Assessment of Machine Learning of Optimal Solutions for Robotic Walking

Rodrigo Matos Carnier and Yasutaka Fujimoto
Yokohama National University, Yokohama, Japan
Email: Rodrigo.carnier@gmail.com, fujimoto@ynu.ac.jp

***Abstract*— The generation of optimal solutions for robotic bipedal walking using whole-body dynamics is well known to have a big computational cost, preventing online trajectory generation for optimal control methods that satisfy Pontryagin's Principle and its Conditions of Optimality. However, bipedal walking has fundamental kinematic and dynamic characteristics that shape different solutions for different parameters in similar curves. Such characteristics were previously defined in biomechanical literature as movement primitives. Recently, studies generated parametrized optimal solutions by performing regressions from training data into movement primitives using Machine Learning. The learned solutions were very close to the actual optimal solution. This study evaluates the precision of such strategy by optimizing the gait of a 6 degrees of freedom planar robot using different Cost Functions, in order to understand if the precision of Machine Learning in recreating optimal solutions is impacted by what is being optimized.**

***Index Terms*— robotic bipedal walking, machine learning, optimal control, movement primitives**

I. Introduction

Optimal Control is a field of study that has become popular in the last 20 years. Even though its mathematical theory was developed more than 60 years ago, it started to be widely investigated only with the advent of greatly increased computational power in the beginning of 2000's [1]. Consequently, many challenges remain, including the realization of proper optimization (in contrast to suboptimization) in real time. While most current implementations of online optimization only decrease a Cost Function, proper optimization satisfies the Conditions of Optimality defined by Pontryagin's Principle or its equivalents (e.g. Hamilton-Jacobi-Bellman Equation). However, the computational cost of solving an optimization problem satisfying these conditions is still critically high for nonlinear systems with many degrees of freedom and very dynamic behavior – in other words, systems with dynamics changing in the milliseconds, needing to be controlled in such a timespan.

Robotic bipedal walking is one of such fields. Inspired in the locomotion of human beings, it has come a long way in its 50 years or so of life, generating impressive prototypes that are able to perform complex human tasks and interact with humans, like Atlas [2], HRP-5P [3] and ASIMO [4]. However, the best prototypes in the world nowadays are still nowhere near the level of performance of human beings [5] [6]. Controlling such an underactuated and unstable mechanical task is one of the reasons, as well as the advanced optimization achieved by human evolution while improving different performance criteria at the same time (e.g. speed of reflexes and energy expenditure). Since evolution is a form of optimization and the human gait is the natural inspiration for robotic walking, Optimal Control is becoming a natural choice for improvement of performance, but it cannot yet perform very well in real time. Therefore, optimal solutions for the locomotion of biped robots are either online but imprecise, or precise but offline.

Another way of improving walking performance is applying Machine Learning to extract better solutions from experience and repetition. This has been done for several different purposes, from decision making of when to step to avoid a fall [7] to the complete gait generation through Reinforcement Learning [8]. Another interesting approach is to reproduce the performance of Optimal Control by generating new optimal solutions from a set of training optimal solutions without reproducing the whole optimization process. Solutions generated this way can be produced very fast, partially circumventing the problems of real time complex optimization. Recently, this has been done by extracting movement primitives from optimal solutions [9] [10]. Movement primitives are fundamental kinematic and dynamic characteristics shared by different solutions of specific classes of movements (like walking, running, jumping or grasping). Given a body morphology (e.g. the human or the avian body morphology), the shape of the curve of a solution for bipedal walking is just slightly changed depending on the parameters of movement (like body mass, length of body limbs, speed of movement, distance traversed). Based on this, Koch et al. [10] extracted movement primitives from optimal solutions for bipedal walking through Principal Component Analysis, parametrized the MP according to step size of walking and then performed a stochastic regression by Gaussian Process to generate new solutions very close to the actual optimal solution.

