Thesis for The Degree of Doctor of Philosophy

Knowledge models and inference frameworks for scientific discovery

With applications in systems biology

Alexander H. Gower

Department of Computer Science and Engineering
Chalmers University of Technology | University of Gothenburg
Gothenburg, Sweden, 2026

Knowledge models and inference frameworks for scientific discovery
With applications in systems biology

Alexander H. Gower

© Alexander H. Gower, 2026
except where otherwise stated.
All rights reserved.

ISBN 978-91-8103-439-4
Doktorsavhandlingar vid Chalmers tekniska högskola, Ny serie nr 5896.
ISSN 0346-718X
DOI https://doi.org/10.63959/chalmers.dt/5896

Department of Computer Science and Engineering
Division of Data Science and AI
Chalmers University of Technology | University of Gothenburg
SE-412 96 Göteborg,
Sweden
Phone: +46(0)31 772 1000

Printed by Chalmers Digitaltryck,
Gothenburg, Sweden 2026.

Simply a paring down, a cleaving to
One object, as the star-gazer who sees

One single comet polished by its fall
Rather than countless, untouched galaxies.

— Elizabeth Jennings
The Diamond Cutter

To the memory of Grandad Tim, my first science teacher.

Knowledge models and inference frameworks for scientific discovery

With applications in systems biology

Alexander H. Gower

Department of Computer Science and Engineering
Chalmers University of Technology | University of Gothenburg

Abstract

Scientific discovery is an active process of designing, testing, and improving theories about the natural world. Automating this process is a grand challenge for 21st century science. This thesis examines scientific inquiry as it relates to machine learning, offering contributions to knowledge representations and reasoning frameworks, demonstrated in systems biology.

Systems biology is an integrationist approach to biological science, meaning organisms are treated as complex systems whose behaviour is dictated by the interaction of their constituent parts. Eukaryotic organisms are extremely complex, and research progress in systems biology can be slow. Recent advances in robotics and artificial intelligence (AI) offer great opportunity for automating scientific discovery in this field. Using the model organism *Saccharomyces cerevisiae* (baker's yeast), this thesis explores: the philosophical motivations for automation in biological research; knowledge models and hypotheses in systems biology; and computational models of metabolism.

The first main contribution is a first-order logic framework for modelling cellular physiology, which enables abduction of hypotheses for improvement of knowledge models, using the automated theorem prover (ATP) iProver. The second contribution is an ontology for describing theory changes and hypotheses in a semantic and storage-efficient manner. The third main contribution is an application of graph neural networks (GNNs) to learn knowledge graph embeddings grounded in empirical data and ontology structures. The final contribution is an end-to-end demonstration of autonomous hypothesis generation and experimentation, with hypotheses modelled using ontology terms to support large language model (LLM) agents and human scientists.

These contributions demonstrate the power of knowledge graphs for autonomous scientific discovery. This thesis also argues that scientific discovery is better modelled as supervised learning—specifically active learning for AI scientists—than reinforcement learning; mapping concepts from machine learning algorithms to the domain produces systems that align with established scientific values, leading to improved theories.

Keywords

Artificial intelligence, scientific discovery, machine learning, abduction, automated theorem provers, knowledge modelling, ontologies, systems biology

List of Publications

Appended publications

This thesis is based on the following publications:

[**Paper I**] **A. H. Gower**, K. Korovin, D. Brunnsåker, F. Kronström, G. K. Reder, I. A. Tiukova, R. S. Reiserer, J. P. Wikswo, R. D. King, *The Use Of AI-Robotic Systems For Scientific Discovery*
In press: Computational Approaches to Discovery: AI for Science.

[**Paper II**] F. Kronström, **A. H. Gower**, I. A. Tiukova, R. D. King, *RIMBO – An Ontology for Model Revision Databases*
In International Conference on Discovery Science (pp. 523–534). Cham: Springer Nature Switzerland. (October 2023). https://doi.org/10.1007/978-3-031-45275-8_35

[**Paper III**] **A. H. Gower**, K. Korovin, D. Brunnsåker, E. Y. Bjurström, P. Lasin, I. A. Tiukova, R. D. King, *LGEM$^+$: Automated Improvement of Metabolic Network Models and Model-Driven Experimental Design through Abduction*
To be submitted to PLoS Computational Biology. Extension of **Paper [a]**

[**Paper IV**] F. Kronström, **A. H. Gower**, D. Brunnsåker, I. A. Tiukova, R. D. King, *Graph Neural Network based Hierarchy-Aware Embeddings of Knowledge Graphs: Applications to Yeast Phenotype Prediction*
Under review at Neurosymbolic Artificial Intelligence. https://neurosymbolic-ai-journal.com/system/files/nai-paper-925.pdf

[**Paper V**] E. Y. Bjurström, **A. H. Gower**, P. Lasin, O. I. Savolainen, I. A. Tiukova, R. D. King, *Investigating uncharacterised genes in Saccharomyces cerevisiae using Robot Scientists*
Sci Rep 16, Article 10999 (March 2026). https://doi.org/10.1038/s41598-026-46236-z

[**Paper VI**] D. Brunnsåker, **A. H. Gower**, P. Naval, E. Y. Bjurström, F. Kronström, I. A. Tiukova, R. D. King, *Agentic AI Integrated with Scientific Knowledge: Laboratory Validation in Systems Biology*
Accepted (May 2026) at Journal of the Royal Society Interface.

Other publications

The following publications were published during my PhD studies, or are currently in submission/under revision. However, they are not appended to this thesis, due to contents overlapping that of appended publications or contents not related to the thesis.

[**a**] **A. H. Gower**, K. Korovin, D. Brunnsåker, I. A. Tiukova, R. D. King, *LGEM$^+$: A First-Order Logic Framework for Automated Improvement of Metabolic Network Models Through Abduction*
In International Conference on Discovery Science (pp. 628–643). Cham: Springer Nature Switzerland. (October 2023). https://doi.org/10.1007/978-3-031-45275-8_42

[**b**] D. Brunnsåker, G. K. Reder, N. K. Soni, O. I. Savolainen, **A. H. Gower**, I. A. Tiukova, R. D. King, *High-throughput metabolomics for the design and validation of a diauxic shift model*
npj Systems Biology and Applications, Volume 9, Issue 1, Article number 11 (April 2023). https://doi.org/10.1038/s41540-023-00274-9

[**c**] G. K. Reder, **A. H. Gower**, F. Kronström, R. Halle, V. Mahamuni, A. Patel, H. Hayatnagarkar, L. N. Soldatova, R. D. King, *Genesis-DB: a database for autonomous laboratory systems*
Bioinformatics Advances, Volume 3, Issue 1 (August 2023). https://doi.org/10.1093/bioadv/vbad102

Status of publications

Paper I Received review from Springer LNCS, book title: "Computational Approaches to Discovery: AI for Science", has been returned to publisher and accepted for publication. In press.

Paper II Published October 2023, International Conference on Discovery Science.

Paper III Shorter version published October 2023, International Conference on Discovery Science. Extension received review from Springer Machine Learning, currently being revised and prepared for submission to PLoS Computational Biology.

Paper IV Received review from Neurosymbolic Artificial Intelligence, after revision sent back to editors on 2026-01-26.

Paper V Published March 2026, Nature Scientific Reports.

Paper VI Submitted to Journal of the Royal Society Interface on 2026-01-12. Received peer review, and resubmitted after revision on 2026-04-04. Accepted 2026-05-08.

Statement of Contributions

Paper I Conceived the idea and main points. Conducted the literature research. Wrote the manuscript.

Paper II Co-conceptualised the study. Conducted experiments to test and demonstrate revisions on Yeast8 GEM. Contributed to curation and preparation of the data.

Paper III Co-conceptualised the study. Designed the logical predicate and clause structure. Developed and tested the code to generate logical theory structures from GEMs. Prepared and curated the data. Designed and executed the experiments. Co-conducted the microarray expression data analysis. Designed and prepared the figures, wrote the manuscript.

Paper IV Co-conceptualised the study. Co-designed and co-implemented the semantic and regularisation losses. Co-designed method for training of a model in the absence of a prediction task; wrote and tested code for the demonstration of this method. Designed the algorithm for link evaluation; wrote and tested code for this method. Designed and prepared Figures 7 and 8. Co-wrote the manuscript.

Paper V Designed and wrote the code for the simulation methods. Wrote the automation scripts for the cultivation robot. Contributed to the data analysis.

Paper VI Co-conceptualised the study. Significant contributions to the design of the framework, including the generation and analysis agents and coordination. Co-designed the ontologies and databases. Contributed to the automation scripts for the cultivation robot. Designed and prepared Figure 5. Co-wrote the manuscript.

Acknowledgment

My first thanks are to my supervisor, Professor Ross King, for the opportunity to pursue such interesting research, for the academic direction and freedom, and for the many valuable discussions we have had in the past five years. Thank you to my co-supervisor, Dr Ievgeniia Tiukova, for our discussions and her feedback, encouragement, and support. Thank you to my examiner, Professor David Sands, for his advice during the course of my studies and in the preparation of this thesis.

Thank you to all the co-authors of the papers included in this thesis. I am particularly grateful to Dr Konstantin Korovin for the collaboration we have shared and the informal supervision he has given me. Thanks to Dr Gabriel Reder, Dr Praphapan Lasin (Beera), and Dr Prajakta Naval for our time as colleagues – in particular to Gabe for his guidance and advice on navigating life as a PhD student.

Daniel – you have been with me from my first day to my last in this journey, and I would not have made it without you. Thank you for a rewarding partnership in many endeavours. Erik – thank you for your kindness, insight, and creativity. Filip – amongst many things, thank you for your calmness, passion, and empathy. To all three of you: I'm proud of the work we have done together, grateful for all our discussions and arguments, and happy for the friendship we have.

Lena, thank you for always having my back and for picking me up so many times. Chris, Nico, and Emilio, thank you for your thoughtful friendship, trust, and support.

Thank you to everyone at the division of Data Science and AI for making me feel welcome, and for a lively and supportive environment in which to pursue my research. In particular, thanks to the PhD students and post-docs—for sustaining an enthusiastic and diverse academic community, and for inspiring one another—and to the administrative staff, especially Andrea, Fatima, Clara, and Helny. Thanks to all those not named here, from Chalmers, the WASP Graduate School, Gothenburg, and elsewhere, who have been a part of the past five years. Thank you to the Chalmers Library and its staff for access to many of the resources that informed this thesis. Thank you to the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Alice Wallenberg Foundation for supporting my PhD studies.

To my many excellent teachers—especially Tom Rainbow, Margaret Harding,

Ursel Kiehne, Dominic Vella, and Philippe Trinh—thank you for the energy you spent and faith you placed in me, especially when I least deserved it.

Thanks to my friends—Samuel, Hannah, Rowie, Cameron, Mica, Lauren, Mitch, Charlotte, Stefan, Felix, Henry, Anthony, Luke, Jonathan, and Emil—for your support and for making our world a better place. To Annick, Lennart, and Nyasha for helping me feel welcome in Sweden and your family.

For your courage, our joy, and our strength, thank you to my siblings, Tom and Anna, and to Ella. To Anaëlle, for walking beside me. And to my parents, John and Diana, for your love, ever felt.

Summary

Automation of scientific discovery

- Scientific discovery should be viewed by machine learning researchers as supervised learning, specifically as an active learning problem – active learning shares a natural analogy with the scientific method, and is suitable for the design of robot scientists (closed-loop AI-robotic discovery systems).
- Reinforcement learning is an unsatisfactory paradigm for scientific discovery, for several reasons: definition of the agent, environment, and reward gives rise to philosophical challenges; action–feedback time is not short; and the evaluation of a reward function is not cheap, even if one could be defined.
- For other aspects of robot scientists, such as laboratory automation, reinforcement learning algorithms may be useful.
- The process of designing a robot scientist involves mapping the activities of the scientific method to appropriate agents and coordinating them.
- Logical models and large language models (LLMs) can be combined in an agentic system to form and refine hypotheses, and to design experiments for a robot scientist.
- Several automation workflows were developed, and integrated into both human-driven and fully automated studies.
- Automated experimentation led to discovery of novel biochemical-stressor interactions in *S. cerevisiae*.

Knowledge representations

- Representing hypotheses, experiment design, and empirical data using description logics, enables data to be efficiently reused to assess hypotheses, and construct clear, reproducible scientific knowledge. To enable this, this thesis contributes a novel formulation for describing hypotheses in description logic.

- Knowledge graphs in biology, including those community-curated, can be used as the basis for first-order logic (FOL) theories of metabolism.
- We design an ontology to capture and explain changes in computational biology models. This ontology can be used as a schema for a model revision database, stored as a knowledge graph, with each revision an unambiguous and storage-efficient patch.

Inference methods

- Deductive inference on FOL theories of metabolism using an automated theorem prover (ATP) can: (a) predict growth/no-growth of *S. cerevisiae* strains in a minimal growth medium; and (b) predict expression of metabolic pathways of such strains.
- By extending an ATP to include algorithms for inductive inference through reverse consequence finding, novel hypotheses about the biological system can be abduced from the logical theory.
- We present options for the evaluation of these hypotheses using internal consistency, external consistency, and predictive accuracy.
- Simulation models can be used to instantiate abstract hypotheses about uncharacterised genes in *S. cerevisiae* for comparison with empirical data. This is demonstrated on the previously uncharacterised open reading frame YGR067C, which we propose regulates genes related to ethanol consumption and respiration during the glucose phase of the diauxic shift.
- Incorporating measures of internal consistency (semantic loss) into graph neural network (GNN)-based box embeddings significantly improves predictive power, demonstrated on a mutant strain fitness prediction task for *S. cerevisiae*. Semantic loss also enables the learning of box embeddings in the absence of a prediction task.
- Applying measures of parsimony (regularisation loss) increases the stability of training GNN-based box embeddings on knowledge graphs.
- Box embeddings learned using a GNN, to convey information from knowledge graph neighbours, learn the class hierarchy of the ontologies more effectively than without the GNN.
- Knowledge graph box embedding distance (from one set of embeddings to another) contains information that could be useful for evaluating candidate revisions to the graph.

Contents

Part I

Introductory Chapters

Chapter 1

Introduction

Scientific discovery is an active process of designing, testing, and improving theories about the natural world. To design an AI-robotic system that can automate this process is a grand challenge for 21st-century science.

One of the most important areas of modern science is the study of eukaryote[1] biology. Many advances were made during the 20th century in the understanding of the fundamental components and processes of eukaryotic life. In turn, society has greatly benefited from application of this knowledge to medicine, agriculture, and engineering. However, science remains without an accurate predictive model of the physiology of one organism, or universal theories and laws for the behaviour of eukaryotic systems.

Part of the reason why progress in biology is limited by today's scientific methods is the diversity and complexity of the systems. Hundreds of research hours can be spent in the study of one particular gene, yet the limits of human capability and the economic resource available to the researcher will hamper progression to a complete understanding of the gene and its roles.

Scientific discovery automation therefore has great potential in biology, particularly in the systems biology paradigm: an integrationist approach to biological science, meaning organisms are treated as complex systems whose behaviour is dictated by the interaction of their constituent parts (Kohl et al., 2010). Recognising that the complexity must be embraced to explain their emergent behaviour, biologists have adopted this approach when studying eukaryotes. Systems biology has always relied on mathematical models to handle complexity, yet still research progress can be slow. Recent advances in robotics, and more importantly in artificial intelligence (AI), offer great opportunity for automating scientific discovery in this field.

Using the model organism *Saccharomyces cerevisiae*, baker's yeast, this thesis explores: the philosophical and practical motivations for automation in biological research; knowledge models, experimental data, and hypotheses in systems biology; computational models of metabolism; and design and engineering principles for robot scientists—closed-loop discovery systems.

[1] *eukaryote*—an organism that has a membrane-bound cell nucleus; includes all animals, plants, and yeasts (e.g. *Saccharomyces cerevisiae*).

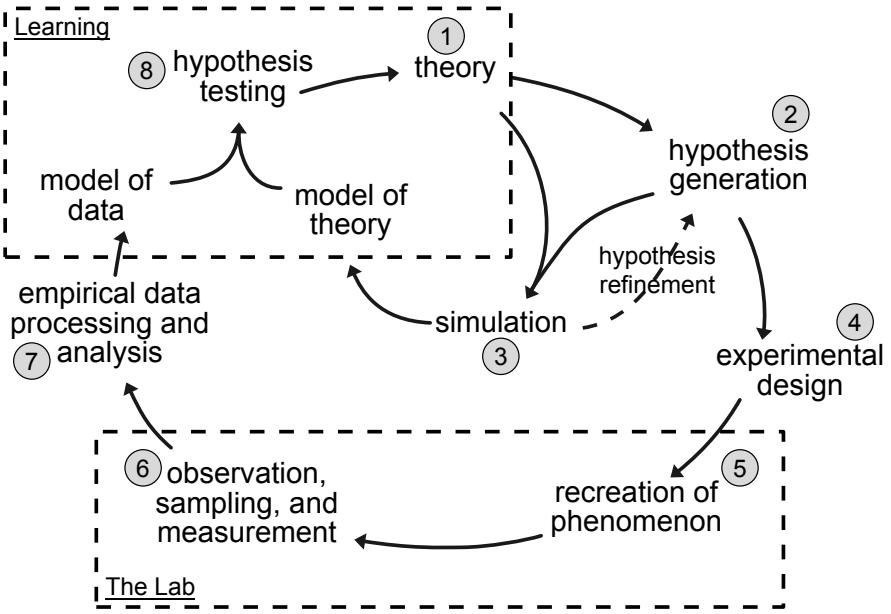


Figure 1.1: A representation of the research cycle in systems biology: (1) starting from an existing theory, (2) hypotheses are generated and refined, possibly using (3) simulations. (4) Experiments are designed to test these hypotheses. (5) These experiments are conducted to reproduce phenomena of interest, and (6) observations, samples, and measurements are taken. (7) These empirical data are processed and analysed to reach a model of the data. A model of theory—the expectation of what will have occurred based on the current theory—is reached through thought or (3) simulation. The expectation and observation are compared using (8) hypothesis tests, which gives information on recommended changes to the (1) theory. The contributions of the papers in this thesis to each part of this research cycle are summarised in Table 1.1.

Step	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Paper II	x							
Paper III	x	x	x					x
Paper IV	x	x	x	–	–	–	–	–
Paper V	x	x	x	–	–	–	–	x
Paper VI	x	x	x	x	x	x	x	x

Table 1.1: Contributions of Papers II–VI to the research cycle steps outlined in Fig. 1.1. 'x': a novel contribution was made to this step; '–': this step was part of the study, but existing methods were used.

The first main contribution is a first-order logic framework for modelling cellular physiology, which enables abduction of hypotheses for improvement of knowledge models, coupling a set of predicates and clauses expressing biochemical reaction processes with an efficient automated theorem prover (ATP), iProver. The second main contribution of this thesis is an ontology for describing theory changes and hypotheses in a semantic and storage-efficient manner.

The third main contribution is an application of graph neural networks (GNNs) to learn knowledge graph embeddings that are grounded both in empirical data and ontology structures. The fourth main contribution is an application of modelling techniques to refine a broad hypothesis about an uncharacterised gene in yeast, leading to better understanding of the gene's function. And the final main contribution is an end-to-end demonstration of autonomous hypothesis generation and experimentation, with particular treatment of the modelling of hypotheses using existing and new ontology terms, so that the knowledge graphs can be exploited and improved by LLM agents and human scientists.

