THE ROBOT SCIENTIST ADAM

Ross D. King, Jem Rowland, Wayne Aubrey, Maria Liakata, Magdalena Markham,
Larisa N. Soldatova, Ken E. Whelan, Amanda Clare, Mike Young, and Andrew Sparkes
Aberystwyth University

Stephen G. Oliver and Pınar Pir
Cambridge Systems Biology Centre, University of Cambridge

Despite science's great intellectual prestige, developing robot scientists will probably be simpler than developing general AI systems because there is no essential need to take into account the social milieu.

Computer science is increasingly setting the general scientific agenda. Using computers to execute experiments produces ever-greater amounts of scientific data.[1] This growth in data has, in turn, required the ever-greater use of computers for modeling and analysis.

Computer science is also changing how researchers express and communicate scientific knowledge. Most such knowledge is expressed using only natural language—the exception that proves the rule being mathematics. The desire to make scientific data more amenable to computer analysis motivates developments such as ontologies and the Semantic Web.[2] These developments provide general benefits, since scientific knowledge is best expressed using formal logic, as this promotes semantic clarity, which in turn supports the free exchange of scientific knowledge.

AI BACKGROUND

The branch of artificial intelligence (AI) devoted to scientific applications is known as *scientific discovery*. Pioneering work in this area resulted in development of Dendral in the 1960s to analyze mass-spectrometric data, followed by Meta-Dendral, one of the first machine-learning systems.[3] The use of machine learning in science is now ubiquitous and no longer considered "AI."

After Dendral, programs such as Prospector, Bacon, Fahrenheit, and more recent successors have demonstrated the automation of some planning, analysis, and discovery portions of the scientific enterprise.[4]

Despite these successes, most scientific discovery programs have been designed to produce an answer and then stop. They do not fully "close the loop" in the sense of examining the experiments' results, deciding what to do next, and thus potentially cycling forever. Most scientific discovery systems have another major limitation—they lack the ability to physically execute their own experiments. This ability is crucial for AI systems to work semi-independently in laboratories. Jan Żytkow was the first to think of connecting scientific discovery systems to laboratory equipment.[5]

Robot scientists

Robot scientists are a natural development of the trend toward ever-greater computer involvement in science. They are physically implemented laboratory automation systems that apply techniques from AI to automatically execute cycles of scientific experimentation.[6,7] A robot scientist, such as the one represented in Figure 1, originates hypotheses to explain observations, devises experiments to test these hypotheses, physically runs the experiments using laboratory automation equipment, interprets the experimental results, and then repeats the cycle.

The motivation for developing robot scientists is both philosophical and technological. A strong philosophical position holds that we do not fully understand a phenomenon unless we can make a machine that reproduces it. The advantage of this approach to the philosophy of science, compared to the traditional ones, is that it is constructive and objective.

The technological motivation for robot scientists is to increase science's productivity and efficiency. In many areas of scientific research our ability to generate data outstrips our ability to analyze it.[1] Robot scientists have the potential to change this through the integration of cycles of high-throughput hypothesis formation and testing.

Robot scientists also have the potential to improve the repeatability and reuse of scientific knowledge by enabling the description of experiments in greater detail and semantic clarity. As robot scientists conceive and execute experiments automatically, they can formalize and digitally curate all aspects of the scientific process: hypotheses, experimental goals, results, conclusions, and so on.[8]

In 2004, we proposed the concept of a robot scientist and demonstrated that it could automate cycles of hypothesis generation and experiments.[6] This proof-of-principle work was limited in three ways: The robot scientist was only capable of rediscovering known scientific knowledge, the experiments were not fully automated (many steps were performed by a technician under instruction from the robot scientist), and the steps in the scientific process were neither formalized nor explicit.

Application domain: Yeast functional genomics

Our work focuses on the application domain of yeast (*Saccharomyces cerevisiae*) functional genomics. Yeast is the organism used to make bread and wine, but its main interest to biologists is that yeast cells can serve as a model for human cells. Yeast cells are simpler in structure and easier to manipulate than human cells, and, thanks to evolutionary conservation, most of what is true for yeast holds true for human cells. The yeast genome, first sequenced in 1996, encodes approximately 6,000 genes. This provides the yeast "parts list."
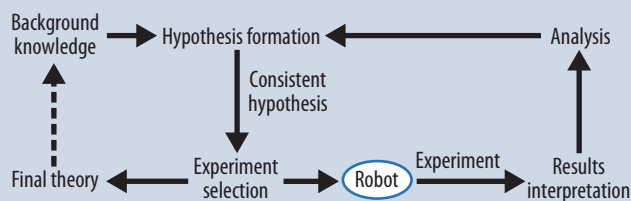
Figure 1. Robot scientist hypothesis-generation and closed-loop experimentation. This physically implemented laboratory-automation system applies AI techniques to execute cycles of scientific experimentation.

The field of molecular biology known as *functional genomics* tackles the problem of discovering the relationship between genes and their functions. Despite yeast being arguably the best understood of all organisms, we do not know the function of the proteins encoded by approximately 1,000 genes (www.yeastgenome.org). Conversely, protein functions are known to exist in yeast where the genes encoding them are unknown; these functions are known as locally orphan.

