**Title and Subtitle:**
A Computation Infrastructure for Knowledge-based Development of Reliable Software Systems

**Authors:**
Constable, Robert

**Performing Organization:**
Cornell University
Ithaca, NY 14853

**Funding Numbers:**
FA95550-05-1-0188

**Abstract:**
The Verification and Automated Reasoning research group at Cornell is involved in several DoD-related research projects that use formal methods, theorem proving, and knowledge management techniques to support the development of reliable software systems. These activities aim at making formal logical tools capable of solving difficult DoD tasks, using them for the development of safety-critical DoD software, making formalized algorithmic knowledge and logical software development tools accessible to researchers and programmers, and providing highly automated support for the training of researchers and programmers in the systematic design of reliable software.

Due to the huge search spaces and high processing demands of formal reasoning tools, our prototype Logical Programming Environment and its associated Formal Digital Library require a large number of processors and large amounts of memory to run efficiently in state-of-the-art applications. With additional computing resources, funded under this grant the Cornell group contributed significantly to the DoD mission. The research instrumentation, described below provided the necessary computation infrastructure for making our research on system verification feasible, and it opened our proof system to remote users.
The Verification and Automated Reasoning research group at Cornell is involved in several DoD-related research projects that use formal methods, theorem proving, and knowledge management techniques to support the development of reliable software systems. These activities aim at making formal logical tools capable of solving difficult DoD tasks, using them for the development of safety-critical DoD software, making formalized algorithmic knowledge and logical software development tools accessible to researchers and programmers, and providing highly automated support for the training of researchers and programmers in the systematic design of reliable software.

Due to the huge search spaces and high processing demands of formal reasoning tools, our prototype Logical Programming Environment and its associated Formal Digital Library require a large number of processors and large amounts of memory to run efficiently in state-of-the-art applications. With additional computing resources, funded under this grant the Cornell group contributed significantly to the DoD mission. The research instrumentation, described below provided the necessary computation infrastructure for making our research on system verification feasible, and it opened our proof system to remote users.
1 Knowledge-based Development of Reliable Software Systems

Experience has shown that it is extremely difficult and costly to build reliable and secure distributed real-time embedded software systems. Yet these are precisely the systems responsible for superior US military capability, and their overall practical value is increasing. A principal reason that these systems are hard to build is that they are inherently complex and hard to understand and specify. Because of these problems, serious design flaws are not discovered until late in the implementation process when correcting them is extremely expensive. Dealing with these questions is one of the most important challenges for computer science.

The Verification and Automated Reasoning research group at Cornell is developing formal methods and tools for the design, implementation, verification and optimization of software systems to address these problems. As a basis for efficient and practical correct-by-construction and secure-by-construction distributed programming we have developed a prototype Logical Programming Environment (LPE) that includes an extensive collection of automated reasoning tools.

Our experience has shown that describing system requirements, models and components in a declarative fashion can significantly reduce the above difficulties and enable bolder designs that can be trusted. Therefore, a digital library of formal algorithmic knowledge (FDL) is one of the key components of our logical programming environment. The library contains vast amounts of highly structured mathematical knowledge and enables researchers and programmers to search for knowledge that is relevant for the development of specific software systems. It provides a logical organization of the material so that it becomes more widely usable and shared among researchers and programmers involved in creating more reliable hardware and software. The library also includes logical accounting mechanisms to accomplish the highest standards of correctness and accuracy currently imaginable while enabling researchers to contribute new formal algorithmic knowledge to the repository that was originally developed in a variety of formal systems.

Our research group is active in two DoD-related research areas

- We are developing formal reasoning tools for our logical programming environment that aim at providing automated support for the development of secure, reliable, autonomous, distributed, real-time, and embedded software systems.

- We are applying these tools to state-of-the-art DoD applications such as a networked information systems, security protocols, and autonomous engagement protocols.

