Computational design of energy-efficient legged robots: Optimizing for size and actuators

G. Fadini[1], T. Flayols[1], A. Del Prete[2], N. Mansard[1,3], P. Souères[1]

Abstract— This paper presents a computational framework for the design of high-performance legged robotic systems. The framework relies on the concurrent optimization of hardware parameters and control trajectories to find the best robot design for a given task. In particular, we focus on energy efficiency, presenting novel electro-mechanical models to account for the losses of the actuators due to friction and Joule effects. Thanks to a bi-level optimization scheme, featuring a genetic algorithm in the outer loop, our framework can also optimize for the duration of the motion, the actuators, and the size of the robot. We present a novel approach to scale both the actuators and the robot structure in a way that ensures structural integrity by maintaining constant the normalized deflection of the links. We validated our approach by designing a two-joint monoped robot to execute a jumping task. Our results show that our framework can lead to remarkable energy savings (up to **60%**) thanks to the concurrent optimization of robot size, motion duration, and actuators.

Fig. 1. Structure of the monoped and the real platform, courtesy of Max Planck Gesellschaft

I. INTRODUCTION

With the rise of advanced robotic platforms in many domains of human activities, robot design is becoming a key challenge. Robots must be able to fulfill specific task requirements while at the same time consuming the least amount of energy. These two goals are not easy to meet simultaneously and designing a robot that optimizes both of them cannot be simply guided by the experience or the intuition of engineers. Systematic and versatile approaches need to be developed to cope with this multi-objective problem. On top of it, a crucial requirement for any robotic platform is to be properly sized for the task it is intended to perform. Finding the good size requires to consider several questions at a time: a bigger robot may require different actuators to work, a different control strategy, or both. In addition to the scaling problem, the drive for better autonomy motivates the need for efficient actuation capabilities. This requires proper models, both for energy dissipation and for energy recovery from the motor in case of motion inversion. As energy efficiency, task performance and scale are highly interconnected, the need for tools that can explore all these design dimensions simultaneously is of paramount interest.

In this sense, the co-design approach, which aims at optimizing together hardware and control of the system, has proven in the literature to provide better solutions than classical engineering methods that address only one side of the problem at a time [1–4]. Co-design was for instance used to generate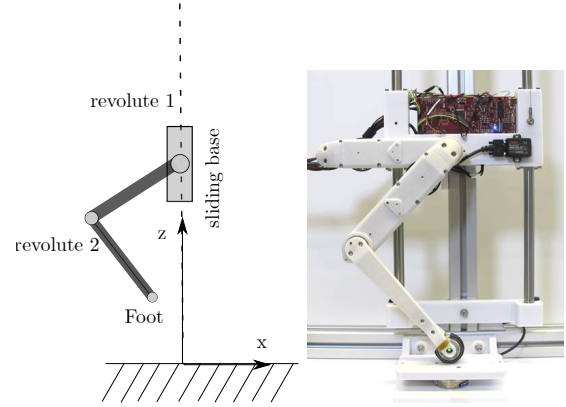 motion laws that could better exploit the under-actuated dynamics to build passive walkers that require minimal control energy [5, 6]. Different studies were conducted to see the effect of various energy minimization strategies applied to the modeling of the actuation for a hopping system [7]. The trade-off between different energy policies was also investigated [8]. At the same time, to obtain natural and robust motions on bio-inspired systems, optimization was also used in various works. In this sense, a study was conducted on a quadruped avatar to find the proper control sequence by exploiting a flexible spine to obtain more realistic motions [9]. The optimization relied on a genetic algorithm and the optimization variables were related to the gait timings and the actuation of the flexible spine. A genetic algorithm was also exploited for a biped with the aim to perform both timing and plant modifications. A model of muscular actuators was used to this end and the minimization dealt with timing and topology [10].

In the light of these previous contributions, this paper proposes a fully integrated approach to optimally design robotic platforms for specific tasks, given a model of their dynamics and their actuation, under the co-design paradigm. Our first contribution is to rely on a purely energetic cost function based on a novel electro-mechanical model of the actuators. This model describes the actuator losses due to Joule effect in the motor winding as well as viscous and Coulomb friction as functions of the design parameters (motor and gear-box). Thanks to this, we do not need to add hand-tuned regularization terms to our cost function, which is typically needed to obtain natural looking motions. The second contribution is a consistent method to scale the robot. Understanding how the robot size impacts the

[1] LAAS-CNRS, Université de Toulouse, CNRS, Toulouse, France
[2] Industrial Engineering Department, University of Trento
[3] Artifical and Natural Intelligence Toulouse Insitute (ANITI)

power consumption is of utmost interest to obtain the design that allows for the most efficient motion, but has been mostly overlooked in the literature. The modification of the robot size should be carefully performed in order to avoid weakening the robot structure. Our method encompasses this problem by suggesting a proper structural change in which the size, mass and inertia are adapted consistently while conserving the structural integrity. A case study involving the test bed of the Solo [11] quadruped leg is used to illustrate the consistency of the method (see Fig. 1). The considered task involves a contact phase during which the robot pushes on the floor to propel its body upward in order to reach a target position with the leg stretched vertically. This work is carried out within the framework of the Open Dynamic Robot Initiative, which aims at developing an open torque-controlled modular robot architecture for legged locomotion research. The objective of this co-design study is to develop a versatile framework for improving the robot design to increase its mobility performance.

