

Chapter 12

Discovery of Emergent Sorting Behavior using Swarm Intelligence and Grid-Enabled Genetic Algorithms

Dimitris Kalles

Hellenic Open University, Greece

Alexis Kaporis

University of the Aegean, Greece

Vassiliki Mperoukli

Hellenic Open University, Greece

Anthony Chatzinouskas

Hellenic Open University, Greece

ABSTRACT

The authors in this chapter use simple local comparison and swap operators and demonstrate that their repeated application ends up in sorted sequences across a range of variants, most of which are also genetically evolved. They experimentally validate a square run-time behavior for emergent sorting, suggesting that not knowing in advance which direction to sort and allowing such direction to emerge imposes a $n/\log n$ penalty over conventional techniques. The authors validate the emergent sorting algorithms via genetically searching for the most favorable parameter configuration using a grid infrastructure.

INTRODUCTION

Sorting has been one of the first areas of computer science to showcase efficient algorithms to stand the test of time. While algorithms like quick-sort, merge-sort and heap-sort (Cormen, et al., 2009) are good at sorting with relatively few constraining assumptions, research focus gradually shifted to designing algorithms that can harness special cases in the input to accommodate some practical settings. Such special cases refer, for example,

to input that is partially ordered (Manilla, 1985; Estivill-Castro and Wood, 1992), or drawn from a constrained distribution, as is the case in bucket-sort (Cormen, et al., 2009) and spread-sort (Ross, 2002). Moreover, researchers have also studied extremely inefficient sorting algorithms to see whether inefficiencies can be mitigated by very simple changes in these algorithms (Gruber, et al., 2007).

Our motivation to examine sorting is to investigate what is the minimum structure of local

DOI: 10.4018/978-1-4666-6078-6.ch012

operators whose repetitive application ends up in sorted sequences. In our work, we further relax the assumption that we know which way to sort, yet arrive at a sorted sequence by simple local interactions. These are the basic elements of Emerge-Sort (Kalles, et al., 2012), our approach to self-organizing sorting without knowledge of sorting direction, which we experimentally validate across a range of variants, some of which are genetically evolved. We observe an $\Omega(n^2)$ run-time behavior and point out that Emerge-Sort is immune to arbitrary sequence modifications.

A simple example may help to visualize such a context. Consider a database which is distributed across several sites, with each site maintaining a sorting that best suits its local needs (either largest-first or smallest-first). If there is a need to redistribute objects across the sites, according to their order, so as to avoid asking all sites where an object might be located, one can either opt for some sort of centralized control over who-comes-first or, simply, let each object re-arrange itself in the local neighborhood. Additionally, when new objects are inserted dynamically into any of the available sites, one can either explicitly direct each one of them to the correct database partition or, simply, wait until the object finds its way to a partition. Similar problems have been long studied in the distributed computing literature where, up to now, solvability guarantees were accompanied by complex structured multiple passes, sometimes to deal with adversarial behavior in some of these passes (Flocchini, et al., 2004; Prasath, 2010; Israeli & Jalfon, 1993); they have also attracted the attention of the multi-agent systems community (Casadei, et al., 2006; Casadei, et al., 2007) and the consensus reaching community (Bénézit, et al., 2011).

The prominence of locality in the swarm intelligence approach to problem-solving has long been highlighted since many traditional problems (graph partitioning and clustering, among others, as described by Bonabeau, et al. [1998], Bonabeau [1997], Franks and Sendova-Franks [1992] and

Kuntz, et al. [1997]) have been recast in terms of swarm behavior. Swarm intelligence is at the junction of randomized behavior and local operations and has been demonstrated to solve difficult problems such as task scheduling (Bonabeau, et al., 1998). Now, emergent sorting is a behavior that has been documented in insect societies and modeled via swarm intelligence principles (Bonabeau, et al., 1999; Casadei, et al., 2007), albeit in an unconventional way; therein, sorting takes place in 2-D and consists of forming concentric circles where items of similar size are at roughly the same distance from the centre. 2-D sorting has received relatively scant attention since it is usually seen as a (difficult) case of clustering, another classic problem that has been also recently addressed with swarm intelligence (Handl & Meyer, 2008). It is interesting to note that swarm intelligence, beyond emergent sorting, has been also related to autonomic computing research, mainly through the observation of autonomic computing principles in biology inspired systems and the transfer of relevant concepts to problems in distributed computing (Babaoglu, et al., 2006).

Seen from a more conventional computer science perspective, swarm intelligence draws on several aspects of distributed computing (Dijkstra, 1974). It is not surprising that distributed sorting algorithms have been designed and progressively refined to address related problems of increasing relaxation of assumptions. We thus witness Loui (1984) first framing the problem of distributed sorting across n nodes linked in a chain, which, however, he based on an explicit three-stage process; first, elect a leader, then, using a carefully crafted encoding of the numbers to be sorted, insert values in the right place and let all nodes know about that encoding, and, finally, selectively delete values from all nodes, so that each node is left with the proper value of the sorted order. Based on that idea, Gerstel and Zaks (1997) extended the proofs of various properties of distributed sorting algorithms to tree-structures of connected processors. Flocchini, et al. (2004)

21 more pages are available in the full version of this document, which may be purchased using the "Add to Cart" button on the publisher's webpage:

www.igi-global.com/chapter/discovery-of-emergent-sorting-behavior-using-swarm-intelligence-and-grid-enabled-genetic-algorithms/110463

Related Content

Determination of Optimal Clusters Using a Genetic Algorithm

Tushar, Shibendu Shekhar Roy and Dilip Kumar Pratihari (2008). *Data Mining and Knowledge Discovery Technologies* (pp. 98-117).

www.irma-international.org/chapter/determination-optimal-clusters-using-genetic/7515

Impact of Balancing Techniques for Imbalanced Class Distribution on Twitter Data for Emotion Analysis: A Case Study

Shivani Vasantbhai Vora, Rupa G. Mehta and Shreyas Kishorkumar Patel (2022). *Research Anthology on Implementing Sentiment Analysis Across Multiple Disciplines* (pp. 412-432).

www.irma-international.org/chapter/impact-of-balancing-techniques-for-imbalanced-class-distribution-on-twitter-data-for-emotion-analysis/308500

Automatic Human Emotion Classification in Web Document Using Fuzzy Inference System (FIS): Human Emotion Classification

P Mohamed Shakeel and S Baskar (2022). *Research Anthology on Implementing Sentiment Analysis Across Multiple Disciplines* (pp. 1064-1075).

www.irma-international.org/chapter/automatic-human-emotion-classification-in-web-document-using-fuzzy-inference-system-fis/308533

Classification of Imbalanced Data with Random sets and Mean-Variance Filtering

Vladimir Nikulin (2008). *International Journal of Data Warehousing and Mining* (pp. 63-78).

www.irma-international.org/article/classification-imbalanced-data-random-sets/1808

The Role of Knowledge Management (KM) in Aged Care Informatics: Crafting the Knowledge Organization

Margee Hume, Craig Hume, Paul Johnston, Jeffrey Soar and Jon Whitty (2016). *Big Data: Concepts, Methodologies, Tools, and Applications* (pp. 1668-1686).

www.irma-international.org/chapter/the-role-of-knowledge-management-km-in-aged-care-informatics/150235