日本オペレーションズ・リサーチ学会
2004年秋季研究発表会

Constraint Programming-based Scheduling Cooperated with Rules-based Systems

03000500 アイログ(株) チェン ユウ CHENG Yu

1 Introduction

This paper introduces a new approach, which cooperates constraint programming-based scheduling with rules-based systems to form a new application framework for solving manufacturing scheduling problems. By this approach, constraint programming -based scheduling can provide feasible and good solutions to a schduling problem, while fast-changing business conditions have impact upon the problem can be managed by the rule-based syetem. It can also be integrated to an interactive Graphical User Interface (GUI) built with Gantt Chart to provide more user-friendly interactive interface to the business users. The maintenance of the business object model and rules used by the scheduling engine can be supported via a rule editor syetem, enabling the agility to modify business models and policies under fast-changing business conditions.

2 Scheduling problem and solution approaches

The task of scheduling is to produce the detailed plan of production activities (such as assembling, machining, chemical reactions, mixing, forming or separating) on different machines or production lines, in order to efficiently produce quality finished products in a timely manner, while satisfying customer demand for the finished product. This requires the scheduling engine to generate the best valid scheduling dedicated for various criterion and the requirements from business users and at the same time satisfy all the possible constraints during the production processes.

2.1 Various solution approaches

There basically exist three different paradigms for solving scheduling problems.

- **Rule-based systems and heuristic-based approach**

Traditionally, rule-based systems have been used to store scheduling heuristic as rules, which are used as scheduling algorisms for generate solutions. In particular cases, it is also used as a tool for checking feasible scheduling solutions or for finding out any violated constraints in a solution. Basically, this approach requires the rule database to grow with the knowledge rule and heuristic newly acquired and can lead to huge, barely manageable systems. As knowledge rule and heuristic usually evolve quite rapidly, being updated weekly and even daily with some online services, maintaining such a rule base can quickly become an extremely difficult task. Besides, when rule-based systems are used as scheduling algorisms to generate solutions, it is hard to adequate for most complex scheduling scenarios and specific constraints in the processes of manufacturing operations. For these reasons, rule-based system alone used as scheduling algorisms is currently losing ground.

- **Procedure-based approach**

In the procedure -based approach, the scheduling problems are described as a specific mathematical model, or a simulation model, or a logical model in a computer program. To solve the problem, the system must describe what to do and how to solve as a step by step procedure, like how to select or generate a particular part without violating constraints. However, for real scheduling problems, it is usually a difficult task to make such kinds of solving procedures when too many complex constraits and scheduling scenarios needs to be considered at the same time. Particularly, when some critical constraints are violated, the solving procedures should be repeat again from the start. The computing time could be increased significantly when too many of such kinds of backtracking computation.

- **Constraint programming -based approach**

Scheduling problems can be treated as a particular type of Constraint Satisfaction Problem (CSP). With Constraint Programming (CP) technology, it is essential possible and efficient to detect and remove infeasible (or bad) schedulings from the solution space. This can be done easily with constraint propagation algorithms. It is also possible to compute the user's best scheduling in terms of cost, quality or delivery time. However, with CP approach only, it is not easy to capture constantly changing business conditions, to obtain high performance of maintaining the changing of requirements from business users, such as scheduling rules, policies or constraints specified by business users. For axample, as a scheduling result, it is obviously reqired to satisfy all the demand required by business users, to make all the orders from all the customers. However, usually, not all the customers and their orders have equal priority. For axample, loyal customers with large orders, not surprisingly, should receive higher priority. This describes a potential business policy that the scheduling system must attempt to adhere by. Interestingly, policies like this constantly change, are abandoned, or are newly created. Unfortunately with CP approach only, most systems have limited flexibility to handle these changes, and the result is that any new or modified business policies cannot be reflected in production schedules immediately.

2.2 CP plus rule-based approach -an innovative approach

While rule-based system and CP may seem similar from a high level, they're quite different in terms of the technologies and their applicable uses. It is commonly accepted that rule-based systems approach is not adequate for most complex scheduling scenarios, which is the reason optimization engines based on CP are currently considered the state of the art. CP technology gives the end users the ability to interact with a scheduling engine in the method they prefer, rather than a predetermined path developed with a business-rule modeler. However, maintenance on scheduling applications considered exceptionally costly and consistently higher than expected. Rule-based systems provide features for simplifying scheduling application maintenance and reducing maintenance costs with the respect to the change of business polices. Thus, obviously, a new approach is to use constraint programming-based scheduling cooperated with rules-based systems, which uses both of the two technologies together, and which allows business users to change scheduling rules and use CP technology to provide solutions, which satisfy the

constraints respecting to the rules.

