellular behaviours. Motivated by this, we set out to develop methods to accelerate the creation and characterisation of new genetic parts designed to control transcriptional flows. We developed methods to characterise large libraries of terminator genetic parts in parallel, enabling new parts with desired functions to be rapidly designed. This can speed up the design-build-test-learn cycle commonly used in synthetic biology.

We began in Chapter 4, using nanopore DNA sequencing (DNA-seq) to characterise the composition and mutations of large DNA libraries. We showed that a simple, low-cost method could be used to combinatorially assemble DNA via ligation of annealed oligonucleotides encoding genetic part variants in a specified order within a plasmid backbone. We constructed two libraries of terminators complete with upstream sequences, which could enable gene ratios to be tuned transcriptionally. Each library consisted of spacer, modifier and core-terminator (which encodes a hairpin and U-tract) regions. Measuring library composition using DNA-seq necessitated optimising BLASTN parameters to enable nanopore sequencing reads (which have up to 10% errors) to be assigned to genetic designs in a process known as demultiplexing. We found that a constraint on the assembled genetic part sequences was necessary to ensure that each design could be distinguished from all others: the designs could not have a stretch of > 10 identical nucleotides compared to other designs.

The composition of genetic designs in a DNA library is seldom investigated yet is crucial for ensuring that entire libraries can be characterised using sequencing assays. We showed that DNA libraries assembled *in vitro* and amplified *in vivo* using different cell strains and protocols resulted in significantly different library compositions. We showed that after DNA assembly, DNA library composition was reasonably uniform and that this was maintained upon amplification

only if the encoded RNA and protein were not expressed. Consensus genetic design sequences generated from the DNA-seq data revealed increased indel mutations during amplification with expression. This may arise from a selection pressure which favours mutations that reduce burden of the genetic design [43]. However, we found that different designs were selected in each replicate, indicating that the most abundant designs arise, at least in part, by chance. All assembled libraries showed the presence of larger mutations (insertion sequences, plasmid dimers) in a minority of sequencing reads (<5%). Since these can be filtered out during data processing they do not pose a problem for sequencing based characterisation. Whilst indels can be avoided by amplification without expression, the ability to identify mutations highlights opportunities for engineering biology: using natural variation to diversify DNA libraries, or to identify regions of burden within genetic constructs via mutation frequencies. The main scientific findings of Chapter 4 relate to proving that ligation of genetic part duplexes can be used to combinatorially assemble large DNA libraries and how DNA library composition varies with DNA library amplification method.

Next, in Chapter 5 we measured the termination efficiency of a library of transcriptional terminators using direct RNA sequencing (dRNA-seq). This was made possible by using the sequence of the terminator as an intrinsic barcode. The transcriptional profile generated from the sequencing reads reveals precisely where transcription termination occurs for each design. We found several general features in the dRNA-seq read profiles that are important for their interpretation. Read profiles show drops in read depth away from the intrinsic barcode. The effect of drops in read depth on termination efficiency is negligible (amounting to a deviation of 1-2%) and can be corrected for. We built a simple model using read profiles of the RNA calibration strand, a control RNA sequence that can be included in the sequencing experiments. Our model explains the drop in read depth upstream of barcodes in terms of three processes: fragmentation of transcripts, incomplete sequencing adaptor ligation and truncation of sequenced transcripts. However, it does not explain drops in read depth downstream of terminators, which warrants further investigation and could be due to a gradual fall-off of T7 RNA polymerase during transcription or reduced stability of the resultant transcripts relative to the RNA calibration strand control. We used our model to predict the deviation between actual and measured termination profiles and make small corrections (< 2%) to the measured termination efficiencies. Drops in read depth which look like transcription termination but are in fact caused by promiscuous adaptor ligation can be avoided by ensuring that polyadenylation is efficient during sequencing library preparation. The complexities revealed for this well defined and simple experimental setting indicate how useful *in vitro* experiments can be for teasing apart the multitude of effects involved in measuring transcript isoforms.

The measured termination profiles reveal where each terminator terminates transcription at a nucleotide resolution. These profiles showed that upstream genetic context can significantly effect, and thereby tune, the termination efficiency of a terminator, but that this rarely effects the

position of termination. Termination is found to always occur at more than one nucleotide, leading to the production of a variety of terminated transcript isoforms. Whilst uncommon, some cases are found where upstream genetic context changes the ratio of terminated transcript isoforms. The median termination efficiency of terminators correlates with the number of uridines transcribed downstream of the core-terminator hairpin and abnormally weak terminators can be explained in terms of the absence of a co-transcriptionally folded hairpin structure. The main scientific finding of Chapter 5 is that terminators stop transcription to varying degrees depending upon the upstream sequence, which can tune the termination efficiency of T7 RNAP *in vitro*. Furthermore, a method was developed that can measure the termination of libraries of terminators using nanopore direct RNA sequencing.

In Chapter 6 we design, create and test two further DNA libraries which explore the effect of motifs within and also upstream of the core-terminator on termination efficiency. The range of termination efficiencies revealed in Chapter 5 leads us to treat our designs as valves (rather than off-switches) for tuning the number of RNA polymerase molecules able to continue transcription past the core-terminator part. As well as tuning termination efficiencies over ranges of up to 68%, we found that modifiers can insulate terminators from spacer sequences further upstream.

Modifiers designed to interact with the core-terminator via base-pairing and structural interactions uncovered some principles for modifier and terminator design. Short hairpin structures in the modifier were the best at insulating terminators from upstream genetic context. This could occur by restricting the ensemble of RNA secondary structures that are possible at the point of termination. Sequence complementarity between the core-terminator and the modifier was also found to significantly decrease termination efficiency but the effect reduces with distance between the complementary motifs. We hypothesise that base-pairing arising within the RNA molecule during transcription can reduce the ability to terminate transcription. The degree to which termination efficiency decreases varied dependent on the region of the core-terminator targeted: base-pairing with the stem generally had a larger affect than base-pairing with the loop. This could be since the loop is unpaired during terminator hairpin formation whereas the stem must base-pair to form the hairpin. The free energy released from base-pairing in the stem also had the highest contribution to the ratcheting of the U-tract off the DNA [142, 150] and so disrupting it may reduce termination. A further library focusing on core-terminators reveals that increasing the length of the terminator U-tract generally increases termination whereas removing native upstream context decreases termination. The scientific findings of Chapter 6 are principles for designing novel transcriptional valve sequences.

Chapter 6 concludes with utilisation of the engineered transcriptional valves to control the stoichiometry of RNA parts. This approach reflects recent studies which reveal that terminators tune transcription in microorganisms to produce multiple transcript isoforms [149]. Ratios of three consecutive CRISPR guide RNAs (gRNAs) were predictively regulated using a gradual decrease in RNAP flux modulated by our engineered transcriptional valves. The termination

efficiency of valves was found to decrease as their proximity to the promoter increases. This could be due to increased read-through of the terminators caused by an increase in RNAP traffic. This was supported by characterisation of a small library of terminators at a lower concentration of T7 RNAP which revealed that for some terminators, termination efficiency increased as a result of decreased T7 RNAP traffic. This demonstrated that transcriptional valves can be applied in bio-design and that their function depends on genetic context.

With preliminary data showing how these transcriptional valves could be applied to engineer arrays of CRISPR gRNAs, in Chapter 7, the potential impacts of this prototype technology were anticipated and reflected upon. These are two elements of the AREA framework for responsible innovation which can be used to guide technology development [202]. Whilst no scientific contributions are made in this chapter, a contribution to knowledge is made by adapting the matrix of convivial technology (MCT) [274] for quantification and visualisation to allow technology assessments to be easily compared. Innovation of the CRISPR-valve array was considered under four scenarios: patented *in vitro* or *in vivo* use and open-source *in vitro* or *in vivo* use. The user and use arising from each approach was predicted and assessed using the MCT. I completed one final activity to reflect upon my research: accounting my personal research journey. The MCT and research journey offer opportunities for researchers to reflect on the antecedents and potential impacts of their research.

The transcriptional valves developed in this work can regulate the expression of RNA parts. The advantage of regulating gene expression with valves (rather than promoters) is that ratios of genetic parts can be coupled at the transcriptional level and that a single recruited RNAP can regulate multiple genetic parts. The valve determines the ratio of transcript isoforms of consecutive genetic parts. It remains to be seen how the noise (variability) of promoters and terminators differs and this would determine their merits and limitations for transcription of genetic parts. In any case, these transcriptional valves could be used for regulation of many types of genetic part beyond CRISPR guide RNAs such as aptazymes or small transcription activating RNAs [147]. Multiplexed sequencing methods offer a multitude of uses for increasing the information gathered during high-throughput sequencing studies of genetic designs. Our insights into designing sequence-based intrinsic barcodes for error-prone nanopore sequencing reads are useful for research communities beyond synthetic biology. DNA barcoding strategies are used in environmental microbiology studies to increase the resolution of sequences gathered and more accurately infer the composition of microbial communities [130]. Likewise, modelling dRNA-seq data could help to interpret transcriptome datasets collected using nanopore sequencing [105]. Besides the scientific research presented in this thesis, the exploration of tools for researching and innovating responsibly by considering the bigger picture could inspire more researchers to do so.

## 8.1 Future Directions

### 8.1.1 Studying intrinsic termination mechanisms

Aspects of transcriptional termination mechanisms are intricate and continue to be revealed in greater detail [192, 222]. Nascent and transcribed RNA molecules exist as dynamic ensembles of probable alternative equilibrium structures [184]. RNAP are flexible structures too: they can change conformations during the process of termination [69]. These molecules interact with one another dynamically during the process of termination [299]. *E. coli* RNAP and T7 RNAP are widely utilised in synthetic biology [281] and their mechanisms of termination are well studied [222]. Terminators which evolved to terminate *E. coli* can terminate T7 RNAP *in vitro* as shown in this thesis and previously [237]. Similarly, the T7 terminator can terminate *E. coli* RNAP *in vivo* and *in vitro* [296]. Multiplexed sequencing methods could be used to investigate the terminator cross-compatibility and through comparison of genetic designs, the fundamental mechanisms of termination.

The evolutionary origin, structure and function of T7 RNAP and *E. coli* RNAP differ [285] yet they can both undergo intrinsic termination. T7 RNAP is a single-subunit RNAP, compared to the multi-subunit *E. coli* RNAP, which has five subunits and weighs 4-fold more [253]. The structure of T7 RNAP resembles a right hand, with 'finger', 'palm' and 'thumb' subdomains holding the DNA template and newly-forming RNA molecule [140]. Only single-subunit RNAPs can initiate transcription directly, without the need for the accessory or regulatory factors needed by multi-subunit RNAPs [285]. Similarly, termination of the multi-subunit *E. coli* RNAP sometimes depends on accessory factors such as *Rho*, which are not required for T7 RNAP termination. Multi-subunit RNAs resemble a crab claw, with two 'jaw' domains that interact with DNA downstream of the point of transcription [285]. The rate of elongation of T7 RNA polymerase is approximately 5-fold greater than that of *E. coli* RNAP [252]. This means that T7 RNAP travels far ahead of the ribosome, at approximately 8-fold greater speed [118]. As a result, unlike *E. coli* RNAP, ribosome-RNAP coupling does not occur for T7 RNAP. Consequentially, it cannot synthesise key bacterial biomolecules [155] since RNA folding depends on RNAP dynamics as well as RNA dynamics. Despite their differences, there are similarities in the transcriptional processes of these two RNAP.

Ultimately, both enzymes generates an RNA copy of a DNA template. Both have an exit channel from which nascent RNA emerges, which accommodates five nucleotides of nascent RNA for *E. Coli* [87] and the same for T7 RNAP [294]. The active site of the elongation complex (EC) of both T7 [256] and *E. Coli* [271] can hold a DNA-RNA hybrid of length eight nucleotides. In the bacterial system, the displacement and release of an accessory sigma-factor expands the active-site cavity whereas for the T7 system, refolding occurs [271]. Both systems form a kink between the DNA-RNA hybrid and downstream DNA [285, 294]. These RNAP have independent evolutionary histories and are related to other single- and multi-subunit RNAP

[285]. Our findings could therefore be relevant to the termination of many RNAP across the web of life. The U-tract and hairpin are important factors of intrinsic termination in bacteria as shown in this thesis and previous studies [47, 49, 237]. However, this is not always the case, for instance the hairpin is not essential in for transcription termination in Archaea [56]. Applying multiplexed sequencing to study the same terminator library with other RNAPs *in vitro* could enable an understanding of how termination mechanisms vary in isolation from cellular context and processes, which can influence termination measurements *in vivo*.

The mechanism of intrinsic termination relies on the sequence encoded in the DNA template and the transcription elongation complex (EC) which comprises RNAP, DNA and RNA [222]. There are four key steps in the process of intrinsic termination [222]. First, the EC pauses at a U-tract, then a hairpin nucleates within the RNA exit channel, the hairpin forms completely and finally, the EC dissociates [222]. There are a variety of ways that the EC can pause [299]. For terminators, EC pausing relies on transcription of a series of uridines, which is thought to reduce the stability of the DNA-RNA hybrid since A:U base-pairs have fewer hydrogen bonds than G:C base-pairs [192]. An elemental pause sequence (EPS) also appears important, this is a G:C base-pair at the position where the DNA-RNA hybrid is attempting to unwind [192].

Hairpin formation relies upon a G-C rich motif and often contains the EPS at its base; at the top is a loop of unpaired nucleotides [192]. As the EC progresses along the DNA template, there is a kinetic competition between elongation and termination [222]. Pausing at the U-tract favours the termination pathway as it allows time for the hairpin to form in the RNAP exit channel, initiating the series of events that cause termination [107]. Single-molecule studies have revealed that pauses can be minutes in length during a termination event [293]. Other sequences are thought to play a role in pause initiation efficiency and duration, such as those in the active site, downstream DNA channel and RNA exit channel [222]. Since pausing initiates the termination pathway and determines the length of time for the hairpin to form, such sequences also influence termination efficiency [222].

This thesis focused on sequences within and proximal to the RNA exit channel: sequence context upstream of the terminator. Lubkowska *et al.* showed that the *E. coli* RNAP EC interferes with hairpin formation as it transcribes between 6-8 nt downstream of the hairpin stem, but not beyond that [170]. They infer that for terminators with a U-tract immediately after the hairpin, just prior to termination, the downstream arm of the hairpin stem is protected from base-pairing with the upstream arm. This protection can make hairpin folding sensitive to the upstream sequence context, since the upstream arm can engage in competing interactions with the nascent RNA [170]. Even when the upstream arm of the hairpin is not protected, upstream RNA can still influence the RNA structure ensemble at the point of termination [237]. We have shown that upstream sequence context can change termination efficiency and studying further terminator libraries could elucidate how it influences hairpin formation more precisely. Our methods could also be used to make and study libraries which vary sequences in the active site and downstream

DNA channel of the EC.

Structural studies elucidate aspects of the interaction of RNAP with nascent RNA. The RNA exit channel is sterically constrained and studies have shown that *E. coli* RNAP may guide the formation of RNA structures [128]. A cryo-EM structure of the his operon attenuator hairpin stabilising a paused *E. coli* RNAP EC reveal that the paused hairpin has interactions with the RNA exit channel [128]. In the paused EC state, the exit channel is positively charged, which may facilitate hairpin nucleation, formation. There is also a positively charged route outside of the exit channel which may provide a route for upstream RNA to facilitate hairpin formation [128]. In this paused state, a global conformational RNAP change that causes allosteric inhibition has occurred and only the RNA has translocated; the DNA has not [128]. Whilst these studies indicate that *E. coli* RNAP may guide nascent RNAP structure formation, any role of a positively charged exit channel in nucleating duplexes remains to be tested [128].

Positive charges lining the RNA exit channel and its surroundings are conserved among bacterial RNAPs and may chaperone formation of RNA secondary structures within and nearby [229]. T7 RNAP has a positive charge covering nearly the entirety of its interior and through the pores and channels (which are unique to the EC) to the surface [256]. Furthermore, an RNA hairpin could be computationally docked within T7 RNAP without steric clashes and invasion of the RNA exit channel by a hairpin interacting with upstream RNA is proposed to lead to catalytic impairment and EC destabilisation [237].

A structure of an EC bordering on intrinsic termination has not yet been obtained, however, it is assumed that terminator hairpin formation in the exit channel is supported by the same RNA chaperoning functions of RNAP as for pause hairpins [229]. No structure has been observed with a 5'-RNA strand interacting with the hairpin within the exit channel either, however, space for it appears to be available and could lead it to open up further; this has been observed in other RNAPs [229]. Biochemical studies suggest that hairpin invasion is followed by changes to the RNAP conformation which cause the hairpin to visit other sites within RNAP, eventually leading to EC disruption [20].

Transcriptional termination mechanisms can be affected by translational processes too. Coupling of transcription with translation is an accepted paradigm in prokaryotes that lack physical barriers between the two processes [9]. RNAP can be linked with the ribosome via synchronised rates of transcription and translation rates or protein 'bridges' such as *NusG* [9]. The ribosome can actually suppress intrinsic termination during translation [283]. Custom terminator libraries could be used to understand how sequence motifs influence such coupling mechanisms. Our methods could also be applied in different biological contexts to study how termination occurs *in vivo*, with the ribosome, compared to *in vitro*. Comparison of T7 RNAP and *E. coli* RNAP, which travels ahead of the ribosome, could tease apart the affect of ribosome-RNAP coupling *in vivo* or using cell-free extracts.

Whilst we have completed this work *in vitro*, various long read sequencing methods could

be developed for *in vivo* characterisation of genetic parts. This necessitates a way of taking a measurement that connects the genetic sequence of each design in the library to the phenotype that it produces. Long read DNA-seq works for measuring phenotypes encoded in DNA [141, 236] whereas dRNA-seq works for phenotypes encoded in the transcriptome such as promoters and terminators. The DNA-seq method in Chapter 4 could be extended to characterise cells sorted based upon their termination phenotype using the GFP and RFP fluorophores surrounding the terminator. For example, using FACS followed by sequencing [32]. The nanopore sequencing and bioinformatic methods developed in this thesis will offer template methods for such investigations. However no nucleotide resolution information would be gleaned with this approach. This is a fundamental challenge for those developing sequencing based assays [163]: it is hard to connect translational phenotypes (such as expressed proteins) to their genotype of origin.

Terminators can also be used as switches, and the mechanisms underlying this use are diverse and multifaceted. Attenuators and riboswitches are terminators with switch-like behaviour [19, 128]. Their mechanism relies upon changes in RNA secondary structure in response to cellular stimuli. Generally, attenuators regulate the leader regions of bacterial operons using pauses induced by a specific RNA structure; riboswitch activation can involve one of the many types of RNA structure induced RNAP pauses too [299]. Both riboswitches and attenuators can use ligands to control transcription and translation. For instance a ribosome binding site (RBS) may be available for translation initiation and upon ligand binding, RNA structural changes conceal the RBS within a hairpin, preventing translation [19]. For transcription, ligand binding can alter the presence of a terminating hairpin, allowing transcriptional read-through to occur [19]. In recent studies currently under peer review, ligand-induced structural changes have also been shown to reposition the RNA and change RNAP conformation to expand the RNAP exit channel and enable transcription to continue [280]. Ligand-responsive RNA structures such as these can diversify synthetic biology tools.

High-throughput microplate assays or directed evolution could be used to optimise attenuators and riboswitches for use in bespoke applications. Nanopore sequencing could enable multiplexed analyses of libraries of these RNA-encoded tools or to track their sequences as they evolve to respond to a cellular processes. Whilst single-molecule, *in silico* and *in vivo* structural methods [300] are better suited to studying the fate and structure of individual RNAP [129], the sequencing methods in this thesis are well-suited to comparative analyses of the function of terminator sequences in different genetic or cellular contexts. By comparing these datasets with one another, yet more details of mechanisms of transcriptional termination are likely to be unveiled.

### 8.1.2 Engineering DNA library composition

The DNA assembly method we use could be extended to generate libraries of genetic designs where multiple genetic parts which are distant in the genetic construct are varied combinatorially. The number and accuracy of reads from long read sequencing experiments continually improves,

making barcode demultiplexing and full library characterisation easier. The pooled DNA-seq approach in Chapter 4 could be applied to understand the composition of DNA libraries generated using other DNA assembly methods [67]. Alternative amplification methods such as the use of PCR could be investigated too. This knowledge of how to generate uniform libraries of genetic constructs improves the chance of engineering complete DNA libraries and generating complete data-sets.

### 8.1.3 Uses of transcriptional valves

The transcriptional valves that we have developed and characterised will be of use those applying synthetic biology to develop *in vitro* cell-free tools [127, 262]. The main anticipated use of valves is regulation of the transcription of arrays of genetic parts. However before predictive engineering of valve arrays *in vitro* can be completed, some investigations must be completed to understand and model how termination efficiency is affected by proximity to the promoter. Once this is done, the regulation of RNA tools such as gRNAs using valves could be extended to application *in vivo*. CRISPR gRNAs to modulate multiple regions of genomes are increasingly transcribed together on arrays [34, 223], making this application potentially useful. The T7 RNA polymerase valves could be used for expressing RNA using orthogonal transcription with T7 RNAP in a range of cellular hosts. By comparing these results to our *in vitro* characterisation, valves could be used to reveal how transcription, translation and mRNA degradation interact [284].

### 8.1.4 Responsible bio-engineering

The MCT and research journey are tools that can be used to anticipate the impacts of this research and reflect upon them. They could prove useful for other doctoral researchers seeking to research responsibly. Furthermore, they open conversations about the research objectives and direction. They may lead to consideration of the impacts of the mode of innovation and alternative innovation strategies. More tools and allocated time and resources for researchers to reflect upon their research can help researchers to guide the trajectory of their research reflexively, in response to their experiences. Responsibility when engineering living organisms and technologies is being actively discussed [137, 204, 205]. *In vitro* tools such as those developed in this thesis offer a means of undertaking synthetic biology without having to engineer living organisms. Nonetheless, since any technology can disrupt social, economic, ethical and environmental systems [290], there is a case for considering their wider impacts during development.

## 8.2 Outlook

This work illuminates a new approach to genetic design for synthetic biology: using transcriptional terminators as valves to tune transcriptional flows. Our results provide a starting point for further investigations into how to tune and insulate terminators using base-pairing and RNA structure

which will continue to reveal how sequence context affects terminators. Furthermore, the simple and inexpensive methods for creating and nanopore sequencing large libraries of genetic parts open possibilities for multiplexed characterisation of large genetic designs not possible with current techniques. The potential of nanopore DNA sequencing that we demonstrate are likely to inspire evaluation of the composition and mutations of various DNA assembly methods and even to study burden within genetic designs. This may lead to unprecedented utilisation of the many forms of natural variation to diversify libraries of genetic designs. Nanopore direct RNA sequencing could also be used for detailed characterisation of libraries of terminators and other genetic parts in a variety of biophysical contexts, guided by the limitations that we have highlighted. The use of these datasets to predict function from sequence is probable and could lead to more predictive genetic design. Our use of transcriptional valves to control CRISPR gRNAs may offer a way to control several genes simultaneously. The approaches presented for considering the wider impacts of biotechnologies may guide the development of this and other synthetic biology tools. Taken together, the outlook beyond this work is judicious application of multiplexed nanopore sequencing and genetic parts, pending the practicality of molecular biotechnology in the midst of ongoing crises of the environment, economy and equity.

