Integration of Machine Learning and Optimization for Robot Learning

Amir Mosavi and Annamaria R. Varkonyi-Koczy

Abstract Learning ability in Robotics is acknowledged as one of the major challenges facing artificial intelligence. Although in the numerous areas within Robotics machine learning (ML) has long identified as a core technology, recently Robot learning, in particular, has been witnessing major challenges due to the theoretical advancement at the boundary between optimization and ML. In fact the integration of ML and optimization reported to be able to dramatically increase the decision-making quality and learning ability in decision systems. Here the novel integration of ML and optimization which can be applied to the complex and dynamic contexts of Robot learning is described. Furthermore with the aid of an educational Robotics kit the proposed methodology is evaluated.

Keywords: Machine learning · Optimization · Robotics · LEGO

1 Introduction

Today learning has become a major part of the research in Robotics [1]. Machine learning (ML) algorithms in robotics in particular, within autonomous control and sensing, are being used to tackle difficult problems where large quantities of datasets are available which enable Robots to effectively teach themselves [2]. ML as a sub-field of computer science has evolved from the study of pattern recognition and computational learning theory [3]. Furthermore ML is considered as a field of study in artificial intelligence that gives computers the ability to learn from data [4]. To do so ML explores the development of models that can predict and learn from an available dataset [5]. Such models operate with the aid of algorithms capable of making data-driven predictions rather than following explicit codes [6]. Consequently ML is often used in a range of problems where designing precise algorithms is not practical. In this sense ML can replace the human expertise in information treatment [7]. For that matter ML provides the algorithmic tools for dealing with datasets and providing predictions. In fact ML tends to imitate human skills, which in most cases, act exceptional in identifying satisfactory solutions by theoretical or experience-based considerations [8].

Applications of ML in Robotics which highly contribute to Robot learning is vast and yet progressing in a fast pace [1]. Robot vision [9], Robot navigation [10], field Robotics [11], humanoid Robotics [12], legged locomotion [13], modeling vehicle dynamics [14], medical and surgery Robotics [15], off-road rough-terrain mobile Robot navigation [16], are few of the areas within Robotics for which utilizing ML technologies has become popular. It is therefore clearly evidenced that ML has in recent years become an essential part of Robotics. And this has been in fact a response to the frustration with the problems for which it has been proven difficult to conventional coding solutions. For instance in a variety of Robotics platforms the imitation learning techniques [17], inverse optimal control methods [18], programming-by-demonstration [19], and supervised learning techniques [7] have become norm. Further most notable ML technologies utilized in Robotics include; ML techniques for big data [20], self-supervised learning [16], reinforcement learning [21], and multi-agent learning [22].

2 Integration of Machine Learning and Optimization

The intersection research area of optimization and ML has recently engaged leading scientists [23]. ML has made benefit from optimization and on the other hand ML contributed to optimization as well. Today ML is seen as an exceptional replacement for human expertise in information manipulation [24]. In addition ML has the proven ability to simplify optimization functions [25]. Optimization on the other hand is the source of immense power for automatically improving decisions [26]. However in real-life applications, including Robotics, optimization has not had the chance to be used to its full potential [27]. This has been often due to the absence, complexity, or inefficient optimization functions of the complicated problem at hand [28]. Yet in such cases ML has shown the ability of modeling whole or part of the optimization functions on the basis of the availability of a reliable dataset [24]. A number of case studies concerning Robotics problems have been surveyed in literature, e.g. [29], where ML technologies simplify complicated optimization functions.

Nevertheless the long-term vision for Robot learning would be the development of a fully automated system with self-service usage [19]. To reach this goal the novel idea of integration of ML and optimization [20] aims at simplifying the whole learning process by automating the decision-making tasks in an effective manner without requiring a costly learning curve [1]. In this context the learning process is seen as a byproduct of an automated optimal decision.

Learning from the available dataset integrated with optimization can be applied to a wide range of complex, dynamic, and stochastic problems [23]. Such integration has been reported exceptional in increasing the automation level by putting more power at the hands of final-user [30]. Final-user should however specify dataset, desired outputs and CPU time. CPU time is to be set to put a limitation on optimization algorithms' run-time which can be referred as "learning time". The novel integration of ML and optimization has already been used in solving numerous complex cases [31]. Considering these examples, it is observed that once a combination of right ML technologies and optimization designed, suitable for the problem at hand, further algorithm selection, adaptation, and integration, are done in an automated way, and a complete solution for learning is delivered to the final-user [6].

