POLITECNICO DI TORINO
Repository ISTITUZIONALE

A data-driven model inversion approach to cancer immunotherapy control

(Article begins on next page)

22 December 2024

A data-driven model inversion approach
to cancer immunotherapy control

Carlo Novara and Milad Karimshoushtari

Abstract— A novel data-driven control design approach for Multiple Input Multiple Output nonlinear systems is proposed in the paper, relying on the identification of a polynomial prediction model of the system to control and its on-line inversion. A simulated study is then presented, concerning the design of a control strategy for cancer immunotherapy. This study shows that the proposed approach may be quite effective in treating cancer patients, and may give results similar to (or perhaps better than) those provided by "standard" methods. The fundamental difference is that "standard" methods are typically based on the unrealistic assumption that an accurate physiological model of the cancer-immune mechanism is available; in the approach proposed here, the controller is designed without such a strong assumption.

I. Introduction

Standard approaches to control are based on first-principle models: they assume that an accurate physical model describing the dynamics of the system to control is available. Typical examples are feedback linearization [14], Lyapunov function based control [20] and NMPC (Nonlinear Model Predictive Control) [1], [2]. However, in many real-world applications, deriving an accurate model is an extremely difficult task, since the system dynamics is often not well known and/or too complex. Robust methods might be used to deal with this problem, [12], [29]. However, deriving the necessary uncertainty models is not an easy task even for LTI (Linear Time Invariant) systems (see e.g. [7] and the references therein), and is a largely open problem in the case of nonlinear systems. Another relevant issue is that, in the case of nonlinear system, even when a reliable model is available, designing a (nonlinear) controller is in general difficult.

Data-driven methods have been introduced to cope with these problems, see, e.g., [19], [6], [5], [10], [3], [24], [16]. Several of these approaches are also described in [21], with useful discussions on their practical advantages and drawbacks. However, guaranteeing general a-priori stability and accuracy properties is not easy for many data-driven methods. Typically, the controller is first designed using some heuristic or some more systematic technique, then it is tested/tuned in simulation, and finally it is implemented (and possibly tuned) on the real plant to control. Another issue is that many data-driven methods are based on neural networks or similar approximating functions, whose design involves non-convex optimization in large dimensional spaces. There is thus no guarantee that the resulting controller is adequate to solve the control problem.

The authors are with Politecnico di Torino, Italy. Email: carlo.novara@polito.it, milad.karimshoushtari@studenti.polito.it.

In this paper, a novel data-driven control method for general MIMO nonlinear systems is proposed, called Nonlinear Inversion Control (NIC), allowing us to overcome all these relevant problems: the method does not require a physical model of the plant to control; it can guarantee a priori important properties such as closed-loop stability and tracking error accuracy; it is relatively simple; it is systematic, in the sense that the design parameters can be chosen by means of a precise procedure; it is numerically efficient, allowing a "fast" implementation on real-time processors. The NIC method is based on the identification from data of a polynomial prediction model and its online inversion through efficient optimization algorithms. An important point is that closed-loop stability is directly enforced by the identification algorithm used to derive the model, thus avoiding the need of additional on-line operations finalized at stabilization (as typically done in NMPC). A SISO version of the NIC approach can be found in [11], [27].

Thanks to these features, the NIC approach may be particularly effective in therapy control, where typically there is a huge variability among the patient population and, moreover, it is extremely difficult to derive accurate physiological models. The NIC approach allows indeed the design of therapy control strategies that are personalized for each patient, avoiding the population variability problem. Also, it does not require accurate physiological models: control design is carried out from a set of data acquired in a preliminary phase of the therapy.

A simulated study is presented, concerning the design of a control strategy for cancer immunotherapy, based on interleukin, a type of proteins that help the immune system to produce cells which can defeat cancer. This study shows that the proposed approach may be quite effective in treating cancer patients, giving results similar to (or perhaps slightly better than) those provided by a "standard" method. The fundamental difference is that this "standard" method is based on the unrealistic assumption that an accurate physiological model of the cancer-immune mechanism is available; in the NIC approach, the controller is designed without using such a strong information.

II. Problem formulation

Consider a MIMO (Multiple Input Multiple Output) nonlinear discrete-time system of the form

$$y_t = h\left(u_t^-, y_t^-, \xi_t^-\right) \tag{1}$$

$$u_t^- \doteq (u_{t-1}, \ldots, u_{t-n})$$
$$y_t^- \doteq (y_{t-1}, \ldots, y_{t-n})$$
$$\xi_t^- \doteq (\xi_{t-1}, \ldots, \xi_{t-n})$$

where $u_t \in U \subset \mathbb{R}^{n_u}$ is the input, $y_t \in \mathbb{R}^{n_y}$ is the output, $\xi_t \in \Xi \subset \mathbb{R}^{n_\xi}$ is an unmeasured disturbance, n is the system order and $t \in \mathbb{Z}$ is the time; U and $\Xi \doteq \{\xi \in \mathbb{R}^{n_\xi} : \|\xi\| \leq \bar{\xi}\}$ are compact sets, where U accounts for possible constraints on u_t.

