REPORT ON LPNMR '97
28-31 July, 1997
Dagstuhl, Germany

Robert Milnikel, Jr.
Mathematical Sciences Institute, Cornell University
407 College Avenue, Ithaca, NY 14850, U.S.A.

November 3, 1997

The fourth International Conference on Logic Programming and Nonmonotonic Reasoning (LPNMR '97) was held in the outstanding facilities at Schloß Dagstuhl, Germany from Monday the 28th through Thursday the 31st of July, 1997. This year's installment of LPNMR reflected a new stage in the development of this cross-disciplinary field, with demonstrations of ten implemented systems joining the submitted papers, invited talks and panel discussions. Previous meetings were held in 1991 (Washington, D.C., USA), 1993 (Lisbon, Portugal), and 1995 (Lexington, Kentucky, USA). Information on this fourth meeting can be obtained on the world wide web (WWW) at `<URL:http://www.uni-koblenz.de/~lpnmr97/>`.

Of the 54 participants, 35 traveled to the Saarland in the southwest of Germany from other countries, including places as far-flung as Russia, Israel, and the United States. 58 authors combined for 19 accepted papers and 10 systems descriptions which were selected for presentation at the meeting and publication in the proceedings (Springer-Verlag LNAI 1265).

After a welcome from conference chairs Jürgen Dix and Ulrich Furbach of the University of Koblenz, V.S. Subrahmanian braved an early morning and system incompatibilities to give the first invited address, "Towards a Theory of Interestingness." After mentioning applications in data mining, profiling, and screening, Dr. Subrahmanian went on to present an outline of the structure and semantics of interestingness programs. He and his research team are working on building interestingness servers on top of the reasoning system HERMES. Unfortunately, a summary of this talk was not included in the conference proceedings, but further information on Dr. Subrahmanian's research and the HERMES project can be found at the URLs `<URL:http://www.cs.umd.edu/~vs/>` and `<URL:http://www.cs.umd.edu/projects/links/>`.

The morning's session continued with a presentation by Nicola Leone of joint work with Francesco Buccafurri and Pasquale Rullo. He described an extension of disjunctive datalog by two types of constraints: strong constraints which must be satisfied, and a system of weak constraints which must be satisfied if possible. In addition to examples of problems well represented by this extension, Dr. Leone analyzed the complexity of computation in such a system. Chris Pollett followed with work (joint with Jeff Remmel) on the complexity of nonmonotonic logics with quantified Boolean constraints. Also discussed were the addition of constraints to circumscription and the relative succinctness of the above mentioned formalisms versus other formalisms. The morning closed with a complexity analysis of transformation algorithms for computing the well-founded models of non-disjunctive logic programs, and a comparison with the alternating fixpoint procedure. Ulrich Zukowski presented this joint work with Stefan Brass and Burkhard Freitag.

Is nonmonotonic reasoning always harder? Uwe Egly (in work with Hans Tompits) answered "No." Indeed, as a completion formula can simulate the cut rule, there are cases where circumscription or completion can provide a non-elementary decrease in proof length over the cut-free sequent calculus. Riccardo Rosati followed with a look at the computational properties of the propositional fragment of Levesque's logic of only knowing. In particular, it is at the second level of the polynomial hierarchy, and so is comparable to several other nonmonotonic formalisms. In the final non-systems talk of the day, Jennifer Seitzer (in joint work with John Schlipf) looked at classes of normal logic programs for which we could be guaranteed to compute models in linear time. These classes were obtained by limiting the number of times a variable appears in either the head or body of a rule.

The last session of the day was the first of demonstrations of implemented systems for logic programming with nonmonotonic reasoning. Chandrabose Aravindan presented the system DisLoP. This work with Jürgen Dix and Ilkka Niemelä builds on the PROTEIN theorem prover a system which will be able to handle disjunctive logic programs and nonmonotonic negations. Different modules deal with positive programs, minimal model reasoning, and D-WFS semantics. Michael Schroeder demonstrated REVISE, a system for revising contradictory extended logic programs developed with Carlos Viegas Damásio and Luis Moniz Pereira. Examples of performance in the realm of circuit diagnosis complemented the discussion of the system's algorithms. Gerald Pfeifer closed the session by talking about the architecture for a Disjunctive Deductive Database system. The modules discussed were a rules and graphs handler, a procedure for efficient ground instantiation of programs, a model generator, and a separate model checker. This was joint work with Thomas Eiter, Nicola Leone, Cristinel Mateis, and Francesco Scarcello.

