Financial Information Grid –an ESRC e-Social Science Pilot

K. Ahmad [1],T. Taskaya-Temizel[1], D. Cheng[1], L. Gillam[1], S. Ahmad[1], H. Traboulsi[1], J.Nankervis[2]

[1]Dept. of Computing, University of Surrey, [2]Dept. of Economics, University of Essex

Abstract

The large volume of time-varying quantitative and qualitative data available online is needed in a realistic simulation in theoretical econometrics from an academic perspective. And the analysis of such data is strategically important for trading in the financial markets. A distributed environment has been created that offers two services – time-serial and news analysis – using Globus and Java Commodity Grids. A demonstrator that can be used for bootstrapping and for detecting financial market 'sentiment' in real time is reported in this paper.

1. Introduction

Financial markets are strategically important for any economy. The markets generate substantial amounts of data, including numerical and textual data about financial instruments. The markets have a symbiotic relationship with reports related to people and organizations within the markets and outside. Here there is a deluge of both quantitative and qualitative data that is used on the one hand in developing theories of how markets behave, and on the other by the market traders, regulators and the public at large. Many areas of social science require the handling of the two types of data that is available from different sources and has to be processed speedily and efficiently: a grand challenge for grid technologies and e-Science.

There are three specific areas that we deal with in the multidisciplinary Financial INformation Grid (FINGRID) project involving computer scientists and econometricians: First, large scale-simulation for building and testing reliable models; second, the analysis of textual data about the markets, especially related to perceptions about the markets; and third, the fusion of the results of the analysis of qualitative data with the quantitative for generating a decision. A 13 node grid has been developed and interfaced with a real-time news and data feed.

BOOTSTRAPPING LARGE-SCALE SIMULATIONS: The computation of risks in financial markets is the focus of much empirical and theoretical development. Statistical inferences obtained from distributions of simulated data are reported to be more reliable than inferences gained from asymptotic theory; distributions based on asymptotic theory are strictly valid only when the sample size is infinitely large' (J. MacKinnon 2002). Bootstrap tests and Monte Carlo tests are examples of simulation-based tests. In order to make more realistic statistical inference from data using bootstrapping, one should use significant amount of replication numbers (c. 10000 times).

MARKET SENTIMENT: In addition to the very quantitative data related to trading volumes and price movements, the financial traders rely on market sentiment: This sentiment, is often expressed in news reports and editorials, and ranges from views about national economies to the imminent take-overs, mergers and acquisitions and from people leaving/joining an organization to news about political and economic successes and failures.

Methods based on corpus linguistics and information extraction have been used to identify *sentiments* in as unambiguous manner as is possible from natural language texts. Specifically, methods have been developed (a) for identifying key terms in financial reports; and (b) for learning a local grammar that is used for constructing phrases to express sentiment about financial instruments. Automatic terminology and ontology extraction and the identification and use of local grammars is domain independent and has the scope for much wider application (H. Traboulsi *et al.* 2004).

FUSING QUANTITATIVE AND QUALITATIVE INFORMATION: Time serial data related to financial instruments[i], for example, currency, stocks, derivatives, often exhibit nonstationarity. In order to extract long-term trends, seasonal variation, and the random component, in a complex time-series, increasingly multi-scale analysis is used that

deals with both time and frequency domains, for example, wavelet analysis with some success (S. Ahmad *et al.* 2004).

The positive and negative sentiments related to a financial instrument, for instance, stock-exchange index, currency, or share, change over time and can be ordered as a time series. This series is correlated with that of a time-series of the traded volume or price of an instrument (or multi-scale analysis of the same), for generating *buy* or *sell* signals. An initial prototype for data fusion of a sentiment time-series and that of an instrument has been developed and is available on the web (T. Taskaya *et al.* 2003).

The FINGRID project is investigating the relevance of the Grid in particular and e-Social Science in general for dealing with quantitative and qualitative financial information. The problems being looked at in the project are both data- and compute-intensive: furthermore the data is proprietary and the computation costs includes the provision of a cluster of machines.

2. Motivation

What motivates us is the confluence of model building and data fusion activities that are traditionally conducted along a quantitative/qualitative divide. Financial traders merge conclusions made by those across the divide as they regard both types of data and analysis as equally important. Much the same is true of policy and decision makers involved in a range of socio-economic fields. E-Science methods and techniques can help in both model building and data fusion.

