Decision Support Systems 28 (2000) 177–189
Decision Support Systems
www.elsevier.com/locate/dsw

The POPCORN market. Online markets for computational resources ☆

Ori Regev [a,*], Noam Nisan [a,b]

[a] Institute of Computer Science, Hebrew University, Jerusalem, Israel
[b] School of Computer Science, Interdisciplinary Center, Herzliya, Israel

Abstract

The POPCORN project provides an infrastructure for globally distributed computation over the whole Internet. It provides any programmer connected to the Internet with a single huge virtual parallel computer composed of all processors on the Internet, which care to participate at any given moment. POPCORN provides a market-based mechanism for trade in CPU time to motivate processors to provide their CPU cycles for other peoples' computations. ''Selling'' CPU time is as easy as visiting a certain web site with a Java-enabled browser. ''Buying'' CPU time is done by writing a parallel program using the POPCORN paradigm. A third entity in the POPCORN system is a ''market'' for CPU time, which is where buyers and sellers meet and trade. The POPCORN system may be visited on our web-site: http://www.cs.huji.ac.il/~popcorn.

Keywords: Global computation; Internet; Resource allocation; Markets; Java

1. Introduction

There are currently millions of processors connected to the Internet. At any given moment, many, if not most of them, are idle. An obvious and appealing idea is to utilize these idle processors for running applications that require large computational power. This would allow what may be termed ''global computing'' — a single computation carried out in cooperation between many processors worldwide.

Similar ideas in the context of local area networks are quite well known by now, especially due to the influence of the work on ''Network of Workstations'' [1]. There are several added complications, though, in the global case of cooperation over the whole Internet. First, there are major technical difficulties due to code mobility, security, platform heterogeneity, and coordination concerns. Then, there is a matter of scale as the Internet is much more ''distributed'': The communication bandwidth is smaller, the latency higher, the reliability lower. On the positive side, the potential number of processors is huge.

A much more fundamental difference is due to the distributed *ownership* of the processors on the Internet. Since each processor is owned and operated by a different person or organization, there is no a-priori motivation for cooperation (Why should my computer work on your problem?). Clearly, a motivation

☆ Expanded version of a talk presented at the First International Conference on Information and Computation Economies, Charleston, SC, 1998.

* Corresponding author.
E-mail address: regev@cs.huji.ac.il (O. Regev).

0167-9236/00/$ - see front matter © 2000 Elsevier Science B.V. All rights reserved.
PII: S0167-9236(99)00067-6

for cooperation, such as payments for CPU time, must be provided by a global computing system.

The POPCORN system provides an infrastructure for such ''global computation'' addressing all these difficulties. It has been implemented and was operable until recently for over a year through its web site over the Internet [38]. Descriptions of the system can be found in Refs. [6,21,32,37].

This paper focuses on POPCORN's approach to the last issue, that of motivating cooperation. The POPCORN system provides an online electronic market for trade-in CPU time. Buyers and sellers of CPU time connect to it via the Internet, and the market matches buyers and sellers according to economic criteria. It should be emphasized that the buyers and sellers are computer programs (acting for humans) rather than directly humans. It seems very likely that such totally automated electronic markets will play a large role in many forms of Internet cooperation (not just for CPU time), and that general mechanisms for such markets need to be developed and understood.

The design and implementation of such online electronic markets will of course draw on the vast literature available regarding real-world markets [2,11–13,35]. However, one should note that many differences exist. First, there are many technical issues of communication, implementation, etc. Second, the fact that the market is not intended for humans, but rather for programs, makes a difference. Third, in many cases, and in particular in the case of the POPCORN market, even the basic definitions of money, goods and trade need to be defined. We hope that our experiences with the POPCORN market may shed further light on several aspects of online electronic markets.

1.1. Paper structure

Section 2 provides an overview of the POPCORN system; further information can be found in Refs. [6,21,32,37]. Section 3 describes the outline of the economic notions and mechanisms that underline the POPCORN system. Section 4 provides preliminary analysis of the POPCORN markets. Section 5 describes our simulations of the POPCORN trade. In Section 6, we mention some related works and Section 7 outlines directions for further research.

2. An overview of the POPCORN system

The POPCORN system provides an infrastructure for global computation over the Internet. POPCORN's basic function is to provide any programmer on the Internet with a simple virtual parallel computer. This virtual machine is implemented by utilizing all processors on the Internet that care to participate at any given moment. The system is implemented in Java and relies on its ubiquitous ''applet'' mechanism for enabling wide-scale safe participation of remote processors.

There are three distinct types of entities in the POPCORN system.

(1) The parallel program written (in Java) using the POPCORN paradigm. This program acts as a CPU time ''buyer''. The program is written using the POPCORN programming paradigm that was designed to fit ''global computing''. This paradigm is described in Refs. [21,32].

(2) The CPU time ''seller'' who allows its CPU to be used by other parallel programs (instead of standing idle). This is done as easily as visiting a web site using a Java-enabled browser, and requires no download of code.

(3) The ''market'' which serves as a meeting place and matchmaker for buyers and sellers of CPU time.

The POPCORN programming paradigm, used by the buyer program, achieves parallelism by concurrently spawning off many sub-computations, termed ''computelets''. The POPCORN system automatically sends these computelets to a market (chosen by the user), which then forwards them to connected CPU-time sellers, who execute them and return the results. The matching of buyers and sellers in the market is dynamic, is done according to economic mechanisms, and results in a payment of the buyer to the seller.

The system is intended for coarse-grained parallelism. The computational efficiency is mostly determined by the ratio between the computation time of computelets to the communication effort needed to send them and handle the overhead. To achieve high efficiency, computelets should be relatively heavy in terms of computation time. Currently, seconds of CPU time per computelet are a minimum, and tens of seconds seem more typical. For very large-scale