After a full day's program, nothing could be more relaxing or invigorating than a concert in the castle's ornate, baroque music room. How fortunate we all were, therefore, that Jennifer Seitzer, who had presented a paper a few hours earlier, was willing to share her musical gifts with us in the evening. Before turning to computer science, Ms. Seitzer had earned a degree in piano performance, and displayed considerable talent and technique in a program which included Bach, Beethoven, and Chopin. The concert was so well received that an evening of music may become a tradition at future LPNMR's.

Our second invited speaker, Mirosław Truszczyński, started Tuesday with a talk entitled "Automated Reasoning with Nonmonotonic Logics." He first talked about basic algorithmic techniques used in building automated reasoning systems for nonmonotonic logics and gave an overview of complexity results. Then, in keeping with the new focus on implementations, he described several of the major working systems, including DeReS which was developed by Dr. Truszczyński and his collaborators at the University of Kentucky. Finally, he discussed the need for thorough testing of current systems and the need for real application domains. A partial answer to the first concern is the University of Kentucky's TheoryBase, a generator of test theories and programs, but the question of real applications will be critical in the next several years.

Howard Blair opened Tuesday's first session of submitted papers with a description of covered normal logic programs as cellular automata. If looked at in this way, we can see more clearly the ways in which programs simulate each other; in particular, we can see how to simulate covered programs with Horn programs. This was joint work with Fred Dushin and Polar Humenn. Tomi Janhuen presented a generalization of Moore's autoepistemic logic with separate modalities for beliefs and disbeliefs. One distinct advantage of this approach is that the relationship between autoepistemic and default logic is quite straightforwardly captured. In the final talk of the morning, William Rounds (in joint work with Guo-Qiang Zhang) used powerdomains to encode constraints in default logics. Conclusions about extensions in these logics were drawn from general theorems about Scott domains.

Przymusinski's Autoepistemic Logic of Beliefs and the associated static semantics were put in the context of a more general formalism by Alexander Bochman. These semantics are part of a broad family of nonmonotonic formalisms constructed in accordance with a common completion schema. Pierro Bonatti then introduced an extension of resolution for skeptical stable model semantics. By looking at a strict subset of program rules, floundering can be avoided in some cases with this approach. The session closed with Thomas Eiter presenting joint work with James Lu and V.S. Subrahmanian. He discussed the usefulness of Turi's notion of a constrained atom to non-ground representations of both stable model and well-founded semantics. This has consequences for partial pre-computations at compile time; associated algorithms were presented.

Rounding out Tuesday afternoon were three more systems demonstrations. Ulrich Zukowski started the late afternoon session with LOLA, a deductive database system which is joint work with Burkhard Freitag. LOLA partitions a program into pieces which can be evaluated bottom-up after a magic set transformation, then communication between these is compiled into function calls which realize top-down control. This approach works for programs with modularly stratified negation. ACLP, a system based on Abductive and Constraint Logic Programming, was then presented by Antonis Kakas. This project (joint work with Costas Mourlas) integrates abduction into a constraint logic programming environment, building on the language ECLiPSe. Not only was the theory behind the language presented, but experimental data showing reasonable performance times in various "real-world" problems. An object-oriented approach was taken by Paul-Thomas Kandzia in his system FLORID (F-LOgic Reasoning In Databases). Nonmonotonic reasoning was not the focus of the development of this F-logic based system, but can be handled by means of user-defined stratification of inheritance relations.

