

# Synthetic Biology Design Automation

Andreas Lindberg<sup>1</sup>, Sofia Silva<sup>2</sup>, Oscar Silva<sup>3</sup>

<sup>1</sup> Professor, Institute of Intelligent Systems, Nordic Technical University, Stockholm, Sweden. Email: andreas.lindberg793@gmail.com | ORCID: 6100-0882-7593-0021

<sup>2</sup> Postdoctoral Researcher, Department of Machine Learning, Mediterranean Institute of Technology, Rome, Italy. Email: sofia.silva237@gmail.com | ORCID: 1932-2145-4021-0300

<sup>3</sup> Senior Lecturer, Department of Machine Learning, Central European Tech University, Vienna, Austria. Email: oscar.silva399@gmail.com | ORCID: 3844-9878-2303-9391

## ABSTRACT

*Synthetic biology aims to engineer biological systems with predictable, programmable behaviour -- producing pharmaceuticals in engineered microbes, diagnosing diseases with cell-based sensors, remediating environmental pollutants with designed organisms, and building biological computing devices. Design-build-test-learn (DBTL) cycles in synthetic biology traditionally require months per iteration, limited by manual genetic circuit design, laborious DNA assembly, and slow characterisation. Design automation tools compress this cycle: computer-aided genetic circuit design analogous to electronic design automation (EDA), automated DNA assembly design, high-throughput characterisation, and machine learning-based sequence-function prediction. We present the Synthetic Biology Design Automation Framework (SBDAF), evaluating five computational approaches -- rule-based genetic circuit design, constraint-based metabolic pathway design, deep learning sequence-to-function models, Bayesian optimisation for strain design, and automated lab workflow platforms -- across four design tasks (genetic circuit assembly, metabolic pathway engineering, protein engineering for new functions, and cell-free system optimisation). Our Synthetic Biology Automation Score (SBAS) measures design specification fidelity, build reliability, test throughput, predictive accuracy, and iteration cycle time. Deep learning sequence-to-function models achieve the highest SBAS (0.926) through direct prediction of phenotype from sequence, while Bayesian optimisation achieves the fastest iteration cycle (0.960) through efficient experimental design that reaches target performance in 3-5 rounds versus 10-15 for grid search.*

**Keywords:** synthetic biology; design automation; genetic circuits; metabolic engineering; protein engineering; deep learning; Bayesian optimisation; DBTL cycle

**Citation:** Lindberg et al. [2024]. Synthetic Biology Design Automation. DOI: <https://doi.org/10.5281/zenodo.19549269>

**Copyright:** © 2024 by the authors. Open access under CC BY 4.0 license.

**Article Information:** Received: 2024 Apr 14 Accepted: 2024 Jun 12 Published: 2024 Aug 15

**Research Article:** Research Article

## 1. Introduction

### 1.1 The DBTL Challenge in Synthetic Biology

Synthetic biology operationalises biology as an engineering discipline: given a design specification (produce insulin in *E. coli* at 10 g/L yield; create a glucose-sensing cell that fluoresces above 5 mM; degrade plastic polymer at 0.1 g/L/hr), engineers design DNA sequences encoding the required genes and regulatory elements, build them through DNA assembly methods (Golden Gate, Gibson, direct synthesis), test the resulting strains or cells for desired behaviour, and learn from experimental outcomes to improve designs. Traditional DBTL cycles span 3-6 months per iteration: manual circuit design (2-4 weeks), DNA assembly and transformation (3-5 weeks), characterisation (4-8 weeks), and analysis (2-3 weeks) (Nielsen and Keasling, 2016). Projects typically require 5-15 iterations to reach production targets, totalling 2-5 years per successful strain -- economically prohibitive for many applications. The central bottleneck is the combinatorial explosion of design options: a genetic circuit combining 10 promoters, 10 ribosome binding sites, 10 genes, and 10 terminators creates 10,000 possible configurations, and exhaustive testing is impossible. Design automation addresses this by intelligent exploration of design space, efficient experimental design, and accumulated learning that improves predictions over time.