The contributions of this thesis serve to demonstrate the power of knowledge graphs as a basis for autonomous scientific discovery. Their structure can inform algorithmic hypothesis generation, interface with robotic systems for laboratory experiments, enable fast deduction and theory repair, and relate to the metadata of science. We also claim that scientific discovery should be viewed as a supervised learning problem rather than reinforcement learning, and in the case of AI scientists, one of active learning. By mapping concepts from machine learning algorithms to the domain, systems are obtained that align with established scientific values, which leads to improved theories.

This thesis examines the nature of scientific inquiry as it relates to machine learning, offering contributions to knowledge representations and reasoning frameworks, as well as demonstrations of their application in the field of systems biology.

1.1 Structure of the thesis

Chapter 2 introduces the automation of scientific discovery, first by taking a look at science itself, and then moving into more detail of its automation. Chapter 3 introduces the domain of yeast systems biology, and the modelling frameworks used by the community and in this thesis. Chapter 4 provides an introduction to some techniques and tools used in this thesis, particularly those that are not explained in detail in the appended papers.

I recommend that the book chapter 'AI–Robotic Systems for Scientific Discovery' (**Paper I**, Gower et al., 2026) be read alongside the introductory chapters (Chapters 2–4), as it contains further discussion on many of the points raised in the introductory chapters, and serves as an introduction to topics, ideas, and design methodologies presented in the remainder of the thesis.

Chapter 2

The Automation of Scientific Discovery

Scientific discovery is the generation of new knowledge through organised enquiry. The methods used to generate scientific knowledge, and the ways that knowledge is stored and communicated, vary widely across domains. However there are common values and tools that have enabled philosophers to characterise scientific enquiry to a certain degree (Schindler, 2022). These general principles of method are grounded in either general ideas of truth and rationality, specific scientific ideas, or common sense (Gauch, 2003).

Scientific knowledge has an implicit connection to the activities and context from which that knowledge was derived. In other words, it is not possible to separate the knowledge that is gained through scientific enquiry from the presuppositions and tools used to arrive at those conclusions.

So, science can refer to an activity—scientific method—and also information—scientific knowledge. Both method and knowledge are inextricable; science is a cycle of hypothesis, experimentation, and reflection.

2.1 How does science advance?

Consequently, scientific advances are both theoretical and methodological. Advancements in science are often thought of as knowledge, but there are inevitable reliances on and advances in technology, experimentation, and inference as well. In most cases, the knowledge would not be meaningful without grounding in a relevant framework.

Scientific knowledge advances through the formation of hypotheses about a particular domain, which are then tested empirically, either through controlled experimentation or observation.

Scientific method advances through, amongst other mechanisms, the creation and adoption of new technologies, developments in problem-solving techniques, and in the creation of new representations.

Hypotheses vary greatly in impact. Many hypotheses are incremental, offering small changes within an existing model or inference framework. Occasionally, hypotheses change the way scientists look at a particular problem, by introducing a new formulation or representation of the problem, or some insight that is incompatible with models that have come before. And sometimes, entirely new problems are identified—historically these have often come from new technologies that led to observations not possible before. An example being one famous problem in biology from the past few decades: the prediction of protein structure from gene sequence. This problem form became apparent after technology enabled three-dimensional imaging of proteins (late 1950s), and genome sequencing (1970s) (Fersht, 2008).

Kuhn (1970) crystallised this non-monotonic aspect of scientific progress. For Kuhn, scientific advance is characterised by periods of "normal science"—where advances are made through problem-solving—and occasional periods of revolutionary science: paradigm shifts that change the rules and parameters of the problems being solved, leading to a new period of normal science. Kuhn argues that these paradigm shifts are akin to revolutions, in that they are precipitated by an accumulation of anomalies—disagreements between theory and empirical evidence—and ultimately lead to a situation where

> "an existing paradigm has ceased to function adequately in the exploration of an aspect of nature to which that paradigm itself had previously led the way."

So science progresses both incrementally, and occasionally with large, disruptive change.

There is an interrelation between technology, experimentation, inference, and knowledge. New technologies enable the design of new instruments, new knowledge can be used to design new technologies, and so on. In the latter half of the 20th century, information technologies have revolutionised scientific inference, not least in enabling simulation of complex phenomena (Hey & Trefethen, 2020). And in the 21st century, this information technology has continued to rapidly develop, and has led to an increase in automation technologies and big data in science (Hajkowicz et al., 2023; Van Noorden & Perkel, 2023). Over the past few decades, there has been growing evidence that for the first time in human history scientific advance is driven not only by humans, but also by machines (Musslick et al., 2025; OECD, 2023).

2.2 Motivation and history of the automation of science

There are many motivations for the automation of science. Some are extrinsic: science needs automation to increase the pace of discovery, to address problems that are otherwise inaccessible, and to overcome economic constraints. Others are intrinsic: automation of science is a worthy pursuit on its own merit, as it leads to greater understanding of the nature of science itself.

Early attempts to automate scientific discovery were motivated by a desire to understand the nature of human problem solving, and to challenge prevailing philosophical ideas about creativity in science. For a long time, philosophers of science, including Karl Popper, did not believe that it was worth attempting to reproduce scientific thought, in particular the generation of hypotheses, in a logical computer (Giza, 2022). It was in the 1960s, with projects like DENDRAL and MetaDENRAL—systems for the elucidation of chemical structures using mass spectra, and the first computer programs to apply heuristic search to generate scientific hypotheses (Buchanan & Feigenbaum, 1981)—that an engineered system was demonstrated capable of creative problem solving in science.

Work in the decades following these pioneering projects centred around the induction of empirical laws from data, an archetypal example being BACON, a program that finds laws governing a specified set of dependent and independent variables, using empirical data, which rediscovered versions of Kepler's third law and the ideal gas law, amongst others (Langley, 1979; Langley & Zytkow, 1989).

Later work has been aimed at developing closed-loop discovery systems—machines that can make scientific discoveries independently of human interaction. This is the concept of the robot scientist, first introduced by King et al. (2009) with Adam, a robot that discovered new scientific knowledge in yeast functional genomics. Later robot scientists include Eve, which discovered new drug candidates for tropical diseases (Williams et al., 2015), Robot Chemist (Burger et al., 2020), and work by Angello et al. (2022) to develop a closed-loop automated discovery system for organic chemistry. The cycles of experimentation that are executed by robot scientists vary depending on the domain, but have a core similarity. The general structure of these cycles is to form hypotheses, perform deductive simulations using a computational model to obtain predicted behaviour, and test these predictions by performing experiments and collecting data. With each cycle the robot scientist seeks to improve the quality of its predictions by forming better hypotheses. This is a supervised learning problem, specifically a type of active learning. Robot scientists are discussed in greater detail in **Paper I**.

One natural extrinsic motivation for the automation of science is that scientists should use all available tools to increase the quality of their observations, experiments, and inference, as in doing so they come closer to the phenomena they are seeking to understand. Technological development is a key driver of scientific discovery, and the strengths of machines and computers can be used to augment human scientists. Just as scientists have adopted computers and lab machinery, so will they adopt automated systems to improve their research, focusing on the advantages that machines have over humans. Machines do not tire, and can perform repeated tasks with high precision and consistency. Computers can reason at scales well beyond human minds, and can recall facts with perfect accuracy. They can verify logical arguments at high throughput, and autonomous research platforms can use software and hardware version control, logs, and audits to improve replicability and reproducibility. The complex technical challenges of modern science require robot scientists for

all their superhuman qualities mentioned above. With the right knowledge models and inference tools, closed-loop automated scientists have the potential to produce high-quality research.

Automation technologies also allow scientific research that otherwise would not be possible. DENDRAL arose from the joint motivations of a computer scientist, Ed Feigenbaum, wanting to build systems that could perform empirical induction and a biochemist, Joshua Lederburg, working on developing a system to identify chemical compounds on Mars (Lindsay et al., 1993). It was desirable for such a system to operate independently of human control, due to long radio signal delays. Ultimately, DENDRAL never flew on the Viking missions, but the resultant system and its successor MetaDENRAL would have been quite capable of conducting independent scientific research in an environment hostile to human life. Closer to home, automation technologies can limit human exposure to hazardous materials and pathogens. Better *in silico* models and digital twins can eliminate the need for some experiments, and allow for others that might never have been deemed safe or ethical had they been conducted on real subjects.

Other motivations are economic. Human society faces many severe challenges—ecological, environmental, medical, and social—that require better scientific understanding to shape decision-making and engineering. The demand for science greatly outstrips supply. Human scientists are an expensive and scarce resource. In many industries, pressures on human resource and a desire to increase efficiency, along with key technological innovations, have led to a growing adoption of automation over the past two centuries. First with the steam engine, then electrical power, and latterly with information technologies (Karabegović et al., 2020). Economists now refer to a fourth industrial revolution of autonomous systems, termed "Industry 4.0", driven by big data, analytics, cloud services and the internet of things (Frank et al., 2019), and accelerated by AI (Piccialli et al., 2025). Closed-loop robot scientists can drastically reduce human bottlenecks in science, while reducing the unit cost for research. Viewing science through an economic lens, opportunities emerge to meet the ever-growing technological demand for more and better science through automation. Other potential economic benefits of robot scientists include:

- fairer distribution of research, through reduced unit cost and better access through cloud labs and shared infrastructure;
- more resource-efficient experimentation due to the high-precision of experimental hardware, measurement devices, and statistical techniques (for example, the microchemostats for the robot scientist, Genesis, are several orders of magnitude smaller than a typical bench chemostat);
- more efficient re-use of scientific data, through high-quality logs, metadata, and machine-readable knowledge bases; and
- better working hours and conditions for human scientists.

Risks to these benefits being realised are clear, given that today's economic systems heavily favour actors with high starting capital, and incentivise monopoly over collaboration, profit-seeking over fairness, and exploitation over equity. However, science is fundamentally different to commercial industries, so there

is an opportunity for "Science 4.0" to strike its own path.

And there is an intrinsic motivation to the automation of science, in that it will help us understand the nature of science. Within any given scientific domain, the construction of a robot scientist is an engineering process that requires the specification of the elements of scientific method in that domain, the elicitation of system requirements, and the design and construction of subsystems. On a higher level, this process also requires the concrete specification, in machine-interpretable terms, of what constitutes scientific discovery in that domain – the machine needs to know it when it sees it. The *process* of building a robot scientist leads to insight into the discovery problems of that domain.

The following section presents some ideas relevant to the engineering of robot scientists. This concept is further discussed in **Paper I**, and in **Paper VI**.

2.3 Engineering robot scientists

To achieve the goal of closed-loop scientific discovery, robot scientists need subsystems that are capable of undertaking the different tasks in science. Seifrid et al. (2022) categorise the activities that are to be automated as either being *cognitive processes* or *motor function*. I would add a third category of *sensing* (or *perception*). Broadly speaking, cognitive processes require software, motor function and sensing require hardware (robotics and sensors).

While this distinction is a helpful abstract tool, a slightly different approach has been taken when designing robot scientists in the past, one that has been most informative for the projects in this thesis (in particular **Paper VI**). Namely to identify subtasks in the overall research cycle, and design subsystems to perform them, together with a coordination system. These subtasks may require a combination of cognition, motor control, and sensing, so this distinction is often a more natural engineering choice.

When engineering these systems it is important to consider each as part of the whole, as each will depend on those systems that it relies upon, and those that rely on it. The interfaces between systems are points of friction, that can provide starting points for engineering. For example, when designing the robot scientist in **Paper VI**, the capabilities of the robotic experimental platforms influenced the form of hypotheses and the experimental design programs.

To expand on the statements made in Chapter 1, and the research cycle shown in Fig. 1.1, the subtasks of any scientific method can be coarsely grouped into the following categories: hypothesis generation, experimental design, experimentation, empirical data analysis and processing, hypothesis testing, knowledge modelling and management, supervision and coordination, communication, and other auxiliary and supporting tasks. Robot scientists will need subsystems for each of these, and hereafter is provided a short introduction to each subtask and some of the considerations taken at each stage.

2.3.1 Hypothesis generation

Central to any scientific method is the formation of hypotheses about the domain of study. Given a theory, a hypothesis[1] is a set of statements that extend the theory to provide an explanation or prediction for a phenomenon (or a group of phenomena). From a hypothesis, its consequences are deduced using the theory context for the hypothesis. These consequences are then empirically tested. This is the hypothetico–deductive method of science, explored further in **Paper V**.

Hypotheses must be formed so as to be compatible with the existing theory and inference techniques. For example, a hypothesis of a new biochemical reaction could be introduced as a new row in the stoichiometric matrix of the current theory (See Section 3.4). The hypothesis together with the theory is used to draw predictions about what the resultant behaviour of the system will be.

Hypotheses are changes to the theory of a phenomenon, and as a result to record a hypothesis we need to make reference to the original theory. In **Paper II** we designed an ontology for recording changes to genome-scale metabolic models (GEMs).

The specificity of hypotheses Thompson and Skau (2023) state that for a hypothesis to be specific, it must be narrow, in the sense that the hypothesis applies to a smaller subset of phenomena than a broader hypothesis.

This can be demonstrated through an example, taking a hypothesis automatically generated by the robot scientist in **Paper VI**. The following are four hypotheses, where each becomes a narrower statement, applying to an increasingly restricted set of phenomena. At each stage of narrowing, the restricted parts of the hypothesis statement are underlined.

H1 "Yeast growth differs in some conditions compared to others."

H2 "Yeast growth differs in chemical supplementation with differing intracellular amino acid concentration compared to chemical supplementation with normal intracellular amino acid concentration."

H3 "Yeast growth is decreased in caffeine supplementation with raised intracellular arginine concentration compared to caffeine supplementation with normal intracellular arginine concentration."

H4a "AUC for OD600 of 20h yeast culture is decreased in 5mM caffeine supplementation with 5mM arginine supplementation compared to 5mM caffeine supplementation with no arginine supplementation."

This example raises a few interesting points. Firstly, by restricting the scope, the hypotheses get more and more specific. At each stage, the replacements

[1]The word *hypothesis* comes originally from Greek, the prefix *hypo-* meaning "under", and *thesis* meaning "a proposition". The meaning being that a hypothesis is a foundation for a subsequent argument (Harper, n.d.).

correspond to subclasses or sub-relations of the concepts in the broader hypothesis that preceded. When designing the hypothesis ontology for **Paper VI** we used the hierarchies in the domain ontologies to reason about hypotheses.

Secondly, note that hypothesis **H4a** is not strictly a more specific formulation of **H3**, because the concept of "intracellular arginine concentration" was replaced with "arginine supplementation". The reason changes like this were made to hypotheses in **Paper VI** is to reach a hypothesis that we could directly test in an experiment. But this is no longer a subclass or an equivalence of **H3**. An auxiliary hypothesis is actually needed to close the logic:

H4b "5mM arginine supplementation into yeast culture results in raised intracellular arginine concentration compared to no arginine supplementation."

Given these two statements, $\mathbf{H4} \equiv \mathbf{H4a} \wedge \mathbf{H4b}$ is now a more specific version of **H3**.

Thirdly, these different levels of abstraction work for different types of learning. Higher abstractions work well for qualitative learning; **H2** was more or less the form of hypothesis that was used to constrain the hypothesis generation in **Paper VI**, and all hypotheses in that work are more specific versions of **H2**.

2.3.2 Experimental design

To design an experiment that provides relevant data to evaluate a hypothesis requires knowledge of what the expected measurable indicators of the system behaviour are. These measurable indicators can inform the transformation of a hypothesis into something testable, as above from **H3** to **H4**.

In biology, the observable characteristics of a system are called its phenotype. Components of phenotype in yeast include: the rate of growth, the state of gene expression (transcriptome), the chemicals present (metabolome), or the shape of the cells (morphology). Phenotype is measured using various instruments to augment or replace human perception, for example by using a mass spectrometer to characterise the metabolome.

Experiment design also takes into account which experimental interventions are achievable with the available equipment and resources. For example, the original version of the hypothesis **H3** in **Paper VI** was the inverse: that a reduced intracellular concentration of arginine would lead to increased resistance to caffeine. The experimental agents had no direct way of introducing a higher intracellular concentration of arginine. Our robot was given the experimental restriction to only design experiments where the intervention was an increase in amino acid concentration, as this was something we could achieve through supplementation. This is an example of where the restrictions for experimental design led to a change in the original hypotheses, again showing the influence that experimental setups have on hypothesis formation.

Going from a hypothesis to an experimental design is rarely straightforward and requires creativity. In closed-loop discovery systems, this is an ideal application of LLMs, as they have the flexibility to generate alternative formulations of a hypothesis. We found in **Paper VI** that they are able to work within

laboratory constraints when provided with suitable prompts, and could produce outputs that we could symbolically verify before executing an experiment.

2.3.3 Experimentation

Human scientists conduct experiments with physical skill and the use of equipment. The skill of experimental scientists is in the manipulation of physical samples or systems, in the translation of an experimental plan to reality, and in responding to the unexpected or undefined variables that come up.

For robot scientists, robotics can be used to replace physical manipulation, and software to replace the coordination and analysis. Coordination software can be programmed to deal with certain types of problems or events during the experimentation, by responding dynamically. However, from our experience these software programs are not nearly as flexible or smart as human scientists in dealing with the unexpected. On the other hand, with robots it is possible to have much closer monitoring of an experiment, with every action automatically recorded in logs. In theory this makes experiments more repeatable, but this information is currently under-used.

Laboratory automation systems often take a modular approach, where different "work stations" are designed for different tasks. This is the approach taken for Adam (King et al., 2009) and Eve (Williams et al., 2015) which used robot arms and rails to move samples between work stations, and for the Robot Chemist (Burger et al., 2020), which used a commercially available mobile robot. This modular approach can be scaled up to an industrial level, in *cloud labs*. Taking inspiration from cloud computing, the idea is to centralise and share lab equipment, where researchers can "order" experiments remotely that are then executed autonomously, with data sent back to the researcher (Arnold, 2022).

2.3.4 Empirical data processing and analysis

Data collected during the experiment need analysing to identify if the experiment went as planned, and then to evaluate the hypotheses. The first step of evaluating if the experiment went as planned involves answering the question: did we actually achieve our desired intervention? For example, **H4b** was a necessary component of testing hypothesis **H4**, and in **Paper VI** we used intracellular metabolomics to confirm that supplementation of arginine actually led to increased intracellular concentration.

We also used an automated metabolomics processing pipeline (Brunnsåker et al., 2023; Reder et al., 2024) and wrote programs for the analysis of the growth profile data. Automating data analysis is a relatively mature aspect of robot scientists in most domains, so the focus in engineering a robot scientist often lies in connecting the empirical data back to the original hypotheses. This requires choosing appropriate statistical tests, and ensuring that data are presented in a suitable way for the test.

2.3.5 Supervision and coordination

Supervision involves setting the direction for a scientific study, providing guidance on decisions, setting priority, and identifying risks.

Coordination, or orchestration, is about managing the different agents involved in a scientific study and making sure they are doing the right things at the right times, resolving conflicts through scheduling or dynamic problem solving. For robot scientists, this is done by software. Eve (Williams et al., 2015), for example, uses bespoke software for the automation of experiments, which has plugins for the software and firmware required to communicate with the different instruments and its robotic arms, as well as a database. Scripts can be written for an experiment, including dynamic response to the experiment. The software needs to know the capabilities of each robot at its disposal and the experimental plan. Ideally, the systems should have some quality checks that they can use, to check that the experiment is running according to plan, or that maintenance might be required.