II. METHODOLOGY

A. Problem statement and proposed solving strategy

The proposed approach is suited to enable the design optimization for a large variety of robots. The main investigated physical features $\mathcal{D}$ are the following ones.

- Structure length, mass and inertia.
- Actuator peak capabilities, torque and velocity limits.
- Gear ratio n.
- Motor mass m_m and rotor inertia I_m.

On the other hand the control parameters $\mathcal{C}$ to be optimized are in the form of phase timings for the task (e.g contact time). In the literature the controller parameters have been optimized with this approach [12].

The problem is stated as a standard Optimal Control Problem (OCP) augmented with the additional decision variables. The task is fixed (e.g., a target position to reach, and a fixed time to reach it) and the terminal cost enforces it. The problem is to minimize the cost function in Eq. 1 by selecting the *best* trajectories of the control $\boldsymbol{u}$ and state $\boldsymbol{x}$, and the *best* parameters of the design $\mathcal{D}$ and control $\mathcal{C}$.

$$\underset{\boldsymbol{x},\boldsymbol{u},\mathcal{D},\mathcal{C}}{\text{minimize}} \quad \Phi(\boldsymbol{x}_T,\boldsymbol{u}_T,\mathcal{D},C)+\int_{t_0}^{T}\mathcal{L}(\boldsymbol{x}(t),\boldsymbol{u}(t),\mathcal{D},C)\mathrm{dt}$$
$$\dot{x}(t)=f(t,\boldsymbol{x}(t),\boldsymbol{u}(t),\mathcal{D},C) \qquad (1)$$
$$\text{subject to} \quad h(t,\boldsymbol{x}(t),\boldsymbol{u}(t),\mathcal{D},C)\le 0$$
$$g(t_0,T,\boldsymbol{x}(t_0),\boldsymbol{x}(T),\mathcal{D})\le 0$$

The optimization is performed by a two-stage algorithm on a discrete version of problem (1). This splitting of the problem makes it tractable and was first implemented to study gait timing in [13]. Moreover the approach is chosen because it is fairly general and can be readily adapted to the study of more complicated problems. The algorithm is structured as follows (Fig. 2):

- A genetic optimizer randomly samples the physical design and control parameters $\mathcal{D},\mathcal{C}$ for a given problem, generating a population
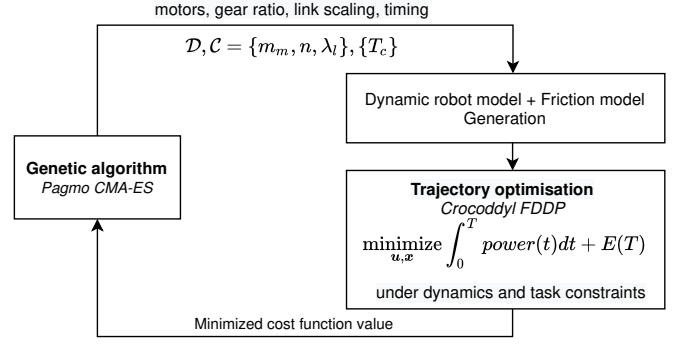
Fig. 2. Simplified scheme of the bi-level optimization.

- For each individual of the population, a model of the problem is instantiated for the given task and cost function. At this stage, all the parameters are fixed and the problem takes the form of a standard OCP. The trajectory optimization is performed by using a DDP-like approach [14].
- The final values of the cost function are sent back to the genetic algorithm, which in turns generates the next population. This last step is connected to the first one and this loop is reiterated until an ending condition is met.

The selection of the most suited designs at each iteration enables a more efficient exploration of the different combinations of all parameters.

