Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Distributed Mining of Molecular Fragments

Giuseppe Di Fatta, Michael R. Berthold

RT-ICAR-PA-03-08 **Dicembre 2003**

Consiglio Nazionale delle Ricerche, Istituto di Calcolo e Reti ad Alte Prestazioni (ICAR)
– Sede di Cosenza, Via P. Bucci 41C, 87036 Rende, Italy, URL: *www.icar.cnr.it*
– Sezione di Napoli, Via P. Castellino 111, 80131 Napoli, URL: *www.na.icar.cnr.it*
– Sezione di Palermo, Viale delle Scienze, 90128 Palermo, URL: *www.pa.icar.cnr.it*

Consiglio Nazionale delle Ricerche
Istituto di Calcolo e Reti ad Alte Prestazioni

Distributed Mining of Molecular Fragments

Giuseppe Di Fatta[1], Michael R. Berthold[2]

Rapporto Tecnico N.8: **RT-ICAR-PA-03-08**

Data: **Dicembre 2003**

[1] Istituto di Calcolo e Reti ad Alte Prestazioni, ICAR-CNR, Sezione di Palermo Viale delle Scienze edificio 11, 90128 Palermo
[2] University of Konstanz, Computer and Information Science, 78457 Konstanz, Germany

1. INTRODUCTION

The main goal of life sciences research is the discovery of new drugs. All major pharma companies rely on the introduction of several new, highly successful new drugs (so-called blockbuster medications) per year in order to finance the extremely expensive and lengthy drug discovery process. Typically, it takes about 10 years until a newly identified drug candidate reaches the market. Despite advances in the analysis of the human genome and better understandings of the underlying biological interactions, one crucial step in drug discovery remains the so-called High Throughput Screening and the subsequent analysis of the generated data. In this screening, hundreds of thousands of potential drug candidates are automatically tested for a desired activity, such as blocking a specific binding site or attachment to a particular protein. This activity is believed to be connected to, for example, the inhibition of a specific disease. Once all these candidates have been automatically screened it is necessary to concentrate on a few hundred promising candidates for further, more careful (and cost-intensive) analysis. Many tools concentrate on techniques that allow the biochemist to explore the results of the screening analysis to determine which molecules to investigate further. This step is crucial for the success of the entire drug discovery process. Loosing a potential blockbuster drug here can result in a loss of up to one billion euro later on. A promising approach focuses on the analysis of the molecular structure and the extraction of pieces of these molecules that are correlated with activity. Often these pieces consist of subgraphs, that is groups of atoms and their connectivity. These pieces can then be used to identify groups of promising molecules to the user because of the representation, which is immediately understandable to the chemist/biologist.

A number of approaches to find such discriminative "molecular fragments" have recently been published [FSG, gSPAN, MoFa] but they are all limited by the complexity of the underlying problem. Finding frequent subgraphs in a set of graphs, which this problem can be translated to, is computationally extremely expensive. Some of these algorithms can therefore operate on very large molecular databases but only find small fragments [FSG, gSPAN] whereas others can find larger fragments but are limited by the maximum number of molecules they can analyze [MoFa, MolFea].

2. METHODOLOGY AND TECHNICAL ISSUES

Finding discriminative fragments in a set of molecules can be seen as analyzing the space of all possible fragments, that is all subgraphs that can be found in the entire molecular database. Obviously this set of all existing fragments is enormous, a single molecule of average size can already contain on the order of hundred thousand different fragments. Existing methods to find discriminative fragments usually organize the space of all possible fragments in a lattice, which models subgraph relationships, that is, edges connect fragments that differ by exactly one atom and/or bond. The search then reduces to traversing this lattice and reporting all fragments that fulfill the desired criteria. Based on existing data mining algorithms for market basket analysis [APRIORI, ECLAT] these methods conduct depth-first [MoFa] or breadth first searches [gSPAN, FSG]. Distributing such search algorithms on parallel resources is non-trivial and an existing collaboration of the host institution with the University of Erlangen and Partek concentrates on solutions for closely parallelized computing resources. The Programming Languages group at University of Erlangen has developed an automatic parallelization virtual machine for Java (JavaParty) and Partek develops massively parallel computers. Many problems, such as distributing the original database aren't as critical since closely coupled computation resources usually have shared, fast access to external storage resources.