We describe the two out of three key areas of our activities in the FinGrid project,

(a) bootstrapping,
(b) sentiment analysis,

and the current grid-oriented interest in these areas. For reasons of brevity, the third area, multi-scale analysis has been covered elsewhere (S. Ahmad *et al.* 2004).

2.1. Bootstrapping

2.1.1. Definition

The financial services perform complex financial analysis on a deluge of data coming from many data vendors such as banks, stock markets, and companies. There are many computer-based methods such as bootstrapping, which can be classified as solutions for 'compute-centric' problems. These methods require considerable number of iterations or resampling for producing reliable output. However, the number of iterations can be set up to a limit, which an average computer can handle. This means that the modellers have to contend with small-scale simulations thereby compromising the output (J. Eckles 2003).

Bootstrap procedures are defined as data based simulation methods that estimate the distribution of estimators by re-sampling observed data. Bootstrap method assumes that the observed data is a representative of the unknown population. The simple bootstrap algorithm works on independent, but not necessarily identically distributed and strictly stationary data. The algorithm works as follows:

1. Let X be an observation and n the size of the observation: $\mathbf{X} = (x_1, x_2, \ldots, x_n)$;
2. Draw a random subsample of size n from X with the subsample replaced B times (B is called the replication number)
3. Test statistics on the simulated data.

For drawing realistic statistical inference, replication numbers between 10,000 and 20,000 are favoured. Recently, interest has been shown in *block bootstrapping*, where data is divided into blocks and then the blocks are re-sampled randomly with replacement. Robustness offered by block bootstrapping makes computing stock and bond returns attractive.

2.1.2. Current Practice

There has been considerable interest in grid technologies within the financial markets. This is largely motivated by the observation that only 10% of the power of a desktop machine is effectively utilised thus leaving the desktop dormant most of the time. Grid technologies provide ways for soaking up this dormant power. The interest here is in computing risks and portfolios using simulation based tests. The computation using grid technologies shows considerable performance improvements. Consider the following three prototype systems:

(a) *Risk Management:* IBM, working together with Morgan Stanley, Hewitt Associates and NLI Research Institute, have successfully outperformed in various financial analysis applications' computing time - a reduction to 49 minutes from 4 hours - in a financial risk management solution using a Monte Carlo simulation.

(b) *Portfolio Computation:* Statistical models for calculating the size of capital needed against a given portfolio has been reported to have completed in less than 10 minutes using grid

technology as compared to a minimum of 120 minutes (M. Friedman 2003).
(c) *Legacy Computing and Migration: GridServer*[TM] and *GRIDesign*[SM], two Grid-based technologies, have been used in migrating existing conventional programs onto a grid[ii] using Globus.

2.2. Sentiment Extraction

2.2.1. Definition

The behaviour of financial markets is governed by the gains and losses of the investors in the markets. A qualitative, largely intuitive, and often disputed factor that may influence the behaviour is market sentiment: "the mood of a given investor or the overall investing public, either bullish or bearish"[iii]. The metaphors, *bullish* and *bearish*, so-called animal metaphors, refer to the aggressive or recessive (shy) mood of the investors and perhaps of the traders. Sentiment or market sentiment is related to the very quantitative *market volatility index*: an index based solely on the increment and decrement in prices of financial instruments and is "used as an indicator of investor sentiment, with high values implying pessimism and low values implying optimism"[iv].

Financial reports, especially summaries of stock market behaviour on an hourly or daily basis, are reported in terms of stocks that *rose* most and stocks that *fell* most. The market movement is described in terms metaphors related to market trends and cycles: uptrend, downtrend, boom-and-bust *cycles*, *peaks* and *troughs* (of cycles). The expression of optimism or pessimism relies on a choice of words whose meaning is generally understood. This is not to say that the words used in the expression of market sentiment have been standardised much in a way the terminology of science and technology are standardised. Rather, there is a general consensus on how to express optimism or pessimism about an instrument.

