Sviluppo di modelli e tecniche di Intelligenza Artificiale per la mobilità sostenibile (Final Report)

Filippo Bistaffa Alessandro Farinelli

1 Introduction

This project aims at tackling the theoretical and practical challenges posed by new forms of sustainable transport, which are quickly emerging as a new form of commuting (especially in urban scenarios) thanks to real-world ridesharing services (e.g., Uber and Lyft).[1] Specifically, this domain of study involves problems that require the formation of *collectives* of commuters, which, in practice, are associated to cars with several passengers who travel sharing the same ride. The formation of such collectives is often subject to some feasibility constraints represented as graphs, which model the social aspects inherent in ridesharing scenarios.

In the context of Artificial Intelligence (AI), this problem is usually referred as *Graph-Constrained Coalition Formation* (GCCF), a technique that represents one of the key approaches for coordination in Multi-Agent Systems (MAS) [5]. More in general, GCCF can be seen as a particular type of Constraint Optimisation (CO) [6], since it requires to maximise some objective function (or, in the case of ridesharing, to minimise the overall transport cost and the pollutant emissions), while satisfying the above mentioned feasibility constraints. CO represents a fundamental technique that allows to address a wide variety of optimisation problems in several contexts. CO techniques have been successfully employed in MAS to face a number of multi-agent coordination challenges.

On the other hand, it is well known that collective formation problems pose very computationally-demanding tasks [5], as a consequence of the combinatorial nature of the interactions that the entities in the system can undertake. Henceforth, one of the main objects of this project is to study and

[1] `http://www.cityam.com/242760/uber-reveals-london-ride-sharing-figures-for-uberpool`.

implement solution techniques that can deal with such computational requirements, since we aim at developing solutions that feature the level of scalability required by actual-world applications.

In recent years, highly-parallel architectures have been successfully applied in many different scenarios (e.g., AI [1], bioinformatics [7], and deep learning [8]) as a tool to achieve tremendous performance improvements [9]. For instance, as *Nature* recently noted, early progress in deep learning was "made possible by the advent of fast GPUs that were convenient to program and allowed researchers to train networks 10 or 20 times faster." These overwhelming advancements encourage the investigation of parallelisation also in other computationally demanding fields like CF, with the objective of achieving the same benefits.

2 Results achieved during the project

One of the main results achieved during this project is CUBE [1, 2], a GPU version of the Bucket Elimination (BE) algorithm, which represents one of the most important CO solution techniques. CUBE has been presented in two independent publications, one paper in a very prestigious international journal (*IEEE Transactions on Cybernetics*) and one paper in the top European AI conference (*European Conference on Artificial Intelligence*):

[1] Filippo Bistaffa, Nicola Bombieri, and Alessandro Farinelli. "An Efficient Approach for Accelerating Bucket Elimination on GPUs." In *IEEE Transactions on Cybernetics*, in press, 2016.

[2] Filippo Bistaffa, Nicola Bombieri, and Alessandro Farinelli. "CUBE: A CUDA approach for Bucket Elimination on GPUs." In *European Conference on Artificial Intelligence*, ECAI, pages 125–132, 2016.

CUBE employs a novel parallelisation methodology, which is specifically designed to consider two fundamental aspects of the GPU algorithmic design, i.e., thread independence and memory management. In the design of CUBE, our main objective was to devise a solution that fulfils three key requirements. First, since BE is a general algorithm that can be applied to several problems, our framework should be likewise general to allow a wide adoption among different domains. Second, our approach should be able to achieve a high computational throughput, by means of optimised memory accesses to avoid bandwidth bottlenecks, a careful load-balancing to fully exploit the available computational power, and the adoption of well-known parallel primitives [10,

11] to reduce the CPU workload to the minimum. Third, our solution should not be limited by the amount of GPU memory.