Suppose that the system (1) is unknown, but a set of noise-corrupted measurements is available:

$$\mathcal{D} \doteq \{\tilde{y}_t, \tilde{u}_t\}_{t=1-L}^{0} \qquad (2)$$

where $\tilde{u}_t \in U$, $\tilde{y}_t \in Y$, $Y \subset \mathbb{R}^{n_y}$ is a compact set, and the tilde is used to indicate the recorded data. Experiment design techniques may be possibly used to obtain these data, [34].

Let $\mathcal{Y}^0 \subseteq R^n$ be a set of initial conditions of interest, $R \subset \mathbb{R}^{n_y}$ a compact set, $\mathcal{R} \doteq \{r = (r_1, r_2, \dots) : r_t \in R, \forall t\}$ a set of output sequences of interest and $\Xi \doteq \{\xi = (\xi_1, \xi_2, \dots) : \xi_t \in \Xi, \forall t\}$ the set of all possible disturbance sequences.

The problem is to control the system (1) such that, for any $\xi = (\xi_1, \xi_2, \dots) \in \Xi$, and for any initial condition $y_0^- \in \mathcal{Y}^0$, the output sequence $y = (y_1, y_2, \dots)$ of the controlled system tracks any reference sequence $r = (r_1, r_2, \dots) \in \mathcal{R}$.

In this paper, an original data-driven method is proposed, based on the identification from data of a polynomial prediction model and its online inversion of this model via efficient optimization.

III. POLYNOMIAL PREDICTION MODEL

Consider that the system (1) can be represented in the τ-step ahead prediction form

$$y_{t+\tau} = g\left(u_t^+, q_t^-, \xi_t^v\right) \qquad (3)$$

$$u_t^+ \doteq (u_{t+\tau-1}, \dots, u_t)$$
$$q_t^- \doteq (u_t^-, y_t^-)$$
$$\xi_t^v \doteq (\xi_{t+\tau-1}, \dots, \xi_{t-n})$$

where $g(\cdot) \doteq h^{\tau+1}(\cdot)$, h^i indicates the ith self-composition of the function h and τ is the prediction horizon. This representation can be easily obtained by iteration of (1).

The prediction model that we introduce is an approximation of the system (3), of the form

$$\hat{y}_{t+\tau} = f\left(u_t, q_t^-\right). \qquad (4)$$

For simplicity, this model is supposed of the same order as the system (3) but all the results presented in the paper hold also when the order is different.

A parametric structure is taken for the vector-valued function f. In particular, each component f_j of f is parametrized as

$$f_j(\cdot) = \sum_{i=1}^{N} \alpha_{ij} \phi_i(\cdot) \qquad (5)$$

where ϕ_i are polynomial basis functions, α_{ij} are parameters to be identified and $j = 1, \dots, n_y$. The basis function choice is in general a crucial step, [31], [13], [28]. Here, the main motivations for choosing polynomial functions are two: 1) polynomials have proven to be effective approximators in a huge number of problems; 2) as we will see later, they allow a very efficient controller evaluation.

The model parameters α_{ij} are identified from the data (2) by means of convex optimization: Define

$$z_j \doteq \begin{bmatrix} (\tilde{y}_{j,n-L+\tau})^\top \\ \vdots \\ (\tilde{y}_{j,0})^\top \end{bmatrix}$$

$$\Phi \doteq \begin{bmatrix} \phi_1\left(\tilde{u}_{n-L}, \tilde{q}_{n-L}^-\right) & \cdots & \phi_N\left(\tilde{u}_{n-L}, \tilde{q}_{n-L}^-\right) \\ \vdots & \ddots & \vdots \\ \phi_1\left(\tilde{u}_{-\tau}, \tilde{q}_{-\tau}^-\right) & \cdots & \phi_N\left(\tilde{u}_{-\tau}, \tilde{q}_{-\tau}^-\right) \end{bmatrix}$$

where $\tilde{y}_{j,i}$ is the jth component of $\tilde{y}_i$ and the tilde denotes the samples obtained from the data set (2). Define also the set

