Real-Time Dynamic Optimization-based Advisory System for Batch Process Operation

Smriti Shyamal and Christopher L.E. Swartz*

Department of Chemical Engineering, McMaster University, 1280 Main Street West, Hamilton, Ontario, Canada, L8S 4L7

E-mail: swartzc@mcmaster.ca

Abstract

Batch and semi-batch process operations have a broad application in chemical engineering industry for producing multiple product grades using appropriate raw materials. Major batch processes are still subject to limited automation and decisions related to input amounts and timings are taken by the operators. This lead to sub-optimal batch operation due to complex behaviors and relationships between variables not being considered in the decision making. In this work, we introduce a real-time advisory system for batch/semi-batch processes to support the operator decision making in real-time. Economically optimal process operation is achieved by employing a first-principles dynamic model in the optimization formulation. The advisory system keeps itself cognizant of the on-line states by using a multi-rate Moving Horizon Estimator (MHE) that runs in parallel with the plant. Upon operator's request for a decision support, an on-call economics-based multi-tiered dynamic optimizer comes into action to recommend an input sequence for optimal performance. The optimizer effectively handles end-point constraints on product specifications or quantity by tackling infeasibilities through three tiers. The advisory system also warns the operator of the need to extend the batch while mitigating disturbances during the batch operation. Our

case studies on an Electric Arc Steelmaking operation demonstrate a major economic improvement when the dynamic optimization-based advisory system is used. We show that the online computational load is under 5 seconds on average when our proposed multi-tiered initialization scheme is used for solving the large-scale optimal control problems.

1 Introduction

Batch and semi-batch process operations are widely used in chemical engineering industry for production of a wide range of products such as pharmaceuticals,[1–4] steel,[5] polymers,[6] petroleum[7] and food products.[8,9] The limited duration of operation has posed multiple challenges related to control and optimization, different from continuous systems.[8] Batch operations are characterized by absence of steady state and significant variations in process conditions.[10] Thus, an economics-based objective function is more suitable for dynamic optimization. Constraints on the end product specifications or quantity (also called as terminal constraints) are very difficult to satisfy in particularity when the system is subject to multiple disturbances and challenging initial conditions.[11] Availability of limited senor measurements also cause problems when designing control strategies for batch/semi-batch processes such as Electric Arc Steelmaking.[12–14] For developing a real-time optimization scheme, availability of a first-principles model plays an important role as it provides the information related to future batch operation at any point in time during the process.[8] These issues directly effect the performance and application of an advanced closed-loop control algorithm such a Nonlinear Model Predictive Control (NMPC) for batch processes.[6,15,16] Thus, practical application of a completely automated system using a detailed first principles model is still limited in many industries where batch processes are employed. In practice, operators of complex semi-batch processes such as Electric Arc Furnace (EAF) and Basic Oxygen Furnace (BOF)[17] have shown reluctance to putting the plant in full closed-loop control. However, there is an opportunity of designing an advisory system where the opera-

tors have a choice weather to implement or not implement the control moves calculated by an economics-based real-time dynamic optimization.

Dynamic optimization aims to find the optimal control trajectories for a given time horizon when a dynamic process model is provided.[18] The model can either be a differential algebraic equation (DAE) or partial differential equation (PDE) system. The model consists of mathematical equations which can best describe the transient behavior of the plant. Besides the mathematical model, the dynamic optimization problem formulation has two other key ingredients: the performance criteria to maximize or minimize and the constraints. Any of the three direct approaches viz. simultaneous, sequential and multiple shooting can be used to solve the dynamic optimization problem. Since batch processes are of fixed length, a shrinking horizon dynamic optimization formulation is preferred.[19–23] The dynamic optimization can be called at any point during the batch operation to find optimal input moves for rest of the batch duration. An economics-based objective function aims to directly maximize profit while respecting the process constraints. Although constraints on final product quality at the end of the batch process can easily be incorporated, achieving a feasible solution of the dynamic optimization problem can be difficult. Also, following a input profile obtained from a single optimization can lead to end point constraints not getting satisfied. This is because disturbances further down the time horizon will not be accounted for. Hence, we need the input trajectories to be updated through re-optimizations.[24] A systematic real-time dynamic optimization algorithm is needed to assist the operators in maximizing economics while addressing the issue of terminal constraints.

State knowledge is necessary if a re-optimization needs to be carried out during the batch operation. The reconstructed states using the available measurements will provide the initial conditions for the dynamic model used in the dynamic optimization formulation. This is useful since a state estimator will give a close estimate of where the plant is at during the process

operation while incorporating any past effects due to disturbances.[25] Although many tools such as Extended Kalman Filter (EKF),[26,27] Unscented Kalman Filter (UKF),[28] particle filter etc. are used by researchers for state estimation, moving horizon estimation (MHE) has become popular over the last decade due to its straightforward way of handling constraints, use of a full-scale dynamic model in optimization and utilization of state-of-the-art numerical optimization algorithms for fast solution times.[29] MHE reconstructs the state by solving a dynamic optimization problem subject to the model and other specified constraints.[30] Since MHE uses a past fixed-sized window (that's why named 'moving horizon') of measurements, the problem size is kept under control. However, when problem size is not a major issue then an expanding horizon least-squares estimation can be used for some application.[6] Significant effort is usually directed to solve the MHE problems quickly so as to avoid feedback delays. It is to be noted that the use of a past history of measurements by MHE provides a straightforward way to use measurements with different sampling rates (multi-rate sampling).[31,32] Recently, expanding horizon MHE was coupled with NMPC for controlling the crystal size with the shape distribution in a batch crystallization process.[2] MHE has also been compared with other state estimators for a batch crystallization process.[33] Multi-rate MHE with NMPC was used to design a real-time energy management strategy for EAF operation.[12] The dynamic optimization problem os MHE has also been solved for EAF using the sequential approach of dynamic optimization.[34]

We have a particular interest in EAFs since it is a highly energy intensive semi-batch process and a real-time advisory system is envisaged to save huge operating costs. EAFs are widely used in the steel industry worldwide for production of steel by melting scrap steel collected from different sources. Approximately 25% of the world's steel production is using EAFs.[35] A single batch operation approximately consumes 400 kilowatt-hours/ton of steel produced.[36] Electric Arc Steelmaking is currently operated with a low level of process automation and control, and decision related to input amounts and timings are determined by

the operators. The desired product quality is achieved by altering the chemistry of scrap by addition of fluxes. EAF operating decisions are taken based on what has worked in the past. Although operator's experience is important for safe process operation, multi-variable interacting relationships are difficult to take into account. It is essential to develop model-based optimization strategies to support operator's decision making. Research works have focused on developing a first principles based model for EAF and then incorporating it an optimization framework.[5] Complex physical phenomenon such as melting, addition of energy and reagents, and changes in composition are described using detailed mathematical equations of the model. The model can be subsequently used in an optimization framework to determine optimal inputs. Since EAFs are characterized by availability of very few measurements, applying state estimation for an on-line implementation is very difficult. Moreover, the measurements have different sampling rates and unknown disturbances often effect the process operation. Extended Kalman Filter (EKF) was employed by Billings and coworkers[37] for estimating states when the EAF process is in the refining stage. Since the model had only 4 states, it is insufficient to capture the detailed process complexity. The high energy consumption by EAFs motivates the research and development of advanced control, estimation and optimization strategies for optimal operation. However, due to large fluctuations during the process, harsh operating conditions and the limited availability of measurements, real-time process optimization is very challenging. We are also interested in the end-point constraint which specifies that EAF should melt the charged scrap steel by a sufficient amount when the batch ends. In this work, we test the real-time advisory system to enhance the EAF operation and generate savings through optimal utilization of inputs.

In this article, we introduce a real-time dynamic optimization-based advisory system for batch/semi-batch plant operations. The advisory system aims to provide decision support to the operators by suggesting economics-based optimal input profiles and handles extension of batch length if end-point constraints are not met. It uses multi-rate MHE to calculate state

estimates at every sampling time, however, they are only sent to the multi-tiered optimizer if operator calls for decision support. The operator triggered framework is very useful for industries where a hybrid between open and closed-loop control is desired. We also describe a unique way to update the covariance matrix associated with the MHE arrival cost. The multi-tiered optimization is composed of three optimization tiers to handle in-feasibilities caused due to end-point constraints on product specifications and quantity. The hierarchical or multi-tiered approach[38] for prioritizing objectives in dynamic optimizations has been employed to design frameworks for optimal operation of plants during shutdowns.[39–42] The multi-tiered optimization strategy proposed in this paper utilizes a first principles model for prediction where the corresponding initial conditions are received from multi-rate MHE. We have also introduced a sub-tier to further handle the extensions related to terminal constraints. A shrinking horizon formulation with an economics-based objective function is used for the optimization tiers. A key bottleneck in implementing the real-time advisory system is the computation burden associated with solving the large scale dynamic optimization problems online. For obtaining fast solutions to these optimal control problems we employ the simultaneous dynamic optimization approach which transforms the problem into a sparse nonlinear-programming (NLP) problem by discretizing the states and inputs.[43] Furthermore, we have extended the MHE-NMPC initialization scheme proposed in[12] to develop a warm-start strategy for effective initialization of the dynamic optimization and MHE problems in the advisory system. Next, the effectiveness of the advisory system is tested for the highly energy intensive semi-batch EAF process using three case studies. We have chosen Implicit Euler scheme to discretize the DAE model. Finally, the convergence of MHE and computational results are described in details. Our results show a major improvement in profit obtained while the terminal constraints are handled systematically.

We have structured the remainder of the paper as follows: Section 2 introduces the dynamic optimization problem we are trying to solve with a brief description about model

contraction. Section 3 presents the multi-tired optimization where the working of the three tiers is explained. The multi-rate MHE problem along with the arrival cost update formulation is discussed in section 4. Section 5 describes the algorithmic working of the real-time dynamic optimization-based advisory system. The description of the sub-tier and the multi-tired initialization scheme is provided in this section. Section 6 deals with the application of the advisory system for Electric Arc Steelmaking process. We have also given a description about the process model and the implementation structure for the case studies discussed in this section. Section 7 concludes the paper by identifying future research directions.

2 Dynamic optimization problem

In this section, we describe the general form of dynamic optimization problem we are attempting to solve in this work.

2.1 Model formulation

A first principles-based model can be represented as the following explicit form of differential-algebraic equation (DAE) system,

$$\dot{\mathbf{x}}(t) = \mathbf{f}'_d(\mathbf{x}(t), \mathbf{z}'(t), \mathbf{u}(t), \mathbf{p}) \tag{1a}$$

$$\mathbf{0} = \mathbf{f}'_a(\mathbf{x}(t), \mathbf{z}'(t), \mathbf{u}(t), \mathbf{p}) \tag{1b}$$

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{z}'(t)) \tag{1c}$$

$$\mathbf{x}(t_i) = \mathbf{x}_{0i} \tag{1d}$$

$$t \in [t_i, t_f] \tag{1e}$$

where $\mathbf{x}$ represents the set of differential/state variables and $\mathbf{z}'$ is the set of algebraic variables. The process is described through differential equations $\mathbf{f}'_d$ and algebraic equations $\mathbf{f}'_a$. The inputs for the system are represented by $\mathbf{u}$. System parameters are denoted by $\mathbf{p}$. We

have provided additional equations $\mathbf{h}$ to clearly describe the evolution of system outputs $\mathbf{y}$ from initial time t_i to final time t_f when initial condition for the states $\mathbf{x}_{0i}$ is given.