Our work focuses on this latter problem. Specifically, we designed the robot scientist Adam to generate and experimentally test hypotheses about which genes encode locally orphan enzymes—proteins that catalyze specific chemical reactions.
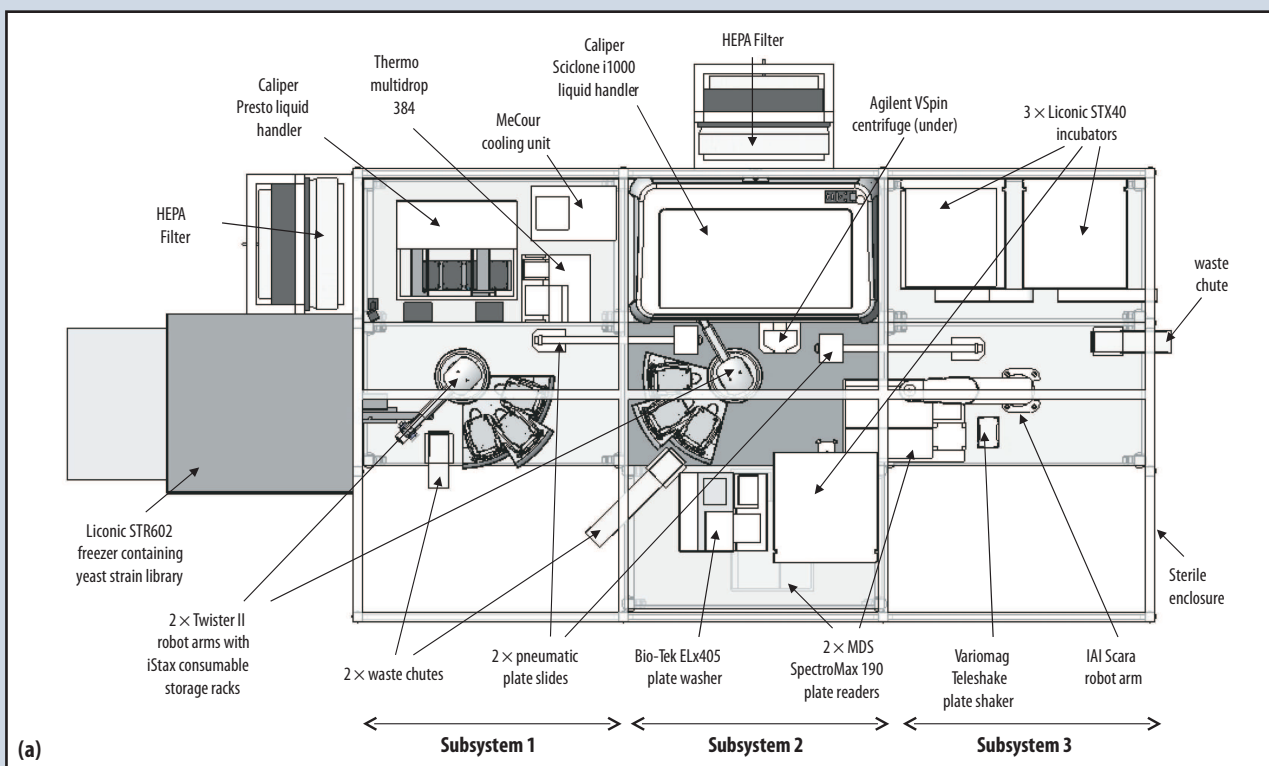
ADAM

We developed Adam to overcome the limitations of our proof-of-principle robot scientist. Adam's hardware can fully automate microbiological experiments, its software is sophisticated enough to discover new scientific knowledge, and its scientific investigations are explicit and formalized in logic.

Hardware

Adam is designed to execute microbial and especially yeast-growth experiments, as Figure 2 shows. These experiments involve selecting microbial strains and growth media, then observing the growth of the strain in the media over several days. We designed Adam to initiate approximately 1,000 strain/medium combinations a day. Adam's fully automated hardware alleviates the need for a technician except to periodically add laboratory consumables and remove waste.

Adam's individual design of the experiments and utilization of complex internal cycles distinguish it from other complex laboratory systems (such as high-throughput drug screening pipelines and X-ray crystallography systems). We have yet to encounter any laboratory automation system that can execute anything approaching the same number of individually planned experiments that Adam can.

Figure 2. Robot scientist Adam. (a) Plan for Adam, actual size ~5m × 3m × 3m; (b) Adam conducts experiments by making optical density measurements.

Software

Adam's core lies in its software, which requires many modules.

Background knowledge. To form novel hypotheses about a scientific domain, Adam needs a knowledge base about the subject. We therefore developed a logical model of yeast metabolism that includes most of the genes, proteins, enzymic functions, and metabolites in yeast. There are approximately 1,000 objects in each class. The model's structure is in essence a directed labeled hypergraph (with metabolites as nodes and enzymes as arcs) encoded in the Prolog logic programming language. We also provided Adam with a database of general bioinformatic data.

