PHILOSOPHICAL TRANSACTIONS A

royalsocietypublishing.org/rsta

Opinion piece

Cite this article: Battleday R, Gershman S. 2026 Artificial intelligence for science: The easy and hard problems. *Phil. Trans. R. Soc. A* **384**: 20240530.
https://doi.org/10.1098/rsta.2024.0530

Received: 16 December 2024
Accepted: 25 July 2025

One contribution of 18 to a theme issue 'World models in natural and artificial intelligence'.

Subject Areas:
artificial intelligence

Keywords:
discovery, inference, problem solving, science, cognitive science, artificial intelligence, generative AI

Author for correspondence:
Ruairidh Battleday
e-mail: battleday@g.harvard.edu

Artificial intelligence for science: The easy and hard problems

Ruairidh Battleday[1,2] and Sam Gershman[1,2,3]

[1]Department of Psychology, [2]Center for Brain Science, and [3]Kempner Institute for the Study of Natural and Artificial Intelligence, Harvard University, Cambridge, MA, USA

RB, 0000-0003-0292-1674; SG, 0000-0002-6546-3298

A suite of impressive scientific discoveries has been driven by recent advances in artificial intelligence. These almost all result from training flexible algorithms to solve difficult optimization problems specified in advance by teams of domain scientists and engineers with access to large amounts of data. Although extremely useful, this kind of problem solving only corresponds to one part of science—the 'easy problem'. The other part of scientific research is coming up with the problem itself—the 'hard problem'. Solving the hard problem is beyond the capacities of current algorithms for scientific discovery because it requires continual conceptual revision based on poorly defined constraints. We can make progress on understanding how humans solve the hard problem by studying the cognitive science of scientists and then use the results to design new computational agents that automatically infer and update their scientific paradigms.

This article is part of the theme issue 'World models in natural and artificial intelligence'.

1. The easy problem

Most work applying artificial intelligence (AI) to science has focused on what might be called the 'easy problem'. This is a relative term, since the easy problem is actually quite hard. A scientist specifies a function that they want to optimize (e.g. a function that generates a protein's structure given its amino-acid sequence). Included in the specification is the input for the function (e.g. the

THE ROYAL SOCIETY PUBLISHING

amino-acid sequence), the output (e.g. the three-dimensional structure) and a way to compare the function's output with the ground truth or a desired range of outputs (e.g. the average three-dimensional distance of an amino-acid residue from where it should be). The scientist then finds or collects a dataset, usually very large, with examples of the ground truth and applies AI optimization tools to the problem. So far, this kind of application has been highly successful, with new discoveries of tertiary protein structures, antibiotics and nuclear fusion reactor designs (see [1] for a review).

What makes this problem 'easy' is not the form of the solution (which may require a great deal of engineering work) but rather the form of the problem. It is clear from the beginning what needs to be optimized, and what kinds of tools can be brought to bear on this problem. The engineering breakthrough comes from building much better versions of these tools. In other words, the problem is relatively easy because it does not require any conceptual breakthroughs of the sort involved in the discovery of relativity theory, genetics or the periodic table.

2. The hard problem

The fundamental barrier to automating science is conceptual. Great scientists are not simply extraordinary optimizers of ordinary optimization problems—problems in which the data, space of solutions and an objective function for assessing the fit between them are already well defined. It is not like Einstein had a better function approximator in his brain than his peers did. More commonly, great scientists are ordinary optimizers of extraordinary optimization problems, and it is the formulation of the problem itself—not its solution—that is the truly hard problem. The hard problem is the 'problem problem'.

One might be tempted to relegate the hard problem to the fringes of 'revolutionary science', which rarely erupt into mainstream scientific practice, whereas the easy problem occupies the focus of the 'normal science' that scientists spend most of their time on [2]. However, normal science is not simply optimization. This is obvious to any first-year graduate student trying to figure out what to work on. For example, a student may arrive at the optimization problem of finding the best model of a particular cellular signalling pathway, but only after deciding that the pathway is functionally important, that it can be reasonably isolated and measured, and that a formal framework (e.g. differential equations) is applicable to describing the measurements. Normal science isn't a catalogue of optimization problems waiting to be solved by a queue of grad students. Their fundamental barrier is the same one facing AI scientists: It is the conceptual problem of formulating an optimization problem. This encompasses both major conceptual breakthroughs, like relativity theory, and the more modest ones achieved by graduate students on a regular basis, which nonetheless remain out of reach for existing AI systems. In the twentieth century, the philosophers of science Karl Popper and Larry Laudan recognized this when emphasizing the primacy of problem proposal and refinement for the scientific process [3–5].

3. The problem with optimization

Much of the classic work on AI science (mainly by Simon and their collaborators [6], but also more recently [7–10]) focuses on the easy problem. For Simon and Langley, this approach was premised on the psychological thesis that scientific cognition was essentially the same as regular problem solving, only applied to a different (and sometimes more challenging) set of problems. Consequently, they developed algorithms that emulated human problem solving and applied these to model seminal scientific discoveries, including the (re-)discovery of oxygen with STAHLp [11]. More modern methods for automated physics have also inferred many existing and novel laws, including classical and quantum problems with AI Feynman and nonlinear dynamical systems with SINDy [7–9], and discovery algorithms in biology have advanced our ability to solve many difficult problems, including AlphaFold2 for protein folding [10].

This success is analogous to the earliest use of computers, in which they were used to complete calculations too laborious for any human. Algorithms that solve the easy problems of science are useful, even essential, to progress. For example, there is an increasing discrepancy between the number of amino-acid sequences that are discovered in biology and the recovery of the three-dimensional protein structures they correspond to using the experimental method.

While recognizing the importance of such algorithms, we should also recognize their limitations. Several decades ago, Chalmers *et al*. challenged the idea that this kind of optimization was a complete model of scientific discovery and investigation [12]. Systems like STAHLp are only able to solve scientific problems and make discoveries, they argued, because the modellers have *represented the inputs and outputs to the problem in hindsight*. Only relevant data have been included, and those data are already organized such that the proposed heuristics will be able to easily extract the right solution. In other words, most AI scientists have been provided a representation of the scientific problem that already includes the basic primitives needed for the final theory, but skirt the central problem of representation itself: Where do the primitives come from, and how do we know if we have discovered the right ones?

Simon insisted (contra Popper [3]) that there *was* a logic of scientific discovery, but Simon's proposal was really a logic of scientific problem solving—how to sequentially search through hypotheses given a problem statement and primitive representations [13]. This is not discovery in the sense of problem creation. The latter involves representation learning in service of the problem, but also something deeper: identification of the objective function itself.

4. Solving the hard problem

In contemplating how to build AI systems that solve the hard problem, it is instructive to look at how human scientists do it. At a high level, humans break the hard problem down into two sub-problems:

- *Domain specification*. What are the relevant phenomena that need to be explained by a theory?
- *Constraint specification*. What kinds of constraints need to be imposed on a theory based on existing knowledge (both domain-specific and domain-general)?

Once the domain and constraints have been specified, we can define an optimization problem (theory search); hence, we have converted the hard problem into the easy problem. However, it is uncommon for real scientists to do a single pass from hard to easy, because they often realize that the problem they are solving is the wrong one. This may happen for several reasons. One is that a theory turns out to be internally inconsistent or paradoxical. Another is that the theory may (with suitable modification) be able to explain a broader range of phenomena, prompting a respecification of the domain. Conversely, phenomena that were previously included in a domain may need to be excluded if no adequate unifying theory is found for all the phenomena. Respecification can also happen when new empirical phenomena are reported. In a related vein, constraint respecification can happen when domains are merged, split, expanded or shrunk.