B. Energy minimization

The considered cost function takes into account the electro-mechanical parameters of the actuation in order to evaluate the power losses. The goal of the optimizer is to find the motion that achieves the task while minimizing the required electrical energy. A straightforward way to do so would be to include all friction effects in the system dynamics. Instead of doing that, we decided to leave the dynamics friction-less (as often done in robotic trajectory optimization) and account for friction in the cost function. The idea behind evaluating the energy losses is that, on a physical system, when the friction torques are compensated by the actuator torques, the generated motion becomes indistinguishable from the one obtained with an idealized friction-less model. The difference is the additional energy necessary to overcome the friction torque $\boldsymbol{\tau}_f$. Indeed, to execute the motion computed with a friction-less dynamics, the motors need to generate some additional torques to overcome friction. From the control $\boldsymbol{u}$ the total motor torque can be computed:

$$\boldsymbol{\tau}_t = \boldsymbol{\tau}_m + \boldsymbol{\tau}_f = \frac{\boldsymbol{u}}{n} + \boldsymbol{\tau}_f \qquad (2)$$

with n the gear ratio (see Fig. 3). As a result the friction is just present in the cost function, and not in the dynamics, and the goal of the optimizer is to minimize energy expense by choosing proper trajectories of $\boldsymbol{u}$ and $\boldsymbol{x}$.
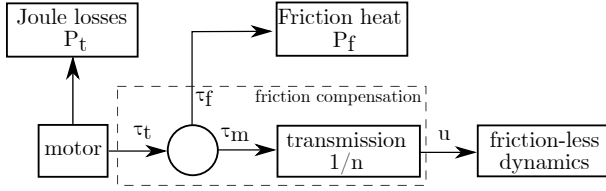
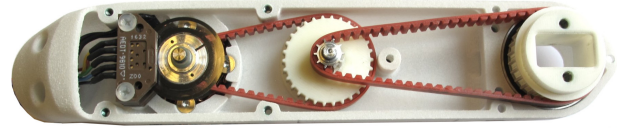Fig. 3. Scheme of the compensation of friction in the cost function

Fig. 4. Actuation module from the Open Dynamic Robot Initiative. It consists in a lightweight PMSM motor associated with two 3:1 gear reduction stages. Key parameters: 160mm link, weight of 150g for an output torque of 2.5Nm

C. Power cost function components

The energy consumption should account for the energy losses due to the non-ideal behaviour of the actuators. If the task consists in moving from a fixed initial state to a fixed fully specified state, since the integral of the mechanical power supplied by the actuators is equal to the difference of mechanical energy (potential+kinetic) between the final and the initial state, the consumed mechanical energy is fixed (i.e. independent of the chosen trajectory). Hence in this case it is not a quantity that needs to be added to the cost function.

The two dissipative mechanisms that we consider are expressed in form of Coulomb and viscous friction at the transmission level, and Joule effect from the winding resistance of the motor. The friction reported to the motor axis is then:

$$\tau_f = \tau_\mu \text{sign}(\omega_m) + b\omega_m \approx \tau_\mu \tanh(\gamma\omega_m) + b\omega_m \text{ [Nm]}, (3)$$

where $\omega_m \left[\frac{\text{rad}}{\text{s}}\right]$ is the motor angular speed, τ_μ [Nm] is the Coulomb friction parameter, b [Nms] is the viscous friction parameter. To approximate the non-differentiable sign function we use the hyperbolic tangent with $\gamma \gg 1$. The value of τ_f allows to compute the value of the losses as:

$$P_f = \tau_f^\top \omega_m \qquad \text{[W]} \qquad (4)$$

The losses due to Joule effect must be expressed by considering the total torque at the motor

$$P_t = \tau_t^\top \mathbf{K} \tau_t \qquad \text{[W]} \qquad (5)$$

where $\mathbf{K}$ is a diagonal square matrix mapping the motor torques to the power dissipated by Joule effect. The main diagonal contains the squared reciprocal of the motor constants K_m. The quantities introduced in (4) and (5) are non-negative and the formulation is also able to encompass the non-perfect regeneration of the electrical energy Joule when the system is in a regenerative phase. In that case, even if the current is back flowing, there is still a non-null dissipation at the level of the winding resistance of the motor and at the joint level.

D. Modeling of the actuators and actuation bounds

Following the methodology proposed in [7, 8], the characteristics of the PMSM motors are interpolated from the values of the *Antigravity, Turnigy, MultiStar* and *PropDrive* datasheet in order to have a good model of the properties of the actuators used in the Open Dynamic Robotic Initiative [11] (see Fig. 4). All the motor characteristics are parametrized with respect to the mass of the motor m_m [kg].

Using the following exponential regression, the peak torque of the real actuator was found with about 15% surplus:

$$\begin{cases} \text{Stall torque} & \tau_{max} = 5.48 \, m_m^{0.97} & \text{[Nm]} \\ \text{Motor constant} & K_m = 0.15 \, m_m^{1.39} & \text{[Nms]} \\ \text{Rotor inertia} & I_m = 7.19 \cdot 10^{-4} \, m_m^{1.67} & \text{[Kg m}^2\text{]} \end{cases} \quad (6)$$

The choice of the motors modifies the plant dynamics as well. Indeed all the bodies to which the motors are attached have an additional localized mass m_m and the effect of the rotor inertia I_m is included in the dynamics as suggested in [15].