Hypothesis generation. Adam uses abductive reasoning to form hypotheses.[9] Abduction is best explained in comparison with deduction, as Figure 3 shows.

Deduction provides the basis for traditional logic and the foundation for mathematics and computer science. Deductive reasoning is sound—that is, if you start with truth you can only infer new truths. Unfortunately, in the absence of the ultimate "theory of everything," deduction is insufficient for scientific inference, for it only works out the consequences of what is already known.

The easiest way to think about abduction is as deduction in reverse. Abductive reasoning is not in itself sound, and empirical observations are required to ensure the truth of inferences. For example, there are many other possible explanations for why a cell might not grow other than

being unable to synthesize tryptophan. Instead, abduction infers hypotheses that might possibly be true. The great fictional detective Sherlock Holmes often claimed to have "deduced" the solution to a crime—but this is technically incorrect: He used abductive reasoning, as can be ascertained by thinking up other unlikely but not logically impossible solutions.

In our original work on robot scientists, we used a purely logical approach to hypothesis formation based on applying abductive logic programming to a logical model of a yeast metabolism.[6] Unfortunately, this general method is too inefficient to deal with large knowledge bases such as our model of yeast metabolism. We thus developed an alternative approach based on using standard bioinformatic methods coupled with the insight that standard genome annotation is a vast process of abductive hypothesis formation.

Experiment selection. Adam follows a hypothetico-deductive methodology: Once it abductively infers a hypothesis, the consequences can be deduced for experimental testing. Given a set of hypotheses, one wishes to infer an experiment that will efficiently discriminate between them. Assuming that every experiment has an associated monetary cost, and that each hypothesis has a probability of being correct, we seek to choose a series of experiments that minimize the expected cost of eliminating all but one hypothesis. This problem is NP complete and thus in general intractable. However, it can be shown that the problem is the same as finding an optimal decision tree, and we know this problem can be solved "nearly" optimally in polynomial time. In previous work, we compared this experiment selection strategy with random experiment selection and a simple strategy of choosing the cheapest experiment; we showed that intelligent experiment selection works significantly better than either random selection or the simple strategy.[6]

Planning experiments. Adam applies classical experimental design techniques to plan its experiments. In our functional genomics work we used a two-factor design: a "wild-type" v deletion strain, defined medium v, and defined medium + metabolite. The wild type is the standard yeast with nothing removed—the "vanilla" type in computer science terminology—while the deletion strains are mutant versions with genes removed that hypothesized to encode an orphan enzyme. Adam used standard 96-well plates to grow the yeast, which enabled 24 repeats of each strain and medium combination. To control for intraplate environmental effects, we used six Latin squares per plate. Such an experimental design could not have been employed for manual experiments because of the complexity of the liquid's handling steps.

Laboratory automation software. Adam requires this software to coordinate and drive the different pieces of laboratory automation equipment. This type of software

Types of Logic

Deduction
    Rule: If a cell grows then it can synthesize tryptophan.
    Fact: Cell cannot synthesize tryptophan.
    ∴   Cell cannot grow.
Given the rule $P \rightarrow Q$, and the fact $\neg Q$, infer the fact $\neg P$ (modus tollens)

Abduction
    Rule: If a cell grows then it can synthesize tryptophan.
    Fact: Cell cannot grow.
    ∴   Cell cannot synthesize tryptophan.
Given the rule $P \rightarrow Q$, and the fact $\neg P$, infer the fact $\neg Q$

Figure 3. Types of logic. Researchers use both deduction and abduction to form hypotheses. Deduction provides the basis for traditional logic, while abduction can be considered as deduction in reverse.

is generally only available commercially, and could be greatly improved, so we would like to develop an alternative open source solution.

Analyzing observations. Adam's experimental observations focus on the optical density (OD) of the yeast cultures, which relates to the number of yeast cells. In the course of its investigations, Adam made 6,657,024 OD observations, measured at 595 nm that, when plotted versus time, formed 26,495 growth curves. These data are held in a MySQL relational database. Adam used spline fitting to form continuous growth curves, and from these extracted biologically relevant parameters.
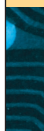
Deciding on hypotheses. Given a set of experimental observations, Adam must decide if they are consistent with a hypothesis. This was perhaps the hardest part of the project, for yeast has been so well studied that the functions of all the genes whose removal results in qualitative differences in growth have already been discovered. This means that individually removing the remaining genes is expected to produce only quantitative differences in growth between the mutants and the wild type. Therefore, Adam used decision trees and random forests, with resampling methods, to decide whether the difference in growth between the deletion strain growing on a defined medium and on a defined medium + metabolite could be discriminated from the difference in growth between the wild type growing on a defined medium and on a defined medium + metabolite. We confirmed these results using standard statistical methods.

RESULTS BREAKDOWN

To demonstrate the automation of experiment cycles using a robot scientist, we programmed Adam to repeat the experiments of our first, semiautomated, robot scientist.[6]

These experiments concern the rediscovery of functional genomics knowledge about the aromatic amino acid biosynthesis pathway in yeast. We compared Adam's results with those performed by the previous robot scientist, and found that Adam's were slightly better than the original. This demonstrates that cycles of experimentation can be fully automated,[7] and confirms the original robot scientist's results.

