09/04/2024 17:46

(Article begins on next page)

On Designing Optimal Trajectories for Servo-Actuated Mechanisms: Detailed Virtual Prototyping and Experimental Evaluation

Marcello Pellicciari, Giovanni Berselli, Federico Balugani

Abstract—**Programmable servo-actuated mechanisms can enhance the flexibility and the reconfigurability of modern manufacturing systems. Differently from fully-mechanical design solutions (such as mechanical cams) and especially in the case of high-dynamic motions, servomechanisms performance depends on several interacting factors, namely electric motor and linkage dynamics, controller efficacy, and requested motion law. In particular, Point-To-Point (PTP) trajectories are usually designed in order to comply with technological constraints, imposed by the required interaction with the handled product, and to maximize some optimality criterion such as, for instance, energy efficiency or limited actuation torques. In this context, the present paper proposes a novel method for designing energy and peak-power optimal PTP motions. A standard optimization problem is solved by means of either cubic or quintic splines. Nonetheless, differently from previous approaches, the optimization cost functions are based on a virtual prototype of the system, which comprises behavioral models of power converter, controller and electric motor coupled with the mechanical system. Results are then compared with experimental data obtained on a physical prototype. The comparison quantitatively shows that better-behaved PTP trajectories can be designed by including the dynamic contribution of each sub-system component.**

Index Terms—**Virtual Prototyping, Trajectory Generation, Mechatronic Design Methods, Electronic Cams, High-speed Machinery, Intelligent Manufacturing.**

I. INTRODUCTION

Manufacturing flexibility and reconfigurability are the key challenges for production systems which have to dynamically adapt to ever changing process requirements [1]. In this context, even the most demanding tasks characterized by high motion dynamics and strict accuracy requirements are progressively being accomplished by means of Servo-Actuated Mechanisms (SAM). If compared to the extremely fast and precise movements achievable via fully-mechanical drives, such as cam systems, SAM offer the possibility to specifically re-configure/adapt/optimize Point-To-Point (PTP) motions for each different machine task (related, for instance, to multiple product sizes). On the other hand, the adoption of these programmable SAM introduces new engineering challenges mainly due to the potential position errors at high-dynamics, whose correction usually require intensive experimental tuning [2]. For what concerns SAM design and optimization, any action which aims at improving the system behavior should take into account that the overall system performance is the direct consequence of several interacting factors, namely mechanical system and electric motor dynamics, controller performance, power converter behavior and requested motion law. Nonetheless, at the state-of-the-art, SAM are in fact designed and developed with concurrent tools and methods, which are intrinsically separated and hardly integrated. For instance, the linkage mechanism is designed by mechanical engineers, in order to provide reliable solutions in terms of size, stiffness, weight and cost though considering idealized motion laws. In parallel, the actuator model is usually chosen by the control engineers who also develop practical trajectories and tune the control parameters in order to minimize potential tracking errors. Naturally, some portions of the motion law are not necessarily process-relevant and, therefore, they can be freely designed to a certain extent.

Concerning the design of these optimal motions, several approaches have been presented in the past literature, which are well addressed in multiple books [3] and surveys [4]. These techniques have been mainly applied in the field of industrial robotics and may be classified on the basis of employed *cost function*, *trajectory interpolation scheme*, *type of constraints*, and *optimization algorithm*.

As for the cost function, according to [5], the most significant optimality criteria are minimum execution time (i.e. time-optimal motions), minimum jerk, and minimum actuator effort. In particular, the latter case can be tackled via minimization of actuator torque or torque-change, or via optimization of either motor energy consumption or peak power. For example, time and jerk optimal PTP motions for serial robots are discussed in [6] and [7]. Minimum torque and torque change motions are discussed in [8] and [9], whereas energy-optimal trajectories have been proposed in [10], [11], [12]. Other researchers also considered multi-objective optimization such as time-jerk [13] or time-energy optimal [14] problems.

As for the trajectory interpolation scheme, two main strategies are found in the past literature, which are referred hereafter as *knot-based* and *S-curve-based* approach. In the knot-based approach, the motion profile is generated by interpolating a series of intermediate points called knots or control points. In this case, trigonometric, cubic, quintic and higher order polynomial splines have been extensively employed in the scientific literature. For instance, the problem of finding an optimal curve which interpolates a sequences of nodes in an industrial robot's joint space was firstly formalized in [15].

Corresponding author: Dr. Giovanni Berselli, email:giovanni.berselli@unimore.it

The same algorithm exploiting cubic B-spline motions was then presented in [16]. Similarly, the generation of smooth trajectories via quintic B-splines and trigonometric splines are reported in e.g. [13] and [17]. Alternatively, in the S-curve-based approach, the definition of knot points is not required. For example, a general algorithm for designing generalized polynomial S-curve trajectories is presented in [18].

As for the optimization constraints, according to [19], they can be classified into *system constraints* and *task constraints*. System constraints are related to the kinematic and dynamic limitations of the considered mechanism. Specific examples are maximum joint velocity, acceleration and jerk, bound on the joint angles, maximum payload and maximum torque. Task constraints are application-dependent requirements such as obstacle collision avoidance or specified path (e.g. in case of gluing, laser cutting, arc welding, etc.). For instance, time-jerk optimal and energy-optimal motions accounting for the physical limitations of the real manipulator are discussed in [5], [20], whereas model and methods to deal with obstacle avoidance and path tracking are discussed in [8].

Finally, for what concerns optimization algorithms, two main categories can be distinguished, namely *direct* and *indirect* methods [20]. Indirect methods are based on the Pontryagin Maximum Principle [21], whose main disadvantages are the small convergence domain and the difficulty in incorporating inequality constraints. Direct methods, as reported in [20], are based on a discretization of the dynamic variables (states, controls) which leads to a parameter optimization problem treated via numerical methods. Specific examples are non-linear optimization techniques such as Sequential Quadratic Programming [5], [20], Dynamic Programming [22], and Interval Analysis [23], or heuristic/meta-heuristic techniques such as Neural Networks [24], Genetic [10], and Multi-Objective Evolutionary Algorithms [25]. An interesting thorough comparison of the past literature, primarily based on the aforementioned classification can be found in [26], which also clearly highlights the benefits and limits of each method.

Despite the type of optimization technique, any chosen algorithm is based on a system model which computes the variable of interest on the basis of an input motion. For instance, in [27], torque-optimal paths are derived by considering the robots rigid-body dynamics but neglecting any contribution due to the actuation system. In [28], energy-optimal trajectories are computed on the basis of the power consumption due to the motors' resistive losses. Similarly, a simplified model of the servo-system based on rated parameters and also comprising inverter losses is employed in [29], whereas the influence of the control system on the feasibility of the final motion law is included in [30]. Nonetheless, each of the aforementioned algorithms might fail to provide a reliable (or even feasible) solution if an accurate prediction of the physical behaviour is not available. Owing to the aforementioned considerations, the purpose of the present paper is to provide a detailed virtual prototype of a single degree of freedom (d.o.f.) SAM to be effectively used for optimal trajectory planning. The proposed optimization technique is characterized by the following features:

- A standard Sequential Quadratic Programming algorithm is used for off-line numerical simulations (similarly to [5]).
- The motion profiles are generated via cubic or quintic splines interpolating a series of knot points whose position is varied by the optimization algorithm (similarly to [16]).
- Either the electric energy or the peak power directly drawn from the network are taken as cost functions.

Differently from all the previous literature, where one or more sub-system component was always neglected, the SAM model includes the dynamics of mechanical system, electric motor, and power converter (comprising inverter, rectifier and DC-bus). Also the detailed controller behaviour is accounted for, in order to embody possible position control errors as an optimization constraint and to check for the actual feasibility of the optimized trajectories. The simulation results are finally validated against experimental data showing that practical energy and peak-power optimal trajectories can be effectively obtained by means of the proposed approach.

II. SERVO-ACTUATED MECHANISM MODELLING

Permanent Magnet Synchronous Motors (PMSM) are today the de facto industry standard for position controlled servo-systems [31]. In high speed machinery, PMSM are usually connected to single-d.o.f. linkage mechanisms, commonly a Slider Crank Mechanism (SC) or a four-bar linkage. In particular, a conceptual scheme of the single-d.o.f. SAM considered in this paper is depicted in Fig. 1(a) and comprises a power converter (connected to a power source) which drives a PMSM connected to a SC (taken as a case study). The SC physical prototype is shown in Fig. 1(b). The converter comprises rectifier, servo inverter, and DC-link with drain resistor (also referred to as brake chopper).

In addition, Fig. 1(a) depicts both forward and backward power flows that have been conceptually decomposed into several terms, namely:

- total electrical power delivered to the system, P_{el}, to the inverter, P_{inv}, and to the PMSM, P_{pmsm};
- power, P_c, to charge/discharge the DC bus capacitance C;
- power flowing through the stator inductances, P_{dq};
- overall mechanical power delivered to the PMSM shaft, P_{mech}, and subsequently delivered to the SC mechanical structure and external user, P_{scm}.

In parallel the overall power loss can be described by the following terms:

- power loss, P_f^L, due to mechanical friction in the SC structure;
- electric power loss in the PMSM composed of copper, P_{Cu}^L, and iron, P_{Fe}^L, losses.
- electric power loss in the inverter stage composed of:
  ○ load-independent losses, P_{li}^L, due to permanent power requirements of the control electronics;
  ○ conduction, P_{cd}^L, and switching, P_{sw}^L, losses in the inverter IGBTs (see e.g. [29], [32]);

- electric power loss, P_D^L, due to the activation of the drain resistor, R_D.

In the following, with obvious notation of symbols, generic actual values are referred to as u, whereas reference (desired) values are written as u^*. Note that, owing to the high-precision requirements, only direct-drive solutions will be considered (absence of gear reducer).