CUBE achieves the objectives set above by means of a novel preprocessing algorithm that rearrange the input data, so to achieve optimised memory accesses. Such an arrangement enables pipelined data transfers, hence optimising the transfer time, and it allows the use of highly efficient routines for the fundamental parts of BE. CUBE is not limited by the amount of GPU memory, since our data layout allows us to process large tables by splitting them into manageable chunks that meet the memory capabilities of the GPU. Now, the capability of each thread to efficiently access its input data is a crucial aspect in the design of GPU algorithms [12], as it directly determines the final computational throughput. Within CUBE, we avoid unnecessary, expensive memory accesses by proposing a technique that allows threads to locate their input data only on the base of their own ID. We take advantage of the *data reuse* pattern inherent in the composition and the marginalisation operations by caching the input data in the *shared memory*, i.e., the fastest form of memory in the GPU hierarchy [12]. According to the project guidelines, CUBE has been tested on a realistic dataset [13]. Our results show that CUBE achieves a speed-up up to 696× *wrt* the CPU version of BE, and it is up to two orders of magnitude faster than recent GPU implementations.

In addition, during the project we published two papers in two very prestigious international AI journal.

On the one hand, a comprehensive version of our previous work [14] has been published in *ACM Transactions on Intelligent Systems and Technology*:

[3] Filippo Bistaffa, Alessandro Farinelli, Jesús Cerquides, Juan A. Rodríguez-Aguilar, and Sarvapali D. Ramchurn. "Algorithms for Graph-Constrained Coalition Formation in the Real World." In *ACM Transactions on Intelligent Systems and Technology*, volume 8, issue 4, 2017.

This paper presents CFSS, a branch-and-bound solution algorithm for GCCF, which has been thoroughly evaluated by means of realistic datasets, according to the project guidelines. CFSS also represents our main solution technique in the context of ridesharing [14].

On the other hand, a comprehensive version of our previous conference papers focusing on ridesharing [15, 16] has been published in *Artificial Intelligence*, one of the most prestigious AI journals:

[4] Filippo Bistaffa, Alessandro Farinelli, Georgios Chalkiadakis, and Sarvapali D. Ramchurn. "A Cooperative Game-Theoretic Approach to the Social Ridesharing Problem." In *Artificial Intelligence*, volume 246, pages 86–117.

Such a work presents a CGT approach that addresses two fundamental aspects of the ridesharing problem. First, we focus on the optimisation problem of forming the travellers' coalitions that minimise the travel cost of the overall system. To this end, we model the formation problem as GCCF. Our approach allows users to specify both *spatial* and *temporal* preferences for the trips. Second, we tackle the *payment allocation* aspect of SR, by proposing the first approach that computes fair (i.e., *kernel-stable*) payments for systems with thousands of agents. We conduct a systematic empirical evaluation that uses real-world datasets (i.e., GeoLife and Twitter). We are able to compute optimal solutions for medium-sized systems (i.e., with 100 agents), and high quality solutions for very large systems (i.e., up to 2000 agents). Our results show that our approach improves the social welfare (i.e., reduces travel costs) by up to 36.22% with respect to the scenario with no ridesharing. Finally, our payment allocation method computes kernel-stable payments for 2000 agents in less than an hour—while the state-of-the-art is able to compute payments only for up to 100 agents, and does so 84 times slower than our approach.

Finally, the lead researcher of the project, Filippo Bistaffa, has been recently awarded with a *Marie Skłodowska-Curie Actions* (MSCA) Individual Fellowship (one of Europe's most competitive and prestigious fellowships). Such fellowship amounts to a total of 158.121 €, which will fund the "Collectiveware: highly-parallel algorithms for collective intelligence" 2-year project, aiming to further develop the research work conducted during the current project, especially in the context of ridesharing.

