MPC as a DVI: Implications on Sampling Rates and Accuracy

Mihai Anitescu and Victor M. Zavala

***Abstract*— We show that closed-loop model predictive control (MPC) can be cast as a differential variational inequality (DVI). The DVI abstraction reveals that standard sampled-data MPC implementations (in which the control law is computed using states that are sampled at predefined sampling intervals) corresponds to an explicit Euler time-stepping scheme applied to the DVI. As expected, this explicit scheme induces large approximation errors (with respect to the optimal solution manifold of the DVI) when long sampling intervals are used. Motivated by this observation, we propose to use an implicit Euler scheme which allows us to use much longer sampling intervals, but note that this scheme requires the solution of a complicated variational inequality at every sampling time. We thus propose to use a predictor-corrector scheme, which can be implemented with off-the-shelf optimization tools and which mimics the performance of the implicit scheme.**

I. Problem Statement and Basic Definitions

We begin by defining a dynamical system of the form:

$$\dot{x}(t) = f(x(t), u(x(t))) \tag{1}$$

where $t \in \mathbb{R}_+$ is time, $x : \mathbb{R}_+ \to \mathbb{R}^{n_x}$ is the state trajectory (an implicit function of time), and $u : \mathbb{R}^{n_x} \to \mathbb{R}^{n_u}$ is the control trajectory. The control trajectory $u(x(t))$ is an implicit function of the state $x(t)$ (and thus of time).

In model predictive control (MPC), the aim is to compute the control action $u(x(t))$ by solving an optimization problem in real-time using the state $x(t)$. We represent this optimization problem as:

$$u(x(t)) := \operatorname{argmin}_{v \in \mathscr{U}} \varphi(x(t), v), \tag{2}$$

where $\varphi : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \to \mathbb{R}$ is a twice continuously differentiable cost function and $\mathscr{U} \subseteq \mathbb{R}^{n_u}$ is the feasible control set. Here, we highlight that the solution of (2) is an implicit function of the current state $x(t)$.

The evolution of the dynamical system (1) over the interval $[t, t_+]$ (where $t_+ := t + h$) and *for a fixed control* u is expressed in integral form as:

$$\begin{aligned} x^u(t_+) &= x(t) + \int_t^{t_+} f(x(\tau), u)\, d\tau \\ &= x(t) + h f(x(t), u) + \mathscr{O}(h^2). \end{aligned} \tag{3}$$

Here, the $\mathscr{O}(h^2)$ approximation follows from the Taylor expansion:

$$\begin{aligned} x^u(t_+) &= x(t) + h\dot{x}(t) + \mathscr{O}(h^2) \\ &= x(t) + h f(x(t), u) + \mathscr{O}(h^2). \end{aligned} \tag{4}$$

M. Anitescu is with the Mathematics and Computer Science Division, Argonne National Laboratory, Lemont, IL. Email: anitescu@mcs.anl.gov. V. M. Zavala is with the Department of Chemical and Biological Engineering, University of Wisconsin-Madison, Madison, WI. Email: victor.zavala@wisc.edu.

The error term $\mathscr{O}(h^2)$ is an implicit function of u, we do not highlight this implicit dependence in order to keep the notation compact (this can be done without loss of generality).

A fundamental problem of MPC is that the control action $u(x(t))$ cannot be computed in real-time (i.e., at every time instant $t \geq 0$). Consequently, for implementation, a sampled-data scheme is used [1]. Here, the control action is computed at the state $x(t)$ sampled at instance $t = kh$ (with positive integer $k \in \mathbb{Z}_+$) and given by $u(x(t))$. This control action is fed into the system and *held constant* over period h to give the future state:

$$x^e(t_+) := x(t) + \int_t^{t_+} f(x(\tau), u(x(t)))\, d\tau \tag{5}$$

at the new sampling time $t_+ = t + h = (k+1)h$ (with sampling interval $h \in \mathbb{R}_+$). The new state $x^e(t_+)$ is used to compute the new control action $u(x^e(t+h))$ and the process is repeated. Starting at $t = 0$ (kh with $k = 0$), the state evolution $x^e(t)$ under the sampled-data MPC scheme can be expressed as:

$$\dot{x}^e(t) = f(x(t), u(x(kh))),\ t \in [kh, (k+1)h], \tag{6}$$

for $k \in \mathbb{Z}_+$. This *sampled-data MPC* scheme only provides *an approximation* of the optimal evolution $x(t)$ satisfying:

$$\dot{x}(t) = f(x(t), u(x(t))),\ t \in \mathbb{R}_+ \tag{7}$$

We refer to the optimal state evolution $x(t)$ as the *optimal solution manifold*. This is illustrated in Figure I. We highlight that, in the optimal evolution, the control law is computed at every instant $t \in \mathbb{R}_+$. This contrasts with sampled-data MPC in which the control law is only computed at the sampling instants $t = kh$, $k \in \mathbb{Z}_+$. To quantify the local error induced by sampled-data MPC over an interval $[t, t_+]$, we will compare the approximate state (5) with the optimal state:

$$x(t_+) := x(t) + \int_t^{t_+} f(x(\tau), u(x(\tau)))\, d\tau. \tag{8}$$

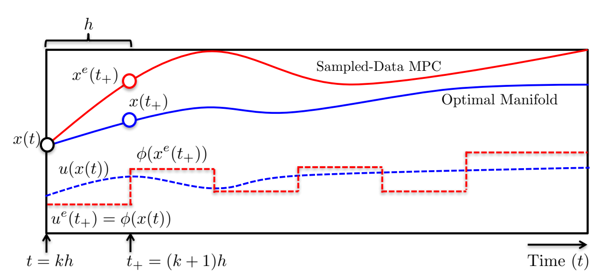

Fig. 1. Sketch of optimal solution manifold and sampled-data MPC.

A wide range of alternatives have been proposed in the literature to reduce the sampling interval h and with this allow sampled-data MPC to better approximate the optimal manifold. Key developments along these lines are based on the observation that the control law $u(x(t))$ does not have to be solved exactly to track the solution manifold [2]–[7]. These works have shown that one can drastically reduce the latency associated to the computation of the control law and thus the sampling interval. In particular, the reported schemes aim to compute approximate solutions of the nonlinear optimization problem (2) by solving a surrogate of lower complexity such as a quadratic program (QP). The solution of the QP can be interpreted as a predictor step that is tangential to the solution manifold and that has an $\mathcal{O}(h^2)$ approximation error [2], [4]. In addition, it has been shown that it is possible to compute a corrector step to improve accuracy to $\mathcal{O}(h^4)$ by solving an additional QP [5].

In this work, we do not aim at reducing computational latency of MPC but instead study the interplay between the sampling time, the system dynamics, and the control law computation. In particular, we aim to analyze how the approximation of the optimal manifold is affected by the selection of the state-sampling scheme and by the computation of the control law. These observations will allow us to construct alternative schemes that *sample states less frequently* and that can better approximate the solution manifold compared to sampled-data MPC. By sampling less frequently, we are able to *accommodate larger* applications. Our results are based on the key observation that the optimal manifold solution solves a differential variational inequality (DVI) and that sampled-data MPC schemes can be interpreted as time stepping schemes applied to the DVI. We also show how to use approximation results for generalized equations to establish error bounds for time stepping schemes.

II. MPC AS A DVI

To establish that the optimal solution manifold solves a DVI, we begin by observing that the solution $u(x(t))$ of the optimization problem (2) satisfies the generalized equation [4], [8]:

$$0 \in \Phi(x(t), u(x(t))) + \mathcal{N}_{\mathcal{U}}(u(x(t))) \tag{9}$$

where $\Phi(x(t), u(x(t))) := \nabla_u \varphi(x(t), u(x(t)))$ is the gradient of the cost function $\varphi(\cdot,\cdot)$ with respect to the controls, and $\mathcal{N}_{\mathcal{U}}(\cdot)$ is the normal cone operator:

$$\mathcal{N}_{\mathcal{U}}(u) := \begin{cases} \{v \,|\, (u-u')^T v \geq 0 \quad \forall u' \in \mathcal{U}\} & \text{if } u \in \mathcal{U} \\ \emptyset & \text{if } u \notin \mathcal{U} \end{cases} \tag{10}$$

It is clear that any solution of the generalized equation also satisfies the first-order optimality condition:

$$(u(x(t)) - u')^T \Phi(x(t), u(x(t))) \geq 0, \quad \forall u' \in \mathcal{U}. \tag{11}$$

We note that this is a variational inequality, which we express as $VI(\mathcal{U}, \Phi)$ [9]. The variational inequality has an associated solution set $\mathrm{SOL}(\mathcal{U}, \Phi)$. Consequently, we have that $u(x(t)) \in \mathrm{SOL}(\mathcal{U}, \Phi(x(t), \cdot))$. Finally, we note that for $\mathcal{U} = \mathbb{R}^{n_u}$, the generalized equation becomes the set of nonlinear equations $\nabla_u \varphi(x(t), u(x(t))) = 0$.

