

Use of Series Parallel Digraph Analysis in Generating Instructions for Multiple Users

Margaret Mitchell

Division of Information and Communication Sciences
Macquarie University
Sydney, New South Wales, 2109

margm@ics.mq.edu.au

Abstract

Series Parallel (SP) digraphs are a common method of representing instructions with partially ordered actions, but not always an ideal one, as relationships among groups of actions may not be easily noticed when a plan becomes large. Nor are traditional linear textual instructions the best representation, since, while they do allow explicit reference to actions or groups of actions, they do not facilitate human agents working in parallel. We present the use of SP digraph analysis as a method of overcoming this problem. We employ Natural Language Generation techniques to produce English instruction output from the analysis, which embodies the advantages of both these representations.

Introduction

Series Parallel (SP) digraphs can be used to represent instructions whose constituent actions are partially ordered, for example in PERT networks, though we should note that instructions can also have an underlying structure that is non-SP.

One of the uses of SP digraphs is to allocate the tasks of a plan among several agents. This is possible due to the fact that these graphs show which groups of actions may be performed in parallel with each other and which may not. In this way, agents may be delegated not only individual actions but entire sections of a plan.

Presenting instructions in the form of a chart, however, is not appropriate for every situation. If a chart is too large to fit on one page, presenting this chart in an instruction book will confuse, rather than enlighten the reader, who will not have an overall view of the plan. Also, the more complex a chart becomes, the more difficult it is to perceive every relationship that the chart represents.

At the same time, traditional instruction manuals tend not to contain information that allows the kind of delegation we have described. Instead, the tasks are presented in a purely linear fashion so that the user may execute the tasks in the order given. This method of presentation is easy to achieve, but results in problems when more than one agent is executing the instructions, or when the next

task cannot be performed, for example due to materials not having arrived on time.

In these situations, purely linear instructions do not indicate to the users whether some later task, X, can be performed prior to an earlier task, Y being completed. Even if the prerequisites of each task were given, a user cannot decide if some group of actions can be performed independently of tasks other users are performing, without examining the prerequisites of every action in this group and frequently communicating with other users regarding their progress.

What we would like are instructions that give us the best of both worlds; the parallel and series relationships that are so easy to represent in a graph - but with relationships between groups of actions explicitly specified, so that the reader will not miss them among the many relationships contained in a large plan. As parallelism becomes more embedded, indicating group relationships unambiguously becomes more difficult. It also becomes difficult when dealing with plans whose underlying structure forms a non-SP graph.

Graph Analysis Method and Application

We present an analysis of the underlying SP digraphs of a plan, based on series-parallel graph decomposition, combined with a collapsing of purely linear sequences. Natural Language Generation techniques are subsequently applied to the output of this analysis, resulting in corresponding English instructions.

The analysis performed on the SP digraph is specific to the process of text generation that it precedes. The emphasis is on obtaining results that permit a natural aggregation of plan components into groups that conform to series and parallel relationships. This in turn permits the use of natural language generation techniques to emphasize sequence relationships when sequential execution is important, and parallel relationships when parallel execution is possible, without compromising one or the other.

The output has both the desirable characteristics we have described, as our analysis of the SP digraph is what drives the process of text generation. The analysis can also be used to produce visualizations of the graph that result in instructions promoting maximum use of parallelism by a given finite number of human agents.

¹ Copyright © 2001, Australian Computer Society, Inc. This paper appeared at Visualisation 2000, Pan-Sydney Workshop on Visual Information Processing. Conferences in Research and Practice in Information Technology, Vol. 2. P. Eades and J. Jin, Eds. Reproduction for academic, not-for profit purposes permitted provided this text is included.

Our instruction generation method is a continuation of our previous work (Wasko² and Dale 1999), and improves significantly on instructions for partially ordered plans previously produced by Mellish and Evans (1989), especially with respect to elimination of ambiguity.

Our method deals with all SP digraphs, and is currently being embodied in a program that produces this output automatically. Next, we intend to use methods of non-SP digraph analysis to produce English multiple-user instructions whose underlying structures are graphs that are non-SP.

References

- WASKO, M., and DALE, R. (1999): Generating Natural Language Descriptions of Project Plans. *Lecture Notes in Artificial Intelligence: Proc. AI99 12th Australian Joint Conference on Artificial Intelligence*, Sydney, Australia, **1747**:109-121, Springer-Verlag.
- MELLISH, C. and EVANS, R. (1989): Natural Language Generation from Plans. *Computational Linguistics* **15**(4):233-249.

² The author of this abstract was previously known as Margaret Wasko