A. Modeling of the rigid-body mechanism dynamics

With reference to Fig. 1(a), let one define ϑ_c, ϑ_r as the crank and connecting rod angular positions measured with respect to the slider direction of motion, y_c, y_r, y_s as the y-axis (vertical) center of gravities' coordinates of crank, rod and slider respectively, x_r as the x-axis (horizontal) center of gravity coordinate of the crank. Given the slider position profile, $y_s(t)$, as function of time t, the crank angle profile can be found via simple position analysis as:

$$\vartheta_c(t) = \arccos\left[(y_s^2 + c^2 - r^2)(2y_s c)^{-1}\right] \quad (1)$$

where c and r are crank and rod lengths. Resorting to the chain rule and choosing ϑ_c as the mechanism input coordinate, the links' velocities which are relevant to the following calculations can be computed as follows:

$$\dot{x}_r(t) = \partial x_r/\partial \vartheta_c \, \dot{\vartheta}_c \qquad \dot{y}_r(t) = \partial y_r/\partial \vartheta_c \, \dot{\vartheta}_c$$
$$\dot{\vartheta}_r(t) = \partial \vartheta_r/\partial \vartheta_c \, \dot{\vartheta}_c \qquad \dot{y}_s(t) = \partial y_s/\partial \vartheta_c \, \dot{\vartheta}_c \quad (2)$$

Recalling that only non process-relevant portions of the overall motion law are considered, the only external force acting on the SC is due to the weight and inertia of a possible payload attached to the SC slider. Let one then denote m_c, m_r, m_s, m_e as the masses of crank, rod, slider and payload respectively, $J_{c,O}$ as the moment of inertia of the crank along its rotation axis, $J_{r,G}$ as the moment of inertia of the rod along its center of gravity. Following the procedure described in [33, p. 113],

the mechanism kinetic energy, W_k, is expressed as:

$$W_k = \frac{1}{2}\left[J_{c,O}\dot{\vartheta}_c^2 + m_r\left(\dot{x}_r^2 + \dot{y}_r^2\right) + J_{r,G}\dot{\vartheta}_r^2 + (m_s + m_e)\dot{y}_s^2\right] \quad (3)$$

$$= \frac{1}{2}J_{red}\dot{\vartheta}_c^2 \quad (4)$$

In particular, Eq. 4 is found by expressing the links' velocities in Eq. 3 by means of the formulations from Eq. 2, and by defining the reduced moment of inertia, J_{red}, as:

$$J_{red} = J_{c,O} + m_r\left[(\partial x_r/\partial \vartheta_c)^2 + (\partial y_r/\partial \vartheta_c)^2\right] + \quad (5)$$
$$+ J_{r,G}(\partial \vartheta_r/\partial \vartheta_c)^2 + [m_s + m_e](\partial y_s/\partial \vartheta_c)^2$$

It is worth noting that $J_{red} = J_{red}(\vartheta_c)$ is written as function of the crank angle, ϑ_c, and not as function of time t. As for the potential energy due to gravity, W_p, the following expression holds:

$$W_p = g[m_c y_c + m_r y_r + (m_s + m_e)y_s] \quad (6)$$

where g is the acceleration of gravity.

For what concerns dissipative effects due to friction, crank friction torque, $\tau_{f,c}$, and slider friction force, $F_{f,s}$, are modelled using the following effective formulations:

$$\tau_{f,c} = \mu_c \, sign\left(\dot{\vartheta}_c\right) + v_c \, \dot{\vartheta}_c \quad (7)$$

$$F_{f,s} = \mu_s \, sign\left(\dot{y}_s\right) + v_s \, \dot{y}_s \quad (8)$$

where μ_c and v_c are crank Coulomb and viscous friction coefficients, whereas μ_s and v_s are slider Coulomb and viscous friction coefficients. Friction torques on the passive rotational joints, namely crank-rod and rod-slider joints, are neglected for application purposes. The force $F_{f,s}$ and the torque $\tau_{f,c}$ are then referred to the mechanism input coordinate ϑ_c by imposing that the overall power related to all frictional loads must be equal to the power related to a unique (and fictitious) frictional torque τ_f. Thus:

$$\tau_f \dot{\vartheta}_c = \tau_{f,c}\dot{\vartheta}_c + F_{f,s}\dot{y}_s \quad (9)$$
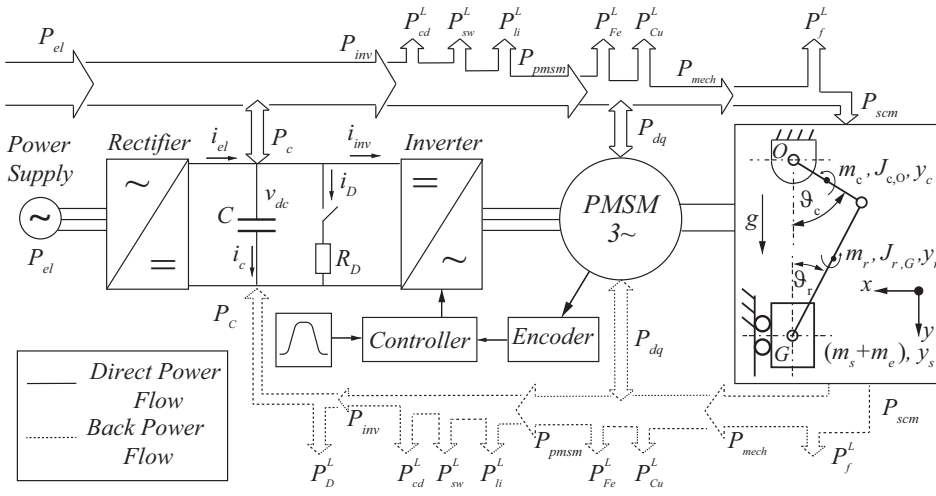
(a) Servo-actuated mechanism schematic.

(b) Slider-crank mechansim prototype.

Fig. 1. Schematic of a single d.o.f. servo-system (a) and slider-crank mechanism prototype (b). The continuous arrow depicts the direct (forward) power flow whereas the dashed arrow depicts the backward power flow.
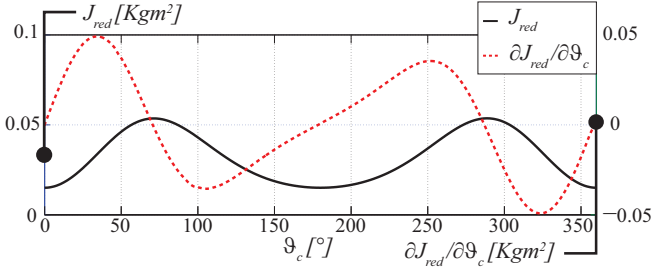
Fig. 2. Values of J_{red} (leftmost y-axis) and $\partial J_{red}/\partial \vartheta_c$ (rightmost y-axis) as functions of ϑ_c.

Hence, by substituting Eqs. 7 and 8 in Eq. 9, and isolating the frictional torque τ_f, leads to the following expression:

$$\tau_f = \tau_{f,c} + F_{f,s}\dot{y}_s\dot{\vartheta}_c^{-1} \tag{10}$$
$$= sign\left(\dot{\vartheta}_c\right)\left[\mu_c + \mu_s\,sign\left(\partial y_s/\partial \vartheta_c\right)\right] + \dot{\vartheta}_c\left[\nu_c + \nu_s\left(\partial y_s/\partial \vartheta_c\right)\right]$$

As for the PMSM output torque, τ_m, it can be computed resorting to the Lagranges equation of the second kind, such that:

$$\tau_m = \frac{d}{dt}\left(\frac{\partial W_k - W_p}{\partial \dot{\vartheta}_c}\right) - \frac{\partial W_k - W_p}{\partial \vartheta_c} + \tau_f \tag{11}$$

Substituting Eqs. 4 and 6 in Eq. 11 and differentiating (see [33, p. 116] for details) leads to the SC equation of motion, which can be written as:

$$\tau_m = J_{red}\ddot{\vartheta}_c + \frac{1}{2}\frac{\partial J_{red}}{\partial \vartheta_c}\dot{\vartheta}_c^2 + \frac{\partial W_p}{\partial \vartheta_c} + \tau_f \tag{12}$$

Finally, Eq. 12 can be recast as follows:

$$\ddot{\vartheta}_c = J_{red}^{-1}\left[\tau_m - \frac{1}{2}\frac{\partial J_{red}}{\partial \vartheta_c}\dot{\vartheta}_c^2 - \frac{\partial W_p}{\partial \vartheta_c} - \tau_f\right] = f_{dyn}(\tau_m, \dot{\vartheta}_c, \vartheta_c) \tag{13}$$

where the highest order derivative $\ddot{\vartheta}_c$ has been isolated and the function f_{dyn} has been defined for the purpose of the following sections.

In particular, once SC dimensions, inertial and friction properties are known, the quantities J_{red}, W_{pot}, and τ_f can be computed by means of either general-purpose MBD software [34], whose usage can effectively speed-up time consuming calculation even in case of very simple closed-loop mechanisms [35], or through kinematic analysis following well known procedures outlined, for instance, in [33]. Figure 2 reports the values of J_{red} and $\partial J_{red}/\partial \vartheta_c$ as functions of ϑ_c for
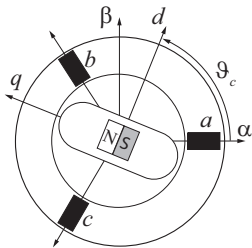
the considered SC, highlighting the high variability of both parameters across the SC range of motion. Note that, given the desired profiles for $\vartheta_c^*(t)$, $\dot{\vartheta}_c^*(t)$, $\ddot{\vartheta}_c^*(t)$ as function of time, t, and supposing negligible tracking errors (ideal controller), Eq. 12 allows to compute a reference value of the PMSM torque, τ_m^*. Nonetheless, as shown in [34] and also highlighted in the following sections, such simplified modeling procedure fails to provide accurate results when high dynamic motions are considered.

As for the mechanical power flow, as schematized in Fig. 1(a), the overall mechanical power delivered to the SC is simply given by:

$$P_{mech} = P_{scm} + P_f^L \tag{14}$$

The friction power loss, P_f^L, can be computed from Eq. 10 as:

$$P_f^L = \tau_f\dot{\vartheta}_c =$$
$$= sign\left(\dot{\vartheta}_c\right)\dot{\vartheta}_c\left[\mu_c + \mu_s\,sign\left(\partial y_s/\partial \vartheta_c\right)\right] + \dot{\vartheta}_c^2\left[\nu_c + \nu_s\left(\partial y_s/\partial \vartheta_c\right)\right] \tag{15}$$

The portion of mechanical power counteracting gravity and inertial load, P_{scm}, can be computed from Eqs. 10 and 12 as:

$$P_{scm} = (\tau_m - \tau_f)\dot{\vartheta}_c = \left[J_{red}\ddot{\vartheta}_c + \frac{1}{2}\frac{\partial J_{red}}{\partial \vartheta_c}\dot{\vartheta}_c^2 + \frac{\partial W_p}{\partial \vartheta_c}\right]\dot{\vartheta}_c \tag{16}$$

B. Modeling of PMSM dynamics

With reference to Fig. 3, recalling basic modelling procedure of industrial servo-motors [36], the PMSM behaviour can be described in either static three-phase (*abc-axis*) or two-phase ($\alpha\beta$-*axis*) coordinate systems attached to the PMSM stator, or in a rotating two-phase reference system (*dq-axis*) attached to the PMSM rotor. Accordingly, by considering the well known Park's theory [37], the equivalent PMSM circuits in the *dq-axis* can be schematized as in Fig. 4. In particular, similarly to [38], this accurate PMSM dynamic model have been defined assuming linear magnetic circuit, negligible leakage inductances (as in [39]), and a sinusoidal flux density distribution in the air-gap. In addition, iron losses are modelled by means of a resistive term, R_c, inserted in the traditional PMSM equivalent circuit (described, for instance, in [40]).