The key point is that problem creation and problem solving are cyclically coupled in scientific practice, and while the chosen problem formulation may appear obvious at the end of this process, it is seldom so right at the start. Defining formal desiderata for a suitable terminal problem formulation is beyond the scope of this paper, but might include (i) a family of solutions containing one or more that fit the data very well, (ii) a relevant subset or superset of the original data that still captures the original phenomenon of interest, (iii) a simple mathematical form, and (iv) a consistent relationship to adjacent problems and theories. For most current AI scientists, on the other hand, the modelling team has already conducted reasonably mature domain and constraint specification once, in advance—in the representation and selection of data, and in the representational scheme for potential scientific theories (outputs) and the objective function that assesses them, respectively.

Box 1: STAH L_p .

Beliefs

The inputs and outputs of STAHLp are two types of belief:
Componential model: "Substance X is COMPOSED OF {Y, Z})";
X = Y, Z
Reaction: "{W, X} REACT to produce {U, Z})";
W, X → U, Z

Production rules

STAHLp applies a set of production rules to generate further beliefs:

Substitute:	**Reduce**:	**Infer-components**:
W, X → U, Z X = Y, Z ∴ W, Y, Z → U, Z	W, Y, ~~Z~~ → U, ~~Z~~	U = W, Y

Objective

The objective function STAHLp uses to assess the consistency of a theory:

$$\begin{cases} \text{Fail if } nil \in \text{production rules(beliefs)} \\ \text{Continue otherwise.} \end{cases}$$

Belief revision rules

If an inconsistent belief is generated, STAHLp applies a different set of belief revision rules to the beliefs upstream of the problematic statement.

In the following sections, we motivate the distinction between the easy and hard problems with three case studies from the birth of modern chemistry, physics and molecular biology. For each case study, we summarize the elements of the problem, the historical setting and modern computational systems that have tried to recapture some aspects of these discoveries. We will argue that none of these systems offers a complete solution to the hard problem.

5. Case study 1: the discovery of oxygen

In the eighteenth century, it had been observed that lead increased in weight when it was slowly heated (which today we call 'oxidation', but at that time was called 'calcination'). This was difficult to explain with contemporary chemical theories, because they posited that something *left* a metal when it was heated (a type of inflammable earth called 'phlogiston'). In 1774, the English chemist Joseph Priestley collected and identified a particularly inflammable and respirable form of air following the thermal reduction of calx of mercury (mercury oxide) [14]. The French chemist Antoine Lavoisier eventually called this air 'oxygen' and posited that it went *into* the metal during calcination, causing the weight change [15]. Lavoisier's course of investigations was so successful that he has been credited as having started the Chemical Revolution and introduced the principled application of the conservation of mass into the quantitative sciences.

(a) STAHLp

Rose & Langley proposed a computational model called STAHLp to account for the discovery of the role of oxygen in calcination reactions [11] (see Box 1). The input to STAHLp is a set of interconnected beliefs about (i) which substances are present before and after a particular reaction or (ii) the chemical composition of each substance. These inputs are encoded using two types of variable: the predicates (or programs) REACTS and COMPOSED OF, and chemical names which

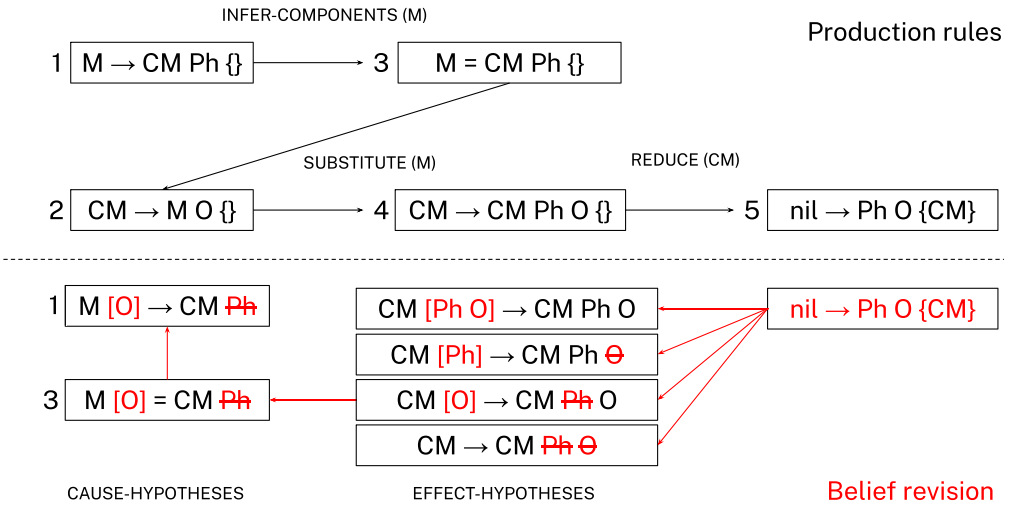

Figure 1. The discovery of oxygen by STAHLp. The input is two beliefs, where the first belief, inherited from Georg Ernst Stahl, states that mercury (M) is composed of calx of mercury (CM) and phlogiston (Ph). The second, reflecting an empirical observation by Joseph Priestley, states that calx of mercury is composed of mercury and a colourless gas (O). Through forward chaining, the system reaches a circularity present in the initial beliefs—that mercury-calx can be decomposed into itself, phlogiston and oxygen. The belief revision rules are then triggered, leading to a set of effect hypotheses, then cause hypotheses, giving the revisions needed to overcome the circularity. Reproduced from [11].

the functions operate over. STAHLp's desired output is a consistent 'theory'—a set of beliefs that entail the inputs and do not contradict each other.

STAHLp solves this problem by applying a set of 'production rules' to its beliefs, generating further beliefs until no more rules can be applied. If the system generates an inconsistent belief, STAHLp throws an error. At that point, a second set of 'belief revision' heuristics is applied to try to identify the source of the inconsistency and correct it. After the lowest-cost correction is made, STAHLp applies its production rules to generate the updated theory entailed by the new starting beliefs. This is what Simon meant by scientific problem-solving as search through a hypothesis space [13].

Rose and Langley showed that for a particular pair of beliefs, STAHLp 'discovers' oxygen (figure 1). When STAHLp's production rules are applied to these observations, they produce an inconsistent belief, which in turn triggers its update-belief rules, generating a set of 'effect hypotheses' that 'balance' the inconsistent belief (belief 4 in figure 1). By back-tracing where the left and right sides come from, STAHLp can generate 'cause-hypotheses' about how the initial beliefs should be updated to correct the inconsistency. The cause-hypothesis that affects the least downstream statements is chosen—in this case, the addition of 'oxygen' to the left-hand side of belief 1.

