

Editor: Richard Doyle
Jet Propulsion Lab
rdoyle@jpl.nasa.gov

MAPGEN: Mixed-Initiative Planning and Scheduling for the Mars Exploration Rover Mission

Mitchell Ai-Chang, John Bresina, Len Charest, Adam Chase, Jennifer Cheng-jung Hsu, Ari Jonsson, Bob Kanefsky, Paul Morris, Kanna Rajan, Jeffrey Yglesias, Brian G. Chafin, William C. Dias, and Pierre F. Maldague, *NASA Ames Research Center and the Jet Propulsion Laboratory*

The Mars Exploration Rover mission is one of NASA's most ambitious science missions to date. Launched in the summer of 2003, each rover carries instruments for conducting remote and in situ observations to elucidate the planet's past climate, water activity, and habitability.

Science is MER's primary driver, so making best use of the scientific instruments, within the available resources, is a crucial aspect of the mission. To address this criticality, the MER project team selected MAPGEN (*Mixed Initiative Activity Plan Generator*) as an activity-planning tool.

MAPGEN combines two existing systems, each with a strong heritage: the APGEN activity-planning tool¹ from the Jet Propulsion Laboratory and the Europa planning and

scheduling system² from NASA Ames Research Center. This article discusses the issues arising from combining these tools in this mission's context.

Combining systems

In a most exciting development, two NASA rovers—named Spirit and Opportunity—were slated to arrive at the Red Planet in January, at two scientifically distinct sites. (Spirit arrived successfully on 3 January, with Opportunity scheduled to arrive 24 January—see Figures 1 and 2.) Each rover will have an operational lifetime of 90 sols (Martian days) or more and can traverse an integrated distance of one kilometer or more, although the maximum range from the landing site might be less. Scientifically, MER seeks to

- Determine the aqueous, climatic, and geologic history of a site where on Mars conditions might have been favorable to the preservation of evidence of prebiotic or biotic processes
- Identify hydrologic, hydrothermal, and other processes that have operated at the landing site
- Identify and investigate Martian rocks and soils that have the highest-possible chance of preserving evidence of ancient environmental conditions and possible prebiotic or biotic activity
- Respond to other discoveries revealed by rover-based exploration

Each sol, operations personnel on Earth receive telemetry from the rovers. On the basis of the downloaded data, they must construct, verify, and uplink a detailed sequence of commands for the next sol to the rovers. Thus, operations personnel must formulate a viable sequence that satisfies the mission goals within tight deadlines. To help address this critical need, MAPGEN can automatically generate plans and schedules for science and associated engineering activities; assist in hypothesis testing, such as what-if analysis on various scenarios; support plan editing; analyze resource usage; and perform constraint enforcement and maintenance.

APGEN has served as a multimission tool for several flight projects (including Cassini and Deep Impact), while Europa flew onboard NASA's Deep Space 1 as part of a technology experiment to demonstrate the first onboard

Editor's Perspective

On the evening of 3 January, at approximately 8:35 pm PST, Spirit, the first of the two Mars Exploration Rovers, landed successfully on the surface of Mars, bouncing and rolling over a kilometer before coming to rest. Spirit, as it was programmed to do, then deflated and retracted its air bag cushions, finding itself in the optimal base-petal-down configuration, ready to begin its scientific mission of exploration. Getting there entailed traversing successfully the critical entry, descent, and landing (EDL) phase of its mission, which Spirit accomplished in exemplary fashion.

At the time of this writing, Spirit has rolled off its landing platform and is engaging surface operations. Its twin, Opportunity, is near the very end of its cruise phase, preparing to engage EDL on 24 or 25 January to a successful conclusion on the other side of the planet.

Surface operations for each rover are planned nominally for 90 Martian days (sols), and to maximize the science return accomplished during this intense, brief period, the mission has engaged the use of AI technology in the form of a planning and scheduling system, to be used to determine each sol's worth of science activities. This system, called MAPGEN, utilizes technology developed in NASA's Intelligent Systems Program and is a joint effort of the Ames Research Center and the Jet Propulsion Laboratory.

—Richard Doyle

closed-loop AI-based system (called Remote Agent³). MAPGEN leverages these tools' strengths to give the MER user a comprehensive tool for science activity planning and scheduling. Given MAPGEN's nature, MER expects to use the system in two modes, at different times:

- With the planner turned off and only the basic APGEN functionality operating (representing the traditional approach to science activity planning)
- With the planner turned on and the full MAPGEN functionality operating, with advanced planning scheduling and constraint maintenance

Here we describe the second mode.