So coordination is largely for the purpose of collaboration between the different agents toward a shared goal. But there are also antagonistic aspects to coordination. That agents can be challenged, and held to account. It could even be the case that different agents might be directly competing with one another for resource or attention.

Humans can take initiative to self-supervise and jointly coordinate. The same is not true in general for machines, so the explicit design of supervision and coordination systems is especially important for robot scientists.

2.3.6 Communication

Of course, many of the activities listed here require communication between different agents, human or machine. Here, communication is taken to be with external parties, through publication of data and results, and receiving feedback, new theories, and new techniques. Communication design for robot scientists is not explicitly explored in this thesis, although it bears relevance to all the included papers.

2.3.7 Knowledge modelling and management

The knowledge and data used by scientists are recorded, both for the internal processes of the research group, and for communication with external parties. These could include data from experiments, intermediary results, protocols, theories, models, and hypotheses.

There are various definitions for *theory* and *model*. For the purposes of this section, a theory is defined as a set of general axioms and laws that apply to a specific set of phenomena that occur in the world, and a model is defined as a specific interpretation of axioms and laws that makes them true, and that represents concepts in the world. Models exist "between theory and the world" (Winther, 2015). For example there is a theory which explains in general terms that a reaction will occur in the presence of necessary substrates and conditions generating products. And a model of this theory might be a set of

specific reactions that occur during respiration in *S. cerevisiae*, with symbols that correspond to chemical species, enzymes, masses, and temperatures.

For a robot scientist, theories and models must be recorded in machine-interpretable formats. Theories and models can be recorded at different levels of specificity, with models inheriting properties while becoming more specific. This is related to the specificity of hypotheses described in Section 2.3.1. Ontologies are excellent technologies for capturing these types of propagation in scientific knowledge. Propagation can naturally be captured using class and property hierarchies.

2.3.8 Auxiliary and supporting activities

Aside from the core subtasks of the scientific method above, there are inevitably other activities that scientists undertake during the course of the scientific method. Drawing a clean distinction between scientific and non-scientific activities is not straightforward, but considered for the purposes of this discussion are those tasks that directly support or enable the core tasks above.

Examples of these activities in biology are: cleaning of equipment or lab spaces; restocking and resupplying consumables; transfer of equipment, samples, or files; or curating and managing software and data storage.

Quite often, as was the case with Adam (King et al., 2009), Eve (Williams et al., 2015), and with the work in this thesis, most notably in **Paper VI**, these tasks have been carried out by humans and not been automated. There can be several reasons for this, but usually it comes down to either limited return for a substantial investment (the activity is trivial for a human, but expensive to design a computer–robotic system for), or that it is of limited philosophical value to the research project (automating hypothesis design is a more interesting and valuable research contribution than automating the cleaning of a work surface).

There is in principle nothing stopping these activities being automated, and to scale up robot scientists this will be important. With scale, the returns may be worth the investment, and certainly it becomes a more interesting research question than for a small laboratory.

Chapter 3

Mathematical Models of Yeast Physiology

3.1 Systems Biology

Systems biology is an approach to studying biological systems that aims to understand how the behaviour of the system arises from the interaction of its constituent parts (Kohl et al., 2010). Systems biology, and its link to complex systems, is described in detail in Section 4 of the appended book chapter (**Paper I**), where we also outline why systems biology is a good domain in which to automate scientific discovery. Partly this is because of the inherent complexity of the systems involved, which limits the progress that can be made with non-automated scientific methods. Robot scientists can manage complexity more precisely and at larger scales than humans, both in experimentation and in analysis.

A core concept in systems biology is the connection of *genotype*—the DNA sequence information of an organism—to *phenotype*—the observed characteristics of the organism. The reasoning behind treating DNA sequence information as the source for phenotypical observations comes from two hypotheses, proposed by Crick (1958) as the "Sequence Hypothesis", and the so-called "Central Dogma" of molecular biology, that the primary form of information transfer within cells is as follows: DNA is *transcribed* to RNA which is in turn *translated* to proteins. Specifically, the "Central Dogma" stated that information passes from nucleic acids to proteins, but is not transferred from proteins to nucleic acids. Later, Crick (1970) clarified this hypothesis stating that this hypothesis was intended to apply to the general case for living organisms, and that though cases of information transfer from proteins to nucleic acids were theoretically possible there was no evidence for these interactions at the time.

Developments in molecular biology since the 1970s have demonstrated numerous potential violations of this mechanistic view of information transfer between molecules in biological systems. Some examples are: reverse transcription, from RNA to DNA (Baltimore, 1970; Temin & Mizutani, 1970); and

post-translational protein modification, for example through phosphorylation (Krebs & Graves, 2000). Opinion is divided on whether these truly represent violations of the original hypothesis. On the one hand, examples such as those raised above challenge this hypothesis. However one could argue that even in these cases, the original source of the sequence information remains the DNA.

Regardless, either interpretation justifies structuring enquiry around the genotype–phenotype relationship. Systems biology enables a nuanced approach allowing for feedback loops and incorporating additional classes of molecules, such as sugars and lipids, into a complex system of signalling, gene regulation, and metabolism. Each of these concepts is discussed in more detail below.

3.1.1 Eukaryote cellular metabolism

Metabolism refers to the consumption, transformation, and production of chemical compounds, by an organism, through various biochemical reactions. Metabolism has three main purposes: to make energy available, to make building blocks for structures, and to eliminate waste.

A metabolic pathway is a set of reactions, usually a sequence or a cycle. Pathways are a useful modelling tool, as they allow grouping together reactions that facilitate particular functions subordinate to the purposes listed above. They are a good subject for study, being often small enough to be tractable for detailed methods and analysis. However, to form a good model of most phenomena, it is often necessary to consider superpathways (collections of pathways) and the metabolism of the whole cell or organism.

Biochemical reactions come in various types. Many require catalysis[1] by enzymes, formed of proteins. The set of feasible reactions in a particular organism is therefore largely determined by its genome, which encodes for all proteins the organism can produce. Many reactions are feasible in multiple different organisms, though the specific gene for the reaction may differ as there are often several different enzymes that can catalyse the same reaction (isoenzymes). Pathways are often partially or entirely evolutionarily conserved across organisms.

In systems biology, knowledge about metabolism is drawn from various sources and compiled in metabolic network models (MNMs), which are digital representations of pathways, superpathways, and catalysis. Metabolic network models are covered in more detail in Section 3.3.1, and in **Paper III**.

3.1.2 Gene regulation in eukaryotes

Genes are segments of DNA[2], and the process of synthesising functional products (e.g. proteins) from the DNA is referred to as gene expression. The state of gene expression in a cell controls which processes occur within the cell. A significant part of this control is through the expression of metabolic genes, those which encode proteins that form enzymes. Eukaryotes regulate

[1] *catalysis*—reactions are catalysed by a molecule when it accelerates the reaction without being itself consumed.

[2] Specifically, genes are segments of DNA that are transcribed into RNA.

genes and their activity in response to environmental stimuli and also within the organism. This is achieved by eukaryotic cells at several levels.

1. Transcription is controlled by limiting the amount of messenger RNA (mRNA) that is produced from a given gene.
2. Post-transcription there are events that regulate the translation of RNA into proteins.
3. Post-translation there are mechanisms which modify proteins, which can affect their activity.

Gene regulatory pathways govern the production of proteins through controlling gene expression (Ma et al., 2016). An example of a mechanism of gene regulation specific to eukaryotes is physically restricting access to DNA promoters[3] through the structure of chromatin. DNA is wound tightly on nucleosomes that form the chromatin fibre, and modifications to the chromatin structure can regulate gene expression (Klemm et al., 2019).

Transcription factors are proteins that recognise and bind to a segment of DNA adjacent to the genes they regulate. There are a variety of processes through which transcription factors regulate genes, but essentially they control the rates of transcription. Messenger RNA (mRNA) transcription most often cannot occur without the help of transcription factors. As the ground state for transcription is restrictive, positive regulation is the predominant form of control. Transcription factors are themselves regulated, resulting in a complex interaction network (Alberts, 2017). **Paper V** explores the hypothesis that the gene being studied is a transcription factor that regulates respiratory pathways in *S. cerevisiae*.

3.1.3 Cell signalling

Cells interact with each other and the environment by sending and receiving signals. These signals can also be sent internally. This process of cell signalling is enabled by the binding of small molecules known as ligands to an effector molecule, frequently a protein. The binding of the ligand to a particular site on the effector causes a change which allows the effector to perform a function. This could be up- or down-regulating (increasing or decreasing expression of) a particular gene product, or modifying an enzyme complex to increase or decrease its activity. Signalling, like gene regulation, is a complex mode of control in eukaryotes, with many interacting signalling molecules and other systems such as metabolism and gene expression.

As a result, though many individual signalling interactions such as the binding of adrenaline to adrenoreceptors are well-studied (Rasmussen et al., 2007, 2011), cell signalling networks are in general poorly understood in most eukaryotes. Yet they can provide promising explanations for phenomena such as ageing (Greer & Brunet, 2008), and effective therapeutic options for diseases, for example leukaemia (Weisberg et al., 2005).

[3] *DNA promoter*—a binding site in the genome where RNA transcription begins.

3.2 *Saccharomyces cerevisiae*: the model organism for eukaryote biology

Eukaryote cellular physiologies are extremely complex systems. Gene regulation and enzymatic catalysis are but two of many processes that enable information propagation and feedback, both within cells and across cell boundaries; and the entities and processes involved occur across a huge range of timescales and length scales.

As discussed in Chapter 2, when studying systems directly is impractical or infeasible, models should be used. This is often the case with eukaryotes. For example, there are many experiments that would be undesirable or immoral to conduct using human subjects, and *in vitro*[4] experiments often do not recreate the complexity of the object systems. By using the yeast *Saccharomyces cerevisiae* as a model organism, the aim is to understand processes relevant to other eukaryotes through yeast. These could be evolutionarily conserved functions, or by transplanting genes of interest.

S. cerevisiae is the model organism for the eukaryotic cell, for a number of reasons. Firstly, there are tools available for easy genetic manipulation of yeast (and fewer ethical and legal issues in doing so than with higher eukaryotes). Cultivation cost is relatively low, in terms of the key resources: money, time, space and human resource. And *S. cerevisiae*'s was the first eukaryotic genome to be fully sequenced (Goffeau et al., 1996). There is also a wealth of experience and knowledge on *S. cerevisiae*, much of it stored in community-curated knowledge graphs that can be used as a rich prior for discovery. Yeast is also heavily used for bioengineering purposes, where genotype and conditions are manipulated to efficiently produce a desired product, for example a pharmaceutical. A history of the development of yeast research is covered in Oliver (2022).

3.3 Community knowledge and models in systems biology

Scientific knowledge can be stored and communicated in different forms. The types of knowledge models used will depend on (1) the type of data and knowledge being represented, (2) the technologies used to collect, store, and communicate data, and (3) the inference techniques that will be applied to the data. For example, metabolic reaction data are statements about a process that involves inputs and output and relative quantities. The natural way to represent a chemical reaction is to use a mathematical formula, for instance as demonstrated in **Paper III**:

$$\text{D-gluconate 6-P} + \text{NADP} \overset{\text{EC 1.1.1.44}}{\rightleftharpoons} \text{ribulose 5-P} + \text{CO2} + \text{NADPH}$$

[4] *in vitro*—experiments performed with biological material, but outside of the biological context; from the Latin for *in glass*.

If using paper and pen to take notes on a particular reaction, this representation is quite natural and sufficient. To encode this reaction into a computer, there are many options. For example, each reaction could be stored in an array, such as:

```
['D-gluconate 6-P','1','NADP','1','<->','EC 1.1.1.44',
 'ribulose 5-P','1','CO2','1','NADPH','1']
```

This form is easily read by a computer, and would allow for the construction of, for example, a differential equations model using a simple program.

Frequently, it is not known in advance which inference techniques will be applied to a particular knowledge base, so preserving as much information as possible in the knowledge base is preferable. A highly suitable scheme for knowledge modelling that allows detailed recording of knowledge is a graph model. Graph databases encode information using nodes (entities and literals) and edges (relations and properties). Facts are statements about a relation between two entities, or a literal property of an entity, and are recorded with an edge between two nodes. Many scientific databases use a graph model, including those most useful in systems biology. Graph databases are often constructed using ontologies—formal definitions for a domain of the terms used to define its entities, relations, and properties—as graph schemata.

The largest and most prominent organised effort for constructing and maintaining ontologies in biology is the Open Biological and Biomedical Ontologies (OBO; Jackson et al., 2021). Community standards organisations, such as OBO and COMBINE (the COmputational Modeling in BIology NEtwork; COMBINE, n.d.) organise efforts to develop and maintain data ontologies and data standards. In systems biology, ontologies used include the following.

Systems Biology Ontology (SBO) Arising from a desire for better software infrastructure for computational models, SBO describes the computational modeling domain for systems biology. SBO entities are split into eight subclasses of 'systems biology representation', defined as a "representation of an entity used in a systems biology knowledge reconstruction, such as a model, pathway, network."

These direct subclasses are: 'mathematical expression', 'metadata representation', 'modelling framework', 'occurring entity representation', 'participant', 'participant role', 'physical entity representation', and 'systems description parameter'.

SBO has its origins in the systems biology markup language (SBML), which was developed as a common syntax for computational models. In addition to the common syntax of SBML, SBO allows for a common understanding of the semantics of the components of computational systems biology models (Engel et al., 2025).

Relation Ontology (RO) Curated by OBO Foundry as a way to ensure that entities between different OBO ontologies, such as GO and APO, are able to be connected in a standardised and logically consistent way (Mungall et al., 2020).

Gene Ontology (GO) A massive project, GO is for describing the functions of genes, and has almost 40,000 terms split over three main sub-domains:

molecular function describes the interaction at a molecular level. For example, a gene product could be a "catalyst".

cellular component captures where in the cell the function takes place.

biological process describes which higher-level functions the gene products are a part of. An example might be "DNA repair".

There are approaching 10m GO annotations, that is relations between specific genes and terms in GO. These relations are specified using RO, and primarily curated by the members of the GO Consortium, which includes many of the actors and research groups responsible for curating the other ontologies and databases mentioned here (Ashburner et al., 2000; The Gene Ontology Consortium, 2026).

Ascomycete Phenotype Ontology (APO) Used to define terms about phenotypes (observable characteristics) of ascomycete fungi, of which *S. cerevisiae* is one. As discussed in **Paper VI**, APO is designed for recording hypotheses about genotype–phenotype relations, and was created for the Saccharomyces Genome Database (SGD) (Engel et al., 2025).

Chemical Entities of Biological Interest (ChEBI) ChEBI refers to both the database of chemical entities, and the ontology that is used by the database. The ChEBI ontology contains terms for defining chemicals according to their biological roles, chemical roles, and chemical structure (Malik et al., 2026).

3.3.1 Genome-scale metabolic models

Metabolic network models (MNMs) are computational models of the metabolism of a particular organism. For *S. cerevisiae*, the first MNM was created in 1995, and was a model of the central carbon metabolism (van Gulik & Heijnen, 1995). Genome-scale metabolic models (GEMs) are MNMs that aim to cover the whole of an organism rather than a specific sub-component or system. GEMs are recorded and communicated using SBML, and are used as the basis for creating simulation models according to one of the mathematical frameworks described in Section 3.4. GEMs can be manually curated, or learned from data, and in the case of *S. cerevisiae* the research community and industry are invested in developing and maintaining GEMs.

The very first GEM for *S. cerevisiae*, iFF708, was presented in 2003; the significance of the number 708 was the number of genes that were included in the GEM, along with 825 metabolites, and 1145 reactions. iFF708 modelled two cellular compartments—the cytoplasm and the mitochondria—and included transport reactions between compartments and the extracellular space.

In the decades since iFF708, many GEMs have been released, gradually increasing the coverage of the models. A history of this development, along with key statistics about the models, is shown in the timeline in Fig. 3.1. A

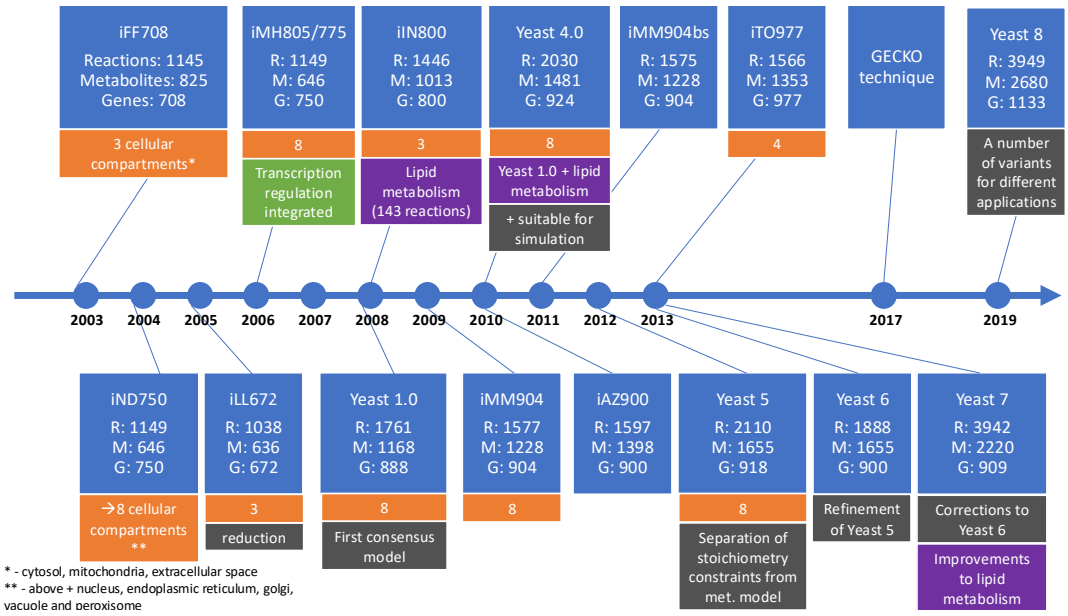


Figure 3.1: Timeline of *Saccharomyces cerevisiae* genome-scale metabolic model (GEM) development up to 2019. Developments since then are coordinated through the YeastGEM project with more frequent releases and version control (Zhang et al., 2024). The development of these GEMs since 2003 is covered in detail in Chen et al. (2022).

significant landmark was reached in 2008 with the release of Yeast 1.0, the first *S. cerevisiae* GEM arrived at through community consensus. This consensus-driven approach to GEM development continues to this day with yeast-GEM, the latest version of which being Yeast9 (Zhang et al., 2024). Yeast-GEM was used for modelling in **Paper III** and **Paper V**. In **Paper II** we introduced an ontology for describing changes to GEMs semantically.

3.4 Modelling Frameworks

Models of yeast vary from deterministic and high-resolution to descriptive or pedagogical. As discussed in **Paper I**, there are various desirable qualities of scientific models, including: predictive power; parsimony; explanatory usefulness; consistency across contexts; and consistency with different scientific models. For most models of *S. cerevisiae* it is desirable to be able to exploit the models to make predictions about real-world behaviour of the system, and some of the mathematical frameworks commonly used are discussed briefly below.

