2022-01-0089 Published 29 Mar 2022

Using GPU to Accelerate Backward Induction for Vehicle Speed Optimal Control

Zhaoyuan Ma Worcester Polytechnic Institute

Xiangrui Zeng Huazhong University of Science and Technology

Citation: Ma, Z. and Zeng, X., "Using GPU to Accelerate Backward Induction for Vehicle Speed Optimal Control," SAE Technical Paper 2022-01-0089, 2022, doi:10.4271/2022-01-0089.

Received: 24 Jan 2022 Revised: 24 Jan 2022 Accepted: 23 Jan 2022

Abstract

This paper proposes a method to adapt backward induction, which is used to solve the vehicle speed optimal control problem for energy efficiency, to a computer with a GPU to accelerate the computation. A common application of this type of problem is to control a vehicle on a given route with surrounding vehicles, road grades, traffic signals, stop signs, speed limits, and other conditions. Several indicators can be used to determine the performance of the controller, including the energy consumption of the trip, the driving speed smoothness, and the traveling time to a given destination. Solving this optimization problem globally by backward induction is time-consuming, due to the large searching space of the vehicle's distance, velocity, and acceleration. The proposed method converts the single thread implementation to a parallel process that runs on a consumer-level GPU. This is done by choosing the problem scale, separating independent sub-processes, and pruning the data to accommodate the GPU programming requirement. The method is tested on a simulated route with a leading vehicle, a traffic light, and speed limits. The historical behaviors of the leading vehicle are known, and they are used to predict its future behaviors in a stochastic way. Compared to the CPU-based backward induction, the proposed GPU-based version solves the given problem 15 to 30 times faster, depending on the preset granularities of variables.

Introduction

Vehicles consumed more than 50% of transportation petroleum usage across America in 2017 and 2018 [1]. According to the report done by U.S. Energy Information Administration, about 70% of energy consumption was contributed by vehicles in the transportation sector in 2020 [2]. Society is making effort to lower vehicles' energy consumption and greenhouse gas emission. Fuel efficiency can be achieved by different approaches, such as improving aerodynamics and reducing weight, integrating electric motor with the internal combustion engine, selecting gear wisely to keep the engine running near its' optimum operating condition. Research shows that in an urban driving scenario, more than half of the energy consumption happens at intersections, mainly caused by acceleration and idling [3]. Experienced drivers anticipate the upcoming traffic light state and the movement of the surrounding vehicles, to avoid accelerating to an unnecessary speed, which may require braking after that. Another example is to brake smoothly when a red light is far ahead. Chances exist that the vehicle could preserve some kinetic energy before reaching the intersection while the traffic light turns from red to green. Similarly, using vehicle speed optimal control can manipulate the vehicle's behavior along the route, avoid energy-intensive manners to reduce energy consumption. This control can also benefit the vehicle's traveling time to a given target [4]. Such a controller has the potential of improving the fuel economy performance of a cruise control system, especially under the urban driving scenario with leading vehicles and intersections.

To improve vehicles' energy economy with speed control, different attempts have been made to approach the global optimal or a sub-optimal solution to achieve a lower computation time. Nunzio et al. propose an eco-driving solution that prunes infeasible velocities to create a graph and plans with Dijkstra's algorithm [5]. Eckhoff et al. solve the problem with a rule-base controller [6]. Huang et al. use the sequential convex optimization method to minimize the fuel consumption and traveling time while meeting ride comfort (jerk limits) [7]. Luo et al. [8] and Miao et al.[9] develop controllers with genetic algorithms. PMP (Pontryagin's minimum principle) has been applied to process problems with traffic lights [10] or road grade information [11]. MPC (model predictive control) is a common approach for the speed optimizing problem. Asadi et al. use a rule-based MPC to achieve a near-optimal solution with a reasonable level of computation [4]. Li et al. design a cost function and relevant constraints for their MPC controller [12]. Held et al. incorporate Quadratic program (QP) into MPC [13].

FIGURE 4 Vehicles' trajectory examples. The horizontal dash line refers to the time when the traffic light turns red. The vertical dash line refers to the traffic light position.