In the past 20 years the Verification and Automated Reasoning research group has made fundamental contributions to programming technology and demonstrated that they apply in practice. We have provided evidence that our methods are effective [LKvR+99, LvRB+01, BKvRC01, BKvRL01, Kre04] and described the theoretical advances on which our work rests [Con97, Con98, CH00, Kop02]. We have developed a logic of events [BC03, Con04] for
naturally specifying distributed computing tasks, for reasoning about events, for verifying protocols, and for synthesizing correct distributed processes through formal reasoning. We have built prototype implementations of the logical programming environment and the formal digital library [Hic01, ABC+02] and connected it to theorem provers such as Nuprl [CAB+86, ACE+00, Kre02, Nup], MetaPRL [HNC+03, Met], JProver [SLKNO1], and PVS [ORR+96, PVS, ABC+03].

The new computation infrastructure funded by this AFOSR grant allowed several researchers to perform heavy-duty theorem proving tasks simultaneously. This means that multiple library processes and logic-editors, and between 24 and 60 proof engines (refiners), model checkers, and SAT solvers could be active at the same time, each requiring between 500MB and 4GB of memory. The equipment also allowed external users to browse, search, combine, and add formal material to our knowledge base.

1.1 Specific Equipment Purchased

Internal Cluster: 32 processors, 68GB RAM nodes, 1200GB file system

- 1 Dell Poweredge 2850, 2x3.6Ghz, 8GB RAM, 4x300GB disk $9,260.50
- 7 Dell PowerEdge 1850, 2x3.6Ghz, 4GB RAM, 36GB disk $26,055.75
- 4 Dell PowerEdge 1850, 2x3.8GHz, 4GB RAM, 36GB disk $13,602.00
- 4 SunFire X4100, 2xAMD, 4GB RAM, 1x36GB disk $22,397.20
- 1 Dell Poweredge 4210 Frame $2,734.75
- 1 HP Switch $1,499.00

Web cluster: 12 dual-core processors, 44GB RAM, 900GB file system

- 1 Dell 1850 2x2.8Ghz dual-core, 8GB RAM, 2x300G disk (fileserver) $4,928.25
- 1 Dell 1850 2x2.8Ghz dual-core, 4GB RAM, 2x150G disk (gateway) $4,785.75
- 4 Sun X4200 2xAMD 285 (2.6Ghz dual-core), 8GB RAM, 2x73GB disk $25,748.60
- 1 HP Switch $1,499.00

Workstations and Lab equipment

- 6 Dell OptiPlex GX620 MT, 3.4Ghz, 2GB RAM, 160GB Disk, Monitor $10,275.60
- 1 Dell OptiPlex GX620 MT, 3.4Ghz, 2GB RAM, 160GB Disk, Monitor $2,706.16
- 1 Dell OptiPlex GX620 MT, 3.4Ghz, 4GB RAM, 250GB Disk, Monitor $2,713.33
• 1 Dell Precision 380n, 3.2Ghz DC, 4GB RAM, 250GB Disk, Monitor $5,771.81
• 1 Apple PowerMac G5, 2x2.5Ghz DC, 4GB RAM, 250GB Disk, Monitor $5,828.00
• 3 Dell Latitude D610, 2Ghz, 2GB RAM, 80GB Disk, 14.1" Screen $6,912.06
• 1 Dell Precision M90, 2.16Ghz DC, 4GB RAM, 100GB Disk, 17" Screen $6,007.27
• 1 Apple MacBookPro, 2Ghz DC, 2GB RAM, 100GB Disk, 15.4" Screen $2,508.00
• 1 Dell 3400MP Projector $2,709.98
• 1 Dell 3100cn Color Laser Printer $1,661.50
• 1 Dell 1600n Multifunction Laser Printer $899.35
• 1 Dell 1700n Laser Printer $602.31
• Charges for on-site Installation $3,958.00

Total Expenditures for System: $165,064.17

2 DoD Research Activities Supported by the Equipment

The Verification and Automated Reasoning research group was involved in several DoD-related research projects that aim at an increased capability to protect the nation's software infrastructure through the use of formal reasoning, theorem proving, and knowledge management techniques. The main thrust of these activities was to make formal logical tools capable of solving difficult DoD tasks, to use them for the development of reliable safety-critical DoD software systems, and to make formalized algorithmic knowledge and logical software development tools accessible to researchers and programmers, thus providing highly automated support for educating a new generation of researchers and programmers in the systematic design of reliable software.

