A New Scheme for Dynamic Processor Assignment for Irregular Problems

Ranjan k Sen
Indian Institute of Technology
Kharagpur, India

We present an experimental study of a new approach for task assignment for irregularly structured problems under dynamic conditions. Our method is fast and gives solutions that are always close to those obtained for simulated annealing. The target parallel computer we have considered has the Boolean n-cube interconnection structure.

1 Introduction

An important phase in parallel computation is the assigning of processors with computational jobs. The overall computational load may be equipartitioned and assigned evenly among the available processors. This results in an even balancing of computational loads. However, the demand for interprocessor communication can considerably influence efficiency. This is particularly so for message oriented distributed memory multi-computers with different interconnection topologies.

Ideally, one would like to match the inter-task communication structure with the processor interconnection topology. A formal graph theory model of this problem was given by Bokhari [9]. The general mapping problem is more difficult than the subgraph isomorphism problem which is known to be NP complete [6]. Several graph embedding schemes for special source and target graphs have been found. For the general case numerous heuristic techniques, that includes randomized techniques such as simulated annealing, have been reported [5,7,10,11,13].

A basic feature of irregularly structured problems that are expected to run on a distributed memory parallel computer is the uncertainty of its structure. In this situation we can use heuristic techniques of the kind mentioned above or randomized techniques such as simulated annealing. The costs of the heuristic solutions are determined through standard benchmarks that are usually not representative of irregular problems. There is not guarantee of the cost in general. Techniques such as simulated annealing takes considerable processing time and are not suitable for dynamic situations. What we need is a techniques that can ascertain bounded cost solutions in general and take acceptable processing time.

In this paper we propose a mapping scheme for the hypercube target computer that guarantees a bounded cost solution in general. The running time of the
scheme is very low. Thus, our scheme is highly suitable for computations with irregular structure. Also, as the running time is low it can be used for remapping for cases that require dynamic mapping [16].

The mapping scheme is based on finding a maximum matching of a graph followed by isometric embedding on an n-cube. The work has been motivated by the availability of bounded error approximation algorithms for obtaining a suboptimal solution of maximum cut [2,3,18]. We focus on an objective function for the mapping that encodes only the cost of interprocessor communication. We assume that an even distribution of computation on to the processing nodes is made (without actual allocation of which processor has which task). The problem then reduces to that of optimizing the allocation of tasks to the processors so that communication overhead is minimized. The volume of interprocessor communication is determined by the extent of communication necessary between the processors. We simplify by assuming this to be unity. We can easily extend this to higher values by assuming parallel edges in the source graph (computational graph). The source graph is an undirected graph which represent the computation after the initial partition with respect to computational load has been made. In using resource costs we have used two simple objective functions. The first is that of maximization of cardinality [9] and the second is the minimization of the average of the lengths of the paths in the n-cube that connects communicating nodes.

2 The mapping problem

A parallel computation is characterised by a problem graph in which the vertices correspond to the tasks (parts of computation) and the edges to the data communication between the tasks. The weight of a task represents the computational load of the task. Similarly, the weight of an edge gives the relative amount of communication between the two tasks. The parallel computer can also be represented as a graph, called the host graph. Both the source and the host graphs are undirected. The vertices in the host graph correspond to the processors and the edges represent communication links. The mapping problem is that of determining a suitable map that assigns the vertices of the problem graph to those of the host graph. One common objective function used in early works on mapping is the maximization of the number of edges of the problem graph that have adjacent images of the end vertices in the host graph (cardinality)[9]. Later, various other objective functions were used [8]. In [12] Shen and Tsai minimizes the maximum turnaround time over all processors. Turnaround time for a processor is given by the sum of the computation and communication load. Another type of objective