MAPGEN system design and functionality

MAPGEN's primary users are MER mission tactical planners and scientists who'll manipulate the science objectives in concert with specific engineering constraints to further refine the plans to conform within known resource bounds. Although the tool could deal with strategic timelines, it's slated for use solely for day-to-day tactical planning.

MAPGEN provides key capabilities beyond those of APGEN, including generating a plan and actively enforcing flight and mission rules with conflict resolution, such as forbidden activity overlaps or resource violations. The planner's domain model specifies forbidden overlaps and other constraints. The model, in turn, derives from an activity dictionary that describes the comprehensive set of abstract activities the science user would be expected to need, as well as flight and mission rules based on the project's flight rules dictionary. The planner's underlying constraint engine enforces the domain model's rules.

Figure 3 presents a functional view of MAPGEN. During the activity plan-generation phase for uplink, science users construct a list of observations for each sol. Each observation consists of a collection of coordinated high-level activities. APGEN expands these into lower-level activities based on the definitions in the activity dictionary. These are provided to MAPGEN along with relative and absolute timing constraints. These activities, together with the supplied engineering activities—for example, the need to have the hazard cameras document the rock target before performing contact science—and initial con-

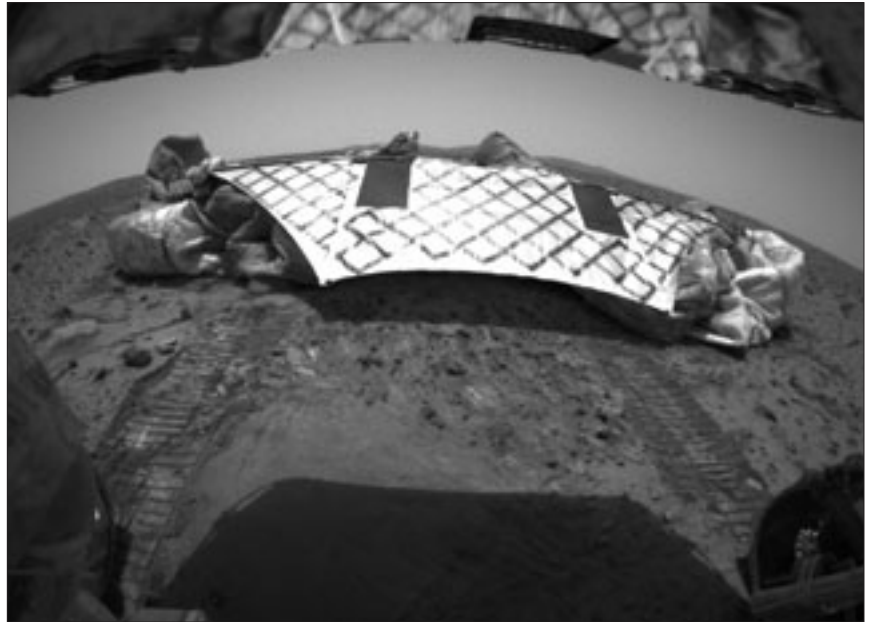


Figure 1. Image from the Mars Exploration Rover Spirit's rear hazard identification camera showing the rover's hind view of the lander platform. (photo courtesy of the Jet Propulsion Laboratory)

ditions, form the basis for the start of the planning phase.

The planner uses the domain model and generates a possible plan, which the APGEN GUI displays as a candidate solution for the user to modify. Alternatively, MAPGEN allows selective incremental planning of these high-level observation goals. In using this alternative method, the user must determine the order in which observation

goals are solved by selecting them in the GUI. The user can interactively experiment with alternatives—for example, by exploring *what-if* scenarios. Intermediate results feed into the next iteration cycle in this mixed-initiative style of converging to a final plan that the user finds appropriate. Once this iterative process is complete, the plan outputs to a file for use in the next uplink process phase.