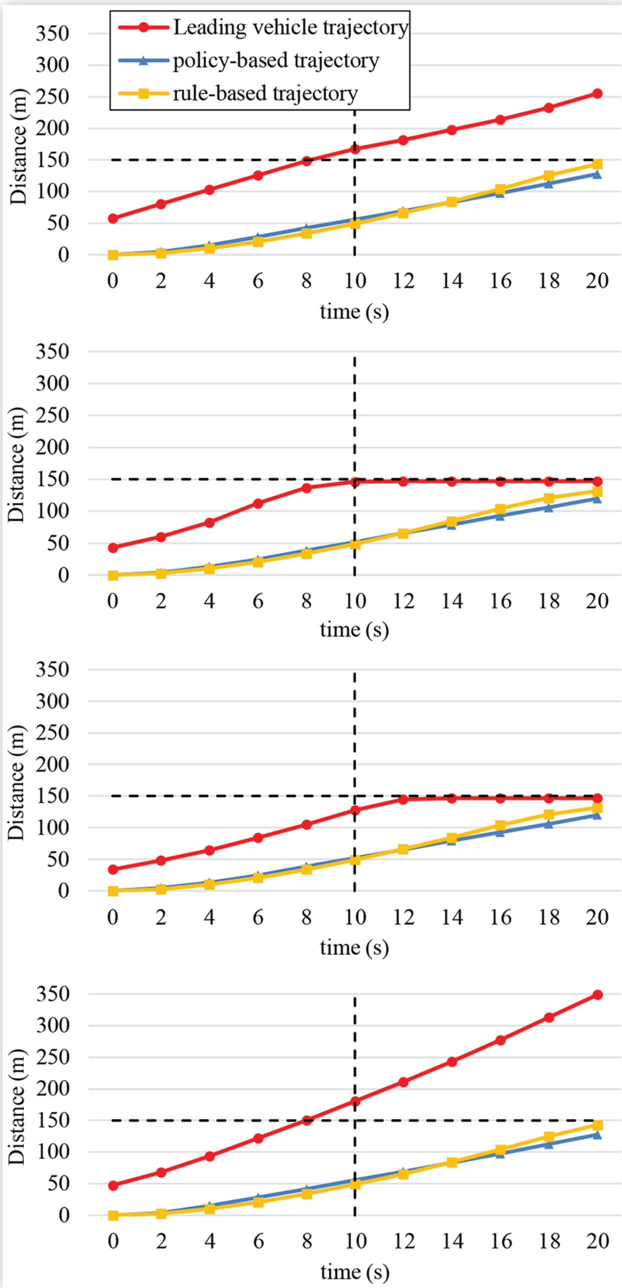

TABLE 2 Comparison for CPU and GPU implementation. Presenting the average time spent for backward induction. Inside the bracket is the standard deviation.

Granularity	Avg. CPU Time	Avg. GPU Time	Speedup
N_d = 121	10.11(±0.02)*s*	0.68(±0.01)*s*	15×
N_d = 241	72.80(±0.07)*s*	3.00(±0.01)*s*	24×
N_d = 361	248.15(±0.01)*s*	8.31(±0.01)*s*	30×

Because in a real-world scenario, this can be done before the trip and saved as offline data. Measuring the time starts at selecting and transferring necessary subsets from the above-mentioned data, and ends after being able to access the policy table in the computer's memory. As a result, using a GPU significantly reduces the processing time. When the number of states increases, the rate of speed up also increases.

TABLE 3 Functions' time cost. H2D stands for the host to the device. D2H stands for the device to the host.

Granularity	cudaMalloc	Memcpy, H2D	Memcpy, D2H	Kernel	Others
N_d = 121	24%	8%	4%	32%	32%
N_d = 241	5%	8%	2%	40%	45%
N_d = 361	2%	7%	2%	57%	32%

An analysis of GPU computation time cost is done by using NVVP (NVIDIA Visual Profiler). This tool shows the time consumption of cudaMalloc (allocate memory), cudaMemcpy (data transferring between the host/computer memory and the device/GPU memory), and the kernel executing time during the whole program executing cycle. The "other" is not listed by the Visual Profiler. It includes extracting the subset from prepared data, waiting for the kernel to start, etc. For different N_d, cudaMalloc (allocate memory) roughly takes the same amount of time (about 150*ms*). With the increase of N_d, the time consumed by the kernel plays a more important role. Table 3 shows the percentage of time that is spent doing each function.

