

Artificial Intelligence and Scientific Creativity

Simon Colton and Graham Steel

July 16th, 1999

1 Introduction

There has been much recent success for AI systems undertaking creative tasks in scientific domains such as astronomy, biology, medicine, chemistry, physics and mathematics. In many scientific domains, we can build on the wealth of philosophical and computational studies into creative aspects of human intelligence, and use the abstract nature of the data to derive specialist algorithms for discovery. To achieve high level scientific creativity, the computational techniques employed are often domain specific. However, there are aspects of scientific creativity that can be identified and applied across domains. The process of developing the core notions of machine discovery in science is underway, as emphasised by, amongst others, the 1995 AAAI Spring Symposium dedicated to scientific discovery, the machine discovery workshops at ICDC'98 and ECAI'98, the 1997 issue of Artificial Intelligence dedicated to scientific discovery (vol. 91, issue 2), and the forthcoming machine discovery issue of the International Journal of Human-Computer Studies.

This process was continued at the 1999 AISB Symposium on AI and Scientific Creativity, which took place in Edinburgh, Scotland, in April. Papers presented at the symposium addressed the theoretical aspects of and computational possibilities for machine creativity. They also reported on systems implemented to achieve automated discovery in science. The intention of the symposium was that that the papers proposing models of scientific creativity would help researchers concerned with implementing discovery programs, and the papers discussing the successes and techniques employed in working systems will help researchers extract general frameworks for scientific machine discovery. This note is a survey of *current* research on creativity in science, and in particular the automation of discovery tasks in science. While it provides an opportunity to detail the papers presented at the symposium, we endeavour to extract commonalities between the research discussed and to present the work in a wider context.

To write creative programs which perform discovery tasks in science, it is important to understand the frameworks for creativity in science. These include philosophical frameworks, as discussed in section 2, which can draw on case histories and psychological studies, and computational frameworks available for machine discovery programs, which are discussed in section 3. Also, it is vital to learn from implementations of discovery programs, as discussed in section 4. Understanding the aims, the techniques employed and results achieved for a particular discovery program will enable new implementations to build on, extend and improve the work of others and will lead to more successful creative programs in science. One of the goals of the machine discovery community is to increase the quality and quantity of creative programs which act as assistants to scientists. It is hoped that the AISB Symposium and this survey will help us to take a step towards this goal.

2 Frameworks for Scientific Creativity

Creativity in science has provided philosophers with much fuel for discussion and debate. Recent work has involved philosophers producing and implementing computational models of creativity in science to clarify some of the issues involved, [13]. Three important areas in the philosophy of scientific creativity were covered in the symposium, namely the advancement of science through the introduction and maintenance of inconsistencies, the transfer of ideas from a known domain to a new domain and the social aspects of scientific discovery.

Lorenzo Magnani, from the Department of Philosophy at the University of Pavia, discussed the role of inconsistencies in the advancement of a scientific theory and the part that scientific creativity plays in generating and maintaining inconsistencies, [27]. Magnani discussed inconsistencies within the framework of explanation-driven, or abductive, reasoning, where certain facts or laws are inferred in order to explain some phenomenon. Inconsistencies can arise as empirical anomalies which the current theory cannot explain, or can be generated by the radical innovation of scientists who suggest a new theory to compete with the current theory in explaining the same phenomenon. It is usually necessary to maintain both theories until the possibility of rejecting one becomes feasible. For example, we maintain both Newtonian mechanics and general relativity, even though they are inconsistent in certain respects. Maintaining inconsistencies also involves creative processes, in particular, what Magnani calls the extraction of the *best possible information* - drawing the most accurate and scientifically valid conclusions that we can in the presence of two inconsistent hypotheses. Magnani presented a clear description of the role of creativity in proposing and maintaining inconsistencies in science.