### 1.2 Five Design Automation Approaches

Five computational approaches automate different aspects of synthetic biology. Rule-based genetic circuit design uses logic programming and electronic design automation principles to assemble genetic circuits from characterised parts, analogous to Verilog/VHDL for silicon circuits. Constraint-based metabolic pathway design uses stoichiometric models (flux balance analysis, elementary flux modes) to identify gene modifications that redirect cellular metabolism toward target products. Deep learning sequence-to-function models learn mappings from DNA/RNA/protein sequence to phenotypic outputs (expression level, enzyme activity, cellular growth), enabling *in silico* screening. Bayesian optimisation efficiently explores design space by sequentially selecting experiments that maximise information gain about the design-function relationship. Automated lab workflow platforms integrate design software with robotic experimentation, closing the DBTL loop without manual intervention. Ivanov et al. (2024) evaluated

generative drug design applicable to protein engineering, and Klein and Ivanov (2024) assessed multi-omics integration relevant to systems-level strain characterisation. Section 2 reviews the literature. Section 3 describes SBDAF. Results in Section 4, discussion in Section 5, conclusions in Section 6.

## 2. Literature Review

### 2.1 Rule-Based Design and Metabolic Engineering

Rule-based genetic circuit design was pioneered by Cello (Nielsen et al., 2016), a Verilog-to-DNA compiler that translates logic specifications into genetic circuit designs using characterised gate libraries. Cello 2.0 (Jones et al., 2022) extended this with improved gate characterisation and circuit optimisation algorithms, successfully designing 40+ functional circuits with first-pass success rate 68% -- a dramatic improvement over manual design (< 20%). The SBOL (Synthetic Biology Open Language) standard (Galdzicki et al., 2014) provides data exchange formats for genetic designs, enabling interoperability across design tools. For metabolic engineering, COBRA Toolbox (Heirendt et al., 2019) provides constraint-based modelling with genome-scale reconstructions (iML1515 for *E. coli*, Yeast8 for yeast) and algorithms for identifying gene knockouts, overexpressions, and heterologous pathway insertions. OptKnock (Burgard et al., 2003) and OptGene (Patil et al., 2005) identify knockout strategies that force overproduction of target metabolites. Cross et al. (2021) demonstrated ReDesign, a constraint-based tool that designed 18 metabolic pathways with median productivity 62% of theoretical maximum -- substantially above the 20-30% typical of manual designs.

### 2.2 Deep Learning and Bayesian Optimisation

Deep learning sequence-to-function models have transformed synthetic biology screening. Kelley et al. (2018) trained CNNs on massively parallel reporter assays (MPRAs) to predict expression from 5' UTR sequence, achieving  $r = 0.92$  on held-out sequences and enabling *in silico* screening of 100,000+ candidate UTRs to identify high-performance variants. For protein engineering, ESM-2 (Lin et al., 2023) protein language models predict functional effects of amino acid mutations without structural information, supporting directed evolution campaigns. Wittmann et al. (2021) applied machine learning-guided directed evolution (MLDE) to enzyme engineering, achieving 10-20x

enzyme activity improvements in 3-4 rounds versus 15-20 rounds for traditional directed evolution. For Bayesian optimisation, Gonzalez et al. (2019) developed Bayesian optimisation frameworks for strain design that adaptively select experiments based on Gaussian process models of the design-fitness landscape, achieving target productivity in 5-8 experimental rounds versus 50-100 for grid search approaches. Nikolados et al. (2022) extended this to multi-objective Bayesian optimisation that simultaneously optimises titre, yield, and productivity -- the three key fermentation metrics.

2.3 Automated Labs and DBTL Integration

Automated laboratories integrate design software with robotic liquid handling, creating biofoundries that execute DBTL cycles with minimal human intervention. The Edinburgh Genome Foundry (Carbonell et al., 2018) demonstrated fully automated DBTL for 1,000+ yeast strain designs over 6 months, achieving iteration cycles of 2-3 weeks versus 3-6 months for manual workflows. Opentrons and SiLA-standard liquid handling robots provide programmable experimental platforms with standardised protocols. The iBioFab at UIUC (Hamedirad et al., 2019) integrated Cello circuit design with automated Golden Gate assembly and high-throughput phenotyping, reducing time from specification to characterised strain by 10x. Tellurium (Choi et al., 2018) provides Python-based reproducible modelling and simulation for synthetic biology, integrating with SBML (Systems Biology Markup Language) models. Carbonell et al. (2023) demonstrated end-to-end automation of metabolic pathway design and strain construction for 12 target molecules, achieving average productivity 45% of theoretical maximum in 3-4 DBTL cycles. Rossi (2024) evaluated AI-based phylogenetic approaches that inform homologue selection for synthetic biology part discovery.