With reference to Fig. 4, let one define L_d and v_d^* as the inductance and reference voltage referred to the direct axis (*d-axis*), L_q and v_q^* as the inductance and reference voltage referred to the quadrature axis (*q-axis*), R as the stator resistance, R_c as the core resistance, p as the number of PMSM pole pairs, K_t as

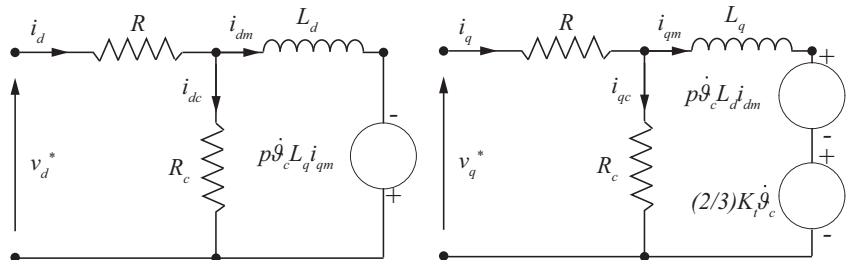Fig. 3. *abc-axis*, $\alpha\beta$-*axis*, and *dq-axis* frames.

Fig. 4. Dynamic *d-q* axis equivalent circuit of a PMSM.

the motor torque constant, i_d, i_{dc} and i_{dm} as the d-axis current components flowing through R, R_c and L_d respectively, i_q, i_{qc} and i_{qm} as the q-axis current components flowing through R, R_c and L_q respectively.

Resorting to the Kirckoff's circuit laws, the state-space equations of the dynamic system can be computed as:

$$\frac{di_{dm}}{dt} = \frac{R_c}{L_dR_c + RL_d}\left[v_d^* - Ri_{dm} + p\dot{\vartheta}_c i_{qm}L_q\frac{R_c + R}{R_c}\right]$$
$$= f_d(v_d^*, i_{dm}, i_{qm}, \dot{\vartheta}_c) \qquad (17)$$

$$\frac{di_{qm}}{dt} = \frac{R_c}{L_qR_c + RL_q}\left[v_q^* - Ri_{qm} - (2/3K_t + pL_d i_{dm})\dot{\vartheta}_c\frac{R_c + R}{R_c}\right]$$
$$= f_q(v_q^*, i_{dm}, i_{qm}, \dot{\vartheta}_c) \qquad (18)$$

Similarly, the current components i_d and i_q can be computed as:

$$i_d = i_{dm} + \frac{-p\dot{\vartheta}_c L_q i_{qm} + L_d di_{dm}/dt}{R_c} \qquad (19)$$
$$= f_{nd}(i_{dm}, i_{qm}, di_{dm}/dt)$$

$$i_q = i_{qm} + \frac{\dot{\vartheta}_c(2/3K_t + pL_d i_{dm}) + L_q di_{qm}/dt}{R_c} \qquad (20)$$
$$= f_{nq}(i_{dm}, i_{qm}, di_{qm}/dt)$$

Once the d-axis and q-axis current components are known, the PMSM output torque can be calculated as:

$$\tau_m = K_t i_{qm} + 3/2p(L_d - L_q)i_{dm}i_{qm} = f_m(i_{qm}, i_{dm}) \qquad (21)$$

In addition, the PMSM input power, P_{pmsm}, is given by:

$$P_{pmsm} = \frac{3}{2}\left[v_q^* i_q + v_d^* i_d\right] = f_{pmsm}(i_d, i_q) \qquad (22)$$

In particular, as depicted in Fig. 1, the term P_{pmsm} can be decomposed as:

$$P_{pmsm} = P_{mech} + P_{Cu}^L + P_{Fe}^L + P_{dq} \qquad (23)$$

where the power flowing through the stator inductances, P_{dq}, the copper losses, P_{Cu}^L, and the iron losses, P_{Fe}^L, can be respectively computed as:

$$P_{dq} = \frac{3}{2}\left[L_d i_{dm}\frac{di_{dm}}{dt} + L_q i_{qm}\frac{di_{qm}}{dt}\right] \qquad (24)$$

$$P_{Cu}^L = \frac{3}{2}R\left[i_d^2 + i_q^2\right] \qquad (25)$$

$$P_{Fe}^L = \frac{3}{2}R_c\left[i_{dc}^2 + i_{qc}^2\right] \qquad (26)$$

Note that: a) if necessary, the temperature dependence of both torque constant, K_t, and stator resistance, R, can be accounted for employing simple linear relations detailed in e.g. [41]; b) the factor $3/2$ in Eqs. 21, 22, 24, 25, and 26 is due to the Park transformation [37] which refers electrical quantities to the rotating d-q frame; c) the functions f_d, f_q, f_{nd}, f_{nq}, f_m and f_{pmsm} in Eqs. 17, 18, 20, **??**, 21, and 22 respectively, have been defined for the purpose of the following sections.

C. Control architecture

The PMSM controller, acting on the DC/AC module of the power converter, is usually based on a cascade feedback structure with a fast inner loop for current control, and outer closed-loops for speed and position control. This scheme improves the robustness of the controller with respect to unavoidable modeling errors and disturbances. In particular, PMSM vector control allows the implementation of several strategies, while maintaining control over the motor torque. Referring to the terminology proposed in [42], which also provides a through description and a comparison of each technique, possible control strategies are zero d-axis current, unity power factor, constant mutual flux linkages, maximum torque per unit current, and maximum efficiency. In the following, due to its wide spread use in the industrial scenario, the zero d-axis current control is implemented, whose block diagram schematic is depicted in Fig. 5. In this control system, stator currents are measured and then referred to the dq-axis frame through the Clarke (abc-axis to $\alpha\beta$-axis) and the Park ($\alpha\beta$-axis to dq-axis) transformations [36]. The current values i_d and i_q are used as the negative feedback quantity of the electric current loop which computes the reference voltage values v_d^* and v_q^*. These voltages are transformed into stator voltages through the inverse Park and Clark transformation and used to produce PWM signal to control the inverter.

The model block diagram concerning PMSM dynamics and control architecture coupled with SC dynamics is depicted in Fig. 6, highlighting the mutual relations among the variables described in the previous sections. Note that, for simplicity, the block diagram reports continuous transfer functions whereas sampled digital control laws are implemented in the simulator. In particular, the effects of the computation delay, the discretization of the integral/derivative terms, and the sampling time of real controllers are introduced in the model (implying that the current loop sampling rate is faster if compared to the velocity one, the same argument applying to the velocity loop with respect to the position loop). In particular, with reference to Fig. 6, $R_{PI}^{dq}(s)$ refers to the Proportional-Integral (PI) control loops for d-axis and q-axis currents, whereas $R_{PI}^v(s)$ refers to the velocity PI control. Henceforth, current and velocity
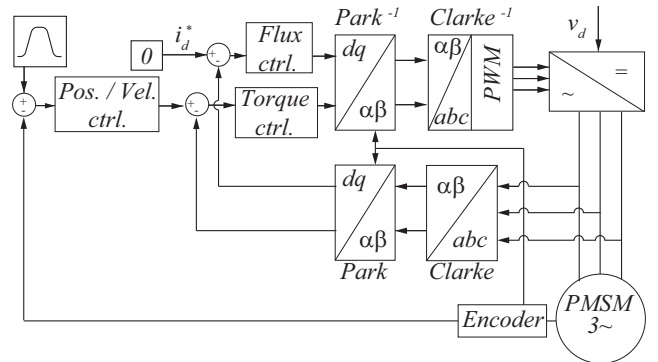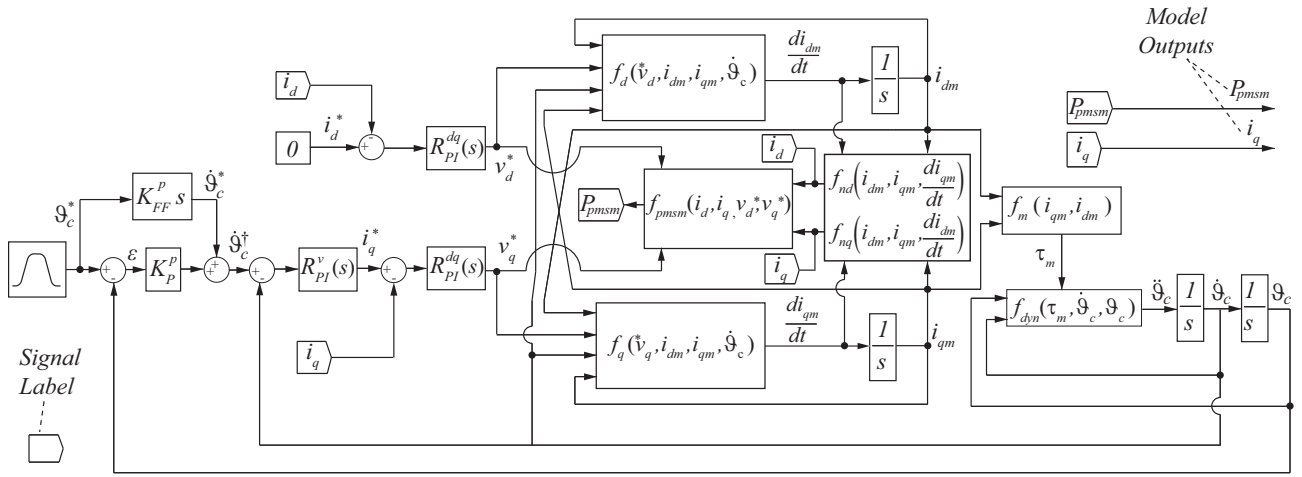

Fig. 5. Zero d-axis current control.

Fig. 6. Block diagram of control architecture, PMSM dynamics and SC dynamics. Signal labels on P_{pmsm}, i_q, i_d are introduced for clarity. The functions f_{dyn}, f_d, f_q, f_{nd}, f_{nq}, f_m and f_{pmsm} are defined in Eqs. 13, 17, 18, 20, **??**, 21 and 22.

controllers can be expressed as:

$$v_d^* = K_P^{dq}(i_d^* - i_d) + K_I^{dq} \int (i_d^* - i_d)dt \text{ where } i_d^* = 0 \quad (27)$$

$$v_q^* = K_P^{dq}(i_q^* - i_q) + K_I^{dq} \int (i_q^* - i_q)dt$$

$$i_q^* = K_P^v(\dot{\vartheta}_c^\dagger - \dot{\vartheta}_c) + K_I^v \int (\dot{\vartheta}_c^\dagger - \dot{\vartheta}_c)dt$$

where a traditional control strategy requiring $i_d^* = 0$ has been implemented [38], whereas K_P^{dq}, K_I^{dq}, K_P^v and K_I^v are the proportional and integral gains of the current and velocity controllers respectively. As for the position control loop, it comprises a proportional term with velocity feed-forward, such that:

$$\dot{\vartheta}_c^\dagger = K_P^p \varepsilon + K_{FF}^p \dot{\vartheta}_c^* \quad (28)$$

where K_P^p and K_{FF}^p are proportional and feed-forward gains of the position controllers, whereas $\varepsilon = \vartheta_c^* - \vartheta_c$ is the position tracking error. Given the desired motion law ($\vartheta_c^*(t)$, $\dot{\vartheta}_c^*(t)$, $\ddot{\vartheta}_c^*(t)$), the PMSM and the SC dynamics can be coupled to the controller model in order to provide a better assessment of the motor torque.