The present study investigates the precision of such approach. Koch et al. [10] investigated only the learning of optimal solutions minimizing the square of torques

Manuscript received March 14, 2020; revised November 1, 2020.

doi: 10.18178/ijmerr.10.1.44-48

applied on joints. In the present study we will confront the results of minimization of torques with the minimization of energy consumption, modelled as friction in the joints. In the sequence of this introduction, we will describe the dynamics of the model, the optimal control problem formulation, the learning methodology used and the gait synthesis of our biped model. An evaluation of success of the machine learning method will be explained in the last section, together with a discussion of the results.

II. Dynamics

The model used for studying the optimization of biped walking is a planar model of robot with 6 DOF. Control of walking is performed only in the sagittal plane. The model is treated as a manipulator fixed in the ground (Fig. 1), since a model with the ankle of the support foot fixed to the floor can be considered equivalent to a free floating robot model as long as the forces and torques applied by the floor to the fixed foot never pull it to the floor, nor take the Zero-Moment Point [11] away from the base of support of the robot.

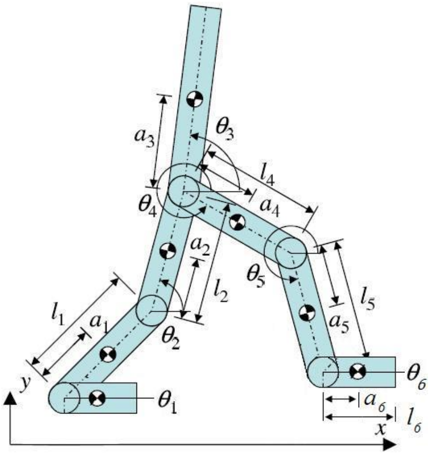

Figure 1. Planar model with 6 DOF, with representation of link angles.

The system states are the absolute angular position and the angular velocity of the robot links. The position is represented by θ and is measured in respect to the horizontal axis (x) of the world reference, fixed outside the robot. The dynamics of the mechanical system is derived in the usual way from its Euler-Lagrange Equations [12]:

$$M(\theta)\ddot{\theta} + B(\theta,\dot{\theta}) + C(\dot{\theta}) + G(\theta) = \eta \quad (1)$$

where M is the Total Inertia matrix, B is the Coriolis and centrifugal effects, C is the friction acting on joints, G is the gravitational effect, η is the internal torques working on the links by joint actuators, J is the jacobian of the external forces applied on the end-effector by contact with the ground, and R is the ground reaction forces.

If we group the two vectors of states – angle of links and angular velocity of links – in a single vector of states, we obtain the following concise formulation of the system dynamics in state-space:

$$f(x,u) = \dot{x}(t) = \begin{bmatrix}\dot{\theta}\\ \ddot{\theta}\end{bmatrix} = \begin{bmatrix}\dot{\theta}\\ M^{-1}(\eta - B - C - G)\end{bmatrix} \quad (2)$$

III. Optimal Control Problem

The optimal control method used in this study is a pseudospectral method based on the Pontryagin's Principle and the Covector Mapping Theorem [1]. The method relies on discretizing the system and minimizing the control Hamiltonian of the system in order to achieve the optimal solution.

In this work, we optimized two different cost functions: minimization of actuators efforts and minimization of energy consumption. These optimizations were performed independently of each other. For each one, a different analytic formulation of cost was used:

1. **Minimization of Actuators Efforts:** A common objective of optimization in robotics is the generation of smooth controls. This can be done by minimizing the square of torques applied in the joints $\tau^T\tau$.
2. **Minimization of Energy Consumption:** In our dynamic model, energy loss is modeled as viscous friction in the joints ($C = \mu\dot{q}$), which is proportional to the joint angular velocities ($\dot{q}$). The optimization is performed by minimizing the square of friction $(\mu\dot{q})^T(\mu\dot{q})$.

In both costs, joint quantities are being used (τ and $\dot{q}$). However, our dynamic formulation was based on links reference. To maintain reference coherence, τ and $\dot{q}$ are expressed in terms of η and $\dot{\theta}$ through the transformations below:

$$\dot{q} = K^{-1}\dot{\theta} \qquad \tau = G^{-1}\eta$$

$$K = \begin{bmatrix}1&0&0&0&0&0\\1&1&0&0&0&0\\1&1&1&0&0&0\\1&1&1&1&0&0\\1&1&1&1&1&0\\1&1&1&1&1&1\end{bmatrix} \quad G = \begin{bmatrix}1&-1&0&0&0&0\\0&1&-1&0&0&0\\0&0&1&-1&0&0\\0&0&0&1&-1&0\\0&0&0&0&1&-1\\0&0&0&0&0&1\end{bmatrix} \quad (3)$$

where *G* and *K* are transformation matrixes that depend on the kinematic chain of the robot.