APPENDIX A

# APPENDIX

## A.1 Abbreviations

| Abbreviation | Meaning |
|---|---|
| CRISPR | clustered regularly interspaced palindromic repeats |
| DNA | deoxyribonucleic acid |
| DNA-seq | DNA sequencing |
| dRNA-seq | direct RNA sequencing |
| FACS | fluorescence activated cell sorting |
| GFP | green fluorescent protein |
| gRNA | guide RNA |
| indel | insertion or deletion |
| MCT | matrix of convivial technology |
| nt | nucleotide |
| PCR | polymerase chain reaction |
| RFP | red fluorescent protein |
| RNA | ribonucleic acid |
| RNAP | RNA polymerase |
| RNA-seq | RNA sequencing |
| RRI | responsible research and innovation |
| RT | reverse transcriptase |
| SNP | single nucleotide polymorphism |
| sort-seq | FACS followed by sequencing |
| TX | transcription |

Table A.1: **Abbreviations.**

## A.2 Libraries and genetic part sequences

| **Library** | **Description** | **Parts used for assembly or specific designs** | **Total designs** |
|---|---|---|---|
| L1 | Initial set to test methodology | S1, S2, S3, S4, S5<br><br>M1N, M1A, M1U, M1S, M1T, M1X, M2N, M2A, M2U, M2S, M2T, M2X, M3N, M3A, M3U, M3S, M3T, M3X<br><br>T2, T3, T4, T5, T6, T7 | 540 |
| L2 | Random library to explore all design parameters | S10, S16, S18, S19, S20, S21, S22<br><br>M10, M11, M12, M13, M14, M15, M16, M17, M18, M19, M20, M21, M22<br><br>T10, T12, T13, T14, T15, T16, T17, T18, T20, T21, T27, T29, T33 | 1183 |
| L3 | Library focused on modifier design principles | S10, S16, S18, S19, S20, S21, S22<br><br>M50, M51, M52, M53, M54, M55, M56, M57, M58, M59, M60, M61, M62, M63, M64, M65, M66, M67, M68, M69, M70, M71, M72, M73, M74, , M76, M77, M78, M79, M80, M81, M82, M83, M84, M85, M86, M87, M88, M89, M90, M91<br><br>T10, T16, T29, T99 | 1260 |
| L4 | Library focused on terminator design principles | T50, T51, T52, T53, T54, T55, T56, T57, T58, T59, T60, T61, T62, T63, T64, T65, T66, T67, T68, T69, T70, T71, T72, T73, T74, T75*, T76, T77, T78, T79, T80, T81, T82, T83, T84, T85, T86, T87, T88, T89, T90 | 41 |
| L5 | CRISPR gRNA arrays | S20-M22-T29, S21-M10-T16, S10-M17-T20, S16-M12-T13, S18-M18-T33, S19-M19-T34 | 6 |

* T75 also referred to as T99U

Table A.2: **List of designed libraries**

| **Reference** | **Sequence** |
|---|---|
| pGRmT7 | cagagcgaggtatgtaggcggtgctacagagttcttgaagtggtggcctaactacggctacacta<br>gaaggacagtatttggtatctgcgctctgctgaagccagttaccttcggaaaaagagttggtagc<br>tcttgatccggcaaacaaaccaccgctggtagcggtggtttttttgtttgcaagcagcagattac<br>gcgcagaaaaaaaggatctcaagaagatcctttgatcttttctacggggtctgacgctcagtgga<br>acgaaaactcacgttaagggattttggtcatgagattatcaaaaaggatcttcacctagatcctt<br>ttaaattaaaaatgaagttttaaatcaatctaaagtatatatgagtaaacttggtctgacagtta<br>ccaatgcttaatcagtgaggcacctatctcagcgatctgtctatttcgttcatccatagttgcct<br>gactccccgtcgtgtagataactacgatacgggagggcttaccatctggccccagtgctgcaatg<br>ataccgcgagacccacgctcaccggctccagatttatcagcaataaaccagccagccggaagggc<br>cgagcgcagaagtggtcctgcaactttatccgcctccatccagtctattaattgttgccgggaag<br>ctagagtaagtagttcgccagttaatagtttgcgcaacgttgttgccattgctacaggcatcgtg<br>gtgtcacgctcgtcgtttggtatggcttcattcagctccggttcccaacgatcaaggcgagttac<br>atgatcccccatgttgtgcaaaaaagcggttagctccttcggtcctccgatcgttgtcagaagta<br>agttggccgcagtgttatcactcatggttatggcagcactgcataattctcttactgtcatgcca<br>tccgtaagatgcttttctgtgactggtgagtactcaaccaagtcattctgagaatagtgtatgcg<br>gcgaccgagttgctcttgcccggcgtcaatacgggataataccgcgccacatagcagaactttaa<br>aagtgctcatcattggaaaacgttcttcggggcgaaaactctcaaggatcttaccgctgttgaga<br>tccagttcgatgtaacccactcgtgcacccaactgatcttcagcatcttttactttcaccagcgt<br>ttctgggtgagcaaaaacaggaaggcaaaatgccgcaaaaaagggaataagggcgacacggaaat<br>gttgaatactcatactcttccttttcaatattattgaagcatttatcagggttattgtctcatg<br>agcggatacatatttgaatgtatttagaaaaataaacaaataggggttccgcgcacatttccccg<br>aaaagtgccacctgacgtCTAATACGACTCACTATAGGGAGAGctagctactagagaaagaggag<br>aaatactagatgcgtaaaggagaagaacttttcactggagttgtcccaattcttgttgaattaga<br>tggtgatgttaatgggcacaaattttctgtcagtggagagggtgaaggtgatgcaacatacggaa<br>aacttacccttaaatttatttgcactactggaaaactacctgttccatggccaacacttgtcact<br>actttcggttatggtgttcaatgctttgcgagatacccagatcatatgaaacagcatgacttttt<br>caagagtgccatgcccgaaggttatgtacaggaaagaactatatttttcaaagatgacgggaact<br>acaagacacgtgctgaagtcaagtttgaaggtgatacccttgttaatagaatcgagttaaaaggt<br>attgattttaaagaagatggaaacattcttggacacaaattggaatacaactataactcacacaa<br>tgtatacatcatggcagacaaacaaaagaatggaatcaaagttaacttcaaaattagacacaaca<br>ttgaagatggaagcgttcaactagcagaccattatcaacaaaatactccaattggcgatggccct<br>gtcctttaccagacaaccattacctgtccacacaatctgccctttcgaaagatcccaacgaaaa<br>gagagaccacatggtccttcttgagtttgtaacagctgctgggattacacatggcatggatgaac<br>tatacaaaTTAGCAgaattcactagtagcggccgctgcagaaagaggagaaatactagatggctt<br>cctccgaagacgttatcaaagagttcatgcgtttcaaagttcgtatggaaggttccgttaacggt<br>cacgagttcgaaatcgaaggtgaaggtgaaggtcgtccgtacgaaggtacccagaccgctaaact<br>gaaagttaccaaaggtggtccgctgccgttcgcttgggacatcctgtccccgcagttccagtacg<br>gttccaaagcttacgttaaacacccggctgacatcccggactacctgaaactgtccttcccggaa<br>ggtttcaaatgggaacgtgttatgaacttcgaagacggtggtgttgttaccgttacccaggactc<br>ctccctgcaagacggtgagttcatctacaaagttaaactgcgtggtaccaacttcccgtccgacg<br>gtccggttatgcagaaaaaaaccatgggttgggaagcttccaccgaacgtatgtacccggaagac<br>ggtgctctgaaaggtgaaatcaaaatgcgtctgaaactgaaagacggtggtcactacgacgctga<br>agttaaaaccacctacatggctaaaaaaccggttcagctgccgggtgcttacaaaaccgacatca<br>aactggacatcacctcccacaacgaagactacaccatcgttgaacagtacgaacgtgctgaaggt<br>cgtcactccaccggtgcttaataacgctgatagtgctagtgtagatcgctactagagccaggcat<br>caaataaaacgaaaggctcagtcgaaagactgggcctttcgttttatctgttgtttgtcggtgaa<br>cgctctctactagagtcacactggctcaccttcgggtgggcctttctgcgtttatatactagagt<br>caattgaggtagaggtacacacgcgaactccgatagccaattcagagtaataaactgtgataatc<br>aatgcagtgcttcctcgctcactgactcgctgcgctcggtcgttcggctgcggcgagcggtatca<br>gctcactcaaaggcggtaatacggttatccacagaatcaggggataacgcaggaaagaacatgtg<br>agcaaaaggccagcaaaaggccaggaaccgtaaaaaggccgcgttgctggcgtttttccataggc<br>tccgcccccctgacgagcatcacaaaaatcgacgctcaagtcagaggtggcgaaacccgacagga<br>ctataaagataccaggcgtttcccccctggaagctccctcgtgcgctctcctgttccgaccctgcc<br>gcttaccggatacctgtccgcctttctcccttcgggaagcgtggcgctttctcatagctcacgct<br>gtaggtatctcagttcggtgtaggtcgttcgctccaagctgggctgtgtgcacgaacccccccgtt<br>cagcccgaccgctgcgccttatccggtaactatcgtcttgagtccaacccggtaagacacgactt<br>atcgccactggcagcagccactggtaacaggattag<br>***(T7 promoter and stop codon mutation are capitalised)*** |

| | |
|---|---|
| pVGA010 | AGGTAACTAGAGGGACTGCGACGTttaattaaTAATACGACTCACTATAGGGAGAGGATCCCTGC<br>CGTATAGGCAGATCAGTGTGTACTAAGTACTATCTGAGAGCCAAAAATGGCAAGTTCAGATAAGG<br>CCAGACCGTTACCAGCTTAAATAAGCGCTGCCGTATAGGCAGGAATTCTTCAAAGCTACGAGCGC<br>TAGAGATGTGAGACCCTAAGAGCATTCGCTGAGAGTTACACGATACTGACTATTGCCCAGCGTTG<br>AACCTACGACAGTCTCTTATTGACGAGTAAAGTGCTAAGCTTCTGCCGTATAGGCAGGACACATC<br>TTAGAGTATGTAGGAATAGAAAACAAAAGTTTAAGTTATTCTAAGGCCAGTCCGGAATCATCCTA<br>AAAAGGAGCTGCCGTATAGGCAGGTCGACCTAATTATGTCTCAAAAGCTCGAAGATTACACCTAA<br>GAGCTTGAAATCGGATACTTCCTGAACTGCGAATTGCCAAATAGTTACCGAAAGTGTCCTGACCC<br>AGTTGAGGCGTTTACTCTCTAGACTGCCGTATAGGCAGTCTCAAGCTAGACTCTAGTGCATTTTG<br>GCGTCGAAAGACGAAGTAAAATGAAGGCGAGACCGATATCAACTGGAAGCAGTGCTGCCGTATAG<br>GCAGCCCGGGCCTAGCATAACCCCGCGGGGCCTCTTCGGGGGTCTCGCGGGGTTTTTTGCTGAAA<br>GAACTAGTTCTAAACGTTGGTCC |
| pVGA011 | AGGTAACTAGAGGGACTGCGACGTttaattaaTAATACGACTCACTATAGGGAGAGGATCCCTGC<br>CGTATAGGCAGATCAGTGTGTACTAAGTACTATCTGAGAGCCAAAAATGGCAAGTTCAGATAAGG<br>CCAGACCGTTACCAGCTTAAATAAGCGCTGCCGTATAGGCAGGAATTCTTGTCGCTAAAGAAACC<br>TTTCCCAATTAATACATAAGAGCAAGGCGTGACTACAACCAATCTTCTATTCTGCGAGAGTAAAG<br>TTTTGCCCAGAAATCATCCTTAGCGAAAGCTAAGGATTTTTTTTATCTGAAAAGCTTCTGCCGTA<br>TAGGCAGGACACATCTTAGAGTATGTAGGAATAGAAAACAAAAGTTTAAGTTATTCTAAGGCCAG<br>TCCGGAATCATCCTAAAAAGGAGCTGCCGTATAGGCAGGTCGACGGAATCGCTGATCTACAGAAC<br>GGTCCTTATGGGTAAGAGCTTTCTCCGAAGTGTAGTAAAAAAATAAAAATGCCGTTATTAAATAG<br>CCTGCCATCTGGCAGGCTTTTTTTATCGTCTAGACTGCCGTATAGGCAGTCTCAAGCTAGACTCT<br>AGTGCATTTTGGCGTCGAAAGACGAAGTAAAATGAAGGCGAGACCGATATCAACTGGAAGCAGTG<br>CTGCCGTATAGGCAGCCCGGGCCTAGCATAACCCCGCGGGGCCTCTTCGGGGGTCTCGCGGGGTT<br>TTTTGCTGAAAGAACTAGTTCTAAACGTTGGTCC |
| pVGA012 | AGGTAACTAGAGGGACTGCGACGTttaattaaTAATACGACTCACTATAGGGAGAGGATCCCTGC<br>CGTATAGGCAGATCAGTGTGTACTAAGTACTATCTGAGAGCCAAAAATGGCAAGTTCAGATAAGG<br>CCAGACCGTTACCAGCTTAAATAAGCGCTGCCGTATAGGCAGGAATTCCTGTGTACCGGGAACCA<br>GCCAGACTACACAGGGTAAGAGCGTATCCAGACTTATTGAGGTTTACGCACTATGCCCTGAAATA<br>TCCAGCGGATCAAGAAAATTCGTTGGATATTTTTTAAGCTTCTGCCGTATAGGCAGGACACATCT<br>TAGAGTATGTAGGAATAGAAAACAAAAGTTTAAGTTATTCTAAGGCCAGTCCGGAATCATCCTAA<br>AAAGGAGCTGCCGTATAGGCAGGTCGACGTGCAGAGACAAGCGTTTGGGGCACCAGCACAGTAAG<br>AGCCAGGAACTTATCAATAGTCGCCCGAAAGGGTGCCAGTTAACCAAAAAGGGGGGATTTTATCT<br>CCCCTTTAATTTTTCCTTCTAGACTGCCGTATAGGCAGTCTCAAGCTAGACTCTAGTGCATTTTG<br>GCGTCGAAAGACGAAGTAAAATGAAGGCGAGACCGATATCAACTGGAAGCAGTGCTGCCGTATAG<br>GCAGCCCGGGCCTAGCATAACCCCGCGGGGCCTCTTCGGGGGTCTCGCGGGGTTTTTTGCTGAAA<br>GAACTAGTTCTAAACGTTGGTCC |
| pVGA013 | AGGTAACTAGAGGGACTGCGACGTttaattaaTAATACGACTCACTATAGGGAGAGGATCCCTGC<br>CGTATAGGCAGATCAGTGTGTACTAAGTACTATCTGAGAGCCAAAAATGGCAAGTTCAGATAAGG<br>CCAGACCGTTACCAGCTTAAATAAGCGCTGCCGTATAGGCAGGAATTCATCACTCACACATCGCT<br>CGAGATCGGTACGGGGTAAGAGCATAGCCGAGATTATCCACCAGCAACAGTTCGTTATTGTAGTG<br>ATTTGCCGCTGATGCCAGAAAGGGTCCTGAATTTCAGGGCCCTTTTTTTACATGGATTGAAGCTT<br>CTGCCGTATAGGCAGGACACATCTTAGAGTATGTAGGAATAGAAAACAAAAGTTTAAGTTATTCT<br>AAGGCCAGTCCGGAATCATCCTAAAAAGGAGCTGCCGTATAGGCAGGTCGACCTGTGTACCGGGA<br>ACCAGCCAGACTACACAGGGTAAGAGCGATTACAGAAGCGTGGTATTTTTTATTTTTTGCCACTG<br>ATTTTTAAGGCGACTGATGAGTCGCCTTTTTTTTGTCTTCTAGACTGCCGTATAGGCAGTCTCAA<br>GCTAGACTCTAGTGCATTTTGGCGTCGAAAGACGAAGTAAAATGAAGGCGAGACCGATATCAACT<br>GGAAGCAGTGCTGCCGTATAGGCAGCCCGGGCCTAGCATAACCCCGCGGGGCCTCTTCGGGGGTC<br>TCGCGGGGTTTTTTGCTGAAAGAACTAGTTCTAAACGTTGGTCC |

Table A.3: **List of plasmid and array sequences**

| ID | Forward strand oligonucleotide sequence (5'–3') | Reverse strand oligonucleotide sequence (5'–3') | Library | Description |
|---|---|---|---|---|
| pT7 | CTAATACGACTCACTATAGGGAGAG | CTAGCTCTCCCTATAGTGAGTCGTATTAGACGT | – | Promoter |
| S10 | AATTCCTGTGTACCGGGAACCAGCCAGACTACACAGGGTAA | GCTCTTACCCTGTGTAGTCTGGCTGGTTCCCGGTACACAGG | L2 | Spacer |
| S16 | AATTCGTGCAGAGACAAGCGTTTGGGGCACCAGCACAGTAA | GCTCTTACTGTGCTGGTGCCCCAAACGCTTGTCTCTGCACG | L2 | Spacer |
| S18 | AATTCTTCAAAGCTACGAGCGCTAGAGATGTGAGACCCTAA | GCTCTTAGGGTCTCACATCTCTAGCGCTCGTAGCTTTGAAG | L2 | Spacer |
| S19 | AATTCCTAATTATGTCTCAAAAGCTCGAAGATTACACCTAA | GCTCTTAGGTGTAATCTTCGAGCTTTTGAGACATAATTAGG | L2 | Spacer |
| S20 | AATTCTTGTCGCTAAAGAAACCTTTCCCAATTAATACATAA | GCTCTTATGTATTAATTGGGAAAGGTTTCTTTAGCGACAAG | L2 | Spacer |
| S21 | AATTCGGAATCGCTGATCTACAGAACGGTCCTTATGGGTAA | GCTCTTACCCATAAGGACCGTTCTGTAGATCAGCGATTCCG | L2 | Spacer |
| S22 | AATTCATCACTCACACATCGCTCGAGATCGGTACGGGGTAA | GCTCTTACCCCGTACCGATCTCGAGCGATGTGTGAGTGATG | L2 | Spacer |
| M10 | GAGCTTTCTCCGAAGTGTAGTAAAAAAATAAAAA | GGCATTTTTATTTTTTTACTACACTTCGGAGAAA | L2 | Modifier |
| M11 | GAGCGATTACAGAAGCGTGGTATTTTTTATTTTT | GGCAAAAAATAAAAAATACCACGCTTCTGTAATC | L2 | Modifier |
| M12 | GAGCCAGGAACTTATCAATAGTCGCCCGAAAGGG | GGCACCCTTTCGGGCGACTATTGATAAGTTCCTG | L2 | Modifier |
| M13 | GAGCCCTATTTACCTCAGT | GGCAACTGAGGTAAATAGG | L2 | Modifier |
| M14 | GAGCTAGACAGTAATACCC | GGCAGGGTATTACTGTCTA | L2 | Modifier |
| M15 | GAGCCTATCTGGTGCTACA | GGCATGTAGCACCAGATAG | L2 | Modifier |
| M16 | GAGCTTATCGGTTACCAGA | GGCATCTGGTAACCGATAA | L2 | Modifier |
| M17 | GAGCGTATCCAGACTTATTGAGGTTTACGCACTA | GGCATAGTGCGTAAACCTCAATAAGTCTGGATAC | L2 | Modifier |
| M18 | GAGCATTCGCTGAGAGTTACACGATACTGACTAT | GGCAATAGTCAGTATCGTGTAACTCTCAGCGAAT | L2 | Modifier |
| M19 | GAGCTTGAAATCGGATACTTCCTGAACTGCGAAT | GGCAATTCGCAGTTCAGGAAGTATCCGATTTCAA | L2 | Modifier |
| M20 | GAGCATAGACTTTCGTGGATTATTACCTTACAACTGATAGGACGGACTC | GGCAGAGTCCGTCCTATCAGTTGTAAGGTAATAATCCACGAAAGTCTAT | L2 | Modifier |
| M21 | GAGCATAGCCGAGATTATCCACCAGCAACAGTTCGTTATTGTAGTGATT | GGCAAATCACTACAATAACGAACTGTTGCTGGTGGATAATCTCGGCTAT | L2 | Modifier |
| M22 | GAGCAAGGCGTGACTACAACCAATCTTCTATTCTGCGAGAGTAAAGTTT | GGCAAAACTTTACTCTCGCAGAATAGAAGATTGGTTGTAGTCACGCCTT | L2 | Modifier |
| T10 | TGCCGCTGATGCCAGAAAGGGTCCTGAATTTCAGGGCCCTTTTTTTACATGGATTGA | CTAGTCAATCCATGTAAAAAAAGGGCCCTGAAATTCAGGACCCTTTCTGGCATCAGC | L2 | Terminator |
| T12 | TGCCACTGATTTTTAAGGCGACTGATGAGTCGCCTTTTTTTTGTCTA | CTAGTAGACAAAAAAAAGGCGACTCATCAGTCGCCTTAAAAATCAGT | L2 | Terminator |
| T13 | TGCCAGTTAACCAAAAAGGGGGGATTTTATCTCCCCTTTAATTTTTCCTA | CTAGTAGGAAAAATTAAAGGGGAGATAAAATCCCCCCTTTTTGGTTAACT | L2 | Terminator |
| T14 | TGCCCGTGTTCCTGAACGCCCGCATATGCGGGCGTTTTGCTTTTTGA | CTAGTCAAAAAGCAAAACGCCCGCATATGCGGGCGTTCAGGAACACG | L2 | Terminator |
| T15 | TGCCTCTGAATGCGTGCCCATTCCTGACGGAATGGGCATTTCTGCGCAA | CTAGTTGCGCAGAAATGCCCATTCCGTCAGGAATGGGCACGCATTCAGA | L2 | Terminator |

| | | | | |
|---|---|---|---|---|
| T16 | TGCCGTTATTAAATAGCCTGCCATCTGGCAGGCTTTTTTTATCGA | CTAGTCGATAAAAAAAGCCTGCCAGATGGCAGGCTATTTAATAAC | L2 | Terminator |
| T17 | TGCCCGTCTGCGTATGGAACGTGGTAACGGTTCTACTGAAGATTTA | CTAGTAAATCTTCAGTAGAACCGTTACCACGTTCCATACGCAGACG | L2 | Terminator |
| T18 | TGCCTACTTCTTACTCGCCCATCTGCAACGGATGGGCGAATTTATACCCA | CTAGTGGGTATAAATTCGCCCATCCGTTGCAGATGGGCGAGTAAGAAGTA | L2 | Terminator |
| T20 | TGCCCTGAAATATCCAGCGGATCAAGAAAATTCGTTGGATATTTTTTA | CTAGTAAAAAATATCCAACGAATTTTCTTGATCCGCTGGATATTTCAG | L2 | Terminator |
| T21 | TGCCAAACACGTAGGCCTGATAAGCGAAGCGCATCAGGCAGTTTTGCGTA | CTAGTACGCAAAACTGCCTGATGCGCTTCGCTTATCAGGCCTACGTGTTT | L2 | Terminator |
| T27 | TGCCTTTCAGCAAAAAACCCCTCAAGACCCGTTTAGAGGCCCCAAGGGGTTATGCTAGGA | CTAGTCCTAGCATAACCCCTTGGGGCCTCTAAACGGGTCTTGAGGGGTTTTTTGCTGAAA | L2 | Terminator |
| T29 | TGCCCAGAAATCATCCTTAGCGAAAGCTAAGGATTTTTTTTATCTGAAA | CTAGTTTCAGATAAAAAAAATCCTTAGCTTTCGCTAAGGATGATTTCTG | L2 | Terminator |
| T33 | TGCCCAGCGTTGAACCTACGACAGTCTCTTATTGACGAGTAAAGTGCTA | CTAGTAGCACTTTACTCGTCAATAAGAGACTGTCGTAGGTTCAACGCTG | L2 | Terminator |
| S1 | AATTCGACTTTCACGTGAACCTGTTCCCAATATAA | GCTCTTATATTGGGAACAGGTTCACGTGAAAGTCG | L1 | Spacer |
| S2 | AATTCAATGTGGAACTCTTCGCTCATGTAGAATAA | GCTCTTATTCTACATGAGCGAAGAGTTCCACATTG | L1 | Spacer |
| S3 | AATTCGGTGCAGCGGAGAAAAGATTTGCTACCTAA | GCTCTTAGGTAGCAAATCTTTTCTCCGCTGCACCG | L1 | Spacer |
| S4 | AATTCCTTGATATAAAACTTCCGGGAGTAGGATAA | GCTCTTATCCTACTCCCGGAAGTTTTATATCAAGG | L1 | Spacer |
| S5 | AATTCCAAGAACTCGTTTTCCTATATGGCGTCTAA | GCTCTTAGACGCCATATAGGAAAACGAGTTCTTGG | L1 | Spacer |
| M1N | GAGCTTTCTCCGAAGTGTAGTAAATAAAGCGTCC | GGCAGGACGCTTTATTTACTACACTTCGGAGAAA | L1 | Modifier |
| M1A | GAGCTTTCTCCGAAGTGTAGTAAATTTTATTTTT | GGCAAAAAATAAAATTTACTACACTTCGGAGAAA | L1 | Modifier |
| M1U | GAGCTTTCTCCGAAGTGTAGTAAAAAAATAAAAA | GGCATTTTTATTTTTTTACTACACTTCGGAGAAA | L1 | Modifier |
| M1S | GAGCTTTCTCCGAAGTGTAGTAAACCCGAAAGGG | GGCACCCTTTCGGGTTTACTACACTTCGGAGAAA | L1 | Modifier |
| M1T | GAGCTTTCTCCGAAGTGTAGTAAA | GGCATTTACTACACTTCGGAGAAA | L1 | Modifier |
| M1X | GAGCTTTCTCCGAAG | GGCACTTCGGAGAAA | L1 | Modifier |
| M2N | GAGCAAGGACTTTCTCTACTGATTGTAAGACCGA | GGCATCGGTCTTACAATCAGTAGAGAAAGTCCTT | L1 | Modifier |
| M2A | GAGCAAGGACTTTCTCTACTGATTTTTTATTTTT | GGCAAAAAATAAAAAATCAGTAGAGAAAGTCCTT | L1 | Modifier |
| M2U | GAGCAAGGACTTTCTCTACTGATTAAAATAAAAA | GGCATTTTTATTTTAATCAGTAGAGAAAGTCCTT | L1 | Modifier |
| M2S | GAGCAAGGACTTTCTCTACTGATTCCCGAAAGGG | GGCACCCTTTCGGGAATCAGTAGAGAAAGTCCTT | L1 | Modifier |
| M2T | GAGCAAGGACTTTCTCTACTGATT | GGCAAATCAGTAGAGAAAGTCCTT | L1 | Modifier |
| M2X | GAGCAAGGACTTTCT | GGCAAGAAAGTCCTT | L1 | Modifier |
| M3N | GAGCCAGGAACTTATCAATAGTCGTTGTGACACT | GGCAAGTGTCACAACGACTATTGATAAGTTCCTG | L1 | Modifier |
| M3A | GAGCCAGGAACTTATCAATAGTCGTTTTATTTTT | GGCAAAAAATAAAACGACTATTGATAAGTTCCTG | L1 | Modifier |
| M3U | GAGCCAGGAACTTATCAATAGTCGAAAATAAAAA | GGCATTTTTATTTTCGACTATTGATAAGTTCCTG | L1 | Modifier |