(b) Lavoisier

The modelling choices behind STAHLp neatly illustrate the easy versus hard distinction. It is hard to argue with the assumption that reactions and compositions of substances were the central concepts in chemistry—indeed, this was how Stahl himself defined its scope [16]. However, leading up to the chemical revolution, chemists had a different way of thinking about the internal structure of substances, in which observable substances arose from mixtures of the latent primitives of earth, water and sometimes fire. Notably, most continental European chemists would not have

considered air—what we now call gas—to have chemical properties and enter into chemical combinations [17]. By re-specifying the 'definitions of chemistry' to include air, Lavoisier was able to include measurements about gross changes in air volume in his theories, which in turn explained the gross weight changes during combustion, calcination and reduction by the chemical fixation or release of air. This kind of conceptual innovation, through the revision of hierarchical ontological constraints, has been implicated at the core of human conceptual development [18,19].

Next, Lavoisier broadened his scope to include all operations that fix or release air [20], with the aim of tracing the flow of air and water through different coupled reactions in order to infer the chemical composition of a more complex substance (like chalk). This richer set of data about weight and volume changes led to the development of quantitative models based on tables and rudimentary equations. Prior to Lavoisier, chemists categorized and weighed solids and liquids before and after reactions, but did not routinely measure the air surrounding these materials. So, their 'equations' were seldom balanced, and the conservation of mass was used more as a *post hoc* and abstract principle rather than a tool for quantitative purposes. In addition, in the eighteenth century, it was widely held that substances could dissipate away to nothing—from diamond to phlogiston itself [17].

Discrepancies in subsequent experiments led Lavoisier to the conclusion that there must be different *subtypes* of air with different densities. This led to the development of new equipment to measure those densities and, ultimately, the finding that the air of the atmosphere was in fact a composite of these subtypes, rather than an elemental root. He then showed that the reduction of calx of mercury with charcoal produced a different air (carbon monoxide and carbon dioxide) than the reduction of calx of mercury without charcoal, eventually calling the latter air 'oxygen'. Lavoisier explained the differences between these two reactions by positing an underlying, potentially infinite range of chemical primitives that could take the familiar three states of matter depending on how much of the 'matter of fire' was coupled with them. These primitives were simply those that could be isolated by the tools of chemistry at the time. This was the beginning of the main Chemical Revolution—actually more of an inversion, in that the things previously considered elemental (earth, water, air, fire) were now considered complex, whereas previously complex things like carbon were now considered elemental.

With this historical framing, it is easier to see how STAHLp's specification of the problem is essentially a modern one. The objective function for STAHLp is based on the detection of nil statements, whereas Lavoisier had to develop the conceptual machinery—the right representation of the problem—to represent a reaction in terms of the total weight of materials at the start and end, and in doing so established the loss function to be optimized—the inference of a consistent and useful set of equations. In other words, he constructed the right representation of the problem. This placed emphasis on the use of a density constant to relate changes in air volume to changes in weight. Furthermore, a new chemical name could not be added arbitrarily and had to be placed within the existing ontological structure that initially did not include air. 'Oxygen' is simply not a valid entry [17]. Finally, there is also the question of data selection. The creators of STAHLp include only two facts from the many heterogeneous and often inconsistent observations and beliefs in eighteenth-century chemistry. If they took into account others—for example, that when nitrous acid was poured on mercury, coloured vapours and fumes were given off—the model's conclusions may well have changed. And, there was actually a great deal of inconclusive or even negative evidence that metals apart from lead increased weight on calcination, detracting from the general statement that an air entered into metals.

6. Case study 2: the electromagnetic field

In the mid-nineteenth century, Michael Faraday published a set of discoveries and observations related to electromagnetic induction. Faraday recorded the intensity of current induced in a copper wire by moving it around magnets of various shapes, strengths and numbers, as well as other electrical circuits embedded in magnetic media, arguing that the most useful representation for

these data was in terms of *lines of magnetic force* [21]. He had speculated on what might be the cause of these patterns but had been largely unsuccessful [22]. The Scottish physicist James Clerk Maxwell derived a brilliant and creative theoretical solution to this problem that provides the foundation of modern physics—the mathematical representation of the electromagnetic field.

No computational model has been proposed to emulate Maxwell's discovery. However, several influential models target the general setting of deriving physical laws from datasets of this sort [7–9]. Here, we will focus on AI Feynman [8], an algorithm that uses *symbolic regression* to recover natural laws from physical data (see Box 2).

(a) AI Feynman

AI Feynman takes a data table, comprising data samples (rows) of a dependent variable and several independent variables (columns), and outputs a symbolic formula representing a theory of the system as well as a set of predictions in the input space. Variables take continuous values, correspond to measurements of the physical system and are augmented with type information representing their fundamental physical units (metre, second, kilogram, kelvin and volt). In order to infer the theory of how the independent variables determine the dependent variable, AI Feynman cycles through a set of pre-specified computational strategies premised on commonalities in the functional forms of solutions to known physical problems (figure 2). The algorithm stops when the squared-error loss between its predictions and the input is low enough and then checks whether its current solution is equivalent to the ground-truth expression.

For example, the data in figure 2 comprise samples from one dependent variable, F, and nine independent variables corresponding to the masses and three-dimensional positions of two objects, and Newton's constant G. The algorithm runs through its predetermined steps: algebraic manipulations yield a reduced set of dimensionless variables; the application of a neural network component identifies translational symmetry; a good factorization is found; then polynomials are fit to two subsets of transformed variables. The end result of this process is an equation that accounts for the data below some error threshold, ϵ (see figure 3).

(b) Newton and Maxwell

Although the input variables chosen *for each problem* in AI Feynman might seem logical, they in fact correspond to quite an advanced stage of problem solving—when scientists have already constructed an idealized model for the system at hand.[1] For example, Newton had to *posit* the idea of a gravitational constant, expressed implicitly in terms of proportionality, and he had to posit that these quantities were the *only* influential factors when explaining gravity—that action-at-a-distance was the correct framework to use, rather than the transmission of forces through an underlying medium.

Box 2: AI Feynman.

Stepwise objective	Final objective
$\begin{cases} 1 \text{ if } \lvert\lvert \frac{(\hat{f}(x)-x)^2}{n} \rvert\rvert^{\frac{1}{2}} < \epsilon_{\text{step}} \\ 0 \text{ otherwise.} \end{cases}$	$\begin{cases} \text{Pass if SIMPLIFY}(\hat{f}-f) == 0 \\ \text{Fail otherwise.} \end{cases}$

AI Feynman assesses whether the Euclidean distance between predictions and data is small enough after each step, and whether its symbolic expression is the same as the ground truth law.

[1]This is actually the process that Richard Feynman goes through in his lectures when giving the historical background of the problem statement.