Figure 2. Image from Spirit's panoramic camera showing angular and rounded rocks on the martian terrain. (photo courtesy of the Jet Propulsion Laboratory)

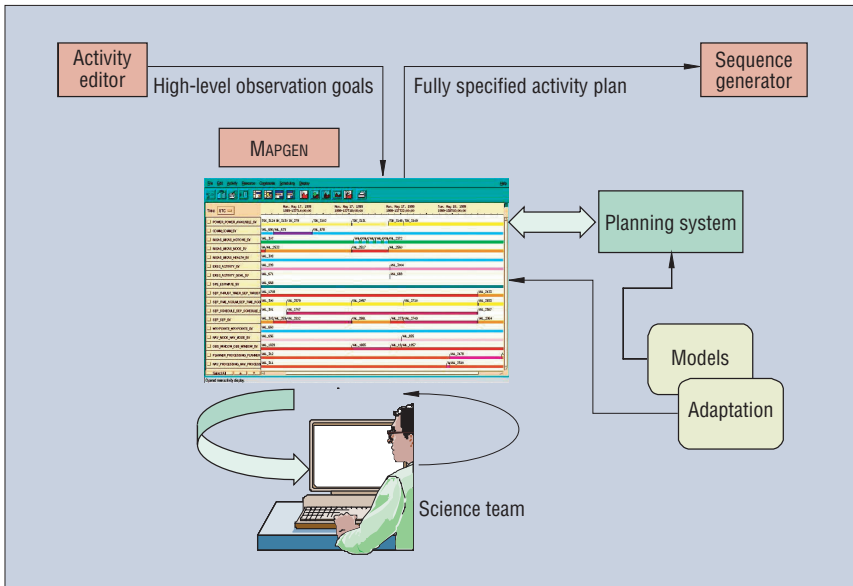


Figure 3. The MAPGEN system's functional organization.

MAPGEN's primary objectives then are to generate complex, valid plans that are free of conflicts and resource violations, enforce a principled approach for plan generation and modification, encapsulate high-fidelity models into activity planning, and enable visualization and manipulation of these plans for effective mixed-initiative interaction with the user

By achieving these objectives, we expect MAPGEN to provide these mission benefits: leave more time for science analysis, planning, and verification during activity planning; provide a better characterization of the constraints involved in activity planning, resulting in more robust science plans and hence fewer plan revisions downstream for the engineering analysis; reduce the

mission staff's workload; and provide higher science return

As Figure 4 shows, APGEN and the planner connect through a well-defined interface that synchronizes the planner's database with the sets of activities that the user manipulates. This permits access to planner functionality through normal use of the APGEN GUI, so an experienced APGEN user who uses MAPGEN sees the usual GUI and can interact with it in familiar ways.

User input

In general, the system needs a specification of an initial plan state, user observation goals and their priorities, expansions of compound activities, standard constraints (flight rules), and sol-based (daily) constraints.

The initial plan state might contain initial conditions in the classical sense—at the beginning of the plan—but it might also contain expected conditions throughout the plan period, as well as a skeleton plan of activities arising from engineering requirements. This information resides in input files in APGEN's Activity Plan File (APF) format.

The plan goals are specified as a set of observations, each comprising a set of top-level activities designed to elicit some specific scientific result. The top-level activities can expand into lower-level activities according to fixed rules. The scientists assign each observation and top-level activity a priority, which MAPGEN uses to decide which observations to delete in whole or in part to yield a valid plan within the limitations imposed by the constraints.

Prior to the mission, the APGEN development team built an adaptation file, based on the activity dictionary, that describes the expansions of compound-activity types. Similarly, on the planner side, a model describes the activities and mission constraints. These files also describe other relatively permanent information, such as duration and relative placement of activities.

The user employs a separate tool, the *constraints editor*, to enter the sol-based or daily constraints. This specialized browser-based system facilitates the rapid entering, visualization, and consistency checking of temporal constraints. After the constraints editor user formulates constraints that help fulfill the intent of the scientific observation, the planner enforces these constraints to provide a more desirable solution that captures the intent. Figure 5 shows a specific pane in the constraints editor browser window.

MAPGEN provides a temporary storage area—the *hopper*—for observations and other activity requests, which can be entered either directly by the user or via an input APF file. The hopper is a "holding area" for observations and activities that have not yet been made part of the plan. Constraints on activities in the hopper are held in abeyance until the activities are selected for planning. If an attempt to place an activity in the plan fails, the activity moves to the hopper. The user can also directly move activities from the plan to the hopper in order to try out alternative plans.