D. Electrical drive model

A conceptual scheme of the electrical drive, comprising inverter, rectifier and DC bus, is depicted in Fig. 1(a). In most cases, unless the electric drive system is equipped with a regenerative module (active front end [36]), the energy flow towards the AC network is prevented and the electrical power input is unidirectional (i.e. $P_{el} \geq 0$). Therefore, when the PMSM operates in generator mode (i.e. $P_{inv} \leq 0$), the mechanical energy is converted back into electrical energy and stored in the DC bus capacitor C (see Fig. 1(a)), whose voltage increases. Possible excessive over-voltages are then dissipated in a drain resistor, R_D, which is activated when needed.
In order to correctly compute the total electrical power delivered to system, P_{el}, it is necessary to numerically evaluate the DC bus voltage, v_{dc}, and its variation during PMSM

functioning, along with the inverter power P_{inv}. The calculation is based on a recursive differential model, the derivatives being approximated by finite differences and Δt indicating the integration time step. In the following, the sub-index $j-1$ for any variable refers to the value of the previous integration step, the sub-index j (current integration step) being omitted for clarity.
With reference to Fig. 1(a), let us firstly compute the inverter input power, P_{inv}, as follows:

$$P_{inv} = P_{pmsm} + P_{li}^L + P_{cd}^L + P_{sw}^L \quad (29)$$

where, as said, the term P_{li}^L models load-independent losses due to permanent power requirements of the control electronics, the term P_{cd}^L models conduction losses assumed proportional to the square of the q-axis motor current i_q, the term P_{sw}^L models switching losses in the inverter IGBTs [32] assumed proportional to the absolute value of the inverter input current $i_{inv,[j-1]}$ (computed at the previous integration step). Finally, the term P_{pmsm} is simply computed from Eq. 23. Therefore, defining the constant factors R_{inv} and K_{igbt}, the following relations hold:

$$P_{cd}^L = R_{inv} i_q^2 \quad (30)$$

$$P_{sw}^L = K_{igbt} |i_{inv,[j-1]}| = K_{igbt} \left| \frac{P_{inv,[j-1]}}{v_{dc,[j-1]}} \right| \quad (31)$$

Equations 29 and 31 highlight that the power delivered to the inverter, P_{inv}, at the current integration step depends on the DC-bus voltage, $v_{dc,[j-1]}$ computed at the previous integration step.
Once P_{inv} is known, it is possible to compute v_{dc} at the current integration step. Let one reasonably assume that the AC/DC converter is an ideal three-phase, full-bridge rectifier and no rectifier energy losses are accounted for (i.e. $P_{el} = i_{el}v_{dc}$, see Fig.1(a)). In a fictitious case where the DC-bus capacitance is absent [36] and the electrical load is purely resistive, the rectifier ideally generates a voltage signal which oscillates
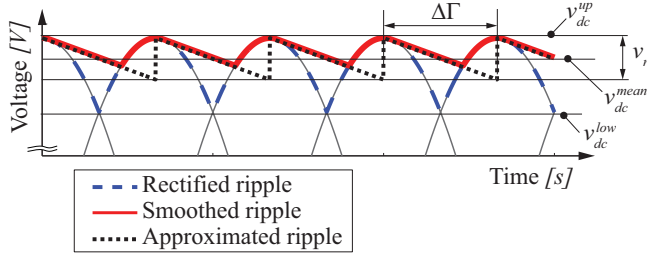
(a) DC-bus voltage ripple at constant electrical load.
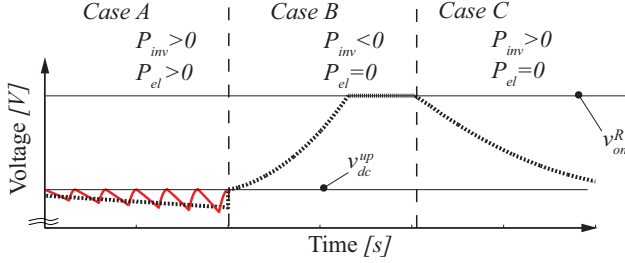
(b) DC-bus voltage variation for dynamic loading.

Fig. 7. Qualitative behaviour of the DC-bus voltage.

between a lower and an upper value. These values can be respectively computed as $v_{dc}^{low} = \sqrt{1.5}v_{ll}$ and $v_{dc}^{up} = \sqrt{2}v_{ll}$, having defined v_{ll} as the RMS line-to-line supply voltage of the AC network. This condition is qualitatively depicted by the blue dashed line in Fig. 7(a). The voltage ripple, v_r, is simply computed as $v_r = v_{dc}^{up} - v_{dc}^{low}$, whereas the time period between two voltage peaks is $\Delta\Gamma = (6f)^{-1}$, namely a characteristic constant for the considered rectifier type, f being the AC network frequency. Nonetheless, in the real case, the DC-bus capacitor, C, is rapidly charged when the diodes are closed (i.e. in conduction mode) and it supplies energy to the load when the diodes are opened (i.e. non conduction mode), resulting in an overall voltage ripple reduction. This condition is qualitatively depicted by the continuous red line in Fig. 7(a). Having recalled this basic remarks, let one assume that:

• the DC-bus voltage linearly decreases when the capacitor supplies energy to the load, i.e. the current, i_{inv}, is constant in the considered time frame;

• the DC-bus voltage instantly increase to the value v_{dc}^{up} in correspondence to each rectified voltage peak.

This condition is qualitatively depicted by the dotted line in Fig. 7(a), the voltage ripple being simply estimated as $v_r = Q/C$, where Q is the charge drawn from the capacitor during the time $\Delta\Gamma$.

Following from the aforementioned considerations, without loss of generality, let one assume that the DC-bus voltage, $v_{dc}(0)$, at the time $t = 0$, is approximated by the maximum line voltage value v_{dc}^{up}. Starting from this initial condition, three possible scenarios are possible, which are also conceptually depicted in Fig 7(b):

• *Case A:* the SC accelerates, the power requirement from the inverter is positive (i.e. $P_{inv} > 0$) and electrical power is taken

from the network. In this case, the term P_{el} is assumed equal to the term P_{inv}, namely $P_{el} = P_{inv}$. In parallel, a load-dependent voltage ripple is present, which can be approximated as:

$$v_r = \frac{Q}{C} = \frac{i_{inv,[j-1]}\Delta\Gamma}{C} = \frac{i_{inv,[j-1]}}{6fC} \quad (32)$$

Hence, the DC-bus voltage, v_{dc}, along with actual energy, E_C, stored in the capacitor can be approximated as:

$$v_{dc} = v_{dc}^{mean} \equiv v_{dc}^{up} - v_r/2 \quad (33)$$

$$E_C = 0.5Cv_{dc}^2 \quad (34)$$

The resulting DC-bus voltage variation is qualitatively depicted in the leftmost part of Fig. 7(b), the continuous red line and the dotted line respectively representing the real voltage oscillations and the approximation obtained via Eq. 33. In addition, the associated energy flow (direct flow) is depicted in Fig.1(a) by the solid line arrow and the corresponding electric drive state is referred to as *loading condition*. • *Case B:* the SC decelerates up to a point where the PMSM operates in generator mode. In this case, the power requirement from the inverter is negative (i.e. $P_{inv} < 0$) but no electrical power is exchanged with the network (i.e. $P_{el} = 0$) since a rectifier which prevents any power backflow is used (passive front end). Therefore, the inverter power is stored in the DC-bus capacitance whose voltage increases. Due to the over-voltage protection, some of the energy is dissipated in the drain resistor, R_D, if a voltage upper limit, v_{on}^R, is reached. In particular, the brake chopper is regulated such that the power lost in the drain resistor, P_D^L equals the inverted power, P_{inv}, until the DC-bus voltage drops below v_{on}^R, that is:

$$P_D^L = \begin{cases} -P_{inv} & \text{if } v_{dc,[j-1]} \geq v_{on}^R \text{ and } P_{inv} < 0 \\ 0 & \text{otherwise.} \end{cases} \quad (35)$$

Resorting to the power balance on the DC-bus, it is now possible to firstly calculate the power stored in the DC bus capacitor as $P_c = -P_{inv} - P_D^L$. Subsequently, DC-bus energy storage, E_C, and voltage, v_{dc} can be computed as:

$$E_C = E_{C,[j-1]} + (-P_{inv} - P_D^L)\Delta t \quad (36)$$

$$v_{dc} = \sqrt{2E_C/C} \quad (37)$$

The resulting DC-bus voltage variation is qualitatively depicted in the central part of Fig. 7(b), the associated energy backflow is depicted in Fig.1(a) by the dashed line arrow and the corresponding electric drive state is referred to as *recuperation*, as long as useful energy is stored in the capacitor.

• *Case C:* starting from a condition where $v_{dc} \geq v_{dc}^{up}$, the PMSM may restart to operate in motor mode and the power requirement from the inverter becomes positive (i.e. $P_{inv} > 0$). Nonetheless, the DC-bus over-voltage does not allow for the conduction mode of the rectifier diodes and no electrical power is drawn from the network (i.e. $P_{el} = 0$ until v_{dc} drops below v_{dc}^{up}). Therefore, Eqs. 36 and 37, which model the recuperation condition, still apply. The resulting DC-bus voltage variation is qualitatively depicted in the rightmost part of Fig. 7(b).
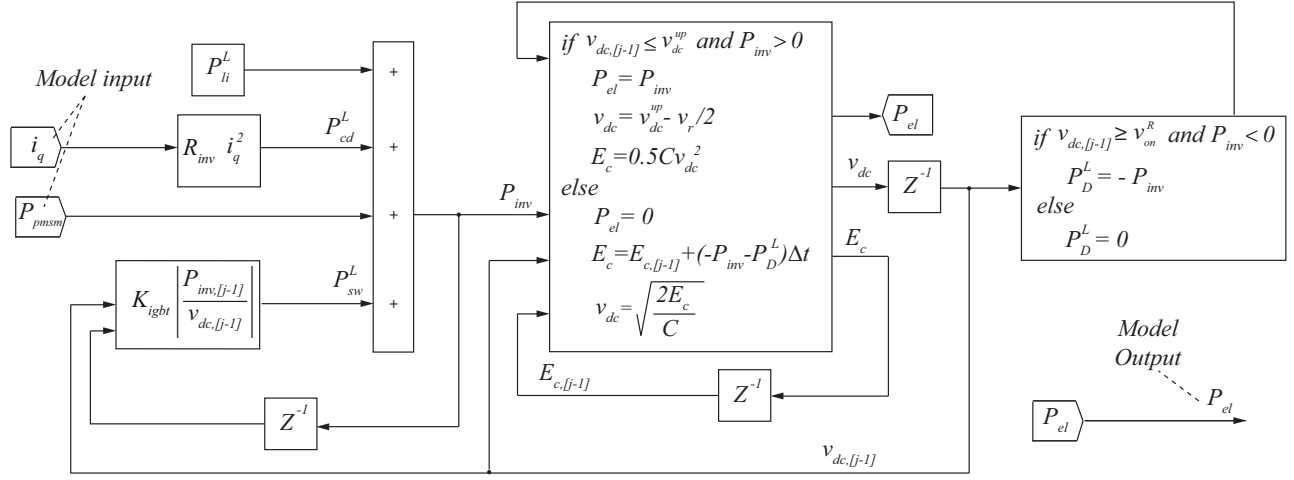
Fig. 8. Block diagram of the electrical drive power model. The block Z^{-1} depicts a unit delay.