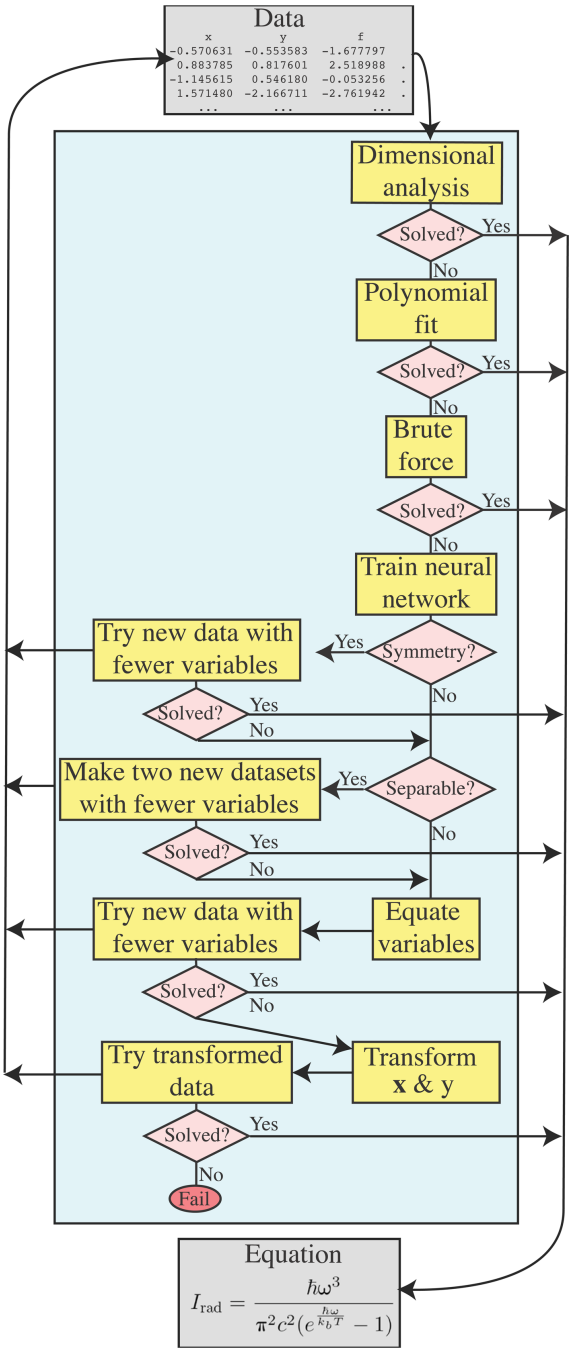

Figure 2. The steps that AI Feynman goes through when applied to solve a scientific problem, given in the form of a mystery table. Reproduced from [8].

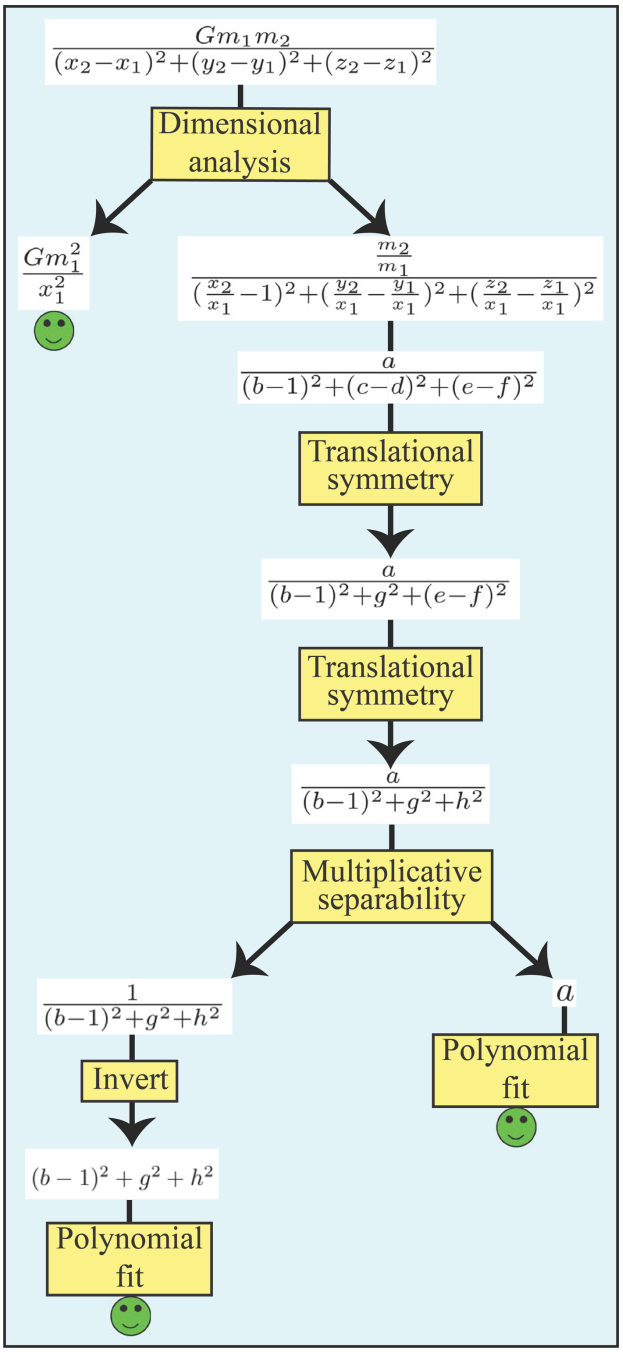

Figure 3. AI Feynman recovers the correct expression at the top of the diagram by applying its problem-solving steps. Reproduced from [8].

Nancy Nersessian has given a thorough cognitive–historical analysis of Maxwell and the development of the electromagnetic field concept [23]. A striking feature of Maxwell's problem-solving is how explicit he was about the scope of his theories and the utility of intermediate models. Selectively restricting the domain allowed him to identify which parameters or features of the intermediate model were essential, and an analysis of those features afforded selective expansion of the domain—a process Nersessian has called 'generic abstraction'.

Maxwell began by restricting his scope to Faraday's data on electromagnetic induction and lines of force, in order to make progress given the ill-defined and heterogeneous state of electrical science at the time. First, he constructed a *descriptive* mathematical model of Faraday's observations and theoretical postulations, based on the continuum mechanics of stresses in an underlying medium [24]. He then used this new representation of the data to design an explanatory dynamical model, iterating between adding to the range of phenomena he was considering and adapting his intermediate model to accommodate them and generate new predictions [25,26]. He began with magnetic phenomena and showed that the constraints provided by his descriptive analysis could be fit by a vortex model. From this model, he could calculate the magnetic force at any point in the medium by carrying over the system of equations describing the mechanical force that would be exerted in the vortex model and replacing mechanical variables with magnetic ones [27]. When he generalized this model to a *medium* composed of these vortices, however, he found the model unsatisfactory because of the friction that would occur between adjacent vortices. This brought to mind the idle wheels interposed between rotating machine gears, from which he introduced the idea of idle-wheel particles to communicate between vortices. Idle-wheel *particles* provided a good way to model electrical current, so his next step was to include electromagnetic phenomena. But this required the relaxation of the model to allow the particles to translate in a conductive medium and to rotate without generating any friction. Using the new model, he could bring in a set of equations to represent electrical current as the flux density of these particles, driven by the circumferential velocity of the vortices [27]. Maxwell continued this process of domain relaxation and model building to include electrostatic phenomena and the polarization of light.

The process of building and abstracting intermediary models is closely related to the process of making analogies. Accounts of analogy are widespread in science, usually as the initial source of inspiration leading up to a theoretical revolution or discovery [28]. In particular, Maxwell made several analogies between a domain where he had detailed knowledge (vortices and bearings, from continuum mechanics) and the new setting. This was licensed by a more general analogy, given as an explicit assumption, between mechanical media that could transmit mechanical force and a hypothesized medium called the aether that could transmit electromagnetic force.

Like Lavoisier, Maxwell was guided in this process of abstraction by *ontological knowledge* about the structure of different physical and mathematical systems, which also helped him sequentially assemble and modify the mathematical expressions underlying the model. Perhaps these idealized models played a role in Lavoisier's early investigations, albeit in a simpler form involving crude movements of air and changes of weight. What is clear is that this process is not captured by systems like AI Feynman, which are given the problem variables from the mature idealized model and lack the flexibility to alter their own conceptual systems—analogous to providing the symbol i before deriving a general solution to the problem of polynomial root finding.