E. Scaling choices for the actuators and robot size

a) Actuation: We want to understand how scaling the transmission (i.e. the gear ratio) affects the friction parameters with respect to a nominal gear ratio η for which we have measurements, namely: viscous friction $b_0 = 1.5 \cdot 10^{-5}$ [Nms] and Coulomb friction $\tau_{\mu,0} = 5.3 \cdot 10^{-3}$ [Nm] [11]. To do so, we assume that, to obtain an overall gear ratio n, we can stack in series several stages with the reference transmission η, so that $n = \eta^k$, with k the overall number of stages. Extending the reasoning to a continuous space yields $k = \frac{\log(n)}{\log(\eta)}$ [1]. By reporting the Coulomb friction for each stage i to the motor axis, considering the reduction stack, the overall Coulomb friction appears to be linear in the number of stages. We compute the torque contribution given by Coulomb friction and report it to the motor axis as:

$$\tau_\mu = \sum_{i=0}^{k} \frac{\tau_{\mu,0}\eta^i}{\eta^i} \approx k\tau_{\mu,0}, \qquad \tau_\mu \propto \tau_{max} \qquad (7)$$

So that if the motor torque increases, the belt must be changed as well. We assume that the belt number is linearly proportional to the motor torque, as if multiple motors were working in parallel with the same belt type. A similar consideration was used in [7] for gearbox reductions, stating that the Coulomb friction is proportional to the stall torque. With this in mind the friction torque is at the same time linearly proportional to the number of stages and to the motor torque. For the viscous friction constant, we have to consider the effect of the transmission across the stages: each subsequent axis i has a lower angular velocity and the torque reported to the motor axis is also reduced by the same factor.

---

[1]The reduction ratios corresponding to a non-integer number of stages can always be approached with slight variations in the number of teeth of the pulleys.

So the overall torque on the motor axis resulting from all the viscous friction in the transmission is given by:

$$\tau_b = \sum_{i=0}^{k} \frac{b_0 \omega_m}{\eta^{2i}} = \underbrace{\left(1 + \frac{1 - \eta^{-2k}}{\eta^2 - 1}\right) b_0}_{b} \omega_m \tag{8}$$

The damping coefficient b, from the analytical form, is almost equal to b_0 since for $\eta^2 \gg 1$ each stage is adding a little contribution to the damping.

b) Robot size: We focus now on how change in sizing impacts the dynamics. There are some caveats when dealing with the modification of the structure: the robot cannot be shaped freely because this might yield to weak and unusable design. Thus, we exploit the current knowledge of the system under the following simplifying assumptions:

- The same material is used, no material property will change (e.g., density ρ, Young modulus Y).
- The shape of the system is fixed, only the aspect ratio can change: for each body we consider an homothetic scaling along the principal length with a ratio λ_l and uniform cross-section dimension scaling ratio λ_s with respect to the nominal sizes.
- For structural integrity the deflection δ, normalized with respect to the principal link length l, must be kept constant across the scaling.

$$\frac{\delta}{l} \propto \frac{F l^2}{Y \mathcal{I}} \propto F \frac{\lambda_l^2}{\lambda_s^4} \propto m a \frac{\lambda_l^2}{\lambda_s^4} \propto a \frac{\lambda_l^3}{\lambda_s^2} \tag{9}$$

Such quantity depends on section inertia moment of area $\mathcal{I} \, [\mathrm{m}^4]$ and the inertial load F, which scales linearly with the mass ($m \propto \lambda_l \lambda_s^2$), and the acceleration of the link a. The last requirement, under Froude dynamical similitude [16] (a constant across scaling), results in the relationship

$$\lambda_s = \lambda_l^{3/2} \tag{10}$$

If the scaling law is used as a constraint, the design variables are reduced and structural integrity is enforced. Note that this choice penalizes larger structures more than an isotropic scaling $\lambda_s = \lambda_l$. We are thus aware that more favourable scaling choices, that would not make the moving inertia increase as much, exist and may be adopted with the loss of conservatism. Under these assumptions, the mass and the inertia tensor of the bodies that build the model are modified consistently to the transformation as well as all the lengths that are involved in the problem.