Owing to the aforementioned consideration, the grid power requirement, P_{el}, can be computed as follows:

$$
\begin{aligned}
\left.
\begin{array}{l}
P_{el} = P_{inv} \\
v_{dc} = v_{dc}^{up} - v_r/2 \\
E_c = \frac{1}{2}Cv_{dc}^2
\end{array}
\right\} & \text{ if } v_{dc,[j-1]} \leq v_{dc}^{up} \text{ and } P_{inv} > 0 \\
\\
\left.
\begin{array}{l}
P_{el} = 0 \\
E_c = E_{C,[j-1]} + (-P_{inv} - P_D^L)\Delta t \\
v_{dc} = \sqrt{\frac{2E_c}{C}}
\end{array}
\right\} & \text{ otherwise }
\end{aligned}
\tag{38}
$$

Finally, the block diagram concerning the aforementioned algorithm is reported in Fig. 8. As previously said, the knowledge of the power requirement P_{el} from Eq. 38 can be used for designing energy optimal trajectories as explained in the following section.

III. OPTIMIZATION APPROACH

With reference to Fig. 9, a generic position profile $\vartheta_c(t)$ can be conceptually divided in areas where strict constraints are imposed by the process requirements and areas which are not process-relevant. In the latter case, it is possible to design optimal trajectories on the basis of some relevant cost function. Let one consider, for instance, the forward ($\dot{\vartheta}_c > 0$) PTP crank motion $\vartheta_c(t)$ for $t \in [0, T]$, T being the final forward motion time.

Starting from a simple fifth order polynomial, that is the Root-Mean-Square (RMS) jerk-optimal PTP motion [7], the pro-

Fig. 9. Generic position profile.

posed optimization approach is based on either cubic or quintic splines which interpolate a series of *virtual* points within the interval of interest (naturally, also other approaches are possible, e.g. [10]). The position of these breakpoints is varied by the optimization algorithm in order to provide optimal motion laws. In particular, the time span $t \in [0, T]$ is divided into $n \in \mathbb{N}$ equi-spaced subintervals, n being a user-defined variable chosen such that $n > 3$. The vector of breakpoints (including the PTP extremities) and the corresponding *virtual* crank positions are defined as:

$$
\boldsymbol{\xi} = [\xi_i]^T = [iT/n]^T, \quad \mathbf{q} = [q_i]^T, \quad i = 0, \ldots, n
\tag{39}
$$

where $q_0 = \vartheta_c(0)$ and $q_n = \vartheta_c(T)$. The corresponding piecewise polynomial (the so-called *pp-form* spline [43]) is defined as:

$$
\vartheta_c(\mathbf{q}, t) \equiv \left\{ \mathscr{P}_k(\mathbf{q}, t) = \sum_{j=0}^{m} D_{j,k}(\mathbf{q})(t - \xi_k)^j, \right.
$$
$$
\left. \xi_k \leq t \leq \xi_{k+1}, \ k = 0, \ldots, n-1 \right\}
\tag{40}
$$

where $D_{j,k}$ are the local polynomial coefficients. Equation 40 describes either a cubic splines with $4n$ unknown coefficients if $m = 3$, or a quintic spline with $6n$ unknown coefficients if $m = 5$. In addition, the same equation simplifies into a single polynomial if $n = 1$.

In the case of cubic splines, the number of unknowns can be reduced by imposing the following constraints:

○ $3(n-1)$ conditions for the continuity of positions, velocities and accelerations at the intermediate breakpoints, as:

$$
\begin{aligned}
\mathscr{P}_k(\mathbf{q}, \xi_{k+1}) &= \mathscr{P}_{k+1}(\mathbf{q}, \xi_{k+1}); \\
\dot{\mathscr{P}}_k(\mathbf{q}, \xi_{k+1}) &= \dot{\mathscr{P}}_{k+1}(\mathbf{q}, \xi_{k+1}); \\
\ddot{\mathscr{P}}_k(\mathbf{q}, \xi_{k+1}) &= \ddot{\mathscr{P}}_{k+1}(\mathbf{q}, \xi_{k+1}); \quad \text{for } k = 0 \ldots n-2
\end{aligned}
\tag{41}
$$

○ 6 conditions for imposing positions, velocities and accelerations at the endpoints, that is:

$$
\begin{aligned}
\mathscr{P}_0(\mathbf{q}, 0) &= q_0; & \dot{\mathscr{P}}_0(\mathbf{q}, 0) &= 0; & \ddot{\mathscr{P}}_0(\mathbf{q}, 0) &= 0; \\
\mathscr{P}_{n-1}(\mathbf{q}, T) &= q_n; & \dot{\mathscr{P}}_{n-1}(\mathbf{q}, T) &= 0; & \ddot{\mathscr{P}}_{n-1}(\mathbf{q}, T) &= 0;
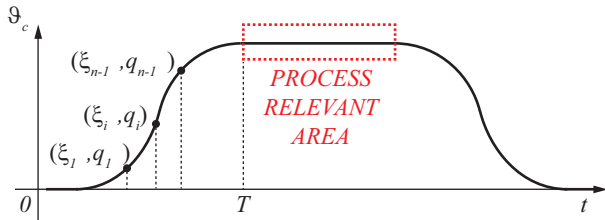\end{aligned}
\tag{42}
$$

In the case of quintic splines, the number of unknowns can be reduced by imposing the following constraints:

○ $3(n-1)$ conditions as in Eq. 41, together with $2(n-1)$ conditions for the continuity of jerks and quirks at the intermediate breakpoints, as:

$$\dddot{\mathscr{P}}_k(\mathbf{q},\xi_{k+1}) = \dddot{\mathscr{P}}_{k+1}(\mathbf{q},\xi_{k+1});$$
$$\ddddot{\mathscr{P}}_k(\mathbf{q},\xi_{k+1}) = \ddddot{\mathscr{P}}_{k+1}(\mathbf{q},\xi_{k+1}); \quad \text{for } k=0\ldots n-2 \tag{43}$$

○ 6 conditions as in Eq. 42, together with 2 conditions for imposing the jerk values at the endpoints, that is:

$$\dddot{\mathscr{P}}_0(\mathbf{q},0) = 0; \quad \dddot{\mathscr{P}}_{n-1}(\mathbf{q},T) = 0; \tag{44}$$

Owing to the aforementioned constraints, $n-3$ degrees of freedom are kept available for both cubic and quintic splines. Therefore, only a subset of the parameter vector $\mathbf{q}$, whose dimension is n, can be considered for optimization purposes. In particular, similarly to [5], the position of breakpoints q_1 and q_{n-1} is constrained when imposing either accelerations at the endpoints in the case of cubic splines, or jerks at the endpoints in the case of quintic splines. As long as q_0 and q_n, are given, the optimization parameter vector is finally defined as:

$$\mathbf{q}_{red} = [q_2,\ldots,q_{n-2}]^T \subset \mathbf{q} \tag{45}$$

Hence, the optimal trajectory planning problem is formulated as a minimization of the cost function, $F(\mathbf{q}_{red})$, by variation of the parameter vector, $\mathbf{q}_{red}$:

$$\min_{\mathbf{q}_{red}} F(\mathbf{q}_{red}), \quad \text{subject to:} \tag{46}$$

$$|\dot{\vartheta}_c(t)| < \dot{\vartheta}_c^{max} \quad |\ddot{\vartheta}_c(t)| < \ddot{\vartheta}_c^{max} \quad |\dddot{\vartheta}_c(t)| < \dddot{\vartheta}_c^{max} \tag{47}$$

$$|\tau_m(t)| < \tau_m^{max} \tag{48}$$

$$|\varepsilon(t)| < \varepsilon^{max} \tag{49}$$

Equations 47 and 48 describe inequality constraints to bound velocity, acceleration, jerk and torque to maximum values equalling $\dot{\vartheta}_c^{max}$, $\ddot{\vartheta}_c^{max}$, $\dddot{\vartheta}_c^{max}$, and τ_m^{max} respectively. Similarly, Eqs. 49 describes an inequality constraint to bound the position tracking error to a positive maximum value, ε^{max}, defined by the user according to the application requirements. Note that $\varepsilon(t)$ can be computed only if the control system effects are included in the simulations.

For what concerns the definition of the optimization cost function, $F(\mathbf{q}_{red})$, several approaches which are directly on indirectly connected to energy minimization can be defined, such as:

$$F(\mathbf{q}_{red}) \equiv \begin{cases} F^E(\mathbf{q}_{red}) = \int_0^T P_{el}(\mathbf{q}_{red},t)dt & \text{Case I,} \\ F^{PP}(\mathbf{q}_{red}) = max(P_{el}(\mathbf{q}_{red},t)) & \text{Case II,} \\ F^T(\mathbf{q}_{red}) = \int_0^T [\tau_m^*(\mathbf{q}_{red},t)]^2 dt & \text{Case III,} \\ F^{PT}(\mathbf{q}_{red}) = max(\tau_m^*(\mathbf{q}_{red},t)) & \text{Case IV.} \end{cases} \tag{50}$$

● *Case I - Energy optimal trajectories:* cost function F^E computes the time integral of the electrical power, P_{el}, namely the energy drawn from the AC network over the cycle time T.

● *Case II - Peak power optimal trajectories:* cost function F^{PP} simply computes the maximum value of the electric power P_{el} over the cycle time T.

● *Case III - Torque optimal trajectories:* cost function F^T computes the mean square of the reference PMSM torques, τ_m^*, as calculated from Eq. 12. In this case, PMSM dynamics, control architecture, and electric drive behaviour are neglected and the case is described for comparison purposes. In fact, the cost function F^T have been employed in e.g. [19] for generating energy optimal motions (lower torques result in reduced motor currents and consequently, less copper losses).

● *Case IV - Peak torque optimal trajectories:* cost function F^{PT} computes the maximum value of the reference PMSM torques, τ_m^*, as calculated from Eq. 12. Similarly to Case III, F^{PT} is used for comparison purposes and it is indirectly connected to peak power minimization.