The explicit DAE formulation described above is very general and has been used in literature to represent a wide variety of first principles model. Typically, inputs are parameterized either as piecewise constants or piecewise linear functions. The compact representation can be sometimes deceiving when the DAE system is very large scale. The size of vectors $\mathbf{x}$ and $\mathbf{z}'$ can be very large which in turn increases the number of model equations. The DAE models form an integral part of optimal control problems and it is recommended to modify the DAE system (variable and equation scaling etc.) while maintaining the model fidelity rather than using the raw equations directly. Sometimes the model size is so large and/or complicated that model reduction techniques need to employed. However, when maintaining the model fidelity is important, *model contraction* can be very useful. This is described in the subsequent subsection. It is to be noted that *model contraction* is entirely an optional step but we do suggest to try it and critically observe the system performance.

2.2 Model contraction

We describe *model contraction* as a procedure to eliminate a subset of algebraic variables by converting them into dependent variables $\mathbf{d}$. Such an elimination process will also remove

an equal number of algebraic equations.[44] The contracted model is given as,

$$\dot{\mathbf{x}}(t) = \mathbf{f}_d(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}) \tag{2a}$$

$$\mathbf{0} = \mathbf{f}_a(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}) \tag{2b}$$

$$\mathbf{d}(t) = \mathbf{f}_e(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}) \tag{2c}$$

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{z}(t), \mathbf{d}(t)) \tag{2d}$$

$$\mathbf{x}(t_i) = \mathbf{x}_{0i} \tag{2e}$$

$$t \in [t_i, t_f], \tag{2f}$$

where, $\mathbf{d} \subset \mathbf{z}$ and we have the modified set of algebraic variables for the contracted model as $\mathbf{z}' = \mathbf{d} \cup \mathbf{z}$. It is to be noted that the above model contraction preserves the model fidelity however, the model sparsity is sacrificed to obtain a more compact DAE model than the original one. A useful outcome of this procedure is that equations (2c) describing the evolution of $\mathbf{d}$ can be considered external to the model where $\mathbf{d}$ is post-calculated. Although we represent the new set of differential equations as $\mathbf{f}_d$, dimension of $\mathbf{f}_d$ is equal to dimension of $\mathbf{f}'_d$.

Remark: *Model contraction* is implemented in popular software packages such as CasADi and JModelica.[45] However, to implement it from scratch the user guides of CasADi and JModelica (and the references given there) are useful starting points.

2.3 Optimization formulation

Given the current states $\mathbf{x}_i$ provided by a state estimation strategy at time t_i, the dynamic optimization problem takes the following general form

$$\max_{\mathbf{u}(t),\, t\in[t_i,t_f]} \quad \Phi(t_f) := \int_{t_i}^{t_f} \psi(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p})dt \tag{3a}$$

$$\text{subject to} \quad \dot{\mathbf{x}}(t) = \mathbf{f}_d(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}), \tag{3b}$$

$$\mathbf{0} = \mathbf{f}_a(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}), \tag{3c}$$

$$\mathbf{u}^L \leq \mathbf{u}(t) \leq \mathbf{u}^U, \tag{3d}$$

$$\mathbf{x}^L \leq \mathbf{x}(t) \leq \mathbf{x}^U, \quad \forall t \in [t_i, t_f) \tag{3e}$$

$$\mathbf{x}_f^L \leq \mathbf{x}(t) \leq \mathbf{x}_f^U, \quad \forall t = t_f \tag{3f}$$

$$\mathbf{z}^L \leq \mathbf{z}(t) \leq \mathbf{z}^U, \tag{3g}$$

$$\mathbf{x}(t_i) = \mathbf{x}_i, \tag{3h}$$

where Φ represents the objective function. We seek to maximize its value at the final time t_f. The objective function value is obtained through integration of ψ over the time horizon $t \in [t_i, t_f]$. We aim to find the optimal inputs $\mathbf{u}$ so as to maximize Φ while respecting constraints on inputs (3e), algebraic variables (3g) and state variables (3e), (3f). (3f) is called end-point constraint since it bounds the state variable at final time t_f. The superscripts L and U represent the lower and upper bounds respectively. The model equations (3b) and (3c) act as strict equality constraints for the optimization problem. This general representation is very useful in optimizing different types of objective functions such as an economics-based objective function, set-point tracking MPC objective function etc.

A key bottleneck in solving the above dynamic optimization problem is the end-point constraint (3f). This is typically hard to satisfy since it is possible that with a given $\mathbf{x}_i$, no set of inputs can help land the states in the region $[\mathbf{x}_f^L, \mathbf{x}_f^U]$ at t_f. An immediate consequence can

be optimization solvers flagging an infeasibility output solve message leading to an overall halt of an advanced control procedure. This is highly undesirable from a practical implementation standpoint and an effective optimization strategy is needed to generate useful optimization results even when we are faced with such adverse situations. Development of a resilient strategy is necessary when a trivial removal of the end-point constraint is not acceptable. Having described the general dynamic optimization problem, in the next subsection we propose a novel multi-tiered optimization strategy for dynamic optimization with end-point constraints.

3 Multi-tiered Optimization

The dynamic optimization problem with end-point constraints is handled through multi-tiered optimization. The strategy incorporates three sequential dynamic optimization tiers where the last 2 tiers comes into effect when the previous tier fails to give a solution. An important characteristics of the strategy is that the three tiers always ensure that we arrive at a reasonable solution for the original optimization problem even though infeasibilities are caused due to end-point constraints. We describe each of the three tiers in the subsequent subsections.

3.1 Tier 1: Direct optimization

In this tier, we directly attempt to solve the original dynamic optimization problem given in (3a)-(3h). Any of the three dynamic optimization approaches viz. sequential, simultaneous and multiple-shooting can be employed to solve the problem. However, the NLP solver used in the three approaches are allowed to carry out the optimization up to a maximum number of iterations max_{iter}. This helps in terminating the solver optimization procedure within reasonable time. The max_{iter} is a tunable parameter and a suitable value can be chosen depending on the problem size and computational time needed for each solver iteration.

However, if the solver gives out an "optimal solution found" (or similar) message then we do not move to the next tier. If only we catch an "infeasible problem" or "maximum iterations reached" (or similar) solver message then we move to the tier 2.

3.2 Tier 2: Feasibility restoration through horizon extension

In tier 2, we reformulate the original dynamic optimization problem of Tier 1 so that we can restore feasibility. The time horizon $t \in [t_i, t_f]$ considered in Tier 1 is extended in integral steps. The extended problem for N_e integer time steps extension is given as

$$\max_{\mathbf{u}(t),\, t\in[t_i, t_f+N_e\Delta T]} \quad \Phi(t_f + N_e\Delta T) := \int_{t_i}^{t_f+N_e\Delta T} \psi(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p})dt \tag{4a}$$

$$\text{subject to} \quad \text{Extended set of equations :}$$

$$(3b) - (3d), (3g), (3h), \tag{4b}$$

$$\mathbf{x}^L \leq \mathbf{x}(t) \leq \mathbf{x}^U, \quad \forall t \in [t_i, t_f + N_e\Delta T) \tag{4c}$$

$$\mathbf{x}_f^L \leq \mathbf{x}(t) \leq \mathbf{x}_f^U, \quad \forall t = t_f + N_e\Delta T. \tag{4d}$$

This extension provides additional inputs for manipulation by dynamic optimization. These additional degrees of freedom provide a straightforward way to restore feasibility. We describe a serial and a parallel implementation for Tier 2:

- **Serial implementation**: First, a 1 time step extension problem ($N_e = 1$) is formulated and solved. If it is infeasible or we hit max_{iter} then a second problem where $N_e = 2$ is solved. This sequence of problem formulation and an attempt to solve it is continued only up to a limited number of time step extensions N_{ext}. This make sure that we are only solving a countable number of optimization problems in Tier 2. If we keep getting infeasibility or max iteration exit messages then we goto Tier 3, otherwise we stop in Tier 2 and report the solution obtained with the associated extension (number of time steps: $N_{e_{min}}$ value) required to restore feasibility.

- **Parallel implementation**: We formulate and solve N_{ext} extended problems in parallel. This effectively uses available parallel computational architecture. Solver exit messages for all the N_{ext} solves are collected and then if none of them is solved to full optimality then we move to Tier 3. If 1 or more problems are solved then we choose the extension $N_{e_{min}}$ associated with the solve with minimum extension value.

Both the implementations attempt to find the minimum integral extension required to restore feasibility. If feasibility in not restored then we move to the final Tier 3 which is described below.

3.3 Tier 3: Feasibility restoration through horizon extension and end-point relaxation

In this tier we reformulate the extended problem of Tier 2 by removing the end-point constraint. We instead minimize the state variable deviation at $t_f + N_{ext}\Delta T$ from the end-point upper $\mathbf{x}_f^U$ and lower bounds $\mathbf{x}_f^L$. The reformulated problem is given as

$$\max_{\mathbf{u}(t),\, t\in[t_i, t_f+N_{ext}\Delta T]} \quad \Phi(t_f + N_{ext}\Delta T) := \int_{t_i}^{t_f+N_{ext}\Delta T} \psi(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p})dt \\ - ||\mathbf{x}(t_f + N_{ext}\Delta T) - \mathbf{x}_f^U||^2_{D^{-1}} - ||\mathbf{x}(t_f + N_{ext}\Delta T) - \mathbf{x}_f^L||^2_{F^{-1}} \tag{5a}$$

$$\text{subject to} \quad (4b), (4c) \tag{5b}$$

where the modified objective function contains two additional least squares terms to minimize end-point state deviations. The weighting for each of the state variables in both the terms is given by diagonal tuning matrices D and F. This relaxed problem effectively restores the feasibility and we stop at Tier 3. Thus, the use of three tiers make sure that we arrive at a reasonable solution to the original problem in circumstances where the original problem is not feasible. The solutions thus obtained from either of the three tiers can be implemented

on the plant. The working of the three tiers is summarized in Fig. 1.

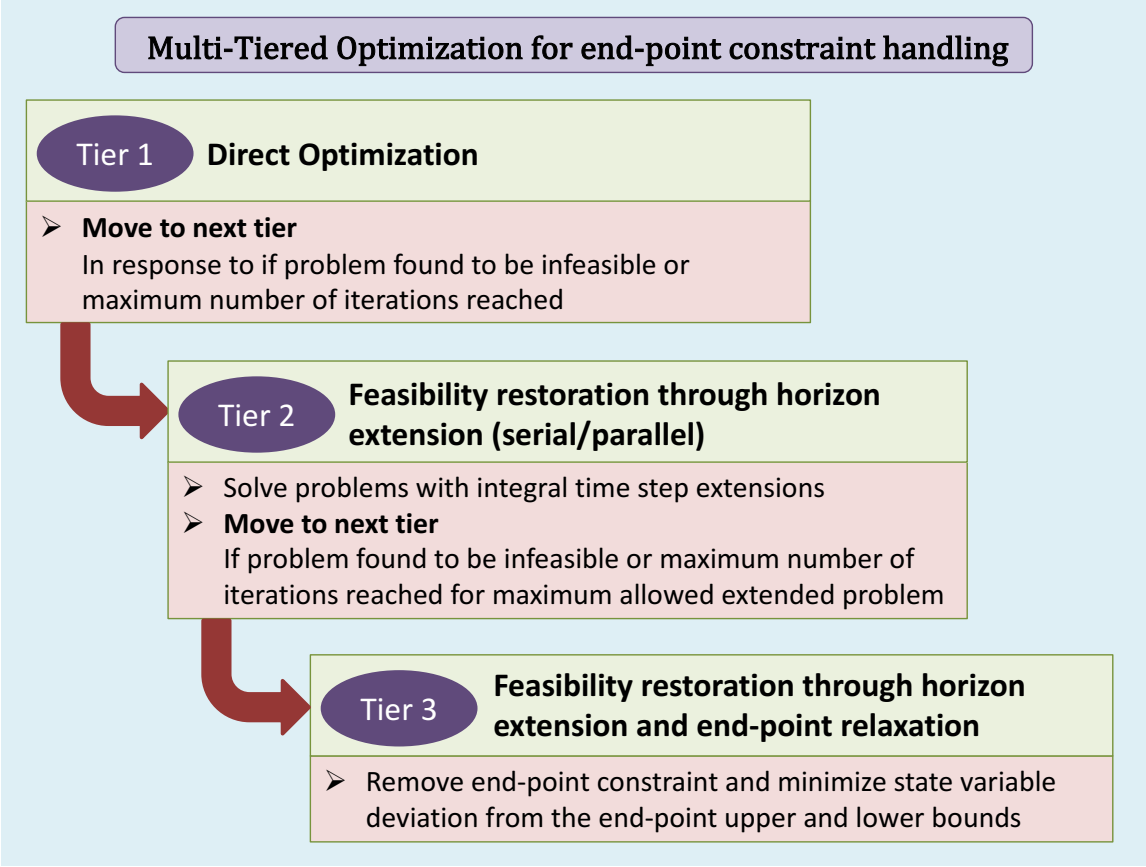

Figure 1: Multi-tiered optimization strategy: Movement from tier-to-tier.

