A Massively Parallelized Machine Learning Algorithm for Dynamic Incentive Problems

Philipp Renner[††], Simon Scheidegger[**]
[†]School of Management, University of Lancaster, UK
[*]Department of Finance, HEC, University of Lausanne, Switzerland
[†]p.renner@lancaster.ac.uk
[*]simon.scheidegger@unil.ch

***Abstract*—We propose a massively parallelized and optimized machine learning framework to solve high-dimensional dynamic Incentive models on modern hybrid, GPU- and MIC-based clusters. The key innovation of our method is to first combine set-valued dynamic programming techniques with unsupervised machine learning to determine irregularly shaped feasible sets. Second, we generate training data from those pre-computed feasible sets to recursively solve the dynamic incentive problem by a supervised machine learning algorithm. Third, to speed up the time-to-solution process, we propose a generic hybrid parallelization scheme for dynamic incentive problems that allows efficient use of contemporary high-performance computing (HPC) hardware. This combination, all implemented in C++ and Fortran, enables us to analyze dynamic incentive models of a complexity that was previously considered to be intractable. To demonstrate the broad applicability of our computational framework, we study an insurance-like dynamic adverse selection problem with up to ten different, persistent types. Our numerical experiments on the Cray XC50 "Piz Daint" system at the Swiss National Supercomputer Centre (CSCS) show that our framework shows a speedup-up of at least three orders of magnitude. As next steps, we plan to apply this generic computational framework to topical models from the finance literature, calibrate them, and finally also estimate them.**

***Index Terms*—Dynamic Contracts, Principal-Agent Model, Dynamic Programming, Machine Learning, Gaussian Processes, High-Performance Computing, Heterogeneous Systems, CUDA, GPU, MIC**

I. Introduction

Dynamic incentive problems are of key importance in model-based economics and finance. They occur whenever two parties form a contract with asymmetric information, and encompass questions such as manager remuneration, insurance contracts, and optimal taxation (see, e.g., [1] and references therein for a thorough review). Solving these models is a formidable task, as standard recursive techniques (see, e.g., [2]) are often unworkable in these situations. To this end, previous research[1] extensively studied methods of recursively representing incentive-compatible contracts in order to make them formally tractable. Although the theory of contracting has played a very prominent role in economics and finance over the past few decades (see, e.g., [13], [14]), the quantitive properties of these models have largely been unexplored due to their computational complexity. Two obstacles arise in these models in general. First, unobservable continuous actions lead to incentive constraints that are themselves optimization problems. In the literature, they are commonly referred to as *moral hazard* problems. Second, in so-called *adverse selection* problems, unobserved, persistent discrete types make it necessary to look at a high-dimensional dynamic program with an unknown domain.

In the research proposed here, we want to study dynamic models of adverse selection—that is, where a principal interacts with an agent without knowing the agent's type. In each period, the agent receives an income shock determining his type. He then reports this shock to the principal who in turn picks her action dependent on this report. In finance, a prototypical setting of this kind could be a lender-borrower relationship. In such a setup, the principal lends money to an agent without knowing whether the agent reported his situation truthfully. If the shocks are correlated, then this problem becomes very difficult to solve.

Two major bottlenecks create substantial difficulties in solving such problems numerically. The first issue is the determination and approximation of multi-dimensional and (possibly) non-convex equilibrium correspondences—that is, feasible sets of a dynamic program. The second one is performing dynamic programming on them. Thus, we also have to repeatedly and efficiently approximate functions on irregularly shaped domains. To the best of our knowledge, there exists at present no solution framework in the dynamic incentive context that can cope with all these issues at once.

To remedy those shortfalls, we proposed recently in [15] to pre-compute the time-invariant feasible set for the continuous state variables by combining ideas from [16], [17] with Bayesian Gaussian mixture models —a method from unsupervised machine learning. Next, we solved the recursively formulated dynamic incentive problem on an irregularly shaped domain by applying a massively parallelized dynamic programming algorithm that uses Gaussian process regression—a method from supervised machine learning (see, e.g., [18])—to approximate high-dimensional value and policy functions on the entire feasible set. This combination enables us now to analyze dynamic incentive models that were previously considered to be intractable.

A defining feature of Gaussian process machine learning is that they combine the best of two worlds—namely, those

[1]For an incomplete list of research in discrete-time settings, see, e.g., [3]–[7], and for models in continuous time, see, e.g., [8]–[12].

of grid-free methods such as Monte Carlo and of function approximation theory. Gaussian processes learn a function based on the observations available at so-called design points, and do so without any geometric restriction. Gaussian processes, therefore, stand in stark contrast to ordinary, grid-based approximation schemes for high-dimensional state spaces such as Smolyak's method, adaptive sparse grids, high-dimensional model reduction, or projection methods. Those grid-based approximators are restricted to hyper-rectangular state spaces and thus are not a natural modeling choice in the context of solving dynamic adverse selection models.

From the computational point of view, two major bottlenecks create difficulties in achieving a fast time-to-solution process when solving large-scale dynamic adverse selection models with this iterative method, namely,

(i) in each iteration step, several economic functions need to be approximated and interpolated. For this purpose, the function values have to be determined at many points in the high-dimensional state space, and
(ii) each point involves solving relatively large optimization problems.