| | | | | |
|---|---|---|---|---|
| M3S | GAGCCAGGAACTTATCAATAGTCGCCCGAAAGGG | GGCACCCTTTCGGGCGACTATTGATAAGTTCCTG | L1 | Modifier |
| M3T | GAGCCAGGAACTTATCAATAGTCG | GGCACGACTATTGATAAGTTCCTG | L1 | Modifier |
| M3X | GAGCCAGGAACTTAT | GGCAATAAGTTCCTG | L1 | Modifier |
| T2 | TGCCCGTAAAAACCCGCCGAAGCGGGTTTTTACGTAACA | CTAGTGTTACGTAAAAACCCGCTTCGGCGGGTTTTTACG | L1 | Terminator |
| T3 | TGCCAGTAAAAACCCGCCGAAGCGGGTTTTTACGTAACA | CTAGTGTTACGTAAAAACCCGCTTCGGCGGGTTTTTACT | L1 | Terminator |
| T4 | TGCCAAAAAAAACACCCTAACGGGTGTTTTTTTTTTTA | CTAGTAAAAAAAAAAAACACCCGTTAGGGTGTTTTTTTTT | L1 | Terminator |
| T5 | TGCCAGAATTCAGTCAAAAGCCTCCGACCGGAGGCTTTTGACTATTACTACTAGA | CTAGTCTAGTAGTAATAGTCAAAAGCCTCCGGTCGGAGGCTTTTGACTGAATTCT | L1 | Terminator |
| T6 | TGCCAGAATTCAGCCCGCCTAATGAGCGGGCTTTTTTTTACTAA | CTAGTTAGTAAAAAAAAGCCCGCTCATTAGGCGGGCTGAATTCT | L1 | Terminator |
| T7 | TGCCAGAAAAGAGGCCTCCCGAAAGGGGGGCCTTTTTTCGTTTTA | CTAGTAAAACGAAAAAAGGCCCCCCTTTCGGGAGGCCTCTTTTCT | L1 | Terminator |
| T50 | AATTCTAAAAATATCCAACGAATTTTCTTGATCCGCTGGATATTTTTTTTCAGA | CTAGTCTGAAAAAAAATATCCAGCGGATCAAGAAAATTCGTTGGATATTTTTAG | L4 | preprint+u-tract,T20_preprint |
| T51 | AATTCTAAagttaaccaaAAAGGGGGGATTTTATCTCCCCTTTttttttcctA | CTAGTaggaaaaaAAAGGGGAGATAAAATCCCCCCTTTttggttaactTTAG | L4 | preprint+u-tract,T13_preprint |
| T52 | AATTCTAAcgtgttcctgAACGCCCGCATATGCGGGCGTTtttttttgA | CTAGTcaaaaaaaAACGCCCGCATATGCGGGCGTTcaggaacacgTTAG | L4 | preprint+u-tract,T14_preprint |
| T53 | AATTCTAAcgtctgcgtaTGGAACGTGGTAACGGTTCTATTTTTTTTctgaagatttA | CTAGTaaatcttcagAAAAAAAAATAGAACCGTTACCACGTTCCAtacgcagacgTTAG | L4 | preprint+u-tract,T17_preprint |
| T54 | AATTCTAAtacttcttacTCGCCCATCTGCAACGGATGGGCGATTTTTTtttatacccA | CTAGTgggtataaaAAAAATCGCCCATCCGTTGCAGATGGGCGAgtaagaagtaTTAG | L4 | preprint+u-tract,T18_preprint |
| T55 | AATTCTAAaaacacgtagGCCTGATAAGCGAAGCGCATCAGGCTTTTtttgcgtA | CTAGTacgcaaaaAAAAGCCTGATGCGCTTCGCTTATCAGGCctacgtgtttTTAG | L4 | preprint+u-tract,T21_preprint |
| T56 | AATTCTAAtctgaatgcgTGCCCATTCCTGACGGAATGGGCATTTTTtttctgcgcaA | CTAGTtgcgcagaaaAAAAATGCCCATTCCGTCAGGAATGGGCAcgcattcagaTTAG | L4 | preprint+u-tract,T15_preprint |
| T57 | AATTCTAACAGCGTTGAACCTACGACAGTCTCTTATTGACGAGTAAAGTGCTA | CTAGTAGCACTTTACTCGTCAATAAGAGACTGTCGTAGGTTCAACGCTGTTAG | L4 | Terminator, Negative control, T33 |
| T58 | AATTCTAAAAATAGTTACCGAAAGTGTCCTGACCCAGTTGAGGCGTTTACTCA | CTAGTGAGTAAACGCCTCAACTGGGTCAGGACACTTTCGGTAACTATTTTTAG | L4 | Terminator, Negative control, T34 |
| T59 | AATTCTAAAAGACCCCCGCACCGAAAGGTCCGGGGGTTTTTTTTA | CTAGTAAAAAAAACCCCCGGACCTTTCGGTGCGGGGGTCTTTTAG | L4 | Terminator, *E. coli*, *ilvBN* |
| T60 | AATTCTAAcaacaatgacAAGCGGTGGAGATCTTCTCTGCCGCTTtttttttcatA | CTAGTatgaaaaaaaAAGCGGCAGAGAAGATCTCCACCGCTTgtcattgttgTTAG | L4 | Terminator, *E. coli*, ECK120015452 |
| T61 | AATTCTAAGTCAGTCGTCAGACGCCGGTTAATCCGGCGTTTTTTTTGACGCCCACA | CTAGTGTGGGCGTCAAAAAAAACGCCGGATTAACCGGCGTCTGACGACTGACTTAG | L4 | Terminator, *E. coli*, ECK120051408 |
| T62 | AATTCTAAAAAAAGTAACTAATGAGAAAAGCGCAGGGTGAAAGCCCTGCGCTTTTTCTTA | CTAGTAAGAAAAAGCGCAGGGCTTTCACCCTGCGCTTTTCTCATTAGTTACTTTTTTTAG | L4 | Terminator, *B. subtilis*, *rpmF* |
| T63 | AATTCTAAACTGAGTAATAGTATGGTTTTAAACGAGACCCCTGTGGGTCTCGTTTTTTGA | CTAGTCAAAAAACGAGACCCACAGGGGTCTCGTTTAAAACCATACTATTACTCAGTTTAG | L4 | Terminator, *B. subtilis*, *tufA* |
| T64 | AATTCTAAGAGGTGTAAGAAAAAAGCCAGAGCTTTGAAAAAGGTTCTGGCTTTTTTTCTA | CTAGTAGAAAAAAAGCCAGAACCTTTTTCAAAGCTCTGGCTTTTTTCTTACACCTCTTAG | L4 | Terminator, *B. subtilis*, *rpmGA* |
| T65 | AATTCTAACAGAGTAATCTGAAGCAACGTAAAAAAACCCGCCCCGGCGGGTTTTTTTATA | CTAGTATAAAAAAAACCCGCCGGGGCGGGTTTTTTTACGTTGCTTCAGATTACTCTGTTAG | L4 | Terminator, *E. coli*, *rpl* |

| | | | | |
|---|---|---|---|---|
| T66 | AATTCTAAAAATGCTTGATTAAAAAGGCGCTACTCGGCATGGGGAAGCGCCTTTTTTATA | CTAGTATAAAAAAGGCGCTTCCCCATGCCGAGTAGCGCCTTTTTAATCAAGCATTTTTAG | L4 | Terminator, *E. coli*, *fis* |
| T67 | AATTCTAAGGCCGCATATCAGCTTAAAAAATGAACCATCGCCAACGGCGGTGGTTTTTTA | CTAGTAAAAAACCACCGCCGTTGGCGATGGTTCATTTTTTTAAGCTGATATGCGGCCTTAG | L4 | Terminator, *E. coli*, *sod* |
| T68 | AATTCTAAATCTAAGCTCAATAAGAGGCTATCAGGCTTAACCGCTTGGTAGCCTTTTTGA | CTAGTCAAAAAGGCTACCAAGCGGTTAAGCCTGATAGCCTCTTATTGAGCTTAGATTTAG | L4 | Terminator, *V. natriegens*, *groE* |
| T69 | AATTCTAATTAGCTCTTAACTGAGTTGAAAAAGAGGCGGCTTTATAGTCGCCTTTTTTGA | CTAGTCAAAAAAGGCGACTATAAAGCCGCCTCTTTTTCAACTCAGTTAAGAGCTAATTAG | L4 | Terminator, *V. natriegens*, *PN96_1* |
| T70 | AATTCTAAGTAACGTTTTAAGTTAATAAGAAGCCCCGAGTTATGCTCGGGGCTTTTTGTA | CTAGTACAAAAAGCCCCGAGCATAACTCGGGGCTTCTTATTAACTTAAAACGTTACTTAG | L4 | Terminator, *V. natriegens*, *PN96_2* |
| T71 | AATTCTAAGATCGCTAGACTAAGAGACCCCGTCTTCCGAAAGGGAGGCGGGGTCTTTCTA | CTAGTAGAAAGACCCCGCCTCCCTTTCGGAAGACGGGGTCTCTTAGTCTAGCGATCTTAG | L4 | Terminator, *C. crescentus*, *saA* |
| T72 | AATTCTAAGCCTCTGACGATTCGAAAGCGCCGCCGGGTTTCGTCCCGGCGGCGCTTTTCA | CTAGTGAAAAGCGCCGCCGGGACGAAACCCGGCGGCGCTTTCGAATCGTCAGAGGCTTAG | L4 | Terminator, *C. crescentus*, *CNA_1* |
| T73 | AATTCTAACATAGCCCGTGAGCGAAACGCCCCGGAGGTCCGCCTCCGGGGCGTTTTTCTA | CTAGTAGAAAAACGCCCCGGAGGCGGACCTCCGGGGCGTTTCGCTCACGGGCTATGTTAG | L4 | Terminator, *C. crescentus*, *CNA_2* |
| T74 | AATTCTAATGGGGGGTATGGGGGGTATGGGGGGTATGGGGGGTATGGGGGGTATGGGGGA | CTAGTCCCCCATACCCCCCATACCCCCCATACCCCCCCATACCCCCCATACCCCCCATTAG | L4 | G-quadruplex |
| T75 | AATTCTAATACTACGCTATCCACTCCGCCCCCTTGGGGCCTCTAAACGGGTCTTGAGGGGTTTTTTTTTA | CTAGTAAAAAAAACCCCTCAAGACCCGTTTAGAGGCCCCAAGGGGGCGGAGTGGATAGCGTAGTATTAG | L4 | Terminator, tract variant, T-theta, poly-T |
| T76 | AATTCTAAATCTAAGATATGAAGGGAATCCCCTTGGGGCCTCTAAACGGGTCTTGAGGGGAAAAAAAAA | CTAGTTTTTTTTTCCCCTCAAGACCCGTTTAGAGGCCCCAAGGGGATTCCCTTCATATCTTAGATTTAG | L4 | Terminator, tract variant, T-theta, poly-A |
| T77 | AATTCTAAAGGCCAGATCTAGAAGCATGCCCCTTGGGGCCTCTAAACGGGTCTTGAGGGGCCCCCCCCA | CTAGTGGGGGGGGCCCCTCAAGACCCGTTTAGAGGCCCCAAGGGGCATGCTTCTAGATCTGGCCTTTAG | L4 | Terminator, tract variant, T-theta, poly-C |
| T78 | AATTCTAATCGACGACGTAACCGGCCTTCCCCTTGGGGCCTCTAAACGGGTCTTGAGGGGGGGGGGGGA | CTAGTCCCCCCCCCCCCTCAAGACCCGTTTAGAGGCCCCAAGGGGAAGGCCGGTTACGTCGTCGATTAG | L4 | Terminator, tract variant, T-theta, poly-G |
| T79 | AATTCTAATCTCCTGCATTCCTCGTACAGGGTCCTGAATTTCAGGGCCCTTTTTTTTA | CTAGTAAAAAAAAGGGCCCTGAAATTCAGGACCCTGTACGAGGAATGCAGGAGATTAG | L4 | Terminator, tract variant, T10, poly-T |
| T80 | AATTCTAATCTCTTCTATCCCGTCAAACGGGTCCTGAATTTCAGGGCCCAAAAAAAAA | CTAGTTTTTTTTTTGGGCCCTGAAATTCAGGACCCGTTTGACGGGATAGAAGAGATTAG | L4 | Terminator, tract variant, T10, poly-A |
| T81 | AATTCTAAGGCCTAATATCTCACCCTAAGGGTCCTGAATTTCAGGGCCCCCCCCCCCA | CTAGTGGGGGGGGGGGCCCTGAAATTCAGGACCCTTAGGGTGAGATATTAGGCCTTAG | L4 | Terminator, tract variant, T10, poly-C |
| T82 | AATTCTAATTAACAGAACACCAGAATCCGGGTCCTGAATTTCAGGGCCCGGGGGGGGA | CTAGTCCCCCCCCGGGCCCTGAAATTCAGGACCCGGATTCTGGTGTTCTGTTAATTAG | L4 | Terminator, tract variant, T10, poly-G |
| T83 | AATTCTAATATGTTAAATACCGTTCGCCTCCTTAGCGAAAGCTAAGGATTTTTTTTA | CTAGTAAAAAAAATCCTTAGCTTTCGCTAAGGAGGCGAACGGTATTTAACATATTAG | L4 | Terminator, tract variant, T29, poly-T |
| T84 | AATTCTAAACTGAACCACGAATACGCAATCCTTAGCGAAAGCTAAGGAAAAAAAAAA | CTAGTTTTTTTTTTCCTTAGCTTTCGCTAAGGATTGCGTATTCGTGGTTCAGTTTAG | L4 | Terminator, tract variant, T29, poly-A |
| T85 | AATTCTAACACTCGTTTATCCTTAACTATCCTTAGCGAAAGCTAAGGACCCCCCCCA | CTAGTGGGGGGGGTCCTTAGCTTTCGCTAAGGATAGTTAAGGATAAACGAGTGTTAG | L4 | Terminator, tract variant, T29, poly-C |
| T86 | AATTCTAAATTTCGTAATAACCTTTAGCTCCTTAGCGAAAGCTAAGGAGGGGGGGGA | CTAGTCCCCCCCCTCCTTAGCTTTCGCTAAGGAGCTAAAGGTTATTACGAAATTTAG | L4 | Terminator, tract variant, T29, poly-G |
| T87 | AATTCTAATGTCTGAATGCCGTTATCTCGCCTGCCATCTGGCAGGCTTTTTTTTA | CTAGTAAAAAAAAGCCTGCCAGATGGCAGGCGAGATAACGGCATTCAGACATTAG | L4 | Terminator, tract variant, T16, poly-T |
| T88 | AATTCTAAGAGCATACGAATAGAACATGGCCTGCCATCTGGCAGGCAAAAAAAAA | CTAGTTTTTTTTTGCCTGCCAGATGGCAGGCCATGTTCTATTCGTATGCTCTTAG | L4 | Terminator, tract variant, T16, poly-A |

| | | | | |
|---|---|---|---|---|
| T89 | AATTCTAAACGGGTAGAAGGATGACAACGCCTGCCATCTGGCAGGCCCCCCCCCA | CTAGTGGGGGGGGGCCTGCCAGATGGCAGGCGTTGTCATCCTTCTACCCGTTTAG | L4 | Terminator, tract variant, T16, poly-C |
| T90 | AATTCTAAAAGTGTGAGGTCAAATAAAGGCCTGCCATCTGGCAGGCGGGGGGGGA | CTAGTCCCCCCCCGCCTGCCAGATGGCAGGCCTTTATTTGACCTCACACTTTTAG | L4 | Terminator, tract variant, T16, poly-G |
| T99 | TGCCAACTAGCATAACCCCTTGGGGCCTCTAAACGGGTCTTGAGGGGTTTTTTGCA | CTAGTTGCAAAAAACCCCTCAAGACCCGTTTAGAGGCCCCAAGGGGTTATGCTAGTT | L4, L3 | Terminator, T-theta |
| M50 | GAGCTACTACGCTATCCACTCCGCTTCCCATCACGAGATCTTGAAATTC | GGCAGAATTTCAAGATCTCGTGATGGGAAGCGGAGTGGATAGCGTAGTA | L3 | Modifier, T10 Loop interactor (near) |
| M51 | GAGCATCTAAGATATGAAGGGAATTGGCACAACGACTGAACCAGGACCC | GGCAGGGTCCTGGTTCAGTCGTTGTGCCAATTCCCTTCATATCTTAGAT | L3 | Modifier, T10 Stem1 interactor (near) |
| M52 | GAGCAGGCCAGATCTAGAAGCATGCACGACAACCCTTTCACGGGCCCTG | GGCACAGGGCCCGTGAAAGGGTTGTCGTGCATGCTTCTAGATCTGGCCT | L3 | Modifier, T10 Stem2 interactor (near) |
| M53 | GAGCTCGACGACGTAACCGGCCTTCTCAACATTATACAAATCCAGATGG | GGCACCATCTGGATTTGTATAATGTTGAGAAGGCCGGTTACGTCGTCGA | L3 | Modifier, T16 Loop interactor (near) |
| M54 | GAGCTACTTTGTCTTCTCTACCACCTCTTGCCCGTATGCCTTGGCAGGC | GGCAGCCTGCCAAGGCATACGGGCAAGAGGTGGTAGAGAAGACAAAGTA | L3 | Modifier, T16 Stem1 (near) |
| M55 | GAGCGCATAAAGACGGGAGAAAGAGTTACAGCAAAGAGAAGGCCTGCCA | GGCATGGCAGGCCTTCTCTTTGCTGTAACTCTTTCTCCCGTCTTTATGC | L3 | Modifier, T16 Stem2 (near) |
| M56 | GAGCTAGTTACCCAACCGTGCGCATGTGATCCTGTTAAGATGCTTTCGC | GGCAGCGAAAGCATCTTAACAGGATCACATGCGCACGGTTGGGTAACTA | L3 | Modifier, T29 Loop interactor (near) |
| M57 | GAGCTCTCCTGCATTCCTCGTACATAGCTTAAACTTGATTTAAGGATGA | GGCATCATCCTTAAATCAAGTTTAAGCTATGTACGAGGAATGCAGGAGA | L3 | Modifier, T29 Stem1 interactor (near) |
| M58 | GAGCTCTCTTCTATCCCGTCAAACTAAGTACCAAACGTCATTCCTTAGC | GGCAGCTAAGGAATGACGTTTGGTACTTAGTTTGACGGGATAGAAGAGA | L3 | Modifier, T29 Stem2 interactor (near) |
| M59 | GAGCTGGTGCGTAGTAGACTTAACAAGATGTGATTTCGAAGCGTTTAGA | GGCATCTAAACGCTTCGAAATCACATCTTGTTAAGTCTACTACGCACCA | L3 | Modifier, T-theta Loop interactor (near) |
| M60 | GAGCGTGTAGGTATGTGTCGGTCGTTGTGGTGGTTTAGTGTCCAAGGGG | GGCACCCCTTGGACACTAAACCACCACAACGACCGACACATACCTACAC | L3 | Modifier, T-theta Stem1 interactor (near) |
| M61 | GAGCTAATTCTGAGCTAGACTATGATTCCCTCCAACAAATGCCCCTCAA | GGCATTGAGGGGCATTTGTTGGAGGGAATCATAGTCTAGCTCAGAATTA | L3 | Modifier, T-theta Stem2 interactor (near) |
| M62 | GAGCTGAAATTCGGCCTAATATCTCACCCTAAAGACTAATACTTCCCGC | GGCAGCGGGAAGTATTAGTCTTTAGGGTGAGATATTAGGCCGAATTTCA | L3 | Modifier, T10 Loop interactor (far) |
| M63 | GAGCCAGGACCCTTAACAGAACACCAGAATCCTAGCGAACCCAACCTCT | GGCAAGAGGTTGGGTTCGCTAGGATTCTGGTGTTCTGTTAAGGGTCCTG | L3 | Modifier, T10 Stem1 interactor (far) |
| M64 | GAGCGGGCCCTGGCCGAACGTCTAGCCCACCAAATGAAGCCTTAAGAGA | GGCATCTCTTAAGGCTTCATTTGGTGGGCTAGACGTTCGGCCAGGGCCC | L3 | Modifier, T10 Stem2 interactor (far) |
| M65 | GAGCCCAGATGGGAACCGCCATCTAACAGAAGTAAACATTACCCATCAG | GGCACTGATGGGTAATGTTTACTTCTGTTAGATGGCGGTTCCCATCTGG | L3 | Modifier, T16 Loop interactor (far) |
| M66 | GAGCTGGCAGGCTCCGCTGCACTCCCGTTAATCCCATATTATTCTTCAT | GGCAATGAAGAATAATATGGGATTAACGGGAGTGCAGCGGAGCCTGCCA | L3 | Modifier, T16 Stem1 interactor (far) |
| M67 | GAGCGCCTGCCATATGTTAAATACCGTTCGCCTGCTTAACCTACTTGAT | GGCAATCAAGTAGGTTAAGCAGGCGAACGGTATTTAACATATGGCAGGC | L3 | Modifier, T16 Stem2 interactor (far) |
| M68 | GAGCGCTTTCGCACTGAACCACGAATACGCAAATAACCCAGCTACCGAA | GGCATTCGGTAGCTGGGTTATTTGCGTATTCGTGGTTCAGTGCGAAAGC | L3 | Modifier, T29 Loop interactor (far) |
| M69 | GAGCAAGGATGACACTCGTTTATCCTTAACTATCTACCTTATCACTCTA | GGCATAGAGTGATAAGGTAGATAGTTAAGGATAAACGAGTGTCATCCTT | L3 | Modifier, T29 Stem1 interactor (far) |
| M70 | GAGCTCCTTAGCATTTCGTAATAACCTTTAGCATAGCATCACAGACTAC | GGCAGTAGTCTGTGATGCTATGCTAAAGGTTATTACGAAATGCTAAGGA | L3 | Modifier, T29 Stem2 |

| | | | | |
|---|---|---|---|---|
| | | | | interactor (far) |
| M71 | GAGCCGTTTAGATTAAGAGAGGAGATAGTCAATAACAGAACAAGAGCGT | GGCAACGCTCTTGTTCTGTTATTGACTATCTCCTCTCTTAATCTAAACG | L3 | Modifier, T-theta Loop interactor (far) |
| M72 | GAGCCCAAGGGGGCCGGATTCGAACATTCTACCTATTACACAGCTTTAA | GGCATTAAAGCTGTGTAATAGGTAGAATGTTCGAATCCGGCCCCCTTGG | L3 | Modifier, T-theta Stem1 interactor (far) |
| M73 | GAGCCCCCTCAAGGAGAGTTAATCGAAGAGAATCAGACACAAGGCGGAA | GGCATTCCGCCTTGTGTCTGATTCTCTTCGATTAACTCTCCTTGAGGGG | L3 | Modifier, T-theta Stem2 interactor (far) |
| M74 | GAGCCAGCCCAGAGTAGTATTTCTCCGAAGTGTAGTAAAAAAATAAAAA | GGCATTTTTATTTTTTTACTACACTTCGGAGAAATACTACTCTGGGCTG | L3 | Modifier, M11 A-tract interactor (near) |
| M75 | GAGCCGACGTGTTATCTAAGATTACAGAAGCGTGGTATTTTTTATTTTT | GGCAAAAAATAAAAAATACCACGCTTCTGTAATCTTAGATAACACGTCG | L3 | Modifier, M11 T-tract interactor (near) |
| M76 | GAGCAAAAAATAAAAAGGAGGAGCAGCATACTTACAAGACACGGGATAT | GGCAATATCCCGTGTCTTGTAAGTATGCTGCTCCTCCTTTTTATTTTTT | L3 | Modifier, A-tract interactor (far) |
| M77 | GAGCTTTTTTATTTTTTGTCTGAATGCCGTTATCTCTTCTCAATATGAA | GGCATTCATATTGAGAAGAGATAACGGCATTCAGACAAAAAAATAAAAAA | L3 | Modifier, T-tract interactor (far) |
| M78 | GAGCGAATAGAACATGGAACAATCAGCACTGAGCGGCACTTCGGTGCCG | GGCACGGCACCGAAGTGCCGCTCAGTGCTGATTGTTTCCATGTTCTATTC | L3 | Modifier, Long HP TTCG (near) |
| M79 | GAGCACGGGTAGAAGGATGACAACGGGAAGAGCGCACGGgagaCCGTGC | GGCAGCACGGtctcCCGTGCGCTCTTCCCGTTGTCATCCTTCTACCCGT | L3 | Modifier, Long HP GAGA (near) |
| M80 | GAGCAAGTGTGAGGTCAAATAAAGCAATAAGTCGGTGCCGAAAGGCACC | GGCAGGTGCCTTTCGGCACCGACTTATTGCTTTATTTGACCTCACACTT | L3 | Modifier, Long HP GAAA (near) |
| M81 | GAGCGTAGTAGATAAGGTGGATGTGGGACGAGCAGTGGGGCGTTCGCGC | GGCAGCGCGAACGCCCCACTGCTCGTCCCACATCCACCTTATCTACTAC | L3 | Modifier, Short HP TTCG (near) |
| M82 | GAGCGATTAGGAAGATTAGGCACATTACAGACTGGAACGCCCGAAAGGG | GGCACCCTTTCGGGCGTTCCAGTCTGTAATGTGCCTAATCTTCCTAATC | L3 | Modifier, M12 Short HP GAAA (near) |
| M83 | GAGCGGCTAAAGAGAAAGATTAACGTTCCACACACGCGACGCGAGAGCG | GGCACGCTCTCGCGTCGCGTGTGTGGAACGTTAATCTTTCTCTTTAGCC | L3 | Modifier, Short HP GAGA (near) |
| M84 | GAGCCGGCACTTCGGTGCCGTTTCCTTATTGCCCTCGCCCGTCGCCCGA | GGCATCGGGCGACGGGCGAGGGCAATAAGGAAACGGCACCGAAGTGCCG | L3 | Modifier, Long HP TTCG (far) |
| M85 | GAGCGCACGGgagaCCGTGCATAGAATAGTAACACAGCGCGCAAGGAAC | GGCAGTTCCTTGCGCGCTGTGTTACTATTCTATGCACGGtctcCCGTGC | L3 | Modifier, Long HP GAGA (far) |
| M86 | GAGCGGTGCCGAAAGGCACCATTAGTACACTCACGACCCGATCCCTTAG | GGCACTAAGGGATCGGGTCGTGAGTGTACTAATGGTGCCTTTCGGCACC | L3 | Modifier, Long HP GAAA (far) |
| M87 | GAGCGCGTTCGCGCAAGACCAATTTCAAAGCGGCGTCCTTATTTACCAT | GGCAATGGTAAATAAGGACGCCGCTTTGAAATTGGTCTTGCGCGAACGC | L3 | Modifier, Short HP TTCG (far) |
| M88 | GAGCTCGCTGCCCAACAACTTACTCTCGAATATGACACTCCCGAAAGGG | GGCACCCTTTCGGGAGTGTCATATTCGAGAGTAAGTTGTTGGGCAGCGA | L3 | Modifier, M12 Short HP GAAA (far) |
| M89 | GAGCCGCGAGAGCGGCAAGTTAAGACCGTACATCACTTCAAGAGAGCAC | GGCAGTGCTCTCTTGAAGTGATGTACGGTCTTAACTTGCCGCTCTCGCG | L3 | Modifier, Short HP GAGA (far) |
| M90 | GAGCCCGTTGTTGTTTGATCGGCAGATCTGAGCCTGGGAGCTCTCTGCC | GGCAGGCAGAGAGCTCCCAGGCTCAGATCTGCCGATCAAACAACAACGG | L3 | Modifier, Structure Pseudoknot1 |
| M91 | GAGCCCTGTGCCGAGGGCGCAGTGGGCTAGCGCCACTCAAAAGGCCCAT | GGCAATGGGCCTTTTGAGTGGCGCTAGCCCACTGCGCCCTCGGCACAGG | L3 | Modifier, Structure Pseudoknot2 |
| S10 | AATTCCTGTGTACCGGGAACCAGCCAGACTACACAGGGTAA | GCTCTTACCCTGTGTAGTCTGGCTGGTTCCCGGTACACAGG | L3 | Spacer |
| S16 | AATTCGTGCAGAGACAAGCGTTTGGGGCACCA | GCTCTTACTGTGCTGGTGCCCCAAACGCTTGTCT | L3 | Spacer |