We then applied Adam to the discovery of novel scientific knowledge. Adam generated and tested 20 hypotheses about which genes encode 13 orphan enzymes. The weight of the experimental evidence for the hypotheses varied (based on observations of differential growth), but 12 hypotheses with no previous evidence were confirmed with $p < 0.05$ for the null hypothesis. We presume that the discovery of the genes encoding these enzymes is particularly hard because they have remained unknown despite decades of research. To ensure that Adam arrived at the correct conclusions, we checked them manually.

> **The development of robot scientists makes possible formalizing and recording all aspects of a scientific investigation using logic.**

For the genes hypothesized to encode the enzyme 2-aminoadipate transaminase[10] we used the gold-standard biological method of purifying the proteins encoded by the genes and testing them for enzymic function. These experiments confirmed Adam's conclusions. For the others, we searched the biological literature for information. This revealed the existence of strong empirical evidence for the correctness of six hypotheses whose enzymes were not actually local orphans. Adam considered them orphans because it was using an incomplete bioinformatic database. These six genes therefore constitute a positive control for Adam's methodology.

One interesting possible error was also revealed. In this particular case, Adam's experimental evidence was strongly supportive of the hypotheses, whereas the literature provided strong evidence that the gene encoded a regulatory protein that controls the production of specific enzymes. As Adam's background knowledge did not include any information on regulatory proteins, Adam was incapable of forming abductive hypotheses about them. Significantly, the literature also provided a hint that Adam's original hypothesis might also be correct—the mapping between genes and enzyme function is many-to-many.

TOOLS FOR DESCRIBING SCIENCE

The tradition of reporting scientific research in papers has become outdated—it uses 17th- rather than 21st-century technology. Scientific investigations should be reported online using logical formalisms. Natural languages, while well designed to express poetry, are notoriously ambiguous and unsuited to expressing scientific knowledge.

It is also difficult to access for computer analysis—as the need for text mining proves. We do not argue for the elimination of natural-language descriptions because they aid human understanding. Rather, we advocate their augmentation with logical formalisms that fully and unambiguously describe scientific investigations. Their implementation would make scientific knowledge more explicit, reproducible, and reusable.

The development of robot scientists makes possible formalizing and recording all aspects of a scientific investigation using logic. The first step will be to define an explicit ontology that describes what exists. For the core organization of Adam's formalization, we used the EXPO[11] ontology of scientific experiments. This ontology formalizes generic knowledge about experiments: goals, hypotheses (if any), results, conclusions, and so on. Specifically, for Adam, we developed LABORS, a customized version of EXPO expressed using the description logic language OWL-DL.

The application of LABORS produces experimental descriptions in the Datalog logic-programming language. Its use resulted in a formalization of scientific investigations that involved more than 10,000 different units of experimental research. The result, a nested tree-like structure 10 levels deep, logically connects the experimental observations to the experimental metadata, as Figure 4 shows. Each structural level—studies, cycles, trials, tests, replicates, and so on—is characterized by a specific set of properties, such as the metadata shown in Figure 5. We believe that such nested structures typify scientific experiments, for which the testing of a top-level hypothesis requires that researchers plan many levels of supporting work. However, Adam's work is atypical in the scale and depth of its nesting. Most experiment annotation ontologies lack the design flexibility to deal with such complex investigations.