3 Current and future work

Our current research work focuses on the development of an *online ridesharing* framework, on the basis of our previous work [4, 15, 16]. Specifically, we consider a dynamic ridesharing scenario, in which agents can enter and leave the system at any time, with a known probability distribution based on historical data. Such setting is modelled as a *Markov Decision Process* (MDP), a very well known mathematical framework for modelling decision making in scenarios with uncertainty (such as online ridesharing). The MDP literature offers a wealth of techniques that allow to compute an optimal *policy*, i.e., a function that, given any state of the system, returns the optimal action to make, so as to maximise the expected reward over a potentially infinite time horizon [17]. Unfortunately, the *curse of dimensionality* hinders the application of these techniques in domains characterised by a high-dimensional state space (such as online ridesharing).

Against this background, we are currently investigating the application of *Monte-Carlo Tree Search* (MCTS), an algorithm usually employed to compute policies when the size of the search space is very large. Our choice is further motivated by the successful use of MCTS in conjunction with deep neural networks in other online decision making problems [18].

Furthermore, we aim at extending and applying the expertise acquired during the design of CUBE to accelerate other CO techniques, such as AND/OR search-based approaches [19]. In particular, we are exploring two promising research directions. One the one hand, we are implementing a GPU hybrid DFS/BFS approach that adopts an heuristic function based on BE to identify and prune less-promising subtrees, so to maintain only the portions of the search-space that can improve upon the current best solution. Notice that a hybrid DFS/BFS approach is used since a pure-DFS one would not be suitable for parallelisation, since, in general, DFS is known to be hard to parallelise [20]. One the other hand, we are also investigating a more complex approach, i.e., a GPU algorithm based on best-first search [21], which currently represents the state-of-the-art sequential approach for CO. Such an approach can potentially achieve very good performance results, but it also requires a deeper study due to the inherent complex nature of the underlying search algorithm and the relatively small number of works that focused on best-first search on GPUs.

In another research work, we aim at tackling the computational complexity inherent in the GCCF problem by approximating it with a more succinct representation, namely *Induced Subgraph Games* (ISGs) [22]. ISGs constitute an interesting and widely studied model in the Cooperative Game Theory (CGT) literature, since they have interesting computational properties due to the inherent simple structure of the model. As an example, solution algorithms for GCCF are generally very efficient on ISGs [14, 23]. Unfortunately, this simplicity limits the applicability of ISGs, as it is well-known that ISGs are not capable to perfectly represent every GCCF problem. In fact, to the best of our knowledge, no actual-world applications of ISGs exist in the literature, as real-world scenarios usually exhibit a structure that is too complex for ISGs (e.g., ridesharing cannot be perfectly represented as an ISG [15]). No research to date has examined the extremely important problem of how to *approximate* a generic GCCF problem as a ISG, so to minimise the error in such an approximation. Against this background, we are currently working on APEQIS (APproximately EQuivalent IS-represented cooperative games), an approach able to achieve such an objective.

Our preliminary results are very promising. APEQIS can compute ISG approximations of GCCF problems with thousands of agents, showing that this is a viable approach for actual-world scenarios. Moreover, we imple-

mented a GPU version of APEQIS that, in our experiments, is 135× faster than the sequential counterpart employing CPLEX (i.e., the state-of-the-art optimisation solver). The computed ISG can be solved very quickly *wrt* the original GCCF problem (i.e., orders of magnitude faster). In addition, our experiments show that the quality of the optimal solution in the new, approximated ISG is comparable *wrt* the optimal solution in the original problem. At the moment, we are currently working on a technique to provide theoretical bounds on the quality of such an approximation.

References

[1] F. Bistaffa, N. Bombieri, and A. Farinelli. "An Efficient Approach for Accelerating Bucket Elimination on GPUs". In: *IEEE Transactions on Cybernetics* (2016).

[2] F. Bistaffa, N. Bombieri, and A. Farinelli. "CUBE: A CUDA Approach for Bucket Elimination on GPUs". In: *European Conference on Artificial Intelligence*. 2016, pp. 125–132.