III. RESULTS: MONOPED JUMP

A. Case Study Description

a) Model: In this part we present an application of the presented approach to the case of a jumping monoped, whose body is constrained to move along a vertical prismatic joint (see Fig. 1). This model reproduces the test bed of the Solo leg [17]. Under-actuated dynamics, contact constraints and timing make it a minimal—yet interesting—benchmark case of legged robot. The leg includes two actuated revolute joints

Type	Cost component	Value
Terminal cost	*Task completion*	5e3
Running cost	*Power*	2e0
	Torque bounds	1e1
	Penalty Ground violation contact phase	1e6
	Penalty Ground violation flying phase	1e3

and a body with mass $m_b = 0.37 \, [\mathrm{kg}]$. The reference robot model is instantiated with the nominal design parameters and in the optimization its size and actuators are modified according to the design parameters.

b) Software Framework: For trajectory optimization, this work uses Crocoddyl, a cutting-edge open source library developed for the optimal control of legged robots [14]. In particular we use the FDDP algorithm, this allows to better deal with the sparsity of the optimization problem. For the robot dynamics modeling and computations the Pinocchio library is used [18]. The gradients and Hessians of the highly nonlinear power cost are analytically derived. For the genetic optimization the Pagmo library from ESA [19] is exploited for its versatility and the variety of already implemented algorithms. Among them, the black-box CMA-ES [20] is selected to optimize the co-design parameters. The reasons for this choice are its good convergence properties and its capacity to avoid local minima. The same algorithm was previously chosen for the design of robotic feet [21] and other integrated design problems [10].

c) Task Formulation: The task that the robot should perform consists in jumping, with a fixed overall completion time $T = 1 \, [\mathrm{s}]$. Initially the robot is standing still on the ground with a straight leg. After a contact phase of variable duration T_c, during which the robot can propel its body upwards by pushing on the ground, it breaks the contact and jumps to reach the target position $\boldsymbol{p} = [0, 0, 0.64]^\top \, [\mathrm{m}]$ with its foot. The reaching task is expressed as a terminal cost on the complete state: the foot must reach $\boldsymbol{p}$ with a fully stretched vertical leg configuration and null velocities. The energy is computed in the cost as a running cost, consisting in the integral of the power losses. Inequality constraints cannot be directly formulated in Crocoddyl, so we have approximated them with penalty functions that are null as long as the values are within the limits and grow quadratically when the limits are violated. In particular, we have introduced a penalty on the control bounds and the penetration of the ground as running costs. The weights of the different terms in the cost function are reported in Table I. For integrating the system dynamics we have used an explicit Euler integration scheme with time-steps of $\Delta t = 10^{-3} \, [\mathrm{s}]$.

The following intervals of the design parameters are considered: motor mass $\boldsymbol{m}_m \in [0.05, 1] \, [\mathrm{kg}]$, low-reduction gear ratios $\boldsymbol{n} \in [3, 20]$ and scaling of the links $\pm 20\%$ with respect to the nominal leg design $\lambda_l \in [0.8, 1.2]$, $T_c \in [0.2, 0.8] \, [\mathrm{s}]$. In all the tests, in order to reduce the number of design variables, we use the same motor and gear ratio for all joints.
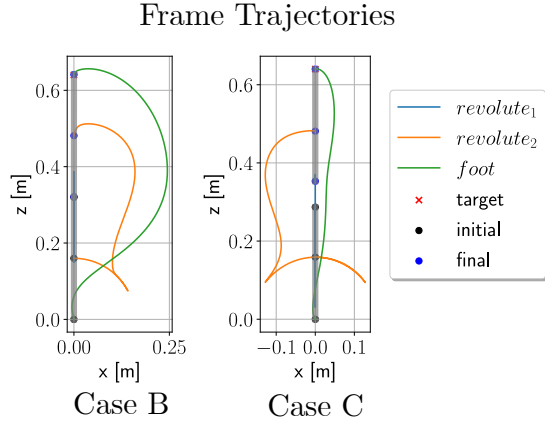
Fig. 5. Optimized joint trajectories

B. Numerical Results

With respect to the optimal solution found fixing $T_c = 0.7[\text{s}]$ and using the current design version of the actuator module Fig. 4, that will be referred to as Case A, the following two optimizations are considered:

- Case B: The scale and the timing are fixed ($\lambda_l = 1$ and $T_c = 0.7[\text{s}]$), whereas the actuator parameters m_m and n are optimized.
- Case C: The size, the timing and the actuator parameters are optimized.

The results obtained for these cases are described and compared in the following.

a) Case B: To better understand the impact of the actuator choice, a preliminary study was made by fixing $T_c = 0.7\text{s}$ and $\lambda_l = 1$. In this case, given the small dimension of the search space, the optimization was performed directly by sampling the space $m_m \times n$ with a 19×18 uniformly spaced grid as shown in Fig. 7. In this case the routine took 1 hour and 39 min, with an average time for trajectory optimization of 14.72s. This figure depicts the trade-off between n and m_m. It appears that small motors with very low reduction lead to high costs because they produce τ_m at the expense of an increased P_t. At the same time high values of m_m and n seem to be sub-optimal since they induce an increase of P_f and of the mass to be moved. Therefore, for fixed scaling and timing, the best combination was obtained in between those two extreme cases, with $m_m = 0.1[\text{kg}], n = 12$. Additionally, for any choice of m_m there seems to exist an optimal value of n, ideally providing the best trade-off between higher friction torque τ_f and increased output τ_m.