AI systems for discovery, including AI Feynman, increasingly use deep neural networks (DNNs) as components to improve and combine problem primitives through representation learning. As their depth increases, DNNs have been shown to form increasingly abstract stimulus or task representations, which may even consist of entire frameworks. While increasing capability for abstraction with, e.g. depth, does in some sense mean deeper networks can search over frameworks, current systems still do so in the context of optimization based on a problem formulation provided by the system designer, rather than solving the hard problem itself.

The kinds of laws AI Feynman can derive are also limited by its processing steps. This is motivated by an analysis of common characteristics of physical laws—they contain variables with units, low-degree polynomial structure, compositionality, smoothness, symmetry and separability. But again, these constraints arose out of analysis of the existing laws of physics and provide constraints that restrict the subsequent class of models in an inflexible manner—Maxwell *invented* dimensional analysis to help solve difficult physics problems.

Once again, there is also the question of which datapoints are chosen. In figure 3, the data are not taken from systems far from the scientist or near large masses, where the behaviour of light

(its speed or deflection, respectively) needs to be taken into account. Recognizing and adjusting for these factors were essential parts of proving the theory and then taking it forward.

7. Case study 3: protein folding

Several major conceptual breakthroughs led to the 'protein-folding problem'. Frederick Sanger discovered that proteins are *linear* chains of amino acids based on the isolation and recursive extraction of hydrolyzed fragments of insulin using various media and electrical currents [29]. Evidence that the function of proteins depended on their three-dimensional structure, rather than the identity of individual amino acids, came from X-ray crystallography [30], the structural effects of natural and artificial variation of amino acids [31] and catalytic-rate analyses with different cellular conditions, substrates and inhibitors [32]. But it was Christian Anfinsen Jr. that put forward what became known as 'Anfinsen's dogma'—that under physiological conditions the primary sequence itself, and no other factors, determined the three-dimensional structure [33].

Box 3: AlphaFold2.

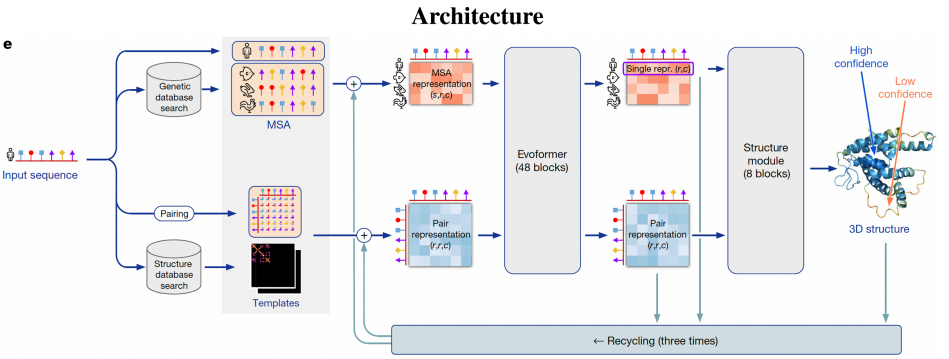

Figure 4. AlphaFold2 input is a one-dimensional amino-acid sequence and any related sequences from genomic databases (the MSA), as well as distance information from previously derived structures. The Evoformer module learns flexible and increasingly abstract re-representations of these inputs over many layers. The Structure module conditions on this information to output rigid three-dimensional frames for each amino acid residue, before adjusting the frames to form a physically plausible structure during fine-tuning. Reproduced from [10].

Objective

$$\mathcal{L} = \begin{cases} 0.5\mathcal{L}_{\text{FAPE}} + 0.5\mathcal{L}_{\text{aux}} + 0.3\mathcal{L}_{\text{dist}} + 2.0\mathcal{L}_{\text{msa}} + 0.01\mathcal{L}_{\text{conf}} & \text{training} \\ 0.5\mathcal{L}_{\text{FAPE}} + 0.5\mathcal{L}_{\text{aux}} + 0.3\mathcal{L}_{\text{dist}} + 2.0\mathcal{L}_{\text{msa}} + 0.01\mathcal{L}_{\text{conf}} + 0.01\mathcal{L}_{\text{exp resolved}} + 1.0\mathcal{L}_{\text{viol}} & \text{fine-tuning} \end{cases}$$

AlphaFold2's training loss contains terms for each representation maintained by the algorithm (the three-dimensional positions of each amino acid residue, the torsion angles between amino acid residues, the pairwise distances from the two-dimensional distogram, the masked one-dimensional input sequence, and the model's confidence score). The fine-tuning loss has two extra terms that penalize the violation of physical constraints.

(a) AlphaFold2

One of the most successful recent discovery algorithms is AlphaFold2 [10], which is a large neural network model that predicts the three-dimensional structure of a protein given its one-dimensional amino-acid sequence (see Box 3). When it was released, AlphaFold2 brought the

average molecular deviation for a protein down from 0.3 to 0.1 nanometres, which was finally precise enough for biologists to make use of. AlphaFold2 uses complex heuristics to solve this optimization problem, based on a great deal of biological and engineering knowledge. At a high level, its *Evoformer* module learns increasingly rich and abstract representations of the one-dimensional primary structure and a two-dimensional matrix of inter-atomic distances between residues, and its *Structure* module uses these representations to build a three-dimensional model of the protein. The network is trained end-to-end, meaning all operations are differentiable and the loss signal from the final three-dimensional positions is back-propagated to inform the update of neural network weights in all operations after the input.

AlphaFold2's input is a multiple sequence alignment (MSA), which augments the protein of interest's one-dimensional amino-acid sequence with additional rows containing similar amino-acid sequences from existing databases (figure 4). If any of the MSA's sequences have already had structures derived, two-dimensional distograms of the pairwise distance between residues and a sequence of torsion angles between adjacent amino-acid residues are added to the inputs. AlphaFold2 outputs a set of atomic co-ordinates, a confidence score in each residue's position, torsion angles between adjacent amino-acid backbones, the two-dimensional distogram between residues and a prediction of any masked parts of the MSA. The objective function during training contains a loss term for each of these representations, with the most important components penalizing the three-dimensional deviations of heavy atoms in the amino-acid chain. The loss function during 'fine-tuning' contains all of these terms, plus two extra terms that penalize the final structure for violating physical constraints.

(b) Anfinsen

AlphaFold2's success comes in large part from the engineering choice of problem statement. In particular, it does *not* solve the original 'problem' of protein folding. Anfinsen and his colleagues began by studying bovine ribonuclease, a protein that could be isolated in abundance using the techniques of the time. Ribonuclease has a functional structure that depends on the (single) correct formation of four disulphide bonds (out of a possible 105 combinations between eight sulfhydryl groups). When these were broken, then allowed to reform in denaturing conditions (containing a high concentration of urea), an inactive mixture of all the possible configurations was generated. In physiological conditions, however, the mixture was converted to the native ribonuclease. Anfinsen and colleagues moved on to study Staphylococcal nuclease, which does not depend on disulphide bonds for its three-dimensional functional structure, analysing the folding and functionality of various subsections of the molecule as well as the temporal and pH dynamics of the rather sharp phase transition between inactive and active molecules and the initial few folding events in renaturation. The cumulation of these events was the 'thermodynamic hypothesis'—that the correctly folded protein occupied the minimum free-energy state in its natural cellular environment [33].