$$SC(l, \gamma, \sigma) \doteq \{\beta \in \mathbb{R}^{(L-n-\tau) \times n_y} :$$
$$\|\tilde{y}_{l,i+\tau} - \tilde{y}_{l,j+\tau} + (\Phi_j - \Phi_i)\beta\|$$
$$\leq \gamma \|\tilde{q}_i^- - \tilde{q}_j^-\| + 2\sigma, \ j \in \mathcal{T}, i \in \Upsilon_j\}$$

where $\mathcal{T} \doteq \{n-L, \dots, -\tau\}$, Φ_k is the kth row of Φ, Υ_k is the index set

$$\Upsilon_k \doteq \{i : \|(\tilde{u}_k, \tilde{u}_k^-) - (\tilde{u}_i, \tilde{u}_i^-)\| \leq \zeta\}$$

and ζ is the minimum value for which every set Υ_k contains at least two elements. Note that SC is defined by a set of linear inequalities in β and σ, and is thus convex in β and σ. In this paper, the following notation for norms is used: $\|\cdot\| \equiv \|\cdot\|_\infty$ is the vector ℓ_∞ norm; $\|\cdot\|_p$ is in general the vector ℓ_p norm.

The matrix $\alpha \in \mathbb{R}^{(L-n-\tau) \times n_y}$, whose entries α_{ij} are the parameters of the model (4)-(5), is identified by means of the following convex algorithm. Note that the algorithm is "self-tuning", in the sense that the required parameters are automatically chosen, without involving extensive heuristic procedures.

Identification algorithm 1: $\alpha = \text{id\_poly\_1}(\mathcal{D}, \hat{\gamma}_\Delta)$
For $j = 1, \dots, n_y$, compute the jth column α_j of α as follows:

1) Solve the preliminary optimization problems

$$\sigma_0 = \min_{\beta \in \mathbb{R}^N} \|z_j - \Phi\beta\|$$

$$\beta_0 = \arg\min_{\beta \in \mathbb{R}^N} \|\beta\|_1$$
$$\text{s.t.} \quad \|z_j - \Phi\beta\| \leq \sigma_0 + \rho\|z_j\|$$

where $\rho > 0$ is typically chosen "small".
2) Solve the optimization problem

$$(\alpha_j, \hat{\sigma}_\Delta) = \arg\min_{(\beta,\sigma)} \sigma$$
$$\text{s.t.} \quad \text{(i)} \quad \beta \in SC(j, \hat{\gamma}_\Delta, \sigma)$$
$$\text{(ii)} \quad \|z_j - \Phi\beta\|_p \leq \sigma\Lambda$$
$$\text{(iii)} \quad \|\beta\|_1 \leq \eta_0$$

where $\eta_0 \doteq \|\beta_0\|_1$ and $\Lambda \doteq \frac{\|z_j - \Phi\beta_0\|_p}{\|z_j - \Phi\beta_0\|}$.

After the preliminary operations carried out in step 1, a convex optimization problem is solved in step 2, providing

the model parameters. This optimization problem, representing the core of the algorithm, can be explained as follows:

1) The constraint (i) forces the function $\Delta \doteq g - f$ to have a Lipschitz constant non larger than $\hat{\gamma}_\Delta$: a result in [25] shows that this condition can be theoretically guaranteed for a sufficiently large number of data L. On the other hand, it is shown in [26] that choosing this constant smaller than 1 guarantees closed-loop stability.

2) As shown in [25], reducing the prediction error $\|z_j - \Phi\beta\|_p$ yields a "small" tracking error. Clearly, there is a trade-off between stability and tracking performance: to satisfy the constraint (i) with $\hat{\gamma}_\Delta < 1$, a "large" value of $\hat{\sigma}$ may be required, resulting in a "large" tracking error. Note that any vector norm p in (ii) can be used to reduce the prediction error. Typical choices are $p = 2$ or $p = \infty$.

3) Bounding the ℓ_1 norm leads to a sparse matrix α, i.e. a matrix with a "small" number of non-zero elements, [32], [9], [22]. Sparsity is important to ensure a low complexity of the model, limiting well known issues such as over-fitting and the curse of dimensionality. Sparsity leads also to an efficient implementation of the model/controller on real-time processors, which may have limited memory and computation capabilities.

IV. NIC CONTROL

In this section, a control approach for nonlinear systems is proposed, called Nonlinear Inversion Control (NIC), relying on the efficient online inversion of the model (4).