Antoine Cornuéjols from the Laboratoire de Recherche en Informatique at the Université de Paris-sud addressed the problem of discovering and learning new concepts in a new, unknown domain, [11]. He introduced a new transfer mechanism called the “tunnel effect” by which new ways to interpret the world could be learned by relying on and relating to old ones. The tunnel effect was described as a four stage process in which (i) some association is made between a well known (source) domain and the new (target) domain, (ii) while interpreting the target domain, expectations and inferences from the source domain are used to fill in gaps in the interpretation, (iii) the model of the target domain is interpreted with no reference to the source domain and (iv) any re-conceptualisations necessary are undertaken. Cornuéjols also discussed how the tunnel effect often leads to further adaptation and conceptual learning, giving, as an example, the development of thermodynamics. Cornuéjols successfully demonstrated the tunnel effect with experiments in which high school students were asked to interpret the flow of electricity in a circuit in terms of energy transfers to and from an energy reservoir. While similar to the use of analogy in scientific creativity, the tunnel effect has the advantage of not needing a particular case to act as the source domain. As Cornuéjols pointed out, the use of one domain to help interpret another is important, as machine learning is usually undertaken alongside given concepts, which are not always available in a new domain.

David Gooding, from the Department of Psychology, University of Bath, is interested in understanding and simulating the social aspects of learning, discovery and problem solving in science, [1]. At the symposium, he detailed experiments with the EXPLORE system in which a society of agents, each representing an individual scientist, come to a conclusion about a problem. Each agent has a set of beliefs about a situation, where a belief is a hypothesis coupled with a subjective probability which indicates the agent’s confidence in the truth of the hypothesis. An agent’s beliefs will change by learning the beliefs of another agent and in the light of results of an action (experiment) designed to confirm a belief based on the current theory. Gooding used the Wason-Laird four card test [EK47] as an example to demonstrate how the set of agents made and adjusted hypotheses and beliefs, determined which actions to perform, carried out these actions and re-evaluated their beliefs until a consensus was reached. The simulations by the EXPLORE system and their explanation by Gooding gave insight into the social processes at work

when a theory is being developed in science. This provides important pointers to developers of scientific discovery programs wishing to enhance their model by accounting for social aspects of the creative process.

3 Considerations for Machine Discovery

Having considered the theoretical frameworks available to model scientific creativity, we must address the practical problems that arise when implementing programs designed to exhibit creativity in scientific domains. Two issues in particular, discovery and justification, need attention when writing creative programs. To be creative, discovery of one kind or another must be made, and we discuss the computational frameworks available to produce novel results in science in section 3.1. Once results have been found, it is essential that programs can assess the possible worth and application of the findings. The evaluation of machine-discovered propositions is detailed in section 3.2.

3.1 Computational Frameworks

Early work on modelling scientific discovery, such as the DENDRAL programs, [16], used heuristic search. Heuristic search consists of a space of possible states and a set of rules (heuristics) for guiding the search through the space. This approach can facilitate automated discovery in science: the search space consists of scientific findings, some of which are hoped to be novel to the domain, and the heuristics encode some domain-specific information about how to find the most interesting results. Heuristic searches can be effective when the search space is large, as extensive searches can bring to light results overlooked by humans, [40]. Recently, though, more sophisticated and specialised computational frameworks have been developed for automatic discovery, and we highlight three here, namely inductive logic programming, genetic programming and closed-loop discovery.

3.1.1 Inductive Logic Programming

Inductive Logic Programming (ILP) is a promising mechanism for machine discovery. In [33], Gordon Plotkin described a bottom-up approach to the problem of learning a concept from empirical evidence. This generalisation-based method led to the development of the GOLEM system by Muggleton and Feng, which has achieved some success in biological research, [32]. For example, when applied to the problem of predicting protein secondary structure, GOLEM out-performed both statistical and neural-net based methods.

There have been several other recent real-world applications of ILP techniques to scientific discovery problems. These include De Raedt's investigation into the biological classification of river water quality [14], in which the author generates classification rules suitable for use by an expert system, based on the numbers of certain invertebrates found in a river. In the field of molecular biology, ILP has been used successfully to predict mutagenicity of compounds [37] and to model structure-activity relations [22].