However, the task left facing scientists working on the protein-folding problem was daunting: accounting for the full physical process by which a protein of N amino acids assumed one of its approximately 8^N possible conformations. Part of the genius of Alphafold2 was that its architects recognized that this problem did not need to be solved in order to make a substantial impact in biology, and they relaxed its constraints to define a new, related problem: the prediction of a final folded state given the one-dimensional sequence, leaving behind the requirement to model the time-evolving movement of the polypeptide chain from one-dimensional denatured to three-dimensional functional state. This respecification allowed them to bring in an abundance of sequence data, which can be used for the new, but not the old, problem. Evolutionary correlations have been used for some time to make arguments about folded structures and function [34], but do not obviously inform folding dynamics. It also allowed them to bring in the very flexible learning algorithms from modern deep learning. In particular, attention-based mechanisms model longer-range dependencies over one-dimensional sequences, making use of the

two-dimensional distogram representations, and training could be split into a free optimization period, where physical constraints can be violated during iterative representation learning and stochastic gradient descent over the residue gas representation, followed by a bespoke fine-tuning stage where physical constraints were met.

The positive consequence of this choice is that some requirements of a solution to the original problem are met—we can predict the structure of hydrophilic proteins with lots of analogous evolutionary sequences well. The negative consequence is that we don't have a model of the folding dynamics that can make good predictions of the structures of orphan molecules like antibodies, lipophilic molecules with no experimentally derived homologous structures, or the effect of a new mutation or ion on the final folded structure.

The authors of AlphaFold do not claim it to be a model of the process of scientific discovery in the same way as STAHLp or AI Feynman—it is a model of a biological process that can make new discoveries by solving the easy problem. However, it is instructive as an example because the scientific and engineering work that went into designing it gives a good demonstration of how scientists themselves solve the hard problem.

Although the scale and complexity of natural biological systems warrant different types of theory and strategies of investigation from chemistry and physics [35], there are also important commonalities—including the decomposition of complicated entities into smaller parts [36], use of ontologies [37] and imagistic intermediate models [23,34]. Another common strategy used by Lavoisier, Maxwell and other scientists is to modify the constraints of the problem because they are aware of potentially informative data or useful methods that are currently out of scope. The authors of AlphaFold used this strategy repeatedly and to great effect, identifying where rich enough sources of data existed—for example, MSAs—to enable different deep learning mechanisms—for example, attention.

8. Understanding the hard problem

The previous sections depict a recurring pattern: much progress in applications of AI to science has been made, but only with the aid of humans specifying the problem formulation. Thus, these systems are essentially solving the easy problem, not the hard problem. What makes the hard problem so hard?

An important and elusive feature of problem specification is that it is not a data modelling problem. The selection of what to model and what constraints to condition on is antecedent to any data modelling problem. It is also not reducible to a representation learning problem, in the sense of figuring out how raw sensory input maps to abstract representations. Of course, representation learning is important, but first, the scientist needs to know what problems the representations are being used to solve.

Are these conceptual breakthroughs just patterns that can be discovered with a sufficiently powerful pattern-recognition system? In a sense, yes, but before that can happen, something has to tell the pattern-recognition system what kind of patterns are interesting, important and useful. What problem is the pattern-recognition system designed to solve, and where does this come from? Sociological, aesthetic and utility considerations enter at the problem specification stage. Building an AI scientist is as much about shaping its tastes, style and preferences as it is about endowing it with powerful problem-solving abilities. While Popper felt the process of problem proposal was fundamental (contra Simon [13]), he also described it as a leap of the imagination that necessarily went beyond existing knowledge and was therefore outside of the scope of logical analysis [38]. With the advance of modern cognitive models of problem-solving and creativity and modern machine learning (ML) and reasoning methods, we can now test formal models of such leaps and expect the results to be extremely informative about the overall discovery process. Again, a look at how we train human scientists is instructive: a good graduate advisor educates students about what problems matter, what phenomena are interesting, which explanations count and so on. These considerations can't be brushed aside as subjective factors irrelevant to the purely

technical problems facing AI systems; they are in fact constitutive of those technical problems. Without them, the technical problems would not exist.

A research programme for attacking the hard problem should begin with the cognitive science of science [39], focusing on the understudied subjective, creative aspects discussed above and how they interact with the objective aspects of problem solving. In the study of science, there is a natural continuum through rare but large breakaway conceptual developments, best studied through cognitive–historical analyses [23,37,40]; daily scientific problem solving, studied through observation of practising labs [41]; and large datasets of decisions in simplified problem-solving settings in online naturalistic games [42]. We can test any hypotheses about science that we derive from these analyses in large-scale online behavioural experiments, where, for example, related studies already provide evidence that humans construct simplified mental representations to plan [43] and that iterative model-based revision of problem statements is a critical part of deriving a successful scientific solution [44].

9. Towards scalable AI scientists that solve the hard problem

Once we understand what human scientists are doing with enough precision that we can formalize their activities, we can try to leverage these insights to build scalable AI scientists. At least initially, it is unlikely that these will be stand-alone systems, but rather more like research assistants or first-year grad students: curious agents with some technical competence but in need of expert guidance. An important development in this regard has been the production of 'generative' AI systems that are primarily trained to create new instances of stimuli based on previous experiences. This is opposed to 'narrow AI', which is deployed to provide a solution to a well-defined problem (i.e. solve the easy problem).

PowerPlay was one of the first generative AI systems to explicitly address the problem of problem invention, as the solution to building an increasingly general problem solver [45]. The strategy used by PowerPlay, inspired by creative self-supervised play by children and animals, is to search for the next simplest task that could not be solved by the current problem solver, but could be with a simple modification (while preserving the ability to solve all previous tasks). Which tasks and modifications are 'simple' depends on their description length—which in turn depends on the experiences the solver has had so far—as well as the ease with which a solution might be verified, a framework closely related to the process of conceptual change and experimentation in scientific problem solving. DreamCoder is another such system, proposed as a model of expert problem solving, that incrementally improves its ability to model a set of data by learning a growing library of programs in response to real and imagined programs [46].

Modern generative AI systems based on large language models (LLMs) have substantially improved capabilities for tackling the hard problem because they can receive instructions and output responses in natural language, diagrams and figures and computer code [47,48]—the formats that most human scientists communicate their theories and discoveries in. This guidance can come in the form of natural language instruction, reading curricula and demonstrations.

AlphaEvolve, a scientific co-pilot based on generative AI, is an interesting recent example of such a system. An expert human scientist or engineer first inputs a prompt containing a problem statement in natural language, a quantitative method to assess the suitability of solutions and an initial solution in computer code, and AlphaEvolve then uses evolutionary search to come up with increasingly better code-based solutions to the problem [49]. This may seem like a classic disjunction of the hard problem, solved by a human, and the easy problem, solved by AI, except for the fact that AlphaEvolve has a meta-learning component that adjusts the prompt itself—although not yet the scoring function. AlphaEvolve has mainly been applied to autonomously verifiable domains like mathematics and computer science, wherein a scoring function can be easily defined and applied. However, recent work has begun to use likelihood functions to apply it to experimental data and generate new cognitive theories of reward learning [50]. The growth of models beyond this requires the examination and emulation of the communal aspects of science

and related cultural institutions. Lab meetings, conferences, and presentations and discussions are ultimately the place where judgements on the quality of a scientific problem are made.