Based on these definitions, it is clear that the optimal state evolution (7) satisfies the DVI:

$$\dot{x}(t) = f(x(t), u(x(t))) \tag{12a}$$
$$u(x(t)) \in \mathrm{SOL}(\mathcal{U}, \Phi(x(t), \cdot)). \tag{12b}$$

The state evolution under the sampled-data MPC implementation (5) can be expressed as:

$$x^e(t_+) = x(t) + hf(x(t), u(x(t))) + \mathcal{O}(h^2) \tag{13}$$

and we define the control action as $u^e(t_+) := u(x(t))$. The state evolution can be written in terms of the DVI as:

$$x^e(t_+) = x(t) + hf(x(t), u(x(t))) + \mathcal{O}(h^2) \tag{14a}$$
$$u(x(t)) \in \mathrm{SOL}(\mathcal{U}, \Phi(x(t), \cdot)). \tag{14b}$$

From this observation, it becomes clear that sampled-data MPC is an *explicit time-stepping scheme applied to the DVI* (12) [9]. From this we can also expect that the approximation error of the optimal solution manifold will be large for large sampling interval h. We also highlight that this explicit sampled-data MPC scheme evaluates the control law at the current state $x(t)$ and thus approximates the slope of the solution manifold at this point. This has the effect of *decoupling the computation of the control law from the system dynamics*.

The time-stepping interpretation of sampled-data MPC also raises the possibility of considering an *implicit sampled-data MPC* scheme that approximates the manifold as:

$$\begin{aligned} x^i(t_+) &= x(t) + hf(x^i(t_+), u(x^i(t_+))) + \mathcal{O}(h^2) \\ &= x(t) + \int_t^{t_+} f(x(\tau), u(x^i(t_+))) d\tau \end{aligned} \tag{15}$$

This scheme will achieve a more accurate approximation of the solution manifold but requires the solution of a complicated *coupled dynamics-control law system*. In particular, the solution of this coupled system satisfies:

$$x^i(t_+) = x(t) + hf(x^i(t_+), u^i(t_+)) + \mathcal{O}(h^2) \tag{16a}$$
$$u^i(t_+) = \mathrm{argmin}_{v \in \mathcal{U}} \; \varphi(x^i(t_+), v) \tag{16b}$$

where $u^i(t_+) := u(x^i(t_+))$. When $U = \mathbb{R}^{n_u}$, this coupled system can be solved as a set of nonlinear equations:

$$x^i(t_+) = x(t) + hf(x^i(t_+), u^i(t_+)) + \mathcal{O}(h^2) \tag{17a}$$
$$0 = \Phi(x^i(t_+), u^i(t_+)). \tag{17b}$$

The implicit scheme can also be expressed in terms of the DVI (12) as:

$$x^i(t_+) = x(t) + hf(x^i(t_+), u(x^i(t_+))) + \mathcal{O}(h^2) \tag{18a}$$
$$u(x^i(t_+)) \in \mathrm{SOL}(\mathcal{U}, \Phi(x^i(t_+), \cdot)). \tag{18b}$$

This representation indicates that the implicit sampled-data MPC scheme is an implicit Euler time-stepping scheme applied to the DVI (12). As shown in [9], the implicit Euler scheme converges to the solution of the DVI (to the optimal manifold) as $h \to 0$. As in the case of differential equations, it has also been observed that an implicit scheme applied to the DVI provides more numerical stability than the explicit counterpart as h increases [10]. In our context, this has the practical consequence that longer sampling intervals can be used. We demonstrate this is Section V.

We highlight that the proposed MPC scheme is standard in the sense that, in the limit of small sampling time, it becomes a continuous-time MPC formulation [11].

III. Predictor-Corrector Scheme

Because the explicit sampled-data MPC scheme is expected to suffer from low-quality approximations as the sampling interval increases and the implicit scheme is difficult to implement in practice, we consider an alternative *semi-explicit* scheme. In particular, we consider the following predictor-corrector scheme:

$$u^e(t_+) \in \mathrm{SOL}(\mathscr{U}, \Phi(x(t), \cdot)) \tag{19a}$$
$$x^e(t_+) = x(t) + \int_t^{t_+} f(x(\tau), u^e(t_+))d\tau \tag{19b}$$
$$u^{pc}(t_+) \in \mathrm{SOL}(\mathscr{U}, \Phi(x^e(t_+), \cdot)) \tag{19c}$$
$$x^{pc}(t_+) = x(t) + \int_t^{t_+} f(x(\tau), u^{pc}(t_+))d\tau. \tag{19d}$$