More recently, Muggleton has developed the general-purpose ILP language PROGOL, [31]. An exciting ongoing project with PROGOL was discussed at the symposium by Chris Bryant from the Department of Computer Science, University of York, [6]. The team have been working towards closed-loop scientific discovery, in which experiments are planned automatically and carried out by robots, with the results analysed using ILP techniques. The robotics technology required to achieve this has recently become available, and this is one of the first real-world applications of such a system. Using ILP techniques, hypotheses under test can be easily stated in English or illustrated diagrammatically, which allows for expert involvement in background knowledge refinement, and facilitates cross-checking in the relevant literature. It also means that final discoveries can be effectively disseminated to the scientific

community for criticism. Bryant described a possible application in the human genome project. Closed-loop discovery, discussed more fully below, using a framework such as ILP, is an idea expected to attract much interest as productive interactions between hardware and software become more common.

3.1.2 Genetic Programming

Genetic Programming (GP) is an extension of John Holland's Genetic Algorithm, [19]. According to John Koza, one of the initial developers of the technique, GP "performs automatic program synthesis using Darwinian natural selection and biologically inspired operations", [24]. In effect, this means the creation of a population of candidate programs which recombine, mutate and eventually evolve into better solutions for the problem. Genetic programming has grown rapidly in the light of encouraging results, and there are now international conferences in the field.

Recently, GP has been applied to several areas of science where a degree of creativity is required. For example, Lee Spector and his team have been using genetic programming to design algorithms for quantum computers, and claim to have produced several quantum algorithms superior to classical ones, [36]. At the AISB symposium, Forrest Bennett of Genetic Programming Inc. presented further recent results of attempts to tackle creative tasks. His group have applied their GP techniques to the design of analogue electrical circuits, [5]. As Bennett pointed out, this is certainly a domain in which successful human designers have been said to be inventive and creative. Bennett's team have achieved much success with a GP approach to this problem, including independent rediscovery of established patented designs, such as the design of an electronic thermometer. They reported 14 similar cases in which genetic programming produced results that were competitive with human designs. Bennett highlighted the 'illogical nature of creativity', and the need for a machine discovery program to take non-logical steps in order to achieve final success. A short psychological case study of Harold S. Black's invention of the negative feedback amplifier was given to illustrate such a 'eureka' step. Fifty years after the invention, Black still had no idea what led him to have the idea as he was crossing the Hudson river on his way to work. Genetic programming has the ability to take such non-logical steps in a controlled manner, and the team believes it is this ability which lies behind their recent successes.

3.1.3 Closed-Loop Discovery

A widely accepted model of scientific research involves a cycle of activity. Empirical data is obtained, and hypotheses formed from observation of the data. Experiments are then devised specifically to test these hypotheses, and in the light of the results the theory is revised and the cycle continues. While many computer-based tools have been developed to assist scientists in individual parts of the process, only more recently has work centred on producing systems capable of carrying out the entire cycle - the field of closed-loop discovery.

An early effort to automate closed-loop discovery was made by Hayes-Roth, [18], in 1983. He described a closed learning cycle, and an automation of the process based on the discussion in Lakatos' book 'Proofs and Refutations', [25]. Five heuristics for attempting to rectify a refuted theory were identified. In 1989, Subramanian followed up this work with a presentation of *factorisation*, a process whereby 'discrimination experiments' are identified - experiments designed specifically to refine the space of hypotheses.

Recent growth in the field has been due to the success of some individual programs, and the greater understanding that has emerged of how to effectively integrate programs for discovery purposes. As discussed above, Chris Bryant presented his group's work in closed-loop discovery in biology at the symposium (see section 3.1.1). It is hoped that their system will control experiments and construct and evaluate hypotheses while working on projects such as the human genome project.

Jan Żytkow of the Department of Computer Science, University of North Carolina at Charlotte, has been researching closed-loop methods in scientific modelling, [41]. Model construction is a complex and important scientific activity, distinct from, but easily confused with, theory formation. Żytkow clarified this distinction at the AISB Symposium: a theory is a result of analysis, and describes a simple element of nature, such as gravity or electromagnetism. A model is a result of synthesis, and typically describes a complex structure involving several theories and interactions. Żytkow described the motivation for a closed-loop approach to the problem of model construction. Given empirical data, it is unlikely that the correct model can be constructed at the first attempt. So, an automated system must include feedback, and a facility for model revision. Żytkow's system involves many levels of feedback loop, to account for the consequences later revisions may have for earlier decisions taken about the model. He demonstrated his work with a walk-through example of Galileo's investigation of motion on the inclined plane. A model for the motion was built using knowledge of mechanics, as well as empirical data generated using a robot arm rolling cylinders down the plane, and sensors to detect movement. Empirical equations obtained automatically from this data were compared against early models. Żytkow explained how the differences between model predictions and empirical equations led to several model improvements.