| | | | | |
|---|---|---|---|---|
| | GCACAGTAA | CTGCACG | | |
| S18 | AATTCTTCAAAGCTACGAGCGCTAGAGATGTG AGACCCTAA | GCTCTTAGGGTCTCACATCTCTAGCGCTCGTAGC TTTGAAG | L3 | Spacer |
| S19 | AATTCCTAATTATGTCTCAAAAGCTCGAAGAT TACACCTAA | GCTCTTAGGTGTAATCTTCGAGCTTTTGAGACAT AATTAGG | L3 | Spacer |
| S20 | AATTCTTGTCGCTAAAGAAACCTTTCCCAATT AATACATAA | GCTCTTATGTATTAATTGGGAAAGGTTTCTTTAG CGACAAG | L3 | Spacer |
| S21 | AATTCGGAATCGCTGATCTACAGAACGGTCCT TATGGGTAA | GCTCTTACCCATAAGGACCGTTCTGTAGATCAGC GATTCCG | L3 | Spacer |
| S22 | AATTCATCACTCACACATCGCTCGAGATCGGT ACGGGGTAA | GCTCTTACCCCGTACCGATCTCGAGCGATGTGTG AGTGATG | L3 | Spacer |
| T10 | TGCCGCTGATGCCAGAAAGGGTCCTGAATTTC AGGGCCCTTTTTTTACATGGATTGA | CTAGTCAATCCATGTAAAAAAAGGGCCCTGAAAT TCAGGACCCTTTCTGGCATCAGC | L3 | Terminator |
| T16 | TGCCGTTATTAAATAGCCTGCCATCTGGCAGG CTTTTTTTATCGA | CTAGTCGATAAAAAAAGCCTGCCAGATGGCAGGC TATTTAATAAC | L3 | Terminator |
| T29 | TGCCCAGAAATCATCCTTAGCGAAAGCTAAGG ATTTTTTTTATCTGAAA | CTAGTTTCAGATAAAAAAAATCCTTAGCTTTCGC TAAGGATGATTTCTG | L3 | Terminator |
| M20 | GAGCATAGACTTTCGTGGATTATTACCTTACA ACTGATAGGACGGACTC | GGCAGAGTCCGTCCTATCAGTTGTAAGGTAATAA TCCACGAAAGTCTAT | L3 | Modifier (reference) |
| M21 | GAGCATAGCCGAGATTATCCACCAGCAACAGT TCGTTATTGTAGTGATT | GGCAAATCACTACAATAACGAACTGTTGCTGGTG GATAATCTCGGCTAT | L3 | Modifier (reference) |
| M22 | GAGCAAGGCGTGACTACAACCAATCTTCTATT CTGCGAGAGTAAAGTTT | GGCAAAACTTTACTCTCGCAGAATAGAAGATTGG TTGTAGTCACGCCTT | L3 | Modifier (reference) |

Table A.4: **List of oligonucleotide sequences used to make genetic parts**

## A.3 Analysis of possible predictors of termination efficiency

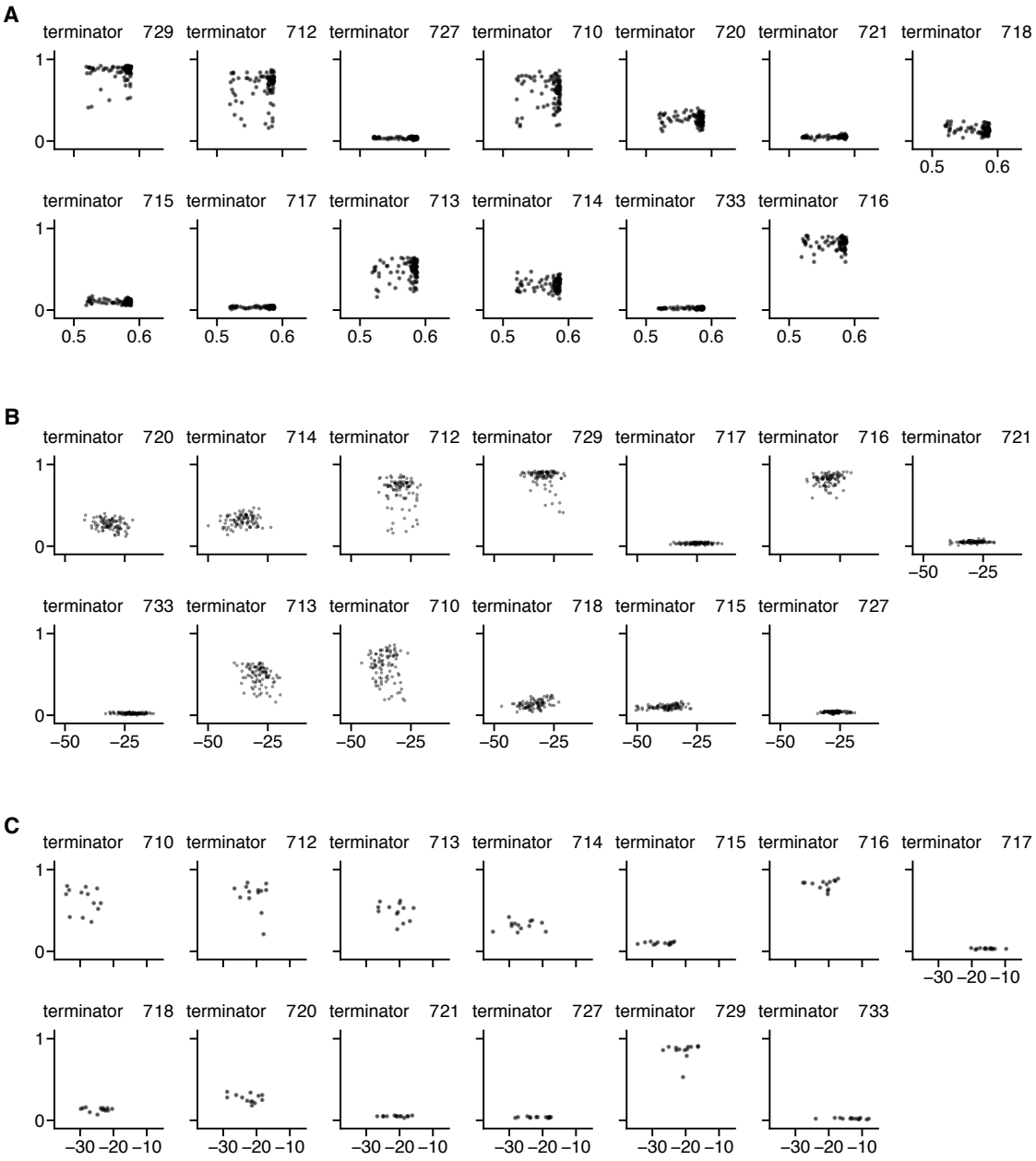



Figure A.1: **Analysis of possible predictors of termination efficiency.** (**A**) Scatter plot for each terminator showing $T_e$ against percentage GC content of each design. Calculation based on 80 nt upstream of 3'-end of design. (b) Scatter plot for each terminator showing $T_e$ against the thermodynamic minimum free energy of each design. Calculation using default settings, based on 120 nt upstream of 3'-end of the design. (c) Scatter plot for each valve showing $T_e$ against the thermodynamic minimum free energy of each valve sequence. All folding energies calculated using RNAfold [104].

## A.4 Insertion site query sequences

| **Reference** | **Sequence** |
|---|---|
| IS150 query 1 | AGCCTGACAAGTATGCGGACGTTAAAAAGCGTATTAGTGAGATTTATCACGAGAATAGAGG CCGATACGGATACCGTAGGGTAACGCTGTCTCTTCATCG |
| IS186 query 1 | TGATTATATCGTCCGGGTTCACTGGCGAGGATTGCGCTGGTTAACTGCAGAAGGAATGCGC TTTGACATGATGGGTTTTCTGCGCGGGCTGGATTGCGGT |
| IS1A query 1 | AACAGCCAGCGCTGGCGCGATTTAGCCCCGACATAGCCCCACTGTTCGTCCATTTCCGCGC AGACGATGACGTCACTGCCCGGCTGTATGCGCGAGGTTA |
| IS2 query 1 | CTCAAGCAACAGCGCATTCTGGCGCATGATCCGGTAAACACGTTTGGCATTGATCGCAGGC ATACCATCAAGTTCTGCCTGTCTGCGAAGCAGCGCCCAT |
| IS30 query 1 | GAGACAATTTATAAAACGCTGTACTTTCGTAGCCGTGAAGCGCTACACCACCTGAATATAC AGCATCTGCGACGGTCGCATAGCCTTCGCCATGGCAGGC |
| IS3 query 1 | TGCGTGCTCAGGGTTACCCCTTTAACGTAAAAACCGTGGCGGCAAGCCTGCGCCGTCAGGG ACTGAGGGCAAAGGCCTCCCGGAAGTTCAGCCCGGTCAG |
| IS4 query 1 | TGAGCAACTTATAGAACAAACCGGCGATAACACTCTGACGTTAATGGATAAAGGTTATTAC TCACTGGGACTGTTAAATGCCTGGAGCCTGGCGGGAGAA |
| random query 1 | GCGAAGATTATCGTGTGTGCCCCGTTATGGTCGAGTTCGGTCAGAGCGTCATTGCGAGTAG TCGTTTGCTTTCTCGAATTCCGAGCGATTAAGCGTGACA |
| IS150 query 2 | GAGGTAGGGCAAACCGCCCCTAATGTTCTCCAAAGAGATTTCAAGGCTACGCGGCCAAACG AGAAGTGGGTTACCGATGTTACTGAATTTGCAGTCAATG |
| IS186 query 2 | AAGCATTAATCAGTAAAACCCGACTGCTCAGCGAGAATCGTCGAAAAGGACGAGTAGTTCA GGCGGAAACGCTGGAAGCAGCGGGCCATGTGCTATTGCT |
| IS1A query 2 | TCAATGATTTTCTGGTGCGTACCGGGTTGAGAAGCGGTGTAAGTGAACTGCAGTTGCCATG TTTTACGGCAGTGAGAGCAGAGATAGCGCTGATGTCCGG |
| IS2 query 2 | AGTCATCGGTTCGTCTGAGAATGACGTGCAACTGCGCACGCGACACCCGGAGACAACGGCT GACTAAGCTTACTCCCCATCCCCGGGCAATAAGGGCGCG |
| IS30 query 2 | TTGGGAGGGCGATTTAGTCTCAGGTACAAAAAACTCTCATATAGCCACACTTGTAGACCGA AAATCACGTTATACGATCATCCTTAGACTCAGGGGCAAA |
| IS3 query 2 | CGTACAGATGAAGGCTGGCTGTATCTGGCAGTGGTCATTGACCTGTGGTCACGTGCCGTTA TTGGCTGGTCAATGTCGCCACGCATGACGGCGCAACTGG |
| IS4 query 2 | AGGCACGAAAAAAGTGGCCGGGACTGGGAAATGAAGTGACTGCCCGCCTGCTGACCGTGAC GCGCAAAGGAAAAGTCTGCCATCTGCTGACGTCGATGAC |
| random query 2 | TGCCTATAGGTAAGAAGGTGTTAGTTAGCTGACCGCTCGCTACCGGTGTATGTGGCCCTGA GGCGTGCTGCTTGAAGAAGTGTGGATCGGAGTAAGATAT |

Table A.5: **Table of insertion site query sequences.**

## A.5 Computational code

```
#!/bin/bash

############################################################
# BASECALLING
############################################################
guppy_basecaller -i "FAST5_INPUT" -s "FASTQ_OUTPUT" --config rna_r9.4.1_70bps_hac
# for DNA-seq, use --config dna_r9.4.1_450bps_fast.cfg

############################################################
# SEQUENCING DATA PRE-PROCESSING
############################################################
# COLLATE SEQUENCING DATA
cat "FASTQ_OUTPUT"/*.fastq > all.fastq
# CONVERT FASTQ TO FASTA FILE
cat all.fastq | paste - - - - | sed 's/^@/>/g'| cut -f1-2 | tr '\t' '\n' > all.fasta

############################################################
# MAKE BLASTN DATABASE OF SEQUENCING READS
############################################################
makeblastdb -in all.fasta -parse_seqids -dbtype nucl

############################################################
# PARALLELISED BLASTN ALIGNMENT OF BARCODES TO READS DATABASE
############################################################
cd "DESIGN_FASTA_FILES"
ls *.fasta | parallel blastn -db all.fasta -query {} -out "BLASTN_OUTPUT"/{}.out \
-outfmt 6 -gapopen 5 -gapextend 2 -reward 2 -penalty -3 -evalue 1 -word_size 4 \
-max_target_seqs 1000000 -max_hsps 1

############################################################
# BLASTN OUTPUT PROCESSING
############################################################
# COLLATE BLASTN ALIGNMENTS
cat "BLASTN_OUTPUT"/*.out > all_alignments.out
# FILTER LARGE DATASETS TO LONGEST ALIGNMENTS
awk '$4 >= 80' all_alignments.out > long_alignments.out

############################################################
# DEMULTIPLEX SEQUENCING READS
############################################################
demultiplex_seq_reads.py
cd "PARSED_READS_LISTS"
ls *.reads | parallel "seqtk subseq all.fastq {} > {}.fastq"

############################################################
# CREATE SEQUENCE ALIGNMENT (SAM) FILE
```

```
############################################################
ls *.fastq | parallel "minimap2 -ax map-ont -L {}.fasta {}.fastq > {}.sam"

############################################################
# CREATE AND PLOT READ AND DELTA PROFILES
############################################################
create_profiles.py
plot_profiles.py
plot_delta_profiles.py

############################################################
# CALCULATE TERMINATION EFFICIENCY
############################################################
ls *.d | parallel "calculate_TE.py {}"

############################################################
```

Figure A.2: **Bioinformatic pipeline**

```
#!/usr/bin/env python3.5

import sys, csv
import numpy as np
import pandas as pd

################################################################################
# CODE
################################################################################

alignment_file = "long_alignments.out"
fastq_file = "all.fastq"
number_alignments = # SPECIFY

# PRE-ALLOCATE AND LOAD DATA INTO ARRAY
m = np.zeros(number_alignments, dtype=[('qseqid', "S11"),('sseqid', "S36"), \
    ('bitscore', "f4"),('aln_len',"i4")])
f = csv.reader(open(alignment_file, 'r'),delimiter="\t")
for i, row in enumerate(f):
    m[i] = str(row[0]), str(row[1]),float(row[-1]),int(row[3])
df_filt = pd.DataFrame(m)
df_filt["qseqid"] = df_filt["qseqid"].str.decode('utf-8')
df_filt["sseqid"] = df_filt["sseqid"].str.decode('utf-8')

# FILTER TO BEST ALIGNMENTS
df_filt["bitscore_max"] = df_filt.groupby(["sseqid"])["bitscore"].transform(max)
best_aln = df_filt[df_filt.bitscore_max == df_filt.bitscore]

# REMOVE READS WITH MULTIPLE BEST ALIGNMENTS
best_aln = best_aln.drop_duplicates(subset="sseqid",keep=False)

# SAVE BEST ALIGNMENTS TO FILE AND DELETE DATA
best_aln.to_csv("best_seq_read_aln.csv")

# SAVE SEQUENCING READ IDS FOR EACH DESIGN
db_ids = pd.read_csv("%s/best_seq_read_aln.csv" % out_folder,\
    usecols=["qseqid","sseqid"]).reset_index()
for k, v in db_ids.groupby('qseqid'):
    # KEEP ONLY DATA FOR DESIGNS WITH SUFFICIENT READS FOR CHARACTERISATION
    if len(v) > 25:
        v["sseqid"].to_csv('PARSED_READS_LISTS/%s.reads' % k,index=False)

del m,f,df_filt,best_aln,db_ids

################################################################################
```

Figure A.3: **Python script for demultiplexing sequencing reads**

```
#!/usr/bin/env python3.5

import pysam
import glob
import re
import csv
import numpy as np


################################################################################
# WHAT TO PLOT - UPDATE THESE
################################################################################
data_path = './_analysis'
fasta_path = './_plasmids'
gff_path = './_gff'
profile_path = './_profiles'

################################################################################
# Functions to load SAM files and generate profiles
################################################################################
def load_seq (filename):
        f = open(filename, 'r')
        lines = f.readlines()
        return lines[1].strip()
def load_gff (filename):
        gff = {}
        data_reader = csv.reader(open(filename, 'r'), delimiter='\t')
        # Process each line
        for row in data_reader:
                if len(row) == 9:
                        chromo = row[0]
                        part_type = row[2]
                        start_bp = int(row[3])
                        end_bp = int(row[4])
                        part_dir = row[6]
                        part_attribs = {}
                        split_attribs = row[8].split(';')
                        part_name = None
                        for attrib in split_attribs:
                                key_value = attrib.split('=')
                                if len(key_value) == 2:
                                        if key_value[0] == 'Name':
                                                part_name = key_value[1]
                                        else:
```

```
                                        part_attribs[key_value[0]] = key_value[1]
                        if part_name != None:
                                if chromo not in gff.keys():
                                        gff[chromo] = {}
                                gff[chromo][part_name] = [part_type, part_dir, start_bp,
                                ↪  end_bp, part_attribs]
        return gff

def create_profile (sam_filename, ignore_ranges=None):
        """ Manually generate a read profile from an alignment (SAM) file.
        """
        samfile = pysam.AlignmentFile(sam_filename)
        profile = np.zeros(samfile.lengths[0])
        for read in samfile.fetch():
             if (read.reference_start is not None
                      and read.reference_end is not None):
                     # Check to see if end point is in an ignore range
                     if ignore_ranges is not None:
                             found = False
                             for idx in range(ignore_ranges.shape[0]):
                                     start_idx = ignore_ranges[idx, 0]
                                     end_idx = ignore_ranges[idx, 1]
                                     if (read.reference_end >= start_idx
                                             and read.reference_end <= end_idx):
                                             found = True
                                             break
                             if found == False:
                                     profile[read.reference_start:read.reference_end] += 1
                     else:
                             profile[read.reference_start:read.reference_end] += 1
        samfile.close()
        return profile
def write_profile (filename, profile):
        """ Write a read depth profile in the standard format
        """
        f_out = open(filename, 'w')
        for idx in range(profile.shape[0]):
                f_out.write(design+'\t'+str(idx)+'\t'+str(int(profile[idx]))+'\n')
        f_out.close()

###############################################################################
# Analysis
###############################################################################
# Generate a list of all SAM files in the data directory
samfiles = glob.glob(data_path+'/*.sam')
# For each SAM file extract the design name and then generate a profile
for f in samfiles:
```

```
        design = (f.split('.')[-2]).strip().split('/')[-1]
        design_bits = design.split('-')
        C = design_bits[0]
        I = design_bits[1]
        T = design_bits[2]
        expand_by=20
        print('Processing:', design, '('+f+')')
        design_seq = load_seq(fasta_path+'/'+design+'.fasta')
        design_gff = load_gff(gff_path+'/'+design+'.gff')
        excluded_sites=np.array([[design_gff[design][C][2],design_gff[design][I][3]+expand_by ⌋
        ↪ ]])
        profile = create_profile(f, ignore_ranges=excluded_sites)
        write_profile (profile_path+'/'+design+'.d', profile)

################################################################################
```

Figure A.4: **Python script for creating sequencing read profiles**

```
#!/usr/bin/env python3.5

import csv,matplotlib,statistics,glob
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
from operator import itemgetter

################################################################################
# WHAT TO PLOT - UPDATE THESE
################################################################################

profile_prefix = './_profiles/'
gff_prefix = './_gff/'
output_prefix = './_profile_plots/'

################################################################################
# PLOT SETTINGS
################################################################################

matplotlib.rcParams['lines.dash_joinstyle']  = 'miter'
matplotlib.rcParams['lines.dash_capstyle']   = 'butt'
matplotlib.rcParams['lines.solid_joinstyle'] = 'miter'
matplotlib.rcParams['lines.solid_capstyle']  = 'projecting'
matplotlib.rcParams['axes.labelsize']  = 8
matplotlib.rcParams['ytick.labelsize']  = 8
matplotlib.rcParams['xtick.labelsize']  = 8
matplotlib.rcParams['ytick.major.pad']  = 0.8
matplotlib.rcParams['ytick.minor.pad']  = 0.8
matplotlib.rcParams['xtick.major.pad']  = 1.5
matplotlib.rcParams['xtick.minor.pad']  = 1.5
matplotlib.rcParams['ytick.direction']  = 'out'
matplotlib.rcParams['xtick.direction']  = 'out'
matplotlib.rcParams['pdf.fonttype']          = 42
pro_light_col,pro_med_col,pro_dark_col = (0.75, 0.75, 0.75),(0.5, 0.5, 0.5),(0.25, 0.25, 0.25)

################################################################################
# SUPPORTING FUNCTIONS
################################################################################

def load_nt_data (filename):
    # This assumes the data is from nt 1 to n (sorted)
    data = []
    data_reader = csv.reader(open(filename, 'r'), delimiter='\t')
    #start data with positions with zero coverage at 5' end
    first_pos = next(data_reader)
    if float(first_pos[1]) > 0:
```

```
        for x in range(0,int(first_pos[1])-1):
            data.append(float(0))
    data.append(float(first_pos[2]))
    # Process each line
    for row in data_reader:
        if len(row) >= 3:
            data.append(float(row[2]))
    return np.array(data)

def load_part_list_from_gff (filename, chrom, region=None):
    # Load the GFF data
    gff = []
    gff_dict = {}
    data_reader = csv.reader(open(filename, 'r'), delimiter='\t')
    for row in data_reader:
        if len(row) == 9:
            cur_chrom = row[0]
            part_type = row[2]
            start_bp = int(row[3])
            end_bp = int(row[4])
            part_dir = row[6]
            part_attribs = {}
            split_attribs = row[8].split(';')
            part_name = None
            for attrib in split_attribs:
                key_value = attrib.split('=')
                if len(key_value) == 2:
                    if key_value[0] == 'Name':
                        part_name = key_value[1]
            if part_name != None and cur_chrom == chrom:
                gff.append([part_name, part_type, part_dir, start_bp, end_bp, part_attribs])
                gff_dict[part_name] = [part_type, part_dir, start_bp, end_bp, part_attribs]
    # Convert to part list for parasbolv
    part_list = []
    for gff_el in sorted(gff, key=itemgetter(3)):
        part_list.append([gff_el[1], None, None])
    return part_list, gff_dict

def plot_profiles (d_name, profile_prefix, gff_prefix, output_prefix):

    d_name_bits = d_name.split('-')
    d_con = d_name_bits[0]
    d_ins = d_name_bits[1]
    d_ter = d_name_bits[2]
    y_data = load_nt_data(profile_prefix+d_name+'.d')
    #convert y_data to percentage of RNAP that pass a point on x axis
    max_depth = int(max(y_data))
```

```
y_data = (y_data/max_depth)*100
x_data = np.arange(0, np.size(y_data))
part_list, gff_dict = load_part_list_from_gff(gff_prefix+d_name+'.gff', d_name)
# Regions to annotate on the graph
pT7_region = [gff_dict['pT7'][2], gff_dict['pT7'][3]]
gfp_region = [gff_dict['GFP'][2], gff_dict['GFP'][3]]
con_region = [gff_dict[d_con][2], gff_dict[d_con][3]+6]
valve_region = [gff_dict[d_ins][2], gff_dict[d_ter][3]+6]
ins_region = [gff_dict[d_ins][2], gff_dict[d_ins][3]+6] # include the scar
ter_region = [gff_dict[d_ter][2], gff_dict[d_ter][3]+7]
rfp_region = [gff_dict['RFP'][2], gff_dict['RFP'][3]]
rfp_ter_region1 = [gff_dict['rrnB_T1'][2], gff_dict['rrnB_T1'][3]]
rfp_ter_region2 = [gff_dict['rrnB_T1'][2], gff_dict['T7Te'][3]]

##############################################################################

# Lines around the graph
matplotlib.rcParams['axes.spines.left']   = True
matplotlib.rcParams['axes.spines.bottom'] = True
matplotlib.rcParams['axes.spines.top']    = False
matplotlib.rcParams['axes.spines.right']  = False

# Create the figure
fig = plt.figure(figsize=(3.8, 0.8))
gs = gridspec.GridSpec(1, 1)

# --------- Read depth track ---------
annotate_lw = 0.4
ax = plt.subplot(gs[0])
# Coding regions
ax.fill_between(x_data[0:valve_region[0]], 0, y_data[0:valve_region[0]], linewidth=0.0,
↪  color=pro_light_col, zorder=-10)
ax.fill_between(x_data[valve_region[1]:rfp_ter_region1[1]], 0,
↪  y_data[valve_region[1]:rfp_ter_region1[1]], linewidth=0.0, color=pro_light_col,
↪  zorder=-10)
ax.axvline(con_region[0], linewidth=annotate_lw, linestyle='--', color=(0,0,0))

# Valve
ax.fill_between(range(ins_region[0], ins_region[1]), 0,
↪  y_data[ins_region[0]:ins_region[1]], linewidth=0.0, color=pro_med_col, zorder=-5)
ax.fill_between(range(ter_region[0], ter_region[1]), 0,
↪  y_data[ter_region[0]:ter_region[1]], linewidth=0.0, color=pro_dark_col, zorder=-5)
ax.axvline(valve_region[0], linewidth=annotate_lw, linestyle='-', color=(0,0,0))
ax.axvline(valve_region[1], linewidth=annotate_lw, linestyle='-', color=(0,0,0))

#RFP terminators
ax.axvline(rfp_ter_region1[0], linewidth=annotate_lw, linestyle='-', color=(0,0,0))
```

```
ax.axvline(rfp_ter_region1[1], linewidth=annotate_lw, linestyle='-', color=(0,0,0))
ax.axvline(rfp_ter_region2[0], linewidth=annotate_lw, linestyle='--', color=(0,0,0))
ax.axvline(rfp_ter_region2[1], linewidth=annotate_lw, linestyle='--', color=(0,0,0))

# Set what is shown
ax.set_ylim([0, np.max(y_data)*1.1])
ax.set_xlim([0,2000]) # rfp_ter_region[1]]
ax.set_xticklabels([])
ax.set_xticks([gfp_region[0], valve_region[0], rfp_region[0], rfp_region[1]])
ax.axvline(pT7_region[1], linewidth=annotate_lw, linestyle='dotted', color=(0,0,0))

# calculate TE
plasmid_len=len(y_data)
pre_design = 778
design_length = plasmid_len-3673
post_design = pre_design + design_length
pre_depth = int(y_data[["pos"]==pre_design])
post_depth = int(y_data[["pos"]==post_design])
TE = (pre_depth - post_depth)/pre_depth

# Save the plot
plt.subplots_adjust(hspace=.3, wspace=.05, left=.08, right=.99, top=.97, bottom=.1)
plt.text(20, -10, "%s \n%s reads \nTE: %s %% \n\n" % (d_name,max_depth,TE))
↪  #,TE_RFP1,TE_RFP2,drop1,drop2)) # \nTE RFP1: %s %%, TE_RFP2: %s %% \nDrop1: %s %%,
↪  Drop2: %s %% \n
fig.savefig(output_prefix+d_name+'_full_profile.png', transparent=True)
plt.close('all')

##############################################################################

matplotlib.rcParams['axes.spines.left']   = True
matplotlib.rcParams['axes.spines.bottom'] = True
matplotlib.rcParams['axes.spines.top']    = False
matplotlib.rcParams['axes.spines.right']  = False
# Create the figure
fig = plt.figure(figsize=(3.8, 0.8))
gs = gridspec.GridSpec(1, 1)
# --------- Read depth track ---------
annotate_lw = 0.4
ax = plt.subplot(gs[0])
# Coding regions
ax.fill_between(x_data[0:con_region[1]], 0, y_data[0:con_region[1]], linewidth=0.0,
↪  color=pro_light_col, zorder=-10)
ax.fill_between(x_data[valve_region[1]:rfp_ter_region1[1]], 0,
↪  y_data[valve_region[1]:rfp_ter_region1[1]], linewidth=0.0, color=pro_light_col,
↪  zorder=-10)
ax.axvline(con_region[0], linewidth=annotate_lw, linestyle='--', color=(0,0,0))
```

```
    # Valve
    ax.fill_between(range(ins_region[0], ins_region[1]), 0,
    ↪  y_data[ins_region[0]:ins_region[1]], linewidth=0.0, color=pro_med_col, zorder=-5)
    ax.fill_between(range(ter_region[0], ter_region[1]), 0,
    ↪  y_data[ter_region[0]:ter_region[1]], linewidth=0.0, color=pro_dark_col, zorder=-5)
    ax.axvline(valve_region[0], linewidth=annotate_lw, linestyle='-', color=(0,0,0))
    ax.axvline(valve_region[1], linewidth=annotate_lw, linestyle='-', color=(0,0,0))
    ax.set_xticklabels([])
    ax.set_xticks([valve_region[0], rfp_region[0]])
    # Set what is shown
    ax.set_ylim([0, np.max(y_data)*1.1])
    ax.set_xlim([gfp_region[1]-50, rfp_region[0]+25])
    # Save the plot
    plt.subplots_adjust(hspace=.3, wspace=.05, left=.08, right=.99, top=.97, bottom=.1)
    plt.text(0.1, -0.2,"%s \n%s reads \nTE: %s %% \n\n" %
    ↪  (d_name,max_depth,TE),transform=fig.transFigure)
    fig.savefig(output_prefix+d_name+'_zoom_profile.png', transparent=True)
    plt.close('all')

################################################################################
# ANALYSIS
################################################################################

d_files = glob.glob(profile_prefix+'*.d', recursive=False)
for f in d_files:
    filename = f.split('/')[-1]
    d_name = filename.split('.')[0]
    print('Processing: '+d_name)
    plot_profiles (d_name, profile_prefix, gff_prefix, output_prefix)

################################################################################
```