The words *rise* and *fall* have many senses and do cross grammatical categories; each could be a noun or a verb. However, financial report writers constrain the meaning by encoding the words within specific patterns – a local grammar. The *verb* rose has many senses , for example, General Rose, color of rose, shares rose, but financial reporting has appropriated the word by co-locating its use with a cardinal number. Therefore, one can remove ambiguous patterns and focus on patterns such as *bond rose 3.1 percent*, *dollar fell by 2 points*, which obey the local grammar.

2.2.2. Current Practice

Various organisations perform news analysis and extract "intelligence" from news, in the light to advise traders/investors on the basis of the analyses that includes sentiment analysis to buy or sell:
(a) The investment bank Dresdner Kleinwort Wasserstein[v] offers a range of advisory services across the Capital Markets.
(b) *Investor Sentiment Report*[vi] is *published* fortnightly and recommends a list of individual stocks that may deliver significant gains for investors.
(c) Spotter[vii] monitors and analyses international press coverage to provide subscribers with insight and understanding on news stories that are of interest.
(d) *Google News* and *Yahoo! News* provide similar services based mainly on general news.

Financial news is an important resource for financial traders. Studies have been conducted to explore the cause-effect relationship between the news and the financial data (V. Lavrenko *et al*. 2000). The studies concluded that there is a strong correlation between the two.

A trader usually scans the news titles and browses the news of interest. As the volumes of available news (5 news item per minute and each between 200-1,000 words long) increase, this task cannot be carried out entirely by humans. We present the chief points of a news item to a trader. When the amount of news is considered, over 2 GB per year, we need either very efficient algorithms (achievable only to some degree) or a distributed environment, which will meet the computational economy requirements. This is the focus of the FINGRID project, but more of that later. We next consider the current state of 'play' in sentiment analysis; a study of the language used in financial reports may show patterns that may perhaps be automatically identified.

2.2.3. Free Text and Sentiment

The lexical environment, essentially the frequently occurring words used in phrases and sentences to embed sentiment terms is quite unique - it is not a pattern used elsewhere in the language except for perhaps in financial reports only. The local grammar - the order in which the surrounding words are used in the sentiment

carrying phrases - is quite constrained and the constraints can be used to clearly identify whether the term was used in a specialist sense - markets moving *up* or *down*- or in an everyday usage *John went up in a lift.* Given sufficiently large randomly-sampled usage of sentiment words, it is possible to have a statistical basis for identifying these words and to learn about the local grammars. This is a computationally intensive task and a grid cluster does appear to speed up the learning and the subsequent identification and extraction of sentiment words (see section 3.3.2).

Corpus analysis techniques have been developed to study languages at various levels of linguistic description, vocabulary, grammar, semantics and pragmatics, by relying almost exclusively on texts and speech produced by language users. Frequency analysis of linguistics tokens, for example, words at the level of vocabulary; phrases and sentences at the level of grammar; relations between the vocabulary and grammatical levels for inferring meaning like collocation, are undertaken. The results of the frequency counts are used to generate statistical metrics and quantitative results are produced, and linguistic hypotheses accepted and rejected using the metrics. Corpus analysis techniques work well with special languages.

There is increasing interest in analysing general language at different levels of linguistic description using grid technologies:

(a) FREQUENCY COUNTS OF TOKENS, TAGGING AND INDEXING: Recently, three NLP studies, which were conducted using Grid environment, have been reported. B. Hughes *et al.* (2004) and B. Hughes *et al.* (2003) undertook some NLP tasks such as word-frequency counting, indexing and training a part-of-speech tagger on two major corpora of English: Brown Corpus and English Gigaword Corpus. The decomposition process was carried out on data.
(b) MORPHOLOGICAL ANALYSIS AND TAGGING: F. Tamburini (2004) attained functional decomposition of the Italian National corpus called CORIS corpus, in which one machine was dedicated to morphological analysis and the other was for POS tagger.

The FINGRID project has developed methods and techniques that can detect sentiment words and can learn the local grammar governing the use of such words. This has been parallelised and implemented on our grid with considerable speed-ups (section 3).

A grid-enabled large-scale analysis of specialist texts will contribute to the emergence of the *semantic grid* (D. Roure *et al.* 2003). What we aim in Fingrid project is to contribute to the Semantic Grid by providing an environment, which will aid multi-modal information fusion and help statisticians building better models.