Here, a control action $u^e(t_+)$ is first estimated using a the current state $x(t)$ (as in the explicit scheme) and is refined using the predicted state $x^e(t_+)$. We note that this scheme can be implemented by using standard optimization solvers to evaluate $u^e(t_+) \in \mathrm{SOL}(\mathscr{U}, \Phi(x(t), \cdot))$ and $u^{pc}(t_+) \in \mathrm{SOL}(\mathscr{U}, \Phi(x^e(t_+), \cdot))$. As can be seen, the scheme involves a predictor step that is equivalent to that of explicit sampled-data MPC plus a corrector step that seeks to get closer to the implicit scheme solution. The predictor-corrector scheme is sketched in Figure III.

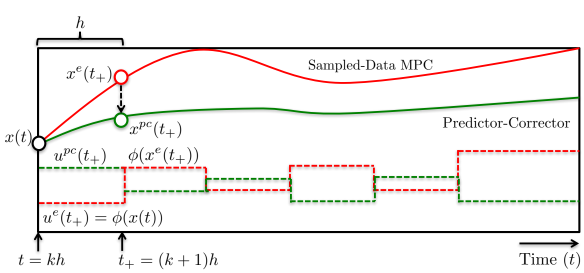

Fig. 2. Sketch of explicit and predictor-corrector sampled-data MPC schemes.

Because, $u^{pc}(t_+) = u(x^e(t_+))$, the evolution of the state under the semi-explicit scheme can also be written as:

$$\begin{aligned} x^{pc}(t_+) &= x(t) + \int_t^{t_+} f(x(\tau), u(x^e(t_+)))d\tau \\ &= x(t) + hf(x(t), u(x^e(t_+))) + \mathscr{O}(h^2). \end{aligned} \tag{20}$$

We now seek to prove that the state of the predictor-corrector scheme (19) is closer to the implicit scheme than the explicit scheme is. In other words, we aim to prove that the distance between $x^{pc}(t_+)$ and $x^i(t_+)$ is shorter than the distance between $x^{pc}(t_+)$ and $x^e(t_+)$.

We make the following assumptions:

- (A1) The dynamic mapping $f(\cdot,\cdot)$ is Lipschitz continuous in both arguments for all $x(t) \in \mathscr{X}, u(t) \in \mathscr{U}$, where $\mathscr{X}$ is a compact set.
- (A2) The optimal solution of problem (2) (given by $u(x(t))$) satisfies the linear independence constraint qualification and the second-order sufficient conditions.
- (A3) The mapping $\Phi(\cdot,\cdot)$ is Lipschitz continuous in both arguments for all $x(t) \in \mathscr{X}, u(t) \in \mathscr{U}$.

Assumption (A1) is standard and implies that $\|f(x_1(t), u_1(t)) - f(x_2(t), u_2(t))\| = \mathscr{O}(\|x_1(t) - x_2(t)\|) + \mathscr{O}(\|u_1(t) - u_2(t)\|)$. Assumption (A2) is required to ensure that the solution mapping $u(\cdot)$ of problem (2) is Lipschitz continuous in the state and thus $\|u(x_1(t)) - u(x_2(t))\| = \mathscr{O}(\|x_1(t) - x_2(t)\|)$ holds for all $x_2(t)$ in a non-empty neighborhood of $x_1(t)$ [8]. We elaborate on this result in Section IV.

Under these assumptions and definitions, we can establish the following result.

Theorem 1: If (A1) and (A2) hold, the states of the explicit, implicit, and predictor-corrector schemes satisfy:

$$||x^{pc}(t_+) - x^i(t_+)|| = \mathscr{O}(h)\,||x^e(t_+) - x^i(t_+)|| + \mathscr{O}(h^2)$$

Proof: We write the state evolution under explicit, implicit, and predictor-corrector schemes as:

$$\begin{aligned} x^e(t_+) &= x(t) + hf(x(t), u(x(t))) + \mathscr{O}(h^2) \\ x^i(t_+) &= x(t) + hf(x(t), u(x^i(t_+))) + \mathscr{O}(h^2) \\ x^{pc}(t_+) &= x(t) + hf(x(t), u(x^e(t_+))) + \mathscr{O}(h^2). \end{aligned}$$