Tuesday was the big day for systems demonstrations, and the participants returned in the evening for four more before a few took a well-deserved sauna at the end of a long day. First up was Gerd Neugebauer with GLUE, a project undertaken with Dorothea Schäfer. GLUE (Graphical Logic programming Utilization Environment) had by far the most polished appearance of any system demonstrated, and has as a main feature the ability to interact easily with heterogeneous outside sources of information. The deductive kernel of the system can also interact with theorem provers such as PROTEIN to broaden its capabilities. Ilkka Niemelä followed with Smodels, which works with well-founded and stable model semantics for range-restricted function-free normal programs. In this joint project with Patrik Simons, extensive pruning is done during backtracking searches, using approximations to stable models and thus running in linear space. This speeds up performance considerably in comparison with other systems for computing stable models. Efficiency was still a theme, but well-founded semantics were the focus of David S. Warren's XSB (joint work with Prsad Rao, Konstantinos Sagonas, Terrance Swift, and Juliana Freire). In addition to being a full Prolog system, the XSB computes well-founded semantics using SLG resolution and handles HiLog terms. Much of the speed of the system derives from its extensive tabling and indexing algorithms. The final system presented at the conference was XRay, joint work of Torsten Schaub and Pascal Nicolas. Prof. Schaub explained that as an enhancement of PTTP (Prolog Technology Theorem Proving) which can handle different types of default information, XRay is both a general implementation platform for semi-monotonic default logics and a logic programming system which integrates disjunction and several types of negation. The use of automated theorem proving technology enhances the overall performance of the system.

Wednesday saw only one talk given, that by invited speaker Bruno Buchberger, whose address was titled "Computing, Solving and Proving: A report of the Theorema project." Dr. Buchberger put forth the proposition that since mathematics consists mostly of computing, solving equations, and writing proofs, and since these activities are closely related, a good computer mathematics system should be capable of engaging in all three of these activities in a unified environment. Computer algebra systems are quite strong in both the computing and solving strands, but not in writing proofs; most theorem provers have the opposite strengths and weaknesses. The Theorema project is making progress toward building a system which has the strengths of both a computer algebra system and a theorem prover by building a higher-order predicate logic theorem prover over the base of the Mathematica computer algebra system. Dr. Buchberger described some of the details of Theorema, its potential uses, and directions for further development. He also discussed the relation of his system to some of the systems presented in Monday's and Tuesday's sessions.

The day continued with two panel discussions. The first, on the implementation of logic programming systems capable of nonmonotonic reasoning, was moderated by Jürgen Dix. In discussing the general goals and expectations of implemented systems, a point of general agreement was articulated by Antonis Kakas: we need not only to be building on each other's theoretical work, but also to be working with and building on each other's systems. Alexander Dikovsky raised the question of coupling systems with widely used programming languages and making sure the systems are available for a variety of platforms; David Warren said that this is being done with some systems. In addressing the question of why the current systems are not being widely used, some of the concerns raised were: applications often do not come in representations and sizes that are immediately compatible with current systems, so we should work on connecting real-world problems and more abstract concepts from logic programming (Torsten Schaub and William Rounds); engineers are not currently taught logic, so we need systems which can be brought into the classroom and used in AI and database classes (Michael Gelfond); we should look at the history of Prolog's gradual acceptance for some direction (Luís Pereira). Some related suggestions for marketing systems better included writing interfaces which did not necessitate facility with the underlying logic (Mirosław Truszczyński) and publicizing problems in nonmonotonic logic as frameworks into which other computations can be coded (as has been done with SAT) (William Rounds). These are only a few of the most general points which were raised in the 90 minute discussion, which also included substantial time given to specific systems and applications.

The afternoon's panel had a more general focus on the current and future relationship of logic programming and nonmonotonic reasoning, as well as the future of the conference itself. Concerning the latter point, the outgoing board of the LPNMR conferences (Anil Nerode, chair; Victor Marek; and V.S. Subrahmanian, who jointly chaired the afternoon's discussion) announced the new board (Georg Gottlob, chair; Michael Gelfond; and Anil Nerode) and the locations of the next two meetings (El Paso, Texas, USA in 1999; and Vienna, Austria in 2001).