3. Fingrid

3.1. Architecture

3.1.1. Overview

The Grid-based analysis, modeling and prediction using financial information, both quantitative and qualitative, require a three-tier architecture. The first tier facilitates the client in sending a request to one of the services (Text Processing Service or Time Series Services) situated in the main cluster comprising machinery for the execution of the two services. The second tier facilitates the execution of parallel tasks in the main cluster and is distributed to a set of slave machines (nodes). The third tier comprises the connection of the slave machines to the data providers. Given an allocated task, the corresponding data is retrieved from the data providers by the slave machines. The main cluster monitors the slave machines until they have completed their tasks, and subsequently combines the interim results. The final result is sent back to the client machine. This architecture has been implemented and shown below.

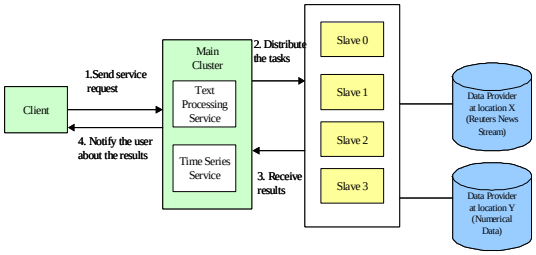

Figure 1: Fingrid Architecture

3.1.2. Services

FinGrid provides two Grid services – the Text Analysis Service and Time Series Service.

In the Text Analysis Service, the news and executables are distributed to the slave clusters. When the cluster finishes its associated task, the

interim result is sent back to the main machine, where the interim results are collated.

The second service is the Time Series Service, which currently performs two functions: *bootstrapping* and *wavelet analysis*. In *bootstrapping*, the total replication number is set in the main cluster. Then, the central program generates *n* number of random replications of the original data. The random replications are distributed evenly among the slave clusters.

Subsequently, for both the services each slave machine performs calculations. At the end, the interim results are sent back once the calculations are finalised. The central program aggregates them and produces the final result. The *Wavelet analysis*, implemented in Matlab, was transformed into a Grid service with the help of JMatlink[viii]. Given a time series, this service generates a web page, which comprises information with respect to the turning points, variance change, trend information and cycles. Then, this web page is transferred to the client machine, where the information is requested.

3.1.3. Configuration and Throughput

Currently our infrastructure consists of thirteen nodes (7 Dell PowerEdge 2650 with 1 GB memory and dual processors; and 6 Optiplex GX150s with 256MB memory, single processor) and a daily newsfeed provided by Reuters Financial Services (c. 35 MB or 6000 news items on average per day; one year is around 2 GB texts). We have developed programs using Reuters API to capture the news, historical time series data and tick data. Reuters supply news with categories, authorships and date information.

3.2. Technology

In our Fingrid implementation, we have employed Globus Toolkit 3.0, which has become a de facto standard in Grid applications. The Globus Toolkit is defined as a community-based, open architecture, open source set of services and software libraries that support Grids and Grid applications (I. Foster *et al.* 1999). The toolkit provides an extensive set of packages with respect to resource management, security, information services and data management. Globus version 3 exploits Open Grid Services Architecture (OGSA) capabilities. OGSA aims to bring standardisation in Grid services. Whilst it inherits several features from Web services, it adds new conventions in terms of dynamic service creation, lifetime management, notification, discovery and manageability (I. Foster *et al.* 2002). Interoperability, provided by OGSA, allows seamless collaboration of the virtual organisations.

Being a part of Globus Toolkit, resource management is accomplished via Java Commodity Grids (CogKit) (G. Laszewski *et al.* 2001). The Java CogKit provides advanced software packages, which bridges between Java and Globus in order to simplify the implementation. The Java CogKit improves the software quality in terms of software reuse, flexibility and maintenance. The Java CogKit comprises services such as GSI (Globus Security Infrastructure) for security, GRAM (Globus Resource Allocation Manager) for remote job submission and monitoring, MDS (Monitoring and Discovery System) for information service access, GSIFTP (FTP with GSI security) for remote data access and myProxy for certificate store. Benefiting from 'single sign on' capability of Grid, jobs can be farmed onto remote machines seamlessly in parallel.