The minimization of torques is then defined as an objective of optimization by (4), while the minimization of friction is defined by (5).

$$J_{torque} = \int_{t_0}^{t_f} \frac{1}{2}(G^{-1}\eta)^T(G^{-1}\eta)dt \quad (4)$$

$$J_{friction} = \int_{t_0}^{t_f} \frac{1}{2}(\mu K^{-1}\dot{\theta})^T(\mu K^{-1}\dot{\theta})dt \quad (5)$$

With these definitions, we solved the optimal trajectory generation of model of Fig. 1 using the commercial optimal solver DIDO. It is necessary to provide three different elements to the solver: 1) the analytic expressions of the system dynamics, of the cost function J and of any path constraints (floor contact constraints, in our case); 2) the boundaries for the search space of states, controls and constraints; and 3) the desired initial and final value of states. In our case, these initial and final states represent the initial and final stance of the bipedal robot model. DIDO then generates trajectories for the states and controls, which bring the robot from the initial stance to the final stance, making the robot take a step. A detailed description of our methodology for solving such a nonlinear multidegree-of-freedom system is given in Carnier and Fujimoto [13].

For more details on DIDO or its pseudospectral method, refer to Ross [14] [15] or Ross and Fahroo [16].

IV. Movement Primitives

Given a mechanical walking system with its morphology, parameters and constraints, the variability of kinematic and dynamic trajectories for the states of the system are restricted to a fundamental dynamic behavior, which can be expressed as proto-trajectories called movement primitives. This similar dynamic behavior in different walking solutions can be observed in the same shape of its state and control trajectories. Which dynamic properties are present in the movement primitive and which are not, depends on the training set of solutions used to extract the movement primitive. If a set of optimal solutions is used, the property of optimality can possibly be extracted.

The process is done as follows: first we generate a number N_{sol} of solutions for different values of a given parameter, which in our case is the length of walking step. Each solution has N_{DOF} trajectories of link angles, each with N_t points of discretization. Then, we group all these time-series trajectories of link angles into a single matrix X of size $(N_{sol}\ N_{DOF}) \times N_t$. In our case, X has a size of 48 × 15. The extraction of the movement primitive is done by performing a Single Value Decomposition (SVD) of X. According to Principal Component Analysis theory, SVD extracts eigenvalues of the Principal Component of a set of solutions, which in our case is the essential dynamics of an optimal solution of biped gait. SVD extracts the three following matrixes:

$$[U, S, V] = \mathrm{SVD}\ (X) \tag{6}$$

The movement primitive then is calculated from these terms according to (7):

$$M = S\ V^{T} \tag{7}$$

These matrixes represent the following information:

1. M: the movement primitive.
2. U: a matrix of weights that multiply the movement primitive in order to generate one or more solutions. This matrix U in particular is the matrix of original weights, that will regenerate X if we perform the multiplication UM.

V. Machine Learning

Machine Learning is a broad definition of many techniques that take a training set of information, solution or decision and extract a structure able to make predictions of the same sort. In very simplified sense, Machine Learning is a much more abstract version of equation regression. For our purposes, Machine Learning is used to parametrize the set of optimal solutions taken as training data, in order to only change the value of the parameter to obtain new solutions.

The method of Machine Learning applied in this study is Gaussian Process. It is a probabilistic type of regression based on gaussian distributions and differ from deterministic regressions by adding flexibilization to the learning process and allowing incremental improvement of learning experience.

For comparison sake, take the deterministic types of regression: these simpler regressions are based on taking observed points to find a function or curve that can approximate the points. This can be done minimizing an error function between the parametrized function on a time instant and the observed point on the same instant (e.g. least square root of distance between observation points and function points). In other words, the regression process consists of calculating the function parameters that make the approximate function fit the observed points best. The resulting function then can be used to predict new points.