As for the initial vector of the optimization parameters, $\mathbf{q}_{red}^0$, it is uniquely obtained by imposing a fifth-order polynomial, $\overline{\vartheta}_c^{P5}(t)$, along the desired breakpoints, the corresponding PTP trajectory being referred to as *Poly5*. As said, this function also represents the simplest PTP trajectory which satisfy the kinematic constraints of Eq. 42 while minimizing the PTP jerk Root-Mean-Square (RMS) value [7].

IV. VALIDATION OF THE METHOD

A. Experimental set-up and model validation

The proposed modelling and optimization approaches are validated against experimental measures obtained on a SAM
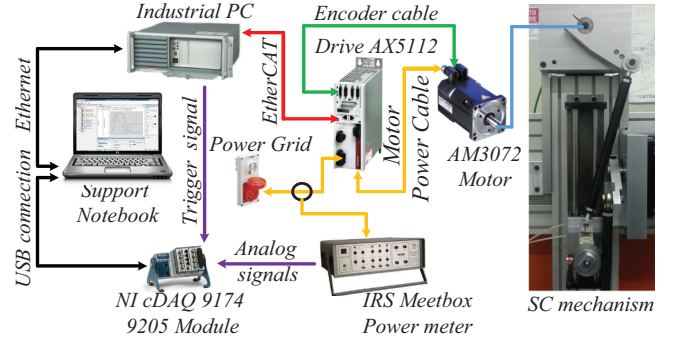

Fig. 10. Experimental Setup

TABLE I
MODEL PARAMETERS.

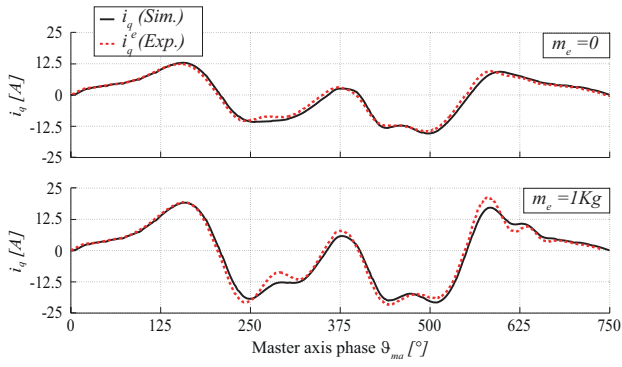| $c = 138mm$	$r = 387mm$	$m_c = 0.54kg$	$m_r = 0.24kg$
$m_s = 1.40kg$	$J_{c,O} = 4.39*10^{-3}kgm^2$	$J_{r,G} = 2.27*10^{-2}kgm^2$	
$\mu_s = 13.8N$	$v_s = 5.75Nsm^{-1}$	$\mu_c = 0.18Nm$	$v_c = 0.036Nms$
$R = 0.68\Omega$	$\mathbf{R_c = 190\Omega}$	$L_d = L_q = 10.35mH$	
$p = 5$	$K_t = 3.23Nm/A$		
$K_P^p = 30$	$K_{FF}^p = 1.00$	$K_P^v = 10$	$v_{on}^R = 890V$
$K_I^v = 26000$	$K_P^{dq} = 50$	$K_I^{dq} = 1900$	$v_{dc}^{up} = 565V$
$p_{sm} = 250\mu s$	$v_{sm} = 125\mu s$	$dq_{sm} = 62.5\mu s$	$f = 50Hz$
$\mathbf{R_{inv} = 0.225\Omega}$	$\mathbf{K_{igbt} = 67.1V}$	$C = 470\mu F$	$f = 50Hz$

Fig. 11. Simulated, i_q, and experimental, i_q^e, q-axis currents in absence (upper graph) and presence (lower graph, $m_e = 1kg$) of payload.
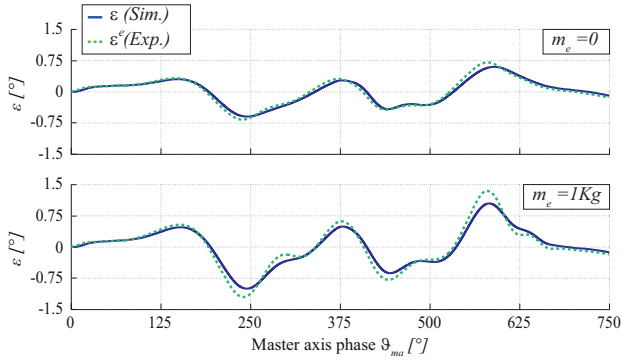

Fig. 12. Simulated, ε, and experimental, ε^e, crank position error in absence (upper graph) and presence (lower graph, $m_e = 1kg$) of payload.

physical prototype, whose schematic is depicted in Fig. 10. The rig is composed of a SC directly coupled with a Beckhoff AM3072 PMSM connected to a Beckhoff AX5112 electrical drive. The control system is based on TwinCAT software, i.e. the PC-based control platform owned by Beckhoff, connected to the drive via EtherCAT fieldbus. The crank positions, ϑ_c, measured through the motor encoder, and Dc-bus voltage, v_{dc} are provided by the TwinCAT software. A Meetbox power meter [44] and a NI cDAQ 9174 9205 Data Acquisition Devices are used to measure the electrical power, P_{el}, drawn from the network (refer to [45, Section 3.2] for further details about the measuring technique).

The model parameters used for the simulation of each sub-system are reported in Tab. I, p_{sm}, v_{sm}, dq_{sm} being position, velocity and current control loop sampling time respectively. In particular, SC geometric and inertial properties are obtained through CAD software. The position and velocity controller gains are initially designed by means of standard PID tuning techniques [46] (i.e. the Ziegler-Nichols method), and subsequently experimentally fine-tuned via a trial-and-error procedure in order to minimize the tracking error of the reference *Poly5* trajectory. As for the current loop gains, the values suggested by the drive producer are retained. Concerning PMSM and drive, all model parameters with the exception of R_c, R_{inv}, and K_{igbt} are taken from the component data sheet.

The rest of the model parameters, highlighted with bold faces in Tab. I, are obtained through system identification techniques described in [47].

The reference position profile employed for model validation is a *rise-dwell-return* trajectory composed of a PTP motion (rise phase) characterized by crank angle $\vartheta_c \in [0°, 100°]$ and master axis phase $\vartheta_{ma} \in [0°, 360°]$, a standstill (dwell phase) characterized by $\vartheta_c = 100°$ for $\vartheta_{ma} \in [360°, 390°]$, and a second PTP motion (return phase) characterized by $\vartheta_c \in [100°, 0°]$ for $\vartheta_{ma} \in [390°, 750°]$. The master angular speed, $\dot{\vartheta}_{ma}$, for all the tests, is set to 500 RPM so that 0.25s is the total travelling time. Concerning the environmental condition of the experiments, each trajectory has been tested for a limited number of times and the effect of the PMSM temperature rise has been neglected. In the industrial scenario, a linear model [41] or the aforementioned identification methods [47] can be used to better evaluate the temperature dependence of both torque constant and winding resistance.

For what concerns the prediction capabilities of the virtual prototype, Fig. 11 depict the values of simulated, $i_q(t)$, and experimental, $i_q^e(t)$, q-axis current respectively obtained in absence, $m_e = 0$, or presence, $m_e = 1kg$ of a known payload. Similarly, Fig. 12 show both simulated, $\varepsilon = \vartheta_c^* - \vartheta_c$, and experimental, $\varepsilon^e = \vartheta_c^* - \vartheta_c^e$ error between desired and actual crank angle in absence, $m_e = 0$, or presence, $m_e = 1kg$ of payload. In particular, ϑ_c is computed by means of the model depicted in Fig. 6, whereas ϑ_c^e is directly measured on the physical prototype. These direct measurements confirm the high quality prediction obtainable by means of the proposed model.

B. Optimization results

The optimization approach presented in the previous section is then employed in order to derive optimal motions, the corresponding cost functions being defined in Eq.50. The constrained non-linear optimization problem is practically solved off-line using the function *fmincon* for Medium-Scale-Optimization on a MATLAB-Simulink environment. In this case, *fmincon* uses a Sequential Quadratic Programming method, where a quadratic programming sub-problem is solved at each iteration. A line-search, is performed using a merit function similar to that proposed by [48]. At each iteration, the optimizer provides an updated $\mathbf{q}_{red}$ vector which is used to build up either cubic or quintic splines via the MATLAB function *spapi*. This spline, which complies with the corresponding constraints from Eqs. 41-44, is fed into the Simulink environment, where the solver ODE 45 computes the cost function to feed into the optimizer. Constraints from Eqs. 47-49 are handled by *fmincon*.

For what concerns numerical results, the effect of an increasing number of breakpoints has been initially evaluated and the corresponding results are listed in Tab. II, which also highlight that cubic splines are to be preferred for what concerns both energy and peak-power optimal motions. The final trajectories, either cubic or quintic splines, appear to converge to a stable solution for $n \geq 10$. Hence, an increasing number of break-

TABLE II
EFFECT OF BREAKPOINTS NUMBER.

Optimization Values		n = 5	n = 8	n = 10
F^E (J)	Cubic spline	42.0	32.7	32.4
	Quintic spline	46.7	35.0	33.2
F^{PP} (W)	Cubic spline	711	474	415
	Quintic spline	921	583	578

TABLE III
OPTIMIZATION RESULTS

	Costs	Unit	Sim.	Var. %	Exp.	Var. %
Init.	Energy	(J)	46.3	Ref. Δ%=0	48.9	Ref. Δ%=0
	Power Peak	(W)	1073	Ref. Δ%=0	1051	Ref. Δ%=0
F^E	Energy	(J)	32.4	Δ%=-30.0	32.8	Δ%=-32.9
	Power Peak	(W)	675	Δ%=-37.1	635	Δ%=-39.6
F^{PP}	Energy	(J)	39.4	Δ%=-14.9	40.2	Δ%=-17.8
	Power Peak	(W)	415	Δ%=-61.3	426	Δ%=-59.4
F^T	Energy	(J)	33.6	Δ%=-27.4	34.5	Δ%=-29.4
	Power Peak	(W)	574	Δ%=-46.5	606	Δ%=-42.3
F^{PT}	Energy	(J)	37.5	Δ%=-19.0	38.4	Δ%=-21.5
	Power Peak	(W)	672	Δ%=-37.4	667	Δ%=-36.5

(a) Energy opitmal trajectories.

(b) Peak-Power optimal trajectories

Fig. 13. Position and jerk profiles for energy optimal and peak-power optimal trajectories, respectively obtained when $F(\mathbf{q}_{red}) \equiv F^E(\mathbf{q}_{red})$ and $F(\mathbf{q}_{red}) \equiv F^{PP}(\mathbf{q}_{red})$ (see Eq. 50). Intial (Poly5), cubic (Spline3), and quintic (Spline5) spline solutions.

points would increase the computation time without substantial improvements of the final results.