Most of our research experiments are based on our prototype Logical Programming Environment (LPE). The logical programming environment includes an extensive collection of automated reasoning tools and is centered around a digital library of formal algorithmic knowledge (FDL), which provides the tools necessary for structuring, organizing, verifying, and authenticating the library contents as well as the necessary services to make the formal knowledge widely usable and shared among researchers and programmers. The formal digital library is already connected to three major theorem proving environments and includes a variety of algorithms together with the corresponding declarative knowledge. To enable a worldwide user community to contribute new formal material to the common repository we have made our prototype FDL accessible through the web at http://www.nuprl.org/FDLProject/fdl_online.html.
The new instrumentation enabled us to integrate more and stronger reasoning tools into the logical programming environment and to import the library contents of existing proof and verification systems into the FDL, which in turn would increase the research community’s ability to create more formal knowledge.

In the following we describe the DoD research projects in which our group is involved.

2.1 AFRL Information Assurance Institute: Verification of Security Protocols

- DoD organization: AFOSR/AFRL
- Project Titles (Information Assurance Institute Tasks):
  - Characterizing the End-to-end QoS Behavior of Networked Information Systems
  - Verifying Security Protocols
- Grant number: F49620-02-1-0170
- Principal Investigator: Fred Schneider
- http://www.cis.cornell.edu/iai
- Duration: November 2006
- Amount of support: $4,138,325
- Source of support: AFOSR/AFRL

Our research group is involved in the joint AFRL/Cornell Information Assurance Institute (IAI). The IAI fosters collaboration between researchers of the Airforce Research Laboratory at Rome, NY and of Cornell’s computer science department by supporting activities aimed at developing a science and technology base to enhance information assurance and the trustworthiness and reliability of networked systems.

Our research efforts in the IAI are directed to bringing formal reasoning techniques to bear in networked information systems, particularly in applications that are of interest for the Airforce Research Laboratory, in order to increase the security and quality of service provided by these systems. Our research group is active in two IAI tasks.

In our first task, our research aims at increasing the trust in a Networked Information System by giving a precise characterization of its end-to-end Quality of Service (QoS) behavior and by verifying critical QoS properties using formal reasoning tools. By quality of service we mean security and performance properties that characterize the entire set of executions but are not necessarily satisfied by each individual execution and by precise characterization we mean a high-level description in the formal language of some theorem proving system.

To support the verification of QoS properties, we are developing automated reasoning tools for our Logical Programming Environment that are capable of verifying system properties based on verifications of component specifications and that derive the requirements on the components from the desired QoS properties of the complete system. These tools
will then be used to verify critical quality of service properties for a small set of example applications that are of interest for the Airforce Research Laboratories, such as AWACS Tracking, Data Management and Routing, and JBI publish & subscribe mechanisms.

In our second task, we are working on the verification of a key security protocol of Cornell’s On-Line Certification Authority (COCA), called Asynchronous Proactive Secret Sharing (APSS), which enables replicating COCA at multiple sites without becoming more vulnerable to attacks. APSS is a complicated protocol on which the security features of COCA depend heavily.

We are using our logic of events to formalize properties of the APSS protocol and explore the use of knowledge-based message automata for this task. Having set up this framework, we can use our LPE to verify knowledge-based protocols and to synthesize them from knowledge-based specifications (in terms of what an adversary should not know). We will simulate the arguments used in strand spaces [THG99] in our logic of events, and to go beyond them by using an approach based on algorithmic knowledge [FHMV95, HP01], where agents only know what they can compute according to some algorithm, to model the resource-bounded nature of an adversary’s knowledge.