3 CP Cooperated with rules-based systems approach

3.1 Scheduling problem modeling: CP engine part

With CP plus rule-based approach, the scheduling problem is represented by a set of variables, with each variable having a specified domain of possible values. The business rules or so-called scheduling rules that describe the regulations to be respected are formulated as constraints among those variables. The CP engine applies the domain reduction to the variables by identifying and removing values that may lead to an invalid scheduling. These algorithms are called constraint propagation algorithms. The constraint propagation algorithms are embedded into a systematic traversal of the relevant parts of the solution space, which is organized in a tree. This enables the system to automatically perform backtracking steps, implement flexible search strategies, find optimal schedulings. Besides, the number of possible scheduling solutions may become extremely large for complex problems. As a consequence, it is necessary to appropriately guide the search to obtain good scheduling solutions. This can be done via search strategies. Strategies control the order in which decisions, i.e., the trying of the various possible values for each field, are made. With CP plus rule-based approach, problem-specific strategies can also be defined by rules.

3.2 Scheduling business rule modeling: rule-based part

The rule-based system manages all of the business rules that can be applied both on the model objects and external sets of data related to the scheduling. Thus, scheduling requirements can be described as rules to state both hard and soft constraints. Hard constraints are formulated via strict business rules, such as "should" or "should not". Soft constraints are formulated in terms of wishes and preferences. A wish characterizes a desire that the user would like to achieve, but it may contradict some other rules. Preferences make it possible for the user to indicate that a choice in a scheduling should be made over another during an automatic search. All of them can be formulated in a rule-like syntax.

3.3 Steps for application design and implementation

Unlike rule-based systems whose engine reasons directly on the application objects, an application with CP plus rule-based approach is centered on the scheduling engine and must have its own object model definition and instances. Because of this, the first and most important/critical step in designing an application is the definition of the Business Object Model (BOM), which describes the application domain objects and their relations.
The second step is to write the set of scheduling rules using the Business Constraint Language (BCL). Here also, as opposed to rule-based systems, this set of rules does not express the process of finding a solution but the set of constraints the scheduling engine will enforce for possible solutions if such exists. Thus, the design and implementation phase can be seen as iteration over these two steps.

3.4 Functionality architecture

The framework provides the following five levels of functionality, summarized in the architecture as below:

1) Interactive Gantt chart: visualize, analyze, interact and update production schedules
2) Optimization algorithms: generate feasible, low-cost detailed schedules
3) Maintenance and scheduling interface: modify object model, rules and algorithm parameters over time
4) Object model: model and capture intricacies of the manufacturing operation
5) External integration: leverage pre-defined integration to existing external ERP/ SCM system

3.5 Integrated Development Environment

With CP plus rule-based approach, both problem modeling and rules management should be handled. A complete set of productivity tools would be helpful, which allows both system developers and business users to graphically edit, manage and test scheduling model and related rules. The major functions of the IDE include:

- Object Model Editor: graphical editor for easily designing the scheduling model structure, and editing and managing scheduling objects and relationships;
- Scheduling Rule Builder: intuitive editor that enables non-technical users to edit scheduling business rules;
- Scheduling Rule Management: the Rule Builder also provides tools to store, version, query, manage and deploy scheduling business rules;
- Graphical User Interface (GUI): allows one to interact and to visualize the solutions from the schduling engine. It has been configured to display the problem objective weights in chart, as well as the solution in a gantt view, a resource usage view and an objective view as well as editing functions.

4 Remarks

This paper promotes and suggests a flexible, powerful constraint programming-based scheduling cooperated with rules-based systems approach to solve complex manufacturing scheduling problems. With the constraint programming-based technology, it is easier and faster to model and solve a resource-constrained scheduling problem, to reduce complexity of the problem by various kinds of constraint propagation. With the rules-based systems technology, it can obtain high performance of maintaining the changing of scheduling requirements, and all the rules can be dynamically created or changed through an interactive point-and-click editor. Thus, it allows business users or nonprogrammers to express scheduling rules in an intuitive way. Meanwhile, optimization, visualization, and business rules technologies are used and make it as a rich, powerful pre-integrated application framework. For more information about related technologies, it is recommended to have a visit to http://www.ilog.co.jp/ to know more detail.

5 Reference

[1] Optimization Technology White Paper, ILOG, Inc. 2001.
[2] ILOG Solver, User Manual, version 6.0, 2003.
[3] ILOG Scheduler, User Manual, version 6.0, 2003.
[4] ILOG JRules, User Manual, version 4.5, 2003.