Figures 13(a) and 13(b) report position and jerk profiles for the obtained energy optimal, $\overline{\vartheta}_c^E(t)$, and peak power optimal motions, $\overline{\vartheta}_c^{PP}(t)$, as compared to $\overline{\vartheta}_c^{P5}(t)$. It is worth noticing that the reference trajectory is symmetric with respect to $\vartheta_{ma} = 375°$. On the other hand the optimized trajectories are asymmetric since the rise phase and return phase are characterized by different energy requirements and peak powers. For what concerns the benefits of the optimized trajectories, the following results can be assessed:

- *Experimental Measurements*. Simulated and experimental power requirements corresponding to initial (*Poly5*) energy (*Spline3*) and peak power (*Spline3*) optimal trajectories are reported in Fig. 14, further underlying the close agreement between virtual and physical prototype. Note that, power measurements are derived as the instantaneous product of three-phase voltages and currents. Then, in order to provide clear plots of the measured instantaneous power to be compared with the model prediction, data filtering has been applied. In particular, Fig. 14(a), 14(b), and 14(c) are obtained using a moving average filter. It is evident the presence of reactive power, that is a power oscillation due to energy alternately stored and released by unmodeled inductors and/or capacitors. The presence of reactive power, whose mean value is clearly null, explains the presence of negative power values flowing from the rectifier. As for the optimized trajectories, cubic splines are represented being the best achievable solution. The experimental measurements include the transient response of the system, highlighting that a stable behaviour can be reached after the first cycle. In addition, Tab. III reports simulation (Sim.) and experimental
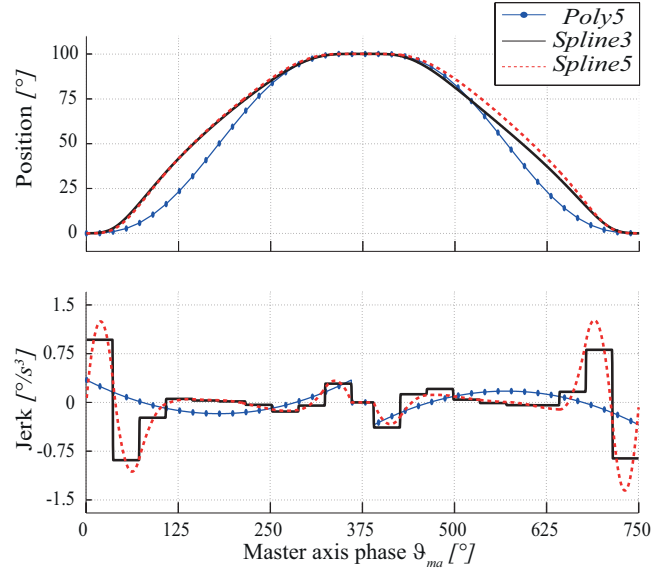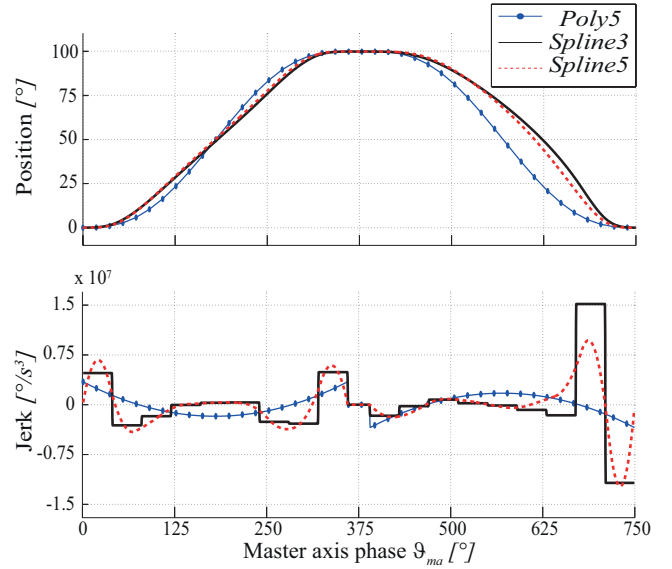
(Exp.) results concerning the energy requirement (over one cycle) for the four cost functions reported in Eq. 50. For what concerns energy minimization, an improvement of 30.0% as compared to the initial (*Poly5*) solution can be achieved by employing the cost function F^E. On the other hand, state-of-the art optimization routines, which employ cost functions F^T and F^{PT} based on partial simplified models, can achieve an improvement of 27.4% and 19.0% only. Being conceived for peak-power reduction, the cost function F^{PP} achieves an improvement of 14.9%. In fact, for what concerns peak-power optimization, the cost functions

TABLE IV
OPTIMIZATION EFFECT ON DIFFERENT ENERGY LOSSES.

	E (J)	$\Delta \int_0^T (P_f^L)dt$ (%)	$\Delta \int_0^T (P_{Cu}^L)dt$ (%)	$\Delta \int_0^T (P_{Fe}^L)dt$ (%)	$\Delta \int_0^T (P_{sw}^L)dt$ (%)	$\Delta \int_0^T (P_{cd}^L)dt$ (%)	$\Delta \int_0^T (P_D^L)dt$ (%)
Init.	46.3	-	-	-	-	-	0
F^E	32.4	-21.9	-45.0	-15.1	-50.1	-45.2	0
F^{PP}	39.4	-16.4	-19.6	-6.8	-32.0	-19.6	0
F^T	33.6	-22.5	-45.3	-12.0	-44.8	-45.4	0
F^{PT}	37.5	-12.1	-34.7	-6.1	-21.9	-34.8	0

TABLE V
ROBUSTNESS OF OPTIMIZATION RESULTS

| Factor level			Cost (J)		Factor level			Cost (J)	
ΔK_t	ΔR	Δv_s	$Init.$	F^E	ΔK_t	ΔR	Δv_s	$Init.$	F^E
-	-	-	54.4	34.1	+	-	-	43.9	28.2
-	-	+	56.4	35.6	+	-	+	45.9	29.7
-	+	-	59.9	37.3	+	+	-	47.5	30.2
-	+	+	62.0	38.8	+	+	+	49.5	31.7

F^{PP} achieve an improvement of 61.3%, whereas F^E, F^T and F^{PT} settle to 37.1%, 46.5% and 37.4% respectively. In conclusion, both in case of energy and peak-power, the description of the energy flow through a detailed virtual prototype allows to obtain better optimization results.
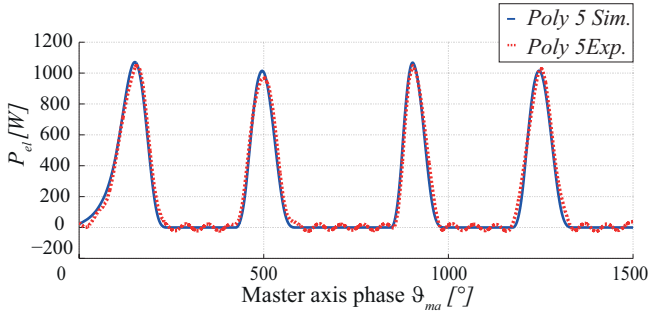
(a) Poly 5 trajectory
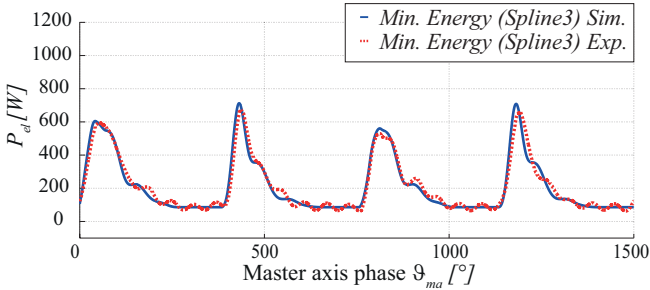
(b) Energy optimal trajectory
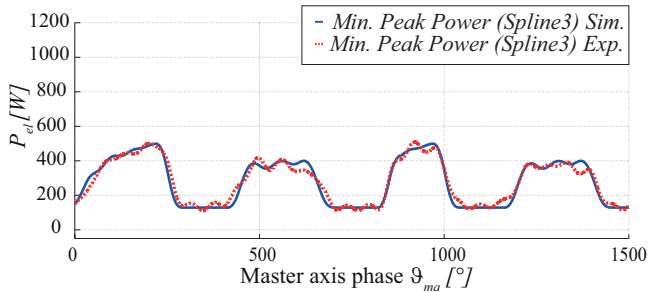
(c) Peak power optimal

Fig. 14. Simulated (Sim.) and experimental (Exp.) power requirement concerning fifth order polynomial (Poly 5), energy optimal (Spline3) and peak-power optimal (Spline3) trajectories.

- *Optimization effect on different energy losses.* As an example, Tab. IV reports the effect of the optimization routine on the energy losses depicted in Fig. 1(a). In particular, the percentage variations of the various terms as compared to their initial values is highlighted. It is interesting to note that, even if the highest improvement in energy demands is achieved when the cost function F^E is employed, friction and copper losses (respectively represented by the variations of P_f^L and P_{Cu}^L) are better reduced if F^T is applied. Nonetheless, in the latter case, the improvement in the overall energy demand is smaller since the effect of iron losses P_{Fe}^L, and inverter losses, P_{sw}^L and P_{cd}^L, losses is not implemented in the model. Also, note that no energy is lost in the drain resistor, meaning that no further benefits could be achieved by increasing the energy storage in the DC-bus capacitor.
- *Robustness of the optimization results.* A standard full factorial experimental design technique [49] is used to assess the influence of variability on motor torque constant, K_t, winding resistance, R, and slider viscous friction coefficient, v_s. These three factors are mostly affected by uncertainty due to unmodeled effects such as temperature dependence. A variability range of $\pm10\%$ has been reasonably considered for all factors, which are varied one at time from the higher to the lower possible values (respectively indicated with a + or a − sign in Tab. V). The same table reports the energy value for the *Poly 5* and for the cubic spline energy-optimal trajectory in the 2^3 cases covered by the full factorial analysis. In particular, numerical results highlight that, even in case of uncertainty, the benefits of the proposed optimization routine are preserved.

V. CONCLUSIONS

A novel method for energy and peak power optimal PTP trajectory generation has been presented. The method is based on the use of either cubic or quintic splines interpolating a series of points whose position is varied by the optimization algorithm. Experimental results concerning a position-controlled slider-crank mechanism are also provided to validate the proposed approach. The main contribution of the paper is to highlight that detailed SAM virtual prototypes are needed when computing the torque/energy requirements in case of high-dynamics motions. Since SAM are increasingly employed as substitutes of mechanical cams in intelligent manufacturing systems, as well as modern automatic machines for packaging, the application of the method in the industrial

scenario potentially enables substantial energy saving without negative effects on the system production rate.