Conclusion

This paper proposes a solution of parallelization of backward induction for a vehicle running on a given route with a leading vehicle and a traffic light. This parallel backward induction implementation is designed to use CUDA's features to optimize the GPU's performance. The experiment result shows that with certain GPU hardware support, parallel computing can be used to significantly improve the processing speed of backward induction to solve a finite-horizon optimal control problem. Depending on the problem scale, parallelization with a GPU speeds up the controller to different levels, 15 to 30 times faster, compared to processing in a single-thread manner on a CPU. Although with the current setup, the computational time is in general not short enough to achieve a real-time solution. It can still help to solve a large-scale problem offline more efficiently. Then the generated data can be used to support real-time control design. Future work will include continuous planning in a repeated manner, comparing different parallel strategies, and optimizing memory arrangement to further reduce time consumption.

References

1. Davis, S.C. and Boundy, R.G., "Transportation Energy Data Book," Edition 39, ORNL/TM-2020/1770, Oak Ridge National Lab. (ORNL), Oak Ridge, TN (United States), United States, 2021.
2. U.S. Energy Information Administration, *Annual Energy Outlook 2021* (United States: U.S. Energy Information Administration, 2021)

3. Wu, L., Ci, Y., Chu, J., and Zhang, H., "The Influence of Intersections on Fuel Consumption in Urban Arterial Road Traffic: A Single Vehicle Test in Harbin, China," *PLoS One* 10, no. 9 (2015): e0137477, doi:10.1371/journal.pone.0137477.
4. Asadi, B. and Vahidi, A., "Predictive Cruise Control: Utilizing Upcoming Traffic Signal Information for Improving Fuel Economy and Reducing Trip Time," *IEEE Trans. Contr. Syst. Technol.* 19, no. 3 (2011): 707-714, doi:10.1109/TCST.2010.2047860.
5. De Nunzio, G., Canudas de Wit, C., Moulin, P., and Di Domenico, D., "Eco-Driving in Urban Traffic Networks Using Traffic Signal Information," in *52nd IEEE Conference on Decision and Control*, 2013, 892-898, doi:10.1109/CDC.2013.6759995.
6. Eckhoff, D., Halmos, B., and German, R., "Potentials and Limitations of Green Light Optimal Speed Advisory Systems," in *2013, IEEE Vehicular Networking Conference*, 2013, 103-110, doi:10.1109/VNC.2013.6737596.
7. Huang, X. and Peng, H., "Speed Trajectory Planning at Signalized Intersections Using Sequential Convex Optimization," in *2017 American Control Conference (ACC)*, 2017, 2992-2997, doi:10.23919/ACC.2017.7963406.
8. Luo, Y., Li, S., Zhang, S., Qin, Z. et al., "Green Light Optimal Speed Advisory for Hybrid Electric Vehicles," *Mechanical Systems and Signal Processing* 87 (2017): 30-44, doi:10.1016/j.ymssp.2016.04.016.
9. Miao, C., Liu, H., Zhu, G.G., and Chen, H., "Connectivity-Based Optimization of Vehicle Route and Speed for Improved Fuel Economy," *Transportation Research Part C: Emerging Technologies* 91 (2018): 353-368, doi:10.1016/j.trc.2018.04.014.
10. Wan, N., Vahidi, A., and Luckow, A., "Optimal Speed Advisory for Connected Vehicles in Arterial Roads and the Impact on Mixed Traffic," *Transportation Research Part C: Emerging Technologies* 69 (2016): 548-563, doi:10.1016/j.trc.2016.01.011.
11. Hu, J., Shao, Y., Sun, Z., and Bared, J., "Integrated Vehicle and Powertrain Optimization for Passenger Vehicles with Vehicle-Infrastructure Communication," *Transportation Research Part C: Emerging Technologies* 79 (2017): 85-102, doi:10.1016/j.trc.2017.03.010.