The proposed approach focuses on distributed computing platforms, in particular in the form of a GRID, that is, collections of relatively loosely coupled, diverse computing resources. Issues related to the distribution of the initial molecular database become critical in this context. Also, the synchronization of intermediate results (such as which parts of a search tree have been traversed already or fragments that were already reported as discriminative) is non-trivial.

A Grid environment [Foster01, Foster02] provides high performance computing facilities and transparent access to them in spite of their remote location, different administrative domains and hardware and software heterogeneous characteristics. A Grid is a combination of distributed and heterogeneous computing, storage and communication resources for executing large-scale applications. A distinction is sometimes made between Data Grids and Computational Grids. Computational Grids normally deal with large-scale computationally intensive problems on small data sets; while Data Grids deal with large-scale data-intensive problems on large amount of data, i.e. typical data mining problems.
In the context of molecular fragments analysis both aspects are present. This makes the effective exploitation of a large-scale computational and storage system a very complex and challenging task.

In general the choice of the appropriate algorithm used to schedule jobs depends on the application. Our focus will be on scheduling algorithms that are suitable for large-scale data-intensive problems. In this case data location is relevant in the operation of job assignments to computational nodes.
When data are stored in a single location, e.g. a centralized database, a job execution has to be preceded by a data movement operation. If the amount of data is particularly relevant, the distributed implementation may be not efficient due to communication overheads and even limitations in the local storage. In this case replications of data can be adopted in order to reduce data movement overhead. Job assignments are performed by a job scheduler and data replications by a dataset scheduler. Their coordination and interaction will determine the overall efficiency and to this aim an optimization strategy has to be devised.
The study of dataset scheduler algorithms and the strategies in the job assignment/dataset replica coordination is still an open field.
Two different approaches have been proposed in [RAFO02] and [BCCM02], which move from opposite considerations.
In [RAFO02] it is assumed that a popular file (dataset) in one site is potentially popular in other sites. Hence, a replication of a popular file is actively created to a destination site, which is chosen either randomly or by selecting the least loaded site. Unfortunately, if the assumption does not hold, data replica may introduce inefficiency in the Grid storage management.
In [BCCM02] the generalization of file popularity has not been assumed. So, dataset replications are created only where explicit access requests are made ("data hotspots"). In this case data access latency may be an issue.
Both job and data scheduling algorithms have a significant impact on the system performance and their choice depend on the particular problem.

The optimization criteria, which will be considered, are the minimization of data replication operations and, of course, the maximization of the response time. The latter can only be achieved through high computation throughput and low data access latency.
Due to the heterogeneous, multi-domain, dynamic nature of Grids, robustness and fault tolerance have to be taken into account in the design of the distributed algorithm.

Note on data access

Bioinformatics databases publicly provide relevant data using a wide range of different systems and format. It has been pointed out the need of open source standards for bioinformatics data formats.
Recently, several initiatives are undergoing to adopt XML schema to provide uniform access to data. This way, data can be easily retrieved by applications through standard interfaces at the data provider website. Web Services based on SOAP over HTTP (a W3C standard) have been widely adopted as the message exchange protocol.
In a future collaboration as continuation of this project a set of web services can be developed in order to access the molecular database. These web services can be adopted by the Data Scheduler to retrieve data subset and create replica. This approach will be evaluated and discussed during the research period.

3. PROPOSED APPROACH

The distributed approach is based on the centralized algorithm described in [MoFa], where the problem of selecting discriminative molecular fragments in a set of molecules has been formulated in terms of frequent subgraphs in a set of graphs.
The method described in [MoFa] organizes the space of all possible fragments in an efficient search tree. An example of such a search tree is shown in Figure 1.
Each possible subgraph of the molecular structures is evaluated in terms of the number of embeddings that are present in the molecules dataset.