The optimal trajectories of the robot links (Fig. 5) and of $\boldsymbol{x}, \boldsymbol{u}$ (Fig. 6) highlight how the system in the contact phase converts the potential energy of the floating base into kinetic energy by letting it slide downward. At the same time bending the knee provides a way to apply a reaction force to push the system upward in the flying phase.

b) Case C: Here the idea is that, by exploiting additional degrees of freedom, we can find optimal design and control for the system. In this case, after evolving a population of 10^2 individuals for 10 times, for solving the co-design
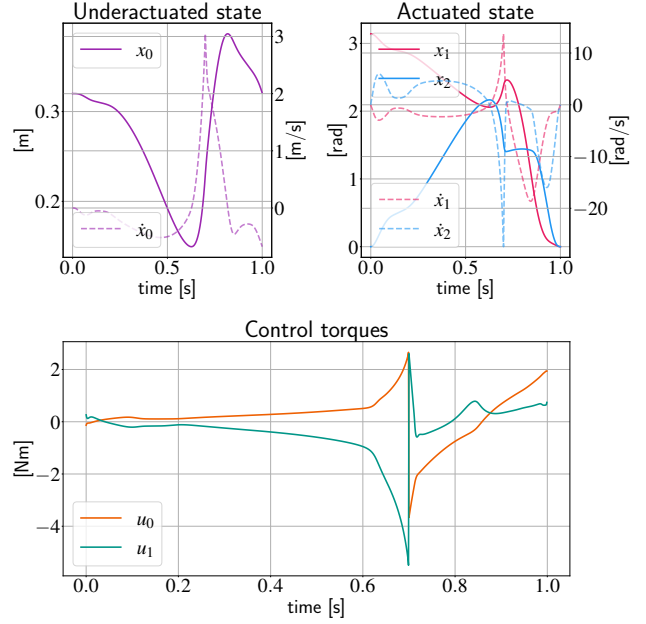

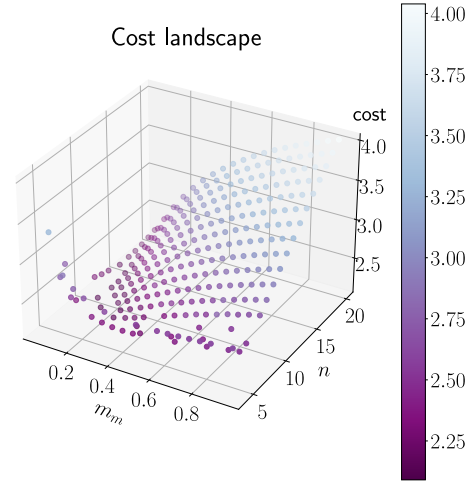Fig. 6. Optimal solution in terms of state and control, Case B

Fig. 7. Cost versus motor mass and gear ratio

optimization, for 10^3 trajectory optimization problems, the overall time was about 9 hours, taking on average 33.72[s] for each Crocoddyl solver call. The optimum is found for:

$$\begin{cases} m_m & = 9.85 \cdot 10^{-2}\text{kg} \\ n & = 4.5 \\ \lambda_l & = [0.83, 0.99] \\ T_c & = 0.729\text{s} \end{cases} \tag{11}$$

The actuators have a higher mass and a lower gear ratio. While the second link is close to the nominal length, the first one almost hits the lower bound of its search space. The optimal trajectories of the frames (Fig. 5) and of $\boldsymbol{x}, \boldsymbol{u}$ (Fig. 8) are quite different from to the ones found as the optimal solutions of case B. In case C, having a lower value of n allows higher joint velocities without incurring in
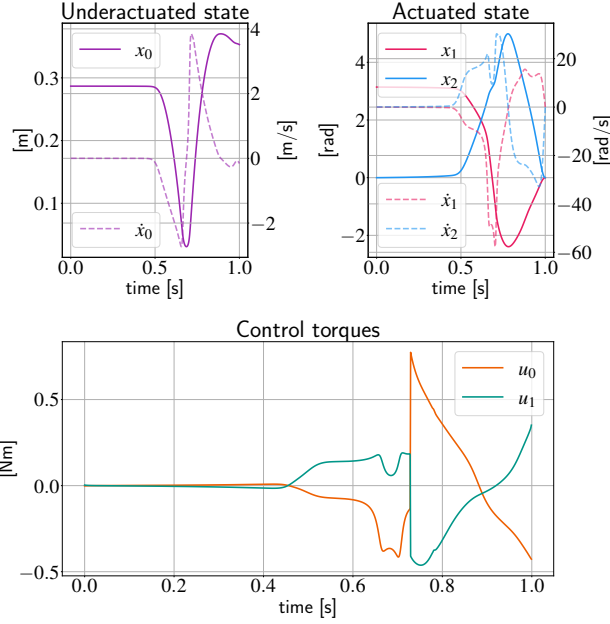
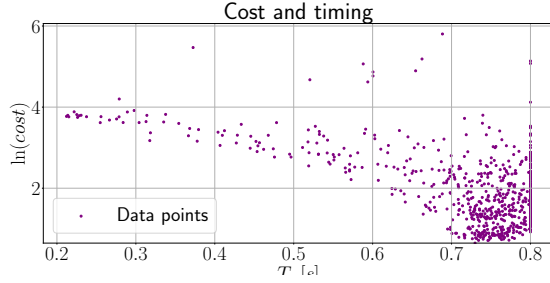Fig. 8. Optimal solution in terms of state and control, Case C
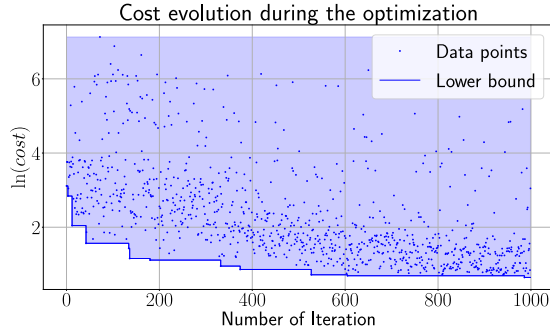