The use of natural language processing for scientific discovery is at the heart of the recently proposed 'AI Scientist' [51], arguably the first artificial scientist that addresses the hard problem. The algorithm itself is a carefully designed system of LLMs, prompting schemes, a coding assistant and templates for papers and conference guidelines that autonomously update ML code in order to generate scientific papers. In its inner loop, the AI Scientist is given access to the training, testing and visualization code for a simple ML model and dataset, along with several suggestions of innovative changes to the code and the overall objective of reducing the model's loss on held-out data. Its outer loop requires that it generate a range of ideas in natural language format, check their novelty using the internet, apply several ideas, write a ML paper for each of the ideas that ran successfully, review the paper and then update the paper. The proposed system comprises a carefully designed interface of language models, prompting schemes, a coding assistant and templates for papers and conference guidelines. From the examples presented, the innovative ideas that the AI Scientist generates are mostly decisions to split variables or processing pathways, add new model components or training metrics based on previously successful strategies in the literature and combine any of the above that improve the final loss. A particularly impressive part of the work is the ability to implement these high-level conceptual changes in the code example, including producing useful visualizations.

Whether scientific strategies developed on a narrow artificial domain, like the ML research done by the AI Scientist, transfer to natural scientific domains, is an interesting open question. It may well be the case that some do—for example, computational strategies for representing and modifying concepts—and some do not—for example, designing experiments. It has certainly been the case that modern 'large reasoning models' based on search can be formulated to perform well in a formal domain, where there are deductively verifiable reward signals, but currently lack the flexible real-world reasoning in the face of vast uncertainty that scientists—and normal people—spend their time thinking about.

It remains to be seen if systems like AlphaEvolve, the AI scientist and their successors can produce radically innovative discoveries. There is already emerging evidence that such models can predict the outcome of neuroscience experiments better than some human scientists [52]. Will such systems replicate human strategies such as ontologically guided constraint respecification, producing and modifying intermediate models and re-specifying the problem based on knowledge of adjacent rich sources of data and available models? Natural language is certainly capable of capturing the ontological structure of knowledge, and multi-modal models should be able to create and maintain imagistic intermediate models of scientific phenomena. Likewise, we would like AI scientists that can recognize when progress has been slow on a particular problem, but adjacent sources of data and powerful models show promise and could be brought in should the problem be modified slightly. We would also like them to recognize when and how to gather more useful data when there is a mismatch with the use case, as recent improvements on using AlphaFold2 to predict protein structures have done.

On the other hand, many scientific developments, including those we have characterized above, come from a reflective consideration of either how to alter model constraints to capture anomalous data [5,37] or where an alteration of model constraints affects the domain, borne out over a course of successive investigations [20,23]. Whether current LLMs can capture this type of reflective continual learning and selective conceptual respecification will require further investigation [53,54], and the first compelling demonstration of AI solving the hard problem may well be the production and defence of a doctoral thesis, rather than the suggestion of a single experiment.

An intriguing final question, alluded to above, is whether a very powerful autoregressive natural language system, trained on the entire scientific literature, instructed to identify and test promising new problem statements and continually updated, would provide a solution to the hard problem. While there is no question that in the limit it would—within scientific papers

prior work and promising future studies are almost always discussed, as are social considerations of importance, taste and utility—in this paper, we have argued that there is still a great deal that is unknown about science that occurs outside the literature, in the individual and collective problem-solving journeys of scientists, which would mean such a model is still some way off. It could even be that publishing theory about these cognitive processes themselves is a major step towards such a fully autonomous system (for example, [55]). For now, humans remain the only intelligent system capable of solving the hard problem. We still have much to learn about building AI scientists by studying ourselves.

Data accessibility. This article has no additional data.

Declaration of AI use. We have not used AI-assisted technologies in creating this article.

Authors' contributions. R.B.: conceptualization, writing—original draft; S.G.: conceptualization, funding acquisition, project administration, supervision, writing—original draft, writing—review and editing.

Both authors gave final approval for publication and agreed to be held accountable for the work performed therein.

Conflict of interest declaration. We declare we have no competing interests.

Funding. This work was supported by the Kempner Institute for the Study of Natural and Artificial Intelligence and by the Schmidt Sciences Polymath Program.

Acknowledgements. We are grateful to Nancy Nersessian, Melanie Mitchell, Giovanni Pezzulo, Frank Keil, Tejas Ramdas, Kim Stachenfeld, Skyler Wang, George Davis-Smith, John Clement, James Whittington, Jürgen Schmidhuber, Jay McClelland and Tom Griffiths for helpful discussions.

References

1. Wang H *et al*. 2023 Scientific discovery in the age of artificial intelligence. *Nature* **620**, 47–60. (doi:10.1038/s41586-023-06221-2)
2. Kuhn TS. 1962 *The structure of scientific revolutions*. Chicago, IL: University of Chicago Press. (doi:10.7208/chicago/9780226458144.001.0001)
3. Popper K. 1959 *The logic of scientific discovery*. London, UK: Hutchinson & Co. (doi:10.4324/9780203994627)
4. Popper K. 1963 *Conjectures and refutations: the growth of scientific knowledge*. London, UK: Routledge. (doi:10.1063/1.3050617)
5. Laudan L. 1977 *Progress and its problems*. Berkeley, CA: University of California Press.
6. Simon HA, Langley PW, Bradshaw GL. 1981 Scientific discovery as problem solving. *Synthese* **47**, 1–27. (doi:10.1007/BF01064262)
7. Schmidt M, Lipson H. 2009 Distilling free-form natural laws from experimental data. *Science* **324**, 81–85. (doi:10.1126/science.1165893)
8. Udrescu SM, Tegmark M. 2020 AI Feynman: a physics-inspired method for symbolic regression. *Sci. Adv*. **6**, eaay2631. (doi:10.1126/sciadv.aay2631)
9. Brunton SL, Proctor JL, Kutz JN. 2016 Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl Acad. Sci. USA* **113**, 3932–3937. (doi:10.1073/pnas.1517384113)
10. Jumper J *et al*. 2021 Highly accurate protein structure prediction with AlphaFold. *Nature* **596**, 583–589. (doi:10.1038/s41586-021-03819-2)
11. Rose D, Langley P. 1986 Chemical discovery as belief revision. *Mach. Learn*. **1**, 423–452. (doi:10.1007/BF00114870)
12. Chalmers DJ, French RM, Hofstadter DR. 1992 High-level perception, representation, and analogy: a critique of artificial intelligence methodology. *J. Exp. Theor. Artif. Intell*. **4**, 185–211. (doi:10.1080/09528139208953747)
13. Simon HA. 1973 Does scientific discovery have a logic? *Philos. Sci*. **40**, 471–480. (doi:10.1086/288559)
14. Priestley J. 1775 An account of further discoveries in air. *Philos. Trans*. **65**, 384–394.
15. Lavoisier A. 1790 *Elements of chemistry in new systematic order, containing all modern discoveries*. Edinburgh, UK: William Creech.