4 Multi-rate Moving Horizon Estimation

For carrying out a multi-tiered optimization at any time step t_i, it is important to reconstruct the states $\mathbf{x}_i$ using the available measurements up to the current time. Different state estimation techniques such as variants of Kalman Filters (Extended Kalman Filter, Unscented Kalman Filter), Particle filters, MHE etc. can be employed. MHE is particularly attractive due to its direct handling of nonlinear system model, easy incorporation of constraints[30,46,47] and straightforward handling of multi-rate measurements. Since MHE solves a dynamic optimization problem to determine the state estimates, state-of-the-art nonlinear programming (NLP) solvers[34,48–50] can be used to arrive at the solutions quickly.

Additionally, MHE uses a past history of measurements which provides a clear means to incorporated measurements arriving at different sampling rates.[31,32,34,51,52] The use of all the available measurements potentially leads to an increase in the system observability and also a reduction in estimation errors.[31,32] The next subsections describes the model used in the MHE formulation, the least-squares problem and the arrival cost calculation.

4.1 Model for State Estimation

We consider the given process described using the following stochastic DAE system

$$\dot{\mathbf{x}}(t) = \mathbf{f}_d(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}) + \mathbf{w}_t(t) \tag{6a}$$

$$\mathbf{0} = \mathbf{f}_a(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}) \tag{6b}$$

$$\mathbf{d}(t) = \mathbf{f}_e(\mathbf{x}(t), \mathbf{z}(t), \mathbf{u}(t), \mathbf{p}) \tag{6c}$$

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \mathbf{z}(t), \mathbf{d}(t)) + \mathbf{v}_t(t) \tag{6d}$$

where $\mathbf{w}_t$ represents the errors in the differential equations called as process noise. $\mathbf{w}_t$ is useful in modeling real plants where the process is often corrupted due to disturbances and noises. (6d) denotes the measurement equation where $\mathbf{v}_t$ accounts for measurement uncertainty and represents the difference between the predicted measurements and the actual sensor measurements $\mathbf{y}(t)$. Typically both $\mathbf{w}_t$ and $\mathbf{v}_t$ are modeled as normally distributed uncorrelated sequence of random variables.

Discretization of the above DAE system is then carried out for subsequent computations. Since discretization is performed according to the optimization strategy employed (simultaneous, multiple shooting), we therefore consider here the conceptual discretization of the DAE system using a zero-order hold on $\mathbf{v}_t$. We also assume that inputs $\mathbf{u}$ are only manipulated at discrete time points k. The subscript k is now introduced here to distinguish variable values at discrete time points from continuous values. Thus, using (6a) and (6b),

and removing $\mathbf{p}$ for ease of representation., we get

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \int_k^{k+1} \mathbf{f}_d(\mathbf{x}(t), \mathbf{u}_k)dt + \int_k^{k+1} \mathbf{w}_t(t)dt, \tag{7}$$

which is represented as

$$\mathbf{x}_{k+1} = \mathbf{F}_d(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k. \tag{8}$$

We also get the discrete time measurement equation as

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k. \tag{9}$$

We note here that (8) represents a closed form solution of the DAE system which is not always possible to get. Here we view it only in an abstract form and thus the algebraic and dependent variables/equations are eliminated. Usually, k is chosen such that $t_k = k\Delta T$ where ΔT is the sampling period.

4.2 Constrained Least-Squares Formulation

We consider the process at a given time instant t_i with a past history of measurements and inputs up to t_i available to us. Rather than using all the measurement and input information, MHE considers a moving window of fixed size of N sampling times. Here, we write the MHE formulation for a multi-rate measurement structure. The key assumptions are that the time points of slow measurements coincide with that of the fast ones and there are no delays associated with availability of measurements. The vector of fast measurements are denoted as $\mathbf{y}_k^F$ while $\mathbf{y}_k^{SF}$ represents a measurement vector containing both the slow and fast measurements. So, a measurement structure for MHE can be represented as, for example, $\{\mathbf{y}_{i-N}^{SF}, \mathbf{y}_{i-N+1}^{F}, \mathbf{y}_{i-N+2}^{SF}, \ldots, \mathbf{y}_{i-1}^{F}, \mathbf{y}_{i}^{SF}\}$. In the example, at time instants $[t_{i-N}, t_{i-N+2}, t_i]$ both the slow and fast measurements are available where as only fast measurements are available

at time instants $[t_{i-N+1}, t_{i-1}]$. The measurement structure of the example can easily be modified to accommodate different multi-rate sampling. As we march along the time, we add the new set of measurements to the structure and drop the earliest set. This feature of MHE restricts the growth of MHE optimization problem. Also, if increase in problem size is not an issue then an expanding horizon problem can be considered to avoid loss of information due to moving horizon.

The multi-rate MHE problem for an observable system is given as

$$\min_{\mathbf{x}_{i-N}, \mathbf{w}_k} \sum_{k=i-N}^{i-1} ||\mathbf{w}_k||^2_{Q^{-1}} + \sum_{\substack{k=i-N \\ k\in\mathbb{I}_F}}^{i} ||\mathbf{v}^F_k||^2_{(R^F)^{-1}} + \sum_{\substack{k=i-N \\ k\in\mathbb{I}_{SF}}}^{i} ||\mathbf{v}^{SF}_k||^2_{(R^{SF})^{-1}} + ||\mathbf{x}_{i-N} - \hat{\mathbf{x}}_{i-N}||^2_{S_i^{-1}} \tag{10}$$

$$\text{s.t. } \mathbf{x}_{k+1} = \mathbf{F}_d(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \quad k = i-N, \ldots, i-1 \tag{11a}$$

$$\mathbf{y}^F_k = \mathbf{h}^F_k(\mathbf{x}_k) + \mathbf{v}^F_k, \qquad k \in \mathbb{I}_F \tag{11b}$$

$$\mathbf{y}^{SF}_k = \mathbf{h}^{SF}_k(\mathbf{x}_k) + \mathbf{v}^{SF}_k, \quad k \in \mathbb{I}_{SF} \tag{11c}$$

$$\mathbf{x}^{LB} \leq \mathbf{x_k} \leq \mathbf{x}^{UB}, \tag{11d}$$

$$\mathbf{w}^{LB} \leq \mathbf{w_k} \leq \mathbf{w}^{UB}, \tag{11e}$$

where $\mathbb{I}_F$ and $\mathbb{I}_{SF}$ represent the sets of sampling times where only fast measurements (super script F) and both slow and fast measurements (superscript SF) are available respectively. $\hat{\mathbf{x}}_{i-N}$ denotes the *a priori* estimate of the state variable value at the earliest time point in the moving horizon. We represent two separate measurements functions $\mathbf{h}^F_k$ and $\mathbf{h}^{SF}_k$ to map the states $\mathbf{x}_k$ to the sensor measurements $\mathbf{y}^F_k$ and $\mathbf{y}^{SF}_k$ respectively. Similarly, $\mathbf{v}^F_k$ and $\mathbf{v}^{SF}_k$ are the associated measurement noises. (11d) and (11e) provides the lower $[\mathbf{x}^{LB}, \mathbf{w}^{LB}]$ and

upper bounds $[\mathbf{x}^{UB}, \mathbf{w}^{UB}]$ on the state variables $\mathbf{x}$ and model noises $\mathbf{w}$ respectively. The covariance matrices associated with the model noise, measurement noise (for F and SF) and initial state error is denoted here as Q, R^F, R^{SF} and S_i respectively. The state estimates $\mathbf{x}_i$ at t_i is obtained by solving the least squares problem.

It can be observed that the MHE least squares objective function is composed of four terms where the first three are used to minimize weighted model and measurement errors. The fourth term is called the arrival cost or cost to arrive which minimizes the initial state discrepancy. The cost to arrive term is important because we are dropping measurements due to the moving nature of MHE. The initial state error covariance is also generally updated every time step thus the subscript i is used for S matrix. In the next subsection we describe the extended Kalman filter covariance update formula[30] which is commonly used for the purpose. However, the update is entirely not necessary when the system is strongly observable.[47]

4.3 Arrival Cost Update

To update arrival cost covariance matrix S using Extended Kalman Filter (EKF) update formula we need a discrete time model linearized at every sampling time. We first consider the continuous time stochastic DAE model (6). Linearizing (6) around a given time t_k gives (suppressing $\mathbf{p}$ for ease of readability)

$$\dot{\mathbf{x}}(t) = P_k\mathbf{x}(t) + J_k\mathbf{z}(t) + K_k\mathbf{u}(t) + \mathbf{w}_t(t) \tag{12a}$$

$$\mathbf{0} = G_k\mathbf{x}(t) + H_k\mathbf{z}(t) + L_k\mathbf{u}(t), \tag{12b}$$

$$\mathbf{d}(t) = O_k\mathbf{x}(t) + T_k\mathbf{z}(t) + V_k\mathbf{u}(t), \tag{12c}$$

$$\mathbf{y}(t) = E_k\mathbf{x}(t) + F_k\mathbf{z}(t) + M_k\mathbf{d}(t) + \mathbf{v}_t(t). \tag{12d}$$

Here, matrices $P_k, J_k, K_k, G_k, H_k, L_x, O_k, T_k, V_k, E_k, F_k$ and M_k are generally obtained by computing respective Jacobian values at the linearized point.[53] Although, now a discretization can be carried out by an appropriate integration of the continuous system through a zero-order hold on $\mathbf{u}(t)$ and $\mathbf{v}_t(t)$, this transformation involves computing matrix exponential. For large scale systems where the system matrices are prone to be ill-conditional, we employ an alternate way of transformation which uses Explicit Euler scheme for discretization. Although the approach is less accurate, we can completely avoid the complications associated with calculating matrix exponentials. Using Explicit Euler scheme, we get the discretized form as

$$\frac{\mathbf{x}_{k+1} - \mathbf{x}_k}{\delta_1} = P_k\mathbf{x}_k + J_k\mathbf{z}_k + K_k\mathbf{u}_k + \mathbf{w}_k, \tag{13a}$$

$$\mathbf{0} = G_k\mathbf{x}_k + H_k\mathbf{z}_k + L_k\mathbf{u}_k, \tag{13b}$$

$$\mathbf{d}_k = O_k\mathbf{x}_k + T_k\mathbf{z}_k + V_k\mathbf{u}_k, \tag{13c}$$

$$\mathbf{y}_k = E_k\mathbf{x}_k + F_k\mathbf{z}_k + M_k\mathbf{d}_k + \mathbf{v}_k, \tag{13d}$$

where δ_1 denotes the Explicit Euler step size. Now, we use (13b) to eliminate $\mathbf{z}_k$ from (13a), (13c) and (13d). Further, using (13c) and rearranging terms gives the discretized state space model to be used for EKF updates,

$$\mathbf{x}_{k+1} = \delta_1(P_k - J_k{H_k}^{-1}G_k + I)\mathbf{x}_k + \delta_1(K_k - J_k{H_k}^{-1}L_k)\mathbf{u}_k + \mathbf{w}_k \tag{14a}$$

$$\begin{aligned}\mathbf{y}_k = (E_k - F_k{H_k}^{-1}G_k - M_kT_k{H_k}^{-1}G_k + M_kO_k)\mathbf{x}_k \\ + (M_kV_k - F_k{H_k}^{-1}L_k - M_kT_k{H_k}^{-1}L_k)\mathbf{u}_k + \mathbf{v}_k\end{aligned}$$

where I is an identity matrix of both dimensions equal to number of states. Replacing, $[\delta_1(P_k - J_k{H_k}^{-1}G_k + I)]$, $[\delta_1(K_k - J_k{H_k}^{-1}L_k)]$, $[(E_k - F_k{H_k}^{-1}G_k - M_kT_k{H_k}^{-1}G_k + M_kO_k)]$ and $[(M_kV_k - F_k{H_k}^{-1}L_k - M_kT_k{H_k}^{-1}L_k)]$ with A_k, B_k, C_k and D_k respectively, we thus have

the time varying discrete state space model in the regular form as

$$\begin{aligned}\mathbf{x}_{k+1} &= A_k\mathbf{x}_k + B_k\mathbf{u}_k + \mathbf{w}_k \\ \mathbf{y}_k &= C_k\mathbf{x}_k + D_k\mathbf{u}_k + \mathbf{v}_k.\end{aligned} \tag{15}$$