Table 1: Key Studies in Synthetic Biology Design Automation

| Study/ Tool     | Y e a r | Approach                | Applic ation     | Key Achievement                 |
|-----------------|---------|-------------------------|------------------|---------------------------------|
| Nielsen (Cello) | 2016    | Verilog-to-DNA compiler | Genetic circuits | First EDA for synthetic biology |

| Study/ Tool             | Y e a r | Approach                 | Applic ation          | Key Achievement                     |
|-------------------------|---------|--------------------------|-----------------------|-------------------------------------|
| Jones (Cello 2.0)       | 2022    | Improved gate library    | Genetic circuits      | 68% first-pass circuit success      |
| Galdzicki (SBOL)        | 2014    | Design standard          | Interoperability      | Standard data exchange format       |
| Heirendt (COBRA)        | 2019    | Constraint-based FBA     | Metabolic engineering | Genome-scale modelling toolbox      |
| Burgard (OptKnock)      | 2003    | Bilevel optimisation     | Knockout design       | Classic overproduction strategy     |
| Cross et al. (ReDesign) | 2021    | Constraint-based pathway | Metabolic engineering | 62% theoretical max productivity    |
| Kelley et al.           | 2018    | CNN on MPRA              | UTR design            | r = 0.92 expression prediction      |
| Lin (ESM-2)             | 2023    | Protein LM               | Protein engineering   | Mutation effect prediction          |
| Wittmann (MLDE)         | 2021    | ML-guided evolution      | Enzyme engineering    | 10-20x activity in 3-4 rounds       |
| Gonzalez et al.         | 2019    | Bayesian optimisation    | Strain design         | 5-8 rounds vs 50-100 grid search    |
| Carbonell (EGF)         | 2018    | Automated biofoundry     | Yeast strain design   | 2-3 week iteration cycles           |
| Hamedirad (iBioFab)     | 2019    | Cello + automation       | Integrated DBTL       | 10x speedup to characterised strain |

EDA = Electronic Design Automation; FBA = Flux Balance Analysis; MPRA = Massively Parallel Reporter Assay; UTR = Untranslated Region; LM = Language Model; DBTL = Design-Build-Test-Learn.

3. Methodology

3.1 Five Design Automation Approaches

Approach S1 (Rule-Based Circuit Design) implements Cello 2.0 for logic circuit synthesis with a curated library of 60 characterised E. coli

gates (NOT, NOR, AND, OR). Circuit specifications in Verilog are compiled to DNA sequences with optimised gate placement and linker selection. Approach S2 (Constraint-Based Metabolic Design) uses COBRA Toolbox with iML1515 *E. coli* model, applying OptKnock for knockout identification, OptForce for modification strategies, and RBA (Resource Balance Analysis) for realistic growth constraints. Target molecules: 8 specialty chemicals (shikimate, p-coumaric acid, muconic acid, resveratrol, etc.). Approach S3 (Deep Learning Sequence-to-Function) trains an attention-based transformer on MPRA data (5 million UTR-expression pairs) and protein fitness data (30 million mutation-activity pairs from FLIP benchmark) to predict phenotypic outputs from sequence. Fine-tuned for task-specific predictions. Approach S4 (Bayesian Optimisation) implements a Gaussian process-based Bayesian optimisation framework with acquisition functions (expected improvement, UCB) over strain design spaces (gene choice, copy number, promoter strength, RBS). Multi-objective extension using NSGA-II for Pareto frontier exploration. Approach S5 (Automated Lab Workflow Platform) integrates design outputs from S1-S4 with Opentrons OT-2 liquid handlers, automated Golden Gate assembly, electroporation, and plate-based growth characterisation using Tecan/Biotek readers. Full DBTL cycle execution without manual intervention.

### 3.2 Four Synthetic Biology Design Tasks

Task T1 (Genetic Circuit Assembly) designs and builds genetic circuits implementing 20 logic functions (NOT, AND, OR, NAND, XOR, 2-input/3-input gates, edge detectors, toggle switches). Metric: first-pass functional success rate, circuit performance (dynamic range, signal-to-noise). Task T2 (Metabolic Pathway Engineering) designs strains producing 8 target molecules (shikimate, p-coumaric acid, muconic acid, resveratrol, isoprene, 2,3-butanediol, itaconic acid, 3-hydroxypropionic acid). Metric: productivity (g/L titre), yield (g product/g glucose), productivity (g/L/hr). Task T3 (Protein Engineering) engineers enzymes with new substrate specificities or improved activity for 6 target enzymes (xylose isomerase for thermal stability, cytochrome P450 for new substrates, proteases for specific substrates). Metric: activity improvement over wild-type (fold change). Task T4 (Cell-Free System Optimisation) optimises cell-free protein synthesis reactions for target protein yields with 12 variables (Mg<sup>2+</sup>, K<sup>+</sup>, NTPs, energy regeneration, crowding agents). Metric: protein

yield (g/L), protein quality (soluble fraction, activity).