All of these models can be constructed from a common knowledge graph, and most of them directly from a GEM. This means that by choosing a knowledge graph structure, as done in the papers in this thesis, the knowledge can be abstracted away from the inference. Not only does this allow tools like RIMBO to record semantically meaningful changes to the model, but also the inference framework most suited to the problem can be chosen.

3.4.1 Differential equations models

One common technique is to model the abundances of genes, proteins and chemical species using systems of coupled ordinary differential equations (ODEs) with time as the dynamic variable. These models have been successfully employed to model various biological processes including: central carbon metabolism; batch fermentative growth; and toxicity responses. The differential equations are often based around reaction kinetics paradigms.

A great challenge with these models is parametrisation; the models can have tens of thousands of parameters, only a fraction of which have experimentally obtained values, which are often condition specific. Machine learning techniques have recently been employed to predict these parameter values (Li et al., 2022; Maeda et al., 2022).

Another challenge is that timescales and length scales across the different systems involved vary across at least five orders of magnitude, from the molecular scale (1×10^{-10} m) to the cellular scale (1×10^{-5} m) (Castiglione et al., 2014; Southern et al., 2008). ODE models based on Michaelis-Menten kinetics do not explicitly model the length dimension, but the timescale of reaction kinetics can vary over orders of magnitude depending on the reaction (Resat et al., 2009), and is vastly quicker than the timescale of gene expression (Carthew, 2021).

A third challenge is that biological processes are stochastic, so deterministic differential equations models will likely be unable to capture a great deal of behaviour. For certain classes of molecules the assumption of continuous concentrations may not be valid due to the low count of molecules and their compartmentalisation (Resat et al., 2009).

3.4.2 Constraint-based modelling

Another popular modelling framework is a constrained optimisation method, where the objective function is defined as a real-valued function representing a biologically realistic quantity. Examples of commonly used objectives are growth maximisation or minimisation of uptake of a particular carbon source (García Sánchez & Torres Sáez, 2014; Orth et al., 2010). Other knowledge about the cell is encoded via constraints on this optimisation. The fundamental constraints are the rates of flux through chemical reactions in the cell. A common approach is to assume the system is in a steady state and the sum of the fluxes for any given chemical species is zero, hence the technique is known as flux balance analysis (FBA). A more detailed explanation of the mathematics of FBA is provided in Section 2.5 of **Paper III**.

Various extensions and modifications to FBA have been proposed that relax or work around this assumption, for example: dynamic flux balance analysis (dFBA) which couples FBA solutions with ordinary differential equations (Mahadevan et al., 2002); flux variability analysis (FVA) which characterises the space of fluxes that give rise to an optimal solution (Mahadevan & Schilling, 2003); and two techniques, regulatory on/off minimisation (ROOM, Shlomi et al., 2005) and minimisation of metabolic adjustment (MOMA, Segrè et al., 2002) which seek to find a likely flux distribution—in comparison to a reference

distribution—after a genotypic change. Some techniques extend FBA to more nuanced constraints, for example through the inclusion of setting reaction flux constraints through abundance of relevant enzymes (enzyme-constrained flux balance analysis, ecFBA, Sánchez et al., 2017).

3.4.3 Logical models

Formal mathematical logic has also been used to model different parts of cellular biology. A common form of logic modelling is propositional logic (Boolean) models of gene regulation and signalling. First-order logic models of metabolism have also been developed previously. For more detail see the introduction to **Paper III**.

3.4.4 Hybrid models

Some approaches seek to extend FBA models yet further by integrating them with separate models for other cellular processes, for example genetic regulation and signalling processes. These hybrid models employ different mathematical formalisms for different cellular processes and rely on bespoke techniques at the boundary between models to integrate them together. One example of a hybrid model is presented in Österberg et al. (2020), which combined enzyme constrained FBA with a Boolean signalling model to predict non-trivial yeast phenotypes. A review of hybrid modelling approaches in systems biology was conducted by Cruz and Kemp (2021), showing a diverse approach to modelling, including combinations of modelling techniques listed here, amongst others, applied to a variety of biological applications.

Chapter 4

Techniques

4.1 Logic

Logics are mathematical languages that relate premises and conclusions and enable formal reasoning (Ben-Ari, 2012). There are various logics that can be used for reasoning in science, and this section presents a brief overview of propositional logic, the concept of satisfiability, normal forms, and first-order logic.

Propositional logic

The elementary form of logic is propositional logic, which deals in assigning truth value to statements about a world (propositions). An example of a proposition would be:

> "Uppsala is the capital city of Sweden."

In the world we live in, this proposition would be assigned the truth value 'false'. But it is possible to consider a world where this statement would be assigned the truth value 'true'. Indeed if we consider the world as it was in the 14th Century then this proposition would be assigned 'true'.

Propositions are atoms; they can be used to build complex formulas using Boolean operators such as negation (`NOT`, $\neg$), conjunction (`AND`, $\wedge$), and disjunction (`OR`, $\vee$).

Satisfiability

An interpretation for a formula is a function that assigns truth values to each atom that appears in the formula. A formula is considered unsatisfiable if it is false in all interpretations. If there is at least one interpretation for which the formula is true, then it is satisfiable. The notion of satisfiability can be extended to a set of formulas naturally, by requiring that there exists an interpretation under which each formula in the set is true.

Normal forms

For any given logical formula there are many possible ways to express an equivalent formula. An example is the following tautology, one of De Morgan's laws:

$$\neg(A \vee B) \iff (\neg A) \wedge (\neg B) \tag{4.1}$$

A way to express logical formulae that has advantages for automated theorem proving is conjunctive normal form (CNF). To be in CNF, a formula is written as a conjunction of disjunctions of literals (atoms or negated atoms); the right hand side of (4.1) is in CNF. It is possible to express every formula in propositional logic in CNF.

A different notation for CNF is clausal normal form. Instead of using the symbols for disjunction and conjunction to construct the formula, we appeal to the structure of the normal form, expressing a formula as a set of sets of literals. For example, the right hand side of (4.1) in clausal normal form as $\{\{\neg A\}, \{\neg B\}\}$. A theory is a set of closed formulae (formulae with no free variables). Clausal normal form is an efficient representation of logical theories that is often used in automated theorem proving.

First-order logic

First-order logic (FOL) extends propositional logic to allow for relations between variables. Relations are represented using predicate symbols, and quantifiers such as *for all* ($\forall$) and *there exists* ($\exists$) are introduced to allow for formulae that express general statements about relations. Instead of atoms being propositional statements, atoms in FOL are a predicate symbol with a list of arguments. These arguments are either constants or variables in the domain of the relation the predicate symbol denotes. For example, FOL allows for a statement such as:

> "A dog is happy if it has a bone."

This could be expressed with the following logical formula, containing new predicates $\mathsf{dog}\backslash 1$, $\mathsf{bone}\backslash 1$, $\mathsf{has}\backslash 2$, and $\mathsf{happy}\backslash 1$:

$$\forall \mathsf{x} \forall \mathsf{y}(\mathsf{dog}(\mathsf{x}) \wedge \mathsf{bone}(\mathsf{y}) \wedge \mathsf{has}(\mathsf{x}, \mathsf{y}) \rightarrow \mathsf{happy}(\mathsf{x}))$$

The atoms in this formula are $\mathsf{dog}(\mathsf{x})$, $\mathsf{bone}(\mathsf{y})$, $\mathsf{has}(\mathsf{x}, \mathsf{y})$, and $\mathsf{happy}(\mathsf{x})$.

FOL has many desirable properties for automated reasoning. It is an expressive language and can therefore be used to represent a wide variety of concepts and theories. However, the fact that it is not as expressive as higher-order logics means it is semi-decidable, meaning that there is an efficient procedure for checking if a formula is in a theory (but no such method for checking that a formula is not in a theory).

As stated earlier, it is possible to express every formula in propositional logic in CNF. The same is not true in FOL, however, Skolem's theorem states that for any closed formula ϕ there exists a formula ϕ' such that ϕ is satisfiable if and only if ϕ' is satisfiable (Ben-Ari, 2012). A useful consequence of this is that any theory can be mapped to one in CNF or clausal normal form to assess satisfiability.

Reasoning

Reasoning on logical theories can be broken down into a few main categories. Deductive reasoning derives conclusions from premises and laws; for example, this includes querying a graph database, or determining whether a first-order logic theory entails a conjecture. Inductive reasoning and abductive reasoning seek to provide explanations for facts by generating either laws (induction) or facts (abduction); for these, statistical inference is needed. For a more detailed explanation of the types of reasoning, see **Paper I**, Section 2.1.

4.2 Description logic and ontologies

An introduction to description logic and ontologies is provided in Section 2 of **Paper IV**.

4.3 Automated theorem provers

Automated theorem provers (ATPs) are software that can perform reasoning tasks on logical theories. A common task for ATPs is deciding whether a conjecture, C, is entailed by a given theory T and optional hypotheses H. In particular in science, the conjecture usually takes the form of some statement rooted in empirical data, and the question is whether the theory, perhaps together with a hypothesis, entails the data. ATPs most often take inputs in the form of FOL theories, and these are commonly written in clausal normal form. Formally, the problem posed to the ATP is:

$$T \wedge H \overset{?}{\models} C.$$

Conjectures are often posed in negated form and then submitted to a SAT solver, an algorithm to decide satisfiability. In which case the proof takes the form of a refutation of the negated form that shows the unsatisfiability of $T \wedge H \wedge \neg C$. Or a demonstration of the satisfiability of $T \wedge H \wedge \neg C$ which shows that C is not entailed by $T \wedge H$ under any interpretation. Constructing such arguments for FOL theories requires heuristic search, and the design and implementation of algorithms for this task is the core activity in developing ATPs for FOL (Korovin, 2008).

ATPs are primarily applied to mathematical reasoning tasks (Urban & Vyskočil, 2013). But they have also been used in engineering applications including software verification (Georgiou et al., 2022) and hardware verification (Goel & Ray, 2022; Khasidashvili et al., 2015). More detail on how ATPs are applied to the systems biology models in this thesis is provided in **Paper III**.

4.4 Artificial neural networks

Artificial neural networks are learned nonlinear models, composed of layers of processing units (neurons), where the input to each neuron is a linear

combination of the output from the previous layer. Nonlinearity is introduced by calculating a nonlinear activation on each neuron, commonly the rectified linear unit (ReLU) (Bengio et al., 2015).

Graph neural networks GNNs are a type of neural network, where the architecture of the network is designed according to the graph structure. Further introduction to graph neural networks is provided in Section 2 of **Paper IV**.

4.5 (Large) language models

Language models are designed to perform tasks where the data are encoded in language. Though many mathematical formalisms have been used to construct language models, including first-order logic, the most prevalent language models today are based on neural networks. These models owe much of their power to the massive scale of the networks used, hence the name large language models (LLMs).

Techniques for language models emerged from the field of natural language processing (NLP). In the definition of this field, and the techniques thereof, is the concept of *natural language*, which is commonly defined as a synonym for human language (Chowdhary, 2020). Most definitions of natural language exclude programming languages. Yet LLMs have demonstrated a high capability to perform tasks with these languages, and in **Paper VI** we demonstrate that LLMs can be used to work with computer languages (such as Prolog, JSON, Python, etc.) in a scientific context.

Chapter 5

Summary of Papers

Paper I (Gower et al., 2026) considers the design of robot scientists, systems that combine artificial intelligence with robotics to automate scientific discovery. It connects machine learning concepts to ideas from the philosophy of science to argue that scientific discovery should be viewed as an active learning problem.

Paper II (Kronström et al., 2023) expands on the concept of scientific discovery as theory choice. In this paper RIMBO, an ontology for representing theory changes in systems biology, is presented. RIMBO is tested on community curated knowledge models for *Saccharomyces cerevisiae*, genome-scale metabolic models (GEMs), and shows that it can capture model changes at scale and precision.

Paper III (an extension of Gower et al., 2023) builds methods for logical inference from these community knowledge models. A first-order logic model of *S. cerevisiae* metabolism is paired with an automated theorem prover to simulate growth and pathway configuration, and hypothesise corrections to the model to fix broken pathways. We propose a method for evaluation of predictions using an alternate quantitative model, to guide simulations and abductions.

Paper IV (Kronstrom et al., 2026) proposes a new method to learn embeddings of ontology concepts that couples graph neural networks with semantic loss functions to penalise embeddings that violate ontology constraints. This method is applied to a curated knowledge graph that draws from community databases, and evaluated by predicting gene interaction effects in *S. cerevisiae*.

Paper V (Bjurström et al., 2026) employs simulation methods to refine a hypothesis about the function of a *S. cerevisiae* gene whose biological function is currently unknown, *YGR067C*. Predictions of phenotype are made, and compared against empirical data. To obtain high-quality empirical data from a dynamic cultivation, we design and employ automated experimental protocols.

Paper VI (Brunnsåker et al., 2025) is where we combine many of the other ideas in this thesis and build a system for automated discovery. A system of software agents is designed, combining symbolic reasoning and LLMs, and integrated with laboratory experimental agents. A new ontology for recording generalised hypotheses in systems biology is designed, and its implementation

allows for efficient data reuse, evaluation of hypotheses, and contributions to scientific knowledge.

Paper I: The Use Of AI-Robotic Systems For Scientific Discovery

Note to reader As mentioned in Chapter 1, this book chapter is a useful partial introduction to the contents of this thesis, and is recommended to be read first as it provides context to other matters discussed in the introductory chapters and in the appended papers.

Problem

Scientific discovery is a process of developing theories and models about phenomena. The automation of science requires the design of systems that can do this independently of human researchers. If an automated discovery system is to make any meaningful progress in a given domain, then it must have capability to perform empirical experiments, in addition to scientific reasoning capabilities. This is the main idea behind robot scientists—systems that combine artificial intelligence (AI) and robotics to develop and test hypotheses.

By nature, the development of robot scientists is a cross-disciplinary endeavour. Domain experts can encode background knowledge for the machine, and inform the design of experimental robotics. Mechanical and electrical engineers can design robotic systems for experimentation. And the AI systems for a robot scientist can be developed by software engineers and machine learning researchers. However, many machine learning researchers do not have a background in the natural sciences, and subsequently may lack an understanding of how best to design software systems that perform well for scientific discovery.

Approach and Results

This book chapter aims to provide machine learning researchers with an introduction to the research problem of the automation of scientific discovery. We address considerations for the design of robot scientists. We begin by examining the scientific method using concepts and models from the philosophy of science. We define the concept of a scientific model and its use by robot scientists. We present three components of the scientific method—logical inference, statistical inference, and parsimony—and how they are applied to the development of scientific models. We finish the opening section by presenting a set of scientific values for examining the quality of a given model which enables comparison between competing models, the central aspect of scientific method that allows for progress.

Section 3 of this book chapter analyses scientific discovery from the perspective of the three main machine learning paradigms: supervised learning, unsupervised learning, and reinforcement learning, see Fig. 5.1. We discuss how aspects of machine learning algorithms could be mapped to aspects of the scientific method. We conclude that scientific discovery should be viewed as a supervised learning problem, with input-output pairs for training data coming either from controlled experiments or from observational studies. We discuss further that semi-supervised learning—where unsupervised learning

is integrated into a supervised learning approach (not shown in Fig. 5.1)—is also a very useful paradigm to adopt, as it can allow for increased performance when labelled data are sparsely available, as is often the case in science.

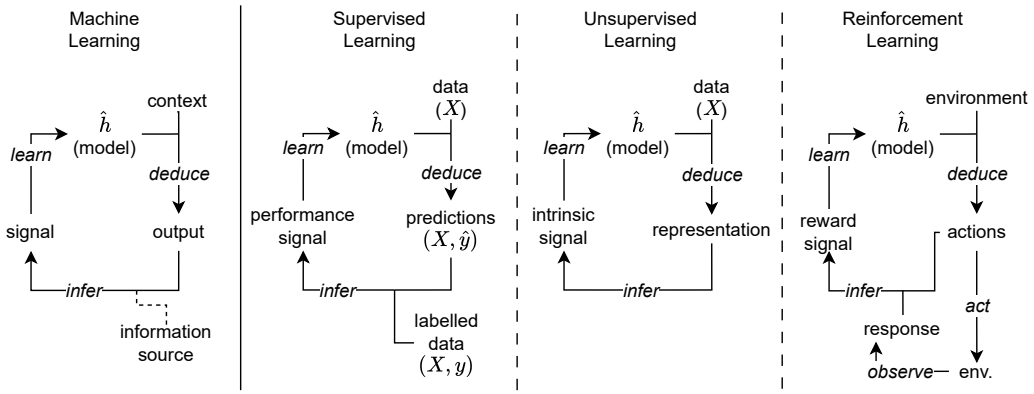


Figure 5.1: Flowcharts representing high level processes for: generic machine learning; supervised learning; unsupervised learning; and reinforcement learning. Unitalicised text represents inputs and outputs; italicised labels for connectors represent processes. In **Paper I**, we analyse scientific discovery from the perspective of each paradigm, concluding that supervised learning is the most useful.

Section 4 is primarily an introduction to the domain of systems biology, covered in Chapter 3 of this thesis. We also cover two examples of robot scientists being applied in systems biology: ADAM (King et al., 2009) and Eve (Williams et al., 2015). Section 5 is a case study using the example of the robot scientist Genesis and LGEM$^+$, covered in Chapter 3 of this thesis and in appended **Paper III** respectively.

Author contributions

The conceptualisation of the project was done by Ross D. King, and Alexander H. Gower. Content included in this chapter was informed by discussions between **A.H.G.** and each of the co-authors (Konstantin Korovin, Daniel Brunnsåker, Filip Kronström, Gabriel K. Reder, Ronald S. Reiserer and John P. Wikswo). The manuscript was written by **A.H.G.** and edited by **A.H.G.**, D.B., K.K., and R.D.K. The project was supervised by R.D.K., and Ievgeniia A. Tiukova. The funding for the project was acquired by R.D.K.

Paper II: RIMBO – An Ontology for Model Revision Databases

Problem

Knowledge in science, and particular in systems biology, is often stored as machine-readable files that allow for computational modelling of the phenomena under study. Genome-scale metabolic models (GEMs), covered in Chapter 3, are the primary models used in systems biology. When improvements are made to these computational models through scientific enquiry, this results in changes to the content of these files. Commonly, a copy is made of the model and changes are incorporated into this copy. However as these models grow in size, complexity, and number this presents challenges.

Changes to models are often bundled together to resolve the problem of multiple large files. However, this breaks the relationship between a single hypothesis and a change to the model. This makes reasoning about changes to models quite difficult, especially for large-scale computational reasoners. Versioning software, such as Git, is used to maintain some community models, e.g. Yeast-GEM Zhang et al., 2024. While effective for combining changes from many sources, these tools do not explicitly connect reasons for changes to the changes themselves in a reliably queryable manner.

A semantic database for model revisions is desirable to address these challenges, and in order to use the models to guide experimental design in closed-loop systems.

Approach and Contributions

We present here an ontology, RIMBO, for capturing and explaining changes to computational biology models, specifically genome-scale metabolic models (GEMs) written in RDF/XML. RIMBO introduces ontology terms, and links with existing ontologies, to enable individual model changes to be linked semantically to the reasons for the change and the details of the change to the model. RIMBO provides traceable model improvement history that can be reasoned about and queried, for example using the graph query language, SPARQL. RIMBO combines classes and relations from existing ontologies with new classes and relations. We demonstrate modelling example revisions to yeast-GEM (v8).