The basic idea of this approach is the following: at each time $t > 0$, given a reference $r_{t+\tau}$ and the current regressor q_t^-, a command u_t^* is looked for, such that the model output $\hat{y}_{t+\tau}$ is "close" to $r_{t+\tau}$:

$$\hat{y}_{t+\tau} = f\left(u_t^*, q_t^-\right) \cong r_{t+\tau}. \tag{6}$$

Note that the latter equality may be not exact for two reasons: 1) $r_{t+\tau}$ may be not reachable; that is, no $u_t^* \in U^\tau$ may exist for which $\hat{y}_{t+\tau}$ is exactly equal to $r_{t+\tau}$; 2) values of u_t^* with a limited ℓ_2 norm may be of interest in order to have a not too high command activity. This kind of inversion is an approximate right-inversion and can be operated also when f is not injective wrt u_t^* (e.g., for some $r_{t+\tau}$ and q_t^-, more than one u_t^* may exist such that (6) holds).

The command input yielding (6) is found solving the optimization problem

$$u_t^* = \arg \min_{u \in U^\tau} J\left(u, r_{t+\tau}, q_t^-\right) \tag{7}$$

where

$$J\left(u, r_{t+\tau}, q_t^-\right) \doteq \left\|r_{t+\tau} - f\left(u, q_t^-\right)\right\|_2^2 + \mu \|u\|_2^2 \tag{8}$$

and $\mu \geq 0$ is a design parameter, determining the trade-off between tracking precision and command activity. The NIC control law is fully defined by (7).

It is important to observe that the objective function (8) is in general non-convex. Moreover, the optimization problem (7) has to be solved on-line, and this may take a long time compared to the sampling time used in the application of interest. To overcome these relevant problems, three algorithms are proposed in the next subsections, allowing an efficient computation of the optimal command input u_t^* for the following cases: A) SIMO system; B) MIMO system with predictor affine in u_t; C) general MIMO system.

Remark 1: If the components of y_t and/or u_t have range of variations with different scales, weighted norms can be used in (8). □

A. SIMO system

The system to control is here supposed to have a single input: $u_t \in \mathbb{R}$. In this situation, for given $r_{t+\tau}$ and q_t^-, the objective function (8) is a polynomial in the scalar variable u. Its minima can thus be found computing the roots of its derivative, as done in the following algorithm.

Control algorithm 1: $u^* = K^1\left(J, r_{t+\tau}, q_t^-\right)$.
Compute the optimal input as

$$u^* = \arg \min_{u \in U^s} J\left(u, r_{t+\tau}, q_t^-\right)$$

where

$$U^s \doteq \left(\mathrm{Rroots}\left(J'\left(u, r_{t+\tau}, q_t^-\right)\right) \cap U\right) \cup \{\underline{u}, \overline{u}\},$$

J' is the derivative of J wrt u, $\mathrm{Rroots}\left(J'\right)$ denotes the set of all real roots of J', and $\underline{u}$ and $\overline{u}$ are the boundaries of U.

Remark 2: The derivative J' can be computed analytically. Moreover, U^s is composed by a "small" number of elements:

$$\mathrm{card}\left(U^s\right) < \deg\left(J\left(u, r_{t+\tau}, q_t^-\right)\right) + 2$$

where card is the set cardinality and deg indicates the polynomial degree. The evaluation of u^* through Algorithm 1 is thus extremely fast, since it just requires to find the real roots of a univariate polynomial whose analytical expression is known and to compute the objective function for a "small" number of values. The computation times required for obtaining u^* are evaluated in [23] by means of extensive simulations. □

B. MIMO system with predictor affine in u_t

Suppose that an accurate prediction model (4) affine in u_t is found (this model can be of any degree in the other variables). Then, the objective function (8) is convex in u, being the sum of two terms given by the square norms of functions affine in u. The problem of minimizing such a convex function is standard. However, minimization has to be performed on-line and widely used algorithms such as those based on interior-point techniques may not be fast enough, [4].

Here, a novel "fast" algorithm is proposed, based on a coordinate minimization scheme where, at each iteration, the cost function (8) is minimized wrt a single component of the command input.

Control algorithm 2: $\quad u^* = K^2\left(J, u0, r_{t+\tau}, q_t^-\right).$

1) Set $j = 1$, $\mathfrak{u}^{(0)} = u0 \in U$ (a simple choice is taking $u0$ as the center of U).