For the above system at a given time, an EKF covariance propagation equation is given as,[30]

$$S_{k+1} = Q + A_k[S_k - S_kC_k^T(R + C_kS_kC_k^T)^{-1}C_kS_k]A_k^{-1}. \tag{16}$$

Although, the superscripts 'SF' and 'F' have been suppressed in (16) for R and C, and in (15), appropriate dimensions according to the multi-rate measurement structure needs to be considered. It is to be noted that other filters such as constrained particle filter[54] and unscented KF[55] can also be employed to update S_i. The update is not carried for batch process with sampling times t_i ($i \leq N$), since measurements are not dropped.

4.4 Disturbance handling

Unmeasured disturbances effect the MHE performance in an adverse manner. To tackle the uncertainties in a more direct manner, the model states are augmented with disturbance states $\mathbf{d}_k$ to construct a modified system for MHE

$$\mathbf{x}_{k+1} = \mathbf{F}_d(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{B}_d\mathbf{d}_k + \mathbf{w}_k, \tag{17a}$$

$$\mathbf{y}_k^F = \mathbf{h}_k^F(\mathbf{x}_k) + \mathbf{v}_k^F, \tag{17b}$$

$$\mathbf{y}_k^{SF} = \mathbf{h}_k^{SF}(\mathbf{x}_k) + \mathbf{v}_k^{SF}, \tag{17c}$$

$$\mathbf{d}_{k+1} = \mathbf{d}_k + \mathbf{w}_{dk}, \tag{17d}$$

where $\mathbf{d}_k$ is driven by a new set of white noise variable $\mathbf{w}_{dk} \sim \mathcal{N}(0, \mathbf{Q}_d)$. A well chosen $\mathbf{B}_d$ ensures that the corresponding state equations have the right impact due to the movement of the disturbance states $\mathbf{d}_k$. More details on the disturbance model and its effect are discussed in.[31]

5 Real-time Advisory System

A number of batch and semi-batch process are currently operated with limited automation and operators take decisions related to input amounts to be put in as well as their respective timings during the batch. Our development is focused on improving this suboptimal operation where more informed decisions can be taken based on a full scale first principles model. Implementing a large scale model for decision support is very challenging owing to the formulation and implementation complexities. In this section we will describe the advisory system in details while providing the mathematics working in the background.

5.1 Algorithmic working

The main objective of the real-time advisory system is to provide a input sequence as decision support to process operators when needed (see Fig. 2). The need is to tied to the operators calling for support and this is the only manual decision component involved in the working of the advisory system. Since the input sequence computed by the system is based on a rigorous optimization involving a high fidelity first-principles model, it takes complex relationships related to the physical phenomenon in to account when deciding on the optimal way to meet the economic objective. A single optimization at the beginning of the batch seems to be a logical way forward but with initial conditions being unknown and multiples disturbances effecting the process, re-optimizations based on current conditions can only provide the right decision support. Since batch/semi-batch processes are characterized by frequent fluctuations due to various additions and difficult operating conditions, an advisory

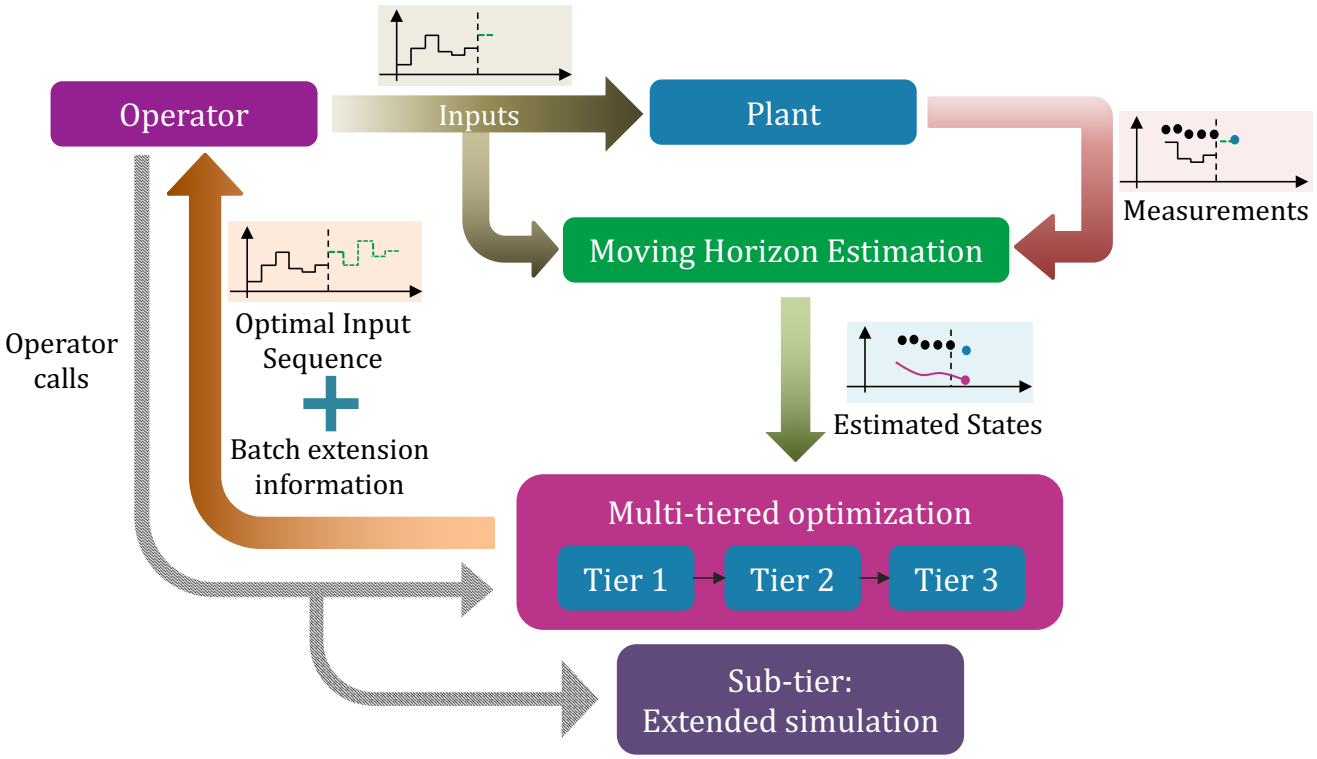

Figure 2: Real-time advisory system.

tool must be capable of handling volatile disturbances. Our advisory system directly targets economics by effectively solving optimal control problems in real-time. Such an implementation for a batch/semi-batch process is further complicated by the natural objective to get a specified amount of desired product by the batch end. Unpredicted disturbances or operator not applying the recommended inputs may lead to a sizable part of the initial reactants getting left or product specifications not met, which is not favorable. So, we have made our advisory system capable of providing information related to lengthening of batch if required in those particular cases, so that operators are aware of the predicted impact and can take remedial action at an early stage.

A key component of the advisory system is multi-rate MHE which runs in parallel with the plant. MHE reconstructs the state vector at each sampling time t_i so that the advisory system is aware of the current state of the process. MHE tackles the three issues viz.

unknown initial condition of the state variables, disturbances acting on the process and plant-model mismatch. A well-tuned MHE is expected to quickly converge to the true plant states while mitigating the uncertainties after a reasonable number of sampling times. MHE solves a least squares problem by employing the nonlinear model directly to arrive at the state estimates. The advisory system uses a multi-tiered initialization scheme in combination with sate-of-the-art NLP solvers to obtain fast solutions to the MHE dynamic optimization problem at each sampling. Additionally, multi-rate measurements associated with the process are taken into account by the multi-rate MHE.

The state estimates are subsequently provided to an economics-based multi-tired optimizer to calculate the optimal input sequence to maximize the profit. We first describe the shrinking horizon approach which is used here to consider evolution of states throughout the entire batch when arriving at the solution, instead of a fixed rolling horizon. Shrinking horizon problems are characterized by a fixed end point time while the current time (where each optimization is solved) only moves. Although a large problem is expected at the start times, problem size decreases linearly as we march along the time. Given the current $\mathbf{x}_i$ states, the optimizer can find an economically optimal set of inputs trajectories $\mathbf{U}_i = \{\mathbf{u_i}, \mathbf{u_{i+1}}, \ldots, \mathbf{u_{f-1}}\}$ at $t_i \in [0, t_f - 1]$ so as to achieve the maximum profit. However, the optimizer is called upon only when the operator triggers the advisory system at $t_i \in [0, t_f - 1]$. The operator may choose to implement the input recommendation up-to a certain time $t_k \in \{t_i, t_{i+1}, \ldots t_{f-1}\}$ by implementing only the initial subset of the input moves $\mathbf{U}_{i,op} = \{\mathbf{u_i}, \mathbf{u_{i+1}}, \ldots, \mathbf{u_{k-1}}\}$ and then recall the advisory system at t_k. However, the operator may choose to deviate from $\mathbf{U}_i$ before the recall by incorporating manually decided input moves in the implemented control sequence, for example, $\mathbf{U}_{i,op} = \{\mathbf{u_i}, \mathbf{u}_{i+1}, \mathbf{u_{op,i+2}}, \ldots, \mathbf{u_{op,k-1}}\}$, where subscript *op* represents the input moves selected by the operator. This flexibility is well suited for actual industrial practice where multiple external factors are important to operate a process. However, it may be argued to incorporate all the possible effects when designing an advisory system, it

is practically not always feasible. Our advisory system considers these global effects impacting the plant indirectly though manual intermittent operation.

Shrinking horizon strategy is not adequate when the plant is at a specific state x_i at t_i from where it is not possible to produce products within a specified tolerance $\delta_{product}$ by the fixed end point of the shrinking horizon t_f. Since inputs moves can only be varied by the optimizer within specified bounds while respecting the end point constraint, infeasibilities are likely to occur. The problematic states x_i are reached due to disturbances impacting the plant or due to the prior input moves implemented by the operator. To ensure we recover from such infeasible solves, our algorithm uses the multi-tiered optimization algorithm which considers extension of horizon and/or relaxing end point constraint to obtain the optimal decision support.