Fig. 9. Effect of timing in the optimization

Fig. 10. Cost evolution versus number of iteration

an excessive P_f. This, together with the structural change, allows the joint torque to be reduced compared to the result of Case B, Fig. 6.

c) Discussion: The total energy required in Case C is reduced with respect to both the nominal design (Case A) and the optimized actuator design (Case B) as shown in Table II. As expected, Case C is showing the best overall improvement, around 61.5%, with respect to the energy

Type	Case A	Case B	Case C
Mechanical Energy	0.96 J	1.32 J	1.47 J
Joint friction	0.87 J	1.60 J	0.38 J
Joule dissipation	4.37J	1.87 J	0.53 J
Total	6.19 J	4.79 J	2.38 J

required for the nominal Case A. The mechanical energy for Case C is higher with respect to reference because motors are twice as large, while the limb size is only slightly changed (17%). This is counter-intuitive and reflects the fact that Joule effects may be extremely detrimental for small motors, as it can be seen in Case A. Case C has much lower friction losses, which can be explained by a lower reduction and lower u with respect to Case B, which minimizes the losses for Joule effect. The effect of timing (Fig. 9) is less trivial to observe and discuss, but there seems to exist a minimum around 0.729[s]. Higher values may not give enough time to the flight phase to reach the goal, while smaller values would not allow to apply efficiently a force on the ground. In Fig. 10 we can appreciate the convergence of the optimization in Case C, as the number of iterations increases there seems to be a diminishing return in the improvement of the cost.

IV. CONCLUSIONS AND FUTURE WORK

This paper presented a computational method to design legged robots that can achieve a user-defined task with minimal energy consumption. The results show that the presented framework is able to exploit the trade-offs between the robot structure, the actuators, and the control strategy, in order to minimize the energy consumption of the task.

This work comes as a first step towards designing the next prototype version of the Open Dynamic Robot initiative. However, beyond this objective our goal is to make this approach applicable to a variety of legged systems, considering even more complex design choices and trade-offs. Examples of more demanding problems are for instance the optimization of the design for various scenarios of locomotion, the topological optimization of the structure or even the selection of the best among different actuator typologies. To do so, since the optimization seems to be already computationally expensive and the complexity of the problems may increase even further, we intend to parallelize computation in the genetic algorithm. Additionally, more complex multi-objective strategies may be necessary to find the best design as in [2, 22]. Regarding the actuator model, in this work we neglected the limits of the actuation bandwidth. This is a crucial information to obtain realistic motions that can be performed on real systems. We are currently investigating a strategy to include bandwidth limits in the framework, without increasing significantly its computational complexity.

REFERENCES

[1] James T. Allison and Sam Nazari. "Combined Plant and Controller Design Using Decomposition-Based Design Optimization and the Minimum Principle". In: vol. Volume 1: 36th Design Automation Conference, Parts A and B. International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Aug. 2010, pp. 765–774. DOI: 10.1115/DETC2010-28887. eprint: https://asmedigitalcollection.asme.org/IDETC-CIE/proceedings-pdf/IDETC-CIE2010/44090/765/2696716/765_1.pdf. URL: https://doi.org/10.1115/DETC2010-28887.

[2] James Allison and Zhi Han. "Co-Design of an Active Suspension Using Simultaneous Dynamic Optimization". In: vol. 136. Jan. 2011. DOI: 10.1115/DETC2011-48521.