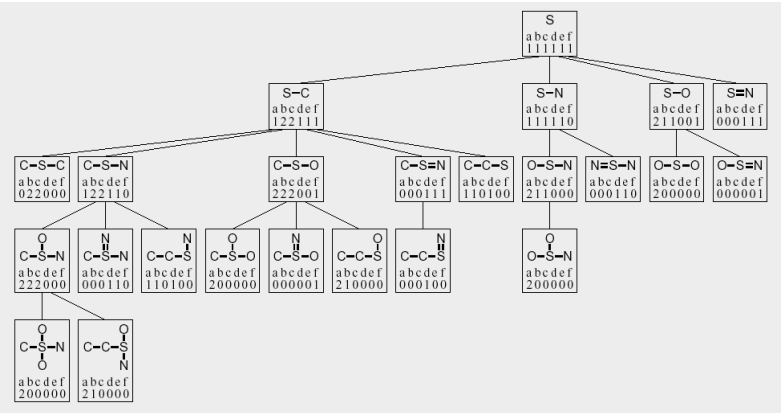

Figure 1: Search tree in molecular fragment mining

From a first analysis of the problem, we can define two approaches by partitioning either the data space or the search space in order to decompose the problem in subtasks.

1. *dataset partitioning performed by the Data Scheduler*
   The dataset is divided in smaller subsets, which are distributed in the Data Grid by the Data Scheduler. An instance of the sequential algorithm is remotely executed for each subset. Once partial results are collected, they have to be merged to produce the final result.
   In this case few issues have to be addressed:
   (a) which criteria are adopted to determine the number of subsets, their dimension, and the way the dataset is partitioned?
   (b) an efficient, possibly distributed, merging algorithm has to be designed.

2. *search space partitioning performed by the Task Scheduler*

The processes distributed in the Grid operate in the entire original dataset, but they evaluate only a subset of all possible subgraphs. Given an order to generate the search tree (e.g. lexicographic order), it can be easily distributed in a simple or multi-tier master-slave computational paradigm.
While this approach can be successfully adopted in a parallel computational paradigm, its adoption in a large-scale multi-domain Grid environment is constrained by the data access latency when no local replication of the dataset is present, and by communication overheads and storage limitations when the entire dataset has to be replicated in remote sites.

Clearly, a general solution for an effective algorithm for a Grid environment should adopt a combination of both approaches.

The appropriate solution for a distributed algorithm in molecular fragments mining, its feasibility and efficiency will be determined and evaluated during the project execution. Anyway, in the remainder of this section we briefly draw a possible direction.

Each node in the search tree (Figure 1) represents a fragment. The search tree is built by progressively growing the embeddings of molecules and this method is proved to be complete. The node evaluation process determines the frequency of such a fragment in the molecule dataset and the subset of molecules where embeddings have been found. Hence, each node implicitly defines a subset of the dataset induced by the fragment. When we consider nodes that belongs to the subtree rooted in a node f_i, we can take into account only the molecules in the subset induced by f_i without loosing completeness.
This way we can define a strategy that combines both search space and data space partitioning.
The pseudocode in Table 1 describes a recursive procedure that implements this strategy, and Figure 2 shows the search tree with the pruning, the replication of the induced subset and the spawning of new processes.

```
Procedure fragment_mining (dataset M, core C)
-   init root_node <- C
-   init dataset_ref <- M
-   recursively generate and explore the search tree (e.g. by DFS)
      o  determine fragment frequency
      o  M_i <- compute molecules subset induced by the fragment f_i
      o  if a given spawning criterion is met
            ▪  request a replication M_i of the induced subset to the
               Data Scheduler
            ▪  request a new process fragment_mining(M_i,f_i) nearby the
               location of the new subset replication
            ▪  prune the search tree in node f_i and proceed
      o  else expand node
```