Francesco Amigoni and Marco Somalvico, from the Artificial Intelligence and Robotics Project, Politecnico di Milano, discussed the role of machines and humans in an intermediate approach to creative tasks called bipolar scientific discovery, [3]. This approach uses a machine both to provide support for discovery (by performing tasks which enable the user to find results), and to act as a model for the discovery (by forming an *operational* description of the results). One application of this approach discussed by Amigoni and Somalvico was the creative dynamic agency, a collection of agents (either workstations or robots), which can make and communicate inferences about the state of the world. Such an agency, called the MORO exploring agency was able to discover features and build a map of an office environment. The agency used cameras, mid-level feature detection algorithms and a communication and collaboration between two robots and two workstations. In using robots to physically explore an environment, this experiment not only exhibited closed-loop abilities, but also demonstrated collaboration and cooperation in an agency designed for discovery.

3.2 Assessing Machine Discoveries

Often, the generation of model results is the easy part of automated creativity in science. Programs can produce a plethora of data which satisfy the criteria for a solution to the problem, and some pruning of the data is necessary to save the user from having to deal with a deluge of information. As the implementation of a discovery program progresses, the pruning measures can often evolve into rules to guide heuristic search, and can sometimes specify the search space itself. Hence, how programs assess their results internally is an important part of machine creativity in science.

Simon Colton, from the Division of Informatics, University of Edinburgh, addressed the problem of enabling a mathematical discovery program to make an assessment of the results it produces, [9]. This is a difficult problem, as often programs have to make immediate assessments of results, but their true worth may only come to light after they have been thoroughly investigated for some time. Colton surveyed five mathematical discovery programs by looking at the methods they used and their degree of success. He paid particular attention to how the programs estimated the 'interestingness' of concepts and conjectures. This highlighted the fact that measures of the empirical plausibility, novelty, surprisingness, applicability, comprehensibility and utility of concepts and conjectures were common in these programs. Colton discussed each of these aspects of interestingness, as well as pointing out how the measures were used in the search for new results. A summary was given in the form of advice on assessing interestingness to future developers of machine discovery programs in mathematics.

Another important consideration when automating scientific creativity is how users can evaluate the

creative value of results from discovery programs, [40]. John McCarthy, from the Computer Science Department of Stanford University presented recent work on a problem he first presented in 1964 as ‘a tough nut for theorem provers to crack’ - the mutilated checkerboard, [28]. Two diagonally opposite squares are removed from a checkerboard, and then the problem is this: can the board be completely covered using 1 x 2 square dominoes laid only horizontally or vertically? The standard creative solution involves observing that the two removed squares were of the same colour, and that a domino always covers one black and one white square. McCarthy’s work has involved trying to express such creative solutions for problems in set theory. If a creative solution to a problem is, as McCarthy defines, one where additional concepts not present in the initial statement of the problem are introduced in order to solve it, then we can isolate these steps in the solution. This constitutes a small step towards programs that can produce creative solutions themselves.

Finally, scientific creativity programs can be used to assess human theories, and can sometimes provide hints at improvements. Jaap Kamps’ work at the Applied Logic Laboratory, University of Amsterdam, has centred around axiomatising social science theories in first order logic, [20]. Many informally stated theories have been treated this way in recent work, and by adopting an AI approach, Kamps has been able to reap further benefits. Using the Otter theorem prover and MACE model generator, he is able to evaluate a proposed formalisation. This facilitates discoveries about its consistency, soundness, and satisfiability. It can also lead to conclusions about its falsifiability, an essential property of a scientific theory. If no state of affairs can falsify a theory, then its validity is trivial, and the theory is worthless. A case study of a sociological theory due to Zeterberg illustrated the implementation of the technique. The formalisation process proceeds in a cyclical manner, in which revisions are repeatedly made in the light of apparent inconsistencies or unintended models. This process then has consequences for our evaluation of the original theory. For example, we may find that it contains a fundamental inconsistency, or that some restrictions can be discarded or relaxed, without compromising the soundness of the theory.