2) For $i = 1, \cdots, n_u$, let

$$\mathfrak{u}_i^{(j)} = K^1\left(J_i^{(j)}, r_{t+\tau}, q_t^-\right)$$

where $K^1\left(J_i^{(j)}, r_{t+\tau}, q_t^-\right)$ is computed by Algorithm 1, minimizing wrt $\mathfrak{u}_i$ the cost function

$$J_i^{(j)} \doteq J(\mathfrak{u}_1^{(j)}, \ldots, \mathfrak{u}_{i-1}^{(j)}, \mathfrak{u}_i, \mathfrak{u}_{i+1}^{(j-1)}, \ldots, \mathfrak{u}_{m_u}^{(j-1)}, r_{t+\tau}, q_t^-).$$

3) If $\left|J_{m_u}^{(j)} - J_{m_u}^{(j-1)}\right| < \varepsilon_J$, where ε_J is a user-defined precision parameter, stop and return the optimal input

$$u^* = \mathfrak{u}^{(j)} = \left(\mathfrak{u}_1^{(j)}, \ldots, \mathfrak{u}_{m_u}^{(j)}\right);$$

else set $j \leftarrow j + 1$ and goto 2.

Remark 3: Convergence of the coordinate descent scheme in Algorithm 2 is guaranteed, see [33]. $\qquad \square$

Remark 4: From Remark 2 and the convergence properties of the coordinate minimization scheme, [33], it follows that the evaluation of u^* through Algorithm 2 is extremely fast. The computation times required for obtaining u^* are evaluated in [23] by means of extensive simulations $\qquad \square$

C. General MIMO system

In several real world applications, the command input acts on the system to control in a nonlinear way, resulting in a function g highly nonlinear in the input. In these situations, controlling the system can be significantly more difficult since this function may be non-invertible (more precisely, non-injective) wrt u_t. The present approach allows us to efficiently deal with these cases. The following algorithm is proposed, based on the two presented above.

Control algorithm 3: $\quad u^* = K^3\left(J, r_{t+\tau}, q_t^-\right).$

1) Set $j = 1$, $u0 = u_{t-1}$.

2) Let

$$\mathfrak{u}^{(j)} = K^2\left(J, u0, r_{t+\tau}, q_t^-\right)$$

where $K^2\left(J, u0, r_{t+\tau}, q_t^-\right)$ is computed by Algorithm 2.

3) If $j = N_{iter}$, where N_{iter} is the user-defined maximum number of iterations, stop and return the locally optimal input

$$u^* = \mathfrak{u}^{(j)};$$

else set $j \leftarrow j + 1$ and goto 2, changing the starting point $u0$.

The choice of $u0$ for $j > 1$ can be made considering a standard space-filling algorithm such as the (multidimensional) Peano Curve, Hilbert curve, H-tree fractal, or a randomized approach.

The maximum number of iterations N_{iter} in step 3 can be chosen on the basis of the sampling time used in the application of interest: N_{iter} should be chosen not larger than the sampling time divided by the the time required for computing u^*.

As already observed, the objective function (8) is in general non-convex. While in the scalar and affine cases we are able to find a global solution, here we find a solution that is not "theoretically" guaranteed to be global. However, it is shown in [23] by means of extensive simulations that Algorithm 3 is able to find always a solution very close to the global optimum, in very short times.

Remark 5: Stability of the closed-loop system formed by the feedback connection of the plant (1) with the controller (7) can be theoretically guaranteed. A proof for the SISO case with model prediction horizon 1 can be found in [26]; the extension to the general MIMO case with prediction horizon larger than 1 can be made using similar techniques. These proofs are developed in a Set Membership framework, [18], [17], [30], and are not reported in this paper for space reasons. $\qquad \square$

V. SIMULATED STUDY: CONTROL DESIGN FOR CANCER IMMUNOTHERAPY

Cancer represents nowadays one of the most relevant causes of death. In the last decades, immunotherapy has become an effective method for treating several types of cancer. New types of immune treatments are currently being studied, which are expected to give an important impact on cancer treatment in the future. In particular, immunotherapies have a greater potential than current treatment approaches to fight cancer more efficiently, to offer longer-term protection, to present fewer side effects, and to benefit more patients with more cancer types.

Roughly speaking, immunotherapy enforces the body natural defenses, allowing them to fight cancer. It can use substances produced either by the body or in a laboratory to improve or restore the immune system functions. Among the existing immunotherapy approaches, one of the most promising is based on a family of proteins called cytokines and, in particular, on a group of cytokines called interleukins. This kind of proteins help the immune system to produce cells that can defeat cancer. An interleukin made in laboratory, called interleukin-2 (IL-2), is employed for example to treat kidney and skin cancers. IL-2 is typically used together with adoptive cellular immunotherapy (ACI), a technique which consists in injecting cultured immune cells that have antitumor reactivity into tumor bearing host.

Despite several positive aspects, immunotherapy still presents several problems: the treatment strongly depends on the personal experience of the physician; there is a huge variability of response among the patient population; it is very difficult to derive accurate physiological models, describing the patient response to several types of treatment. The "standard" control approaches are in general not able to systematically deal with the last two of these problems, because they indeed rely on physiological models. On the contrary, we will see below that the data-driven control

approach proposed in this paper may allow us to overcome all the problems mentioned above.