Author contributions

The conceptualisation of the project was done by Ross D. King, Filip Kronström, and **Alexander H. Gower**. The ontology was designed and curated by F.K. Code to implement RIMBO was also developed and tested by F.K. The experiments to demonstrate revisions on the yeast-GEM were designed and executed by **A.H.G.** and F.K. The scalability experiments were designed and executed by F.K. The data were prepared and curated by F.K. and **A.H.G.** Figures were designed and prepared by F.K. The manuscript was written by

F.K. The project was supervised by R.D.K., and Ievgeniia A. Tiukova. The funding for the project was acquired by R.D.K.

Paper III: LGEM$^+$: Automated Improvement of Metabolic Network Models and Model-Driven Experimental Design through Abduction[1]

Problem

Knowledge about yeast is highly structured due to community efforts to standardise, retain and distribute it. This is covered in more detail in Chapter 3. The models that are stored in these databases are therefore improved upon incrementally. Incremental improvements to these models are generally made through careful study of a particular entity, pathway or process in the organism. Making improvements to these models is a time-consuming process, primarily because of the human resource required to hypothesise and design experiments, as well as analyse results. Hypothesis generation is particularly time consuming, especially in systems biology, because of the complexity of the organisms and models.

With this paper we aimed to provide a framework for improvement of models of yeast metabolism that would reduce the time required of human scientists in the discovery process, and increase the quality of the hypotheses generated. We sought to design an inference framework for genome-scale models, retaining their logical structure, that would allow for deduction of conclusions from theories and hypotheses, as well as abduction of hypotheses to explain gaps in the theories.

Approach and Contributions

The main contributions of LGEM$^+$ as presented in this paper are: (1) a compartmentalised first-order logic (FOL) model of yeast metabolism; (2) a set of algorithms for the extraction and analysis of metabolic pathways from simulations; (3) a two-stage method for the abduction of novel hypotheses on improved models; (4) scalable methods for evaluating these models and hypotheses; (5) an algorithm to integrate FBA with abductive reasoning. We use an ATP and express knowledge encoded in GEMs as FOL axioms, retaining the semantics of the original models (which are encoded in XML files following the SBML standard, see Section 3.3). The form of the axioms is informed by knowledge about general processes such as biochemical reactions, enzyme formation and catalysis.

We tested this framework using several community models (GEMs) on two separate tasks. Firstly, we performed a genome-scale single-gene deletant viability screen. To evaluate this we used experimental data from previously published research as ground truth data for the predictive task. Secondly, we used the ATP to generate reaction pathway configurations for a defined growth medium. We compared these extracted pathways against two sources: ground

[1]This manuscript is an extension of a peer-reviewed paper published in the proceedings of the 26th International Conference on Discovery Science (Gower et al., 2023, listed as **a** under "Other Publications").

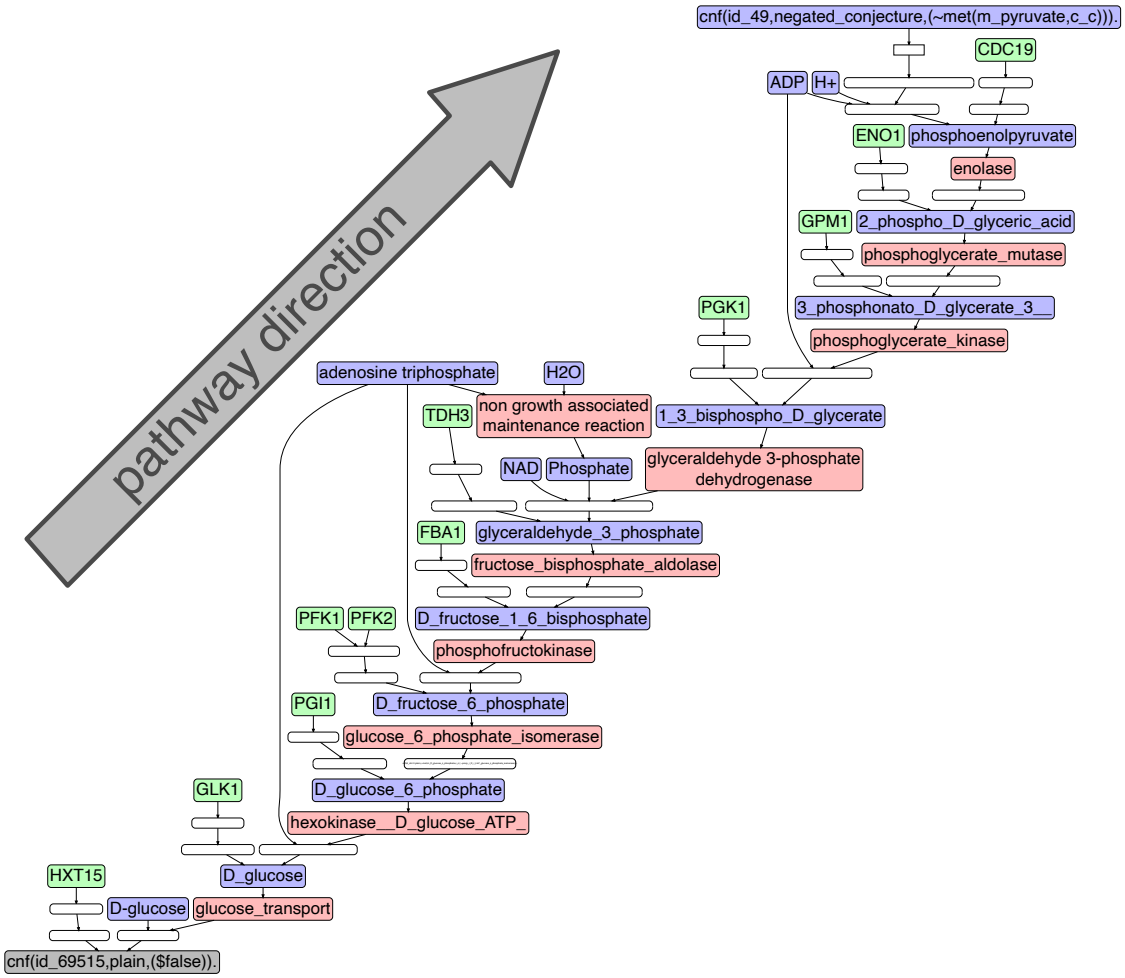


Figure 5.2: Glycolysis pathway

truth data from the literature; and also simulation results from a different deductive approach, FBA (see Section 3.4), but using the same GEM as a background knowledge source.

The single gene deletion prediction showed an F1 score of between 0.237 and 0.342 depending on the background model, which was better than prior qualitative methods. However, an FBA deletion method achieved a higher F1 score, and we conclude that the strictness of the logical model, the lack of quantitative modelling of metabolic reactions, and the lack of axioms for regulatory and signalling effects can explain the relatively low predictive power of the model on this task. For the more specific task of predicting pathways, the model showed good correspondence with the literature, as shown in Fig. 5.2.

Errors in the model are an opportunity for the abduction of hypotheses. We designed an abductive process for improvement of the models, the overall flow of which, including the initial construction of the model from a GEM, is shown in Fig. 5.3. Using the pathways predicted by the model, we present a model-driven experimental design strategy, and demonstrate this with a differential expression study, using the Δpfk2 mutant strain.

The programs used for these experiments were implemented primarily in Bash, Python, and Perl, and iProver (Korovin, 2008), the ATP that was used.

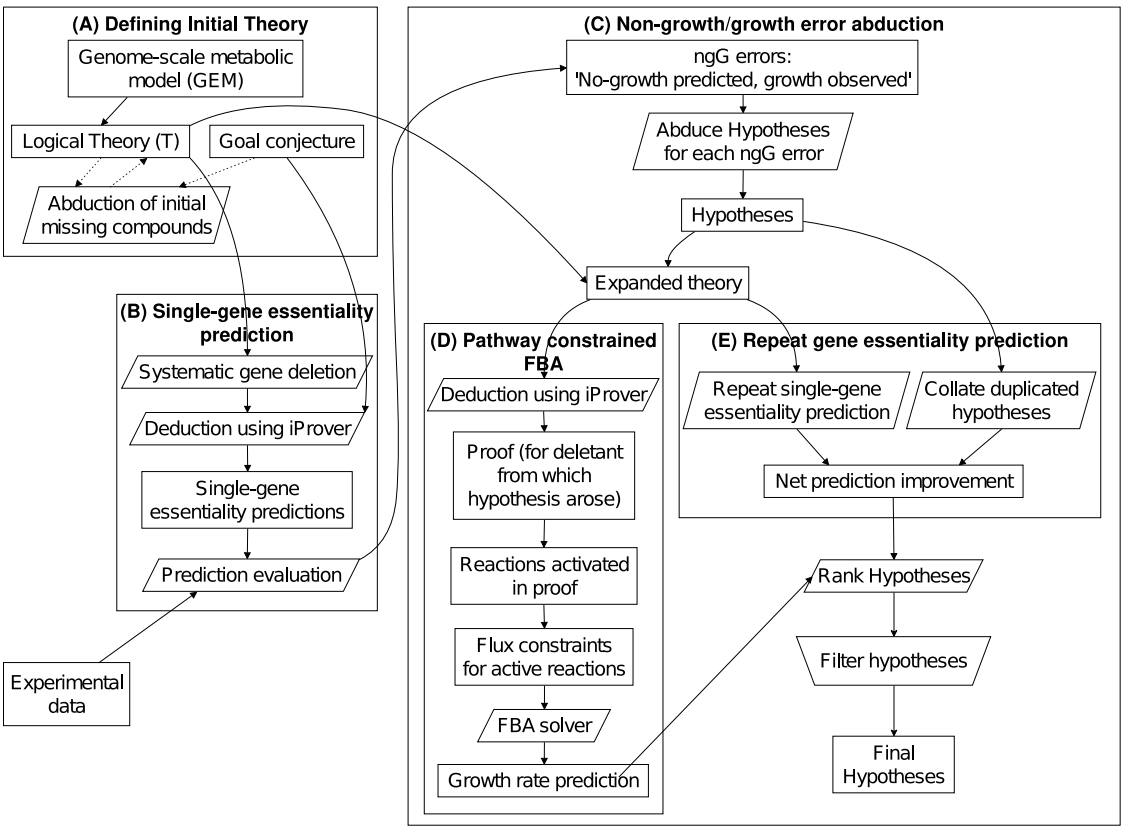


Figure 5.3: Processes in LGEM$^+$. **(A)** defining the logical theory, including abduction of missing compounds to enable viability of base strain; **(B)** single-gene essentiality prediction; **(C)** abduction of hypotheses from *ngG* errors; **(D)** using FBA to assess viability of each hypothesis; and **(E)** repeating single-gene deletion to assess viability of each hypothesis.

We containerised the project using Docker to improve replicability, and also to enable users to reuse the base functions for their own purposes; although we used GEMs constructed for *S. cerevisiae* in this study, this approach could easily be adapted for other organisms given that it is built on the SBML standard. This will also help us to scale this software and integrate it with our robot scientist, Genesis.

Author contributions

The conceptualisation of the project was done by Ross D. King, Konstantin Korovin, and **Alexander H. Gower**. The logical predicate and clause structure were designed by K.K. and **A.H.G.** Code to generate logical theory structures from GEMs was developed and tested by **A.H.G.** Extensions to the ATP iProver to incorporate abduction were developed by K.K. The experiments were designed by R.D.K, Ievgeniia A. Tiukova, K.K., and **A.H.G.**, and executed by **A.H.G.** The microarray expression data analysis was conducted by Erik Y. Bjurström, Praphapan Lasin, and **A.H.G.** The data were prepared and curated by **A.H.G.** Figures were designed and prepared by **A.H.G.** The manuscript was written by **A.H.G.** and K.K, and edited by E.Y.B, Daniel Brunnsåker, and R.D.K. The project was supervised by R.D.K., K.K., and I.A.T. The funding for the project was acquired by R.D.K.

Paper IV: Graph Neural Network based hierarchy-aware Box Embeddings of Knowledge Graphs

Problem

Knowledge graphs (KGs) in biology are often enriched with ontological information. In other words, the schema for the KG is an ontology, which describes the semantics of the domain: the types of entities (classes) involved and the relations between them. The information encoded in these ontologies is an instantiation of part of the domain knowledge, usually curated by experts, and as such contains valuable information that is useful for machine learning on the graph. Many of these ontologies have, in particular, a hierarchical structure for classes using the '`subClassOf`' relation, as organising concepts in this way does not require expertise in logic or ontology engineering.

A type of machine learning that is highly useful when working with KGs is representation learning. There are many programs that do accept graphs as inputs, but many more that don't. Representation learning enables the transformation of KGs into a form that works as an input to a program class that does not accept the KG itself as an input, while encoding useful aspects of the data from the original KG. A form of representation learning that has proven useful for downstream tasks, such as link prediction and property prediction, is embedding KGs into an n-dimensional vector space.

Many KG embedding (KGE) methods are trained primarily on observed triples, using the relational structure of the graph as the main learning signal. When the knowledge graph is built based on ontologies that have rich data about the hierarchy and relations between different entities, we want to capture this ontological information in the embedding. To be able to use this ontological information as a learning signal, the extent to which the current model represents the ontology can be calculated using specialised loss functions, so-called *semantic losses*.

This paper addresses three main research problems.

1. Can GNN based box embeddings learn useful representations of biological knowledge graphs?

2. Can semantic losses be used to train GNN box embeddings in the absence of additional loss terms (e.g. prediction loss)?

3. Can we rank revisions to knowledge graphs based on how much they shift embeddings?

Approach and Contributions

The novel approach of this paper is a method to learn hierarchy-aware box embeddings using GNNs together with semantic losses. The semantic loss is introduced via terms in the optimisation loss. We show that semantic loss can be used on its own to generate KG embeddings adhering to subsumptions defined in ontologies. But more importantly, when semantic losses are used

together with a task-specific prediction loss this leads to improved performance on a biologically relevant prediction task compared to training on the prediction loss alone.

We train concept embeddings as axis-aligned hyperrectangles, or "boxes", defined as the Cartesian product of closed intervals,

$$\mathrm{Box}(\theta) = \prod_{i=1}^{n} [z_i(\theta_i), Z_i(\theta_i)], \tag{5.1}$$

where z_i and Z_i correspond to the lower and upper coordinate along dimension i. We encode boxes with a latent variable vector, $\theta = (\theta_i^z, \theta_i^Z)_{i=1}^n$, using the `MinDeltaBoxTensors` constructor introduced by Chheda et al. (2021), where the upper and lower box coordinates are defined by $z_i = \theta_i^z$ and $Z_i = z_i + \mathrm{softplus}(\theta_i^Z)$. Knowledge graphs are specified in description logic, only using TBox statements by rewriting class assertions, $C(a)$, as $\{a\} \sqsubseteq C$ and role assertions, $r(a, b)$, as $\{a\} \sqsubseteq \exists r.\{b\}$. Having the database facts in the same form as terminological statements from the ontologies allows using the same parameterisation form for all concepts, including the singleton concepts, $\{a\}$, $\{b\}$, etc.

To reach these embeddings, we train the latent variables using a GNN with the architecture derived from the knowledge graph. The network is trained to minimize a loss function that includes semantic losses calculated on each layer. Below are the descriptions of each semantic loss.

$\mathcal{L}_{\sqsubseteq}$**: positive semantic loss** A loss function that rewards full class inclusion, and is applied to subclass axioms in the ontology.

$\mathcal{L}_{\sqsubseteq}^{-}$**: negative semantic loss** A loss function that penalizes partial class inclusion, and is applied to disjointness axioms in the ontology.

Two versions of each loss were tested: one based on the *distance* between two boxes; the other based on the *overlap* of boxes. Detail of their calculation is provided in Section 2 of **Paper IV**.

***S. cerevisiae* gene knowledge graph.** In the case of the knowledge graph describing *S. cerevisiae* genes, built from community databases, the GNN embeddings are used as the input layer of a fully connected neural network to predict digenic deletion fitness (how well strains with two deleted genes grow). The prediction loss is included in the loss function, with ground truth data from Costanzo et al. (2016). The networks are simultaneously trained using the ADAM optimizer to minimize:

$$\mathcal{L} = \mathcal{L}_{\mathrm{MSE}}(y, \hat{y}) + \alpha(\mathcal{L}_{\sqsubseteq} + \beta\mathcal{L}_{\sqsubseteq}^{-}) + \lambda \left\| w \right\|^2 .$$

10-fold cross-validation on the prediction task shows that the model trained with semantic losses is significantly better than that without, and that these approaches both outperform previous embedding methods for the same task.

Learning in the absence of a prediction task. We trained embeddings on the same KG as the digenic fitness task, with some modifications. Negative examples for the negative semantic loss, $\mathcal{L}^{-}_{\sqsubseteq}$, are taken from disjointness axioms from two sources. Firstly, between the three subclasses in the molecular function domain, and between each of these classes and the subclasses of the other two (see Section 4 of **Paper IV** for details). Secondly, for randomly drawn pairs to distinguish between individual classes in the graph.

Again, the network is trained using the ADAM optimizer, this time minimising the loss function:

$$\mathcal{L} = \mathcal{L}_{\sqsubseteq} + \beta\mathcal{L}^{-}_{\sqsubseteq} + \lambda_s R_{\text{small}} + \lambda \left\| w \right\|^2 ,$$

where R_{small} is a regularisation loss which penalises small boxes. The resultant box embeddings, both before and after the GNN, are shown in Fig. 5.4.