4 Machine Discovery Programs

We briefly discuss discovery programs and their successes in mathematics, chemistry, biology, medicine, physics and astronomy.

Discovery programs in mathematics include GRAFFITI, by Simeon Fajtlowicz, [15], and the HR program by Colton et al, [10].

GRAFFITI was designed to help its author make conjectures in graph theory, using heuristic search. It produces conjectures about inequalities between graph theory invariants, such as diameter, chromatic number and temperature. The conjectures are often difficult to settle, and because they can be used to cut down the time take to calculate the invariants, GRAFFITI has been very successful. The conjectures made have led to over 60 papers and 1 PhD thesis, and have been investigated by world-renowned mathematicians, such as the late Paul Erdos.

The HR program was designed to illuminate aspects of discovery processes in pure mathematics. Taking a general approach to concept formation and conjecturing, HR can work in any finite algebra, as well as in graph theory and number theory. The user provides a set of axioms for the domain to be investigated, and a way of breaking down objects in the domain. So, for example, in group theory the user might supply the Cayley table for some groups, or in number theory, the breakdown of the first 10 integers into their divisors. HR then proceeds to make manipulations on these tables using 10 production rules. Each manipulation yields a new concept, with a new table. HR measures the interestingness of these concepts using 5 heuristics. It is then able to make conjectures about the concepts, by observing concepts which have the same data table, but have been constructed in different ways. It uses a cycle of mathematical activity to help assess the interestingness of its results: it can pass first-order conjectures

to Otter [29] for proof, and if this fails, to MACE [30] for counterexample generation. If a theorem is proved, the concepts involved are adjudged to be more interesting than before. For more details of how the program operates, see [8]. HR has added to mathematics, most notably with the invention of 12 interesting and simple integer sequences, which were missing from the Online Encyclopaedia of Integer Sequences, [35].

In [39], Raul Valdés-Pérez describes the application of machine discovery techniques to the field of Chemistry. His MECHEM program was designed to produce good performance on a specific task, rather than with any considerations of generality in mind. However, it has strongly influenced his group's subsequent work in machine discovery, which is evidence for the applicability of the underlying ideas. MECHEM's job is to conjecture the structural transformations that chemically reacting molecules are undergoing in a particular reaction, based on empirical evidence. It has been applied to a number of problems, mostly in catalytic industrial chemistry, and has proposed some novel mechanisms for reactions. Development of the program is ongoing.

Physics is another area of science in which recent advances have been made. A good example is ASTRA, the system developed by Kocabas and Langley for generating process explanations in nuclear astrophysics, [23]. ASTRA has suggested some novel and interesting fusion reactions, and generated reaction pathways for helium, carbon, and oxygen that do not appear in the scientific literature. It has also suggested many uninteresting pathways as well, however, and work is underway to improve the system's inbuilt measures of interestingness. Also, a Japanese team, led by Tohru Araki, have been applying AI discovery techniques to geophysics problems. They have investigated a number of different approaches, including using neural nets to analyse geomagnetic substorms and wavelet analysis to detect geomagnetic pulsations, [4].

Astronomy is a field in which many recent applications of AI discovery programs have been successful. This is in part due to the large amount of data involved in modern astronomical observation, making it a field particularly suited to automated discovery. Kennefick, a researcher at CalTech, has developed an analysis tool called SKICAT, aimed at detecting quasars in digital sky surveys. Quasars are the brightest objects known in the universe, and as such can be detected at greater distances than any other astronomical bodies. This provides valuable evidence about the formation and early history of the first galaxies and large scale-structures to be formed in our universe. As reported in [21], SKICAT has found at least 5 new quasars, and its success rate is comparable with that of the top human research teams.