In the present study, the cancer-immune dynamic model of [15], [8] has been considered. This model, though relatively simple, is able to properly describe the cancer-immune dynamics, accounting also for the IL-2 effects. The model is based on three main variables:

- x_1 [cells/ml]: concentration of activated immune-system cells (also called effector cells);
- x_2 [cells/ml]: concentration of cancer cells;
- x_3 [μg/l]: concentration of IL-2 in the single cancer-site compartment.

The nonlinear state equations describing the time evolution of these variables are the following:

$$
\begin{aligned}
\dot{x}_1 &= cx_2 - \mu_2 x_1 + \frac{p_1 x_1 x_3}{g_1 + x_3} + u_1 + \xi_1 \\
\dot{x}_2 &= r_2(1 - bx_2)x_2 - \frac{ax_1 x_2}{g_2 + x_2} \\
\dot{x}_3 &= \frac{p_2 x_1 x_2}{g_3 + x_2} - \mu_3 x_3 + u_2 + \xi_2
\end{aligned}
\tag{9}
$$

where u_1 and u_2 are the treatment inputs; u_1 [cells/ml/days] represents an external source of effector cells such as ACI cells; u_2 [μg/l/days] represents an external source of IL-2. ξ_1 and ξ_2 are random disturbances, normally distributed, with zero mean and a noise to signal standard deviation ratio of 5%. The same parameter values of [15], [8] have been assumed: $c = 0.035$ days^{-1}, $\mu_2 = 0.03$ days^{-1}, $p_1 = 0.1245$ days^{-1}, $g_1 = 2e7$ [μg/l], $r_2 = 0.18$ days^{-1}, $b = 1e - 9$ [ml/cells], $a = 1$ days^{-1}, $g_2 = 1e5$ [cells/ml], $p_2 = 5$ days^{-1}, $g_3 = 1e3$ [cells/ml], $\mu_3 = 10$ days^{-1}.

The model (9) represents the "true " system to be controlled. Supposing to be in a realistic situation, this system has been assumed unknown. The aim was to control it by means of treatments u_1 and u_2 in order to bring the concentration of cancer cells x_2 close to zero, guaranteeing also correct levels of the concentrations x_1 and x_3.

An overall control period of 18 months was considered, divided into two phases:

Phase 1. Manual open-loop treatment (first 6 months). In this phase, u_1 and u_2 were chosen as random signals normally distributed, with means 400 [cells/ml/days] and 400 [μg/l/days], respectively, and standard deviations 300 [cells/ml/days] and 300 [μg/l/days], respectively. The goals of this first phase are two: first, preventing significant increases of the number of cancer cells; second, collecting data to use for control design in the next phase.   □

Phase 2. Automatic closed-loop treatment (from the 7th to the 18th month). Using the data generated in the first phase (a total of $L = 180$ data, collected with a sampling time $T_s = 1$ day), a NIC controller was designed. The sampling time $T_s = 1$ day was considered for simplicity. Larger sampling times can be more suitable for the patient sake and also for practical reasons. In any case, the procedure proposed here can be applied without being affected by the sampling time choice. The relevant discrete-time variables were defined as $y_t = (x_1(tT_s), x_2(tT_s), x_3(tT_s))$ and $u_t = (u_1(tT_s), u_2(tT_s))$, where $t \in \mathbb{Z}$. Control design was carried out according to the procedure of Sections III and IV, yielding the following parameter values: $\hat{\gamma}_\Delta = 0.9$, $n = 2$, $\tau = 3$, $\mu = 0.001$. The model was identified considering, for each output, 45 polynomial basis functions up to degree 2 (no significant improvements were observed increasing the degree). In average, about 16 functions for each output were selected by the identification algorithm. Then, the control algorithm 3 was applied to the cancer-immune system (9), using the constant reference $r_t = (1.9e4, 0, 0)$, $\forall t$.   □

A possible variant of this two-phase procedure is to follow an adaptive scheme, dividing the second phase in several sub-phases: in each sub-phase, the controller is designed using the data collected in the first phase and the previous sub-phases.

Note that a first preliminary phase is important not only for the NIC approach presented here but for all control approaches (including first-principle and model-free approaches (e.g., PID control)). Indeed, in the case of first principle-approaches, even in the optimistic case that a reasonable physiological model is available, its validation on experimental data is a fundamental step, without which no reliable therapy strategy can be planned. Moreover, in both cases of first-principle and model-free approaches, some initial phase for tuning the controller is necessary. Clearly, the preliminary phase must be as short as possible, to avoid unexpected increases of the cancer cells and let the controller start working as soon as possible.