5.2 Sub-tier: Extended implementation

It is certainly possible that implementation of either of the three solutions obtained from the three tiers cannot actually ensure that the state variable value $\mathbf{x}_{plant}(t_{f_{tier}})$ at the final time $t_{f_{tier}}$ (t_f if Tier 1, $(t_f + N_{e_{min}}\Delta T)$ if Tier 2, $(t_f + N_{ext}\Delta T)$ if Tier 3) is within specified bounds. This is due to disturbances effecting the plant. In order to mitigate such a possibility we propose a Sub-tier: Extended implementation. This sub-tier gets activated at $t_{f_{tier}}$ when estimated states $\mathbf{x}_{est}(t_{f_{tier}})$ at $t_{f_{tier}}$ in not within the specified upper and lower bounds. The state estimates can be obtained using a state estimation algorithm such as Kalman Filter, MHE etc. We discussed MHE in the previous section 4 of the paper in more details.

In this sub-tier we do not carry out optimization but instead we re-implement the last implemented input moves to the plant. The reimplementation is carried out again if the state estimates at the current time is not within bounds. We also put an upper limit N_{extra} on these re-implementations to ensure that the extended implementation in of finite nature.

This sub-tier is an integrated part of the three tier main algorithm which connect it to a practical implementation on a plant. This sub-tier algorithm shows that not only an estimate of the current states is useful for solving control problems but can also be used to design a decision support algorithm.

5.3 Multi-tiered initialization strategy

For decision support to be available to the operators in real-time, solving the optimization problems of the tiers and MHE problem quickly is really important. Since, subsequent MHE and optimization problems have similar structures, it is useful to use the past solves effectively to generate good warm start points for the current solves. Warm-start strategies have been successfully employed by multiple researchers to solve MHE and NMPC problem on-line.[56–59] In a recent work, Shyamal and Swartz[12] proposed a novel initialization scheme to solve on-line MHE-NMPC problems quickly to optimality by effectively using the time between sample times. Here, we adapt the scheme for our advisory system application. We solve predicted optimization problems in background and then use the solutions for warm-starting the actual problems. The proposed multi-tiered initialization strategy generates warm-start points for both the primal and dual variables.

We consider the current time instant (on-line time) as t_i where the state estimates $\mathbf{x}_k$ and the inputs applied to the plant $\mathbf{u}_k$ are available. The time available to us before the next sampling time t_{i+1} is called here as background time. The superscripts for slow and fast measurements are suppressed here for ease of readability. Now, we introduce the multi-tiered initialization strategy for the real-time advisory system:

In background, between t_k and t_{k+1}:

- **Tier 1**: While keeping $\mathbf{w}_k$ as 0, carry out a disturbance-free forward model simulation $\bar{\mathbf{x}}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k)$ using $\mathbf{x}_k$ and $\mathbf{u}_k$. Obtain the predicted measurement $\bar{\mathbf{y}}_{k+1} = \mathbf{h}(\bar{\mathbf{x}}_{k+1})$

and pass it to tier 2.

- **Tier 2**: Use $\bar{\mathbf{y}}_{k+1}$ to construct a predicted MHE problem. Solve it and obtain the predicted state estimates $\tilde{\mathbf{x}}_{k+1}$ to be passed to tier 3. Also, store the optimal primal and dual values $\tilde{s}_k^{mhe}$.

- **Tier 3**: Use $\tilde{\mathbf{x}}_{k+1}$ to construct and solve optimization problems in the multi-tiered optimization strategy. Store the solutions obtained from 3/2/1 optimization tiers as $\tilde{s}_k^{opt1}/\tilde{s}_k^{opt2}/\tilde{s}_k^{opt3}$ (both primal and dual).

On-line, at t_{k+1}:

- **Tier 1**: Obtain the current plant measurements $\mathbf{y}_{k+1}$ and construct the actual MHE problem. Use $\tilde{s}_k^{mhe}$ to warm start the problem; solve it to get the actual current state estimates $\mathbf{x}_{k+1}$ and pass it to tier 2.

  If operator calls the advisory system for decision support go to tier 2 other wise return to the background step.

- **Tier 2**: Construct and solve optimization problems associated with the multi-tiered optimization strategy by using $\tilde{s}_k^{opt1}/\tilde{s}_k^{opt2}/\tilde{s}_k^{opt3}$ for initialization. Obtain the optimal input sequence $\{\mathbf{u}_{k+1}, \ldots, \mathbf{u}_{t_f}\}$ and send it to the operator. Return to the background step.

The usefulness of the multi-tiered strategy depends on how close are predicted and actual problems. Also, the strategy can effectively implemented for any of the three dynamic optimization approaches viz. single shooting, multiple shooting and simultaneous (details of the approaches in.[18] The multi-tiered initialization helps in quickly solving the optimization problems in the tiers of the multi-tiered optimization strategy.

6 Application to Electric Arc Furnace Steel-making Operation

In this section we test the real-time advisory system for Electric Arc Steelmaking semi-batch process. Melting of steel being a highly energy intense batch process, the required energy is provided in form of electrical and chemical energy. The electrical energy is supplied through electric arcs by multiple electrodes to metal solid scrap. Natural gas and oxygen injected through burners provide the required chemical energy after going through combustion. The inputs are manipulated over the entire duration the process. A pool of molten steel forms at the bottom of the electrodes as the batch process goes through the melting stage. A layer of molten slag forms due to oxidation of metals to form metal oxides which floats on top of molten metal. For adjusting the slag chemistry, various fluxes such as lime, dolomite and carbon are added directly. Carbon and oxygen are also lanced to effect the slag composition. In the next subsections, we discuss the dynamic first principles-based model of EAF describe the process phenomenon and how the advisory system framework is adapted for use in the case studies. Finally, we show the economic improvement that is achieve through the use of the advisory system.

6.1 Process Model

EAF operation is characterized by its complexity and the extreme operating nature of the process. This lead to multiple challenges that are associated with modeling the EAF process. Very detailed computational fluid dynamics (CFD) models are available only for a section of the furnace.[60] The CFD models are appropriate for engineering and design study but not suitable for optimal control applications particularly in real-time due to the very long solve times.[61] Models will reasonable simulation times are generally based on simplifying assumptions and typically divide the EAF into multiple zones.[62–64] Detailed physics-based mathematical equations are then employed to describe the heat and mass transfer happening

within and in-between the zones. The dynamic first-principles based EAF model[13] used in this paper considers four equilibrium zones where reactions are limited by mass transfer (see Fig. 3). The four zones are gas, slag-metal, molten-metal and solid scrap zones.

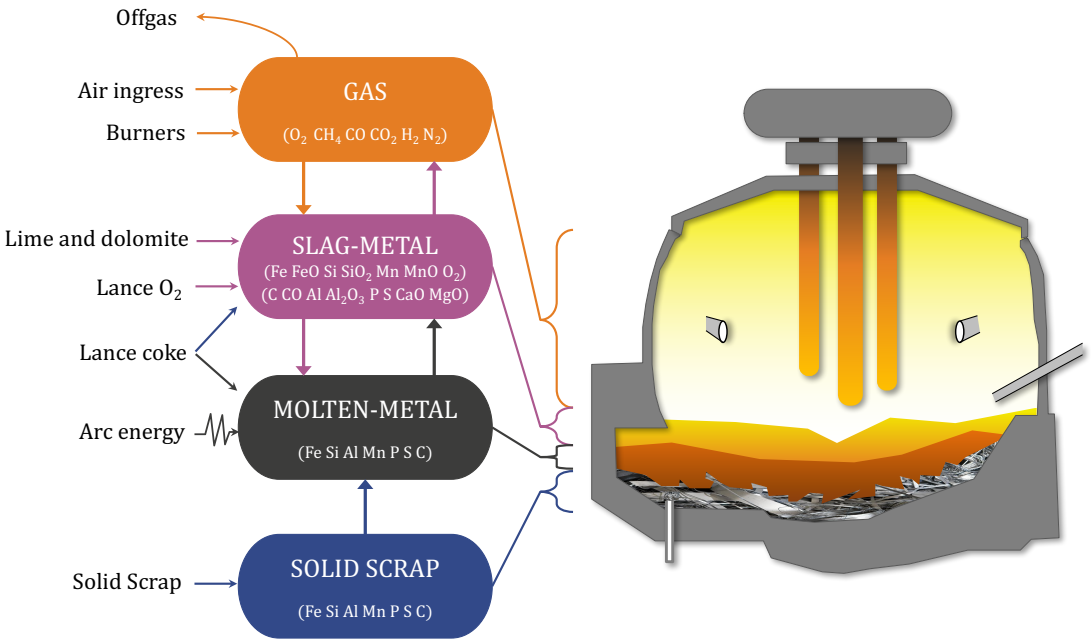

Figure 3: Schematic of the EAF model[12,13] used in the study showing the 4 zones and associated inputs, outputs and material flows.

The *gas zone* represents the gases which fill up the furnace volume above the solid metal scrap. The scrap is first preheated using burners before charging it into the furnace. The electrodes are then lowered and a high voltage setting is selected for the electric arcs. The scrap continues to melt and a liquid-metal pool is formed. The C and O_2 injection then helps in forming the slag layer on top of the molten-metal. The *slag-metal interaction* zone considers the liquid slag materials and the part of the molten-metal layer which remains in contact with the slag. The *molten-metal* zone considers metals in their molten form once scrap begins to melt. It excludes the part of liquid mass already considered in the *slag-metal* zone. The *solid scrap* zone consists of the non-melted scrap mass in solid form. The process phenomenon happening in the each of the four zones are modeled through the mass and energy balances.

A key advantage of the equilibrium-based modeling is that we need to deal with fewer parameters compared to a kinetics-based approach. The multi-zone system[13] assumes chemical equilibrium in *slag-metal* and *gas* zones. The equilibrium is determined by incorporating equations for minimization of Gibbs free energy. The model also considers slag foaming, which is caused by evolution of carbon monoxide, through empirical relationships. The materials are tracked within and between each zone through mass transfer equations and elemental balances. Detailed mathematical relationships take in to account of the heat transfer from arc, chemical reactions, convective and radiative heat transfers. Model parameters were estimated in gPROMS/gEST[65] by employing maximum likelihood function and using real plant data sets. The EAF model was modified in[66] to include models for three JetBoxes which provide oxygen and also a flat geometry was assumed instead of a cone-frustum for scrap melting. The commercial modeling tool gPROMS was use to model the DAE system which consisted of 40 differential and 1050 algebraic states.

The highly nonlinear radiation model was removed in[34] and replaced with a parameter which partitions the arc energy between molten-metal, scrap metal, furnace roof and walls. Parameter were re-estimated to obtain matching profiles with the plant data. Also, since *molten-metal* zone contains negligible amounts of metal oxides, it was assumed that all the metal oxides are contained in the *slag-metal* zone. The state variables associated with metal oxides in the *molten-metal* zone were removed. The modified model contained 29 differential states. The gPROMS model[34] was translated to a Python-based CasADi[67] framework for carrying out optimization using the simultaneous approach.[12,68] Also, a state variable which attained negligible values ($\sim 1.0 \times 10^{-13}$) was removed and replaced with a parameter. In this work we employ the EAF model used in[12,68] to do a detailed analysis using three case studies. The DAE model has 28 differential and 518 algebraic variables. The model further went though a *model contraction* procedure[12,68] described in section 2. In the next

subsection we describe the adaption of the multi-tiered optimization for EAF implementation.

6.2 Multi-tiered Optimization: EAF implementation

We aim to maximize the economics for the EAF batch process by solving a dynamic optimization problem based on the first principles EAF model. The optimization is tackled through three tiers of the multi-tiered optimization strategy before the decision support is provided to the operator. The first tier *Shrinking Horizon Economic Optimization* solves a shrinking horizon dynamic optimization problem with an economic objective and a constraint on the end point solid scrap mass m_{ss}. The second and third tiers come into effect if in-feasibility is detected in tier 1. Second tier attempts to restore feasibility through horizon extension where as the third tier solves a relaxed problem. The three tiers are described in the three subsequent subsections in details in the context of EAF implementation.