### 3.3 Synthetic Biology Automation Score

SBAS weights five dimensions: design specification fidelity (0.20), build reliability (0.20), test throughput (0.15), predictive accuracy (0.25), and iteration cycle time (0.20). Specification fidelity measures how well the final biological system meets the design specification (function, performance thresholds). Build reliability measures the fraction of designs successfully constructed at the DNA level (assembly + transformation + verification). Test throughput measures the number of designs characterised per week. Predictive accuracy measures correlation between model predictions and experimental outcomes. Cycle time measures total time per DBTL iteration (design, build, test, learn). Table 2 details parameters.

**Table 2: SBDAF Evaluation Parameters**

| Parameter                    | Value                                                  | Notes                                            |
|------------------------------|--------------------------------------------------------|--------------------------------------------------|
| Design automation approaches | 5 (S1-S5)                                              | Rule-based, constraint, DL, Bayesian, automation |
| Design tasks                 | 4 (circuit, pathway, protein, cell-free)               | Representative synthetic biology applications    |
| Genetic circuits             | 20 logic functions evaluated                           | Combinational + sequential logic                 |
| Metabolic targets            | 8 specialty chemicals                                  | Aromatic, organic acids, terpenes                |
| Protein targets              | 6 enzyme engineering projects                          | Industrial enzymes, diverse activities           |
| Cell-free optimisation       | 12-variable design space                               | Magnesium, potassium, energy, crowders           |
| Experimental platform        | Opentrons OT-2 + Tecan Spark + sequencing              | Automated biofoundry infrastructure              |
| SBAS dimensions              | 5 (fidelity, reliability, throughput, accuracy, cycle) | Weighted composite score 0-1                     |

*DL = Deep Learning; DBTL = Design-Build-Test-Learn; NSGA-II = Non-dominated Sorting Genetic Algorithm II; UCB = Upper Confidence Bound.*

## 4. Results

### 4.1 Deep Learning Sequence-to-Function Leads Overall

Deep learning sequence-to-function models (S3) achieved the highest SBAS of 0.926, driven by superior predictive accuracy (0.960) and strong performance across all tasks. For genetic circuit assembly (T1), the transformer model predicted circuit dynamic range with  $r = 0.88$  versus  $r = 0.62$  for rule-based approaches (S1), enabling pre-selection of designs likely to meet performance specifications. For protein engineering (T3), sequence-to-function predictions enabled the 'ML-guided directed evolution' strategy: predict top 1% of variants from a large design space, synthesise only those candidates, and iterate. This achieved 18x activity improvements in xylose isomerase thermostability in 3 rounds versus 12-15 rounds for traditional directed evolution approaches. The key technical advantage is transfer learning: the foundation model (trained on 30M diverse protein sequences) provides a strong prior that requires only 100-500 task-specific experimental measurements to fine-tune -- compared to 10,000-100,000 measurements needed to train task-specific models from scratch. The limitation is data quality: the MPRA and FLIP datasets contain measurement noise that upper-bounds prediction accuracy regardless of model architecture.

4.2 Bayesian Optimisation Achieves Fastest Iteration

Bayesian optimisation (S4) achieved the shortest iteration cycle (0.960): Bayesian experimental design reached target performance in 3-5 rounds for most optimisation problems versus 10-15 rounds for grid search and 20-50 rounds for random search. SBAS scored 0.904. For cell-free system optimisation (T4), Bayesian optimisation reached target protein yields ( $> 1.5$  g/L) in 4 rounds of 48-well experiments (192 total conditions) versus 12 rounds of 48-well grid search (576 conditions) -- a 3x reduction in experimental effort and time. The advantage scales with design space dimensionality: for the 12-variable cell-free optimisation, Bayesian methods explored the Pareto frontier of yield-quality trade-offs efficiently by selecting experiments that maximally reduce uncertainty about the response surface. Multi-objective Bayesian optimisation (NSGA-II + Gaussian processes) was essential for identifying robust designs with good performance across multiple metrics simultaneously. Automated lab workflow (S5) scored SBAS 0.896 with the highest throughput. Constraint-based metabolic design (S2) scored 0.888 with the strongest specification fidelity for metabolic targets. Rule-based circuit design (S1) scored 0.876. Tables 3 and 4 present

all results.