A typical path through the formalization would proceed as follows: The *investigation* into the automation of science has a part (the *investigation* into whether the robot scientist Adam can discover some novel science that has a part (the *study* aimed at finding the genes encoding orphan enzymes, which has a part (the *study* of the orphan enzyme E.C.2.6.1.39 in *S. cerevisiae*, which has a part (the "cycle" of studies of the gene YER152C, which has a part (the *study* of the gene YER152C, which has a part (the cycle 1, which has a part (the *trial* of the compound C00047/lysine, which has a part (the *test* of addition of C00047/lysine, which has a part (the *replicate* 1, which has >300 observations). This complete structure resembles a computer program trace and takes up 366 megabytes.
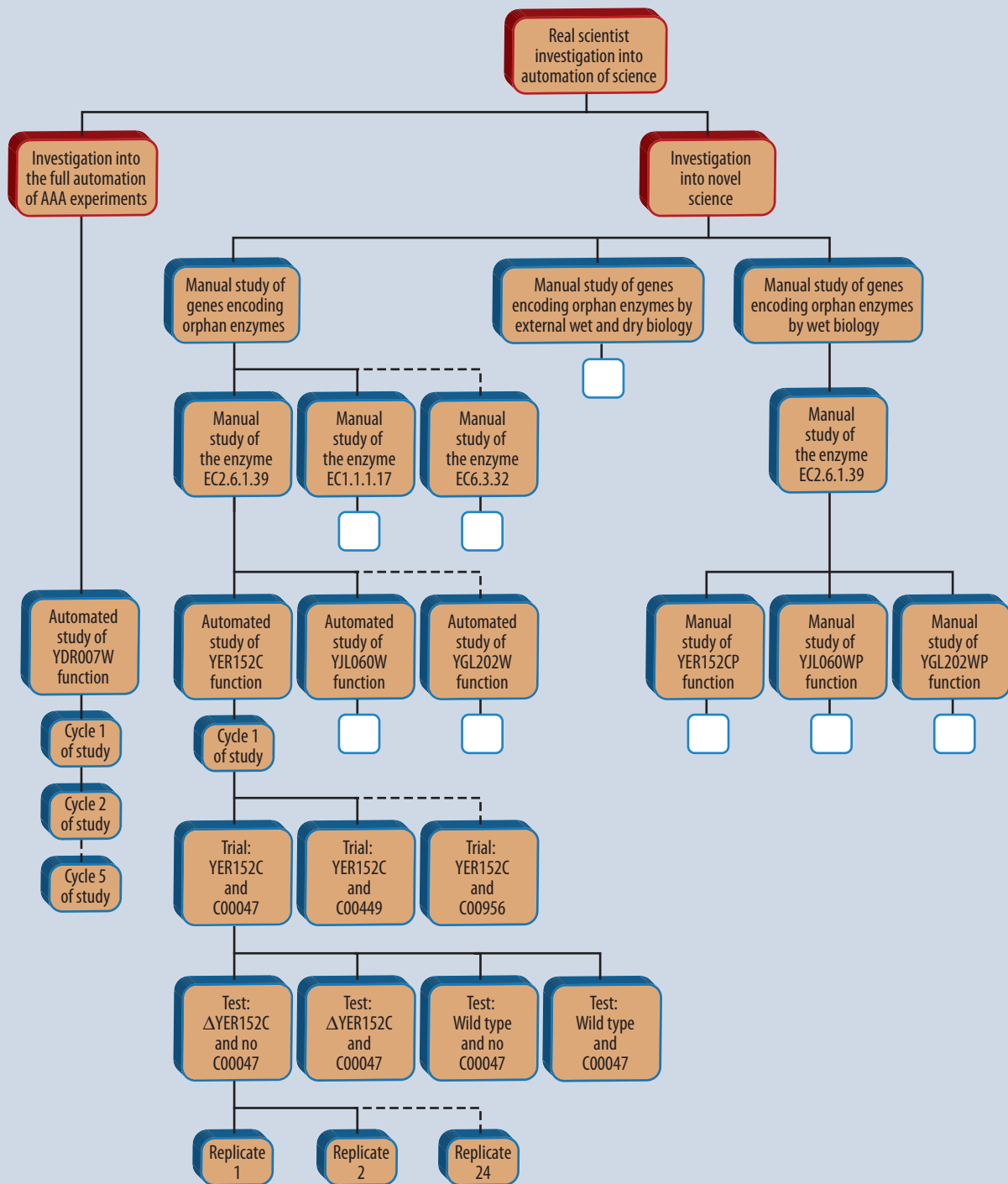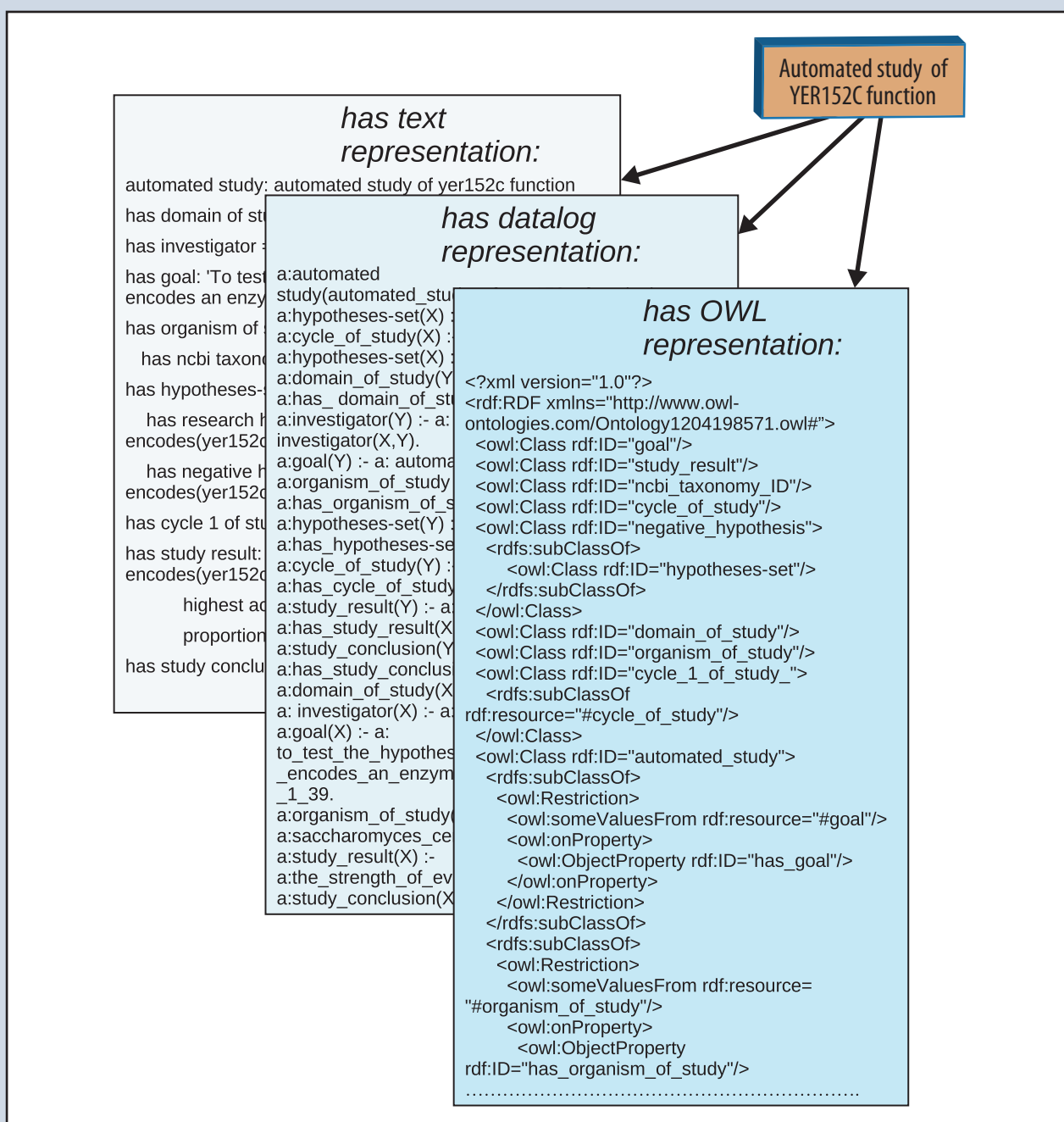
Figure 4. Connecting experimental observations and metadata: Structure of the robot scientist investigation (a fragment).