6.2.1 Tier 1: Shrinking Horizon Economic Maximization

In the first tier we aim to maximize economics subject to the EAF model (2) in order to find the optimal input trajectories. The objective function for the EAF is to maximize profit at the end of the EAF process, which is given as

$$\begin{aligned}\Phi(t_f) :=& c_0 M_{steel}(t_f) - \Bigg(c_1 \int_{t_i}^{t_f} P dt + c_2 \int_{t_i}^{t_f} F_{CH_4,brnr} dt + c_3 \int_{t_i}^{t_f} F_{C_{lance}} dt \\ &+ c_4 \int_{t_i}^{t_f} F_{C_{charge}} dt + \ c_5 \int_{t_i}^{t_f} (F_{O_2,Jetbox1} + F_{O_2,Jetbox2} + F_{O_2,Jetbox3}) dt \\ &+ c_6 \int_{t_i}^{t_f} F_{CaO} dt + \ c_7 \int_{t_i}^{t_f} F_{Dolomite} dt + \ c_8 \int_{t_i}^{t_f} F_{2ndCharge} dt \Bigg), \end{aligned} \tag{18}$$

where M_{steel} is the state variable for the amount of molten steel and its corresponding selling price is denoted by c_0. M_{steel} produced at t_f is the major source of revenue and profit is calculated by subtracting from it the total cost incurred for putting in all the inputs through

the running time of the batch. The 10 inputs $\mathbf{u}$ are: electric arc power P, flow of burner natural gas $F_{CH_4,brnr}$, lancing Carbon $F_{C_{lance}}$, charged Carbon $F_{C_{charge}}$, flow of Oxygen from three JetBoxes $\{F_{O_2,Jetbox1}, F_{O_2,Jetbox2}, F_{O_2,Jetbox3}\}$, lime F_{CaO}, dolomite $F_{Dolomite}$ and second scrap charge $F_{2ndCharge}$. Although, a similar objective function was used to design a real-time energy management strategy in,[12] it considered only 5 inputs. Different objective functions have been used by researches in past, such as minimize power usage,[69] track set-point similar to MPC,[70] and minimize the final time and FeO with offgas CO content.[71] However, MacRosty and Swartz showed that a dynamic optimization can directly aim to maximize profit by balancing input usage with final molten steel produced.[5,72] We calculate corresponding costs by first summing up the inputs used through the batch from t_i to t_f and then multiplying it with the corresponding cost coefficient $c_i, i = 1, \ldots, 8$.

Besides the constraints on the state and algebraic variables, the input constraints are given as

$$P^{min}(t) \leq P \leq P^{max}(t) \tag{19a}$$

$$F_k^{min}(t) \leq F_k \leq F_k^{max}(t), \tag{19b}$$

where superscripts min and max denotes the lower and upper limits respectively. The input constraint have been implemented to ensure that we get the computed optimal input sequence in the realistic operating range of the furnace. Additional, we have an end-point constraint to ensure that all the scrap left in the end $m_{ss}(t_f)$ is not more than δ_{ss}:

$$m_{ss}(t_f) \leq \delta_{ss}. \tag{20}$$

This end point constraint can potentially lead to in-feasibility which is tackled using the tiers 2 and 3. In this, a message is sent to the EAF operators informing that there is high chance that solid scrap will get left at the end of the batch.

6.2.2 Tier 2

If end point constraint (20) causes in-feasibility then optimization horizon is increased in integral steps to final the optimal solution. Typically this situation occurs in the end so solution time should not be an issue. The key idea for a serial implementation is that we start with 1 time step increase first and then keep adding steps till we are able to solve the optimization problem with objective $\Phi(t_f + N_e\Delta T)$ or reach a maximum number of added steps or an infeasible solve. This involves extending the input constraints for the added time segment as well.

6.2.3 Tier 3

In this tier, the end point constraint (20) is removed and the modified objective function is given as

$$\Phi_{Tier3} := \Phi(t_f + N_{ext}\Delta T) - \epsilon(m_{ss}(t_f) - \delta_{ss})^2, \tag{21}$$

where ϵ is a fixed penalty tuning parameter.

6.3 Implementation

We first use *model contraction* in CasADi to remove 397 algebraic variables from the original DAE model containing 28 differential and 518 algebraic variables. The eliminated algebraic variables becomes the dependent variables defined by explicit functions of only the state and algebraic variables. The advisory system is implemented for a discretized model using the Python front end of CasADi. We use implicit Euler scheme with 7 time steps per minute (the sampling time for EAF system is 1 minute) for discretization. The sequential approach was utilized by MacRosty and Swartz[5] to solve the dynamic optimization problems. However, sequential approach can be time-consuming when there is a need to carry out repeated optimizations using a large-scale DAE model.[73] Simultaneous approach, which is useful for

real-time applications such as advisory systems, is employed here for solving the optimization problems. The infinite dimension optimization problem is thus converted to a sparse large-scale NLP problem. We employ interior-point solver IPOPT[74] (with MA27 as the linear solver) to solve the NLP. We have also used IDAS (part of SUNDIALS suite of solvers[75]) for carrying out forward simulations for generating measurements.

For multi-tiered optimization algorithm, the max_{iter} parameter is set to 100. Since, multi-tiered initialization strategy is very effective we found 100 iterations to be a good criteria to stop the optimization solve and move forward in the algorithm. For tier 2 of optimization and sub-tier algorithm maximum allowed extensions were chosen as 3 and 10 respectively. This is reasonable since it is not desirable to extend a EAF batch by more than 10-13 minutes in usual scenarios. To warm start the predicted MHE and optimization problems in the multi-tiered initialization strategy we used the most recent respective solves to initialize the problems. For initializing the predicted MHE, the last MHE solution is taken after removing the primal and dual value associated with the first control stage. We then add the guesses for the primal variables values for the new control stage using a forward simulation. The predicted shrinking horizon optimization problems are initialized by taking the primal and dual variable values from the last solve and dropping the variables values associated with the first control interval.

Plant-model mismatch was incorporated by decreasing the power factor parameter k_p by 10% in the model employed by multi-tiered optimization and MHE. This reduction in the k_p value leads to a significant reduction in the electrical energy transfered to the scrap by the arc power. More details on the exact model equations effected by this mismatch can be found in.[66] To mitigate the effects of the mismatch and other disturbances (described in the next section), we introduced two disturbance states, each for the state variables representing the solid scrap mass and the moles of managenese in the *slag-metal* zone. This is implemented

by using $\mathbf{B}_d$ as $e_i \in \mathbb{R}^{n\times 1}$ where i denotes state variable number in the state variable vector of size n for the solid scrap mass and the moles of managenese and e_i represents the ith unit vector. We chose $\mathbf{Q}_d$ as 0.015 and 0.2 for disturbance states of the solid scrap mass and the moles of managenese respectively.

6.4 Case studies

In this section we present three case studies to demonstrate potential benefits of a real-time implementation of the proposed advisory system. The studies correspond to scenarios where we study the effect of advisory system calls on economics and in-effect how the system deals with disturbances and in-feasibilities. We also analyze the multi-rate MHEs ability to track the true states. Finally, we discuss the computational effort required to solve the MHE and shrinking horizon dynamic optimization problems.

We implement the advisory system using the first principles EAF model comprising of 28 state variables and 512 algebraic variables. The model contraction using CasADi brought down the algebraic variables to 121 while giving 391 dependent variables. The EAF operation considered here is a 60 minutes batch process with 10 manipulated inputs. The process starts with the 1st scrap charged at time $t = 0$ minute and second scrap charged at 25th minute. Although, the scarp charging times are considered to be fixed for the case studies, the charging times itself can also be optimized.

The measurements are available with different sampling rates described in Table 1. There

Table 1: Multi-rate measurement structure for the EAF process.

Time (min)	0 . . . 42	43	44 . . . 46	47	48 . . . 60
Number of measured variables	6	**13**	6	**8**	6

are 6 fast measurements, comprising of the off-gas compositions and the furnace roof and wall temperatures, that are available every 1 minute. This limited availability of the continuous information about the process makes the estimation process extremely challenging. Slag chemistry information in the form of 5 measurements is available at the 43rd minute of the batch duration. Molten metal carbon content and temperature are measured at 43rd and 47th minute. Since, these slow measurements give valuable information about the process, a multi-rate MHE design is essential for optimal performance of the state estimator. The measurements with their corresponding sampling time points and variances are given in Table 2. The MHE performance is demonstrated when the initial conditions are unknown and there is plant-model mismatch as well as measurement noises.

Table 2: Measurement availability for the EAF process.

Measurement	Sampling Time	Variance
Off-gas compositions (CO, CO_2, O_2, H_2) (dimensionless)	Every 1 min	0.01
T_{roof}, T_{wall} (K)	Every 1 min	3
Slag compositions (FeO, Al_2O_3, SiO_2, MgO, CaO) (dimensionless)	43rd min	0.1
Molten-metal temperature (K)	43rd & 47th min	5
Molten-metal carbon content (dimensionless)	43rd & 47th min	0.01

Carrying out an observability analysis is important before proceeding with implementing MHE. This is particularly useful for systems where limited measurements are available. We first linearized the DAE model at each sampling time and then checked the observability metric value[31] at the corresponding time points. We first construct the observability matrix for the system and carry out a Singular Value Decomposition (SVD) to give us the singular values. The linearized model is then transformed to observability canonical form which yields the transformed z-space corresponding to the original x-space of the linearized model. The observability metric value for each state is then computed by projecting it on the z-space while the weight assigned to it being the associated singular value. The lowest observability

metric value identified for our system is 7.0×10^{-7} which implies that the system is fully observable. It is to be noted that instead of carrying out the linearization procedure, output for IPOPT solver can be examined directly to determine the system observability.[49] Once the reduced Hessian is extracted at the end of the IPOPT solve, it provides the information on the solution satisfying the second order conditions.[76]

In the next set of subsection we describe the three case studies conducted to test the real-time advisory system. First case study demonstrates the economic benefit of carrying out an increased number of re-optimizations through calling the advisory system more often. The second case study shows the effectiveness of the advisory system when plant is effected by a major disturbance. The third case study explores the use of three tiers and the sub-tier.

6.4.1 Case study 1: Economic benefit due to operator calling the real-time advisory system

In this case we compare two scenarios: when the optimization in the advisory system is activated twice and when it is activated 7 times by the EAF operators. Since, there is a flexibility in the advisory system for operators to give it a call when decision support is needed, this case illustrates the scenarios where the number of calls can differ. In scenario 1, optimization is first carried out at 0th minute and the calculated moves are only implemented until 30th minute. At 30th minute, a re-optimization is done and the new calculated inputs are subsequently implemented till the end of the batch. In scenario 2, the optimizations are carried out at 0th, 10th, 30th, 35th, 40th, 45th and 50th minute i.e. 6 re-optimizations.

For both the scenarios, MHE is running in background to provide the state estimates when needed. Since, this case study is similar to a real case where large disturbances are not expected, the state estimation model only has 1 disturbance state for amount of manganese in slag. The end point constraint is implemented for solid scrap mass (m_{ss}) at 60th minute.

While there is no lower bound for $m_{ss}(t = 60\text{min})$, we want the final scrap left to be less than 8 kg. Also, the input bounds are (compared to the nominal value): $\pm 10\%$ for $[F_{CaO},\ F_{C_{lance}},\ F_{2ndCharge},\ F_{Dolomite}]$, $\pm 20\%$ for $[F_{C_{charge}}, F_{CH_4,brnr}, P, F_{O_2,Jetbox1}, F_{O_2,Jetbox2},\ F_{O_2,Jetbox3}]$ and $\pm 30\%$ for P.