Flexible time

The planner's use of a *flexible-time* database has interesting consequences because

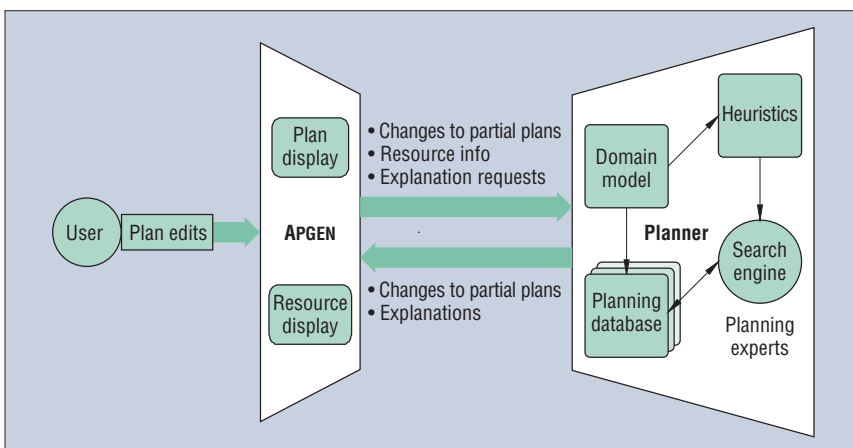


Figure 4. MAPGEN's system architecture.

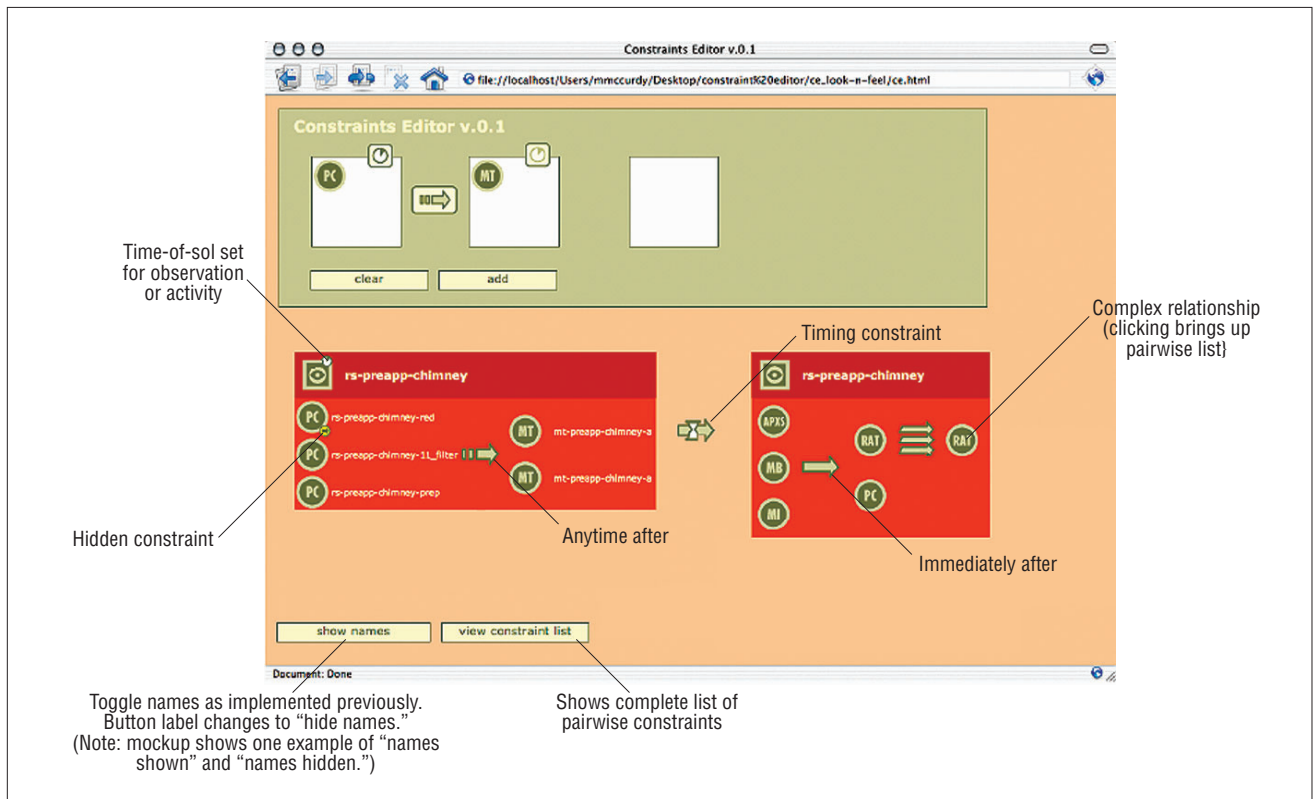


Figure 5. A pane in the constraints editor browser window.

this approach to handling temporal constraints must be reconciled with APGEN and its related tools, which expect to see a fixed time schedule.