Figure A.5: **Python script for plotting sequencing read profiles**

```
#!/usr/bin/env python3.5

import csv
import numpy as np
import statistics
import matplotlib
import matplotlib.pyplot as plt
import matplotlib.gridspec as gridspec
from operator import itemgetter
import glob

################################################################################
# WHAT TO PLOT - UPDATE THESE
################################################################################

profile_prefix = './_profile/'
gff_prefix = './_gff/'
output_prefix = './_profile_plots/'

################################################################################
# PLOT SETTINGS
################################################################################

matplotlib.rcParams['lines.dash_joinstyle']  = 'miter'
matplotlib.rcParams['lines.dash_capstyle']   = 'butt'
matplotlib.rcParams['lines.solid_joinstyle'] = 'miter'
matplotlib.rcParams['lines.solid_capstyle']  = 'projecting'
# Axes titles
matplotlib.rcParams['axes.labelsize']  = 8
# Numbers on each axis
matplotlib.rcParams['ytick.labelsize']  = 8
matplotlib.rcParams['xtick.labelsize']  = 8
# Space between axis and number
matplotlib.rcParams['ytick.major.pad']  = 0.8
matplotlib.rcParams['ytick.minor.pad']  = 0.8
matplotlib.rcParams['xtick.major.pad']  = 1.5
matplotlib.rcParams['xtick.minor.pad']  = 1.5
matplotlib.rcParams['ytick.direction']  = 'out'
matplotlib.rcParams['xtick.direction']  = 'out'
# Make text editable in Adobe Illustrator
matplotlib.rcParams['pdf.fonttype']          = 42

pro_light_col = (0.75, 0.75, 0.75)
pro_med_col = (0.5, 0.5, 0.5)
pro_dark_col = (0.25, 0.25, 0.25)

################################################################################
```

```
# SUPPORTING FUNCTIONS
################################################################################

def load_nt_delta_data (filename):
    # This assumes the data is from nt 1 to n (sorted)
    data = []
    delta_data = []
    data_reader = csv.reader(open(filename, 'r'), delimiter='\t')
    #start data with positions with zero coverage at 5' end
    first_pos = next(data_reader)
    if float(first_pos[1]) > 0:
        for x in range(0,int(first_pos[1])-1):
            data.append(float(0))
            delta_data.append(float(0))
    data.append(float(first_pos[2]))
    delta_data.append(float(first_pos[2]))
    prev_row = float(first_pos[2])

    # Process each line
    for row in data_reader:
        if len(row) >= 3:
            data.append(float(row[2]))
            delta_data.append(float(row[2]) - prev_row)
            prev_row = float(row[2])
    return np.array(delta_data)

def load_nt_data (filename):
    # This assumes the data is from nt 1 to n (sorted)
    data = []
    data_reader = csv.reader(open(filename, 'r'), delimiter='\t')
    #start data with positions with zero coverage at 5' end
    first_pos = next(data_reader)
    if float(first_pos[1]) > 0:
        for x in range(0,int(first_pos[1])-1):
            data.append(float(0))
    data.append(float(first_pos[2]))
    # Process each line
    for row in data_reader:
        if len(row) >= 3:
            data.append(float(row[2]))
    return np.array(data)


def load_part_list_from_gff (filename, chrom, region=None):
    # Load the GFF data
    gff = []
    gff_dict = {}
```

```
    data_reader = csv.reader(open(filename, 'r'), delimiter='\t')
    for row in data_reader:
        if len(row) == 9:
            cur_chrom = row[0]
            part_type = row[2]
            start_bp = int(row[3])
            end_bp = int(row[4])
            part_dir = row[6]
            part_attribs = {}
            split_attribs = row[8].split(';')
            part_name = None
            for attrib in split_attribs:
                key_value = attrib.split('=')
                if len(key_value) == 2:
                    if key_value[0] == 'Name':
                        part_name = key_value[1]
                    # TODO: add else to process attributes
            if part_name != None and cur_chrom == chrom:
                gff.append([part_name, part_type, part_dir, start_bp, end_bp, part_attribs])
                gff_dict[part_name] = [part_type, part_dir, start_bp, end_bp, part_attribs]
    # Convert to part list for parasbolv
    part_list = []
    for gff_el in sorted(gff, key=itemgetter(3)):
        part_list.append([gff_el[1], None, None])
    return part_list, gff_dict

def plot_profiles (d_name, profile_prefix, gff_prefix, output_prefix):
    d_name_bits = d_name.split('-')
    d_con = d_name_bits[0]
    d_ins = d_name_bits[1]
    d_ter = d_name_bits[2]
    y_data = load_nt_delta_data(profile_prefix+d_name+'.d')
    #convert y_data to percentage of RNAP that pass a point on x axis
    full_data = load_nt_data(profile_prefix+d_name+'.d')
    max_depth = int(max())
    y_data = (y_data/max_depth)*100
    x_data = np.arange(0, np.size(y_data))
    part_list, gff_dict = load_part_list_from_gff(gff_prefix+d_name+'.gff', d_name)
    # Regions to annotate on the graph
    pT7_region = [gff_dict['pT7'][2], gff_dict['pT7'][3]]
    gfp_region = [gff_dict['GFP'][2], gff_dict['GFP'][3]]
    con_region = [gff_dict[d_con][2], gff_dict[d_con][3]+6]
    valve_region = [gff_dict[d_ins][2], gff_dict[d_ter][3]+6]
    ins_region = [gff_dict[d_ins][2], gff_dict[d_ins][3]+6] # include the scar
    ter_region = [gff_dict[d_ter][2], gff_dict[d_ter][3]+7]
    rfp_region = [gff_dict['RFP'][2], gff_dict['RFP'][3]]
    rfp_ter_region1 = [gff_dict['rrnB_T1'][2], gff_dict['rrnB_T1'][3]]
```

```
    rfp_ter_region2 = [gff_dict['rrnB_T1'][2], gff_dict['T7Te'][3]]

    ##############################################################################

    matplotlib.rcParams['axes.spines.left']   = True
    matplotlib.rcParams['axes.spines.bottom'] = True
    matplotlib.rcParams['axes.spines.top']    = False
    matplotlib.rcParams['axes.spines.right']  = False

    # Create the figure
    fig = plt.figure(figsize=(3.8, 0.8))
    gs = gridspec.GridSpec(2, 1, height_ratios=[3,1])
    # --------- Read depth track ---------
    annotate_lw = 0.4
    ax = plt.subplot(gs[0])
    # Coding regions
    ax.fill_between(x_data[0:con_region[1]], 0, y_data[0:con_region[1]], linewidth=0.0,
    ↪  color=pro_light_col, zorder=-10)
    ax.fill_between(x_data[valve_region[1]:rfp_ter_region1[1]], 0,
    ↪  y_data[valve_region[1]:rfp_ter_region1[1]], linewidth=0.0, color=pro_light_col,
    ↪  zorder=-10)
    ax.axvline(con_region[0], linewidth=annotate_lw, linestyle='--', color=(0,0,0))
    # Valve
    ax.fill_between(range(ins_region[0], ins_region[1]), 0,
    ↪  y_data[ins_region[0]:ins_region[1]], linewidth=0.0, color=pro_med_col, zorder=-5)
    ax.fill_between(range(ter_region[0], ter_region[1]), 0,
    ↪  y_data[ter_region[0]:ter_region[1]], linewidth=0.0, color=pro_dark_col, zorder=-5)
    ax.axvline(valve_region[0], linewidth=annotate_lw, linestyle='-', color=(0,0,0))
    ax.axvline(valve_region[1], linewidth=annotate_lw, linestyle='-', color=(0,0,0))
    ax.set_xticklabels([])
    ax.set_xticks([valve_region[0], rfp_region[0]])
    #ax.set_yticklabels([])
    #ax.set_yticks([])
    # Set what is shown
    ax.set_ylim([-30, 5])
    ax.set_xlim([ter_region[0], ter_region[1]])
    # Save the plot
    plt.subplots_adjust(hspace=.3, wspace=.05, left=.08, right=.99, top=.97, bottom=.1)
    fig.savefig(output_prefix+d_name+'_zoom_profile.png', transparent=True)
    plt.close('all')

##############################################################################
# ANALYSIS
##############################################################################

d_files = glob.glob(profile_prefix+'*.d', recursive=False)
for f in d_files:
```

```
    filename = f.split('/')[-1]
    d_name = filename.split('.')[0]
    print('Processing: '+d_name)
    plot_profiles (d_name, profile_prefix, gff_prefix, output_prefix)

################################################################################
```

Figure A.6: **Python script for plotting delta profiles**

```
#!/usr/bin/env python3.5

import pandas as pd
import sys

##############################################################################
# CODE
##############################################################################

# IMPORT READ PROFILE
df = pd.read_csv("%s" % (sys.argv[1]),names=["ref","pos","depth"],sep="\t")
ref = sys.argv[1].split("/")[-1].split(".")[0]

# SPECIFY LOCATIONS, FIND DEPTH AND CALCULATE TE
plasmid_len=(df["pos"].iloc[-1])
pre_design = 778
design_length = plasmid_len-3673
post_design = pre_design + design_length
pre_depth = int(df[df["pos"]==pre_design]["depth"])
post_depth = int(df[df["pos"]==post_design]["depth"])
TE = (pre_depth - post_depth)/pre_depth

# SAVE TE
with open("TE.csv","a+") as f:
        if TE:
                f.write("%s,%s,%s,%s\n" % (ref,round(TE,2),pre_valve,post_valve))
        else:
                f.write("%s,0.0,%s,%s\n" % (ref,pre_valve,post_valve))
del df

##############################################################################
```

Figure A.7: **Python script for calculating termination efficiency**

```
import matplotlib
#matplotlib.use('TkAgg')
import matplotlib.pyplot as plt
import matplotlib.patches as patches
import matplotlib.gridspec as gridspec
import numpy as np
import random
import pandas as pd
import math
from statistics import mean
import sys
################################################################
# SETTINGS/PARAMETERS FOR HOW THE GRAPH LOOKS
################################################################

# Axes titles
matplotlib.rcParams['axes.labelsize']  = 8
# Numbers on each axis
matplotlib.rcParams['ytick.labelsize']  = 8
matplotlib.rcParams['xtick.labelsize']  = 8
# Space between axis and number
matplotlib.rcParams['ytick.major.pad']  = 0.8
matplotlib.rcParams['ytick.minor.pad']  = 0.8
matplotlib.rcParams['xtick.major.pad']  = 1.5
matplotlib.rcParams['xtick.minor.pad']  = 1.5
matplotlib.rcParams['ytick.direction']  = 'out'
matplotlib.rcParams['xtick.direction']  = 'out'
# Lines around the graph
matplotlib.rcParams['axes.spines.left']   = True
matplotlib.rcParams['axes.spines.bottom'] = True
matplotlib.rcParams['axes.spines.top']    = False
matplotlib.rcParams['axes.spines.right']  = False
# Make text editable in Adobe Illustrator
matplotlib.rcParams['pdf.fonttype']          = 42

# Colour maps to use for the genetic diagrams
# https://personal.sron.nl/~pault/
cmap = {}
cmap['vl_purple'] = (214/255.0, 193/255.0, 222/255.0)
cmap['l_purple']  = (177/255.0, 120/255.0, 166/255.0)
cmap['purple']    = (136/255.0,  46/255.0, 114/255.0)
cmap['blue']      = ( 25/255.0, 101/255.0, 176/255.0)
cmap['l_blue']    = ( 82/255.0, 137/255.0, 199/255.0)
cmap['vl_blue']   = (123/255.0, 175/255.0, 222/255.0)
cmap['green']     = ( 78/255.0, 178/255.0, 101/255.0)
cmap['l_green']   = (144/255.0, 201/255.0, 135/255.0)
cmap['vl_green']  = (202/255.0, 224/255.0, 171/255.0)
```

```
cmap['yellow']    = (247/255.0, 238/255.0,  85/255.0)
cmap['vl_orange'] = (246/255.0, 193/255.0,  65/255.0)
cmap['l_orange']  = (241/255.0, 147/255.0,  45/255.0)
cmap['orange']    = (232/255.0,  96/255.0,  28/255.0)
cmap['red']       = (220/255.0,   5/255.0,  12/255.0)
cmap['grey']      = (130/255.0, 130/255.0, 130/255.0)
cmap['vl_grey']   = (230/255.0, 230/255.0, 230/255.0)
cmap['l_grey']    =  (200/255.0, 200/255.0, 200/255.0)
cmap['d_grey']    = ( 50/255.0,  50/255.0,  50/255.0)


############################################################
# FUNCTIONS TO PERFORM THE SIMULATION
############################################################

def fragment (reads, prob, t_pos, ref, power=0):
        frag_reads = []
        x = []
        for r in reads:
                if random.random() < prob:
                        frag_pos = random.randint(r[0],r[1])
                        if ref == "bc":
                                if frag_pos < t_pos[0] or frag_pos > t_pos[1]:
                                        frag_reads.append([frag_pos, r[1]])
                                        frag_reads.append([r[0], frag_pos])
                        elif ref == "full":
                                frag_reads.append([frag_pos, r[1]])
                                frag_reads.append([r[0], frag_pos])
                                # Else the read is lost because cut in terminator
                else:
                        frag_reads.append([r[0], r[1]])
        return frag_reads

def adaptor_ligation (reads, prob, t_pos, ref, power=0):
        frag_reads = []
        x = []
        for r in reads:
                if random.random() < prob:
                        # then adaptor is ligated and the RNA can be sequenced
                        frag_reads.append([r[0],r[1]])
                        # Else the read is lost because no adaptor is added
        return frag_reads

def fragment_2 (reads, prob, t_pos, ref, power=0):
        frag_reads = []
        x = []
        for r in reads:
```

```
                if random.random() < prob:
                        frag_pos = random.randint(r[0],r[1])
                        if ref == "bc":
                                if frag_pos < t_pos[0]:
                                        frag_reads.append([frag_pos, r[1]])
                                        # Else the read is lost because cut within design
                        elif ref == "full":
                                frag_reads.append([frag_pos, r[1]])

                else:
                        frag_reads.append([r[0], r[1]])
        return frag_reads



# Build the profile
def build_profile (iso1, iso2, l):
        profile = np.zeros(l)
        for r in iso1:
                profile[r[0]:r[1]] += 1
        for r in iso2:
                profile[r[0]:r[1]] += 1
        return profile

def simulate_seq_frag_adapt (l=1000, t_pos=[450, 500], tot_reads=100000, Te=0.5,
↪  prob_frag=0.5, prob_frag_2=0.5, prob_drop=0.5, f='', offset=0,power=0,ref="bc"):
        # Create the starting full length reads
        iso1_reads = []
        iso2_reads = []
        for idx in range(int(tot_reads*Te)):
                iso1_reads.append([0, t_pos[1]])
        for idx in range(int(tot_reads*(1-Te))):
                iso2_reads.append([0, l])

#         if ref == "bc" or ref == "full":
        # Fragment the reads and filter out those not captured
        frag_iso1_reads = fragment(iso1_reads, prob_frag, t_pos, ref)
        frag_iso2_reads = fragment(iso2_reads, prob_frag, t_pos, ref)
        # Drop some reads half way
        frag_iso1_reads = adaptor_ligation(frag_iso1_reads, prob_drop, t_pos, ref)
        frag_iso2_reads = adaptor_ligation(frag_iso2_reads, prob_drop, t_pos, ref)
        # Fragment the reads again
        frag_iso1_reads = fragment_2(frag_iso1_reads, prob_frag_2, t_pos, ref)
        frag_iso2_reads = fragment_2(frag_iso2_reads, prob_frag_2, t_pos, ref)

        # Generate the profiles
        real_profile = build_profile(iso1_reads, iso2_reads, l)
```

```
        frag_profile = build_profile(frag_iso1_reads, frag_iso2_reads, l)


        #load data profile
        rnacs=pd.read_csv(f,names=["ref","posn","depth"],sep="\t")
        rnacs["posn_corr"] = rnacs["posn"] - offset
        rnacs = rnacs[rnacs["posn_corr"] > 600]
        rnacs = rnacs[rnacs["posn_corr"] <= 1885]
        #print(f,rnacs["depth"].max())
        rnacs["depth_corr"] = rnacs["depth"] / real_profile.max()*100
        data_profile=rnacs["depth_corr"]

        # Correct profile to percentage
        frag_profile = frag_profile/real_profile.max()*100
        real_profile = real_profile/real_profile.max()*100

        frag_profile_fig2 = frag_profile/frag_profile.max()*100
        df_sim = pd.DataFrame ([[int(x) for x in
        ↪  range(len(frag_profile_fig2))],frag_profile_fig2]).transpose()
        df_sim.insert(0,"ref","C10-I20-T21")
        df_sim.to_csv("C10-I20-T21.d", header=False, index=False,sep="\t")
        #print(df_sim)
        return real_profile, frag_profile, data_profile

def plot_read_profile (filename_out, real_profile, frag_profile, data_profile, l, t_pos,
↪  prob_f, prob_d):
        # Plot the profiles
        fig = plt.figure(figsize=(3.4, 1.4))
        gs = gridspec.GridSpec(1, 1)
        ax = plt.subplot(gs[0])
        # Show the profile before fragmentation
        ax.plot(range(l), real_profile, color=(0,0,0), linestyle='-', linewidth=0.8,
        ↪  zorder=-1)
        ax.plot([0,0], [0,real_profile[0]], color=(0,0,0), linestyle='-', linewidth=0.8,
        ↪  zorder=-1)
        ax.plot([l,l], [0,real_profile[-1]], color=(0,0,0), linestyle='-', linewidth=0.8,
        ↪  zorder=-1)
        # In red show the recovered profile after fragmentation
        ax.fill_between(range(l), np.zeros_like(data_profile), data_profile,
        ↪  color=cmap['red'], alpha=0.7, linewidth=0, zorder=-5)
        ax.plot(range(l),frag_profile,linestyle='--', color="black", linewidth=0.8,
        ↪  zorder=-1) #marker=".",color="r",ax=ax)
        # Highlight the barcode
        t_len = t_pos[1] - t_pos[0]
        ax.fill_between(np.array(range(t_len))+t_pos[0], np.zeros(t_len),
        ↪  np.ones(t_len)*real_profile[0], color=cmap['grey'], linewidth=0, zorder=-10)
        # Format the axes
```

```
        ax.set_xlim([-50, l+100])
        ax.set_ylim([0, real_profile[0]*1.1])
        ax.set_yticklabels(['0','','0.5', '', '1.0'])
        # Sort out the formatting of the plot (fill entire frame)
        plt.subplots_adjust(hspace=.0 , wspace=.00, left=.12, right=.95, top=.95, bottom=.14)
        fig.savefig(filename_out, transparent=True)
        plt.close('all')

def plot_sim_profile (filename_out, real_profile, frag_profile, data_profile, l, t_pos,
↪  prob_f, prob_d,Te=0,nreads=0):
        len_spacer = 40
        # Plot the profiles
        fig = plt.figure(figsize=(3.4, 1.4))
        gs = gridspec.GridSpec(1, 1)
        ax = plt.subplot(gs[0])
        # Show the profile before fragmentation
        ax.plot(range(l), real_profile, color=(0,0,0), linestyle='-', linewidth=0.8,
        ↪  zorder=-1)
        ax.plot([0,0], [0,real_profile[0]], color=(0,0,0), linestyle='-', linewidth=0.8,
        ↪  zorder=-1)
        ax.plot([l,l], [0,real_profile[-1]], color=(0,0,0), linestyle='-', linewidth=0.8,
        ↪  zorder=-1)
        # In red show the recovered profile after fragmentation
        ax.fill_between(range(l), np.zeros_like(frag_profile), frag_profile,
        ↪  color=cmap['red'], alpha=0.7, linewidth=0, zorder=-5)
        # Highlight the terminator
        t_len = t_pos[1] - t_pos[0]
        ax.fill_between(np.array(range(t_len))+t_pos[0], np.zeros(t_len),
        ↪  np.ones(t_len)*real_profile[0], color=cmap['grey'], linewidth=0, zorder=-10)
        #calculate TE
        TE_frag = (frag_profile[t_pos[0]+len_spacer]-frag_profile[t_pos[1]])/frag_profile[t_p⌋
        ↪  os[0]+len_spacer]
        TE_real = (real_profile[t_pos[0]+len_spacer]-real_profile[t_pos[1]])/real_profile[t_p⌋
        ↪  os[0]+len_spacer]
        plt.text(1000, 40, "nreads: %.0f \nTE_frag: %.2f \nTE_real: %.2f \n" %
        ↪  (nreads,TE_frag,TE_real))
        # Format the axes
        ax.set_xlim([-50, l+100])
        ax.set_ylim([0, real_profile[0]*1.1])
        # Sort out the formatting of the plot (fill entire frame)
        plt.subplots_adjust(hspace=.0 , wspace=.00, left=.12, right=.99, top=.95, bottom=.14)
        fig.savefig(filename_out, transparent=True)
        plt.close('all')


def TE_diff (real_profile, frag_profile, data_profile, l, t_pos):
        #calculate TE
```

```
        len_spacer = 40
        TE_frag = (frag_profile[t_pos[0]+len_spacer]-frag_profile[t_pos[1]])/frag_profile[t_p
        ↪  os[0]+len_spacer]
        TE_real = (real_profile[t_pos[0]+len_spacer]-real_profile[t_pos[1]])/real_profile[t_p
        ↪  os[0]+len_spacer]
        return TE_frag

def plot_TE_diff (x,y,filename_out):
        # Plot the profiles
        fig = plt.figure(figsize=(2.5, 2.5))
        gs = gridspec.GridSpec(1, 1)
        ax = plt.subplot(gs[0])
        ax.scatter(x,y, s=1.5, color=(0,0,0))
        fit_x = np.linspace(0, 1, 50, endpoint=True)
        fit_y = fit_para(fit_x)
        ax.plot(fit_x,fit_y, linewidth=0.8, color=(0,0,0), linestyle='--')
        # Format the axes
        ax.set_xlim([0, 1.02])
        ax.set_ylim([0, 1.02])
        # Sort out the formatting of the plot (fill entire frame)
        plt.subplots_adjust(hspace=.0 , wspace=.00, left=.12, right=.95, top=.95, bottom=.12)
        fig.savefig(filename_out, transparent=True)
        plt.close('all')

################################################################
# MODELLING
################################################################

# positional information for the RNA CS barcode
t_pos=[597, 714]
l = 1285
len_spacer = 40

prob_f,prob_d,prob_f2= 0.1, 0.87, 0.45

# model where you get random fragmentation then random adaptor ligation then a second random
↪  fragmentation

#plot rep1 profiles
filename_full,filename_bc= 'profiles/RNA_CS_R1.d','profiles/RNA_CS_barcode_R1.d'
real_profile, frag_profile, data_profile = simulate_seq_frag_adapt (l=l, t_pos=t_pos,
↪  tot_reads=720, Te=0.0, prob_frag=prob_f, prob_drop=prob_d,prob_frag_2=prob_f2,
↪  f=filename_full,ref="full")
plot_read_profile ('_plots/rep1_full.pdf', real_profile, frag_profile, data_profile, l,
↪  t_pos, prob_f=prob_f, prob_d=prob_d)
```

```
real_profile, frag_profile, data_profile = simulate_seq_frag_adapt (l=l, t_pos=t_pos,
↪  tot_reads=720, Te=0.0, prob_frag=prob_f, prob_drop=prob_d, prob_frag_2=prob_f2,
↪  f=filename_bc,ref="bc")
plot_read_profile ('_plots/rep1_bc.pdf', real_profile, frag_profile, data_profile, l, t_pos,
↪  prob_f=prob_f, prob_d=prob_d)

# plot rep1 error profile
ls_TE = []
ls_frag_TE = []

#set parameters
repn='R1'
f_name='profiles/RNA_CS_barcode_%s.d' % repn
prob_f,prob_d,prob_f2= 0.1, 0.87, 0.45
gaps = 11

# simulate the read profile for each termination efficiency
for te_val in np.linspace(0.,1.,gaps,endpoint=True):
        # positional information for the design barcode
         Te,l,t_pos,nreads,p =te_val,1750,[776, 893],10000,0.0
         real_profile, frag_profile, data_profile = simulate_seq_frag_adapt (l=l,
         ↪  t_pos=t_pos, tot_reads=nreads, Te=Te, prob_frag=prob_f,prob_frag_2=prob_f2,
         ↪  prob_drop=prob_d, f=f_name,power=0)
         ls_TE.append(te_val)
         TE_d = TE_diff (real_profile, frag_profile, data_profile, l, t_pos)
         ls_frag_TE.append(TE_d)
         #TO PLOT READ PROFILE FOR EACH TE:
         plot_sim_profile ('_plots/%s_%s_profile.png' % (round(te_val,1),repn), real_profile,
         ↪  frag_profile, data_profile, l, t_pos, prob_f=prob_f, prob_d=prob_d,nreads=nreads)

# save data for correcting termination efficiencies
ls_frag_TE = [x.round(2) for x in ls_frag_TE]
ls_done = []
with open("data/error_correction_%s.csv" % repn,"w") as f:
        for x in range(len(ls_TE)):
                if ls_frag_TE[x] not in ls_done:
                        ls_done.append(ls_frag_TE[x])
                        f.write("%s,%s\n" % (ls_TE[x].round(2),ls_frag_TE[xplot_TE_diff(x=ls
                        ↪  _TE,y=ls_frag_TE,filename_out="_plots/%s_error.pdf" %
                        ↪  repn)
```