Both the scenarios are compared in Table 3. We can observe that for scenario 2 we get

Table 3: Results for case study 1.

	Scenario 1	Scenario 2
Times at which optimizations were carried out (min)	$0, 30$	$0, 10, 30, 35, 40, 45, 50$
Number of optimizations	2	7
Economic objective function value ($)	9482.1	9535.2
Actual scrap left (kg)	7.62	7.84
Estimated scrap left (kg)	7.51	7.15
Extension in batch length (min)	1	0

0.56% increase in profit. This increase percent will potentially lead to a major increase in annual steelmaking profit. Also, for scenario 2 the batch length was extended for 1 minute because at 60th minute solid scrap left was more than 8 kg. This demonstrate that frequent re-optimizations help in meeting end point constraints and also extension of batch can be avoided. The re-optimizations indirectly use the available state estimate information which lead to the optimal control moves calculated based on a current plant state. The extension in batch length can sometimes lead to delayed start of subsequent batches and also delays in downstream processes. Although we have not taken these costs in account within this paper, these substantial increases can lead to a significant lowering of the final profit. The inputs given in fig. 4 shows how the calculated inputs differ for the two scenarios.

6.4.2 Case study 2: Major disturbance happens during the batch process.

This case study emulates a situation where plant is effected by major disturbances while the batch process in evolving in time. To create such a situation we chose a parameter that have

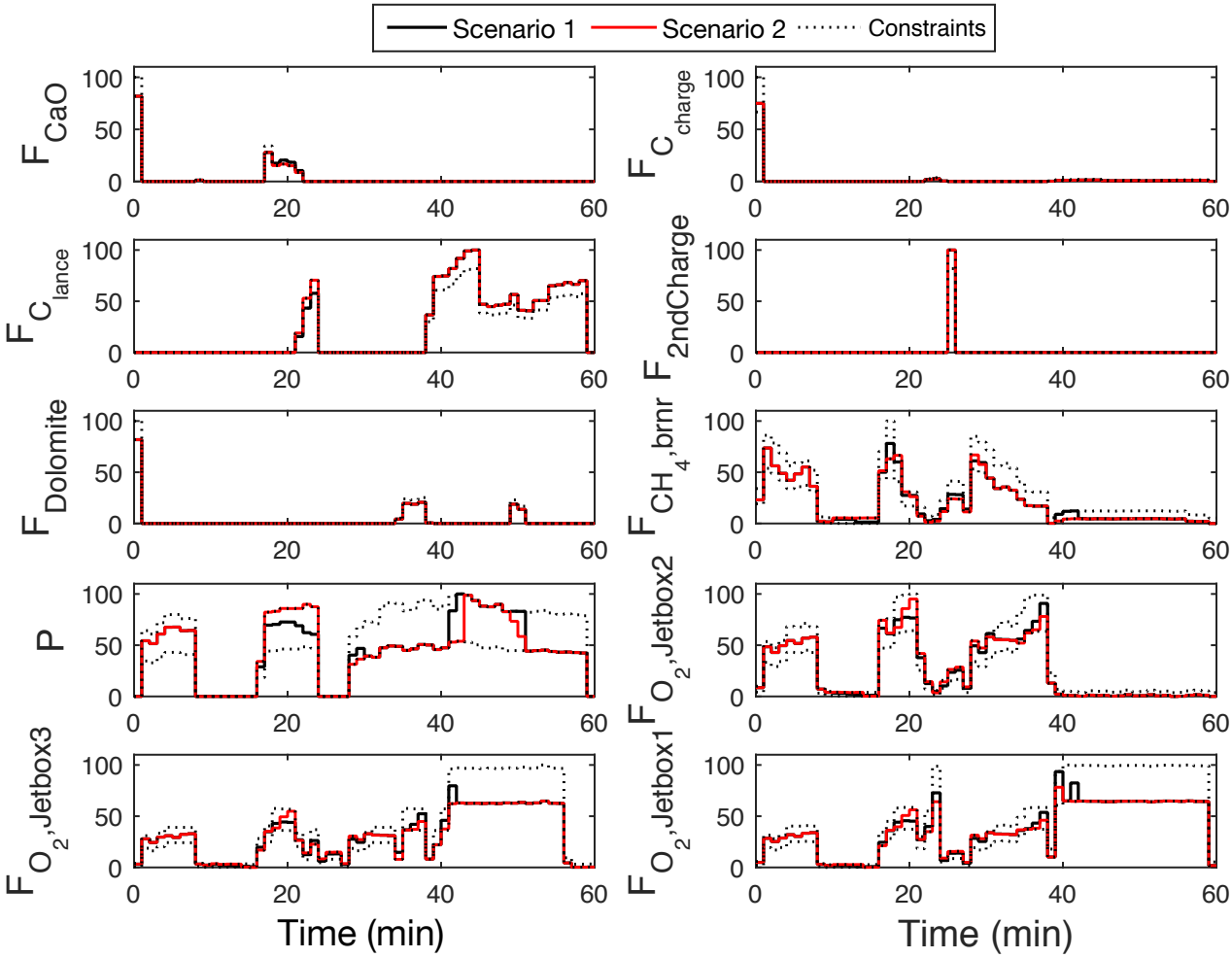

Figure 4: Input profiles for case study 1.

a major impact on the implementation of the solution on the plant. The parameter k_{dm}, that controls the rate of melting,[13] was increased in the plant by 5% at 32nd minute. This step increase leads to a significant decrease in the rate of scrap melting from 32nd minute onwards during the batch. An immediate impact is that more solid scrap will be left at 60th minute and a batch extension will be required. These aspects as well as the economic benefit is discussed here for the 2 scenarios described in case study 1, viz. 1 and 6 re-optimizations respectively.

The results for both the scenarios are compared are given in Table 4. We observe that scenario 2 continues to give 0.56% better economic objective function value. This is because more re-optimizations provide closer to optimal closed-loop input profile. In limit of re-optimizations carried out at every 1 minute, the implementation naturally extends to Nonlinear Model Predictive Control (NMPC). Since, the rate of melting is decreased at 32nd minute due to the disturbance, if we keep implementing input moves calculated at 30th min as in scenario 1, the batch needs to extended by 3 minutes in order to make sure the scrap left is less than 8 kg. However, for scenario 2 no such extension is required since MHE is able to deduce that a major disturbance has happened and this crucial information is supplied to the re-optimizations carried out at $[35, 40, 45, 50]$th minute. This case demonstrates that re-optimizations are important in two ways: increasing economic profit and avoiding batch extension while tackling disturbances.

Table 4: Results for Case study 2.

	Scenario 1	Scenario 2
Times at which optimizations were carried out (min)	$0, 30$	$0, 10, 30, 35, 40, 45, 50$
Number of optimizations	2	7
Economic objective function value ($)	9510.58	9563.64
Actual scrap left (kg)	7.28	7.07
Estimated scrap left (kg)	7.40	7.03
Extension in batch length (min)	3	0

6.4.3 Case study 3: Feasibility restoration through multi-tiered optimization strategy.

In this case study, we carry out single re-optimization at 45th minute of the batch process. Disturbance is introduced through a step increase of k_{dm} by 12.3% at 38th minute during the batch operation. This change significantly reduces the melting rate of solid scrap in the plant. We also lowed the upper and lower bounds of the electric power input to 60% and 80% of the nominal values respectively. This combined change led to tier 1 solving upto the maximum iterations limit of 100 without finding an optimal solution. Then, tier 2 came into action and three extended dynamic optimization problems were solved serially for each of the integral extensions upto 3 time steps. We kept the upper limit of extensions in tier 2 as 3 time steps. Since all the three problems reached maximum iterations, the advisory system moved to tier 3 to solve a relaxed problem with the objective function as

$$\phi = J_{tier2} + 100(m_{solid\,scrap}(t_f) - 0.008)^2. \tag{22}$$

Here, the tuning penalty parameter is 0.01 and the end point upper bound is 8 kg. Since a relaxed extended optimization problem is solved at 45th minute, the sub-tier is activated at 63rd minute as expected because solid scrap left at the time is more than 8 kg. The maximum allowed time extension in sub-tier is kept at 10 minutes. After the 10 step extensions due to the sub-tier, the final scrap mass left is 18.2 kg. Thus, total batch extension is 13 minutes. The results for this case study is presented in table 5.

6.5 Moving Horizon Estimation Results

In this section we discuss the performance of multi-rate MHE. For all the three case studies, Gaussian noise of 1% relative variance was added to the true initial conditions of the state variables. Measurements were perturbed by adding Gaussian noise of relative variances according to the values given in table 2. The covariance matrix Q for the model uncertainty

Table 5: Case study 3 results.

Profit	9353.6
Actual scrap left (kg)	18.2
Estimated scrap left (kg)	12.5
Total extension	13 min
Disturbance introduced at	38th min
Disturbance effect	12% increase in k_{dm}
Reoptimization point	45th minute

was chosen based on multiple trial simulations. We also implemented upper and lower bounds for the state, the algebraic and the model noise variables. A moving horizon length of 6 was selected for all the case studies. Fig.5 represents a subset of the true states and the estimated states for case study 3. We can notice that MHE is able to track the true states in-spite of the plant-model mismatch, initial state discrepancy and measurement noises. Furthermore, the total time horizon of state profiles is 73 minutes which includes the 13 minute extension due to the tiered optimization algorithm.

6.6 Computational Results

We have employed an Intel Core i7-3770 processor (with 4 CPU cores) running Windows 7 at 3.4 GHz for all numerical computations. The CPU times required for solving the multi-rate MHE problems, with (fMHE) and without (nMHE) the use of the proposed multi-tiered initialization scheme, for case study 3 are shown in fig. 6. The average solution times for fMHE and nMHE are 0.49 and 0.71 seconds respectively. So, there is 31% decrease in MHE solve time due to the use of the multi-tiered warm-start strategy. The solution times for the multi-tiered dynamic optimization problems of case study 3 is given in table 6. We can clearly see the advantage of using the multi-tiered initialization scheme. The solution times for the advisory system use in case study 1 and 2 were on average below 5 seconds. Furthermore, the solution time for the predicted problems for all the three case studies were