Table 1: Pseudocode of the distributed mining algorithm

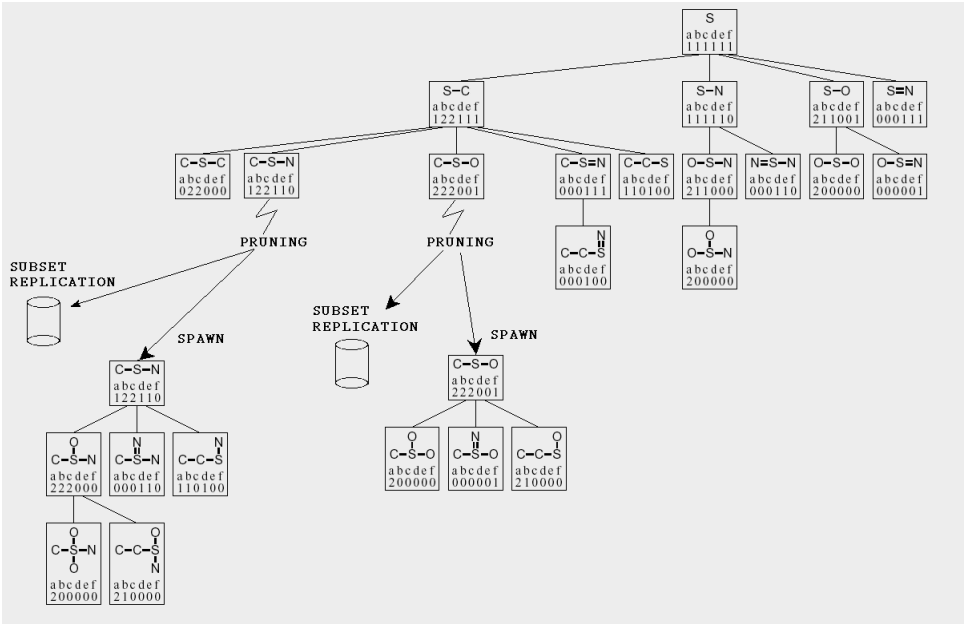

Figure 2: A distributed approach

Load balancing and scalability

In the above solution the work unit is a subtree, whose dimension and equivalent workload may be difficult to be estimated. As a consequence this may lead to a coarse partitioning and load balancing and scalability problems.
A general distributed paradigm described in [LaSa02] could be suitable for our case. It proposes an efficient data and search tree management for applications in distributed computing environments, which is based on a Master-Hubs-Workers paradigm (Figure 3). The Master has global knowledge and assigns subtrees to Hubs in order to balance load among them. A Hub manages a collection of subtrees and balances the load among workers. Each Worker processes a subtree.

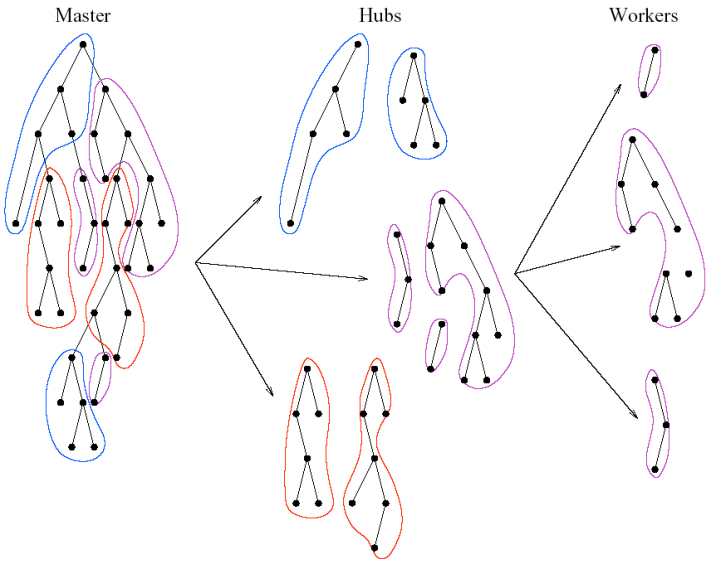

Figure 3: The Master-Hubs-Workers distributed computation paradigm

In our case, the Master process would also create and assign dataset replications to Hubs. And Workers will access data through the SOAP protocol at the Hub they are associated.