Figure A.8: **Python script for simulating and modelling dRNA-seq data**

# BIBLIOGRAPHY


[1] M. A. ALTIERI, *Agroecology: The Science of Sustainable Agriculture*, CRC Press, 2018.

[2] S. F. ALTSCHUL, W. GISH, W. MILLER, E. W. MYERS, AND D. J. LIPMAN, *Basic local alignment search tool*, Journal of Molecular Biology, 215 (1990), pp. 403–410.

[3] M. A. ALVAREZ, A. G. UZMAN, AND M. E. LIAS, *Experimental visualization of mixing pathologies in laminar stirred tank bioreactors*, Chemical Engineering Science, 60 (2005), pp. 2449–2457.

[4] S. L. AMARASINGHE, S. SU, X. DONG, L. ZAPPIA, M. E. RITCHIE, AND Q. GOUIL, *Opportunities and challenges in long-read sequencing data analysis*, Genome Biology, (2020), pp. 1–16.

[5] V. AMARELLE, A. SANCHES-MEDEIROS, R. SILVA-ROCHA, AND M.-E. GUAZZARONI, *Expanding the toolbox of broad host-range transcriptional terminators for proteobacteria through metagenomics*, ACS Synthetic Biology, 8 (2019), pp. 647–654.

[6] A. AMERUOSO, L. GAMBILL, B. LIU, M. C. V. KCAM, AND J. CHAPPELL, *Brave new 'RNA' world—advances in RNA tools and their application for understanding and engineering biological systems*, Current Opinion in Systems Biology, 14 (2019), pp. 32–40.

[7] B. I. ANDREWS, F. D. ANTIA, S. B. BRUEGGEMEIER, L. J. DIORAZIO, S. G. KOENIG, M. E. KOPACH, H. LEE, M. OLBRICH, AND A. L. WATSON, *Sustainability challenges and opportunities in oligonucleotide manufacturing*, Journal of Organic Chemistry, 86 (2021), pp. 49–61.

[8] V. D. APPANNA, *Human Microbes - The Power Within*, Springer, 2018.

[9] I. ARTSIMOVITCH, *Bacterial rna synthesis: back to the limelight*, 2021.

[10] J. G. AW, Y. SHEN, A. WILM, M. SUN, X. N. LIM, K. L. BOON, S. TAPSIN, Y. S. CHAN, C. P. TAN, A. Y. SIM, T. ZHANG, T. T. SUSANTO, Z. FU, N. NAGARAJAN, AND Y. WAN, *In Vivo Mapping of Eukaryotic RNA Interactomes Reveals Principles of Higher-Order Organization and Regulation*, Molecular Cell, 62 (2016), pp. 603–617.

[11] H. AZADI, S. A. DE JONG, B. DERUDDER, P. DE MAEYER, AND F. WITLOX, *Bitter sweet - how sustainable is bio-ethanol production in brazil?*, Renewable and Sustainable Energy Reviews, 16 (2012), pp. 3599–3603.

[12] C. BAEDEN-FULLER AND S. HAELFLIGER, *Business Models and Technological Innovation*, Long Range Planning, 46 (2013), pp. 419–426.

[13] J. BAGGINI, *How the world thinks - a global history of philosophy*, Granta Books, 2018.

[14] H. BAIG, P. FONTANARROSA, V. KULKARNI, J. MCLAUGHLIN, P. VAIDYANATHAN, B. BARTLEY, S. BHATIA, S. BHAKTA, M. BISSELL, K. CLANCY, R. S. COX, A. G. MORENO, T. GOROCHOWSKI, R. GRUNBERG, A. LUNA, C. MADSEN, G. MISIRLI, T. NGUYEN, N. L. NOVERE, Z. PALCHICK, M. POCOCK, N. ROEHNER, H. SAURO, J. SCOTT-BROWN, J. T. SEXTON, G.-B. STAN, J. J. TABOR, M. V. VILAR, C. A. VOIGT, A. WIPAT, D. ZONG, Z. ZUNDEL, J. BEAL, AND C. MYERS, *Synthetic biology open language visual (sbol visual) version 2.2*, Journal of Integrative Bioinformatics, 17 (2020), p. 20200014.

[15] G. BALDWIN, T. BAYER, R. DICKINSON, T. ELLIS, F. P. S., R. I. KITNEY, K. POLIZZI, AND G.-B. STAN, *Synthetic Biology: A Primer*, Imperial College Press, 2012.

[16] P. BANÁŠ, D. HOLLAS, M. ZGARBOVÁ, P. JUREČKA, M. OROZCO, T. E. CHEATHAM, J. ŠPONER, AND M. OTYEPKA, *Performance of Molecular Mechanics Force Fields for RNA Simulations: Stability of UUCG and GNRA Hairpins*, Journal of Chemical Theory and Computation, 6 (2010), pp. 3836–3849.

[17] F. BASTAGLI, J. HAGEN-ZANKER, L. HARMAN, V. BARCA, G. STURGE, AND T. SCHMIDT, *Cash transfers - what does the evidence say? A rigorous review of programme impact and the role of design and implementation features.*, Overseas Development Institute, 2017.

[18] C. BATTY, E. ELLISON, A. OWENS, AND D. BRIEN, *Mapping the emotional journey of the doctoral 'hero': Challenges faced and breakthroughs made by creative arts and humanities candidates*, Arts and Humanities in Higher Education, 19 (2020), pp. 354–376.

[19] A.-S. V. BÉDARD, E. D. HIEN, AND D. A. LAFONTAINE, *Riboswitch regulation mechanisms: Rna, metabolites and regulatory proteins*, Biochimica et Biophysica Acta - Gene Regulatory Mechanisms, 1863 (2020), p. 194501.

[20] M. J. BELLECOURT, A. RAY-SONI, A. HARWIG, R. A. MOONEY, AND R. LANDICK, *Rna polymerase clamp movement aids dissociation from dna but is not required for rna release at intrinsic terminators*, Journal of Molecular Biology, 431 (2019), pp. 696–713.

[21] J. BENDELL AND R. READ, *Deep Adaptation: Navigating the Realities of Climate Chaos*, Polity, 2021.

[22] I. BERVOETS AND D. CHARLIER, *Diversity, versatility and complexity of bacterial gene regulation mechanisms: opportunities and drawbacks for applications in synthetic biology*, FEMS Microbiology Reviews, 43 (2019), pp. 304–339.

[23] C. BLANCO, E. JANZEN, A. PRESSMAN, R. SAHA, AND I. A. CHEN, *Molecular Fitness Landscapes from High-Coverage Sequence Profiling*, Annual Review Biophysics, 48 (2019), pp. 1–18.

[24] S. B. BLATTMAN, W. JIANG, P. OIKONOMOU, AND S. TAVAZOIE, *Prokaryotic single-cell RNA sequencing by in situ combinatorial indexing*, Nature Microbiology, 5 (2020), pp. 1192–1201.

[25] H. BOCHERENS, *The rise of the anthroposphere since 50,000 years: an ecological replacement of megaherbivores by humans in terrestrial ecosystems?*, Frontiers in Ecology and Evolution, (2018), p. 3.

[26] D. BOLLIER AND S. HELFRICH, *Free, Fair, and Alive: The insurgent power of the commons*, New Society Publishers, 2019.

[27] A. BOO, T. ELLIS, AND G.-B. STAN, *Host-aware synthetic biology*, Current Opinion in Systems Biology, 14 (2019), pp. 66–72.

[28] J. L. BRUNELLE AND R. GREEN, *Chapter five - in vitro transcription from plasmid or PCR-amplified DNA*, in Laboratory Methods in Enzymology: RNA, J. Lorsch, ed., vol. 530 of Methods in Enzymology, Academic Press, 2013, pp. 101–114.

[29] E. BRYNJOLFSSON AND A. MCAFEE, *The biggest winners: stars and superstars*, The Risk Institute, 2014.

[30] A. BYRNE, C. COLE, R. VOLDEN, AND C. VOLLMERS, *Realizing the potential of full-length transcriptome sequencing*, Philosophical Transactions of the Royal Society B, 374 (2019), p. 20190097.

[31] J. CALVERT, *Ownership and sharing in synthetic biology: A 'diverse ecology'of the open and the proprietary?*, BioSocieties, 7 (2012), pp. 169–187.

[32] G. CAMBRAY, J. C. GUIMARAES, AND A. P. ARKIN, *Evaluation of 244,000 synthetic sequences reveals design principles to optimize translation in Escherichia coli*, Nature Biotechnology, 36 (2018), pp. 1005–1015.

[33] G. CAMBRAY, J. C. GUIMARAES, V. K. MUTALIK, C. LAM, Q.-A. MAI, T. THIMMAIAH, J. M. CAROTHERS, A. P. ARKIN, AND D. ENDY, *Measurement and modeling of intrinsic transcription terminators*, Nucleic Acids Research, 41 (2013), pp. 5139–5148.

[34] C. C. CAMPA, N. R. WEISBACH, A. J. SANTINHA, D. INCARNATO, AND R. J. PLATT, *Multiplexed genome engineering by Cas12a and CRISPR arrays encoded on single transcripts*, Nature Methods, 16 (2019), pp. 887–893.

[35] B. CANTON, A. LABNO, AND D. ENDY, *Refinement and standardization of synthetic biological parts and devices*, Nature Biotechnology, 26 (2008), pp. 787–793.

[36] P. CARBONELL, A. J. JERVIS, C. J. ROBINSON, C. YAN, M. DUNSTAN, N. SWAINSTON, M. VINAIXA, K. A. HOLLYWOOD, A. CURRIN, N. J. W. RATTRAY, S. TAYLOR, R. SPIESS, R. SUNG, A. R. WILLIAMS, D. FELLOWS, N. J. STANFORD, P. MULHERIN, R. LE FEUVRE, P. BARRAN, R. GOODACRE, N. J. TURNER, C. GOBLE, G. G. CHEN, D. B. KELL, J. MICKLEFIELD, R. BREITLING, E. TAKANO, J. L. FAULON, AND N. S. SCRUTTON, *An automated Design-Build-Test-Learn pipeline for enhanced microbial production of fine chemicals*, Communications Biology, 1 (2018), p. 66.

[37] S. CARDINALE AND A. P. ARKIN, *Contextualizing context for synthetic biology–identifying causes of failure of synthetic biological systems*, Biotechnology Journal, 7 (2012), pp. 856–866.

[38] J. L. CARLIN, *Mutations Are the Raw Materials of Evolution*, Nature Education Knowledge, 3 (2011), p. 10.

[39] S. B. CARR, J. BEAL, AND D. M. DENSMORE, *Reducing DNA context dependence in bacterial promoters*, PLoS One, 12 (2017), p. e0176013.

[40] A. CASINI, G. CHRISTODOULOU, P. S. FREEMONT, G. S. BALDWIN, T. ELLIS, AND J. T. MACDONALD, *R2oDNA designer: computational design of biologically neutral synthetic DNA sequences*, ACS Synthetic Biology, 3 (2014), pp. 525–528.

[41] A. CASINI, M. STORCH, G. S. BALDWIN, AND T. ELLIS, *Bricks and blueprints: methods and standards for DNA assembly*, Nature Reviews Molecular Cell Biology, 16 (2015), pp. 568–576.

[42] S. D. CASTLE, C. S. GRIERSON, AND T. E. GOROCHOWSKI, *Towards an engineering theory of evolution*, Nature Communications, 12 (2021), p. 3326.

[43] F. CERONI, A. BOO, S. FURINI, T. E. GOROCHOWSKI, O. BORKOWSKI, Y. N. LADAK, A. R. AWAN, C. GILBERT, G. B. STAN, AND T. ELLIS, *Burden-driven feedback control of gene expression*, Nature Methods, 15 (2018), pp. 387–393.

[44] D. P. CETNAR AND H. M. SALIS, *Systematic quantification of sequence and structural determinants controlling mRNA stability in bacterial operons*, ACS Synthetic Biology, 10 (2021), pp. 318–332.

[45] J. CHAPPELL, A. WESTBROOK, V. M., AND L. J. B., *Design of small transcription activating rnas for versatile and dynamic gene regulation*, Nature Communications, 8 (2017), pp. 144–145.

[46] L. J. CHEN AND E. M. OROZCO, *Jr. recognition of prokaryotic transcription terminators by spinach chloroplast RNA polymerase*, Nucleic Acids Research, 16 (1988), pp. 8411–8431.

[47] Y.-J. CHEN, P. LIU, A. A. NIELSEN, J. A. BROPHY, K. CLANCY, T. PETERSON, AND C. A. VOIGT, *Characterization of 582 natural and synthetic terminators and quantification of their design constraints*, Nature Methods, 10 (2013), pp. 659–664.

[48] G. M. CHURCH AND S. KIEFFER-HIGGINS, *Multiplex DNA sequencing*, Science, 240 (1988), pp. 185–188.

[49] W. CUI, Q. LIN, R. HU, L. HAN, Z. CHENG, L. ZHANG, AND Z. ZHOU, *Data-driven and in silico-assisted design of broad host-range minimal intrinsic terminators adapted for bacteria*, ACS Synthetic Biology, 10 (2021), pp. 1438–1450.

[50] A. CURRIN, S. PARKER, C. J. ROBINSON, E. TAKANO, N. S. SCRUTTON, AND R. BREITLING, *The evolving art of creating genetic diversity: From directed evolution to synthetic biology*, Biotechnology Advances, 50 (2021), p. 107762.

[51] A. CURRIN, N. SWAINSTON, M. S. DUNSTAN, A. J. JERVIS, P. MULHERIN, C. J. ROBINSON, S. TAYLOR, P. CARBONELL, K. A. HOLLYWOOD, C. YAN, E. TAKANO, N. S. SCRUTTON, AND R. BREITLING, *Highly multiplexed, fast and accurate nanopore sequencing for verification of synthetic DNA constructs and sequence libraries*, Synthetic Biology, 4 (2019), p. ysz025.

[52] J. DABNEY AND M. MEYER, *Length and GC-biases during sequencing library amplification: a comparison of various polymerase-buffer systems with ancient and modern DNA sequencing libraries*, Biotechniques, 52 (2012), pp. 87–94.

[53] M. DAESCHEL, R. ANDERSSON, AND H. FLEMING, *Microbial ecology of fermenting plant materials**, FEMS Microbiology Reviews, 3 (1987), pp. 357–367.

[54] G. D'ALISA, F. DEMARIA, AND G. KALLIS, *Degrowth: a vocabulary for a new era*, Routledge, 2014.

[55] A. DANCHIN, *In vivo, in vitro and in silico: an open space for the development of microbe-based applications of synthetic biology*, Microbial Biotechnology, 15 (2022), pp. 42–64.

[56] D. DAR, D. PRASSE, R. A. SCHMITZ, AND R. SOREK, *Widespread formation of alternative 3' utr isoforms via transcription termination in archaea*, Nature Microbiology, 1 (2016), pp. 1–9.

[57] D. DAR, M. SHAMIR, J. MELLIN, M. KOUTERO, N. STERN-GINOSSAR, P. COSSART, AND R. SOREK, *Term-seq reveals abundant ribo-regulation of antibiotics resistance in bacteria*, Science, 352 (2016), p. aad9822.

[58] D. DAVIS, *Presenting research reflexivity in your phd thesis*, Nursing Research, 28 (2020), pp. 37–43.

[59] A. DAWSON, I. PAEGLIS, AND N. BASU, *Founder as steward or agent? a study of founder ownership and firm value*, Entrepreneurship Theory and Practice, 42 (2017), pp. 886–910.

[60] S. V. DE FREITAS NETTO, M. F. F. SOBRAL, A. R. B. RIBEIRO, AND G. R. DA LUZ SOARES, *Concepts and forms of greenwashing: A systematic review*, Environmental Sciences Europe, 32 (2020), pp. 1–12.

[61] B. DE SOUSA SANTOS, *Manifesto for good living / buen vivir*, in Epistemologies of the South, Routledge, 2014.

[62] M. DEANER AND H. S. ALPER, *Promoter and terminator discovery and engineering*, Springer, 2016.

[63] D. P. DEPLEDGE, K. P. SRINIVAS, T. SADAOKA, D. BREADY, Y. MORI, D. G. PLACANTONAKIS, I. MOHR, AND A. C. WILSON, *Direct RNA sequencing on nanopore arrays redefines the transcriptional complexity of a viral pathogen*, Nature Communications, 10 (2019), p. 754.

[64] B. S. DER, E. GLASSEY, B. A. BARTLEY, C. ENGHUUS, D. B. GOODMAN, D. B. GORDON, C. A. VOIGT, AND T. E. GOROCHOWSKI, *DNAplotlib: Programmable Visualization of Genetic Designs and Associated Data*, ACS Synthetic Biology, 6 (2017), pp. 1115–1119.

[65] N. DHILLON, R. SHELANSKY, B. TOWNSHEND, M. JAIN, H. BOEGER, D. ENDY, AND R. KAMAKAKA, *Permutational analysis of Saccharomyces cerevisiae regulatory elements*, Synthetic Biology, 5 (2020), pp. 1–14.

[66] L. DU, R. GAO, AND A. C. FORSTER, *Engineering multigene expression in vitro and in vivo with small terminators for T7 RNA polymerase*, Biotechnology and Bioengineering, 104 (2009), pp. 1189–1196.

[67] T. ELLIS, T. ADIE, AND G. S. BALDWIN, *DNA assembly for synthetic biology: from parts to pathways and beyond*, Integrative Biology, 3 (2011), pp. 109–118.

[68] D. ENDY, *Foundations for engineering biology*, Nature, 438 (2005), pp. 449–453.

[69] V. EPSHTEIN, C. J. CARDINALE, A. E. RUCKENSTEIN, S. BORUKHOV, AND E. NUDLER, *An allosteric path to transcription termination*, Molecular Cell, 28 (2007), pp. 991–1001.

[70] V. EPSHTEIN AND E. NUDLER, *Cooperation between RNA polymerase molecules in transcription elongation*, Science, 300 (2003), pp. 801–805.

[71] V. EPSHTEIN, F. TOULMÉ, A. R. RAHMOUNI, S. BORUKHOV, AND E. NUDLER, *Transcription through the roadblocks: the role of RNA polymerase cooperation*, EMBO Journal, 22 (2003), pp. 4719–4727.

[72] A. ESCOBAR-ZEPEDA, A. VERA-PONCE DE LEON, AND A. SANCHEZ-FLORES, *The Road to Metagenomics: From Microbiology to DNA Sequencing Technologies and Bioinformatics*, Frontiers in Genetics, 6 (2015), p. 348.

[73] A. ESPAH BORUJENI, J. ZHANG, H. DOOSTHOSSEINI, A. A. K. NIELSEN, AND C. A. VOIGT, *Genetic circuit characterization by inferring RNA polymerase movement and ribosome usage*, Nature Communications, 11 (2020), p. 5001.

[74] F. O. FAPOHUNDA, S. QIAO, Y. PAN, H. WANG, Y. LIU, Q. CHEN, AND P. LÜ, *CRISPR Cas system: A strategic approach in detection of nucleic acids*, Microbiological Research, 259 (2022), p. 127000.

[75] S. H. FARJANA, N. HUDA, M. P. MAHMUD, AND R. SAIDUR, *A review on the impact of mining and mineral processing industries through life cycle assessment*, Journal of cleaner production, 231 (2019), pp. 1200–1217.

[76] J. L. FAULON AND L. FAURE, *In silico, in vitro, and in vivo machine learning in synthetic biology and metabolic engineering*, Current Opinion in Chemical Biology, 65 (2021), pp. 85–92.

[77] J. FIKSEL, *Sustainability and resilience: toward a systems approach*, Sustainability: Science, Practice and Policy, 2 (2006), pp. 14–21.

[78] A. E. FIRTH AND W. M. G.-I. A. PATRICK, *and pedel-aa: new programmes for analyzing protein diversity in randomized libraries*, Nucleic Acids Research, 36 (2008), pp. W281–5.

[79] J. FONTANA, C. DONG, J. Y. HAM, J. G. ZALATAN, AND J. M. CAROTHERS, *Regulated expression of sgRNAs tunes CRISPRi in E. coli*, Biotechnology Journal, 13 (2018), p. 69.

[80] T. J. FOXON, *Technological lock-in and the role of innovation*, in Handbook of Sustainable Development, G. Atkinson, S. Dietz, E. Neumayer, and M. Agarwala, eds., Chapters, Edward Elgar Publishing, 2014, ch. 20, pp. 304–316.

[81] Y. FUJITANI, K. YAMAMOTO, AND I. KOBAYASHI, *Dependence of frequency of homologous recombination on the homology length.*, Genetics, 140 (1995), pp. 797–809.

[82] J. E. GALLEGOS, M. F. ROGERS, C. A. CIALEK, AND J. R. PECCOUD, *robust plasmid verification by de novo assembly of short sequencing reads*, Nucleic Acids Research, 48 (2020), p. e106.

[83] D. R. GARALDE, E. A. SNELL, D. JACHIMOWICZ, B. SIPOS, J. H. LLOYD, M. BRUCE, N. PANTIC, T. ADMASSU, P. JAMES, A. WARLAND, M. JORDAN, J. CICCONE, S. SERRA, J. KEENAN, S. MARTIN, L. MCNEILL, E. J. WALLACE, L. JAYASINGHE, C. WRIGHT, J. BLASCO, S. YOUNG, D. BROCKLEBANK, S. JUUL, J. CLARKE, A. J. HERON, AND D. J. TURNER, *Highly parallel direct RNA sequencing on an array of nanopores*, Nature Methods, 15 (2018), pp. 201–206.