VI. Acknowledgement

## References

[1] T. Tolio et al. SPECIES - Co-evolution of products, processes and production systems. *CIRP Annals - Manufacturing Technology*, 59(2):672–693, 2010.

[2] P. Poignet, M. Gautier, and W. Khalil. Modeling, control and simulation of high speed machine tool axes. In *IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*, pages 617–622, 1999.

[3] J.C. Latombe. *Robot Motion Planning*. Kluwer, Boston, 1991.

[4] J.T. Schwartz and M. Sharir. A survey of motion planning and related geometric algorithms. *Artif. Intell. J.*, 37:157–169, 1988.

[5] A. Gasparetto and V. Zanotto. A technique for time-jerk optimal planning of robot trajectories. *Robotics and Computer-Integrated Manufacturing*, 24(3):415–426, 2008.

[6] J.E. Bobrow, S. Dubowsky, and J.S. Gibson. Time-optimal control of robotic manipulators along specified paths. *The International Journal of Robotics Research*, 4(3):3–17, 1985.

[7] T. Flash and N. Hogans. The coordination of arm movements: An experimentally confirmed mathematical model. *Journal of Neuroscience*, 5:1688–1703, 1985.

[8] J. Gregory, A. Olivares, and E. Staffetti. Energy-optimal trajectory planning for robot manipulators with holonomic constraints. *Systems & Control Letters*, 61(2):279–291, 2012.

[9] Y. Uno, M. Kawato, and R. Suzuki. Formation and control of optimal trajectory in human multijoint arm movement. *Biological Cybernetics*, 61(2):89–101, June 1989.

[10] M. Huang, Y. Hsu, and R. Fung. Minimum-energy point-to-point trajectory planning for a motor-toggle servomechanism. *IEEE/ASME Trans. on Mechatronics*, 17(2):337–344, 2012.

[11] D. Meike, M. Pellicciari, and G. Berselli. Energy efficient use of multirobot production lines in the automotive industry: Detailed system modeling and optimization. *Automation Science and Engineering, IEEE Transactions on*, 11(3):798–809, July 2014.

[12] M. Pellicciari, G. Berselli, F. Leali, and A. Vergnano. A method for reducing the energy consumption of pick-and-place industrial robots. *Mechatronics*, 23(3):326–334, 2013.

[13] A. Gasparetto and V. Zanotto. A new method for smooth trajectory planning of robot manipulators. *Mech. and Mach. Theory*, 42(4):455–471, 2007.

[14] D.H. Jacobsen, S.B. Gershwin, and M.M. Lele. Computation of optimal singular controls. *IEEE Trans. on Automatic Control*, AC-15(Issue 1):67–73, 1970.

[15] C. Lin, P. Chang, and J.Y.S. Luh. Formulation and optimization of cubic polynomial joint trajectories for industrial robots. *IEEE Transactions on Automatic Control*, 28(12):1066–1074, 1983.

[16] C. Wang and J.G. Horng. Constrained minimum-time path planning for robot manipulators via virtual knots of the cubic b-spline functions. *IEEE Trans. on Automatic Control*, 35(5):573–577, 1990.

[17] D. Simon and C. Isik. A trigonometric trajectory generator for robotic arms. *International Journal of Control*, 57(3):505–517, 1993.

[18] K. D. Nguyen, I.M. Chen, and T. Ng. Planning algorithms for s-curve trajectories. In *2007 IEEE/ASME Int. Conf. on Advanced intelligent mechatronics*, pages 1–6, 2007.

[19] A.A. Ata. Optimal trajectory planning of manipulators: A review. *Journal of Engineering Science and Technology*, 2(1):32–54, 2007.

[20] T. Chettibi, H. E. Lehtihet, M. Haddad, and S. Hanchi. Minimum cost trajectory planning for industrial robots. *European Journal of Mechanics - A/Solids*, 23(4):703–715, 2004.

[21] L. S. Pontryagin. *Mathematical Theory of Optimal Processes*. CRC Press, New York, 1987.

[22] T. Balkan, M. Kemal Ozgoren, M. A. Sahir Arkan, and H. Murat Baykurt. A kinematic structure-based classification and compact kinematic equations for six-dof industrial robotic manipulators. *Mech. Mach. Theory*, 36(7):817–832, 2001.

[23] A. Piazzi and A. Visioli. Global minimum-jerk trajectory planning of robot manipulators. *IEEE Trans. on Industrial Electronics*, 47(1):140–149, 2000.

[24] Dan Simon. The application of neural networks to optimal robot trajectory planning. *Robotics and Autonomous Systems*, 11(1):23–34, 1993.

[25] S. Ramabalan, R. Saravanan, and C. Balamurugan. Multi-objective dynamic optimal trajectory planning of robot manipulators in the presence of obstacles. *The International Journal of Advanced Manufacturing Technology*, 41(5-6):580–594, 2009.

[26] Perumaal S. *Trigonometric S-curve based trajectory planning algorithms for precise handling applications of robotic manipulator*. PhD thesis, Anna University, Chennai, 2012.

[27] S.F.P. Saramago and V. Steffen Jr. Optimization of the trajectory planning of robot manipulators taking into account the dynamics of the system. *Mech. Mach. Theory*, 33(7):883–894, 1998.

[28] G. Field and Y. Stepanenko. Iterative dynamic programming: an approach to minimum energy trajectory planning for robotic manipulators. In *IEEE Int. Conf. on Robotics and Automation*, volume 3, pages 2755–2760, 1996.

[29] C. Hansen, J. Oltjen, D. Meike, and T. Ortmaier. Enhanced approach for energy-efficient trajectory generation of industrial robots. In *2012 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 1–7, 2012.

[30] M. Bjorkman and M. Norrlof. A method for reducing the energy consumption of an industrial robot and an industrial robot system. *ABB Technology AG*, (EP2485875 B1), 2013.

[31] E. Kiel. *Drive Solutions - Mechatronics for Production and Logistics*. Springer, 2008.

[32] J. W. Kolar U. Drofenik. A general scheme for calculating switching and conduction losses of power semiconductors in numerical circuit simulations of power electronic systems. In *Proc. of IEEE IPEC, International Power Electronics Conference*, pages 1–7, 2005.

[33] H. Dresig and F. Holzweißig. *Dynamics of Machinery, Theory and Applications*. Springer, 2010.

[34] M. Pellicciari, G. Berselli, M. Ori, and F. Leali. The role of co-simulation in the integrated design of high-dynamics servomechanisms: an experimental evaluation. *Applied Mechanics and Materials*, 278-280:1758–1764, 2013.

[35] J. Ha, R. Fung, K. Chen, and S. Hsien. Dynamic modeling and identification of a slider-crank mechanism. *Journal of Sound and Vibration*, vol. 289:1019–1044, 2006.

[36] B.K. Bose. *Modern Power Electronics and AC Drives*. Prentice Hall PTR, 2002.

[37] R. H. Park. Two-reaction theory of synchronous machines generalized method of analysis-part i. *American Institute of Electrical Engineers, Transactions of the*, 48(3):716–727, 1929.

[38] C. Cavallaro, A.O. Di Tommaso, R. Miceli, A. Raciti, G.R. Galluzzo, and M. Trapanese. Efficiency enhancement of permanent-magnet synchronous motor drives by online loss minimization approaches. *IEEE Transactions on Industrial Electronics*, 52(4):1153–1160, 2005.

[39] F. Fernandez-Bernal, A. Garcia-Cerrada, and R. Faure. Determination of parameters in interior permanent-magnet synchronous motors with iron losses without torque measurement. *IEEE Transactions on Industry Applications*, 37(5):1265–1272, 2001.

[40] P. Pillay and R. Krishnan. Modeling, simulation and analysis of permanent-magnet motordrives: Part i. the permanent-magnet synchronous motor drive. *IEEE Transactions on Industry Applications*, 25(2):265–273, 1989.

[41] R.H. Welch Jr. and G. W. Younkin. How temperature affects a servomotor's electrical and mechanical time constants. In *Industry Applications Conference, 2002. 37th IAS Annual Meeting. Conference Record of the*, volume 2, pages 1041–1046 vol.2, 2002.

[42] R. Monajemy and R. Krishnan. Comparison of torque control strategies based on the constant power loss control system for pmsm. In M.P. Kazmierkowski, R. Krishnan, and F. Blaabjerg, editors, *Control in Power Electronics: selected problems*, pages 225–249. Academic Press, Boston, 2002.

[43] L. Biagiotti and C. Melchiorri. *Trajectory Planning for Automatic Machines and Robots*. Berlin, Germany: Springer-Verlag, 2008.

[44] IRS S.r.l. @http://www.irsweb.it/MeetBOX.html, 2014.

[45] J. G. Webster. *Electrical Measurement, Signal Processing, and Displays*. CRC Press, New York, 2003.

[46] A. O'Dwyer. *Handbook of PI and PID Controller Tuning Rules*. Imperial College Press, London, 2006.

[47] E. Oliva, G. Berselli, and M. Pellicciari. A novel engineering method for the power flow assessment in servo-actuated automated machinery. In *FAIM 2014, International Conference on Flexible Automation and Intelligent Manufacturing*, pages 1–8, 2014.

[48] P.E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, London, 1981.

[49] M. Cavazzuti. *Optimization Methods: From Theory to Design*. Springer, Berlin, 2013.

Marcello Pellicciari joined the University of Modena and Reggio Emilia as a Researcher in the field of engineering design methods in 2001. He is currently Chair of the Engineering Design course at the same university and member of the IEEE technical committee on sustainable production automation. His current research interests are integrated design and virtual prototyping of mechatronic systems, industrial robotics and sustainable manufacturing.
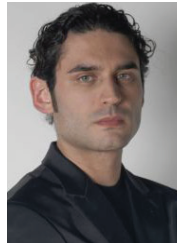
Giovanni Berselli received the Laurea degree (cum laude) in mechanical engineering from the University of Modena and Reggio Emilia, Italy, and the Ph.D. degree in Mechanics of Machines at the University of Bologna, Italy, in 2004 and 2009 respectively. He is currently an Assistant Professor at the University of Modena and Reggio Emilia, where he teaches the course *Design of Automatic Machinery and Industrial Robots*. He was Contract Professor at the University of Bologna and Research Assistant with the Department of Mechanical Engineering, Monash University, Melbourne, Asutralia, and with CEIT, Escuela Superior de Ingenieros de la Universidad de Navarra, San Sebastian, Spain. His research activity is focused on the development of engineering methods for sustainable robotics, and on the design of compliant robotic systems.

Federico Balugani received the Laurea degree in mechanical engineering from the University of Modena and Reggio Emilia, Italy, in 2013. He is currently a Research Associate with the "Enzo Ferrari" Engineering Department of the same University. Federico Balugani is (or has been) involved in several industrial projects with local firms and he currently holds a primary role within the EU project "AREUS - Automation and Robotics for European Sustainable Manufacturing". His research interests include trajectory planning for automatic machines, virtual prototyping of mechatronic systems and energy-efficient automation.