A probabilistic type of regression creates instead a probabilistic distribution that best describes the likelihood of new data matching the observed behavior. Like deterministic regressions, it creates a form of parametrization of the observed data that can be used to predict new information. But unlike deterministic regressions, it takes uncertainty of data into account and give more tools to work around it.

First, uncertainty is accounted for in the calculation of a learned model. Instead of representing a deterministic function, it represents a gaussian distribution that maximizes its marginal log-likelihood of reproducing the observed data (in other words, the parameters of distribution that makes the most probable point of distribution – the mean – closest to the observation points). The probabilistic regression method used in this study – Gaussian Process – further exploits the uncertainty to improve the regression. Since the uncertainty of regression is dependent on the quantity and variability of initial observed data used to generate the parametrized distribution, Bayes' Theorem is used to improve the certainty of regression by allowing new observations to be inserted into the regression, in order to expand it. In this way, an initial regression can be improved over time, by simply decreasing its uncertainty instead of performing a new regression. Much like biological agents evolve and expand their experience to improve their performance.

To machine learn the new solutions with different parameters, first a gaussian regression model needs to be extracted from the training optimal trajectories. This is done by performing n model regressions:

$$n = N_{DOF}\ N_{MP} \tag{8}$$

where N_{MP} is the number of movement primitives (i.e. the number of lines of matrix M). Since our model has 6 DOF and the chosen number of movement primitives is 5, a total of 30 elements need to be learned for each new optimal solution we desire to generate through Machine Learning. The routine *fitrgp* implemented in MATLAB is used for this regression. Since the different training solutions were used varying the parameter of size of stride in gait, this parameter is given as input of the parametrized regression.

After learning the model, new solutions for different parameters are generated by maximizing the marginal log-likelihood of the regression model for a solution with a new desired stride length. This is done through routine *predic*, also implemented in MATLAB.

General details of the followed methodology can be found in Koch et al. [10].

VI. IMPLEMENTATION

The robot model was designed with parameters equivalent to a human of average size. It had 1.35m of height and 48kg of mass distributed in its links (the torso concentrated most of its mass, at 38kg). During gait the hip was kept constant at 0.5m, and initial and final velocities of all links were zero. The gait of this model was designed to satisfy the classic Zero-Moment Point (ZMP) criterion. The ZMP trajectory of our optimized gait is shown in Fig. 2.

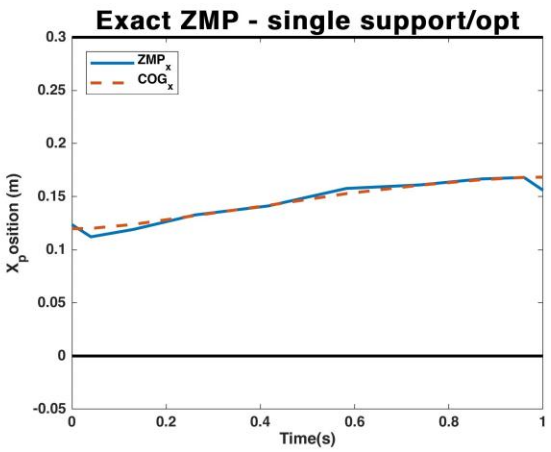

Figure 2. ZMP trajectory for walking step. Black lines represent the boundaries of the base of support.

To generate the training data for Machine Learning, solutions for 9 different sizes of walking step were generated, ranging from 0.4m to 0.8m. From these solutions, the one for 0.6m of walking step size was set apart for comparison with the machine learned solution generated for the same walking step size. In order to evaluate the success of optimization, the non-optimal solution used as initial guess was also simulated, and its data was compared to the data of the optimal solution. Details on the generation of the non-optimal solution and of the optimization process can be found in Carnier and Fujimoto [13].

Below, Fig. 3 presents the trajectories of states and controls for optimal and non-optimal gaits in the case of minimization of friction as described by (5). In the trajectories of states, full lines represent angular position of links and traced lines represent angular velocity.