2.1 Integration

Depending on the characteristics of the problem at hand and availability of dataset an arrangement of local optimization algorithms [32] is essential to come up with an optimal decision. Yet local searches leading to locally optimum is an essential principle for solving the discrete and continuous optimization problems. In this context designing a system that is capable of curing local optimum traps is desirable. To do so reactive search optimization (RSO) methodology [33] is used. RSO methodology implements an integration of ML techniques into local and heuristics search [25] for solving real-life optimization problems [5]. RSO includes a so-called "ML application builder" [34] employed to design a system which receives dataset, guide the research, and delivers a competitive application. In fact the "ML application builder" imitates the human skill in providing the automation to the system which is responsible for algorithm selection and parameter tuning. In fact human brain quickly learns and drives future decisions based on previous observations [8]. This is the main inspiration source for inserting ML techniques into the learning curve. This is referred as brain-computer optimization [35] which is an important building block of RSO. Building blocks of RSO include neural networks, statistics, artificial intelligence, reinforcement learning, and active or query learning [24]. Characteristics of RSO include learning on the job, rapid generation and analysis of many alternatives, efficient analysis of what-if scenarios, flexible decision support, diversity of solutions and anytime solutions [25].

3 Case Study

The proposed case study aims at evaluating the RSO methodology with the aid of the educational Robotics kit of LEGO Mindstorms. The objective is to evaluate the ability of learning of a mobile Robot in locating the darkest spot of a paper sheet

(Fig. 1a). The number of the light intensity inspections is limited to a total of nine sample points within the Cartesian coordinate. Today numerous universities around the world teach artificial intelligence classes with the aid of LEGO Mindstorms platform and many literatures describe the educational benefits of this practice [36].

3.1 Implementation

In order to move in a controlled manner within the Cartesian coordinate system in the identified territory the Robot has been upgraded to a new arrangement (Fig. 1b). There are around 200 LEGO parts coming as the standard LEGO Mindstorms kit to build a Robot [36]. In the presented case study a simple arrangement provides the limited straight movements of the mobile Robot. With adding a Matrix kit to a conventional matrix building system a x-y table is created. In addition the mobile Robot is equipped with a color sensor which measures the light intensity (Fig. 1b).

The prospector here is a color sensor which measures the light intensity and reports to a simple code via a USB cable configuration. According to the simple code as the Robot moves along the Cartesian coordinate over the black and white paper nine samples of light intensity are taken. Then to connect this external code to the Robot learning system the measured points are connected to a design of experiment module to further import into a function generator. In this stage we can plot the results on a 3D graph as it is presented in the Fig. 2a. A second degree

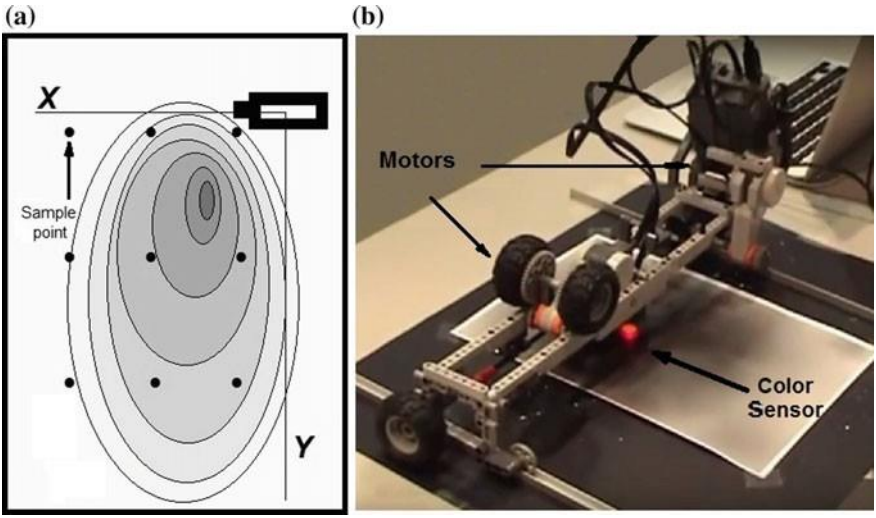

Fig. 1 Robot arrangement; a *Black* and *white* paper sheet presenting a random intensity of light over a Cartesian coordinate system with the coordinates of nine sample points b Robot in action; arrangement of LEGO Mindstorms Robot equipped with a color sensor and a Matrix kit