We overcome these difficulties by massively reducing the number of grid points required (in contrast to Cartesian grid-based methods) to represent the economic functions by using dimension-agnostic sampling techniques and Gaussian Process regression. Also, the time spent in each iteration step is substantially reduced by applying massively parallel processing. Using the Message Passing Interface (MPI) [19], we distribute the workload—that is, the sample points, across compute nodes. The nodes can optionally be equipped with NVIDIA GPUs. Within a single node, the workload is further partitioned among CPU cores with OpenMP [20]. Besides, the linear algebra packages we use deploy AVX, AVX2 or AVX-512 vectorization for Sandy/Ivy Bridge, Haswell/Broadwell or Skylake/KNL, respectively. This scheme enables us to make efficient use of contemporary HPC facilities that consist of a variety of special purpose as well as general purpose hardware and whose performance nowadays can reach dozens of petaflop/s (https://www.top500.org).

In preliminary numerical experiments on ordinary desktop computers, we are capbable of solving at least 10-dimensional test models with persistent shocks, and complex state-space geometries (as introduced in [4]), which is already a game-changer. Moreover, scaling tests on the Cray XC50 "Piz Daint" system at the Swiss National Supercomputer Centre (CSCS) show that our framework scales up more than three orders of magnitude, which will allow us to tackle really complex problems.

In the next steps, we plan on tackling models with shock structures that match more of the moments given in data than was possible previous research. Prior papers on this topic either treated a two-state problem or a continuum of states with a Brownian-Motion to keep the problem tractable. These are too rigid to be used to match empirical moments closely. We, on the other hand, will be able to push those limits with our generic, massively parallelized code framework. After that, we want to use our machinery to perform structural estimation for this model. We will do this by including essential parameters in our state variables, as proposed, e.g., in [21]. Because of the scalability of our code, we will still be able to solve such a problem even with an enlarged state space.

REFERENCES

[1] M. Golosov, A. Tsyvinski, and N. Werquin, "Recursive contracts and endogenously incomplete markets," *Handbook of Macroeconomics*, vol. 2, pp. 725–841, 2016.

[2] N. Stokey, R. Lucas, Jr., and E. Prescott, *Recursive Methods in Economic Dynamics*. Cambridge, MA: Harvard University Press, 1989.

[3] S. E. Spear and S. Srivastava, "On repeated moral hazard with discounting," *The Review of Economic Studies*, vol. 54, no. 4, pp. 599–617, 1987.

[4] A. Fernandes and C. Phelan, "A recursive formulation for repeated agency with history dependence," *Journal of Economic Theory*, vol. 91, no. 2, pp. 223 – 247, 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022053199926194

[5] M. Doepke and R. M. Townsend, "Dynamic mechanism design with hidden income and hidden actions," *Journal of Economic Theory*, vol. 126, no. 1, pp. 235 – 285, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0022053104001954

[6] A. Abraham and N. Pavoni, "Efficient allocations with moral hazard and hidden borrowing and lending: A recursive formulation," *Review of Economic Dynamics*, vol. 11, no. 4, pp. 781 – 803, 2008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1094202508000197

[7] N. Pavoni, C. Sleet, and M. Messner, "The dual approach to recursive optimization: Theory and examples," *Forthcoming, Econometrica*, 2017.

[8] P. M. DeMarzo and Y. Sannikov, "Optimal security design and dynamic capital structure in a continuous-time agency model," *The Journal of Finance*, vol. 61, no. 6, pp. 2681–2724, 2006. [Online]. Available: http://dx.doi.org/10.1111/j.1540-6261.2006.01002.x

[9] Y. Sannikov, "A continuous- time version of the principal: Agent problem," *The Review of Economic Studies*, vol. 75, no. 3, pp. 957–984, 2008. [Online]. Available: http://www.jstor.org/stable/20185061

[10] N. Williams, "On dynamic principal-agent problems in continuous time," 2009.

[11] ——, "Persistent private information," *Econometrica*, vol. 79, no. 4, pp. 1233–1275, 2011. [Online]. Available: http://dx.doi.org/10.3982/ECTA7706

[12] Z. He, B. Wei, J. Yu, and F. Gao, "Optimal long-term contracting with learning," *The Review of Financial Studies*, vol. 30, no. 6, pp. 2006–2065, 2017.

[13] J.-J. Laffont and J. Tirole, *A Theory of Incentives in Procurement and Regulation*, ser. MIT Press Books. The MIT Press, January 1993, vol. 1, no. 0262121743. [Online]. Available: https://ideas.repec.org/b/mtp/titles/0262121743.html

[14] J. Tirole, *The theory of corporate finance / Jean Tirole*. Princeton University Press Princeton, N.J, 2006.

[15] P. Renner and S. Scheidegger, *Machine learning for dynamic incentive problems*. Working paper– Available at SSRN: https://ssrn.com/abstract=3282487 or http://dx.doi.org/10.2139/ssrn.3282487, 2019.

[16] D. Abreu, D. Pearce, and E. Stacchetti, "Optimal cartel equilibria with imperfect monitoring," *Journal of Economic Theory*, vol. 39, no. 1, pp. 251 – 269, 1986. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022053186900281

[17] ——, "Toward a Theory of Discounted Repeated Games with Imperfect Monitoring," *Econometrica*, vol. 58, no. 5, pp. 1041–1063, September 1990. [Online]. Available: https://ideas.repec.org/a/ecm/emetrp/v58y1990i5p1041-63.html

[18] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning)*. The MIT Press, 2005.

[19] A. Skjellum, W. Gropp, and E. Lusk, *Using MPI*. MIT Press, 1999.

[20] G. Jost, B. Chapman, and R. van der Pas, *Using OpenMP - Portable Shared Memory Parallel Programming*. MIT Press, 2007.

[21] S. Scheidegger and I. Bilionis, "Machine learning for high-dimensional dynamic stochastic economies," *Forthcoming, Journal of Computational Science*, 2019.