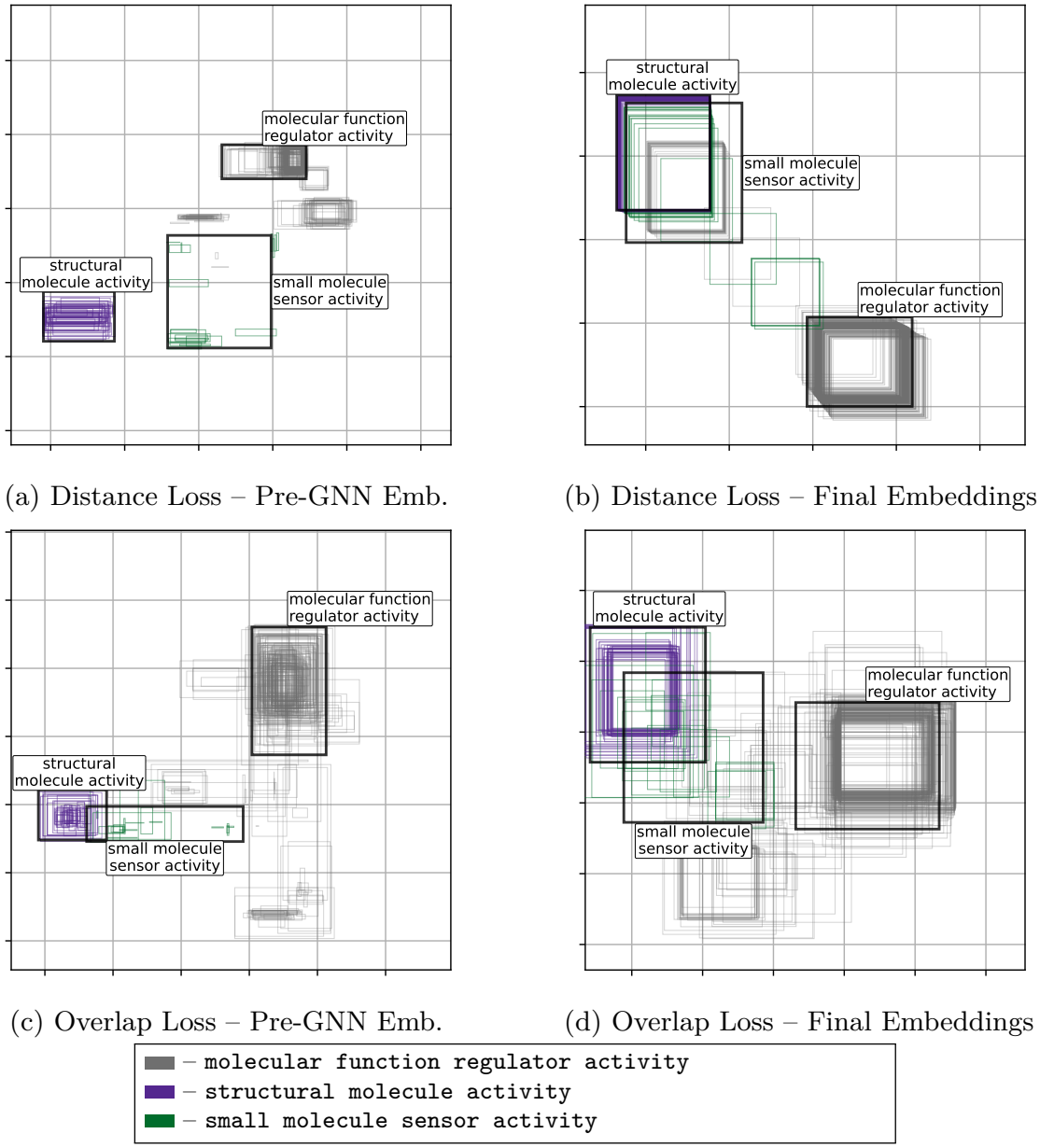
Using box embedding distance to evaluate graph revisions. The box embeddings we trained proved useful for a prediction task in systems biology, and captured structure from the ontologies underlying the knowledge graph. By both measures, predictive accuracy and internal consistency (**Paper I**), the embeddings are a good model of the system of study. This led us to consider using the embeddings to assess hypotheses, and in particular the changes to embeddings that result from a modification to the knowledge graph.

We designed a prototype algorithm for this purpose, that calculates both changes to the semantic losses of the embeddings, and the overall shift in box embeddings themselves, for candidate revisions. We tested this algorithm by training box embeddings on a subset of the overall graph (Using a 70:30 training and test split on graph facts, stratified by relation type), and comparing the distances between true additions and random additions. Random additions were chosen as a cheap proxy in the absence of negative examples (known false facts). Results are shown in Fig. 5.5. Certainly this technique needs refinement, but the initial results from this test are promising, showing significant differences overall between true and random edges. Whether any information can be gained from the absolute change in box embedding distance is unclear, but these preliminary results indicate such measures could be used to rank hypotheses in a robot scientist.

Knowledge graphs are the preferred data structure for systems biology, and many other scientific domains. This work extends inference techniques for knowledge graphs, particularly those with rich ontology structure and onward quantitative prediction tasks. We demonstrated these techniques on hypothesis generation and evaluation tasks. We show that neurosymbolic techniques like this offer a promising direction for scalable and verifiable automated hypothesis generation and evaluation in systems biology.

Notes

This manuscript is an extension of a previously published conference paper (Kronström et al., 2025). The new contributions from this extension were: the



(a) Distance Loss – Pre-GNN Emb.

(b) Distance Loss – Final Embeddings

(c) Overlap Loss – Pre-GNN Emb.

(d) Overlap Loss – Final Embeddings

Figure 5.4: Learned box embeddings in two dimensions for the molecular function domain. 5.4(a) and 5.4(c) show box embeddings prior to input into GNN; 5.4(b) and 5.4(d) show final embeddings for *distance* and *overlap* loss respectively.

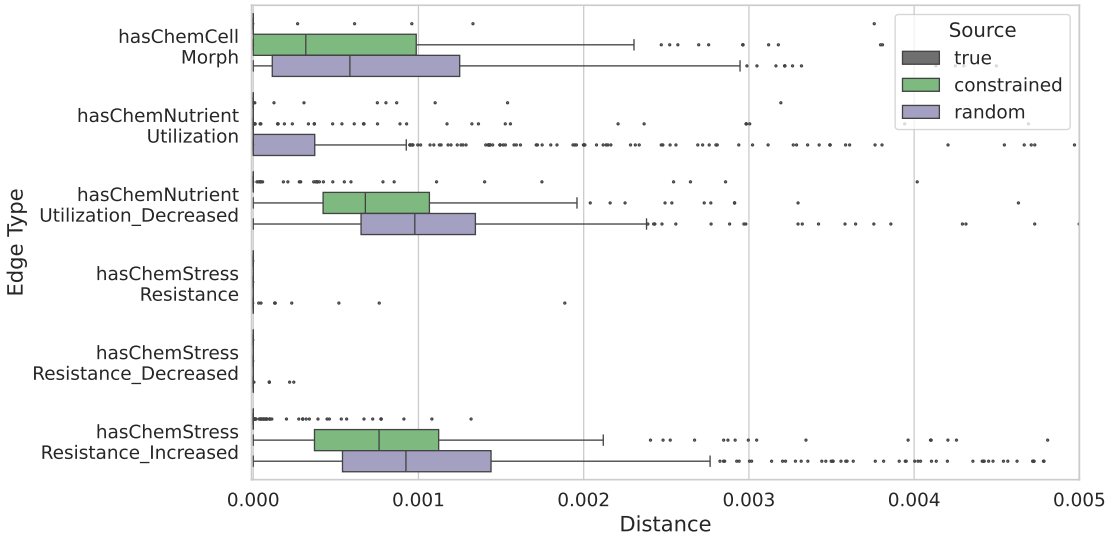


Figure 5.5: Distribution of distances of box embeddings learned from revised graphs $\tilde{\mathcal{G}}$ to the original embeddings learned from $\mathcal{G}_{\text{train}}$, shown by relation type, for a subset of the relation types in the graph. (The method for calculating these differences is described in Section 5.6 of **Paper IV**). For most relations, the distance ranks of randomly drawn edges (constrained to the appropriate class) were significantly different to the ranks of the test edges (p <0.05, two-sided Mann–Whitney U test). Embeddings were learned with inclusion losses.

semantic losses and additional regularisation losses; new training algorithms to embed these losses in the GNN training and ontology preprocessing algorithms; the training example in the absence of a prediction task; and the method for ranking additional links, and its evaluation.

Author contributions

The conceptualisation of the project was done by Filip Kronström, Ross D. King, Daniel Brunnsåker, and **Alexander H. Gower**. F.K. designed the knowledge graph. F.K. and **A.H.G.** designed the machine learning models and the algorithms for embedding distance. D.B. performed and analysed the biological experiment. The project was supervised by R.D.K., and Ievgeniia A. Tiukova and the funding for the project was acquired by R.D.K.

Paper V: Investigating uncharacterised genes in *Saccharomyces cerevisiae* using Robot Scientists

Problem

In biology, we are often faced with the problem of having a rough hypothesis that cannot be immediately tested. This is a very common state for a scientist to be in, and the next step in forming a testable hypothesis is to refine the rough hypothesis. Box et al. (2005) state that given any hypothesis, we necessarily deduct consequences from it to compare against the natural world. These deductions are done using the best available theory of the phenomenon, optionally together with any contextual empirical data that doesn't corrupt the hypothesis test. For example, it is common to constrain simulations in systems biology with empirical data from related experiments, in order to focus the uncertainty in the simulation around the hypothesis.

In this study, we begin from just such a rough hypothesis. We investigate the role of *YGR067C*, an uncharacterised gene in *S. cerevisiae*, that is, a gene whose biological function is unknown. What was known about *YGR067C* at the outset of the study was that its gene product, a protein, contained a zinc finger motif similar to another protein, Adr1p. This protein had been characterised as a respiratory transcription factor. In other words it exerted control over genes important for yeast respiratory function, and was particularly active during the diauxic shift in *S. cerevisiae*[2]. Based on this structural similarity, and previous data from a diauxic shift study (Brunnsåker et al., 2023), we hypothesised that the gene product of *YGR067C* also acted as a transcription factor regulating respiratory genes during the diauxic shift.

The best available theory we have of yeast physiology, as discussed in Chapter 3, is encoded in a community model, in this case the consensus genome-scale metabolic model (GEM) Yeast9 (Zhang et al., 2024). As its function is unknown, *YGR067C* is not included in any of the mathematical models of yeast. So, the steps we took in **Paper III** and **Paper IV** to make predictions based on disruption to genes in the computational model were not an option. We first needed to encode the hypothesis, and once in the model, we could simulate its effect.

Because the diauxic shift is a dynamical phenomenon in yeast, we worked with batch cultivations, which are very sensitive to minor fluctuations in initial conditions and experimental execution. This can complicate statistical inference, so to obtain empirical data to test our predictions we needed to be very precise during experimentation.

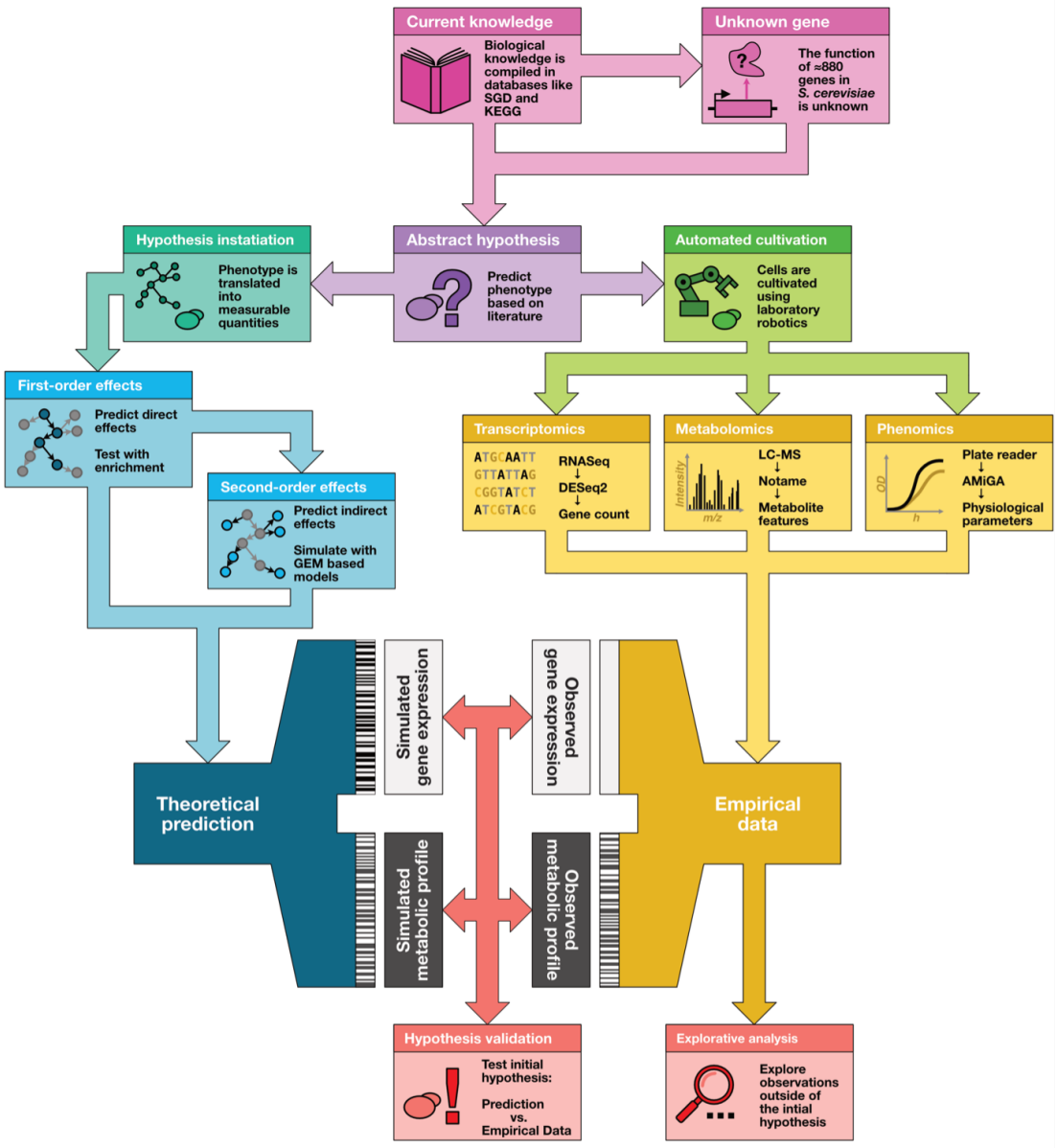


Figure 5.6: Overview of study in **Paper V**. From the abstract hypothesis, deduction was made using several different computational models, and experiments were performed. Simulation and experimentation resulted in data, which after processing, is able to be directly compared to test the hypothesis.

Approach and Contributions

The overall approach we took in this study is summarised in Fig. 5.6. The specifics of the regulation effect of *YGR067C* was not known, and as discussed in Section 3.1.2, there are many different ways genes can be regulated. We tested and compared three different methods of deductive inference from our hypothesis. The first technique was to directly assume a disruption in the respiratory pathways. The second method was to use LGEM$^+$, the logical inference framework developed in **Paper III**. And the third was using a flux balance analysis (FBA) model. For predictions using LGEM$^+$ and FBA simulations, we introduced the effect of the hypothesis by disrupting pathways associated with respiration.

To produce high-quality, consistent data to evaluate these predictions, there were two difficulties to address. Firstly, the most useful empirical data modes were transcriptomics and metabolomics, measurements of gene expression and compound presence respectively; both are subject to problems of noise. Secondly, the diauxic shift can be difficult to reproduce experimentally, and we needed a reliable and reproducible experimental protocol for doing so. To address both of these difficulties, we designed and executed the experiments using the automated laboratory cell, Eve (Williams et al., 2015).

We found that our predictions partially agreed with the experimental data, in particular that there were changes in gene expression in respiratory pathways during the glucose phase. The accuracy of the predictions using the LGEM$^+$ and FBA models was quite low, indicating that the models and the hypothesis instantiation algorithm did not capture the subtleties of the regulation effects. Our results suggest that *YGR067C* is more active during the fermentative glucose phase of growth in the diauxic shift, and we were able to conclude that a more specific formulation of the hypothesis—"*YGR067C* induces ethanol consuming respiratory pathways prior to the diauxic shift"—was consistent with the evidence. This study highlights the need for better ways of translating abstract hypotheses into computational models in systems biology.

Author contributions

Conceptualisation and design of study: Erik Y. Bjurström, Ievgeniia A. Tiukova, and Ross D. King. Hypothesis instantiation and simulation: **Alexander H. Gower**. Cultivation experiment: E.Y.B., **A.H.G.**, and Praphapan Lasin. Multiomics extraction: P.L. and E.Y.B. LC/MS processing: Otto I. Savolainen. Data analysis: E.Y.B. and **A.H.G**. Writing: E.Y.B., **A.H.G.**, O.I.S, I.A.T., and R.D.K. All authors read and approved the final manuscript.

[2]The diauxic shift is a metabolic rewiring event in *S. cerevisiae*, where the organism switches from glucose fermentation to ethanol respiration.

Paper VI: Agentic AI Integrated with Scientific Knowledge: Laboratory Validation in Systems Biology

Problem

The automation of scientific discovery requires AI-robotic systems that can be directed to complex research tasks, execute experiments, collect and analyse data, and design and test hypotheses.

Many applications of AI or robotics in science are for specific parts of a research cycle. In biology labs, for example, robotic liquid handlers are commonly used. The different data modes and fundamental capabilities, as well as different brands, models, and ages of these subsystems, cause a lot of work at the interfaces. Considering the inputs and outputs of these subsystems, we might need to do quite a lot of work to translate between systems. For example in LGEM$^+$ (**Paper III**), the ATP identifies a model for satisfiability of the negated conjecture, which constitutes a broad hypothesis—*"in a $\Delta MET5$ mutant, a lack of methionine production in the cytoplasm leads to an absence of cell growth."* This is not actionable, so in order to design an experiment to test this we would have to analyse the hypothesis in the context of our laboratory and come up with some ideas for experiments. And even if we then had an idea for an experiment, we would have to translate this into an actionable experimental plan before being able to actually execute it, either manually or with robots.

To automate the discovery process from end-to-end, we need to design ways for the subsystems to communicate with one another. In this work, we seek to build a robot scientist that integrates ideas of "agentic AI" systems with structured scientific knowledge: community databases, logic programs, and a hypothesis knowledge graph. The idea was to use the strengths of the respective technologies in the design of the different agents. The central research problems were:

1. How do we formalise hypotheses in such a way that they are
   - as far as possible unambiguous,
   - verifiable with the empirical data we were to collect, and
   - richly defined and grounded in community ontologies.
2. From this hypothesis, how do we generate experimental plans with as little human intervention as possible?
3. How do we execute these plans using automated laboratory systems?
4. How can we ensure the interoperability of the different systems, both internally and with community knowledge?
5. How can we coordinate this system end-to-end?

Approach and contributions

The approach we took was to use the properties of each technology to inform the system design. Starting with the least flexible, the laboratory hardware, we

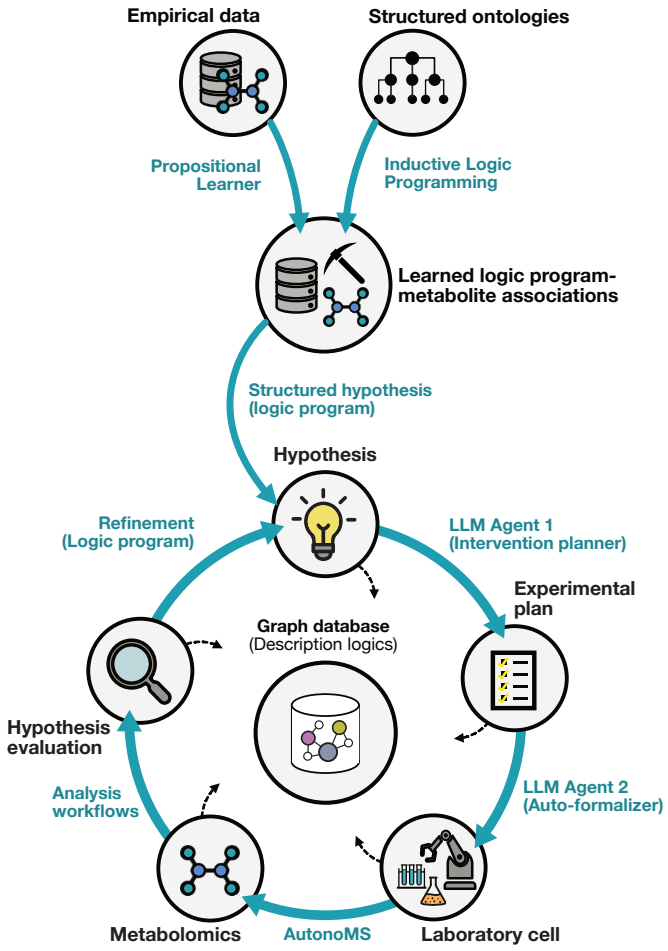


Figure 5.7: An overview of research process that is automated in this study.

defined the classes of experiments we could perform, from which we specified the form of the hypotheses for generation. Hypotheses were of the form

$$\pm\text{amino acid accumulation} \longrightarrow \text{phenotype},$$

where phenotype was a logic program, learned through inductive logic programming (ILP). The forms of the possible logic programs were also specified during the ILP training based on the types of experiments we could perform. If future configurations of experimental hardware and protocols would enable different types of experiments, these structures for hypothesis learning could be altered accordingly.