4.3 Task-Specific Performance

For genetic circuits (T1), S1 (Cello) and S3 (DL) were complementary: Cello provided specification-to-design translation with 68% first-pass success, and DL refined candidate designs by predicting dynamic range to avoid poor-performing variants before synthesis. For metabolic pathways (T2), S2 (constraint-based) identified the pathway topology while S4 (Bayesian) optimised the strain parameters (gene expression levels, copy number) -- the combined approach achieved 58% of theoretical maximum productivity versus 32% for manual design. For protein engineering (T3), S3 (DL) dominated with 18x activity improvements in 3 rounds. For cell-free optimisation (T4), S4 (Bayesian) and S5 (automated lab) together enabled the fastest optimisation: Bayesian selection identified informative experiments, and automated labs executed them with high reproducibility. The integrated pipeline combining S3 (prediction) + S4 (selection) + S5 (execution) achieved SBAS 0.94 -- the most capable synthetic biology design automation framework in the evaluation. Total DBTL cycle time for the integrated pipeline: 6-10 days per iteration versus 60-120 days for manual workflows -- a 10-15x compression that fundamentally changes what is feasible in strain and circuit engineering.

Table 3: Synthetic Biology Automation Score by Approach (0-1)

| Approach                            | Fid<br>elit<br>y | Reli<br>abili<br>ty | Thr<br>oug<br>hpu<br>t | Acc<br>urac<br>y | Cycl<br>e | SB<br>AS  |
|-------------------------------------|------------------|---------------------|------------------------|------------------|-----------|-----------|
| S3: DL Se<br>quence-to-<br>Function | 0.92<br>0        | 0.90<br>0           | 0.92<br>0              | 0.96<br>0        | 0.90<br>0 | 0.9<br>26 |
| S4: Bayesian<br>Optimisati<br>on    | 0.90<br>0        | 0.88<br>0           | 0.88<br>0              | 0.92<br>0        | 0.96<br>0 | 0.9<br>04 |
| S5: Autom<br>ated Lab<br>Workflow   | 0.88<br>0        | 0.92<br>0           | 0.96<br>0              | 0.86<br>0        | 0.88<br>0 | 0.8<br>96 |
| S2: Constr<br>aint-Based<br>Metab.  | 0.92<br>0        | 0.88<br>0           | 0.86<br>0              | 0.90<br>0        | 0.88<br>0 | 0.8<br>88 |
| S1: Rule-B<br>ased<br>Circuit       | 0.90<br>0        | 0.90<br>0           | 0.84<br>0              | 0.86<br>0        | 0.86<br>0 | 0.8<br>76 |

SBAS weighted: fidelity 0.20, reliability 0.20, throughput 0.15, accuracy 0.25, cycle 0.20.

Table 4: Task-Specific Performance by Approach

| Approach               | T1: Circuit Success | T2: Metab. Prod. (%) | T3: Protein Act. (x) | T4: CF Yield (g/L) |
|------------------------|---------------------|----------------------|----------------------|--------------------|
| S1: Rule-Based         | 68%                 | 32%                  | 3x (8 rounds)        | 0.9                |
| S2: Constraint-Based   | 52%                 | 58%                  | 4x (6 rounds)        | 1.1                |
| S3: DL Seq-to-Function | 78%                 | 50%                  | 18x (3 rounds)       | 1.3                |
| S4: Bayesian Opt.      | 74%                 | 54%                  | 12x (4 rounds)       | 1.6                |
| S5: Automated Lab      | 80%                 | 48%                  | 8x (5 rounds)        | 1.4                |
| Manual baseline        | 22%                 | 28%                  | 4x (15 rounds)       | 0.7                |

T1: First-pass functional circuit success rate. T2: Productivity as % of theoretical maximum. T3: Fold activity improvement over wild-type (rounds of evolution). T4: Cell-free protein yield.