ANALYSIS

We argue that Adam has autonomously discovered novel scientific knowledge. Is this really justifiable? Let's start with the term *autonomously*. It would be misleading to give the impression that we can set up Adam and simply come back several weeks later to examine its conclusions. Adam is a prototype, and its hardware and software are brittle. So while Adam can run for a few days without human intervention, having a technician nearby is advisable in case problems occur. The integration of Adam's software modules also must be enhanced so that they work together seamlessly. However, the key point is

Automated study of YER152C function

has text representation:

automated study: automated study of yer152c function

has domain of st...

has investigator ...

has goal: 'To test...
encodes an enzy...

has organism of ...

　has ncbi taxon...

has hypotheses-...

　has research h...
encodes(yer152...

　has negative h...
encodes(yer152...

has cycle 1 of st...

has study result:...
encodes(yer152...

　　highest a...

　　proportion...

has study conclu...

has datalog representation:

a:automated
study(automated_stu...
a:hypotheses-set(X) ...
a:cycle_of_study(X) ...
a:hypotheses-set(X) ...
a:domain_of_study(Y...
a:has_ domain_of_st...
a:investigator(Y) :- a:
investigator(X,Y).
a:goal(Y) :- a: automa...
a:organism_of_study...
a:has_organism_of_s...
a:hypotheses-set(Y) ...
a:has_hypotheses-se...
a:cycle_of_study(Y) ...
a:has_cycle_of_study...
a:study_result(Y) :- a:...
a:has_study_result(X...
a:study_conclusion(Y...
a:has_study_conclus...
a:domain_of_study(X...
a: investigator(X) :- a:...
a:goal(X) :- a:
to_test_the_hypothes...
_encodes_an_enzym...
_1_39.
a:organism_of_study(...
a:saccharomyces_ce...
a:study_result(X) :-
a:the_strength_of_ev...
a:study_conclusion(X...

has OWL representation:

```
<?xml version="1.0"?>
<rdf:RDF xmlns="http://www.owl-
ontologies.com/Ontology1204198571.owl#">
  <owl:Class rdf:ID="goal"/>
  <owl:Class rdf:ID="study_result"/>
  <owl:Class rdf:ID="ncbi_taxonomy_ID"/>
  <owl:Class rdf:ID="cycle_of_study"/>
  <owl:Class rdf:ID="negative_hypothesis">
   <rdfs:subClassOf>
      <owl:Class rdf:ID="hypotheses-set"/>
   </rdfs:subClassOf>
  </owl:Class>
  <owl:Class rdf:ID="domain_of_study"/>
  <owl:Class rdf:ID="organism_of_study"/>
  <owl:Class rdf:ID="cycle_1_of_study_">
   <rdfs:subClassOf
rdf:resource="#cycle_of_study"/>
  </owl:Class>
  <owl:Class rdf:ID="automated_study">
   <rdfs:subClassOf>
    <owl:Restriction>
      <owl:someValuesFrom rdf:resource="#goal"/>
      <owl:onProperty>
       <owl:ObjectProperty rdf:ID="has_goal"/>
      </owl:onProperty>
    </owl:Restriction>
   </rdfs:subClassOf>
   <rdfs:subClassOf>
    <owl:Restriction>
      <owl:someValuesFrom rdf:resource=
"#organism_of_study"/>
      <owl:onProperty>
       <owl:ObjectProperty
rdf:ID="has_organism_of_study"/>
      ……………………………………………………………….
```

Figure 5. Connecting experimental observations and metadata: Example of one structural unit, with the metadata represented in free text, OWL (Web Ontology Language), and Datalog clauses.

that the new scientific knowledge was hypothesized and experimentally tested using an algorithmic/mechanical process that did not depend on human physical or intellectual effort.