The system of dataset replications in the Hubs can be regarded as a distributed and redundant database. Hierarchical protocols for distributed databases have been successfully adopted in the Internet, the most known of which is the DNS protocol [RFC1034, RFC1035].
Paradigms for distributed computation and distributed database still have to be evaluated and will be the object of a future work.

4. CONCLUSIONS AND FUTURE WORK

This technical report describes a large scale distributed approach based on a GRID infrastructure for the search of molecular fragments in extremely large databases.
The adoption of a large scale approach is extremely interesting and opens new possibilities in BioSciences. But its actual effectiveness and the new problem it poses have to be investigated yet. The aim of this report is to perform a preliminary study and to provide insight on the difficulties and on the actual advantages.
The research challenges discussed in this document will be the object of cooperation between ICAR-CNR and the University of Konstanz. One of the research areas at the Bioinformatics group of the University of Konstanz concentrates on the optimization of these methods to make them useable on extremely large datasets (millions of molecules) unlimited by the size of the fragments that can be discovered. Quite obviously, parallel approaches to this type of problem are a promising alternative to the current sequential algorithms.

5. BIBLIOGRAPHY

[APRIORI] R. Agrawal, T. Imielienski, and A. Swami, “Mining Association Rules between Sets of Items in Large Databases”, Proc. of Conf. on Management of Data, 207--216.

[BCCM02] W. H. Bell, D. G. Cameron, L. Capozza, A. P. Millar, K. Stockinger, F. Zini, “Simulation of Dynamic Grid Replication Strategies in OptorSim”, 3rd International Workshop on Grid Computing , Baltimore, MD, USA, 18th November 2002.

[ECLAT] M. Zaki, S. Parthasarathy, M. Ogihara, and W. Li, “New Algorithms for Fast Discovery of Association Rules”, Proc. of 3rd Int. Conf. on Knowledge Discovery and Data Mining (KDD'97), pp. 283-296, AAAI Press, Menlo Park, CA, USA 1997.

[Foster01] I. Foster, C. Kesselman, S. Tuecke, “The Anatomy of the Grid: Enabling Scalable Virtual Organizations”, International J. Supercomputer Applications, 15(3), 2001.

[Foster02] I. Foster, C. Kesselman, J. Nick, S. Tuecke, “The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration”, Open Grid Service Infrastructure WG, Global Grid Forum, June 22, 2002.

[FSG] M. Desphande, M. Kuramochi, and G. Karypis, “Automated Approaches for Classifying Structures”, Proc. of Workshop on Data Mining in Bioinformatics (BioKDD), 11-18, 2002.

[gSPAN] X. Yan and J. Han", “gSpan: Graph-Based Substructure Pattern Mining", Proceedings of the IEEE International Conference on Data Mining ICDM, Maebashi City, Japan, IEEE Press, Piscataway, NJ, USA, 2002.

[LaSa02] L. Ladányi, T.K.R., and M.J. Saltzman, “Implementing Scalable Parallel Search Algorithms for Data-intensive Applications”, The Proceedings of the International Conference on Computational Science (2002), Volume I, 592.

[MoFa] C. Borgelt and M. R. Berthold, "Mining Molecular Fragments: Finding Relevant Substructures of Molecules", Proceedings of the IEEE International Conference on Data Mining ICDM, Maebashi City, Japan, IEEE Press, Piscataway, NJ, USA, 2002.

[MolFea] S. Kramer, L. de Raedt, and C. Helma, “Molecular Feature Mining in HIV Data”, Proc. of 7th Int. Conf. on Knowledge Discovery and Data Mining, (KDD-2001, San Francisco, CA), pp. 136–143, ACM Press, New York, NY, USA 2001.

[RaFo02] Kavitha Ranganathan and Ian Foster, “Decoupling Computation and Data Scheduling in Distributed Data Intensive Applications”, International Symposium for High Performance Distributed Computing (HPDC-11), Edinburgh, July 2002.

[RFC1034] P. Mockapetris, "Domain Names - Concepts and Facilities", RFC 1034, Nov. 1987.

[RFC1035] P. Mockapetris, "Domain Names - Implementation and Specification", RFC 1035, Nov. 1987.