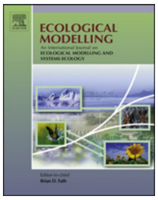
Ecological Modelling 229 (2012) 108–118

Contents lists available at SciVerse ScienceDirect

Ecological Modelling

journal homepage: www.elsevier.com/locate/ecolmodel



Reprint of: Parallel agent-based modeling of spatial opinion diffusion accelerated using graphics processing units☆

Wenwu Tang[a],*, David A. Bennett[b]

[a] Center for Applied Geographic Information Science and Department of Geography and Earth Sciences, University of North Carolina, 9201 University City Blvd., Charlotte, NC 28223, United States
[b] Department of Geography, University of Iowa, Iowa City, IA 52242, United States

ARTICLE INFO

Article history:

Keywords:
Agent-based models
Spatial opinion exchange
Parallel computing
Graphics processing units
Cyberinfrastructure

ABSTRACT

In this article, we describe a parallel agent-based model of spatial opinion diffusion that is driven by graphics processing units (GPUs). Modeling opinion exchange and diffusion across landscapes often involves the simulation of large numbers of geographically located individual decision-makers and a massive number of individual-level interactions. This simulation requires substantial computational power. GPU-enabled computing resources provide a massively parallel processing platform based on a fine-grained shared memory paradigm. This massively parallel processing platform holds considerable promise for meeting the computing requirement of agent-based models of spatial problems. In this article, we focus on the parallelization of an agent-based spatial opinion model using GPU technologies. We discussed key algorithms designed for parallel agent-based opinion modeling: including domain decomposition and mutual exclusion. Experiments conducted to examine computing performance show that GPUs provide a computationally efficient alternative to traditional parallel computing architectures and substantially accelerate agent-based models of large-scale opinion exchange among individual decision makers.

1. Introduction

The purpose of this article is to investigate the utility of graphics processing units (GPUs; see Owens et al., 2008; Pharr and Fernando, 2005) as a high-performance computing architecture for agent-based models (ABMs) of spatial diffusion of human decision-making processes. As motivation for this work we consider spatial opinion exchange as a representative form of spatial diffusion. Research into the spatial diffusion of entities, ideas, or information across a landscape has a long and active history in biology or ecology (Grimm and Railsback, 2005), geography (Cliff et al., 1981; Gould, 1969; Hägerstrand, 1966, 1967) and geographic information science (Goodchild, 1992; Miller, 2003; Yuan and Stewart, 2008), and social science (Epstein and Axtell, 1996; Gilbert and Troizsch, 2005; Gimblett, 2002). The use of ABMs as a computational approach to the study of such processes, however, is a relatively recent development (An et al., 2005; Berger, 2001; Brown et al., 2005; Parker et al., 2003; Sengupta and Sieber, 2007). The representational capability of ABMs allows us to explore complex nonlinear interactions between macro-level emergent patterns and their underlying driving processes in a bottom-up manner. Often, models of spatial diffusion (e.g., disease spread, agricultural innovation, and species dispersal) simulate the spatiotemporal behavior of a large number of geospatial entities that, in turn, produce massive amounts of agent–agent and agent–environment interactions across multiple spatial and temporal scales. When using ABMs to simulate spatial diffusion phenomena computational feasibility is often, therefore, a significant concern that must be considered early on in the model development process. Driven by advances in cyberinfrastructure (Atkins et al., 2003; NSF, 2007), high-performance computing resources (e.g., large computer clusters or GPUs) and related parallel computing approaches are increasingly being used to tackle the computational demands of ABMs (Armstrong, 2000; Haefner, 1992; Wang et al., 2006).

1.1. Parallel computing for agent-based modeling

Parallel computing uses the concept of divide and conquer to harness high-performance computing resources that possess multiple processing elements (Wilkinson and Allen, 2004). A problem of interest (for example, an ABM) that is computationally intensive, or even intractable given a single processing element, is partitioned (domain decomposition) into sub-problems that multiple processing elements solve in a concurrent manner. The size

---

DOI of original article: 10.1016/j.ecolmodel.2011.08.016.

☆ This article is a reprint of a previously published article. For citation purposes, please use the original publication details; Ecological Modelling, 222/19 (2011), pp. 3605–3615.

* Corresponding author.
E-mail address: WenwuTang@uncc.edu (W. Tang).

0304-3800/$ – see front matter © 2012 Published by Elsevier B.V.
doi:10.1016/j.ecolmodel.2012.02.003

of a sub-problem, directly associated with computational load, is dependent on the computing capacity (e.g., computer memory) and capability (e.g., clock rate) of the processing elements available. Processing elements with low (high) computing capacity and capability require sub-problems that are sufficiently small (large) if the underlying computing resources are to be leveraged efficiently. Furthermore, sub-problems will share data to varying degrees depending on the nature of the problem and the domain decomposition strategy used. Message-passing and share-memory techniques must, therefore, often be developed as part of a typical parallel programming implementation (Wilkinson and Allen, 2004). A suite of standard interfaces, exemplified by MPI (Message Passing Interfaces; see MPI, 2011; Snir et al., 1998), PVM (Parallel Virtual Machine; see PVM, 2011), and OpenMP (Open Multi-Processing; see OpenMP, 2011), have been defined to support parallel programming, and the associated libraries have been developed for many platforms.