Flexible time implies that instead of finding a single solution, the planner preserves maximum temporal flexibility by maintaining a set of solutions that satisfy the constraints, represented internally as a *simple temporal network*.⁴ As a result of propagation in the STN, each activity acquires a refined time window for its start time.

By preserving a flexible set of solutions, the planner can often adapt to additional constraints by using the flexibility rather than completely resolving the problem. However, presenting such flexibility to any plan GUI—and to APGEN in particular—poses significant problems. For any plan GUI, providing a visual representation of flexible windows, as well as binary temporal relations such as *before* and *after*, is difficult. Even if the system could find a visual representation, the cognitive burden of interpreting it likely would detract from an intuitive grasp of the plan. In addition, many tools associated with APGEN, such as those that calculate resource use, require a fixed schedule of activities.

Our approach presents a single solution to the user in the APGEN GUI, while the planner maintains the flexible set of solutions as a backup. In other words, while the user sees a traditional fixed timepoint plan in the APGEN GUI, the underlying representation of the plan in the planner is a richer set of adjacent plans that all satisfy the constraint. The theory of STNs guarantees that the user will obtain a solution by assigning to each event the earliest time in its time window.¹ The planner initially presents this earliest-time solution to APGEN (unless the activities already have start times, in which case it chooses a solution that is as close as it can get to the existing times while satisfying the constraints).

Although presented with a single solution, the user nevertheless can access the full set of solutions through what we call a *constrained move*. The user can select an arbitrary activity in the GUI and drag it to a new location in its timeline, provided the new location is within the plan flexibility's scope. (During the move, the user sees the scope as the region between two vertical red lines.) After the user drops the dragged activity at its new location, the planner adds a position constraint to fix it there. This restriction

propagates to the other activities, which change their locations accordingly, by the minimum amount necessary to satisfy all the constraints. Consequently, the planner performs active constraint maintenance, with minimum perturbation of the previous state. The temporal propagation is fast enough that the combined moves appear instantaneous.

Adjustable planning autonomy

The MER application might require varying degrees of autonomy. Because project managers naturally want to obtain as much science benefit as possible, missions like MER run the risk of oversubscribing resources—taking on more observation goals than the systems can reasonably handle. Although the planner itself can construct a workable plan by rejecting lower-priority observation goals, some degree of human user tweaking might produce a superior plan. So, MAPGEN also provides some capabilities to exercise more fine-grained control.

To facilitate a more manual operation mode, an APGEN user can turn to the hopper. The user has a menu of commands that includes both `plan all` and `plan selected` goals. In the latter case, the

Mitchell Ai-Chang is a senior scientist with the NASA Ames where he is the software test lead for MAPGEN and Constraint Editor. Contact him at maichang@ptolemy.arc.nasa.gov.

John Bresina is a senior research scientist at NASA Ames, where he leads the Intelligent Robotics Group and is the deputy lead of MER's MAPGEN team. He'll also serve as one of MER's tactical activity planners. Contact him at bresina@ptolemy.arc.nasa.gov.

Len Charest is the integration lead for the MAPGEN project at the Jet Propulsion Laboratory. Contact him at len.charest@jpl.nasa.gov.

Adam Chase formerly was a computer scientist working on the MAPGEN project at JPL. Contact him at ac5p@earthlink.net.

Jennifer Cheng-jung Hsu is the DeAnza-Foothill test intern at NASA Ames, where she works on testing and validating MAPGEN soft-

ware. Contact her at jennifer@ptolemy.arc.nasa.gov.

Ari Jonsson is a senior scientist with the Research Institute for Advanced Computer Science and a member of the Planning and Scheduling Group at NASA Ames's Computational Sciences Division. Contact him at jonsson@ptolemy.arc.nasa.gov.