Cheeseman et al, [7], have also applied an automated analysis tool to a large-volume data problem. Taking readings from the Infrared Astronomical Satellite, they applied their AUTOCLASS program to produce a new classification system for the Atlas of Low Resolution IRAS Spectra. Their research involved a large degree of human intervention (for example, they decided to run their classifier again, this time on their classification results from the first pass, to produce a set of fewer classifications), but they believe that their tool was of vital importance in this successful research. According to them, these techniques 'have a bright future in the arena of astronomical classification problems'.

In biology, Alberdi and Sleeman have developed a system for revising taxonomies, ReTAX, [2]. Given a pre-established taxonomy and some objects which show the classification to be inconsistent, ReTAX applies a system of refinement operators to propose a new taxonomy in which the inconsistencies have been resolved. ReTAX has replicated historically significant revisions to the taxonomy of the Ericaceae family.

Machine discovery techniques have also been applied to the field of medicine. At the AISB Symposium, Mauro Gaspari described an application of AI techniques to the study of multiple sclerosis, [17]. He first described the complexity of the domain and of the knowledge involved in it. There are still many problems without a satisfactory solution, including the ethiology and the prognostic criteria for the disease. His multi-disciplinary team has worked on the derivation of a complete knowledge-acquisition procedure, identifying the benefits this would offer to the process of diagnosis and treatment of the dis-

ease. A key research direction is the development of progression markers: a transitional phase in the disease has been observed after one or two years. Spotting the onset of this transition early is essential for adequate treatment, which in many cases can prevent evolution of the disease into highly disabling forms. Although this phase has been recognised in the literature in the last few years, it is yet unclear as to whether an adequate progression marker exists. The team's AI approach to this problem has showed promise, and it is sincerely hoped that machine discovery techniques will help towards breakthroughs in the understanding of this disease.

In another medical application, Swanson and Smalheiser describe a system for facilitating discovery across specialities, [38]. If a set of articles in the literature reports an association between variables A and B, and another set of articles describes a relationship between B and C, but nothing has appeared concerning the possibility of a link between A and C, and that link would be of scientific interest, then a gap has been found worth investigation. Their system, ARROWSMITH, aids the user in identifying such situations by looking at citation patterns, and so can lead to new discoveries.

Other programs already discussed in this report include Bryant's system, which uses ILP techniques to facilitate closed-loop discovery in pharmaceutical research (section 3.1.1), Żytkow's system for developing scientific models from empirical data (section 3.1.3), and Amigioni's system for bipolar discovery (section 3.1.3).

5 Conclusions

Computers are used in many ways to facilitate creativity in scientific domains. AI systems have been used with considerable success in at least the following areas of scientific creativity:

1. Producing new descriptions of phenomena, such as the production of reaction mechanisms in chemistry, [39] and particle physics [23], or the invention of descriptions for protein structures, [34], and fold classes of proteins, [31], in molecular biology.
2. Inventing new concepts and categorisations, spotting empirical trends and making new hypotheses, such as the conjectures being made by computers in mathematics, [10], [15].
3. Spotting examples of a phenomenon, as happened recently with the successful automated identification of distant quasars in astronomy, [21].
4. Designing experiments to test hypotheses and performing closed-loop discovery to illustrate the progress of a theory, [6], [41].
5. Making explicit unquestioned assumptions in a domain, such as those identified about leukemia, [12].

There is much real potential for automated discovery programs to produce findings which have a great impact on science. More and more programs are being written to act in creative collaboration with scientists, and this trend promises the research area a bright future. As Darden observes in [13], 'It is becoming more difficult to locate computational discovery work because much of it is published in scientific journals—a good sign that the methods are producing results of interest to practising scientists.' To capitalise on the growing need for computing power to enhance and facilitate discovery in science, the process of generalising machine discovery aims, practises and terminology must continue, as in the papers by Langley, [26], and Valdés-Pérez, [40].

We have identified three areas of scientific creativity to be addressed. Firstly, we must study the frameworks for creativity in science, including philosophical discussions and psychological and historical case studies. We then have to determine what special considerations are needed to enable automated

discovery in science, in particular the computational frameworks available and how to assess findings made by computer. Finally, we can study the aims, methods and results from discovery programs which play an active role in science.