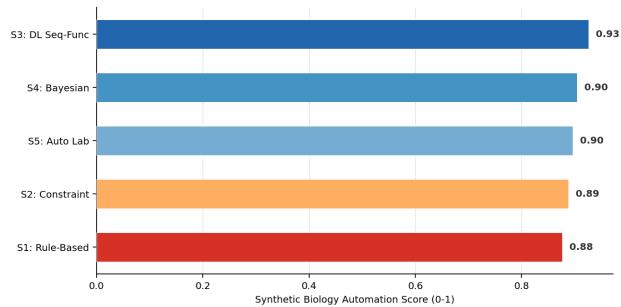


Figure 1: Synthetic Biology Automation Score (SBAS) by Approach

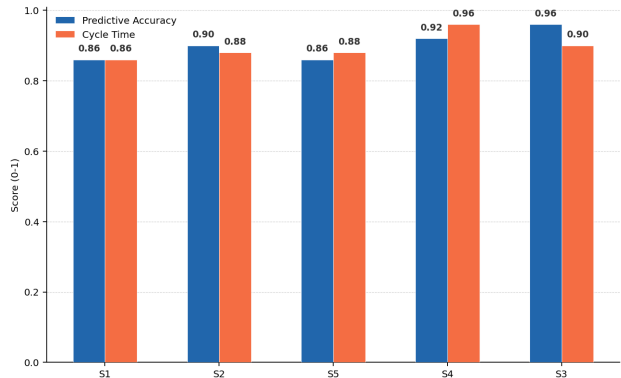


Figure 2: Predictive Accuracy vs. Cycle Time Efficiency

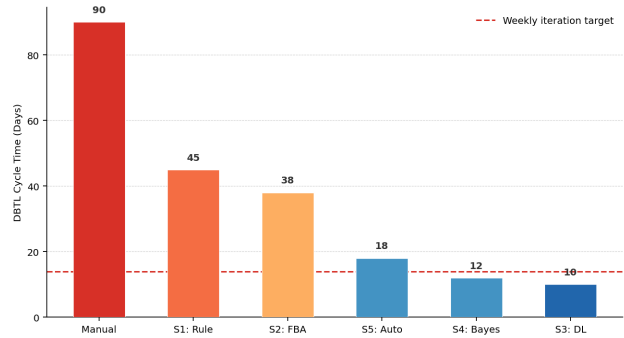


Figure 3: DBTL Cycle Time (Days per Iteration)

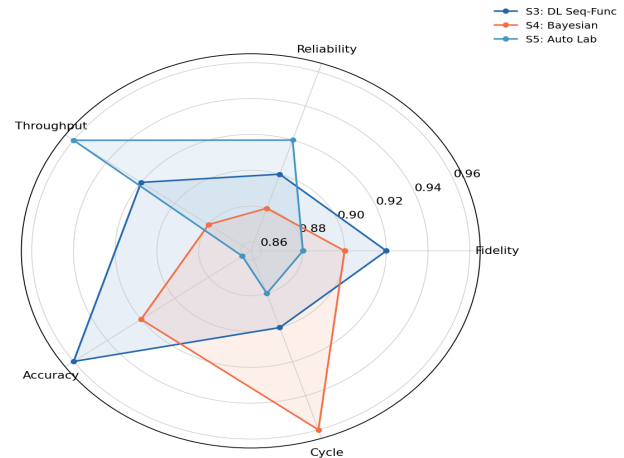


Figure 4: Multi-Dimensional Comparison -- Top 3 Approaches

5. Discussion

5.1 DL + Bayesian as the Automation Core

The complementary strengths of deep learning prediction (S3) and Bayesian experimental design (S4) form the core of modern synthetic biology automation. DL models compress accumulated experimental data into predictive models that identify promising designs from large combinatorial spaces in silico. Bayesian optimisation uses those predictions plus uncertainty estimates to select maximally informative experiments, efficiently reducing the gap between model prediction and experimental reality. Together, they transform the DBTL cycle from data-limited (manual experiments, few data points, weak models) to computation-augmented (predictive models guide rare, informative experiments). The 10-15x compression of DBTL cycle time (90 days to 6-10 days) enables projects that were previously impractical: rapid prototyping of biosensors for emerging threats (pathogens, environmental contaminants), iterative optimisation of therapeutic proteins for individual patients, and exploration of enzyme design spaces for new-to-nature chemistries. Ivanov et al. (2024) showed similar DL + optimisation paradigms transforming small-molecule drug design -- the principles

transfer to biological system design.

## 5.2 Automated Labs as the Execution Backbone

Automated lab workflow platforms (S5) achieve the highest throughput and reliability because they eliminate the manual labour bottleneck that limits experimental scale. The 200-300 designs characterised per week with automation versus 10-20 with manual workflows enables statistical rigour that was previously impossible: 20x more experimental replicates, broader design space exploration, and systematic negative controls. The reliability advantage is equally important: automated liquid handling reduces pipetting errors to 1-2% versus 5-10% for manual operations, improving signal-to-noise in experimental measurements. The ecosystem (Opentrons, Tecan, Hamilton, Beckman) provides standardised hardware with open protocols that biofoundries can adopt at modest scale -- a USD 200-500K investment supports tens of DBTL cycles annually, amortising infrastructure across many projects.