Parallel scenarios in Grid are mainly inspired from the algorithms developed for multi processor machines. There are three major parallel processing techniques (D. Aspinall 1990). The first one, *algorithmic parallelism*, divides an algorithm into sequential parts, in which each fraction will be carried out by a single processor. The second is the *geometric parallelism* technique, which is data-centric: data is distributed statically among the processors and each processor is responsible to work on its assigned data partition. The third technique is called *processor-farming parallelism*, where there is a central controller that divides (a) a task into independent sub tasks (b) allocates each task to any available processor (c) after the completion of tasks, the central controller collects the jobs from all processors and aggregates the results. The processor farming parallelism can be mapped onto a grid cluster – processors are substituted by individual computers. Our FINGRID implementation uses processor farming.

There are other sophisticated job management technologies such as Condor, which provide add-ons like job scheduling, and job distribution. Condor-G, which is a marriage of Condor and Globus technologies, offers more compared to current Globus resource

management technologies. In addition to that, for achieving Grid-based parallelism, there exists a low-level programming model called MPICH-G2 (A Grid Enabled Message Passing Interface), which supports efficient and transparent execution of applications in Grid environment (N. Karonis *et al.* 2003). We aim to benefit from Condor-G and MPICH-G2 technologies at a later stage of the project. The reason is that currently the implemented algorithms in Grid were not designed to support social scientists to conduct experiments on a continual basis. We need to create an environment, which will allow statistical modelling and automatic content analysis to be achieved effortlessly but in a restricted context. We believe that Condor-G and MPICH-G2 will be useful for creating this flexible environment, in terms of parallelism and resource discovery.

3.3. Case Studies & Results

3.3.1. Bootstrapping

In this experiment, we employed existing Fortran implementations of a bootstrapping algorithm. We measured the processing time of the bootstrapping program with different grid node configurations starting from two-node to eight-node. Like others, we have observed a proportional degradation in performance gain as the number of node is increased.

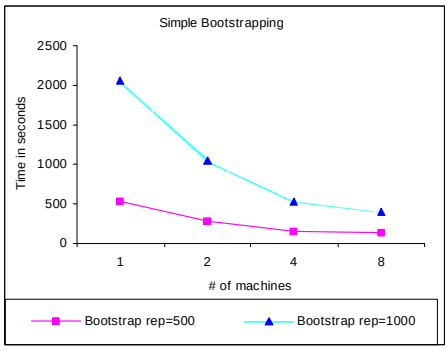

Figure 2: The Bootstrapping service's performance improves when the number of machines is increased

3.3.2. Text Analysis Service

We have followed B. Hughes *et al.* (2003) word frequency counting approach to evaluate the performance of our Grid implementation. The corpora used in our experiments are the Brown Corpus and the Reuters RCV1 Corpus: see Table 1 for details.

Table 1: Distribution of the corpora

	Files	Size (Mb)	Words (M)
Brown	500	5.2	1.0
RCV1	806,791	2576.8	169.9

For the Brown Corpus, the number of words processed per second is similar to Hughes *et al.*, 7,120 versus 6,670 in a single CPU system. This is not the case when the process is carried out on a two node grid. Our grid implementation shows a 98% gain of performance, whereas Hughes *et al.* implementation shows a 20% performance degradation. One may argue that conditions in which the experiments were carried out vary significantly – one is on a local grid, and the other operates over the Internet. The configuration of Hughes *et al.*'s SMP is the same as our two node grid, but whilst Hughes *et al.* show a 27% performance gain, our performance is still 98%. To demonstrate the computational power of our grid, we run the same process over a four node grid and a eight node grid.

Table 2: Words processed across the corpora

	Brown	RCV1
Words/s (1 machine)	7,120	-
Words/s (2 machines)	14,091	5,334
Words/s (4 machines)	23,944	10,532
Words/s (8 machines)	31,453	14,590

It appears that the relative performance of the word frequency counting experiment on the RCV1 corpus is lower than the Brown corpus. However, a key difference between the corpora is that the file format - RCV1 corpus is stored as XML while Brown corpus is in text or .txt format. Therefore it is necessary to parse only the XML files prior to processing.

In our second experiment, we extracted market sentiment from a month (January 2004) of news articles collected from Reuters Newsfeed. This task involved extracting title, date and time the news was published, counting words and sentiment words that obey local grammar for each news. There were 92,918 news articles (around 262 MB) published in that month. We employed three different grid configurations: 2, 4 and 8 node configurations. We observed a performance gain of 50% in using a two-node grid rather than a single machine and a gain of 27% in moving from a two-node grid configuration to a four-node grid configuration. In addition to that, we observed a gain of 0.04%

when moving to an eight-node grid from a four-node grid.