From these relationships we can establish that:

$$\begin{aligned} &||x^{pc}(t_+) - x^i(t_+)|| \\ &= \mathscr{O}(||hf(x(t), u(x^i(t_+))) - hf(x(t), u(x^e(t_+)))||) + \mathscr{O}(h^2) \\ &= \mathscr{O}(h)\mathscr{O}(||x(t) - x(t)||) \\ &\quad + \mathscr{O}(h)\mathscr{O}(||u(x^i(t_+)) - u(x^e(t_+))||) + \mathscr{O}(h^2) \\ &= \mathscr{O}(h)\mathscr{O}(||u(x^i(t_+)) - u(x^e(t_+))||) + \mathscr{O}(h^2) \\ &= \mathscr{O}(h)\mathscr{O}(||x^i(t_+) - x^e(t_+)||) + \mathscr{O}(h^2). \end{aligned}$$

The proof is complete. □

The consequence of this result is that the predictor-corrector scheme (19) is $\mathscr{O}(h)$ closer to the solution of the implicit MPC scheme (15) compared to the explicit counterpart (13). This result indicates that we can expect better accuracy than explicit MPC. Ultimately, this enables us to use longer sampling intervals than sampled-data MPC and with this accommodate larger optimization problems. Moreover, the predictor-corrector scheme can be implemented by solving standard optimization problems, as opposed to the implicit scheme which requires the solution of a complicated variational inequality.

IV. An Approximate Predictor-Corrector Scheme

The solution of two optimization problems in the predictor-corrector scheme is clearly undesirable. Consequently, we seek approximations that can retain the approximation error of the predictor-corrector scheme while reducing latency. This can be done by exploiting well-known approximation results for generalized equations. To illustrate how to achieve this, we express the generalized equation (9) as:

$$0 \in \Phi(x,u) + \mathscr{N}_{\mathscr{U}}(u). \tag{21}$$

where x is the input data (the input state) and where the solution $u(x)$ (the control) being an implicit function of the data. Here, we drop the dependence of $x(t)$ to simplify notation. The solutions $u(x_1), u(x_2)$ thus satisfy:

$$0 \in \Phi(x_1,u(x_1)) + \mathscr{N}_{\mathscr{U}}(u(x_1)) \tag{22a}$$
$$0 \in \Phi(x_2,u(x_2)) + \mathscr{N}_{\mathscr{U}}(u(x_2)) \tag{22b}$$

Approximation results for generalized equations indicate that we can obtain an approximate solution of $u(x_2)$ from the solution v^* of a *perturbed* linearized generalized equation (linearized at $u(x_1)$):

$$0 \in \Phi(x_2,u(x_1)) + \nabla_u \Phi(x_1,u(x_1))(v-u(x_1)) + \mathscr{N}_{\mathscr{U}}(v) \tag{23}$$

The approximate solution is denoted as $\tilde{u}(x_2) := v^*$. Here, we note that $\nabla_u \Phi(\cdot,\cdot) = \nabla^2_{uu}\varphi(\cdot,\cdot)$. A solution of the linearized generalized equation can also be obtained from the solution Δv^* of the QP:

$$\min_{u(x_1)+\Delta v \in \mathscr{U}} \quad \frac{1}{2}\Delta v^T \nabla^2_{uu}\varphi(x_1,u(x_1))\Delta v + \nabla_u \varphi(x_2,u(x_1))^T \Delta v. \tag{24}$$

where $v^* = \Delta v^* + u(x_1)$.

The linearized generalized equation (23) can also be expressed in terms of a linearization of (21) around $(x_1,u(x_1))$:

$$r \in \Phi(x_1,u(x_1)) + \nabla_u \Phi(x_1,u(x_1))(v-u(x_1)) + \mathscr{N}_{\mathscr{U}}(v) \tag{25}$$

by defining $r = r_1 := \Phi(x_1,u(x_1)) - \Phi(x_2,u(x_1))$. Similarly, (22b) can also be written as (25) by defining $r = r_2 := \Phi(x_1,u(x_1)) + \nabla_u \Phi(x_1,u(x_1))(u(x_2)-u(x_1)) - \Phi(x_2,u(x_2))$. This observation is of relevance because, if (A2) holds, the solution of (25) is singled-valued and a Lipschitz continuous function of the perturbation r in a neighborhood of $(x_1,u(x_1))$. This property is known as *strong regularity*.