[84] D. GARENNE AND V. NOIREAUX, *Cell-free transcription–translation: engineering biology from the nanometer to the millimeter scale*, 2019.

[85] M. GASPERINI, L. STARITA, AND J. SHENDURE, *The power of multiplexed functional analysis of genetic variants*, Nature Protocols, 11 (2016), pp. 1782–1787.

[86] M. GEIS, C. FLAMM, M. T. WOLFINGER, A. TANZER, I. L. HOFACKER, M. MIDDENDORF, C. MANDL, P. F. STADLER, AND C. THURNER, *Folding kinetics of large RNAs*, J Mol Biol, 379 (2008), pp. 160–173.

[87] K. GESZVAIN AND R. LANDICK, *The structure of bacterial rna polymerase*, The Bacterial Chromosome, (2004), pp. 283–296.

[88] A. GHELFI AND D. PAPADOPOULOS, *Ecological transition: What it is and how to do it. community technoscience and green democracy*, Tecnoscienza: Italian Journal of Science & Technology Studies, 12 (2022), pp. 13–38.

[89] P.-A. GILLIOT AND T. E. GOROCHOWSKI, *Sequencing enabling design and learning in synthetic biology*, Current Opinion in Chemical Biology, 58 (2020), pp. 54–62.

[90] B. GLASER, *Prehistorically modified soils of central amazonia: a model for sustainable agriculture in the twenty-first century*, Philosophical Transactions of the Royal Society B: Biological Sciences, 362 (2007), pp. 187–196.

[91] C. GLERUP, S. R. DAVIES, AND M. HORST, *'Nothing really responsible goes on here': scientists' experience and practice of responsibility*, Journal of Responsible Innovation, 4 (2017), pp. 319–336.

[92] J. T. GODBOUT AND A. C. CAILLE, *World of the Gift*, McGill-Queen's Press-MQUP, 1998.

[93] M. GODMAN AND S. O. HANSSON, *European public advice on nanobiotechnology—four convergence seminars*, NanoEthics, 3 (2009), pp. 43–59.

[94] S. GOODWIN, J. D. MCPHERSON, AND W. R. MCCOMBIE, *Coming of age: ten years of next-generation sequencing technologies*, Nature Reviews Genetics, 17 (2016), pp. 333–351.

[95] T. E. GOROCHOWSKI, I. AVCILAR-KUCUKGOZE, R. A. L. BOVENBERG, J. A. ROUBOS, AND Z. A. IGNATOVA, *Minimal model of ribosome allocation dynamics captures trade-offs in expression between endogenous and synthetic genes*, ACS Synthetic Biology, 5 (2016), pp. 710–720.

[96] T. E. GOROCHOWSKI, I. CHELYSHEVA, M. ERIKSEN, P. NAIR, S. PEDERSEN, AND Z. IGNATOVA, *Absolute quantification of translational regulation and burden using combined sequencing approaches*, Molecular Systems Biology, 15 (2019), p. e8719.

[97] T. E. GOROCHOWSKI AND T. ELLIS, *Designing efficient translation*, Nature Biotechnology, 36 (2018), pp. 934–935.

[98] T. E. GOROCHOWSKI, A. ESPAH BORUJENI, Y. PARK, A. A. K. NIELSEN, J. ZHANG, B. S. DER, D. B. GORDON, AND C. A. VOIGT, *Genetic circuit characterization and debugging using RNA-seq*, Molecular Systems Biology, 13 (2017), p. 952.

[99] T. E. GOROCHOWSKI, E. VAN DEN BERG, R. KERKMAN, J. A. ROUBOS, AND R. A. L. BOVENBERG, *Using synthetic biological parts and microbioreactors to explore the protein expression characteristics of escherichia coli*, ACS Synthetic Biology, 3 (2014), pp. 129–139.

[100] D. GRAEBER AND D. WENGROW, *The dawn of everything: A new history of humanity*, Penguin UK, 2021.

[101] M. W. GRAY, *Mitochondrial evolution*, Cold Spring Harbor Perspectives in Biology, 4 (2012), p. a011403.

[102] V. GRECO, M. TARNOWSKI, AND T. GOROCHOWSKI, *Living computers powered by biochemistry.*, The Biochemist, 41 (2019), pp. 14–18.

[103] A. A. GREEN, J. KIM, D. MA, P. A. SILVER, J. J. COLLINS, AND P. YIN, *Complex cellular logic computation using ribocomputing devices*, Nature, 548 (2017), pp. 117–121.

[104] A. R. GRUBER, R. LORENZ, S. H. BERNHART, R. NEUB"OCK, AND I. L. HOFACKER, *The Vienna RNA websuite*, Nucleic Acids Research, 36 (2008), pp. W70–4.

[105] F. GRUNBERGER, S. FERREIRA-CERCA, AND D. GROHMANN, *Nanopore sequencing of RNA and cDNA molecules expands the transcriptomic toolbox in prokaryotes*, Biorxiv, (2021).

[106] H. GUBBY, *Is the patent system a barrier to inclusive prosperity? the biomedical perspective*, Global Policy, 11 (2020), pp. 46–55.

[107] I. GUSAROV AND E. NUDLER, *The mechanism of intrinsic transcription termination*, Molecular Cell, 3 (1999), pp. 495–504.

[108] I. GUSAROV AND E. NUDLER, *The mechanism of intrinsic transcription termination*, Molecular Cell, 3 (1999), pp. 495–504.

[109] D. HARAWAY, *Anthropocene, Capitalocene, Plantationocene, Chthulucene: Making Kin*, Environmental Humanities, 6 (2015), pp. 159–165.

[110] R. E. HAURWITZ, S. H. STERNBERG, AND J. A. DOUDNA, *Csy4 relies on an unusual catalytic dyad to position and cleave CRISPR RNA*, The EMBO Journal, 31 (2012), pp. 2824–2832.

[111] Z. HE, Y. DUAN, W. ZHAI, X. ZHANG, J. SHI, X. ZHANG, AND Z. XU, *Evaluating terminator strength based on differentiating effects on transcription and translation*, ChemBioChem, 21 (2020), pp. 2067–2072.

[112] C. E. G. HERRERA, *Microbes and other shamanic beings*, Springer, 2018.

[113] P. HIGGINS, D. SHORT, AND D. SOUTH, *Protecting the planet: a proposal for a law of ecocide*, Crime, Law and Social Change, 59 (2013), pp. 419–426.

[114] M. HORI, H. FUKANO, AND Y. SUZUKI, *Uniform amplification of multiple DNAs by emulsion PCR*, Biochemical and Biophysical Research Communications, 352 (2007), pp. 323–328.

[115] T. HU, N. CHITNIS, M. DIMITRI, AND A. DINH, *Next-generation sequencing technologies: An overview*, Human Immunology, 82 (2021), pp. 801–811.

[116] A. J. HUDSON AND H.-J. WIEDEN, *Rapid generation of sequence-diverse terminator libraries and their parameterization using quantitative term-seq*, Synthetic Biology, 4 (2019).

[117] I. ILLICH, *Tools for conviviality*, Harper and Row, 2018.

[118] I. IOST, J. GUILLEREZ, AND M. DREYFUS, *Bacteriophage t7 rna polymerase travels far ahead of ribosomes in vivo*, Journal of Bacteriology, 174 (1992), pp. 619–622.

[119] C. L. IP, M. LOOSE, J. R. TYSON, M. DE CESARE, B. L. BROWN, M. JAIN, R. M. LEGGETT, D. A. ECCLES, V. ZALUNIN, J. M. URBAN, ET AL., *Minion analysis and reference consortium: Phase 1 data release and analysis*, F1000Research, 4 (2015).

[120] M. IRASTORTZA-OLAZIREGI AND O. AMSTER-CHODER, *Coupled Transcription-Translation in Prokaryotes: An Old Couple With New Surprises*, Frontiers in Microbiology, 11 (2021).

[121] B. R. JACK, D. R. BOUTZ, M. L. PAFF, B. L. SMITH, AND C. O. WILKE, *Transcript degradation and codon usage regulate gene expression in a lytic phage†*, Virus Evolution, 5 (2019).

[122] S. JASANOFF, *The ethics of invention: Technology and the human future*, WW Norton & Company, 2016.

[123] S. T. JENG, J. F. GARDNER, AND R. I. GUMPORT, *Transcription termination by bacteriophage T7 RNA polymerase at Rho-independent terminators*, Journal of Biological Chemistry, 265 (1990), pp. 3823–3830.

[124] M. JESCHEK, D. GERNGROSS, AND S. PANKE, *Combinatorial pathway optimization for streamlined metabolic engineering*, Current Opinion in Biotechnology, 47 (2017), pp. 142–151.

[125] N. I. JOHNS, A. L. C. GOMES, S. S. YIM, A. YANG, T. BLAZEJEWSKI, C. S. SMILLIE, M. B. SMITH, E. J. ALM, S. KOSURI, AND H. H. WANG, *Metagenomic mining of regulatory elements enables programmable species-selective gene expression*, Nature Methods, 15 (2018), p. 323–329.

[126] X. JU, D. LI, AND S. LIU, *Full-length RNA profiling reveals pervasive bidirectional transcription terminators in bacteria*, Nature Microbiology, 4 (2019), pp. 1907–1918.

[127] J. K. JUNG, K. K. ALAM, M. S. VEROSLOFF, D. A. CAPDEVILA, M. DESMAU, P. R. CLAUER, J. W. LEE, P. Q. NGUYEN, P. A. PASTÉN, S. J. MATIASEK, J. F. GAILLARD, D. P. GIEDROC, J. J. COLLINS, AND J. B. LUCKS, *Cell-free biosensors for rapid detection of water contaminants*, Nature Biotechnology, 38 (2020), pp. 1451–1459.

[128] J. Y. KANG, T. V. MISHANINA, M. J. BELLECOURT, R. A. MOONEY, S. A. DARST, AND R. LANDICK, *Rna polymerase accommodates a pause rna hairpin by global conformational rearrangements that prolong pausing*, Molecular Cell, 69 (2018), pp. 802–815.

[129] W. KANG, K. S. HA, H. UHM, K. PARK, J. Y. LEE, S. HOHNG, AND C. KANG, *Transcription reinitiation by recycling rna polymerase that diffuses on dna after releasing terminated rna*, Nature Communications, 11 (2020), pp. 1–9.

[130] S. M. KARST, R. M. ZIELS, R. H. KIRKEGAARD, E. A. SØRENSEN, D. MCDONALD, Q. ZHU, R. KNIGHT, AND M. ALBERTSEN, *High-accuracy long-read amplicon sequences using unique molecular identifiers with Nanopore or PacBio sequencing*, Nature Methods, 18 (2021), pp. 165–169.

[131] K. KATOH, K. MISAWA, K. KUMA, AND T. MIYATA, *MAFFT: a novel method for rapid multiple sequence alignment based on fast Fourier transform*, Nucleic Acids Research, 30 (2002), pp. 3059–3066.

[132] S. E. KATZ, *The Art of Fermentation*, Chelsea Green Publishing, 2012.

[133] C. L. KELLY, G. M. TAYLOR, A. ŠATKUTĖ, L. DEKKER, AND J. T. HEAP, *Transcriptional Terminators Allow Leak-Free Chromosomal Integration of Genetic Constructs in Cyanobacteria*, Microorganisms, 7 (2019), p. 263.

[134] C. KERSCHNER AND M.-H. EHLERS, *framework of attitudes towards technology in theory and practice.*, Ecological Economics, 126 (2016), pp. 139–151.

[135] S. C. KIM, G. PREMASEKHARAN, I. C. CLARK, H. B. GEMEDA, P. L. PARIS, AND A. R. ABATE, *Measurement of copy number variation in single cancer cells using rapid-emulsification digital droplet MDA*, Microsystems and Nanoengineering, 3 (2017).

[136] R. W. KIMMERER, *Braiding Sweetgrass: Indigenous Wisdom, Scientific Knowledge and the Teachings of Plants*, Milkweed Editions, 2020.

[137] E. KIRKSEY, *Living machines go wild - policing the imaginative horizons of synthetic biology*, Current Anthropology, 62 (2021), pp. s287–s297.

[138] H. K. KLEIN AND D. L. KLEINMAN, *The social construction of technology: Structural considerations*, Science, Technology and Human Values, 27 (2002), pp. 28–52.

[139] J. C. KLEIN, M. J. LAJOIE, J. J. SCHWARTZ, E. M. STRAUCH, J. NELSON, D. BAKER, AND J. SHENDURE, *Multiplex pairwise assembly of array-derived DNA oligonucleotides*, Nucleic Acids Research, (2015).

[140] S. KOCHETKOV, E. RUSAKOVA, AND V. TUNITSKAYA, *Recent studies of t7 rna polymerase mechanism*, FEBS letters, 440 (1998), pp. 264–267.

[141] T. KOMANO, *Shufflons: multiple inversion systems and integrons*, Annual Review of Genetics, 33 (1999), pp. 171–191.

[142] N. KOMISSAROVA, J. BECKER, S. SOLTER, M. KIREEVA, AND M. KASHLEV, *Shortening of RNA: DNA hybrid in the elongation complex of RNA polymerase is a prerequisite for transcription termination*, Molecular Cell, 10 (2002), pp. 1151–1162.

[143] E. V. KOONIN, *Does the central dogma still stand?*, Biology Direct, 7 (2012), p. 27.

[144] S. KOSURI AND G. M. CHURCH, *Large-scale de novo DNA synthesis: Technologies and applications*, Nature Methods, 11 (2014), pp. 499–507.

[145] S. KOSURI, D. B. GOODMAN, G. CAMBRAY, V. K. MUTALIK, Y. GAO, A. P. ARKIN, D. ENDY, AND G. M. CHURCH, *Composability of regulatory sequences controlling transcription and translation in Escherichia coli*, Proceedings of the National Academy of Sciences, 110 (2013), p. 14024–14029.

[146] B. J. KOTOPKA AND C. D. SMOLKE, *Model-driven generation of artificial yeast promoters*, Nature Communications, 11 (2020), p. 2113.

[147] M. KUSHWAHA, W. ROSTAIN, S. PRAKASH, J. N. DUNCAN, AND A. JARAMILLO, *Using RNA as Molecular Code for Programming Cellular Function*, ACS Synthetic Biology, 5 (2016), pp. 795–809.

[148] N. F. LAHENS, I. KAVAKLI, R. ZHANG, K. HAYER, M. B. BLACK, H. DUECK, A. PIZARRO, J. KIM, R. IRIZARRY, R. S. THOMAS, G. R. GRANT, AND J. B. HOGENESCH, *IVT-seq reveals extreme bias in RNA sequencing*, Genome Biology, 15 (2014), p. R86.

[149] J.-B. LALANNE, J. C. TAGGART, M. S. GUO, L. HERZEL, A. SCHIELER, AND G.-W. LI, *Evolutionary convergence of pathway-specific enzyme expression stoichiometry*, Cell, 173 (2018), pp. 749–761.

[150] M. H. LARSON, W. J. GREENLEAF, R. LANDICK, AND S. M. BLOCK, *Applied force reveals mechanistic and energetic details of transcription termination*, Cell, 132 (2008), pp. 971–982.

[151] M. LEACH, J. ROCKSTRÖM, P. RASKIN, I. SCOONES, A. C. STIRLING, A. SMITH, J. THOMPSON, E. MILLSTONE, A. ELY, E. AROND, C. FOLKE, AND P. OLSSON, *Transforming innovation for sustainability*, Ecology and Society, 17 (2012).

[152] D. LEGER, S. MATASSA, E. NOOR, A. SHEPON, R. MILO, AND A. BAR-EVEN, *Photovoltaic-driven microbial protein production can use land and sunlight more efficiently than conventional crops*, Proceedings of the National Academy of Sciences, 118 (2021).

[153] R. E. LENSKI, *Experimental evolution and the dynamics of adaptation and genome evolution in microbial populations*, ISME: Multidisciplinary Journal of Microbial Ecology, 11 (2017), pp. 2181–2194.

[154] J. LENT, *The patterning instinct*, Prometheus Books, 2017.

[155] B. T. LEWICKI, T. MARGUS, J. REMME, AND K. H. NIERHAUS, *Coupling of rrna transcription and ribosomal assembly in vivo: Formation of active ribosomal subunits in escherichia coli requires transcription of rrna genes by host rna polymerase which cannot be replaced by bacteriophage t7 rna polymerase*, Journal of Molecular Biology, 231 (1993), pp. 581–593.

[156] H. LI, *Minimap2, pairwise alignment for nucleotide sequences*, Bioinformatics, 34 (2018), pp. 3094–3100.

[157] H. LI, B. HANDSAKER, A. WYSOKER, T. FENNELL, J. RUAN, N. HOMER, G. MARTH, G. ABECASIS, AND R. DURBIN, *The Sequence Alignment/Map format and SAMtools*, Bioinformatics, 25 (2009), pp. 2078–2079.

[158] H. M. LI, *pairwise alignment for nucleotide sequences*, Bioinformatics, 34 (2018), pp. 3094–3100.

[159] R. LI, Q. ZHANG, J. LI, AND H. SHI, *Effects of cooperation between translating ribosome and RNA polymerase on termination efficiency of the rho-independent terminator*, Nucleic Acids Research, 44 (2016), pp. 2554–2563.

[160] X. LIANG, C. LI, W. WANG, AND Q. LI, *Integrating T7 RNA polymerase and its cognate transcriptional units for a host-independent and stable expression system in single plasmid*, ACS Synthetic Biology, 7 (2018), pp. 1424–1435.

[161] D. D. LICATALOSI, A. MELE, J. J. FAK, J. ULE, M. KAYIKCI, S. W. CHI, T. A. CLARK, A. C. SCHWEITZER, J. E. BLUME, X. WANG, J. C. DARNELL, AND R. B. DARNELL, *HITS-CLIP yields genome-wide insights into brain alternative RNA processing*, Nature, 456 (2008), pp. 464–469.

[162] P. LINEBAUGH AND M. REDIKER, *The Many-Headed Hydra: Sailors, Slaves, and the Atlantic Working Class in the Eighteenth Century*, Verso Books, 1990.

[163] G. LISZCZAK AND T. W. MUIR, *Nucleic Acid-Barcoding Technologies: Converting DNA Sequencing into a Broad-Spectrum Molecular Counter*, Angewandte Chemie - International Edition, 58 (2019), pp. 4144–4162.

[164] C. C. LIU, M. C. JEWETT, J. W. CHIN, AND C. A. VOIGT, *Toward an orthogonal central dogma*, Nature Chemical Biology, 14 (2018), pp. 103–106.

[165] S. S. LIU, A. J. HOCKENBERRY, A. LANCICHINETTI, M. C. JEWETT, AND L. A. N. AMARAL, *Nullseq: A tool for generating random coding sequences with desired amino acid and gc contents*, PLoS Computational Biology, 12 (2016), p. e1005184.

[166] C. LOOD, H. GERSTMANS, Y. BRIERS, V. VAN NOORT, AND R. LAVIGNE, *Quality control and statistical evaluation of combinatorial DNA libraries using nanopore sequencing*, Biotechniques, 69 (2020), pp. 379–383.

[167] C. LOU, B. STANTON, Y.-J. CHEN, B. MUNSKY, AND C. A. VOIGT, *Ribozyme-based insulator parts buffer synthetic circuits from genetic context*, Nature Biotechnology, 30 (2012), pp. 1137–1142.

[168] A. A. LOUIS, *Contingency, convergence and hyper-astronomical numbers in biological evolution*, Studies in History and Philosophy of Biological and Biomedical Sciences, (2016), pp. 107–116.

[169] L. G. LOWDER, D. ZHANG, N. J. BALTES, J. W. PAUL, X. TANG, X. ZHENG, D. F. VOYTAS, T. F. HSIEH, Y. ZHANG, AND Y. QI, *A CRISPR/Cas9 Toolbox for Multiplexed Plant Genome Editing and Transcriptional Regulation*, Plant Physiology, 169 (2015), pp. 971–985.

[170] L. LUBKOWSKA, A. S. MAHARJAN, AND N. KOMISSAROVA, *Rna folding in transcription elongation complex: implication for transcription termination*, Journal of Biological Chemistry, 286 (2011), pp. 31576–31585.

[171] L. LUO, E. VAN DER VOET, AND G. HUPPES, *Life cycle assessment and life cycle costing of bioethanol from sugarcane in brazil*, Renewable and Sustainable Energy Reviews, 13 (2009), pp. 1613–1619.

[172] D. L. LYAKHOV, B. HE, X. ZHANG, F. W. STUDIER, J. J. DUNN, AND W. T. MCALLISTER, *Pausing and termination by bacteriophage T7 RNA polymerase*, Journal of Molecular Biology, 280 (1998), pp. 201–213.

[173] M. LYNCH, M. S. ACKERMAN, J. F. GOUT, H. LONG, W. SUNG, W. K. THOMAS, AND P. L. FOSTER, *Genetic drift, selection and the evolution of the mutation rate*, Nature Reviews Genetics, 17 (2016), pp. 704–714.

[174] T. LYNCH, *Writing up your phd (qualitative research) independent study version*, (2014).

[175] L. E. MACDONALD, R. K. DURBIN, J. J. DUNN, AND W. T. MCALLISTER, *Characterization of two types of termination signal for bacteriophage T7 RNA polymerase*, Journal of Molecular Biology, 238 (1994), pp. 145–158.

[176] L. E. MACDONALD, R. K. DURBIN, J. J. DUNN, AND W. T. MCALLISTER, *Characterization of two types of termination signal for bacteriophage T7 RNA polymerase*, Journal of Molecular Biology, 238 (1994), pp. 145–158.

[177] S. Y. MAEZUMI, D. ALVES, M. ROBINSON, J. G. DE SOUZA, C. LEVIS, R. L. BARNETT, E. ALMEIDA DE OLIVEIRA, D. URREGO, D. SCHAAN, AND J. IRIARTE, *The legacy of 4,500 years of polyculture agroforestry in the eastern amazon*, Nature plants, 4 (2018), pp. 540–547.

[178] J. MAHILLON AND M. CHANDLER, *Insertion sequences*, American Society for Microbiology, (1999).

[179] J. MAIRHOFER, A. WITTWER, M. CSERJAN-PUSCHMANN, AND G. STRIEDNER, *Preventing t7 RNA polymerase read-through transcription-a synthetic termination signal capable of improving bioprocess stability*, ACS Synthetic Biology, 4 (2015), pp. 265–273.

[180] S. MANRUBIA, J. A. CUESTA, J. AGUIRRE, S. E. AHNERT, L. ALTENBERG, A. V. CANO, P. CATALÁN, R. DIAZ-URIARTE, S. F. ELENA, J. A. GARCÍA-MARTÍN, P. HOGEWEG, B. S. KHATRI, J. KRUG, A. A. LOUIS, N. S. MARTIN, J. L. PAYNE, M. J. TARNOWSKI, AND M. WEISS, *From genotypes to organisms: State-of-the-art and perspectives of a cornerstone in evolutionary dynamics*, Physics of Life Reviews, 38 (2021), pp. 55–106.

[181] A. MARTELLA, M. FIRTH, B. J. M. TAYLOR, A. GÖPPERT, E. M. CUOMO, R. G. ROTH, A. J. DICKSON, AND D. I. FISHER, *Systematic Evaluation of CRISPRa and CRISPRi Modalities Enables Development of a Multiplexed, Orthogonal Gene Activation and Repression System*, ACS Synthetic Biology, 8 (2019), pp. 1998–2006.

[182] B. R. MARTIN, P. NIGHTINGALE, AND A. YEGROS-YEGROS, *Science and technology studies: Exploring the knowledge base*, Research Policy, 41 (2012), pp. 1182–1204.

[183] N. S. MCCARTY, A. E. GRAHAM, L. STUDENÁ, AND R. LEDESMA-AMARO, *Multiplexed CRISPR technologies for gene editing and transcriptional regulation*, Nature Methods, 11 (2020), p. 1281.

[184] J. S. MCCASKILL, *The equilibrium partition function and base pair binding probabilities for rna secondary structure*, Biopolymers, 29 (1990), pp. 1105–1119.

[185] M. K. MCDONALD, *Emergent: Rewilding Nature, Regenerating Food and Healing the World by Restoring the Connection Between People and the Wild*, Earth Books, 2022.

[186] P. MENENDEZ-GIL AND A. TOLEDO-ARANA, *Bacterial 3'utrs: A useful resource in post-transcriptional regulation*, Frontiers in Molecular Biosciences, 7 (2020), p. 3.

[187] F. MENG AND T. ELLIS, *The second decade of synthetic biology: 2010-2020*, Nature Communications, 11 (2020), p. 5174.

[188] I. M. MEYER AND I. MIKLOS, *Co-transcriptional folding is encoded within RNA genes*, BMC molecular biology, 5 (2004), pp. 1–10.

[189] N. MINSHALL AND A. GIT, *Enzyme- and gene-specific biases in reverse transcription of RNA raise concerns for evaluating gene expression*, Nature Scientific Reports, 10 (2020), p. 8151.

[190] A. MITRA, K. ANGAMUTHU, H. V. JAYASHREE, AND V. NAGARAJA, *Occurrence, divergence and evolution of intrinsic terminators across Eubacteria*, Genomics, 94 (2009), pp. 110–116.

[191] V. MOLODTSOV, M. ANIKIN, AND W. T. MCALLISTER, *The Presence of an RNA:DNA Hybrid That Is Prone to Slippage Promotes Termination by T7 RNA Polymerase*, Journal of Molecular Biology, 426 (2014), pp. 3095–3107.

[192] R. A. MOONEY AND R. LANDICK, *Building a better stop sign: Understanding the signals that terminate transcription*, Nature Methods, 10 (2013), pp. 618–619.