2.2 DARPA IPTO: Boosting real-world reasoning technology

- DoD organization: DARPA IPTO
- Project Title: Boosting reasoning technology through randomization, structure discovery, and hybrid strategies
- Grant number: FA8750-04-2-0216
- Principal Investigator: Bart Selman
- Duration: July 2009
- Amount of support: $3,580,000
- Source of support: DARPA IPTO

Together with our colleague Bart Selman we contributed to a project that aims at developing the next generation of reasoning technology for use in large-scale knowledge- and information-intense intelligent systems. In order to achieve this goal, the design of new inference methods will be combined with techniques for controlling the computational cost of large scale reasoning technology.

Specifically, we helped extend the breadth and performance of the formal reasoning tools in our logical programming environment by integrating SAT solvers, model checkers, proof planning, and proof agent techniques. The automated reasoning power of SAT solvers has proven itself to be unexpectedly effective on a large class of problems, and there is promise that our research will significantly enlarge this class. One of the most promising areas for expanding the range of use is into the realm of applied interactive theorem proving.
2.3 AF SBIR/STTR: Synthesis of Correct Embedded Systems

- DoD organization: AF SBIR/STTR
- Project Title: SCorES, A Logical Programming Environment for Distributed Systems
- Grant number: F045-023-0029
- Principal Investigator: David Gaspari, ATC-NY
- Duration: May 2007 (assuming a successful review in 2005)
- Source of support: AF SBIR/STTR

In this DoD project we cooperated with ATC-NY in the development of a mathematically based tool, SCorES (Synthesis of Correct Embedded Systems) that provided automated support for specifying, developing, verifying, and synthesizing real-time distributed systems at a high level of abstraction.

Pioneering work at Cornell has addressed the difficulties of implementing and maintaining reliable and efficient distributed systems by demonstrating how to model and analyze distributed and real-time behaviors mathematically but great effort is needed to make the resulting techniques applicable.

The collaboration between Cornell and ATC worked to extend declarative and constructive program development methods to distributed and hybrid systems. In this paradigm, specifications are stated declaratively in a logical language and development steps are inferences in a logic for the programming domain, which makes sure that programs are correct by construction. The logic is implemented and supported by editing, refinement, verification and information management tools. SCorES used our logical programming environment and its rich mathematical library to provide reasoning tools and hooks to modules for code synthesis and simulation.

Summary

All of our DoD-related research efforts depend on our ability to bring our logical programming environment and its formal digital library to bear in practice. They provide reasoning tools capable of making significant contributions to the development of high-assurance software systems in large-scale DoD applications. They are crucial for the research community’s ability to rapidly produce formal knowledge and, consequently, more reliable and secure software, which in turn will provide a basis for protecting the nation’s critical software infrastructure.

The instrumentation played an important role in making this happen, enabling Cornell researchers to run processor- and memory-demanding software, thus increasing their productivity when dealing with difficult DoD applications.

Publications

2006
Researchers involved in the DoD projects

The instrumentation was used by researchers and graduate students of the Verification and Automated Reasoning research group at Cornell as well as by researchers that participate in the above DoD research activities.

Robert Constable (professor, PI):
   applications, education & training, theoretical foundations, student supervision, project management
Christoph Kreitz (senior research associate):
automated proof tools, applications, education & training, student supervision

Bart Selman (professor):
large-scale reasoning tools, education & training, theoretical foundations, student supervision

Stuart Allen (research associate):
  system design, theoretical foundations of the library, student supervision

Mark Bickford (senior researcher at ATC-NY):
large scale applications, theoretical foundations

Richard Eaton (chief programmer): system design and implementation

Lori Lorigo (graduate student): system development, applications

Wojciech Moczydlowski (graduate student): theoretical foundations

Evan Moran (graduate student): theoretical foundations

Radhika Lakshmanan (undergraduate student): library content development, automated proof tools

The following people collaborate with our group in the DoD projects

Cornell University:
Robbert Van Renesse (senior research associate, systems group)

ATC-NY:
  David Gaspari (senior researcher)

Airforce Research Laboratory, Rome:
Patrick Hurley
John Faust
References


Alexei Kopylov. Representation of object calculus in type theory, 2002. submitted to LICS.