The results of the proposed two-phase therapy are shown in Figure 1. It can be noted that the concentration of cancer cells x_2 is brought close to zero in less than 400 days, while the levels of the other two states is maintained close to the desired values. This performance is similar to (or perhaps slightly better than) the one obtained in [8]. The important difference is that, in [8], the cancer-immune system (9) is assumed perfectly known; here, the controller was designed without using such a strong information.

Another interesting observation is that, during the open-loop therapy, the number of cancer cells first decreases but, after about 160 days, starts to increase again (until the closed-loop therapy begins to work). This kind of behavior seems to be consistent with the results presented in [15], showing that limit cycles may appear during the time evolution of the cancer-immune system (9).

A further important observation is that the mean drug rate (MDR), defined as $\sum_{k=1}^{N} (u_{1,k} + u_{2,k})/N$, during the first phase resulted to be 779, while the MDR during the second phase resulted to be 738. This shows that the closed-loop control therapy is able to obtain the desired positive effects in term of cancer cells decrease, using however smaller quantities of therapy drugs wrt the open-loop manual therapy. This aspect is clearly fundamental, since these drugs may have in general negative collateral effects on the patient.

To further check the reliability of these results, a Monte Carlo simulation was carried out, where the whole process described above was repeated for 200 trials. Each trial was different from each other under the following aspects: 1) different realizations of u_1 and u_2 were used in the first phase; 2) different realizations of ξ_1 and ξ_2 were used in
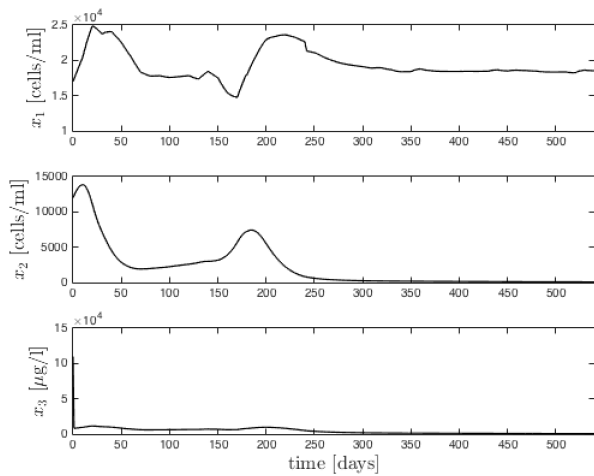
Fig. 1. Behavior of the cancer-immune system states during the two-phase therapy.

both the phases; 3) the system parameters were generated randomly, according to normal distributions with means equal to the nominal values indicated below (9), and standard deviations equal to 10% of the nominal values. The results obtained in the Monte Carlo simulation can be summarized as follows: average time required to bring the cancer cell concentration close to zero $\simeq 350$ or 400 days (maintaining the other two states close to the desired values); average MDR values $\simeq 785$ and 741 for the two phases. These results provide a reliable confirmation that the proposed NIC control approach may be effective in therapy control design, presenting several advantages wrt more "standard" methods.

REFERENCES

[1] Assessment and future directions of nonlinear model predictive control. In R. Findeisen, F. Allgower, and L.T. Biegel, editors, *Lecture Notes in Control and Information Sciences*. Springer, 2007.

[2] Nonlinear model predictive control - theory and algorithms. In L. Grune and J. Pannek, editors, *Communications and Control Engineering*. Springer, 2011.

[3] M.D. Brown, G. Lightbody, and G.W. Irwin. Nonlinear internal model control using local model networks. *IEE Proceedings - Control Theory and Applications*, 144(6):505–514, 1997.

[4] G.C. Calafiore and L. El Ghaoui. *Optimization Models*. Cambridge University Press, 2014.

[5] M.C. Campi and S.M. Savaresi. Direct nonlinear control design: The virtual reference feedback tuning (VRFT) approach. *IEEE Transactions on Automatic Control*, 51(1):14–27, 2006.

[6] Fu-Chuang Chen and H.K. Khalil. Adaptive control of a class of nonlinear discrete-time systems using neural networks. *IEEE Transactions on Automatic Control*, 40(5):791–801, 1995.

[7] J. Chen and G. Gu. *Control-Oriented System Identification: An H_∞ Approach*. John Wiley & Sons, New York, 2000.

[8] Ting Li Chien, Chung Cheng Chen, and Chiou Jye Huang. Feedback linearization control and its application to MIMO cancer immunotherapy. *IEEE Transactions on Control Systems Technology*, 18(4):953–961, 2010.