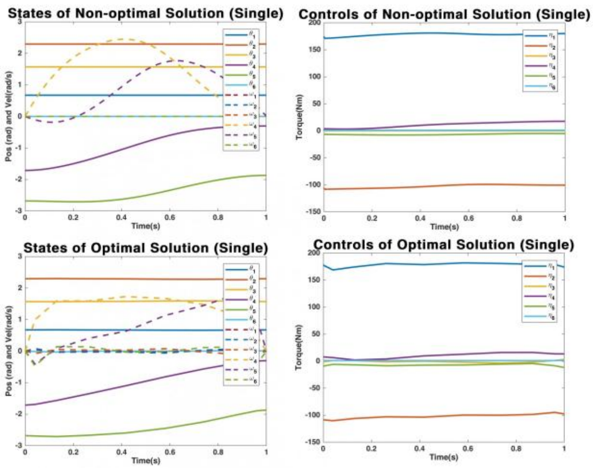

Figure 3. Trajectory of states and controls for minimization of friction. Top: non-optimal solutions. Bottom: optimal solutions.

VII. MACHINE LEARNING RESULTS

The direct evaluation of success of the methodology is the calculation of the Cost Functions (4) and (5) for the non-optimal solution, machine learned solution and optimal solution. The results are given in Table I.

TABLE I. COST FUNCTIONS OF MACHINE LEARNED, NON-OPTIMAL AND OPTIMAL SOLUTIONS

Solution	**Cost Function**	
	Torque $(Nm)^2$	**Friction** $(Nm)^2$
Non-optimal	2.3466e3	0.7494
Optimal	1.2503e3	0.7149
Learned	1.2496e3	0.7153

The results show an improvement (decrease) of costs from optimal to non-optimal solutions of 46.718% for the minimization of torque 4.6038% for the minimization of friction. Even though the optimization of the former seems to be much more successful than of the later, it is explained by the difficult in having a big energy minimization just by decreasing friction for the same task, while the torque used to swing a leg can vary considerably for the same task with different trajectories. Therefore, the real evaluation of the methodology of learning optimal solutions by machine learning is done by verifying the degradation of optimal solution from the actual optimized solution to the learned optimal solution. The degradation is calculated as the percentage increase of the Cost from the optimized solution to the learned solution. For the minimization of torque, it amounts to 0.05599%, while for the minimization of friction, to 0.05595%. In other words, there is practically no degradation at all from the actual optimal solutions to the learned ones, even with a training set of optimal solutions of only 8 different parameters, with 15 points of discretization of time.

The results demonstrate a high ability of replicating the performance of optimal solutions with very few training solutions, and confirms the ability of Machine Learning in generating optimal solutions much faster than actually solving the optimization: in less than half a second, in comparison to the 40s of computation required by the optimal solver to generate an optimal solution.

VIII. SUMMARY

This paper assessed the precision of machine learning optimal solutions for a planar biped robot model in the task of walking. The optimal solutions used as training data were generated by pseudospectral optimization of smoothness and of energy consumption in the bipedal locomotion of a planar model of robot walker, using its whole-body dynamics for precise optimization. Movement primitives that contains the core dynamics of biped gait were extraction from a set of optimal solutions with different parameters (length of stride) and applied Machine Learning to create new optimal solutions from the movement primitives.

Learned optimal solutions were compared with optimal solutions generated by actual optimization process. A very small deterioration of only about 0.055% of the Cost Value of the optimal solution in respect to the actual optimal solution was observed, confirming the ability of the methodology in reproducing optimal solutions with a decrease of computational time from 40s to less than half a second.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Rodrigo Matos Carnier idealized the assessment, conducted the research and wrote the paper; Yasutaka Fujimoto supervised and reviewed both research and paper; all authors had approved the final version.

REFERENCES

[1] I. M. Ross, "A historical introduction to the covector mapping principle," *Advances in the Astronautical Sciences: Astrodynamics*, vol. 122, pp. 05-332, 2005.

[2] G. Nelson, A. Saunders, R. Playter, "The PETMAN and Atlas robots at Boston dynamics," in *Humanoid Robotics: A Reference*, A. Goswami, P. Vadakkepat, pp. 169-186, Springer, 2018.