All three of these areas were addressed at the AISB'99 Symposium on AI and Scientific Creativity, which flourished in the company of symposia and workshops addressing creativity in other fields. We look forward to more discussions of scientific creativity at future conferences, the emergence of more sophisticated discovery programs and the presentation of important results discovered by them.

Acknowledgements

There were two joint sessions of the symposium with the Automated Reasoning Workshop, as there was considerable overlap in interest between the two meetings. We would like to thank the members of the Automated Reasoning Workshop, in particular the chair of that event, Manfred Kerber, for making us very welcome. Thanks also to Josh Singer for his help reviewing abstracts for the symposium, and to Dan Yarlett for comments on an earlier draft of this report. The local organisers of the convention, Helen Pain, Andrew Patrizio and Geraint Wiggins worked tirelessly to make the whole event a success, and they have our sincerest gratitude, as do the team of student helpers who made the logistics manageable. Finally, we would like to thank the programme committee, Dr. Vincent Corruble, Prof. Stephen Muggleton, Prof. Thomas Nickles, Prof. Derek Sleeman, Dr. Raul Valdés-Pérez and Prof. Jan Żytkow for their substantial efforts in bringing this symposium to fruition. Not only did their reputations help attract important papers, but also their reviewing and helpful comments throughout have been of the highest quality.

References

- [1] T. Addis and D. Gooding. Learning as collective belief-revision: A simulation of model-based reasoning about disparate phenomena. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 19–28, Edinburgh, Scotland, April 1999.
- [2] E. Alberdi and D. Sleeman. Retax: a step in the automation of taxonomic revision. *Artificial Intelligence*, 91:257–279, 1997.
- [3] F. Amigioni, V. Schiaffonati, and M. Somalvico. Dynamic agencies and creative AI and scientific discovery. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 72–81, Edinburgh, Scotland, April 1999.
- [4] T. Araki et al. Application of discovery science to solar-terrestrial physics. In Motoda Arikawa, editor, *Discovery Science*, page 451, 1998.
- [5] F. H. Bennett, J. R. Koza, M. A. Keane, and D. André. Genetic programming: Biologically inspired computation that exhibits creativity in solving non-trivial problems. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 29–38, Edinburgh, Scotland, April 1999.
- [6] C. H. Bryant, S. H. Muggleton, C. D. Page, and M. J. E. Sternberg. Combining active learning with inductive logic programming to close the loop in machine learning. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 59–64, Edinburgh, Scotland, April 1999.
- [7] P. Cheeseman et al. A Bayesian classification of the IRAS LRS atlas. *Astronomy and Astrophysics*, 222, 1989.

