N90-27278

# Resource Allocation Using Constraint Propagation

John S. Rogers
Artificial Intelligence Lab
Johnson Research Center
The University of Alabama in Huntsville

## Introduction

Scheduling and resource allocation considerations are fast becoming one of the major areas of research today. Although these are not new problems, technological advances continue to open new and ever increasing areas where having the right resources at the right place at the right time is crucial. As we enter into the space age and as the costs involved in doing specific research and development tasks increase the fundamental equation is truly, Time is Money.

The methods and techniques applied in this area are often as much an art form as a truly mathematical form. This is not to say that the foundations of each of these methods are not mathematically sound, but rather the how, when and why used in applying the mathematical principals can produce rather unique and sometimes more acceptable results. More specifically, there are certain parameters that are common to most resource allocation problems. How a particular application manipulates and utilizes these parameters often determines the degree of performance of a resource allocation system. A technique that can be applied to one of these parameters is presented here.

In many resource allocation and scheduling problems one of the major difficulties facing the investigator is determining how to apply different constraint equations. Numerous research efforts have been done in this area in an attempt to increase the investigator's control and ability in applying constraints. There are many widely varying techniques that are available and frequently used to solve this problem. Usually specific characteristics of the particular domain involved and the types of constraint equations that must be applied are the deciding factors in the selection of the appropriate technique. This is usually adequate for the current application; however, this solution usually does result in a generalized mechanism that can be applied to differing domains with only minor modifications.

In most cases a traditional approach of evaluating the constraining parameters for each one of the activities that is being scheduled at the time it is scheduled is sufficient. This is reasonable when the constraining relations for the activities are functions which depend upon resources solely. However, when an interdependency exists between the activities or if the domain is not constant new problems may arise. A method that addresses this in resource allocation problems is the method of constraint propagation. In the following sections of this paper the concepts involved in applying this technique are presented.

## Dynamic Allocation Scenarios

One area of interest today which can pose some unique and sometimes tedious constraint evaluation problems is that of dynamic domains. In this arena the domains are either constantly changing or possibly modifiable at discrete points of time within the analysis process. This scenario creates a true illustration of reality since all to often activities may require the consumption of a variable amount of a resource. Then as the activity is performed a more refined estimate of future available quantities of the resource is determined. This causes the domain to be dynamically modified. Due to the newly refined domain, a new allocation may need to be performed over the modified range to best utilize the remaining resources.

In approaching this task one might consider some type of readily available dynamic programming technique that would iteratively reformulate the resource allocation scheme based on the current domain state. Usually this is a plausible approach; however, in many instances this can cause severe time delays in the allocation softwares ability to traverse the domains time range. This only tends to magnify the often already excessive computational time required to perform the allocation process.

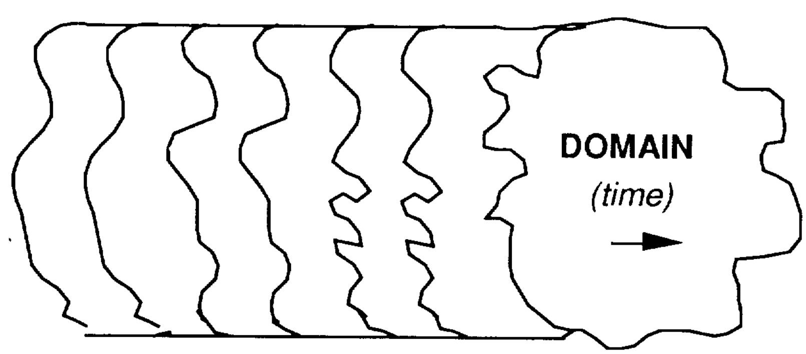


**Figure 1 Dynamic Changing Over Time**

What is needed is an alternative approach that can incorporate the effects domain changes can have on an existing allocation. In such an approach hopefully only those particular activities that are effected will be engaged and therefore reduce any unnecessary constraint calculations. One such technique is *Constraint Propagation*.

## What is Constraint Propagation?

Constraint Propagation is the technique of having localized mechanisms that control the behavior of individual activities as changes in the domain occur. This is more than just a constraining equation that is applied at a local level. It requires activation control to be maintained at a local level while domain changes happen globally. If only those activities that are directly affected by the domain change are activated then a direct reduction in the number of constraint equation evaluations can be achieved. Theoretically this evaluation reduction would result in the time required in the reallocation being reduced. Quite possibly each modification made by the adjustment of the allocated items across the time range could cause a subsequent alteration in the consumption of additional resources. More than likely this would initiate a series of downstream allocation modifications which could in effect initiate even more downstream allocation modifications and so on. The procession of changes effected on the downstream allocation of events can be considered analogous to a propagating wave across the domains remaining range. Thus arises the term Constraint Propagation.