By denoting the solution of (25) as $\psi^{-1}[r]$ we thus have that Lipschitz continuity implies that $\|\psi^{-1}[r_1] - \psi^{-1}[r_2]\| = \mathscr{O}(\|r_1 - r_2\|)$ for any r_1, r_2 in a neighborhood of $r = 0$. Furthermore, we note that $\tilde{u}(x_2) = \psi^{-1}[r_1]$ and $u(x_2) = \psi^{-1}[r_2]$. Note also that $u(x_1) = \psi^{-1}[0]$ and thus $\|\psi^{-1}[r_2] - \|\psi^{-1}[0]\| = \mathscr{O}(\|r_2\|)$. This last observation can be used to establish that the solution of (21) is Lipschitz continuous of the data x and thus $\|u(x_1) - u(x_2)\| = \mathscr{O}(\|x_1 - x_2\|)$. This follows from:

$$\begin{aligned}
&\|u(x_1) - u(x_2)\| = \mathscr{O}(\|r_2\|) \\
&= \mathscr{O}(\|\Phi(x_1,u(x_1)) - \Phi(x_2,u(x_2))\| \\
&= \mathscr{O}(\|x_1 - x_2\|)
\end{aligned} \tag{26}$$

where the last equality follows from the Lipschitz continuity of the mapping $\Phi(\cdot,\cdot)$ (assumption (A3)). These results are established in a general setting in [8] and in the context of real-time optimization in [4]. We use these results to establish the following approximation result.

Lemma 1: Assume that (A2-A3) hold. The solution of the QP (24) $\tilde{u}(x_2)$ satisfies $\|\tilde{u}(x_2) - u(x_2)\| = \mathscr{O}(\|x_2 - x_1\|)$.

Proof: From the Lipschitz continuity of the solution of $\psi^{-1}[\cdot]$ and the definitions of r_1 and r_2 we have that:

$$\begin{aligned}
&\|\tilde{u}(x_2) - u(x_2)\| \\
&= \mathscr{O}(\|r_1 - r_2\|) \\
&= \mathscr{O}(\|\Phi(x_2,u(x_2)) - \Phi(x_2,u(x_1)) \\
&\qquad - \nabla_u \Phi(x_1,u(x_1))(u(x_2) - u(x_1))\|).
\end{aligned} \tag{27}$$

From the mean value theorem we have that

$$\begin{aligned}
&\Phi(x_2,u(x_1)) - \Phi(x_2,u(x_2)) \\
&= \int_0^1 \nabla_u \Phi(x_2, u(x_1) + \chi(u(x_2) - u(x_1)))(u(x_2) - u(x_1))d\chi
\end{aligned} \tag{28}$$

Consequently,

$$\begin{aligned}
\|\tilde{u}(x_2) - u(x_2)\| &= \mathscr{O}(\|u(x_2) - u(x_1)\|) \\
&= \mathscr{O}(\|x_1 - x_2\|),
\end{aligned} \tag{29}$$

where the last bound is obtained by using Lipschitz continuity of the solution of (21). The proof is complete. □

We can now establish properties for the *approximate predictor-corrector scheme*:

$$\tilde{x}^{pc}(t_+) = x(t) + hf(x(t), \tilde{u}(x^e(t_+))) + \mathscr{O}(h^2) \tag{30}$$

where $\tilde{u}(x^e(t_+))$ is obtained from the solution of (24) with $x_1 = x(t)$, $x_2 = x^e(t_+)$, and $\tilde{u}(x^e(t_+)) = u(x_1) + \Delta v^*$. We note that the predictor step $u^e(t_+) = u(x_1) = u(x(t))$ requires the full solution of an optimization problem while the corrector step is obtained by solving a QP obtained as a linearization around $x(t), u(x(t))$. We now show that this approximation does not affect the quality of the predictor-corrector scheme.

Theorem 2: Assume that (A1) and (A2) hold. The approximate predictor-corrector state $\tilde{x}^{pc}(t_+)$ satisfies:

$$\left|\left|\tilde{x}^{pc}(t_+) - x^i(t_+)\right|\right| = \mathscr{O}(h)\mathscr{O}(\left|\left|x^i(t_+) - x^e(t_+)\right|\right|) + \mathscr{O}(h^2). \tag{31}$$

Proof: By defining $x_1 = x(t)$ and $x_2 = x^e(t_+)$ and applying the result of Lemma 1 we obtain:

$$\begin{aligned}
&\|\tilde{u}(x^e(t_+)) - u(x^e(t_+))\| \\
&= \mathscr{O}(\|x^e(t_+) - x(t)\|^2) \\
&= \mathscr{O}(\|x(t) + hf(x(t), u(x(t))) + \mathscr{O}(h^2) - x(t)\|) \\
&= \mathscr{O}(h).
\end{aligned} \tag{32}$$