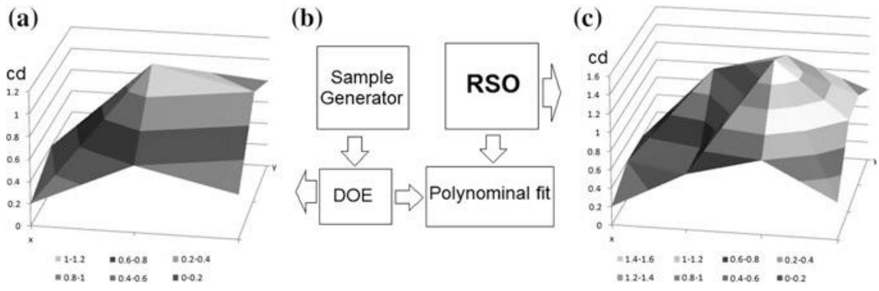

Fig. 2 a 3D graph of the points primary been measured b Building blocks of RSO methodology of learning c 3D graph of the newly generated points

Polynomial fit estimates the distribution of the nine sample points. And RSO runs a continues optimizer in order to predict the optimal points and generate the optimum (Fig. 2b). Robot then is directed to the newly generated optimum accordingly. Matching the predicted optimum with the darkest spot of the sheet proves the accuracy of the model (Fig. 2c).

4 Conclusions

The paper considers the novel integration of ML and optimization for the complex and dynamic context of Robot learning. RSO is introduced as a methodology to implement an integration of ML techniques into local and heuristics optimization for Robot learning. In the proposed case study RSO presents an effective framework based on solving continuous optimization problem with an efficient use of memory and self-adaptive local optimization with self-improvement capabilities in identifying the global optimum. In the case study the ability of learning of a mobile Robot in locating the darkest spot of a paper sheet is evaluated. Matching the predicted optimum with the darkest spot of the sheet proves the accuracy of the model. RSO shown to be able to well imitate the human skills in providing the automation to the system which is responsible for algorithm selection and parameter tuning.

Acknowledgment This work is sponsored by Hungarian National Scientific Fund under contract OTKA 105846 and Research and Development Operational Program for the project "Modernization and Improvement of Technical Infrastructure for Research and Development of J. Selye University in the Fields of Nanotechnology and Intelligent Space", ITMS 26210120042, co-funded by the European Regional Development Fund.