The development and use of ABMs have benefited from parallel computing. Researchers have developed parallel ABMs to support the simulation of fish movement (Wang et al., 2006), deer/elk foraging (Abbott et al., 1997; Uziel and Berry, 1996), vehicle flow (Nagel and Rickert, 2001), and financial market (Deissenberg et al., 2008). In these ABMs, agents and their environments are organized and decomposed into a set of sub-models that are handled by underlying parallel computing resources. These parallel ABMs are often built on coarse-grained parallelism that uses computing resources based on central processing units (CPUs). Both message-passing and shared memory approaches (Wilkinson and Allen, 2004) have been employed to enable agent–agent and agent–environment interactions across multiple processors (Massaioli et al., 2005; Nagel and Rickert, 2001; Tang et al., 2011; Tang and Wang, 2009; Wang et al., 2006). While CPU-based high-performance computing resources allow for the development of parallel ABMs, learning how to use the associated parallel programming interfaces is often challenging and the cost of accessing and using these resources is high. These issues have prompted researchers to seek alternative high-performance computing resources and approaches for the development of ABMs. This trend is reflected in the increased use of general-purpose GPUs.

1.2. *General-purpose graphics processing units*

GPUs are many-core microprocessors that are used extensively to accelerate computer graphic operations (Pharr and Fernando, 2005). The ability of GPUs to support computationally intensive problem-solving was recognized as the computer hardware technology advanced and programming interfaces for general-purpose computation became widely available (Owens et al., 2008, 2007; Pharr and Fernando, 2005). The utility of GPUs for parallel computing is derived from the concept of stream processing in which a problem is partitioned into many problem elements that can be computed by a set of kernel functions (Buck et al., 2004; Owens et al., 2007)—i.e., in a SPMD (Single Program Multiple Data) manner (see Wilkinson and Allen, 2004). Kernel functions, in which algorithms for the problem of interest are implemented, are executed by micro processors on GPUs in a pipelined way. The nature of GPU micro processors limits the amount of data that can be computed in each processor. However, for some problems, this limitation can be overcome through the effective use of the large number of inexpensive micro processors that are typically incorporated into a single GPU (i.e., massively parallel; currently up to a hundred level). Many research domains, including chemistry, economics, medical science, and physics, have recognized the potential of programmable GPUs for data- and compute-intensive analysis and modeling (Owens et al., 2007; Pharr and Fernando, 2005). As a result, GPUs have been increasingly applied in these domains to accelerate the general-purpose data-parallel computation involved in, for example, data mining, image processing, numerical modeling, and physically based simulation (Kirk and Hwu, 2010; Nguyen, 2007; Owens et al., 2007; Pharr and Fernando, 2005). This trend will continue as GPU programming standards and languages that support heterogeneous computing (Brodtkorb et al., 2010) continue to evolve, of which CUDA (Compute Unified Device Architecture; see CUDA, 2011; Nickolls et al., 2008; Ryoo et al., 2008) and OpenCL (Open Computing Language; see OpenCL, 2011) are examples. Please refer to Brodtkorb et al. (2010) for detailed comparison of programming languages for GPUs and other heterogeneous computing platforms.

Researchers have begun to spend their efforts on investigating the capability of general-purpose GPUs in parallel agent-based modeling. Lysenko and D'Souza (2008) presented a GPU-based computational framework for agent-based modeling. Lysenko and D'Souza implemented and tested two ABM templates (Sugarscape and StupidModel; see Epstein and Axtell, 1996; Railsback et al., 2005) within their GPU framework and illustrated that GPUs are suitable for agent-based modeling. Richmond and Romano (2008) developed a GPU-enabled agent-based model for the simulation of pedestrian movement. Afterward, Richmond et al. (2009) proposed a set of specifications that allow for extending existing CPU-based parallel modeling frameworks for ABMs to the GPU environment. Erra et al. (2009) examined the use of GPUs to accelerate an agent-based model of fish movement behavior. Likewise, Li et al. (2009) described a GPU-enabled agent-based fish model to study schooling behavior. These GPU-enabled ABM studies suggest the unique power of GPUs for computationally intensive simulation and, more importantly, acknowledge that the allocation of computation for agent-based interactions to the many-core and shared-memory architecture of GPUs and the associated coordination of computational resources are challenging problems. Thus, further investigations into how massively parallel GPU resources can be efficaciously leveraged to facilitate computationally demanding agent-based modeling are needed.

1.3. *Rationale*

In this paper we focus on an ABM of spatial opinion exchange that leverages the power of parallel computing resources and technologies from GPUs. Opinion exchange among geographically located decision-makers (e.g., about land-use policies, or agricultural innovation) represents a complex dynamic process (Bennett et al., 2011). Individual decision-makers are heterogeneous in terms of their preferences, ability to interact with others, and willingness to modify their own opinion in response to interaction with other agents. This heterogeneity and inter-agent interactions may lead to emergent consensus patterns at the macro level. While ABMs support the representation of these individual-level interactions, their use for the simulation of large-scale spatiotemporal opinion exchange requires considerable support from computational technologies. Scale in this study is specifically related to the spatial, temporal, and decision-making domains. For example, ABMs with fine spatiotemporal resolutions and large spatiotemporal extents, and large number of interacting agents are regarded as large-scale models.

Tang and Bennett (2009) conducted a pilot study on the use of GPU-enabled high-performance computing resources for the agent-based simulation of spatial opinion exchange. Tang et al. (2011) applied message-passing technologies to construct CPU-enabled parallel agent-based opinion modeling within a supercomputing environment. To extend their studies on parallel ABMs, in this article we further investigate the utility of GPUs in enabling large-scale agent-based opinion modeling. The specific objectives of this paper are to: (1) identify and implement