The term *discovered* raises an old argument dating back to Ada Lovelace: Computers cannot originate anything and can only do what they are programmed to do. In his classic paper on "Computing Machinery and Intelligence," Alan Turing called this argument "Lady Lovelace's objection": "The Analytical Engine has no pretensions to *originate* anything."[12] It can *do whatever we know how to order it to perform*" (Lady Lovelace's italics).[12] Turing answered this objection by pointing out that predicting all the consequences of a computer program is impossible, and that the machines he had designed constantly took him by surprise. He then quoted *Ecclesiastes* (well worth a read)

"there is nothing new under the sun" to point out that all original work is based on existing foundations.

Perhaps the deepest answer Turing proposed is the analogy between children and learning machines that can interact with the real world. We agree with Turing's analysis. As it interacts with the real world, and is programmed to learn new scientific knowledge from it, Adam makes its own discoveries.

Finally, regarding the phrase "novel scientific knowledge," some of the mappings between genes and enzyme functions in *S. cerevisiae* that Adam has hypothesized and experimentally confirmed are certainly novel. Although this knowledge is modest, it is not trivial, and in the case of the genes encoding 2-aminoadipate transaminase it sheds light on, and perhaps solves, a 50-year-old puzzle.[11] It is of course still possible that some of Adam's conclusions are wrong, as all scientific knowledge is provisional. However, it seems unlikely that, given the manual checks and the positive controls, all the conclusions are in error.

We believe the robot scientist concept has enormous potential and could change how science is practiced. To extend Adam, we are developing software that lets external users propose hypotheses and experiments. We also plan to automatically publish the logical descriptions of experiments. The idea is to develop methods that enable teams of human and robot scientists to work together.

We have recently developed a new robot scientist called Eve, designed to automate steps in the drug screening and design process. Eve targets neglected Third World diseases such as malaria and schistosomiasis. As we designed Eve primarily to use yeast for drug assays, Adam and Eve will work together. Both are specialized systems. The next goal in hardware development will be to develop laboratory automation systems general enough to execute most standard laboratory tasks through changes in software, rather than having to reconfigure the hardware.

As science is an intellectual pursuit, the greatest research challenge will be to improve the robot scientist AI. At this point, comparing scientific research with the problems of playing chess can be instructive for developing general AI systems that can autonomously move about the world and pass the Turing test.

Automating science presents a challenge of intermediate difficulty. Computer programs now play chess at least as well as do the best humans, and computers now make undeniably beautiful moves. The success of computers in chess became possible because chess is an isolated world that can be modeled by a formal system. Abstract reasoning in chess shares many similarities with science. However, the essence of science is physical experimentation, and thus automating science is much harder than automating chess because the real world must be considered.

Despite science's intellectual prestige, developing robot scientists will probably be simpler than developing general AI systems because there is no essential need to take into account the social milieu. For example, we can safely assume that the real world is not trying to trick anyone.

The other advantage of using science as a testbed for AI systems is that we believe there is a continuum in the difficulty of scientific investigations, from the simple research of Adam through what most human scientists can achieve, up to the level of a Newton or Einstein. This view might be wrong, but at least one Physics Nobel Laureate, Frank Wilczek, is on record as saying that in 100 years' time, the best physicist will be a machine. **C**

References

1. G. Bell, T. Hey, and A. Szalay, "Computer Science: Beyond the Data Deluge," *Science*, vol. 323, 2009, pp. 1297-1298.
2. T. Berners-Lee, J.O. Hendler, and O. Lassila, "The Semantic Web," *Scientific American*, vol. 284, 2001, pp. 34-43.
3. R.K. Lindsay et al., *Applications of Artificial Intelligence for Organic Chemistry: The DENDRAL Project*, McGraw-Hill, 1980.
4. P. Langley et al., *Scientific Discovery: Computational Explorations of the Creative Process*, MIT Press, 1987.
5. J.M. Žytkow, J. Zhu, and A. Hussam, "Automated Discovery in a Chemistry Laboratory," *Proc. 8th Nat'l Conf. Artificial Intelligence* (AAAI 90), T. Dietterich and W. Swartout, eds., AAAI Press, 1990, pp. 889-894.
6. R.D. King et al., "Functional Genomic Hypothesis Generation and Experimentation by a Robot Scientist," *Nature*, vol. 427, 2004, pp. 247-252.
7. R.D. King et al., "The Automation of Science," *Science*, Apr. 2009, pp. 85-89.
8. L.N. Soldatova et al., "An Ontology for a Robot Scientist," *Bioinformatics 22* (special issue ISMB), 2006, pp. e464-e471.
9. C.S. Pierce, *Collected Papers of Charles Sanders Pierce*, Harvard Univ. Press, 1958.
10. T.M. Zabriski and M.D. Jackson, "Lysine Biosynthesis and Metabolism in Fungi," *Natural Product Report*, vol. 17, 2000, pp. 85-97.
11. L.N. Soldatova and R.D. King, "An Ontology of Scientific Experiments," *J. Royal Soc. Interface 3*, 2006, pp. 795-803.