[193] M. MORANGE, *The black box of biology*, Harvard University Press, 2020.

[194] M. W. MURPHY AND C. SCHROERING, *Refiguring the plantationocene: Racial capitalism, world-systems analysis, and global socioecological transformation*, Journal of World-Systems Research, 26 (2020).

[195] M. MUSTONEN, *Copyleft—the economics of linux and other open source software*, Information Economics and Policy, 15 (2003), pp. 99–121.

[196] V. K. MUTALIK, J. C. GUIMARAES, G. CAMBRAY, C. LAM, M. J. CHRISTOFFERSEN, Q.-A. MAI, A. B. TRAN, M. PAULL, J. D. KEASLING, A. P. ARKIN, AND D. ENDY, *Precise and reliable gene expression via standard transcription and translation initiation elements*, Nature Methods, 10 (2013), pp. 354–360.

[197] G. NASERI AND M. A. G. KOFFAS, *Application of combinatorial optimization strategies in synthetic biology*, Nature Communications, 11 (2020), p. 2446.

[198] A. A. NIELSEN, B. S. DER, J. SHIN, P. VAIDYANATHAN, V. PARALANOV, E. A. STRYCHALSKI, D. ROSS, D. DENSMORE, AND C. A. VOIGT, *Genetic circuit design automation*, Science, 352 (2016), p. aac7341.

[199] A. OSHLACK AND M. J. WAKEFIELD, *Transcript length bias in rna-seq data confounds systems biology*, Biology Direct, 4 (2009), p. 14.

[200] R. OWEN, T. BAXTER, D. MAYNARD, AND M. DEPLEDGE, *Beyond regulation - risk pricing and responsible innovation*, Environmental Science and Technology, 43 (2009), p. 6902–6906.

[201] R. OWEN AND M. PANSERA, *Responsible innovation and responsible research and innovation.*, in Handbook on Science and Public Policy, Edward Elgar Publishing Ltd, 2019, pp. 26–48.

[202] R. OWEN, J. STILGOE, P. MACNAGHTEN, M. GORMAN, E. FISHER, AND D. GUSTON, *A framework for responsible innovation*, in Responsible Innovation: Managing the Responsible Emergence of Science and Innovation in Society, John Wiley and Sons, Ltd, 2013.

[203] M. PANSERA AND M. FRESSOLINI, *Innovation without growth - frameworks for understanding technological change in a post-growth era*, Organization, 28 (2020), pp. 380–404.

[204] M. PANSERA, R. OWEN, D. MEACHAM, AND V. KUH, *Embedding responsible innovation within synthetic biology research and innovation: insights from a uk multi-disciplinary research centre.*, Journal of Responsible Innovation, 7 (2020), pp. 384–409.

[205] V. PARACCHINI, M. PETRILLO, R. REITING, A. ANGERS-LOUSTAU, D. WAHLER, A. STOLZ, B. SCHÖNIG, A. MATTHIES, J. BENDIEK, D. M. MEINEL, S. PECORARO, U. BUSCH, A. PATAK, J. KREYSA, AND L. GROHMANN, *Molecular characterization of an unauthorized genetically modified bacillus subtilis production strain identified in a vitamin b2 feed additive*, Food Chemistry, 230 (2017), pp. 681–689.

[206] Y. PARK, E. BORUJENI, A, GOROCHOWSKI, T. E., J. SHIN, AND C. A. VOIGT, *Precision design of stable genetic circuits carried in highly-insulated e. coli genomic landing pads*, Molecular Systems Biology, 16 (2020), p. e9584.

[207] W. M. PATRICK, A. E. FIRTH, AND J. M. BLACKBURN, *User-friendly algorithms for estimating completeness and diversity in randomized protein-encoding libraries*, Protein Engineering, 16 (2003), pp. 451–457.

[208] M. PEPLOW, *Synthetic biology's first malaria drug meets market resistance*, Nature, 530 (2016), p. 389–390.

[209] J. M. PETERS, A. D. VANGELOFF, AND R. LANDICK, *Bacterial transcription terminators: The RNA 3'-end chronicles*, Journal of Molecular Biology, 412 (2011), pp. 793–813.

[210] B. F. PFLEGER, D. J. PITERA, C. D. SMOLKE, AND J. D. KEASLING, *Combinatorial engineering of intergenic regions in operons tunes expression of multiple genes*, Nature Biotechnology, 24 (2006), pp. 1027–1032.

[211] T. PIKKETY, *Capital in the Twenty-First Century*, Harvard University Press, 2014.

[212] O. PORRUA, M. BOUDVILLAIN, AND D. LIBRI, *Transcription termination: Variations on common themes*, Trends in Genetics, 32 (2016), pp. 508–522.

[213] V. POTAPOV, J. L. ONG, R. B. KUCERA, B. W. LANGHORST, K. BILOTTI, J. M. PRYOR, E. J. CANTOR, B. CANTON, T. F. KNIGHT, T. C. EVANS, AND G. J. S. LOHMAN, *Comprehensive Profiling of Four Base Overhang Ligation Fidelity by T4 DNA Ligase and Application to DNA Assembly*, ACS Synthetic Biology, 7 (2018), pp. 2665–2674.

[214] X. POUX AND P.-M. AUBERT, *An agroecological europe in 2050: multifunctional agriculture for healthy eating*, 9 (2018), p. 18.

[215] L. PRAY, *Discovery of DNA structure and function: Watson and Crick.*, Nature Education, 1 (2008), p. 100.

[216] B. PRESTON, *Synthetic biology as red herring*, Studies in History and Philosophy of Biological and Biomedical Sciences, 44 (2013), pp. 649–659.

[217] N. PROFIT VENTURES, *Post-growth innovation infographic*, (2022).

[218] PURPOSE ECONOMY, *Steward ownership*, (2022).

[219] D. QUAMMEN, *The Tangled Tree: A Radical New History of Life*, William Collin, 2012.

[220] B. J. RASOR, B. VÖGELI, G. M. LANDWEHR, J. W. BOGART, A. S. KARIM, AND M. C. JEWETT, *Toward sustainable, cell-free biomanufacturing*, Current Opinion in Biotechnology, 69 (2021), pp. 136–144.

[221] K. RAWORTH, *Doughnut economics: seven ways to think like a 21st-century economist*, Chelsea Green Publishing, 2017.

[222] A. RAY-SONI, M. J. BELLECOURT, AND R. LANDICK, *Mechanisms of bacterial transcription termination: all good things must end*, Annual Review of Biochemistry, 85 (2016), pp. 319–347.

[223] A. C. REIS, S. M. HALPER, G. E. VEZEAU, D. P. CETNAR, A. HOSSAIN, P. R. CLAUER, AND H. M. SALIS, *Simultaneous repression of multiple bacterial genes using nonrepetitive extra-long sgRNA arrays*, Nature Biotechnology, 37 (2019), pp. 1294–1301.

[224] B. RIBEIRO AND P. SHAPIRA, *Private and public values of innovation: A patent analysis of synthetic biology*, Research Policy, 49 (2020), p. 103875.

[225] A. RIP AND R. KEMP, *Technological change*, in Human choice and climate change, S. Rayner and E. Malone, eds., Battelle Press, 1998, pp. 327–399.

[226] J. W. ROBERTS, *Mechanisms of Bacterial Transcription Termination*, Journal of Molecular Biology, 431 (2019), pp. 4030–4039.

[227] D. ROTMAN, *Technology and inequality*, MIT Technology Review, (2014).

[228] L. SAGAN, *On the origin of mitosing cells*, Journal of Theoretical Biology, 14 (1967), pp. 255–274.

[229] N. SAID AND M. C. WAHL, *Transcription complexes as rna chaperones*, Transcription, 12 (2021), pp. 126–155.

[230] A. SAINI, *Want to do better science? admit you're not objective*, Nature, 579 (2020), p. 175.

[231] E. SALMÓN, *Kincentric ecology: indigenous perceptions of the human–nature relationship*, Ecological Applications, 10 (2000), pp. 1327–1332.

[232] F. SANGER, *Sequences, sequences, and sequences*, Annual review of biochemistry, 57 (1988), pp. 1–29.

[233] J. SANTOS-MORENO, E. TASIUDI, J. STELLING, AND Y. SCHAERLI, *Multistable and dynamic CRISPRi-based synthetic circuits*, Nature Communications, 11 (2020), pp. 1–8.

[234] S. W. SCHAFFTER AND R. SCHULMAN, *Building in vitro transcriptional regulatory networks by successively integrating multiple functional circuit modules*, Nature Chemistry, 11 (2019), pp. 829–838.

[235] F. SCHNEIDER, *The jevons paradox and the myth of resource efficiency improvements*, Journal of Cleaner Production, 18 (2010), pp. 600–602.

[236] M. G. SCHUBERT, D. B. GOODMAN, T. M. WANNIER, D. KAUR, F. FARZADFARD, T. K. LU, S. L. SHIPMAN, AND G. M. . CHURCH, *High-throughput functional variant screens via in vivo production of single-stranded DNA*, Proceedings of the National Academy of Sciences, 118 (2021).

[237] A. SCHWARTZ, A. R. RAHMOUNI, AND M. BOUDVILLAIN, *The functional anatomy of an intrinsic transcription terminator*, The EMBO Journal, 22 (2003), pp. 3385–3394.

[238] A. SCHWARTZ, A. R. RAHMOUNI, AND M. BOUDVILLAIN, *The functional anatomy of an intrinsic transcription terminator*, EMBO Journal, 22 (2003), pp. 3385–3394.

[239] M. SCHWARZ-SCHILLING, A. DUPIN, F. CHIZZOLINI, S. KRISHNAN, S. S. MANSY, AND F. C. SIMMEL, *Optimized assembly of a multifunctional rna-protein nanostructure in a cell-free gene expression system*, ACS Nano Letters, 18 (2018), pp. 2650–2657.

[240] F. J. SEDLAZECK, H. LEE, C. A. DARBY, AND M. C. SCHATZ, *Piercing the dark matter: bioinformatics of long-range sequencing and mapping*, Nature Reviews Genetics, 19 (2018), pp. 329–346.

[241] R. SENDER, S. FUCHS, AND R. MILO, *Revised Estimates for the Number of Human and Bacteria Cells in the Body*, PLOS Biology, 14 (2016), p. e1002533.

[242] M. SHAHBAZ, *Does financial instability increase environmental degradation? Fresh evidence from Pakistan*, Economic Modelling, 33 (2013), pp. 537–544.

[243] S. SHAO, L. CHANG, Y. SUN, Y. HOU, X. FAN, AND Y. SUN, *Multiplexed sgRNA Expression Allows Versatile Single Nonrepetitive DNA Labeling and Endogenous Gene Regulation*, ACS Synthetic Biology, 7 (2018), pp. 176–186.

[244] J. SHENDURE, S. BALASUBRAMANIAN, G. M. CHURCH, W. GILBERT, J. ROGERS, J. A. SCHLOSS, AND R. H. WATERSTON, *DNA sequencing at 40: past, present and future*, Nature, 550 (2017).

[245] R. P. SHETTY, D. ENDY, AND T. F. KNIGHT, *Engineering BioBrick vectors from BioBrick parts*, Journal of Biological Engineering, 2 (2008), pp. 1–12.

[246] Y. SHI, *Mechanistic insights into precursor messenger RNA splicing by the spliceosome*, Nature Reviews Molecular and Cell Biology, 18 (2017), pp. 655–670.

[247] A. M. SIDORE, F. LAN, S. W. LIM, AND A. R. ABATE, *Enhanced sequencing coverage with digital droplet multiple displacement amplification*, Nucleic Acids Research, 44 (2016), p. e66.

[248] B. D. SMITH, *Behavior. The ultimate ecosystem engineers*, Science, 315 (2007), pp. 1797–1798.

[249] A. SNIR, D. NADEL, I. GROMAN-YAROSLAVSKI, Y. MELAMED, M. STERNBERG, O. BAR-YOSEF, AND E. WEISS, *The origin of cultivation and proto-weeds, long before neolithic farming*, PLoS One, 10 (2015), p. e0131422.

[250] M. O. SOMMER AND B. SUESS, *(Meta-)genome mining for new ribo-regulators*, Science, 352 (2016), pp. 144–145.

[251] R. STARK, M. GRZELAK, AND J. HADFIELD, *RNA sequencing: the teenage years*, Nature Reviews Genetics, 20 (2019), pp. 631–656.

[252] F. W. STUDIER AND B. A. MOFFATT, *Use of bacteriophage t7 rna polymerase to direct selective high-level expression of cloned genes*, Journal of Molecular Biology, 189 (1986), pp. 113–130.

[253] C. SUTHERLAND AND K. S. MURAKAMI, *An introduction to the structure and function of the catalytic core enzyme of escherichia coli rna polymerase*, EcoSal Plus, 8 (2018).

[254] J. C. TAGGART, J.-B. LALANNE, AND G.-W. LI, *Quantitative control for stoichiometric protein synthesis*, Annual Review of Microbiology, 75 (2021), pp. 243–267.

[255] T. H. TAHIROV, D. TEMIAKOV, M. ANIKIN, V. PATLAN, W. T. MCALLISTER, D. G. VASSYLYEV, AND S. YOKOYAMA, *Structure of a T7 RNA polymerase elongation complex at 2.9 A resolution*, Nature, 420 (2002), pp. 43–50.

[256] T. H. TAHIROV, D. TEMIAKOV, M. ANIKIN, V. PATLAN, W. T. MCALLISTER, D. G. VASSYLYEV, AND S. YOKOYAMA, *Structure of a t7 rna polymerase elongation complex at 2.9 å resolution*, Nature, 420 (2002), pp. 43–50.

[257] M. J. TARNOWSKI AND T. E. GOROCHOWSKI, *Massively parallel characterization of engineered transcript isoforms using direct RNA sequencing*, Nature Communications, 13 (2022), p. 434.

[258] T. A. TATUSOVA AND T. L. MADDEN, *Blast: a new tool for comparing protein and nucleotide sequences*, FEMS Microbiology Letters, 174 (1999), pp. 247–250.

[259] K. TAYLOR AND S. WOODS, *Reflections on the practice of responsible (research and) innovation in synthetic biology*, New Genetics and Society, 39 (2020), pp. 127–147.

[260] J. THOMAS, *Synthetic anti-malarial compound is bad news for artemisia farmers*, The Guardian, (2013).

[261] J. C. THOMPSON, D. K. WRIGHT, S. J. IVORY, J. H. CHOI, S. NIGHTINGALE, A. MACKAY, F. SCHILT, E. OTÁROLA-CASTILLO, J. MERCADER, S. L. FORMAN, T. PIETSCH, A. S. COHEN, J. R. ARROWSMITH, M. WELLING, J. DAVIS, B. SCHIERY, P. KALIBA, O. MALIJANI, M. W. BLOME, C. A. O'DRISCOLL, S. M. MENTZER, C. MILLER, S. HEO, J. CHOI, J. TEMBO, F. MAPEMBA, D. SIMENGWA, AND E. GOMANI-CHINDEBVU, *Early human impacts and ecosystem reorganization in southern-central Africa*, Nature Scientific Advances, 7 (2021).

[262] A. TINAFAR, K. JAENES, AND K. PARDEE, *Synthetic Biology Goes Cell-Free*, BMC Biology, 17 (2019), pp. 1–14.

[263] A. TOYNBEE, *The World and The West*, Oxford University Press, 1953.

[264] M. A. URBINA, A. J. WATTS, AND E. E. REARDON, *Labs should cut plastic waste too*, Nature, 528 (2015), pp. 479–479.

[265] J. A. VALERI, K. M. COLLINS, P. RAMESH, M. A. ALCANTAR, B. A. LEPE, T. K. LU, AND D. M. CAMACHO, *Sequence-to-function deep learning frameworks for engineered riboregulators*, Nature Communications, 11 (2020), p. 5058.

[266] A. H. VAN BRUGGEN, I. M. FRANCIS, AND R. KRAG, *The vicious cycle of lettuce corky root disease: effects of farming system, nitrogen fertilizer and herbicide*, Plant and soil, 388 (2015), pp. 119–132.

[267] A. H. VAN BRUGGEN, E. M. GOSS, A. HAVELAAR, A. D. VAN DIEPENINGEN, M. R. FINCKH, AND J. G. MORRIS JR, *One health-cycling of diverse microbial communities as a connecting force for soil, plant, animal, human and ecosystem health*, Science of the Total Environment, 664 (2019), pp. 927–937.

[268] E. L. VAN DIJK, Y. JASZCZYSZYN, D. NAQUIN, AND C. THERMES, *The Third Revolution in Sequencing Technology*, Trends in Genetics, 34 (2018), pp. 666–681.

[269] G. VARANI, *Exceptionally stable nucleic acid hairpins*, Annual Review of Biophysics and Biomolecular Structure, 24 (1995), pp. 379–404.

[270] R. VASER, I. SOVIĆ, N. NAGARAJAN, AND M. ŠIKIĆ, *Fast and accurate de novo genome assembly from long uncorrected reads*, Genome Research, 27 (2017), pp. 737–746.

[271] D. G. VASSYLYEV, M. N. VASSYLYEVA, A. PEREDERINA, T. H. TAHIROV, AND I. ARTSIMOVITCH, *Structural basis for transcription elongation by bacterial rna polymerase*, Nature, 448 (2007), pp. 157–162.

[272] K. VAVITSAS, *OpenMTA, a paradigm shift in exchanging biological material*, Synthetic Biology, 3 (2018), p. ysy021.

[273] K. VAVITSAS, A. KUGLER, A. SATTA, D. G. HATZINIKOLAOU, P. LINDBLAD, D. P. FEWER, P. LINDBERG, M. TOIVARI, AND K. STENSJÖ, *Doing synthetic biology with photosynthetic microorganisms*, Physiologia Plantarum, 173 (2021), pp. 624–638.

[274] A. VETTER, *The Matrix of Convivial Technology - Assessing technologies for degrowth*, Journal of Cleaner Production, 197 (2018), pp. 1778–1786.

[275] A. VIGOUROUX AND D. BIKARD, *CRISPR tools to control gene expression in bacteria*, Microbiology and Molecular Biology Reviews, 84 (2020), pp. e00077–19.

[276] D. VIPIN, Z. IGNATOVA, AND T. E. GOROCHOWSKI, *Characterizing Genetic Parts and Devices Using RNA Sequencing*, Methods in Molecular Biology, 2229 (2021), pp. 175–187.

[277] C. A. VOIGT, *Genetic parts to program bacteria*, Current Opinion in Biotechnology, 17 (2006), pp. 548–557.

[278] P. H. VON HIPPEL AND T. D. YAGER, *Transcript elongation and termination are competitive kinetic processes.*, Proceedings of the National Academy of Sciences, 88 (1991), pp. 2307–2311.

[279] C. J. VON WINTERSDORFF, J. PENDERS, J. M. VAN NIEKERK, N. D. MILLS, S. MAJUMDER, L. B. VAN ALPHEN, P. H. SAVELKOUL, AND P. F. WOLFFS, *Dissemination of antimicrobial resistance in microbial ecosystems through horizontal gene transfer*, Frontiers in microbiology, (2016), p. 173.

[280] N. WALTER, A. CHAUVIER, J. PORTA, I. DEB, E. ELLINGER, A. FRANK, AND M. OHI, *Structural basis for control of bacterial RNA polymerase pausing by a riboswitch and its ligand*, Researchsquare Preprint, (2021).

[281] W. WANG, Y. LI, Y. WANG, C. SHI, C. LI, Q. LI, AND R. J. LINHARDT, *Bacteriophage t7 transcription system: an enabling tool in synthetic biology*, Biotechnology Advances, 36 (2018), pp. 2129–2137.

[282] Y. WANG, Y. ZHAO, A. BOLLAS, AND K.-F. AU, *Nanopore sequencing technology, bioinformatics and applications*, Nature Biotechnology, 39 (2021), pp. 1348–1365.

[283] M. W. WEBSTER AND A. WEIXLBAUMER, *Macromolecular assemblies supporting transcription-translation coupling*, Transcription, 12 (2021), pp. 103–125.

[284] M. W. WEBSTER AND A. WEIXLBAUMER, *The intricate relationship between transcription and translation*, Proceedings of the National Academy of Sciences, 118 (2021).

[285] F. WERNER AND D. GROHMANN, *Evolution of multisubunit rna polymerases in the three domains of life*, Nature Reviews Microbiology, 9 (2011), pp. 85–98.

[286] J. WEST AND S. GALLAGHER, *Challenges of open innovation: the paradox of firm investment in open-source software*, R&d Management, 36 (2006), pp. 319–331.

[287] R. R. WICK, L. M. JUDD, AND K. E. HOLT, *Performance of neural network basecalling tools for Oxford Nanopore sequencing*, Genome Biology, 20 (2019), p. 129.

[288] R. R. WICK, L. M. JUDD, AND K. E. D. HOLT, *Demultiplexing barcoded oxford nanopore reads with deep convolutional neural networks*, PLoS Computational Biology, 14 (2018), p. e1006583.

[289] A. E. WILLIAMSON, P. M. YLIOJA, M. N. ROBERTSON, Y. ANTONOVA-KOCH, V. AVERY, J. B. BAELL, H. BATCHU, S. BATRA, J. N. BURROWS, S. BHATTACHARYYA, F. CALDERON, S. A. CHARMAN, J. CLARK, B. CRESPO, M. DEAN, S. L. DEBBERT, M. DELVES, A. S. DENNIS, F. DEROOSE, S. DUFFY, S. FLETCHER, G. GIAEVER, I. HALLYBURTON, F. J. GAMO, M. GEBBIA, R. K. GUY, Z. HUNGERFORD, K. KIRK, M. J. LAFUENTE-MONASTERIO, A. LEE, S. MEISTER, C. NISLOW, J. P. OVERINGTON, G. PAPADATOS, L. PATINY, J. PHAM, S. A. RALPH, A. RUECKER, E. RYAN, C. SOUTHAN, K. SRIVASTAVA, C. SWAIN, M. J. TARNOWSKI, P. THOMSON, P. TURNER, I. M. WALLACE, T. N. WELLS, K. WHITE, L. WHITE, P. WILLIS, E. A. WINZELER, S. WITTLIN, AND M. H. TODD, *Open Source Drug Discovery: Highly Potent Antimalarial Compounds Derived from the Tres Cantos Arylpyrroles*, ACS Central Science, 2 (2016), pp. 687–701.

[290] L. WINNER, *Do artifacts have politics?*, Daedalus, (1980), pp. 121–136.

[291] A. XAYAPHOUMMINE, T. BUCHER, AND H. ISAMBERT, *Kinefold web server for RNA/DNA folding path and structure prediction including pseudoknots and knots*, Nucleic Acids Research, 33 (2005), pp. W605–10.

[292] S. S. YIM, N. I. JOHNS, J. PARK, A. L. GOMES, R. M. MCBEE, M. RICHARDSON, C. RONDA, S. P. CHEN, D. GARENNE, V. NOIREAUX, AND H. H. WANG, *Multiplex transcriptional characterizations across diverse bacterial species using cell-free systems*, Molecular Systems Biology, 15 (2019), p. e8875.

[293] H. YIN, I. ARTSIMOVITCH, R. LANDICK, AND J. GELLES, *Nonequilibrium mechanism of transcription termination from observations of single rna polymerase molecules*, Proceedings of the National Academy of Sciences, 96 (1999), pp. 13124–13129.

[294] Y. W. YIN AND T. A. STEITZ, *Structural basis for the transition from initiation to elongation transcription in t7 rna polymerase*, Science, 298 (2002), pp. 1387–1395.

[295] J. YOO AND C. KANG, *Variation of in vivo efficiency of the bacteriophage T7 terminator depending on terminator-upstream sequences*, Molecules and Cells, 6 (1996), pp. 352–358.

[296] J. YOO AND C. KANG, *Variation of in vivo efficiency of the bacteriophage t7 terminator depending on terminator-upstream sequences*, Molecules and Cells, 6 (1996), pp. 352–358.

[297] A. M. YU, P. M. GASPER, L. CHENG, L. B. LAI, S. KAUR, V. GOPALAN, A. A. CHEN, AND J. B. LUCKS, *Computationally reconstructing cotranscriptional RNA folding from experimental data reveals rearrangement of non-native folding intermediates*, Molecules and Cells, 81 (2021), pp. 870–883.

[298] L. S. ZARAMELA, O. MOYNE, M. KUMAR, C. ZUNIGA, J. D. TIBOCHA-BONILLA, AND K. ZENGLER, *The sum is greater than the parts: exploiting microbial communities to achieve complex functions*, Current Opinion in Biotechnology, 67 (2021), pp. 149–157.

[299] J. ZHANG AND R. LANDICK, *A two-way street: regulatory interplay between RNA polymerase and nascent RNA structure*, Trends in Biochemical Sciences, 41 (2016), pp. 293–310.

[300] C. ZHU, X. GUO, P. DUMAS, M. TAKACS, M. ABDELKAREEM, A. VANDEN BROECK, C. SAINT-ANDRÉ, G. PAPAI, C. CRUCIFIX, J. ORTIZ, ET AL., *Transcription factors modulate rna polymerase conformational equilibrium*, Nature Communications, 13 (2022), pp. 1–12.

[301] J. ZOU, M. HUSS, A. ABID, P. MOHAMMADI, A. TORKAMANI, AND A. TELENTI, *A primer on deep learning in genomics*, Nature Genetics, (2018), p. 1.

[302] M. ZUKER AND P. STIEGLER, *Optimal computer folding of large RNA sequences using thermodynamics and auxiliary information*, Nucleic Acids Research, 9 (1981), pp. 133–148.

[303] J. Šponer, G. Bussi, M. Krepl, P. Banáš, S. Bottaro, R. A. Cunha, A. Gil-Ley, G. Pinamonti, S. Poblete, P. Jurečka, N. G. Walter, and M. Otyepka, *RNA Structural Dynamics As Captured by Molecular Simulations: A Comprehensive Overview*, Chemical Reviews, 118 (2018), pp. 4177–4338.