From this bound we can establish that the approximate predictor-corrector step

$$\tilde{x}^{pc}(t_+) = x(t) + hf(x(t), \tilde{u}(x^e(t_+))) + \mathscr{O}(h^2) \tag{33}$$

satisfies

$$
\begin{aligned}
&\|\tilde{x}^{pc}(t_+) - x^{pc}(t_+)\| \\
&= \|hf(x(t), \tilde{u}(x^e(t_+))) - hf(x(t), u(x^e(t_+)))\| + \mathcal{O}(h^2) \\
&= \mathcal{O}(h)\mathcal{O}(\|\tilde{u}(x^e(t_+))) - u(x^e(t_+))\|) + \mathcal{O}(h^2) \\
&= \mathcal{O}(h^2). \qquad (34)
\end{aligned}
$$

Consequently, by using the arguments of Theorem 1, we can establish that:

$$
\begin{aligned}
&\left|\left|\tilde{x}^{pc}(t_+) - x^i(t_+)\right|\right| \\
&= \mathcal{O}(\left|\left|x^{pc}(t_+) - x^i(t_+)\right|\right|) + \mathcal{O}(||\tilde{x}^{pc}(t_+) - x^{pc}(t_+)||) \\
&= \mathcal{O}(h)\mathcal{O}(\left|\left|x^i(t_+) - x^e(t_+)\right|\right|) + \mathcal{O}(h^2). \qquad (35)
\end{aligned}
$$

The proof is complete. □

The approximate predictor-corrector scheme retains the $\mathcal{O}(h)$ improvement over the explicit sampled-data scheme. From a practical stand-point this is relevant because it indicates that an optimization problem and a QP are needed to implement the scheme. One can also further consider using a QP approximation to compute $u(x(t))$ based on a linearized generalized equation obtained at $x(t-h)$. The analysis of this case is left as a topic of future work.

V. Illustrative Study

To illustrate the performance of different sampled-data schemes, we consider an optimal control problem of the form:

$$
\min_{v(\cdot)} \int_t^{t+T} \left((z(\tau) - \bar{z})^2 + v(\tau)\right) d\tau \qquad (36a)
$$

$$
\text{s.t. } \dot{z}(\tau) = (z(\tau) - \tilde{z}) + v(\tau),\ \tau \in [t, t+T] \qquad (36b)
$$

$$
z(t) = x(t). \qquad (36c)
$$

The solution at time t is given by the profile $v^*(\tau), \tau \in [t, t+T]$ from which we extract the control law $u(x(t)) = v^*(t)$. To obtain a finite-dimensional representation of this problem (which is needed to derive our DVI abstraction) we discretize the dynamics by using an implicit Euler scheme with a mesh that matches the sampling interval h. We note that this *internal* discretization scheme is not to be confused with the external time-stepping (sampled-data) schemes. The representation used in our DVI abstraction can also be obtained by propagating the dynamics in the cost function so that the model is only expressed in terms of a discrete-time finite dimensional vector $u \in \mathbb{R}^{n_u}$ that contains the entire piece-wise trajectory over $[t, t+T]$. In the results that we present next, we use the same internal discretization for all sampled-data schemes. With this, we seek to isolate the effect of the sampling interval.

The performance of the explicit, implicit, and predictor-corrector MPC schemes is illustrated in Figure 3. Here, we compare the performance of each scheme for different sampling intervals h. We use the same internal discretization mesh for all the schemes in order to focus our attention on the impact of the sampling scheme. We can see that the explicit sampling scheme induces larger errors as the sampling interval is increased. We also observe that the performance of the predictor-corrector scheme remains closer to the implicit scheme compared to the explicit counterpart. We do not compare the performance of the approximate predictor scheme because the proposed optimization problem is already a QP after internal discretization. In future work we will explore the performance of such scheme, which is is more difficult to implement.