Designing the agents and workflow

We considered the relative strengths and weaknesses of the different tools at our disposal. For the cognitive tasks, including hypothesis generation, knowledge management, and experiment planning, we would use logic programming, ontologies, and LLMs.

Logic programming is directly interpretable, has a high degree of specificity, and is traceable and transparent. However, it is also rigid, constrained to the formalisms it is provided.

LLMs on the other hand are flexible, plastic, and able to adapt across domains using the fluid nature of language. They are also quick to iterate and prototype with, and many have been trained on formal computer languages, so they can work with these as well as natural language. This lends them well to being used at the interface of systems, as they can take any serialised input, and are also reasonably good at formatting outputs according to request. Among their weaknesses are "hallucinations"—outputs that have no basis in training data—that they are opaque and non-interrogable, verbose, and data- and energy-intensive to train. Additionally, the training data set and regime is often unknown, particularly when working with commercial LLMs.

Ontologies allow for precise definitions of terms, and ensure that data have specific meaning. This allows direct integration with community databases, and rich representations. We can also query databases built using ontologies semantically. However, ontologies require engineering and take time to develop, and there are many design choices to be made; designing ontologies in an extensible way takes time and expert knowledge.

Considering the technologies in this way allowed us to identify whether their weaknesses would cause a problem or not. For example, in the context of selecting scientific experiments, "hallucination" perhaps is not a problem in and of itself. Say, for example, that the model "hallucinates" a fact that is not present in the database, and uses this as the basis for onward reasoning and generates a hypothesis. This doesn't present a particular problem in terms of testing the hypothesis in biology, as our oracle will always be the empirical data.

We used LLMs to bridge the gaps between these different structured data, taking advantage of their inherent flexibility and adaptability to different problems. LLM agents were used, for example, to take a draft experimental

plan in English (itself the output of another LLM agent) (more or less what one might provide to a human scientist to perform the experiment) and transform it to machine instructions for the automated laboratory cell. We found after testing that segmenting tasks to different agents resulted in more reliable performance.

The experimental plans were executed by the automated laboratory cell, and an automated mass spectrometry workflow was used. Human intervention in the research cycle was limited to supervisory tasks (setting objectives, evaluating safety) and menial laboratory tasks (transferring cultivation plates between work stations, filling up stocks and consumables).

Graph database for hypotheses and data

We designed an ontology to record hypotheses about yeast phenotype semantically, and to link hypotheses with the experiments designed to test them and the resulting empirical data. Using this ontology we created a graph database. The database was designed with the particular aim of data reuse—answering questions with the database is preferable to conducting new experiments. As a starting point for the ontology design, we observed that the ontology used to record many of the facts in community knowledge graphs, the Ascomycete Phenotype Ontology (APO), was insufficient to record hypotheses of this type, and so it was necessary to create a hypothesis ontology that could go beyond genotype $\longrightarrow$ phenotype hypotheses, of the sort explored in **Paper III** and **Paper V**, and modelled in the graph database in **Paper IV**.

We present a draft ontology for hypotheses, which has at its core the concept of organism states, that are observed at a particular time. A hypothesis is then that one state 'implies' another, a generalisation of the statements allowed by APO. These states are concepts that are a combination of different observables and genotypes, shown in Fig. 5.8B. Much of the meaning in the hypothesis is contained in the 'implies' relation. In the paper, we discuss this in more detail, but the specific definition of what this term means and, more importantly, how to evaluate its truth will vary depending on the domain. In the case of these hypotheses and experiments, 'state 1 *implies* state 2' is taken to mean that if state 1 is observed, then at some later time state 2 is observed (as shown in Fig. 5.8A).

This ontology allowed for statements representing the hypotheses, and also then (when combined with statements encoding the empirical data) allowed connecting hypotheses with the data from experiments. This enabled the reuse of data to partially assess a new hypothesis about proline and formic acid resistance.

Results

Our system was able to identify novel phenotype–phenotype interactions in *Saccharomyces cerevisiae*, including a partial rescue of growth under formic-acid stress by the amino acid aminoadipate. We were able to reuse data from previous experiments to partially answer a hypothesis, using the semantic database. The LLM-agents proved valuable in providing interface between

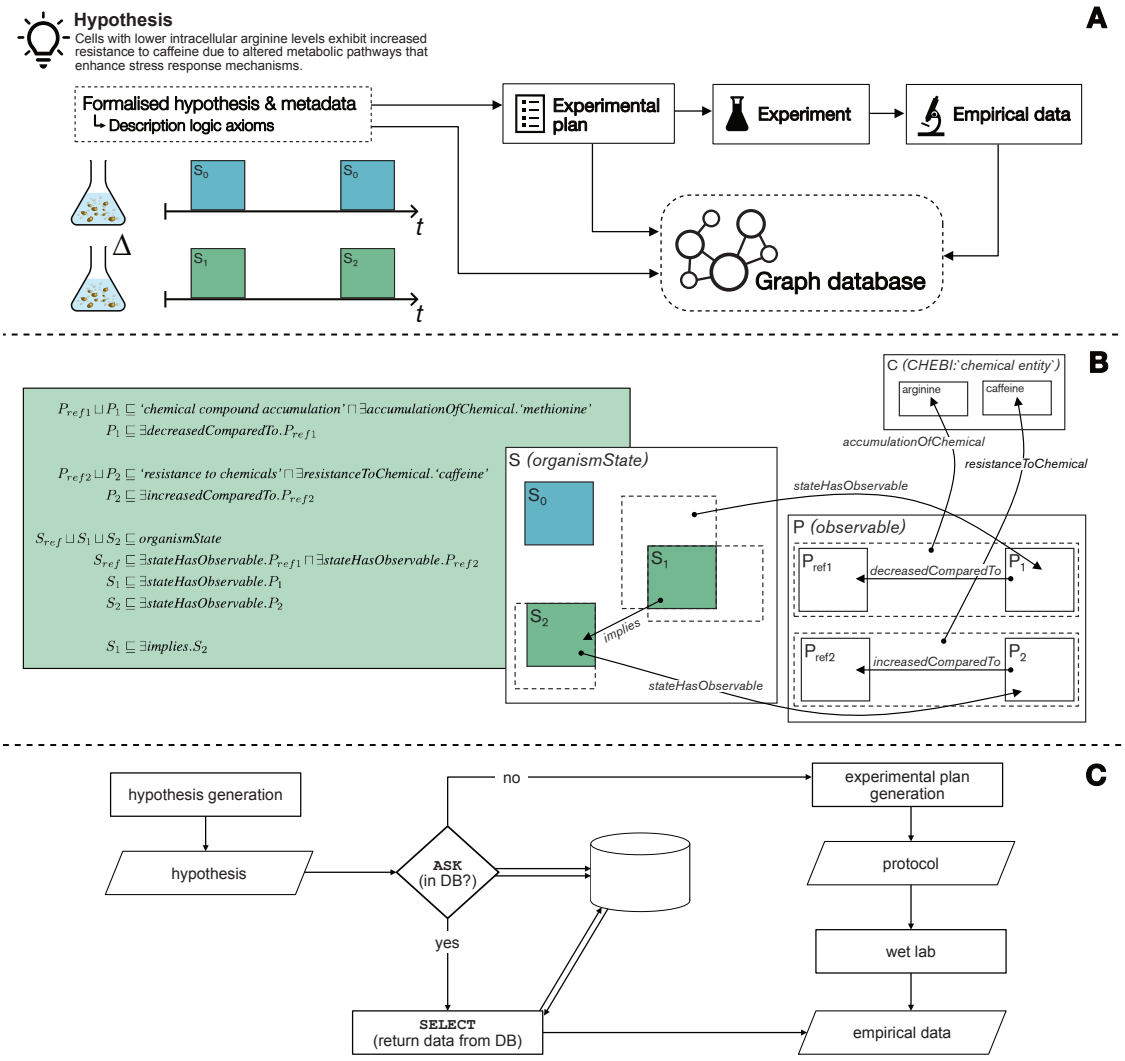


Figure 5.8: **Graph Database and description logics. A.** The hypothesis is formalised into description logic axioms using an extended version of the Ascomycete Phenotype Ontology (APO); the generated experimental plans are stored using terms from the Ontology for Biomedical Investigations (OBI); and empirical data are stored using terms from the Genesis ontology. **B.** *(L)* DL axioms expressing the same hypothesis shown in A; *(R)* a visual representation of the concept inclusions expressed in the statements (with a subset of the relations shown, for clarity). **C.** To make efficient use of data already collected, an `ASK` SPARQL query is executed on the graph database to identify if there are empirical data that can be relevant to the hypothesis; if yes, then a `SELECT` query is executed to return the data, else if no, the full automated experimental generation procedure is followed.

the hypothesis generation steps, and the experimental hardware. We found that a combination of LLMs and symbolic verification was able to generate machine instructions for the experimental robotics, automating a previously high-friction point in the systems biology research process. This work shows the value of designing robot scientists according to a multi-agent framework, and the possibility of retaining the benefits of structured, ontology-based knowledge stores for scientific discovery.

Author contributions

Conceptualization, Daniel Brunnsåker, Ross D. King and **Alexander H. Gower**; Data curation, D.B.; Formal analysis/Visualization/Validation, D.B. and **A.H.G**; Funding acquisition/Supervision, R.D.K. and Ievgeniia A. Tiukova; Investigation, D.B., Prajakta Naval, and Erik Y. Bjurström; Methodology, D.B., **A.H.G**, and P.N.; Project administration, R.D.K., I.A.T., and D.B.; Resources, R.D.K.; Software, D.B., **A.H.G**, and Filip Kronström; Writing - Original draft, D.B., **A.H.G**, and R.D.K.; Writing - review & editing, D.B., **A.H.G**, P.N., E.Y.B, F.K., I.A.T., and R.D.K. All the authors approved the final version of the manuscript.

Bibliography

Alberts, B. (2017, August 7). *Molecular Biology of the Cell* (6th ed.). W.W. Norton & Company. https://doi.org/10.1201/9781315735368 (cit. on p. 19).

Angello, N. H., Rathore, V., Beker, W., Wołos, A., Jira, E. R., Roszak, R., Wu, T. C., Schroeder, C. M., Aspuru-Guzik, A., Grzybowski, B. A., & Burke, M. D. (2022). Closed-loop optimization of general reaction conditions for heteroaryl Suzuki-Miyaura coupling. *Science*, *378*(6618), 399–405. https://doi.org/10.1126/science.adc8743 (cit. on p. 9).

Arnold, C. (2022). Cloud labs: Where robots do the research. *Nature*, *606*(7914), 612–613. https://doi.org/10.1038/d41586-022-01618-x (cit. on p. 14).

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., & Sherlock, G. (2000). Gene Ontology: Tool for the unification of biology. *Nature Genetics*, *25*(1), 25–29. https://doi.org/10.1038/75556 (cit. on p. 22).

Baltimore, D. (1970). Viral RNA-dependent DNA Polymerase: RNA-dependent DNA Polymerase in Virions of RNA Tumour Viruses. *Nature*, *226*(5252), 1209–1211. https://doi.org/10.1038/2261209a0 (cit. on p. 17).

Ben-Ari, M. (2012). *Mathematical Logic for Computer Science*. Springer. https://doi.org/10.1007/978-1-4471-4129-7 (cit. on pp. 27, 28).

Bengio, Y., Goodfellow, I., Courville, A., et al. (2015, October 3). *Deep learning* (Vol. 1). (Cit. on p. 30).

Bjurström, E. Y., Gower, A. H., Lasin, P., Savolainen, O. I., Tiukova, I. A., & King, R. D. (2026). Investigating uncharacterised genes in Saccharomyces cerevisiae using robot scientists. *Scientific Reports*, *16*(1), 10999. https://doi.org/10.1038/s41598-026-46236-z (cit. on p. 31).

Box, G. E. P., Hunter, J. S., & Hunter, W. G. (2005, January 1). *Statistics for experimenters : Design, innovation and discovery / George E.P. Box, J. Stuart Hunter, William G. Hunter.* (2. ed.). Wiley-Interscience. Retrieved December 15, 2025, from https://research.ebsco.com/linkprocessor/plink?id=29a558e5-8aae-3ccf-be6a-284b3981c812 (cit. on p. 47).

Brunnsåker, D., Gower, A. H., Naval, P., Bjurström, E. Y., Kronström, F., Tiukova, I. A., & King, R. D. (2025, August 18). *Agentic AI Integrated*

with Scientific Knowledge: Laboratory Validation in Systems Biology. https://doi.org/10.1101/2025.06.24.661378 (cit. on p. 31).

Brunnsåker, D., Reder, G. K., Soni, N. K., Savolainen, O. I., Gower, A. H., Tiukova, I. A., & King, R. D. (2023). High-throughput metabolomics for the design and validation of a diauxic shift model. *npj Systems Biology and Applications*, *9*(1), 1–9. https://doi.org/10.1038/s41540-023-00274-9 (cit. on pp. 14, 47).

Buchanan, B. G., & Feigenbaum, E. A. (1981). Dendral and meta-dendral: Their applications dimension. In B. L. Webber & N. J. Nilsson (Eds.), *Readings in artificial intelligence* (pp. 313–322). Morgan Kaufmann. https://doi.org/10.1016/B978-0-934613-03-3.50026-X (cit. on p. 9).

Burger, B., Maffettone, P. M., Gusev, V. V., Aitchison, C. M., Bai, Y., Wang, X., Li, X., Alston, B. M., Li, B., Clowes, R., Rankin, N., Harris, B., Sprick, R. S., & Cooper, A. I. (2020). A mobile robotic chemist. *Nature*, *583*(7815), 237–241. https://doi.org/10.1038/s41586-020-2442-2 (cit. on pp. 9, 14).

Carthew, R. W. (2021). Gene Regulation and Cellular Metabolism: An Essential Partnership. *Trends in genetics : TIG*, *37*(4), 389–400. https://doi.org/10.1016/j.tig.2020.09.018 (cit. on p. 24).

Castiglione, F., Pappalardo, F., Bianca, C., Russo, G., & Motta, S. (2014). Modeling Biology Spanning Different Scales: An Open Challenge. *BioMed Research International*, *2014*, 902545. https://doi.org/10.1155/2014/902545 (cit. on p. 24).

Chen, Y., Li, F., & Nielsen, J. (2022). Genome-scale modeling of yeast metabolism: Retrospectives and perspectives. *FEMS Yeast Research*, *22*(1), foac003. https://doi.org/10.1093/femsyr/foac003 (cit. on p. 23).

Chheda, T., Goyal, P., Tran, T., Patel, D., Boratko, M., Dasgupta, S. S., & McCallum, A. (2021, September 10). *Box Embeddings: An open-source library for representation learning using geometric structures*. arXiv: 2109.04997 [cs]. https://doi.org/10.48550/arXiv.2109.04997 (cit. on p. 42).

Chowdhary, K. R. (2020). Natural Language Processing. In K. Chowdhary (Ed.), *Fundamentals of Artificial Intelligence* (pp. 603–649). Springer India. https://doi.org/10.1007/978-81-322-3972-7_19 (cit. on p. 30).

COMBINE. (n.d.). *Combine: Coordinating standards for modeling in Biology*. Combine: Coordinating standards for modeling in Biology. Retrieved May 8, 2026, from https://co.mbine.org/ (cit. on p. 21).

Costanzo, M., VanderSluis, B., Koch, E. N., Baryshnikova, A., Pons, C., Tan, G., Wang, W., Usaj, M., Hanchard, J., Lee, S. D., Pelechano, V., Styles, E. B., Billmann, M., Van Leeuwen, J., Van Dyk, N., Lin, Z.-Y., Kuzmin, E., Nelson, J., Piotrowski, J. S., … Boone, C. (2016). A global genetic interaction network maps a wiring diagram of cellular function. *Science*, *353*(6306), aaf1420. https://doi.org/10.1126/science.aaf1420 (cit. on p. 42).

Crick, F. (1958). On protein synthesis. Symposium of the society for experimental biology XII. *New York: Academic Press*. (Cit. on p. 17).

Crick, F. (1970). Central Dogma of Molecular Biology. *Nature*, *227*(5258), 561–563. https://doi.org/10.1038/227561a0 (cit. on p. 17).

Cruz, D. A., & Kemp, M. L. (2021). Hybrid computational modeling methods for systems biology. *Progress in Biomedical Engineering*, *4*(1), 012002. https://doi.org/10.1088/2516-1091/ac2cdf (cit. on p. 25).

Engel, S. R., Aleksander, S., Nash, R. S., Wong, E. D., Weng, S., Miyasato, S. R., Sherlock, G., & Cherry, J. M. (2025). Saccharomyces Genome Database: Advances in genome annotation, expanded biochemical pathways, and other key enhancements. *Genetics*, *229*(3), iyae185. https://doi.org/10.1093/genetics/iyae185 (cit. on pp. 21, 22).

Fersht, A. R. (2008). From the first protein structures to our current knowledge of protein folding: Delights and scepticisms. *Nature Reviews Molecular Cell Biology*, *9*(8), 650–654. https://doi.org/10.1038/nrm2446 (cit. on p. 8).

Frank, A. G., Dalenogare, L. S., & Ayala, N. F. (2019). Industry 4.0 technologies: Implementation patterns in manufacturing companies. *International Journal of Production Economics*, *210*, 15–26. https://doi.org/10.1016/j.ijpe.2019.01.004 (cit. on p. 10).

García Sánchez, C. E., & Torres Sáez, R. G. (2014). Comparison and analysis of objective functions in flux balance analysis. *Biotechnology Progress*, *30*(5), 985–991. https://doi.org/10.1002/btpr.1949 (cit. on p. 24).

Gauch, H. G. (2003). *Scientific method in practice*. Cambridge University Press. (Cit. on p. 7).

Georgiou, P., Gleiss, B., Bhayat, A., Rawson, M., Kovács, L., & Reger, G. (2022). The rapid software verification framework. *FMCAD*, 255–260 (cit. on p. 29).

Giza, P. (2022). Automated discovery systems, part 1: Historical origins, main research programs, and methodological foundations. *Philosophy Compass*, *17*(1), e12800. https://doi.org/10.1111/phc3.12800 (cit. on p. 9).

Goel, S., & Ray, S. (2022). Microprocessor Assurance and the Role of Theorem Proving. In A. Chattopadhyay (Ed.), *Handbook of Computer Architecture* (pp. 1–43). Springer Nature. https://doi.org/10.1007/978-981-15-6401-7_38-1 (cit. on p. 29).