The discussion itself at the afternoon panel was wide-ranging enough to make an accurate summary difficult; in particular, so many viewpoints were raised on any particular issue that to present each with proper attribution would require something approaching a complete transcription. V.S. Subrahmanian started the discussion with the following questions: There seems to be a drop in interest from students. Is this due to a lack of implementations? A lack of applications? Is our emphasis on stable and well-founded semantics correct if no one outside the LPNMR community seems to be taking much of an interest? What are some alternatives? One response was that the most marketable aspect of nonmonotonic reasoning, its uses in belief revision, has not been explored thoroughly. Another was to suggest directing research toward addressing real applications directly, including questions of which semantics are best suited for various types of applications. (It was pointed out that much of what has been done so far in looking at relationships between formalisms, complexity results, etc., was necessary, but that if the field does not reach out it will implode.) In looking at the issue of the place of logic programming and nonmonotonic reasoning in the computer science and artificial intelligence communities, more direct research into connections with natural language processing was suggested, as was trying to get more LPNMR-type papers accepted at and subjects discussed at more general meetings. Two important distinctions were drawn toward the end of the discussion. The first was between static and dynamic logics, with the point being that we have looked almost exclusively at the former and that the latter is where the future of the subject may lie. The second addressed the context of the research, looking at its scientific value versus its economic and political value. Dr. Subrahmanian closed the panel with a caveat that the road from a theory to effective applications can be long and difficult, and that one should not get discouraged along the way.

The working day finished early on Wednesday to give the participants a chance to take advantage of some of the opportunities afforded by the location of the meeting. In particular, the expedition to one of the oldest surviving and longest running iron refineries in Germany was a brief (and scenic) bus ride away. The knowledgeable guides helped put the overwhelming size of some of the machinery in perspective. (There may never have been such a large collection of computer scientists wearing protective hard hats.) The expedition concluded with the conference banquet, held at a delightful outdoor restaurant with musicians serenading the participants.

The final invited lecturer was Michael Gelfond, whose talk was titled "Towards a Systematic Approach to Representing Knowledge in Declarative Logic Programming." He listed three requirements for a reasonable representation of knowledge: an expressive language with a precisely defined entailment relation; algorithms to compute or approximate these entailments; and a methodology for representing knowledge in these languages. The main thrust of the talk was how we can develop the methodology for representing knowledge by looking at the methodology of procedural programming: developing the program together with its specifications, using correctness concerns as a heuristic guide for programming, and viewing programming as a refinement of specifications. The talk then proceeded with demonstration and argument by examples, mostly dealing with programs for handling inheritance hierarchies, including how the closed world assumption is often implicit in how we deal with these nets.

Thursday's morning session was one of the most unified in subject matter – all three talks dealt with belief revision at some level. Luis Moniz Pereira began by presenting joint work with Carlos Viegas Damásio on paraconsistent semantics and the detection of the support of a contradiction in these semantics. He also discussed how to block the propagation of inconsistencies in this context. Alexander Dikovsky's talk (on joint work with Michael Dekhtyar and Nicolas Spyratos) was more explicitly about updating. The idea of their new method is to restore integrity constraints to a database with the minimum of necessary changes. Dr. Dikovsky pointed out that some existing revision programs can be quite effective on knowledge bases while not working as well on databases. Cees Witteveen discussed revision not of databases, but of nonmonotonic theories. In work with Wiebe van der Hoek, he discussed how one can systematically and syntacticly revise a nonmonotonic theory if the intended set of models for the theory is empty. This approach depends, of course, on the semantics of the nonmonotonic rule system. Dr. Witteveen concluded with a more concrete discussion of the application of these ideas to logic programs.

Simone Contiero opened the final session of LPNMR '97 with a talk on the composition of programs, joint work with Antonio Brogi and Franco Turini. In software design, one topic of recent interest has been a system for allowing extant programs to work together. Dr. Contiero discussed a framework for combining general logic programs with several meta-level operations on programs (such as union, intersection, and restriction) and presented a three-valued semantics of the resulting expressions. Helmut Veith followed with a talk on a strongly related topic, that of being able to combine separate modules of a logic program even when the modules of the program have introduced new logical connectives of their own. Dr. Veith, Thomas Eiter, and Georg Gottlob take a model theoretic approach to this problem based on the use of generalized quantifiers – one can even look at a logic program with or without negation as a generalized quantifier. Vyacheslav Petukhin then spoke on a new nonmonotonic operator, universally quantified embedded implication. In addition to defining various semantics for the extended logic obtained, he also discussed the relation of this logic to normal logic programs. Adnan H. Yahya closed the conference with a discussion of generalized queries and their uses in database maintenance. He presented a scheme in which the query induces an order on minimal models, helping to answer questions of brave and cautious reasoning in disjunctive deductive databases.