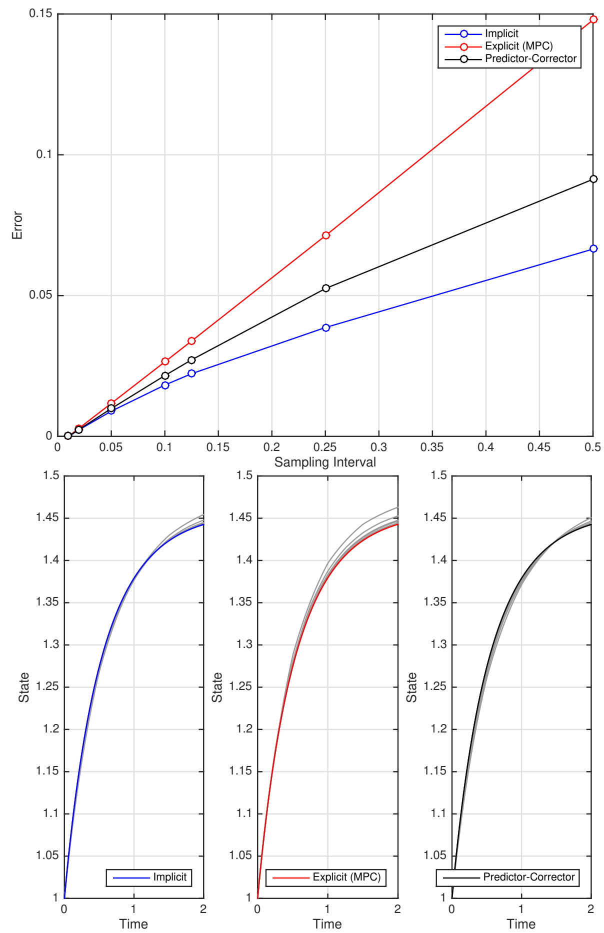

Fig. 3. Top: Manifold error for implicit, explicit, and predictor-corrector (MPC) schemes as a function of sampling interval. Bottom: State evolution for both schemes, solid color is the optimal manifold, and grey shades are approximations for various update intervals.

The results presented here are for a small illustrative example, so computational limitations are difficult to assess. In future work we will investigate more complex applications to highlight these effects.

VI. Conclusions and Future Work

We have shown that a dynamic system driven by the recursive real-time solution of optimization problems (as in MPC) can be cast as a differential variational inequality (DVI). The abstraction reveals that existing sampled-data MPC implementations correspond to an explicit Euler time-stepping scheme applied to the DVI. As expected, this explicit scheme induces large approximation errors when long sampling intervals are used. We propose to use a predictor-corrector scheme, which can be implemented by using standard optimization tools and can reduce approximation errors. As part of future work, we are interested in exploring approximate predictor-corrector schemes that only require QP solutions in order to further reduce computational complexity. Having a DVI abstraction also allows us to consider different time-stepping schemes and theoretical tools to perform error analysis. We are also interested in exploring more complicated formulations with different types of constraints (state and terminal) and cost functions.

ACKNOWLEDGEMENTS

This material is based upon work supported by the U.S. Department of Energy, Office of Science, under an Early Career Award and under Contract DE-AC02-06CH11357.

REFERENCES

[1] R. Findeisen, T. Raff, and F. Allgöwer, "Sampled-data nonlinear model predictive control for constrained continuous time systems," in *Advanced Strategies in Control Systems with Input and Output Constraints*. Springer, 2007, pp. 207–235.

[2] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer, "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations," *Journal of Process Control*, vol. 12, no. 4, pp. 577–585, 2002.

[3] M. Diehl, H. J. Ferreau, and N. Haverbeke, "Efficient numerical methods for nonlinear mpc and moving horizon estimation," in *Nonlinear model predictive control*. Springer, 2009, pp. 391–417.

[4] V. M. Zavala and M. Anitescu, "Real-time nonlinear optimization as a generalized equation," *SIAM Journal on Control and Optimization*, vol. 48, no. 8, pp. 5444–5467, 2010.

[5] A. L. Dontchev, M. Krastanov, R. T. Rockafellar, and V. M. Veliov, "An Euler–Newton continuation method for tracking solution trajectories of parametric variational inequalities," *SIAM Journal on Control and Optimization*, vol. 51, no. 3, pp. 1823–1840, 2013.

[6] A. Simonetto, A. Koppel, A. Mokhtari, G. Leus, and A. Ribeiro, "Prediction-correction methods for time-varying convex optimization," in *Signals, Systems and Computers, 2015 49th Asilomar Conference on*. IEEE, 2015, pp. 666–670.

[7] Q. T. Dinh, C. Savorgnan, and M. Diehl, "Adjoint-based predictor-corrector sequential convex programming for parametric nonlinear optimization," *SIAM Journal on Optimization*, vol. 22, no. 4, pp. 1258–1284, 2012.

[8] S. M. Robinson, "Strongly regular generalized equations," *Mathematics of Operations Research*, vol. 5, no. 1, pp. 43–62, 1980.

[9] J.-S. Pang and D. E. Stewart, "Differential variational inequalities," *Mathematical Programming*, vol. 113, no. 2, pp. 345–424, 2008.

[10] A. Dontchev and F. Lempio, "Difference methods for differential inclusions: a survey," *SIAM review*, vol. 34, no. 2, pp. 263–294, 1992.

[11] L. Grune, D. Nesic, and J. Pannek, "Model predictive control for nonlinear sampled-data systems," *Lecture notes in control and information sciences*, vol. 358, p. 105, 2007.