[3] K. Kaneko, H. Kaminaga, T. Sakaguchi, S. Kajita, M. Morisawa, I. Kumagai, F. Kanehiro, "Humanoid robot HRP-5P: An electrically actuated humanoid robot with high-power and wide-range joints," in *Proc. IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1431-1438, 2019.

[4] S. Shigemi, "ASIMO and humanoid robot research at Honda," in *Humanoid Robotics: A Reference,* A. Goswami, P. Vadakkepat, ed. pp. 55-90, Springer, 2018.

[5] G. H. Z. Liu, M. Z. Q. Chen, and Y. Chen, "When joggers meet robots: the past, present, and future of research on humanoid robots," *Bio-Design and Manufacturing*, vol. 2, no. 2, pp. 108-118, 2019.

[6] E. Krotkov, D. Hackett, L. Jackel, M. Perschbacher, J. Pippine, J. Strauss, G. Pratt, and C. Orlowski, "The DARPA robotics challenge finals: Results and perspectives," *Journal of Field Robotics*, vol. 34, no. 2, pp. 229-240, 2017.

[7] C. Kouppas, Q. Meng, M. King, and D. Majoe, "S.A.R.A.H.: The bipedal robot with machine learning step decision making," *International Journal of Mechanical Engineering and Robotics Research*, vol. 7, no. 4, pp. 379-384, 2018.

[8] C. R. Gil, H. Calvo, and H. Sossa, "Learning an efficient gait cycle of a biped robot based on reinforcement learning and artificial neural networks," *Applied Sciences*, vol. 9, no. 3, 502, 2019.

[9] A. d'Avella and M. C. Tresch, "Modularity in the motor system: Decomposition of muscle patterns as combinations of time-varying synergies," *Advances in Neural Information Processing Systems*, vol. 14, pp. 141-148, The MIT Press, 2002.

[10] K. H. Koch, D. Clever, K. Mombaur, and D. Endres, "Learning movement primitives from optimal and dynamically feasible trajectories for humanoid walking," in *Proc. IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pp. 866-873, 2015.

[11] M. Vukobratovic and B. Borovac, "Zero-moment point - thirty five years of its life," *International Journal of Humanoid Robotics*, Vol.1, No.1, pp.157-173, 2004.

[12] T. Sugihara, Y. Fujimoto, "Dynamic analysis: Equations of motion," in *Humanoid Robotics: A Reference*, A. Goswami, P. Vadakkepat, pp.723-754, Springer, 2018.

[13] R. M. Carnier and Y. Fujimoto, "Numerical techniques for the optimization of gait generation," *IEEJ Journal of Industry Applications*, vol. 10, no. 2, 2021. (Accepted)

[14] *A Beginner's Guide to DIDO: A MATLAB Application Package for Solving Optimal Control Problems*, Elissar Global, 2007.

[15] I. M. Ross, *A Primer on Pontryagin's Principle in Optimal Control*, Collegiate Publishers, 2009.

[16] I. M. Ross and F. Fahroo, "Legendre pseudospectral approximations of optimal control problems," *Lecture Notes in Control and Information Sciences*, vol. 295, pp. 327-342, Springer-Verlag, 2003.

Rodrigo Matos Carnier received the B.E degree in mechatronics engineering from University of Sao Paulo, Sao Paulo, Brazil, in 2013, and the M.E. degree in electrical and computer engineering from Yokohama National University, Yokohama, Japan, in 2017, where he is currently a Ph.D candidate. His research interests include biped locomotion, motion control, robotics, optimal control, computer networks and Internet architecture.

Yasutaka Fujimoto received the B.E., M.E., and Ph.D. degrees in electrical and computer engineering from Yokohama National University, Yokohama, Japan, in 1993, 1995, and 1998, respectively. In 1998, he joined the Department of Electrical Engineering, Keio University, Yokohama, Japan. Since 1999, he has been with the Department of Electrical and Computer Engineering, Yokohama National University, where he is currently a Professor. His research interests include actuators, robotics, manufacturing automation, and motion control. Dr. Fujimoto is an Associate Editor of the IEEJ-JIA and IEEE-TIE.