- [8] S. Colton. HR - automatic concept formation in pure mathematics. <http://www.dai.ed.ac.uk/~simonco/research/hr>, 2000.
- [9] S. Colton and A. Bundy. On the notion of interestingness in automated mathematical discovery. In *AISB'99 Symposium on AI and Scientific Creativity*, Edinburgh, Scotland, April 1999.
- [10] S. Colton, A. Bundy, and T. Walsh. HR: Automatic concept formation in pure mathematics. In *IJCAI '99*, 1999.
- [11] A. Cornuéjols, A. Tiberghien, and G. Collet. Tunnelling effects in cognition: a new mechanism for scientific discovery. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 9–18, Edinburgh, Scotland, April 1999.
- [12] V. Corruble and J. Ganascia. Using formal induction techniques to aid research on human leukemia. *Blood Cells*, 19, 1994.
- [13] L. Darden. Recent work in computational scientific discovery. In M. Shafto and P. Langley, editors, *Proceedings of the Nineteenth Annual Conference of the Cognitive Science Society*, pages 161–166, Mahwah, New Jersey, 1997.
- [14] S. Dzeroski and I. Bratko. Applications of inductive logic programming. In *Advances in Inductive Logic Programming*, pages 65–80. IOS Press, 1996.
- [15] S. Fajtlowicz. On conjectures of Graffiti. *Discrete Mathematics* 72, 23:113–118, 1988.
- [16] E. Feigenbaum and Buchanan. B. DENDRAL and Meta-Dendral: Roots of knowledge systems and expert system application. *Artificial Intelligence*, 59(1-2):223–240, 1993.
- [17] M. Gaspari, L. P. Machello, C. Scandellari, and S. Stecchi. Multiple sclerosis: A domain for the application of advanced AI techniques. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 39–43, Edinburgh, Scotland, April 1999.
- [18] F. Hayes-Roth. Using proofs and refutations to learn from experience. In *Machine Learning: An Artificial Intelligence Approach*. Springer-Verlag, 1983.
- [19] J. Holland. *Adaption in Natural and Artificial Systems*. University of Michigan Press, 1975.
- [20] J. Kamps. On the process of axiomatising scientific theories: Using justification criteria in the context of discovery. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 49–58, Edinburgh, Scotland, April 1999.
- [21] J. D. Kenefick, R. R. De Carvalho, S. G. Djorgovski, M. M. Wilber, E. S. Dickson, and N. Weir. The discovery of five quasars at $z > 4$ using the second palomar sky survey. *Astronomical Journal*, 110(1), July 1995.
- [22] D. King, A. Srinivasan, and M. J. E. Sternberg. Relating chemical activity to structure: an examination of ILP successes. *New Generation Computing*, 13:411–433, 1995.
- [23] S. Kocabas and P. Langley. Generating process explanations in nuclear astrophysics. In *Proceedings of the Machine Discovery Workshop, ECAI*, 1998.
- [24] J. Koza. Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems. Technical Report STAN-CS-90-1314, Dept. of Computer Science, Stanford University, June 1990.

- [25] I. Lakatos. *Proofs and Refutations: The Logic of Mathematical Discovery*. CUP, 1976.
- [26] P. Langley. The computer-aided discovery of scientific knowledge. In *Proceedings of the first international conference on discovery science*, 1998.
- [27] L. Magnani. Inconsistencies and creative abduction in science. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 1–8, Edinburgh, Scotland, April 1999.
- [28] J. McCarthy. Creative solutions to problems. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 44–48, Edinburgh, Scotland, April 1999.
- [29] W. McCune. The OTTER user's guide. Technical Report ANL/90/9, ANL, 1990.
- [30] W. McCune. A Davis-Putnam program and its application to finite first-order model search. Technical Report ANL/MCS-TM-194, ANL, 1994.
- [31] S. Muggleton. Inverse entailment and PROGOL. *New Generation Computing*, 13(3,4), 1995.
- [32] S. Muggleton and C. Feng. Efficient induction of logic programs. In *Proc. of the First Conf. on Algorithmic Learning Theory*, Tokyo, 1990.
- [33] G. Plotkin. *Automatic Methods of Inductive Inference*. PhD thesis, Edinburgh University, 1971.
- [34] S. Schulze-Kremer. Genetic algorithms for protein tertiary structure generation. In *Parallel Problem Solving from Nature 2*. North Holland, 1992.
- [35] N. Sloane. *The Online Encyclopedia of Integer Sequences*. <http://www.research.att.com/~njas/sequences>, 1998.
- [36] L. Spector, W. Langdon, U. O'Reilly, and P. Angeline, editors. *Advances in Genetic Programming 3*. MIT Press, 1999.
- [37] A. Srinivasan, S. Muggleton, R. D. King, and M. J. E. Sternberg. Mutagenesis: ILP experiments in a non-determinate biological domain. In *Fourth International Workshop on Inductive Logic Programming*, pages 217–232, 1994.
- [38] D. R. Swanson and N. R. Smalheiser. An interactive system for finding complimentary literatures: a stimulus to scientific discovery. *Artificial Intelligence*, 91(2), 1997.
- [39] R. Valdés-Pérez. Machine discovery in chemistry: New results. *Artificial Intelligence*, 74:191–201, 1995.
- [40] R. Valdés-Pérez. Principles of human computer collaboration for knowledge discovery in science. *Artificial Intelligence*, 107(2), 1999.
- [41] J. Żytkow. Model construction: Elements of a computational mechanism. In *AISB'99 Symposium on AI and Scientific Creativity*, pages 65–71, Edinburgh, Scotland, April 1999.