[3] Gabriel Bravo Palacios, Andrea Del Prete, and Patrick Wensing. "One Robot for Many Tasks: Versatile Co-Design Through Stochastic Programming". In: *IEEE Robotics and Automation Letters* PP (Jan. 2020), pp. 1–1. DOI: 10.1109/LRA.2020.2969948.

[4] Gianluigi Grandesso et al. "Exploring the limits of a hybrid actuation system through Co-Design". In: (July 2020). DOI: 10.13140/RG.2.2.28566.78400.

[5] Tad McGeer. "Passive Dynamic Walking". In: *The International Journal of Robotics Research* 9 (1990), pp. 62–82.

[6] Gabriele Buondonno et al. "Actuator Design of Compliant Walkers via Optimal Control". In: Sept. 2017. DOI: 10.1109/IROS.2017.8202228.

[7] Yevgeniy Yesilevskiy, Zhenyu Gan, and C. Remy. "Optimal configuration of series and parallel elasticity in a 2D Monoped". In: May 2016, pp. 1360–1365. DOI: 10.1109/ICRA.2016.7487269.

[8] Yevgeniy Yesilevskiy, Zhenyu Gan, and C. David Remy. "Energy-Optimal Hopping in Parallel and Series Elastic One-Dimensional Monopeds". In: *Journal of Mechanisms and Robotics* 10.3 (Apr. 2018). 031008. ISSN: 1942-4302. DOI: 10.1115/1.4039496. eprint: https://asmedigitalcollection.asme.org/mechanismsrobotics/article-pdf/10/3/031008/6403892/jmr_010_03_031008.pdf. URL: https://doi.org/10.1115/1.4039496.

[9] Stelian Coros et al. "Locomotion Skills for Simulated Quadrupeds". In: *ACM Trans. Graph.* 30 (July 2011), p. 59. DOI: 10.1145/2010324.1964954.

[10] Thomas Geijtenbeek, Michiel van de Panne, and A. Frank van der Stappen. "Flexible Muscle-Based Locomotion for Bipedal Creatures". In: *ACM Trans. Graph.* 32.6 (Nov. 2013). ISSN: 0730-0301. DOI: 10.1145/2508363.2508399. URL: https://doi.org/10.1145/2508363.2508399.

[11] Felix Grimminger et al. "An Open Torque-Controlled Modular Robot Architecture for Legged Locomotion Research". In: (2019). arXiv: 1910.00093 [cs.RO].

[12] W.J. Zhang, Q. Li, and L.S. Guo. "Integrated design of mechanical structure and control algorithm for a programmable four-bar linkage". In: *IEEE/ASME transactions on mechatronics* 4.4 (1999), pp. 354–362.

[13] Kevin Wampler and Zoran Popović. "Optimal Gait and Form for Animal Locomotion". In: *ACM SIGGRAPH 2009 Papers*. SIGGRAPH '09. New Orleans, Louisiana: Association for Computing Machinery, 2009. ISBN: 9781605587264. DOI: 10.1145/1576246.1531366. URL: https://doi.org/10.1145/1576246.1531366.

[14] Carlos Mastalli et al. "Crocoddyl: An Efficient and Versatile Framework for Multi-Contact Optimal Control". 2020.

[15] Roy Featherstone. "Gears". In: *Robot Dynamics Algorithms*. Boston, MA: Springer US, 2008. ISBN: 978-0-387-74315-8. DOI: 10.1007/978-0-387-74315-8_9. URL: https://doi.org/10.1007/978-0-387-74315-8.

[16] R. McN. Alexander. "The Gaits of Bipedal and Quadrupedal Animals". In: *The International Journal of Robotics Research* 3.2 (1984), pp. 49–59. DOI: 10.1177/027836498400300205.

[17] Miroslav Bogdanovic, Majid Khadiv, and Ludovic Righetti. "Learning Variable Impedance Control for Contact Sensitive Tasks". In: (July 2019). arXiv: 1907.07500 [robotics].

[18] Justin Carpentier et al. "The Pinocchio C++ library — A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives". In: Jan. 2019. DOI: 10.1109/SII.2019.8700380.

[19] Dario Izzo. "PyGMO and PyKEP: Open Source Tools for Massively Parallel Optimization in Astrodynamics (the case of interplanetary trajectory optimization)". In: Jan. 2012.

[20] Nikolaus Hansen. "The CMA evolution strategy: a tutorial". In: (Jan. 2010).

[21] Damien Marchal, Thomas Morzadec, and Christian Duriez. "Toward Shape Optimization of Soft Robots". In: Apr. 2019. DOI: 10.1109/ROBOSOFT.2019.8722822.

[22] M. Vulliez and S. Zeghloul. "Multi-objective design optimization of the Delthaptic, a new 6-DOF haptic device". In: *2016 IEEE 14th International Conference on Industrial Informatics (INDIN)*. 2016, pp. 248–253. DOI: 10.1109/INDIN.2016.7819167.