16. Stahl G. 1723 *Fundamenta chymiae dogmaticae & experimentalis*. Nürnberg, Germany: Adelbulner für Endter.
17. Guerlac H. 1961 *Lavoisier—the crucial year: the background and origin of his first experiments on combustion in 1772*. Ithaca, NY: Cornell University Press.
18. Keil FC. 1979 *Semantic and conceptual development: an ontological perspective*. Cambridge, MA: Harvard University Press.
19. Rogers T, McClelland J. 2004 *Semantic cognition: a parallel distributed processing approach*. Cambridge, MA: MIT Press.
20. Holmes FL. 1997 *Antoine lavoisier: the next crucial year: or, the sources of his quantitative method in chemistry*. Princeton, NJ: Princeton University Press.
21. Faraday M. 1852 On lines of magnetic force: their definite character and their distribution within a magnet and through space. *Phil. Trans. R. Soc. Lond*. **142**, 25–56.
22. Faraday M. 1852 On the physical character of the lines of magnetic force. *Lond. Edinb. Dublin Philos. Mag. J. Sci*. **3**, 401–428. (doi:10.1080/14786445208647033)
23. Nersessian NJ. 2010 *Creating scientific concepts*. Cambridge, MA: MIT Press.
24. Maxwell JC. 1855 On Faraday's lines of force. In *Scientific papers* (ed. WD Niven), pp. 155–229. Cambridge, UK: Cambridge University Press.
25. Maxwell JC. 1861 On physical lines of force. In *Scientific papers* (ed. WD Niven), pp. 451–513. Cambridge, UK: Cambridge University Press.
26. Maxwell JC. 1864 On physical lines of force. In *Scientific papers* (ed. WD Niven), pp. 526–597. Cambridge, UK: Cambridge University Press.
27. Nersessian NJ. 2002 Maxwell and 'the method of physical analogy': model-based reasoning, generic abstraction, and conceptual change. In *Essays in the history and philosophy of science and mathematics*, pp. 129–166. La Salle, Illinois.
28. Holyoak KJ, Thagard P. 1996 *Mental leaps: analogy in creative thought*. Cambridge, MA: MIT Press.
29. Sanger F, Tuppy H. 1951 The amino-acid sequence in the phenylalanyl chain of insulin. I. The identification of lower peptides from partial hydrolysates. *Biochem. J*. **49**, 463–481. (doi:10.1042/bj0490463)
30. Kendrew JC, Bodo G, Dintzis HM, Parrish RG, Wyckoff H, Phillips DC. 1958 A three-dimensional model of the myoglobin molecule obtained by X-ray analysis. *Nature* **181**, 662–666. (doi:10.1038/181662a0)
31. Perutz M, Kendrew J, Watson H. 1965 Structure and function of haemoglobin: II. Some relations between polypeptide chain configuration and amino acid sequence. *J. Mol. Biol*. **13**, 669–678. (doi:10.1016/S0022-2836(65)80134-6)
32. Thoma JA, Koshland DE. 1960 Competitive inhibition by substrate during enzyme action. evidence for the induced-fit theory. *J. Am. Chem. Soc*. **82**, 3329–3333. (doi:10.1021/ja01498a025)
33. Anfinsen CB. 1973 Principles that govern the folding of protein chains. *Science* **181**, 223–230. (doi:10.1126/science.181.4096.223)
34. Fitch WM, Margoliash E. 1970 The usefulness of amino acid and nucleotide sequences in evolutionary studies. *Evol. Biol*. **4**, 67–109.
35. Bechtel W, Richardson RC. 2010 *Discovering complexity: decomposition and localization as strategies in scientific research*. Cambridge, MA: MIT Press.
36. Beretta M. 1993 *The enlightenment of matter: the definition of chemistry from agricola to lavoisier*. Canton, MA: Science History Publications.
37. Darden L. 1991 *Theory change in science: strategies from mendelian genetics*. Oxford, UK: Oxford University Press.
38. Magee B. 1973 *Popper*. London, UK: Collins.
39. Thagard P. 2012 *The cognitive science of science: explanation, discovery, and conceptual change*. Cambridge, MA: MIT Press.
40. Battleday RM, Gershman SL. In preparation. *States of mind: Lavoisier's conceptual revolution in chemistry*. Princeton, NJ: Princeton University Press.
41. Nersessian N. 2024 In vitro analogies: simulation modeling in bioengineering sciences. In *The routledge handbook of philosophy of scientific modeling* (eds T Knuuttila, N Carrillo, R Koskinen). New York, NY: Routledge. (doi:10.4324/9781003205647-40)
42. Allen K *et al*. 2024 Using games to understand the mind. *Nat. Hum. Behav*. **8**, 1–9. (doi:10.1038/s41562-024-01878-9)

43. Ho MK, Abel D, Correa CG, Littman ML, Cohen JD, Griffiths TL. 2022 People construct simplified mental representations to plan. *Nature* **606**, 129–136. (doi:10.1038/s41586-022-04743-9)
44. Clement J. 2019 Imagistic simulation and physical intuition in expert problem solving. In *Implicit and explicit knowledge* (ed. D Tirosh). Norwood, NJ: Ablex Publishing. (doi:10.4324/9781315789354-35)
45. Schmidhuber J. 2013 Powerplay: training an increasingly general problem solver by continually searching for the simplest still unsolvable problem. *Front. Psychol*. **4**, 313. (doi:10.3389/fpsyg.2013.00313)
46. Ellis K, Wong L, Nye M, Sablé-Meyer M, Cary L, Anaya Pozo L, Hewitt L, Solar-Lezama A, Tenenbaum JB. 2023 DreamCoder: growing generalizable, interpretable knowledge with wake–sleep Bayesian program learning. *Phil. Trans. R. Soc. A* **381**, 20220050. (doi:10.1098/rsta.2022.0050)
47. Brown T *et al*. 2020 Language models are few-shot learners. *Adv. Neural Inf. Process. Syst* **33**, 1877–1901. (doi:10.48550/arXiv.2005.14165)
48. Bosma M, Chi E, Ichter B, Le QV, Schuurmans D, Wang X, Wei J, Xia F, Zhou D. 2022 Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst*. **35**, 24824–24837. (doi:10.52202/068431-1800)
49. Novikov A *et al*. 2025 AlphaEvolve: a coding agent for scientific and algorithmic discovery. *arXiv Preprint arXiv:2506.13131*.
50. Castro PS *et al*. 2025 Discovering symbolic cognitive models from human and animal behavior. In *Forty-Second Int. Conf. Machine Learning*. bioRxiv.
51. Lu C, Lu C, Lange RT, Foerster J, Clune J, Ha D. 2024 The AI scientist: towards fully automated open-ended scientific discovery. See https://arxiv.org/abs/2408.06292
52. Luo X *et al*. 2024 Large language models surpass human experts in predicting neuroscience results. *Nat. Hum. Behav*. **9**, 305–315. (doi:10.1038/s41562-024-02046-9)
53. Mitchell M. 2020 On crashing the barrier of meaning in artificial intelligence. *AI Mag*. **41**, 86–92. (doi:10.1609/aimag.v41i2.5259)
54. Hase P, Bansal M, Kim B, Ghandeharioun A. 2024 Does localization inform editing? Surprising differences in causality-based localization vs. knowledge editing in language models. *Adv. Neural Inf. Process. Syst*. **36**. (doi:10.52202/075280-0774)
55. Almaatouq A, Griffiths TL, Suchow JW, Whiting ME, Evans J, Watts DJ. 2024 Beyond playing 20 questions with nature: integrative experiment design in the social and behavioral sciences. *Behav. Brain Sci*. **47**, e33. (doi:10.1017/S0140525X22002874)