Goffeau, A., Barrell, B. G., Bussey, H., Davis, R. W., Dujon, B., Feldmann, H., Galibert, F., Hoheisel, J. D., Jacq, C., Johnston, M., Louis, E. J., Mewes, H. W., Murakami, Y., Philippsen, P., Tettelin, H., & Oliver, S. G. (1996). Life with 6000 Genes. *Science*, *274*(5287), 546–567. https://doi.org/10.1126/science.274.5287.546 (cit. on p. 20).

Gower, A. H., Korovin, K., Brunnsåker, D., Kronström, F., Reder, G. K., Tiukova, I. A., Reiserer, R. S., Wikswo, J. P., & King, R. D. (2026, February 5). *The Use of AI-Robotic Systems for Scientific Discovery*. arXiv: 2406.17835 [cs]. https://doi.org/10.48550/arXiv.2406.17835 (cit. on pp. 5, 31).

Gower, A. H., Korovin, K., Brunnsåker, D., Tiukova, I. A., & King, R. D. (2023). LGEM : A First-Order Logic Framework for Automated Improvement of Metabolic Network Models Through Abduction. In A. Bifet, A. C.

Lorena, R. P. Ribeiro, J. Gama & P. H. Abreu (Eds.), *Discovery Science* (pp. 628–643, Vol. 14276). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-45275-8_42 (cit. on pp. 31, 37).

Greer, E. L., & Brunet, A. (2008). Signaling networks in aging. *Journal of Cell Science*, *121*(4), 407–412. https://doi.org/10.1242/jcs.021519 (cit. on p. 19).

Hajkowicz, S., Sanderson, C., Karimi, S., Bratanova, A., & Naughtin, C. (2023). Artificial intelligence adoption in the physical sciences, natural sciences, life sciences, social sciences and the arts and humanities: A bibliometric analysis of research publications from 1960-2021. *Technology in Society*, *74*, 102260. https://doi.org/10.1016/j.techsoc.2023.102260 (cit. on p. 8).

Harper, D. (n.d.). *Etymology of Hypothesis*. Online Etymology Dictionary. Retrieved February 14, 2026, from https://www.etymonline.com/word/hypothesis (cit. on p. 12).

Hey, T., & Trefethen, A. (2020). The Fourth Paradigm 10 Years On. *Informatik Spektrum*, *42*(6), 441–447. https://doi.org/10.1007/s00287-019-01215-9 (cit. on p. 8).

Jackson, R., Matentzoglu, N., Overton, J. A., Vita, R., Balhoff, J. P., Buttigieg, P. L., Carbon, S., Courtot, M., Diehl, A. D., Dooley, D. M., Duncan, W. D., Harris, N. L., Haendel, M. A., Lewis, S. E., Natale, D. A., Osumi-Sutherland, D., Ruttenberg, A., Schriml, L. M., Smith, B., … Peters, B. (2021). OBO Foundry in 2021: Operationalizing open data principles to evaluate ontologies. *Database*, *2021*, baab069. https://doi.org/10.1093/database/baab069 (cit. on p. 21).

Karabegović, I., Turmanidze, R., & Dašić, P. (2020). Robotics and Automation as a Foundation of the Fourth Industrial Revolution - Industry 4.0. In V. Tonkonogyi, V. Ivanov, J. Trojanowska, G. Oborskyi, M. Edl, I. Kuric, I. Pavlenko & P. Dasic (Eds.), *Advanced Manufacturing Processes* (pp. 128–136). Springer International Publishing. https://doi.org/10.1007/978-3-030-40724-7_13 (cit. on p. 10).

Khasidashvili, Z., Korovin, K., & Tsarkov, D. (2015). EPR-based k-induction with counterexample guided abstraction refinement. In G. Gottlob, G. Sutcliffe & A. Voronkov (Eds.), *Global conference on artificial intelligence, GCAI 2015, tbilisi, georgia, october 16-19, 2015* (pp. 137–150, Vol. 36). EasyChair. https://doi.org/10.29007/SCV7 (cit. on p. 29).

King, R. D., Rowland, J., Oliver, S. G., Young, M., Aubrey, W., Byrne, E., Liakata, M., Markham, M., Pir, P., Soldatova, L. N., Sparkes, A., Whelan, K. E., & Clare, A. (2009). The automation of science. *Science*, *324*(5923). https://doi.org/10.1126/science.1165620 (cit. on pp. 9, 14, 16, 34).

Klemm, S. L., Shipony, Z., & Greenleaf, W. J. (2019). Chromatin accessibility and the regulatory epigenome. *Nature Reviews Genetics*, *20*(4), 207–220. https://doi.org/10.1038/s41576-018-0089-8 (cit. on p. 19).

Kohl, P., Crampin, E. J., Quinn, T. A., & Noble, D. (2010). Systems Biology: An Approach. *Clinical Pharmacology & Therapeutics*, *88*(1), 25–33. https://doi.org/10.1038/clpt.2010.92 (cit. on pp. 3, 17).

Korovin, K. (2008). iProver – An Instantiation-Based Theorem Prover for First-Order Logic (System Description). In A. Armando, P. Baumgartner & G. Dowek (Eds.), *Automated Reasoning* (pp. 292–298, Vol. 5195). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-71070-7_24 (cit. on pp. 29, 38).

Krebs, E. G., & Graves, J. D. (2000). Interactions between protein kinases and proteases in cellular signaling and regulation. *Advances in Enzyme Regulation*, *40*(1), 441–470. https://doi.org/10.1016/S0065-2571(99)00030-8 (cit. on p. 18).

Kronstrom, F., Gower, A. H., Brunns, D., Tiukova, I. A., & King, R. D. (2026, January). *Graph Neural Network based Hierarchy-Aware Embeddings of Knowledge Graphs: Applications to Yeast Phenotype Prediction*. (Cit. on p. 31).

Kronström, F., Brunnsåker, D., Tiukova, I. A., & King, R. D. (2025). Ontology-based box embeddings and knowledge graphs for predicting phenotypic traits in Saccharomyces cerevisiae. *Proceedings of The 19th International Conference on Neurosymbolic Learning and Reasoning*, 891–912. Retrieved March 17, 2026, from https://proceedings.mlr.press/v284/kronstrom25a.html (cit. on p. 43).

Kronström, F., Gower, A. H., Tiukova, I. A., & King, R. D. (2023). RIMBO - An Ontology for Model Revision Databases. In A. Bifet, A. C. Lorena, R. P. Ribeiro, J. Gama & P. H. Abreu (Eds.), *Discovery Science* (pp. 523–534). Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-45275-8_35 (cit. on p. 31).

Kuhn, T. S. (1970). *The structure of scientific revolutions* ([2d ed., enl). University of Chicago Press. (Cit. on p. 8).

Langley, P. (1979). Rediscovering physics with BACON.3. *Proceedings of the 6th International Joint Conference on Artificial Intelligence - Volume 1*, 505–507 (cit. on p. 9).

Langley, P., & Zytkow, J. M. (1989). Data-driven approaches to empirical discovery. *Artificial Intelligence*, *40*(1), 283–312. https://doi.org/10.1016/0004-3702(89)90051-9 (cit. on p. 9).

Li, F., Yuan, L., Lu, H., Li, G., Chen, Y., Engqvist, M. K. M., Kerkhoven, E. r. J., & Nielsen, J. (2022). Deep learning-based kcat prediction enables improved enzyme-constrained model reconstruction. *Nature Catalysis*, *5*(8), 662–672. https://doi.org/10.1038/s41929-022-00798-z (cit. on p. 24).

Lindsay, R. K., Buchanan, B. G., Feigenbaum, E. A., & Lederberg, J. (1993). DENDRAL: A case study of the first expert system for scientific hypothesis formation. *Artificial Intelligence*, *61*(2), 209–261. https://doi.org/10.1016/0004-3702(93)90068-M (cit. on p. 10).

Ma, S., Kemmeren, P., Aliferis, C. F., & Statnikov, A. (2016). An Evaluation of Active Learning Causal Discovery Methods for Reverse-Engineering

Local Causal Pathways of Gene Regulation. *Scientific Reports*, *6*(1), 22558. https://doi.org/10.1038/srep22558 (cit. on p. 19).

Maeda, K., Hatae, A., Sakai, Y., Boogerd, F. C., & Kurata, H. (2022). MLAGO: Machine learning-aided global optimization for Michaelis constant estimation of kinetic modeling. *BMC Bioinformatics*, *23*(1), 455. https://doi.org/10.1186/s12859-022-05009-x (cit. on p. 24).

Mahadevan, R., & Schilling, C. (2003). The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. *Metabolic Engineering*, *5*(4), 264–276. https://doi.org/10.1016/j.ymben.2003.09.002 (cit. on p. 24).

Mahadevan, R., Edwards, J. S., & Doyle, F. J. (2002). Dynamic Flux Balance Analysis of Diauxic Growth in Escherichia coli. *Biophysical Journal*, *83*(3), 1331–1340. https://doi.org/10.1016/S0006-3495(02)73903-9 (cit. on p. 24).

Malik, A., Arsalan, M., Moreno, C., Mosquera, J., Félix, E., Kizilören, T., Muthukrishnan, V., Zdrazil, B., Leach, A. R., & O'Boyle, N. M. (2026). ChEBI: Re-engineered for a sustainable future. *Nucleic Acids Research*, *54*(D1), D1768–D1778. https://doi.org/10.1093/nar/gkaf1271 (cit. on p. 22).

Mungall, C., Osumi-Sutherland, D., Overton, J. A., Balhoff, J., Clare72, pgaudet, Brush, M., Matentzoglu, N., Touré, V., Dooley, D., Sinclair, M., Bretaudeau, A., Cain, S., Haendel, M., diatomsRcool, Hammock, J., Laporte, M.-A., Jensen, M., & Larralde, M. (2020, July 21). *Oborel/obo-relations: 2020-07-21*. https://doi.org/10.5281/zenodo.3955125 (cit. on p. 21).

Musslick, S., Bartlett, L. K., Chandramouli, S. H., Dubova, M., Gobet, F., Griffiths, T. L., Hullman, J., King, R. D., Kutz, J. N., Lucas, C. G., Mahesh, S., Pestilli, F., Sloman, S. J., & Holmes, W. R. (2025). Automating the practice of science: Opportunities, challenges, and implications. *Proceedings of the National Academy of Sciences*, *122*(5), e2401238121. https://doi.org/10.1073/pnas.2401238121 (cit. on p. 8).

OECD. (2023, June 26). *Artificial Intelligence in Science: Challenges, Opportunities and the Future of Research*. OECD Publishing. Paris. https://doi.org/10.1787/a8d820bd-en (cit. on p. 8).

Oliver, S. G. (2022). From Petri Plates to Petri Nets, a revolution in yeast biology. *FEMS Yeast Research*, *22*(1), foac008. https://doi.org/10.1093/femsyr/foac008 (cit. on p. 20).

Orth, J. D., Thiele, I., & Palsson, B. Ø. (2010). What is flux balance analysis? *Nature Biotechnology*, *28*(3), 245–248. https://doi.org/10.1038/nbt.1614 (cit. on p. 24).

Österberg, L., Domenzain, I., Münch, J., Nielsen, J., Hohmann, S., & Cvijovic, M. (2020). A novel yeast hybrid modeling framework integrating Boolean and enzyme-constrained networks enables exploration of the interplay between signaling and metabolism. *bioRxiv*, 2020.09.11.290817. https://doi.org/10.1101/2020.09.11.290817 (cit. on p. 25).

Piccialli, F., Chiaro, D., Sarwar, S., Cerciello, D., Qi, P., & Mele, V. (2025). AgentAI: A comprehensive survey on autonomous agents in distributed

AI for industry 4.0. *Expert Systems with Applications*, *291*, 128404. https://doi.org/10.1016/j.eswa.2025.128404 (cit. on p. 10).

Rasmussen, S. G. F., Choi, H.-J., Rosenbaum, D. M., Kobilka, T. S., Thian, F. S., Edwards, P. C., Burghammer, M., Ratnala, V. R. P., Sanishvili, R., Fischetti, R. F., Schertler, G. F. X., Weis, W. I., & Kobilka, B. K. (2007). Crystal structure of the human 2 adrenergic G-protein-coupled receptor. *Nature*, *450*(7168), 383–387. https://doi.org/10.1038/nature06325 (cit. on p. 19).

Rasmussen, S. G. F., DeVree, B. T., Zou, Y., Kruse, A. C., Chung, K. Y., Kobilka, T. S., Thian, F. S., Chae, P. S., Pardon, E., Calinski, D., Mathiesen, J. M., Shah, S. T. A., Lyons, J. A., Caffrey, M., Gellman, S. H., Steyaert, J., Skiniotis, G., Weis, W. I., Sunahara, R. K., & Kobilka, B. K. (2011). Crystal structure of the 2 adrenergic receptor–Gs protein complex. *Nature*, *477*(7366), 549–555. https://doi.org/10.1038/nature10361 (cit. on p. 19).

Reder, G. K., Bjurström, E. Y., Brunnsåker, D., Kronström, F., Lasin, P., Tiukova, I., Savolainen, O. I., Dodds, J. N., May, J. C., Wikswo, J. P., McLean, J. A., & King, R. D. (2024). AutonoMS: Automated Ion Mobility Metabolomic Fingerprinting. *Journal of the American Society for Mass Spectrometry*, *35*(3), 542–550. https://doi.org/10.1021/jasms.3c00396 (cit. on p. 14).

Resat, H., Petzold, L., & Pettigrew, M. F. (2009). Kinetic Modeling of Biological Systems. In R. Ireton, K. Montgomery, R. Bumgarner, R. Samudrala & J. McDermott (Eds.), *Computational Systems Biology* (pp. 311–335). Humana Press. https://doi.org/10.1007/978-1-59745-243-4_14 (cit. on p. 24).

Sánchez, B. J., Zhang, C., Nilsson, A., Lahtvee, P.-J., Kerkhoven, E. J., & Nielsen, J. (2017). Improving the phenotype predictions of a yeast genome-scale metabolic model by incorporating enzymatic constraints. *Molecular Systems Biology*, *13*(8), 935. https://doi.org/10.15252/msb.20167411 (cit. on p. 25).

Schindler, S. (2022). Theoretical Virtues: Do Scientists Think What Philosophers Think They Ought to Think? *Philosophy of Science*, *89*(3), 542–564. https://doi.org/10.1017/psa.2021.40 (cit. on p. 7).

Segrè, D., Vitkup, D., & Church, G. M. (2002). Analysis of optimality in natural and perturbed metabolic networks. *Proceedings of the National Academy of Sciences*, *99*(23), 15112–15117. https://doi.org/10.1073/pnas.232349399 (cit. on p. 24).

Seifrid, M., Pollice, R., Aguilar-Granda, A., Morgan Chan, Z., Hotta, K., Ser, C. T., Vestfrid, J., Wu, T. C., & Aspuru-Guzik, A. (2022). Autonomous Chemical Experiments: Challenges and Perspectives on Establishing a Self-Driving Lab. *Accounts of Chemical Research*, *55*(17), 2454–2466. https://doi.org/10.1021/acs.accounts.2c00220 (cit. on p. 11).

Shlomi, T., Berkman, O., & Ruppin, E. (2005). Regulatory on/off minimization of metabolic flux changes after genetic perturbations. *Proceedings of the National Academy of Sciences*, *102*(21), 7695–7700. https://doi.org/10.1073/pnas.0406346102 (cit. on p. 24).

Southern, J., Pitt-Francis, J., Whiteley, J., Stokeley, D., Kobashi, H., Nobes, R., Kadooka, Y., & Gavaghan, D. (2008). Multi-scale computational modelling in biology and physiology. *Progress in Biophysics and Molecular Biology*, *96*(1), 60–89. https://doi.org/10.1016/j.pbiomolbio.2007.07.019 (cit. on p. 24).

Temin, H. M., & Mizutani, S. (1970). Viral RNA-dependent DNA Polymerase: RNA-dependent DNA Polymerase in Virions of Rous Sarcoma Virus. *Nature*, *226*(5252), 1211–1213. https://doi.org/10.1038/2261211a0 (cit. on p. 17).

The Gene Ontology Consortium. (2026). The Gene Ontology knowledgebase in 2026. *Nucleic Acids Research*, *54*(D1), D1779–D1792. https://doi.org/10.1093/nar/gkaf1292 (cit. on p. 22).

Thompson, W. H., & Skau, S. (2023). On the scope of scientific hypotheses. *Royal Society Open Science*, *10*(8), 230607. https://doi.org/10.1098/rsos.230607 (cit. on p. 12).

Urban, J., & Vyskočil, J. (2013). Theorem Proving in Large Formal Mathematics as an Emerging AI Field. In M. P. Bonacina & M. E. Stickel (Eds.), *Automated Reasoning and Mathematics: Essays in Memory of William W. McCune* (pp. 240–257). Springer. https://doi.org/10.1007/978-3-642-36675-8_13 (cit. on p. 29).

Van Noorden, R., & Perkel, J. M. (2023). AI and science: What 1,600 researchers think. *Nature*, *621*(7980), 672–675. https://doi.org/10.1038/d41586-023-02980-0 (cit. on p. 8)
Bandiera_abtest: a
Cg_type: News Feature
Subject_term: Machine learning, Mathematics and computing, Technology, Computer science.

van Gulik, W. M., & Heijnen, J. J. (1995). A metabolic network stoichiometry analysis of microbial growth and product formation. *Biotechnology and Bioengineering*, *48*(6), 681–698. https://doi.org/10.1002/bit.260480617 (cit. on p. 22).

Weisberg, E., Manley, P. W., Breitenstein, W., Brüggen, J., Cowan-Jacob, S. W., Ray, A., Huntly, B., Fabbro, D., Fendrich, G., Hall-Meyers, E., Kung, A. L., Mestan, J., Daley, G. Q., Callahan, L., Catley, L., Cavazza, C., Mohammed, A., Neuberg, D., Wright, R. D., … Griffin, J. D. (2005). Characterization of AMN107, a selective inhibitor of native and mutant Bcr-Abl. *Cancer Cell*, *7*(2), 129–141. https://doi.org/10.1016/j.ccr.2005.01.007 (cit. on p. 19).

Williams, K., Bilsland, E., Sparkes, A., Aubrey, W., Young, M., Soldatova, L. N., De Grave, K., Ramon, J., de Clare, M., Sirawaraporn, W., Oliver, S. G., & King, R. D. (2015). Cheaper faster drug development validated by the repositioning of drugs against neglected tropical diseases. *Journal of the Royal Society, Interface*, *12*(104), 20141289. https://doi.org/10.1098/rsif.2014.1289 (cit. on pp. 9, 14–16, 34, 49).

Winther, R. G. (2015). The Structure of Scientific Theories. Retrieved March 31, 2026, from https://plato.stanford.edu/entrieS/structure-scientific-

theories/ (cit. on p. 15)
Last Modified: 2020-12-29.

Zhang, C., Sánchez, B. J., Li, F., Eiden, C. W. Q., Scott, W. T., Liebal, U. W., Blank, L. M., Mengers, H. G., Anton, M., Rangel, A. T., Mendoza, S. N., Zhang, L., Nielsen, J., Lu, H., & Kerkhoven, E. J. (2024). Yeast9: A consensus genome-scale metabolic model for S. cerevisiae curated by the community. *Molecular Systems Biology*, *20*(10), 1134–1150. https://doi.org/10.1038/s44320-024-00060-7 (cit. on pp. 23, 35, 47).