[9] D.L. Donoho, M. Elad, and V.N. Temlyakov. Stable recovery of sparse overcomplete representations in the presence of noise. *IEEE Transactions on Information Theory*, 52(1):6 – 18, jan. 2006.

[10] S. Formentin, P. De Filippi, M. Corno, M. Tanelli, and S.M. Savaresi. Data-driven design of braking control systems. *IEEE Transactions on Control Systems Technology*, 21(1):186–193, 2013.

[11] S. Formentin, C. Novara, S.M. Savaresi, and M. Milanese. Active braking control system design: the d2-ibc approach. *IEEE/ASME Transactions on Mechatronics*, 20(4):1573–1584, 2015.

[12] A. Freeman and V. Kokotovic. *Robust Nonlinear Control Design*. Birkhuser, Boston, 1996.

[13] K. Hsu, C. Novara, T. Vincent, M. Milanese, and K. Poolla. Parametric and nonparametric curve fitting. *Automatica*, 42/11:1869–1873, 2006.

[14] A. Isidori. *Nonlinear Control Systems*. Springer, 1995.

[15] D Kirschner and J C Panetta. Modeling immunotherapy of the tumor-immune interaction. *Journal of mathematical biology*, 37:235–252, 1998.

[16] F.L. Lewis, D. Vrabie, and K.G. Vamvoudakis. Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers. *Control Systems, IEEE*, 32(6):76–105, 2012.

[17] M. Milanese and C. Novara. Model quality in identification of nonlinear systems. *IEEE Transactions on Automatic Control*, 50(10):1606–1611, 2005.

[18] M. Milanese and C. Novara. Unified set membership theory for identification, prediction and filtering of nonlinear systems. *Automatica*, 47(10):2141–2151, 2011.

[19] K. S. Narendra and K. Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Transaction on Neural Networks*, 1(1):4–27, 1990.

[20] S.G. Nersesov and M.M. Haddad. On the stability and control of nonlinear dynamical systems via vector lyapunov functions. *IEEE Transactions on Automatic Control*, 51(2):203 – 215, 2006.

[21] M. Norgaard, O. Ravn, N. Poulsen, and L. K. Hansen. *Neural Networks for Modeling and Control of Dynamic Systems*. Springer, 2000.

[22] C. Novara. Sparse identification of nonlinear functions and parametric set membership optimality analysis. *IEEE Transactions on Automatic Control*, 57(12):3236–3241, 2012.

[23] C. Novara. Polynomial model inversion control: numerical tests and applications. *arXiv*, (1509.01421), 2015.

[24] C. Novara, M. Canale, M. Milanese, and M.C. Signorile. Set Membership inversion and robust control from data of nonlinear systems. *International Journal of Robust and Nonlinear Control*, 24(18):3170–3195, 2014.

[25] C. Novara, L. Fagiano, and M. Milanese. Direct feedback control design for nonlinear systems. *Automatica*, 49(4):849–860, 2013.

[26] C. Novara and S. Formentin. Data-driven controller design for nonlinear systems: a two degrees of freedom architecture. *arXiv*, 2014.

[27] C. Novara, S. Formentin, S.M. Savaresi, and M. Milanese. Data-driven design of two degree-of-freedom nonlinear controllers: the D2-IBC approach. *Automatica*, 72:19–27, 2016.

[28] C. Novara, T. Vincent, K. Hsu, M. Milanese, and K. Poolla. Parametric identification of structured nonlinear systems. *Automatica*, 47(4):711 – 721, 2011.

[29] Z. Qu. *Robust Control of Nonlinear Uncertain Systems*. Wiley series in nonlinear science, 1998.

[30] F. Ruiz, C. Novara, and M. Milanese. Direct design from data of optimal filters for LPV systems. *Systems and Control Letters*, 59/1:1–8, 2010.

[31] J. Sjöberg, Q. Zhang, L. Ljung, A. Benveniste, B.Delyon, P. Glorennec, H. Hjalmarsson, and A. Juditsky. Nonlinear black-box modeling in system identification: a unified overview. *Automatica*, 31:1691–1723, 1995.

[32] R Tibshirani. Regression shrinkage and selection via the Lasso. *Royal. Statist. Soc B.*, 58(1):267–288, 1996.

[33] P. Tseng. Convergence of a block coordinate descent method for nondifferentiable minimization. *J. of Optimization Theory and Applications*, 109(3):475–494, 2001.

[34] T. Vincent, C. Novara, K. Hsu, and K. Poolla. Input design for structured nonlinear system identification. *Automatica*, 46(6), 2010.