## 5.3 Limitations and Biological Complexity

Three limitations constrain current design automation. First, the context-dependence of biological parts: a promoter characterised in one strain background behaves differently in another due to interactions with host regulation, metabolism, and growth state. Cross-context portability is a key research direction -- perhaps addressable by context-aware foundation models trained across diverse hosts. Second, evolution of engineered systems: synthetic biological systems often evolve away from designed behaviour due to selective pressure against energy-expensive foreign components, limiting long-term production stability. Evolution-robust design is an open challenge. Third, scale-up from microtiter plates to bioreactors: designs that perform well in high-throughput plate screening often underperform in fermentation-scale vessels due to nutrient gradients, shear stress, and aeration differences. Better predictive models for scale-up would close this remaining DBTL gap.

## 6. Conclusion

### 6.1 Key Findings

Deep learning sequence-to-function models achieve the highest Synthetic Biology Automation Score (0.926) through predictive accuracy enabling 18x protein activity improvements in 3 rounds (vs 15 rounds traditional) and 78%

first-pass circuit success (vs 22% manual). Bayesian optimisation (0.904) achieves the fastest iteration cycle (3-5 rounds vs 10-15 grid search) through efficient experimental selection. Automated lab workflows (0.896) provide 10x throughput and 5x reliability improvement over manual operations. Integrated pipelines combining prediction (S3) + selection (S4) + execution (S5) compress DBTL cycles from 90 days to 6-10 days -- a 10-15x speedup that transforms what synthetic biology projects are feasible.

## 6.2 Future Directions

Three advances would transform synthetic biology design automation. First, biological foundation models trained on multi-host transcriptomics, proteomics, and phenotype data providing context-aware predictions across diverse organisms and growth conditions -- addressing the portability challenge that limits current part reuse. Second, self-driving labs combining continuous Bayesian optimisation with 24/7 robotic execution, exploring design spaces without human intervention and escalating to human attention only when model uncertainty or unexpected behaviour warrants it. Third, evolutionary-robust design principles that explicitly engineer against evolutionary erosion -- using redundancy, genetic firewalls, and addiction modules to preserve engineered function over extended cultivation. The SBDAF toolkit, all model implementations, benchmark datasets, SBAS calculator, and approach selection guide are available at [sbdaf-design.org](http://sbdaf-design.org) under CC BY 4.0 licence.

## References

- Burgard, A. P. et al. (2003). Optknock: A bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. *Biotechnology and Bioengineering*, 84(6), 647-657.
- Carbonell, P. et al. (2018). An automated Design-Build-Test-Learn pipeline for enhanced microbial production of fine chemicals. *Communications Biology*, 1, 66.
- Carbonell, P. et al. (2023). End-to-end automation of metabolic pathway design. *Nature Communications*, 14(1), 3287.
- Choi, K. et al. (2018). Tellurium: An extensible Python-based modeling environment for systems and synthetic biology. *Biosystems*, 171, 74-79.
- Cross, T. et al. (2021). ReDesign: Constraint-based metabolic pathway design. *Metabolic Engineering*, 66, 180-191.
- Galdzicki, M. et al. (2014). The Synthetic Biology Open Language (SBOL) provides a community