Bob Kanefsky is a computer scientist with the QSS Group in NASA Ames's Computational Sciences Division where he participated in the development and deployment of software for the Deep Space 1 Remote Agent Experiment, Mars Pathfinder, Mars Polar Lander, and MER. Contact him at kanef@ptolemy.arc.nasa.gov.

Paul Morris is a senior research scientist on the MER mission at NASA Ames, where he also participated on the Remote Agent and the Europa projects. Contact him at pmorris@ptolemy.arc.nasa.gov.

Kanna Rajan is a senior research scientist and member of the management team of the Autonomy and Robotics Area at NASA Ames, where he is principal investigator and project lead for MAPGEN. Contact him at kanna.rajan@nasa.gov.

Jeffrey Yglesias is a programmer analyst for the MAPGEN project and Constraint Editor applications at NASA Ames. Contact him at yglesias@ptolemy.arc.nasa.gov.

Brian G. Chafin is an engineer at JPL. Contact him at brian.chafin@jpl.nasa.gov.

William C. Dias is the lead Tactical Activity Planner on MER surface operations. Contact him at william.c.dias@jpl.nasa.gov.

Pierre F. Maldague is a senior member of the Missions Systems Engineering Section at JPL. Contact him at pierre.f.maldague@jpl.nasa.gov.

planner will try to satisfy only preselected observation goals.

The division between APGEN and the planner separates varying levels of autonomy in constraint handling. The basic APGEN tool enforces constraints passively, flagging violations but not trying to fix them. The planner, by contrast, can actively enforce constraints by rearranging and moving activities. For example, consider a mutual-exclusion flight rule that prohibits moving the robotic arm while the rover is driving. If the current plan violates this flight rule, APGEN will only detect and flag that violation, whereas the planner can add a constraint that orders the activities to prevent the violation.

A similar dichotomy exists for resource constraints. In this case, APGEN plots resource use profiles over time, highlighting any resource violations. The planner has an optional mode where it can be commanded to try to repair these violations, which it does by once again exploiting its backup flexible set of solutions. That is, it installs new constraints designed to level the resource use to best avoid violations. From the planner's viewpoint, it's restricting the set of solutions to avoid potential conflicts. However, the APGEN database receives a new fixed plan in which the violation has been repaired.

Currently, we've implemented the basic combined system and have exercised it in several ground software operational-readiness tests. On the basis of feedback from these tests, we've extended the system to automate additional routine and time-consuming aspects of activity planning. For example, the system provides commands to automatically compute CPU and heater requirements for the activities. Consequently, MAPGEN has become an example of mission-critical software in the uplink process for MER surface operations. Once engaged after the first rover egresses its lander, MAPGEN will become the first AI-based system to control a space platform on another planet's surface.

The time constraints and pressures involved in mission operations often dictate that the mission planner adopt ad hoc solutions for novel issues that arise. Following the mission, we hope to use our experience to stimulate deeper research in these areas and ensure that theoretical studies don't neglect the real problems. ■

Acknowledgments

We thank Nicola Muscettola for his ceaseless efforts to promote advanced technology infusion

and for pushing for the MAPGEN's development. We also thank JPL's Adans Ko, who was instrumental in pushing for MAPGEN to happen and who has guided and supported its development. Finally, we thank Conor McGann for putting in the infrastructure for testing MAPGEN and Will Edgington for implementing the testing program. Portions of this work were performed at the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

1. P. Maldague et al., "APGEN: A Multi-Mission Semi-Automated Planning Tool," *Proc. 1st Int'l Workshop Planning and Scheduling for Space*, AAAI Press, 1997.
2. A. Jonsson et al., "Next Generation Remote Agent Planner," *Proc. 5th Int'l Symp. Artificial Intelligence, Robotics and Automation in Space (ISAIRAS 99)*, European Space Agency Publications Division, 1999, pp. 363–370.
3. N. Muscettola et al., "Remote Agent: To Boldly Go Where No AI System Has Gone Before," *Artificial Intelligence*, vol. 103, nos. 1–2, Aug. 1998, pp. 5–47.
4. R. Dechter, I. Meiri, and J. Pearl, "Temporal Constraint Networks," *Artificial Intelligence*, vol. 49, nos. 1–3, May 1991, pp. 61–95.