12. A.M. Turing, "Computing Machinery and Intelligence," *Mind*, no. 236, 1950, pp. 433-460.

Ross D. King *is a professor at Aberystwyth University. King received a PhD in computer science from the Turing Institute, University of Strathclyde. Contact him at rdk@aber.ac.uk.*

Jem Rowland *is a senior lecturer at Aberystwyth University. Rowland received an MSc in physics from the University of Manchester and is a chartered engineer. Contact him at jjr@aber.ac.uk.*

Wayne Aubrey *is a postdoctoral research associate at Aberystwyth University. Aubrey received a PhD in biology from Aberystwyth University. Contact him at wwa@aber.ac.uk.*

Maria Liakata *is a postdoctoral research associate at Aberystwyth University and was recently awarded a fellowship from the Leverhulme Trust. Liakata received a DPhil in computational linguistics from the University of Oxford. Contact her at mal@aber.ac.uk.*

Magdalena Markham *is a PhD student at Aberystwyth University. Markham received an MSc in chemistry from Maria Curie-Sklodowska. Contact her at mgm07@aber.ac.uk.*

Larisa N. Soldatova *is an RC UK Fellow at Aberystwyth University. Soldatova received a PhD in computer science from the Far East Technical University, Russia. Contact her at lss@aber.ac.uk.*

Ken E. Whelan *is a postdoctoral research associate at Aberystwyth University. Whelan received a PhD in machine learning from the University of Abertay, Dundee. Contact him at knw@aber.ac.uk.*
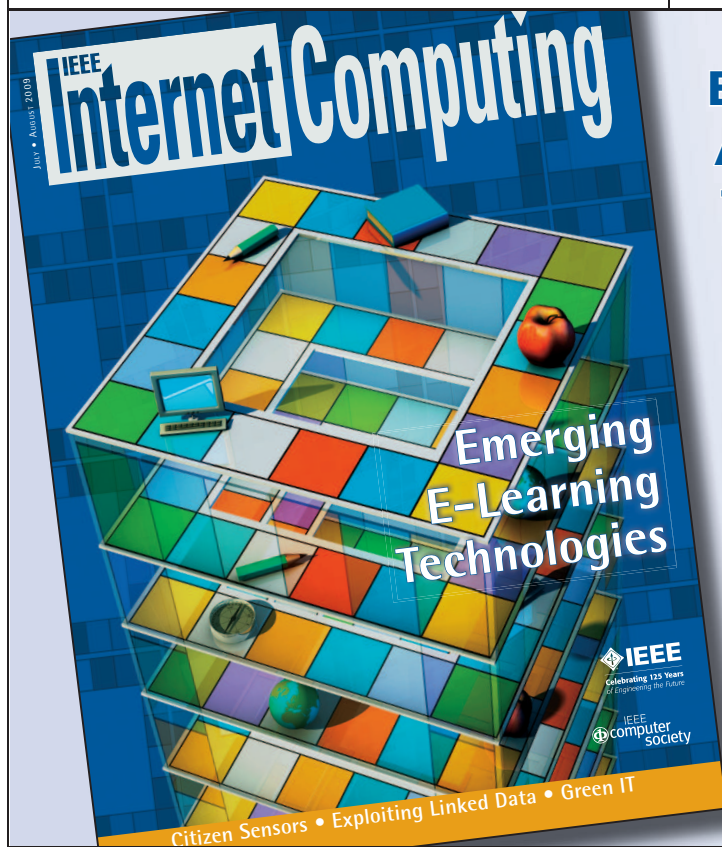
Amanda Clare *is a research fellow at Aberystwyth University. Clare received a PhD in computer science from the University of Wales, Aberystwyth. Contact her at afc@aber.ac.uk.*

Mike Young *is a professor at the Institute of Biological, Environmental, and Rural Sciences, Aberystwyth University. Young received a PhD in plant physiology from the University of East Anglia. Contact him at miy@aber.ac.uk.*

Andrew Sparkes *is a postdoctoral research associate at Aberystwyth University. Sparkes received a PhD in molecular biology from Portsmouth University. Contact him at nds@aber.ac.uk.*

Stephen G. Oliver *is a professor of systems biology and biochemistry at the University of Cambridge and director of the Cambridge Systems Biology Centre. Oliver received a PhD in microbiology and genetics from the National Institute for Medical Research. Contact him at steve.oliver@bioc.cam.ac.uk.*

Pınar Pir *is a research associate at Cambridge Systems Biology Centre. Pir received a PhD in chemical engineering from Boğaziçi University, Istanbul. Contact her at pp305@cam.ac.uk.*