References

1. Connell, J.H., Mahadevan, S.: Robot Learning. Springer Science & Business Media (2012)
2. Knox, W.B., Glass, B.D., Love, B.C., Maddox, W.T., Stone, P.: How humans teach agents. Int. J. Social Robot. 4(4), 409–421 (2012)
3. Bishop, CM., Nasrabadi, NM.: Pattern recognition and machine learning. J. Electron. Imaging 16(4) (2007)
4. Michalski, R.S., Carbonell, JG., Mitchell, TM. (eds.): Machine Learning: an Artificial Intelligence Approach. Springer Science & Business Media (2013)
5. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Elsevier (2011)
6. Mosavi, A., Vaezipour, A.: Reactive search optimization; application to multiobjective optimization problems. Appl. Math. 1572–1582 (2012)
7. Battiti. R., Brunato. M.: Reactive search optimization: learning while optimizing. In: Handbook of Metaheuristics, pp. 543–571. Springer, US (2010)
8. Murphy, R.R.: Human-robot interaction in rescue robotics. Syst. Man Cybernetics Appl. Rev. IEEE Trans. 34(2), 138–153 (2004)
9. Rosten, E., Drummond, T.: Machine learning for high-speed corner detection. In: Computer Vision–ECCV, pp. 430–443. Springer, Berlin (2006)
10. Sofman, B., et al.: Improving robot navigation through self-supervised online learning. J. Field Robotics. 23, 59–75 (2006)
11. Yang, S.Y., Jin, S.M., Kwon, S.K.: Remote control system of industrial field robot. In: 6th IEEE International Conference on Industrial Informatics, pp. 442–447 (2008)
12. Peters, J., Vijayakumar, S., Schaal, S.: Reinforcement learning for humanoid robotics. In: Proceedings of the Third IEEE-RAS International Conference on Humanoid Robots (2003)
13. Kohl, N., Stone, P.: Machine learning for fast quadrupedal locomotion. In: AAAI, pp. 611–616 (2004)
14. Popp, K., Schiehlen, W.: Ground Vehicle Dynamics. Springer, Berlin (2010)
15. Taylor, R.H., Menciassi, A., Fichtinger, G., Dario, P.: Medical robotics and computer-integrated surgery. In: Handbook of Robotics, pp. 1199–1222. Springer, Berlin (2008)
16. Stavens, D., Thrun, S. A.: self-supervised terrain roughness estimator for off-road autonomous driving. arXiv preprint arXiv:1206.6872 (2012)
17. Nehaniv, C.L., Dautenhahn, K.: Imitation and Social Learning in Robots, Humans and Animals: Behavioural, Social and Communication. Cambridge University Press (2007)
18. Mombaur, K., Truong, A., Laumond, J.P.: From human to humanoid locomotion—an inverse optimal control approach. Auton. robots. 28(3), 369–383 (2010)
19. Argall, B.D., Chernova, S., Veloso, M., Browning, B.: A survey of robot learning from demonstration. Robot. Auton. Syst. 57(5), 469–483 (2009)
20. Mosavi, A., Varkonyi-Koczy, A., Fullsack, M.: Combination of machine learning and optimization for automated decision-making. In: Conference on Multiple Criteria Decision Making MCDM, Hamburg, Germany (2015)
21. Kober, J., Bagnell, J.A., Peters, J.: Reinforcement learning in robotics: a survey. Int. J. Robot. Res. 32(11), 1238–1274 (2005)
22. Panait, L., Luke, S.: Cooperative multi-agent learning: The state of the art. Auton. Agent. Multi-Agent Syst. 11(3), 387–434 (2005)
23. Sra, S., Nowozin, S., Wright, S.J.: Optimization for Machine Learning. Mit Press (2012)
24. Battiti, R., Brunato, M., Mascia, F.: Reactive Search and Intelligent Optimization. Springer Science & Business Media (2008)
25. Battiti, R.: Reactive search: Toward self-tuning heuristics. Mod. Heuristic Search Methods 61–83 (1996)
26. Battiti, R., Brunato, M.: Reactive Business Intelligence. From Data to Models to Insight. Reactive Search Srl, Italy (2011)

27. Toussaint, M., Ritter, H., Brock, O.: The optimization route to robotics—and alternatives. KI-Künstliche Intelligenz 29(4), 379–388 (2015)
28. Battiti, R., Campigotto, P.: Reactive search optimization: Learning while optimizing. an experiment in interactive multi-objective optimization. In: Proceedings of MIC (2009)
29. Stone, P., Veloso, M.: Multiagent systems: a survey from a machine learning perspective. Auton. Robot. 8(3), 345–383 (2000)
30. Battiti, R., Brunato, M.: The LION way. Machine learning plus intelligent optimization. Appl. Simul. Model. (2013)
31. Mosavi, A.: Decision-Making in Complicated Geometrical Problems. Int. J. Comput. Appl. 87(19) (2014)
32. Horst, R., Pardalos, P.M.: editors. Handbook of Global Optimization. Springer Science & Business Media (2013)
33. Brunato, M., Battiti, R.: Learning and intelligent optimization (LION): one ring to rule them all. Proc. VLDB Endowment 6(11), 1176–1177 (2013)
34. Battiti, R., Brunato, M.: The LION Way: Machine Learning Plus Intelligent Optimization. Trento University, LIONlab (2014)
35. Battiti, R., Brunato, M., Delai, A.: Optimal Wireless Access Point Placement for Location-Dependent Services. Technical Report # DIT-03-052, University of Trento, Italy (2010)
36. Parsons, S., Sklar, E.: Teaching AI using LEGO mindstorms. In: AAAI Spring Symposium (2004)