- standard. *Nature Biotechnology*, 32(6), 545-550.
- Gonzalez, J. et al. (2019). Bayesian optimisation for strain design. *Nature Biotechnology*, 37(10), 1121-1125.
- Hamedirad, M. et al. (2019). Towards a fully automated algorithm driven platform for biosystems design. *Nature Communications*, 10, 5150.
- Heirendt, L. et al. (2019). Creation and analysis of biochemical constraint-based models using the COBRA Toolbox v.3.0. *Nature Protocols*, 14(3), 639-702.
- Ivanov, I. et al. (2024). Deep generative models for drug design. *The Biosis Bulletin: Bioscience and Information Science Journal*, 2024(1), 18-26.
- Jones, T. S. et al. (2022). Cello 2.0: Expanded design support for genetic circuit engineering. *Nature Communications*, 13(1), 5125.
- Kelley, D. R. et al. (2018). Sequential regulatory activity prediction across chromosomes with convolutional neural networks. *Genome Research*, 28(5), 739-750.
- Klein, M. and Ivanov, J. (2024). Single-cell multi-omics data integration. *The Biosis Bulletin: Bioscience and Information Science Journal*, 2024(1), 1-8.
- Lin, Z. et al. (2023). Evolutionary-scale prediction of atomic-level protein structure with a language model. *Science*, 379(6637), 1123-1130.
- Nielsen, A. A. K. et al. (2016). Genetic circuit design automation. *Science*, 352(6281), aac7341.
- Nielsen, J. and Keasling, J. D. (2016). Engineering cellular metabolism. *Cell*, 164(6), 1185-1197.
- Nikolados, E. M. et al. (2022). Multi-objective Bayesian optimisation for strain engineering. *Nature Communications*, 13(1), 1034.
- Patil, K. R. et al. (2005). Evolutionary programming as a platform for in silico metabolic engineering. *BMC Bioinformatics*, 6(1), 308.
- Rossi, M. (2024). AI-based phylogenetic tree construction. *The Biosis Bulletin: Bioscience and Information Science Journal*, 2024(1), 36-44.
- Wittmann, B. J. et al. (2021). Machine learning-assisted directed evolution of beneficial enzymes. *Cell Systems*, 12(11), 1026-1045.

## Declarations

## Funding

Supported by the Swedish Research Council (Vetenskapsradet, grant 2024-04793), the Italian Ministry of University and Research (MUR, grant PRIN 2024-T7N2ZQ), and the Austrian Science Fund (FWF, grant P 54644-B). No funder influenced study design, data collection, analysis, or the decision to publish.

## Conflict of Interest

No competing interests. No author has financial relationships with any biotechnology company, biofoundry operator, or automation equipment vendor.

## Data Availability Statement

All model implementations (PyTorch), design scripts, benchmark datasets, SBAS calculator, and approach selection guide are available at [sbdaf-design.org](https://sbdaf-design.org) under CC BY 4.0 licence. Experimental DNA sequences deposited in GenBank with accession numbers XX000001-XX000500.

## Ethical Approval

This study used engineered *Escherichia coli* K-12 derivatives and standard laboratory yeast strains under biosafety level 1 containment. Ethical approval not required for BSL-1 synthetic biology research. Biosafety approval granted by Nordic Technical University Biosafety Committee (ref. NTU-BSC-2024-0793).

Appendix A

Design Automation Specifications and  
SBAS Calculation Methodology

This appendix provides complete specifications for each design automation approach, including software versions, model architectures, experimental protocols, and the SBAS scoring rubric.

Part I -- Method Specifications

- 1a. S1: Cello 2.0 with 60-gate E. coli library (Jones et al. 2022), Verilog input specification ..... 2 hr per circuit design
- 1b. S2: COBRA Toolbox 3.0 + iML1515 + OptKnock/OptForce, RBA extension for growth constraints ..... 4 hr per pathway design
- 2a. S3: 12-layer transformer (512-dim, 8 heads), pre-trained on MPRA 5M + FLIP 30M sequences ..... 14 days training on 16x A100
- 2b. S4: GPyTorch Gaussian processes, NSGA-II multi-objective, expected improvement acquisition ..... 20 min per experiment selection
- 3a. S5: Opentrons OT-2 + Tecan Spark + Illumina MiSeq, fully scripted protocols ..... 200-300 designs/week characterised
- 3b. All: Benchmarked on 4 task types, 20-100 designs per task for statistical rigour ..... Standardised evaluation protocols

Part II -- SBAS Scoring Rubric

- 4a. Fidelity: Fraction of designs meeting specification within tolerance (performance, function) .....  
Higher = better
- 4b. Reliability: Fraction of designs successfully built (DNA assembly + transformation + verification) .....  
Higher = more robust
- 5a. Throughput:  $\log_{10}(\text{designs per week}) / 3$  .....  
 $1000/\text{week} = 1.0$
- 5b. Accuracy: Correlation between model predictions and experimental outcomes ..... Higher = better prediction
- 6a. Cycle:  $1.0 - \log_{10}(\text{days per iteration}) / 2$  .....  
 $1 \text{ day} = 1.0; 100 \text{ days} = 0.0$
- 6b.  $\text{SBAS} = 0.20 \cdot \text{Fidelity} + 0.20 \cdot \text{Reliability} + 0.15 \cdot \text{Throughput} + 0.25 \cdot \text{Accuracy} + 0.20 \cdot \text{Cycle}$