Obviously the effectiveness of this methodology is closely tied to the amount of interdependence that exists between the different activities regarding the type and quantity of usage of the various resources. It may be logical to assume that as the commonalties regarding resource requirements between the different activities increases then the level of increased performance in this method decreases. However, logically one would assume that any reduction in the application of constraint equations should produce some amount of time savings. This implies that the overall performance increase seen by the system should always be non-negative.

## How do you design a Constraint Propagation System?

Specific considerations must be made in designing a resource allocation software system that applies the technique of constraint propagation. First, there is the need for local control of processes. This tends to exhibit a requirement for code encapsulation at the activity level while maintaining global access to domain information. Although there are possibly a variety of approaches to achieve this, the best approach is through the implementation of Object Oriented Programming Techniques (OOP). With this approach the individual activities and resources can be constructed as objects. An activity object locally contains all the information describing not only resource consumption requirements and constraints, but also information which controls what activation triggers exist that would cause a re-evaluation of this activity in the allocation process. All the information would exist as specific slot values located within the object itself. Likewise a resource object could contain the resources current state, usage history, any global limitations, and other pertinent resource specific information.

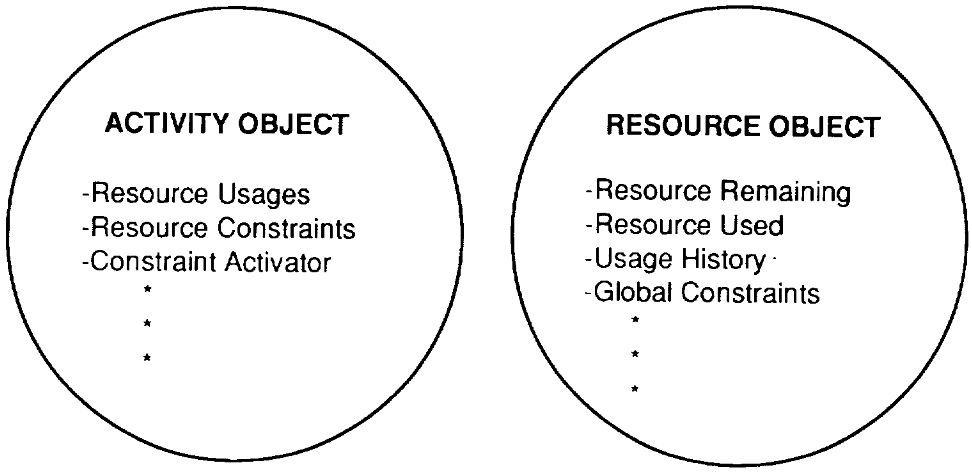


**Figure 2 Object Data Structure**

Many of the desired characteristics of a Constraint Propagation system are similar to those found in Frame Based Reasoning (FBR) systems. In an FBR system components can be designed that describe the control structure of the frame. These so called "daemons" are activated and controlled by the environment in which the frame resides. For example, an "if-decreased" daemon could possibly cause the initiation of the control code when a quantity of a specific resource is reduced for the time frame containing the specific performance of the activity. The range of functionality of these daemons is limited only be the developers ability to tailor the environments object manipulation characteristics.

There are several programming environments (C++, ADA, LISP, etc.) that boast an object data structure. For ease of manipulation and flexibility of control LISP is probably the most conducive of these environments in which to develop a Constraint Propagation system. This is due to the extensive functionality associated with the Lisp object as compared with other programming environments. These structures support the full range of message passing capabilities and can have external functional bodies called "Methods" which can be applied to each object. Simply put, a "Method" is a segment of computer code that acts similarly to a function that can be applied to a specific class of objects. It is through the use of these "Methods" that the behavior of each of the daemon slots within the object is defined and controlled. By the use of this varied level of control constraint propagation techniques become feasible.

## Summary

In this brief presentation the concept of constraint propagation has been discussed. Certainly, performance increases are possible with careful application of these constraint mechanisms. The degree of performance increase is related to the interdependence of the different activities resource usage. Although this method of applying constraints to activities and resources is often beneficial, it is obvious that this is no panacea cure for the computational woes that are experienced by dynamic resource allocation and scheduling problems. A combined effort for execution optimization in all areas of the system during development and the selection of the appropriate development environment is still the best method of producing an efficient system.

## References


1. Howard, Geoffry S., "Object Oriented Programming Explained," Journal of Systems Management, July 1986 pp. 13-18.

2. Cox, Bred J., Object Oriented Programming, Reading, Massachusettes: Addison Wesley Publishing Company,1986.

3. Rich, Elaine, Artificial Intelligence, McGraw Hill, New York, 1983.

4. Ozden, Mufit, "A Dynamic Planning Technique for Continuous Activities Under Multiple Resource Constraints," Management Science, Vol. 33, No. 10, October, 1987 pp.1333 -1346.

5. Dreyfus, S. E. and A. M. Law, The Art and Theory of Dynamic Programming, Academic Press, New York, 1977.