[3] F. Bistaffa, A. Farinelli, J. Cerquides, J. Rodríguez-Aguilar, and S. D. Ramchurn. "Algorithms for Graph-Constrained Coalition Formation in the Real World". In: *ACM Transactions on Intelligent Systems and Technology* 8.4 (2017).

[4] F. Bistaffa, A. Farinelli, G. Chalkiadakis, and S. D. Ramchurn. "A Cooperative Game-Theoretic Approach to the Social Ridesharing Problem". In: *Artificial Intelligence* 246 (2017), pp. 86–117.

[5] J. Cerquides, A. Farinelli, P. Meseguer, and S. D. Ramchurn. "A tutorial on optimization for multi-agent systems". In: *The Computer Journal* (2013).

[6] R. Dechter. *Constraint processing*. Morgan Kaufmann, 2003.

[7] M. C. Schatz, C. Trapnell, A. L. Delcher, and A. Varshney. "High-throughput sequence alignment using Graphics Processing Units". In: *BMC bioinformatics* 8.1 (2007).

[8] Y. LeCun, Y. Bengio, and G. Hinton. "Deep learning". In: *Nature* 521.7553 (2015), pp. 436–444.

[9] S. Greengard. "GPUs Reshape Computing". In: *Communications of the ACM* 59.9 (2016).

[10] N. Satish, M. Harris, and M. Garland. "Designing efficient sorting algorithms for manycore GPUs". In: *IEEE International Symposium on Parallel Distributed Processing*. 2009, pp. 1–10.

[11] S. Sengupta, M. Harris, Y. Zhang, and J. D. Owens. "Scan primitives for GPU computing". In: *Graphics hardware*. 2007, pp. 97–106.

[12] R. Farber. *CUDA Application Design and Development*. Elsevier, 2012.

[13] E. Bensana, M. Lemaitre, and G. Verfaillie. "Earth observation satellite management". In: *Constraints* 4.3 (1999), pp. 293–299.

[14] F. Bistaffa, A. Farinelli, J. Cerquides, J. Rodríguez-Aguilar, and S. D. Ramchurn. "Anytime Coalition Structure Generation on Synergy Graphs". In: *International Conference on Autonomous Agents and Multi-Agent Systems*. 2014, pp. 13–20.

[15] F. Bistaffa, A. Farinelli, and S. D. Ramchurn. "Sharing Rides with Friends: a Coalition Formation Algorithm for Ridesharing". In: *AAAI Conference on Artificial Intelligence*. 2015, pp. 608–614.

[16] F. Bistaffa, A. Farinelli, G. Chalkiadakis, and S. D. Ramchurn. "Recommending Fair Payments for Large-Scale Social Ridesharing". In: *ACM Conference on Recommender Systems*. 2015, pp. 139–146.

[17] R. A. Howard. *Dynamic Programming and Markov Processes*. MIT Press, 1960.

[18] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. "Mastering the game of Go with deep neural networks and tree search". In: *Nature* 529.7587 (2016), p. 484.

[19] R. Marinescu and R. Dechter. "AND/OR branch-and-bound search for combinatorial optimization in graphical models". In: *Artificial Intelligence* 173.16 (2009), pp. 1457–1491.

[20] J. H. Reif. "Depth-first search is inherently sequential". In: *Information Processing Letters* 20.5 (1985), pp. 229–234.

[21] R. Marinescu and R. Dechter. "Best-first AND/OR search for graphical models". In: *AAAI Conference on Artificial Intelligence*. 2007, pp. 1171–1176.

[22] X. Deng and C. H. Papadimitriou. "On the complexity of cooperative solution concepts". In: *Mathematics of Operations Research* 19.2 (1994), pp. 257–266.

[23] T. Voice, M. Polukarov, and N. R. Jennings. "Coalition structure generation over graphs". In: *Journal of Artificial Intelligence Research* 45 (2012), pp. 165–196.