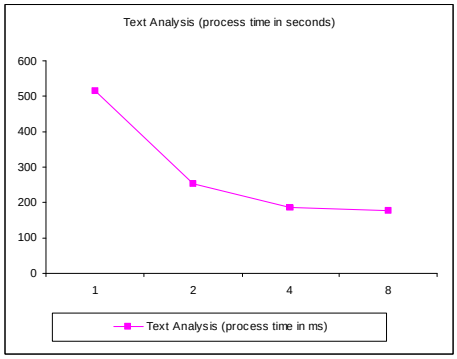

Figure 3: The performance of text analysis service improves in proportionate the number of machines

3.3.3. Fusing Qualitative and Quantitative Data Analysis

The sentiment analysis is conducted for over 50 sentiment words and the words identified using local grammars. For news item produced on a given time interval the results for individual sentiment words are aggregated for positive and negative sentiments separately – this results in two 'time series' of sentiments. It is possible to restrict the sentiment analysis to a specific instrument, as Reuters News is generally tagged with direct or indirect information about the instrument.

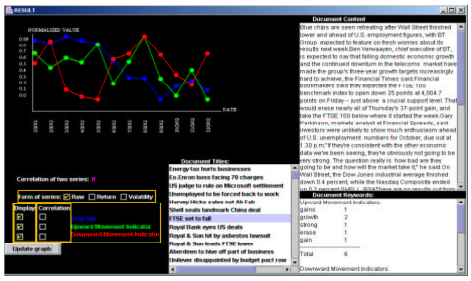

Figure 4: SATISFI in operation: Correlation between news and instrument time series. The user displays headlines for a time period, looks at a specific news item as well as the number of sentiment words.

Alternatively, sentiment analysis is performed covering all financial news available and the instrument used is typically a stock exchange index. We have developed a *sentiment analysis and time series analysis system* (SATISFI) for visualising and correlating the sentiment and instrument time series both as text (and numbers) and graphically as well. The system allows end-users to introduce 'delays' in the correlation to for testing, for example, whether yesterday's news has any effect on today's instrument movement (Figure 4).

We have been looking at more in-depth analysis of time series and are developing a grid-enabled method based on multiscale, specifically wavelet analysis. A single-machine system has been developed that can disentangle trends, cyclical variations, and random elements in a time series. The wavelet analysis helps in computing variance change and volatility in a time series and helps in computing *turning points* - where market changes direction from upwards to downwards movement.

4. Discussion

We identified the following problems that may cause the exponential gain degradation:

1. *The configurations of the machines*: During the distribution of tasks, we did not take into account the configuration of the machines. This meant that some of the machines in the experiments were always delayed in terms of producing the results; the fastest machines were idling while the rest were processing.
2. *One common data source*: All our grid-nodes share one common data source. Network latency occurs due to the number of nodes using the same bandwidth to retrieve files.
3. *Amdahl's law*: Amdahl's law is applicable to our grid, where the fraction of code *f*, which cannot be parallelised, affects speedup factor.
4. *Program constraints*: In the task distribution process, the file size is not considered except for the second experiment (processing a month of Reuters news). In addition, the number of tokens is another important criteria in word frequency computation. If the number of tokens processed by each cluster varies significantly, then this may lead to possible delays.

5. Conclusion

The FinGrid project has achieved three major objectives. First, the project demonstrates how both quantitative and qualitative data from multiple sources can be processed, analysed, and fused. This is a key question in e-Social Science where researchers typically would like to fuse social, economic and political data, which is archived under specialist disciplines. Second, it has raised considerable interest in the financial news information market (K. Ahmad *et al.* 2004). The two objectives have contributed also in terms of improvements to

goods and services and financial software houses and news vendors have shown interest in the project.

Additionally, researchers involved in the project have contributed to wider training in grid technologies by designing and successfully delivering a specialist Master's level course in grid computing.