12. Li, S.E., Li, K., and Wang, J., "Economy-Oriented Vehicle Adaptive Cruise Control with Coordinating Multiple Objectives Function," *Vehicle System Dynamics* 51, no. 1 (2013): 1-17, doi:10.1080/00423114.2012.708421.
13. Held, M., Flärdh, O., and Mårtensson, J., "Optimal Speed Control of a Heavy-Duty Vehicle in the Presence of Traffic Lights," in *2018 IEEE Conference on Decision and Control (CDC)*, 2018, 6119-6124, doi:10.1109/CDC.2018.8619463.
14. Hellström, E., Ivarsson, M., Åslund, J., and Nielsen, L., "Look-Ahead Control for Heavy Trucks to Minimize Trip Time and Fuel Consumption," *Control Engineering Practice* 17, no. 2 (2009): 245-254, doi:10.1016/j.conengprac.2008.07.005.
15. Filev, D.P. and Kolmanovsky, I., "Markov Chain Modeling Approaches for on Board Applications," in *Proceedings of the 2010 American Control Conference*, 2010, 4139-4145, doi:10.1109/ACC.2010.5530610.
16. Zeng, X. and Wang, J., "A Parallel Hybrid Electric Vehicle Energy Management Strategy using Stochastic Model Predictive Control with Road Grade Preview," *IEEE Transactions on Control Systems Technology* 23, no. 6 (2015): 2416-2423, doi:10.1109/TCST.2015.2409235.
17. Zhu, D., Pritchard, E., Dadam, S., Kumar, V. et al., "Optimization of Rule-Based Energy Management Strategies for Hybrid Vehicles using Dynamic Programming," *Combustion Engines* 184, no. 1 (2021): 3-10, doi:10.19206/CE-131967.
18. Zeng, X. and Wang, J., "A Two-Level Stochastic Approach to Optimize the Energy Management Strategy for Fixed-Route Hybrid Electric Vehicles," *Mechatronics* 38 (2016): 93-102, doi:10.1016/j.mechatronics.2015.11.011.
19. Zeng, X. and Wang, J., "Globally Energy-Optimal Speed Planning for Road Vehicles on a Given Route," *Transportation Research Part C: Emerging Technologies* 93 (2018): 148-160, doi:10.1016/j.trc.2018.05.027.
20. Kamalanathsharma, R.K. and Rakha, H.A., "Multi-Stage Dynamic Programming Algorithm for Eco-Speed Control at Traffic Signalized Intersections," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, 2013, 2094-2099, doi:10.1109/ITSC.2013.6728538.
21. Ozatay, E., Onori, S., Wollaeger, J., Ozguner, U. et al., "Cloud-Based Velocity Profile Optimization for Everyday Driving: A Dynamic-Programming-Based Solution," *IEEE Transactions on Intelligent Transportation Systems* 15, no. 6 (2014): 2491-2505, doi:10.1109/TITS.2014.2319812.
22. Xiao, S., Aji, A.M., and Feng, W., "On the Robust Mapping of Dynamic Programming onto a Graphics Processing Unit," in *2009 15th International Conference on Parallel and Distributed Systems*, 2009, 26-33, doi:10.1109/ICPADS.2009.110.
23. Wu, C.-C., Ke, J.-Y., Lin, H., and Jhan, S.-S., "Adjusting Thread Parallelism Dynamically to Accelerate Dynamic Programming with Irregular Workload Distribution on GPGPUs," *International Journal of Grid and High Performance Computing* 6, no. 1 (2014): 1-20, doi:10.4018/ijghpc.2014010101.
24. Jóhannsson, Á.Þ., "GPU-Based Markov Decision Process Solver," Master Thesis, 2009.
25. Nishida, K., Ito, Y., and Nakano, K., "Accelerating the Dynamic Programming for the Matrix Chain Product on the GPU," in *2011 Second International Conference on Networking and Computing*, 320-326, 2011, doi:10.1109/ICNC.2011.62.
26. Berger, K.-E. and Galea, F., "An Efficient Parallelization Strategy for Dynamic Programming on GPU," in *2013 IEEE International Symposium on Parallel Distributed Processing, Workshops and Ph.D. Forum*, 1797-1806, 2013, doi:10.1109/IPDPSW.2013.208.
27. Yamashita, K., Ito, Y., and Nakano, K., "A GPU Implementation of Bulk Execution of the Dynamic Programming for the Optimal Polygon Triangulation," in: , *Parallel Processing and Applied Mathematics*, (Cham: Springer International Publishing, 2018), 314-323, doi:10.1007/978-3-319-78024-5_28, ISBN:978-3-319-78024-5.