Acknowledgements

The authors wish to thank the ESRC e-Social Science Programme for their support for the FinGrid Project (ESRC Project Number RES-149-25-0028)

References

Baden Hughes, and Steven Bird, "Grid-Enabling Natural Language Engineering by Stealth", *In Proc. of HLT-NAACL 2003 (Workshop on SEALTS)*, pp. 31-38, Association of Computational Linguistics, 2003.

Baden Hughes, Steven Bird, Haejoong Lee, and Ewan Klein, "Experiments with data-intensive NLP on a computational grid", *http://www.ldc.upenn.edu/sb/home/papers/grid-experiment.pdf* , 2004.

D. Aspinall, "Structures for parallel processing", *IEEE Computing and Control Engineering Journal*, vol. 1, pp. 15-22, 1990.

David de Roure, Nicholas R. Jennings, and Nigel R. Shadbolt, "The Semantic Grid: a future e-Science infrastructure", Fran Berman, Geoffrey Fox, and Tony Hey (Eds.), *Grid Computing: Making the Global Infrastructure a Reality*, pp. 437-470, John Wiley and Sons, 2003.

Fabio Tamburini, "Building distributed language resources by grid computing", In *Proc. of the 4th International Language Resources and Evaluation Conference*, pp. 1217-1216, Lisbon, 2004.

Gregor von Laszewski, Ian Foster, Jarek Gawor, and Peter Lane, "A Java Commodity Grid", Concurrency and Computation: Practice and Experience, vol. 13, pp. 643-662, 2001.

Hayssam Traboulsi, David Cheng, and Khurshid Ahmad, "Text Corpora, Local Grammars and Prediction", In *Proc. of the 4th International Language Resources and Evaluation Conference*, vol. 3, pp. 749-752, Lisbon, 2004.

Ian Foster, and Carl Kesselman, "Globus: A Toolkit-Based Grid Architecture", Ian Foster and Carl Kesselman (Eds.), *The Grid: Blueprint for a New Computing Infrastructure*, pp. 259-278, San Francisco: Morgan Kaufmann, 1999.

Ian Foster, Carl Kesselman, and Steven Tuecke, "Grid Services for Distributed System Integration", *Computer*, vol. 35, pp. 37-46, 2002.

Jack Eckles, "An Overview of Grid Computing in Financial Services", http://www.jayeckles.com/research/grid.html (Date Accessed: 24-07-2003).

James MacKinnon, "Bootstrap inference in econometrics", *Canadian Journal of Economics*, vol. 35, pp. 615-645, 2002.

Khurshid Ahmad, Tugba Taskaya Temizel, David Cheng, Saif Ahmad, Lee Gillam, Pensiri Manomaisupat, Hayssam Traboulsi, and Matthew Casey, "Fundamental Data to SATISFI the Chartist", *The Technical Analyst Magazine*, pp. 36-38, 2004.

Matthew Friedman, "Grid Computing: accelerating the search for revenue and profit for Financial Markets", *Building an Edge*, pp. 143-145, 2003.

Nicholas T. Karonis, Brian R. Toonen, and Ian Foster, "MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface", Journal *of Parallel and Distributed Computing*, vol. 63, pp. 551-563, 2003.

Saif Ahmad, Tugba Taskaya Temizel, and Khurshid Ahmad, "Summarizing Time Series: Learning Patterns in 'Volatile' Series", Z.R. Yang, R. Everson, and H. Yin (Eds.), *Lecture Notes in Computer Science (Proc. of 5th International Conference on IDEAL)*, Exeter, 2004.

Tugba Taskaya, and Khurshid Ahmad, "Bimodal Visualisation: A Financial Trading Case Study", In *Proc. of the 7th International Conference on Information Visualization (IV'03)*, pp. 320, London, 2003.

V. Lavrenko, M. Schmill, D. Lawrie, P. Ogilvie, D. Jensen, and J. Allan, "Mining of Concurrent Text and Time Series", In *Proc. of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Boston, 2000.

---

[i] A financial instrument is defined as any tool that can be used in order to implement economic policy, hence having monetary value or recording a monetary transaction

[ii] http://www.datasynapse.com

[iii] www.investorwords.com

[iv] www.investorwords.com

[v] http://www.drkw.com/

[vi] http://www.researchandmarkets.com

[vii] http://www.spotter.com

[viii] http://www.held-mueller.de/JMatLink/