28. Steffen, P., Giegerich, R., and Giraud, M., "GPU Parallelization of Algebraic Dynamic Programming," in: Wyrzykowski, R., Dongarra, J., Karczewski, K. and Wasniewski, J. (Eds), *Parallel Processing and Applied Mathematics*, (Berlin, Heidelberg: Springer Berlin Heidelberg, 2010), 290-299, doi:10.1007/978-3-642-14403-5_31, ISBN:978-3-642-14402-8.

29. Aldrich, E.M., Fernández-Villaverde, J., Ronald Gallant, A., and Rubio-Ramírez, J.F., "Tapping the Supercomputer under Your Desk: Solving Dynamic Equilibrium Models with Graphics Processors," *Journal of Economic Dynamics and Control* 35, no. 3 (2011): 386-393, doi:10.1016/j.jedc.2010.10.001.

30. Zhang, N., Lim, E.G., Man, K.L., and Lei, C.-U., *CPU-GPU Hybrid Parallel Binomial American Option Pricing*. Vol. 7 (Hong Kong, 2012)

31. Ruiz, S. and Hernández, B., "A Parallel Solver for Markov Decision Process in Crowd Simulations," in *2015 Fourteenth Mexican International Conference on Artificial Intelligence (MICAI)*, 2015, 107-116, doi:10.1109/MICAI.2015.23.

32. Sampathirao, A.K., Sopasakis, P., Bemporad, A., and Patrinos, P.P., "GPU-Accelerated Stochastic Predictive Control of Drinking Water Networks," *IEEE Transactions on Control Systems Technology* 26, no. 2 (2018): 551-562, doi:10.1109/TCST.2017.2677741.

33. Ortega, G., Hendrix, E.M.T., and García, I., "A CUDA Approach to Compute Perishable Inventory Control Policies using Value Iteration," *J Supercomput* 75, no. 3 (2019): 1580-1593, doi:10.1007/s11227-018-2692-z.

34. Perez, W., Ruhela, A., and Tulpule, P., "Benchmarking Computational Time of Dynamic Programming for Autonomous Vehicle Powertrain Control," SAE Technical Paper 2020-01-0968, SAE International, Warrendale, PA, 2020, doi:10.4271/2020-01-0968.

35. Puterman, M.L., *Markov Decision Processes: Discrete Stochastic Dynamic Programming* (John Wiley & Sons, 2014), ISBN:978-1-118-62587-3

36. CUDA C++ Programming Guide PG-02829-001_v11.1, NVIDIA Corporation, 2020.

37. Nguyen, H., *GPU Gems 3*, 1st ed. (Addison-Wesley Professional, 2007), ISBN:978-0-321-54542-8

38. Chandler, R.E., Herman, R., and Montroll, E.W., "Traffic Dynamics: Studies in Car Following," *Operations Research* 6, no. 2 (1958): 165-184, doi:10.1287/opre.6.2.165.

Contact Information

Zhaoyuan Ma,
Robotics Engineering Department,
Worcester Polytechnic Institute,
zma3@wpi.edu.

Xiangrui Zeng,
School of Mechanical Science and Engineering,
Huazhong University of Science and Technology,
zeng@hust.edu.cn (corresponding author).