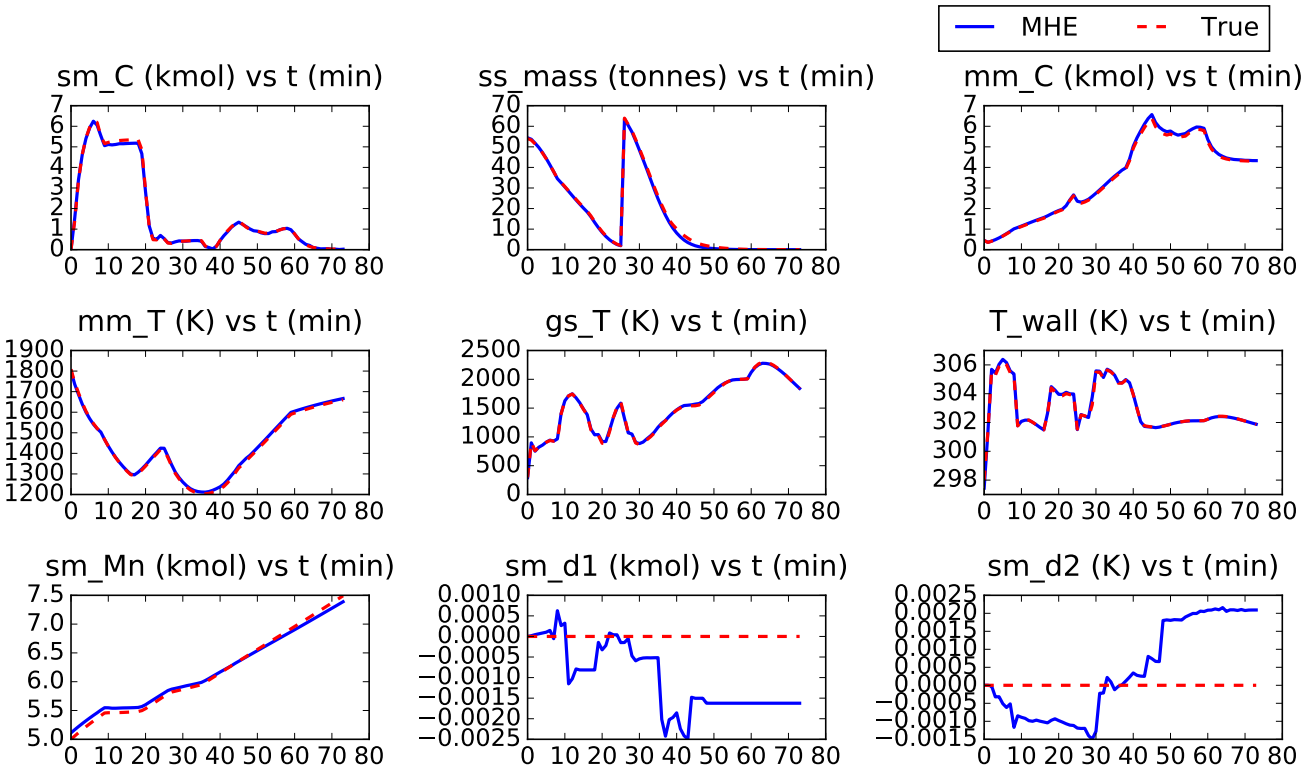

Figure 5: State estimate profiles varying with respect to time (in minutes) for case study 3. sm_C (carbon content in slag-metal zone), ss_mass (mass of solid scrap), mm_C (carbon content of molten-metal zone), mm_T (temperature of molten-metal zone), gs_T (temperature of gas zone), T_wall (temperature of furnace wall), sm_Mn (manganese content in slag-metal zone), sm_d1 (Disturbance state for sm_Mn) and sm_d2 (Disturbance state for ss_mass).

well below 1 minute.

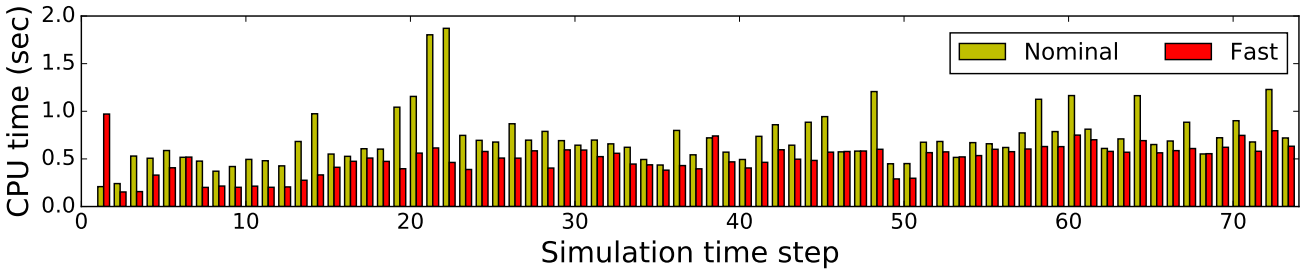

Figure 6: Solution times for multi-rate MHE problems (horizon length is 6 minutes) for case study 3. 'Fast' represents solution times when the multi-tiered initialization scheme is used. 'Nominal' denotes solutions times when the proposed initialization scheme is not utilized.

Table 6: Solution times for the multi-tiered dynamic optimization problems of case study 3.

Time points where multi-tiered optimizer is called	0th minute	45th minute				
Activated tier number	1	1	2			3
Solution time (CPU seconds) without the use of multi-tiered initialization	25.7	6.8	6.6	7.3	7.1	2.1
Solution time (CPU seconds) with the use of multi-tiered initialization	2.1	1.1	0.9	1.2	1.1	0.6

7 Conclusion

In this work we introduced a real-time dynamic optimization-based advisory system for batch and semi-batch operations. The advisory combines the multi-tiered dynamic optimization with MHE to provide online decision support to the plant operators. The three tiers of the optimizer effectively handle the endpoint constraints by restoring feasibility. At any point during the batch operation, the optimizer can be initiated by the operators. The initial conditions for the optimization problem is obtained from multi-rate MHE which runs in background. We also proposed a multi-tiered initialization scheme for obtaining fast solution from the advisory system.

Our case studies on a Electric Arc Steelmaking semi-batch process showed the economic benefit of using the real-time advisory system. Optimal inputs provided by the advisory system made sure that the challenging process is operated economically while respecting constraints. The strong convergence of MHE directly influenced the optimizer results and provided a firm basis for activating the sub-tier algorithm. The advisory system implementation showed that it can fill the gap in current industrial advanced control applications. This will lead to a significant step forward for industries where multiple control challenges are faced due to complicated batch and semi-batch processes.

For the next steps, we aim to formulate the optimization problem to find the optimal extension of batch length when an infeasibility occurs. We would also like to explore the possibility of contraction in time duration of operation. Furthermore, incorporating the lower-level logic-based control in the dynamic optimization formulation would be a useful avenue to study. Also, in-plant evaluation of the real-time advisory system constitutes an important step for industrial usage.

Acknowledgement

We would like to acknowledge support by the McMaster Steel Research Center (SRC) and the McMaster Advanced Control Consortium (MACC) for this work.

References

(1) Dorka, P.; Fischer, C.; Budman, H.; Scharer, J. M. Metabolic flux-based modeling of mAb production during batch and fed-batch operations. *Bioprocess and biosystems engineering* **2009**, *32*, 183–196.

(2) Cao, Y.; Acevedo, D.; Nagy, Z. K.; Laird, C. D. Real-time feasible multi-objective

optimization based nonlinear model predictive control of particle size and shape in a batch crystallization process. *Control Engineering Practice* **2017**, *69*, 1–8.

(3) Christofides, P. D.; El-Farra, N.; Li, M.; Mhaskar, P. Model-based control of particulate processes. *Chemical Engineering Science* **2008**, *63*, 1156–1172.

(4) Garg, A.; Mhaskar, P. Subspace Identification-Based Modeling and Control of Batch Particulate Processes. *Industrial & Engineering Chemistry Research* **2017**, *56*, 7491–7502.

(5) MacRosty, R. D.; Swartz, C. L. E. Dynamic optimization of electric arc furnace operation. *AIChE journal* **2007**, *53*, 640–653.

(6) Jang, H.; Lee, J. H.; Biegler, L. T. A robust NMPC scheme for semi-batch polymerization reactors. *IFAC-PapersOnLine* **2016**, *49*, 37–42.

(7) Zachariah, A.; Wang, L.; Yang, S.; Prasad, V.; de Klerk, A. Suppression of coke formation during bitumen pyrolysis. *Energy & Fuels* **2013**, *27*, 3061–3070.

(8) Bonvin, D. Control and optimization of batch processes. *IEEE control systems* **2006**, *26*, 34–45.

(9) Nagy, Z. K.; Braatz, R. D. Robust nonlinear model predictive control of batch processes. *AIChE Journal* **2003**, *49*, 1776–1786.

(10) Deshpande, S. A.; Bonvin, D.; Chachuat, B. Directional input adaptation in parametric optimal control problems. *SIAM Journal on Control and Optimization* **2012**, *50*, 1995–2024.

(11) Flores-Cerrillo, J.; MacGregor, J. F. Control of batch product quality by trajectory manipulation using latent variable models. *Journal of Process Control* **2004**, *14*, 539–553.

(12) Shyamal, S.; Swartz, C. L. E. Real-time energy management for electric arc furnace operation. *Journal of Process Control* **2018**, *accepted for publication*.

(13) MacRosty, R. D.; Swartz, C. L. E. Dynamic modeling of an industrial electric arc furnace. *Industrial & engineering chemistry research* **2005**, *44*, 8067–8083.

(14) Rashid, M. M.; Mhaskar, P.; Swartz, C. L. E. Handling multi-rate and missing data in variable duration economic model predictive control of batch processes. *AIChE Journal* **2017**, *63*, 2705–2718.

(15) Russell, S.; Robertson, D.; Lee, J.; Ogunnaike, B. Control of product quality for batch nylon 6, 6 autoclaves. *Chemical engineering science* **1998**, *53*, 3685–3702.

(16) Flores-Cerrillo, J.; MacGregor, J. F. Latent variable MPC for trajectory tracking in batch processes. *Journal of process control* **2005**, *15*, 651–663.

(17) Dogan, N.; Brooks, G. A.; Rhamdhani, M. A. Comprehensive model of oxygen steelmaking part 1: model development and validation. *ISIJ international* **2011**, *51*, 1086–1092.

(18) Biegler, L. T. *Nonlinear programming: concepts, algorithms, and applications to chemical processes*; SIAM, 2010.

(19) Golshan, M.; MacGregor, J. F.; Bruwer, M.-J.; Mhaskar, P. Latent Variable Model Predictive Control (LV-MPC) for trajectory tracking in batch processes. *Journal of Process Control* **2010**, *20*, 538–550.

(20) Chang, L.; Liu, X.; Henson, M. A. Nonlinear model predictive control of fed-batch fermentations using dynamic flux balance models. *Journal of Process Control* **2016**, *42*, 137–149.

(21) Aumi, S.; Mhaskar, P. Robust model predictive control and fault handling of batch processes. *AIChE Journal* **2011**, *57*, 1796–1808.

(22) Aydin, E.; Bonvin, D.; Sundmacher, K. NMPC using Pontryagin?s Minimum Principle-Application to a two-phase semi-batch hydroformylation reactor under uncertainty. *Computers & Chemical Engineering* **2018**, *108*, 47–56.

(23) Nagy, Z. K.; Mahn, B.; Franke, R.; Allgöwer, F. Real-time implementation of nonlinear model predictive control of batch processes in an industrial framework. In *Assessment and Future Directions of Nonlinear Model Predictive Control*; Springer, 2007; pp 465–472.

(24) Flores-Cerrillo, J.; MacGregor, J. F. Control of particle size distributions in emulsion semibatch polymerization using mid-course correction policies. *Industrial & Engineering Chemistry Research* **2002**, *41*, 1805–1814.

(25) Ali, J. M.; Hoang, N. H.; Hussain, M. A.; Dochain, D. Review and classification of recent observers applied in chemical process systems. *Computers & Chemical Engineering* **2015**, *76*, 27–41.

(26) Prasad, V.; Schley, M.; Russo, L. P.; Bequette, B. W. Product property and production rate control of styrene polymerization. *Journal of Process Control* **2002**, *12*, 353–372.

(27) Ghobara, E. M. Y. Modeling, Optimization and Estimation in Electric Arc Furnace (EAF) Operation. Ph.D. thesis, 2013.

(28) Qu, C. C.; Hahn, J. Process monitoring and parameter estimation via unscented Kalman filtering. *Journal of Loss Prevention in the Process Industries* **2009**, *22*, 703–709.

(29) Allgöwer, F.; Badgwell, T. A.; Qin, J. S.; Rawlings, J. B.; Wright, S. J. Nonlinear predictive control and moving horizon estimation?an introductory overview. In *Advances in control*; Springer, 1999; pp 391–449.

(30) Rao, C. V.; Rawlings, J. B.; Mayne, D. Q. Constrained state estimation for nonlinear discrete-time systems: Stability and moving horizon approximations. *IEEE Transactions on Automatic Control* **2003**, *48*, 246–258.

(31) Ji, L.; Rawlings, J. B. Application of MHE to large-scale nonlinear processes with delayed lab measurements. *Computers & Chemical Engineering* **2015**, *80*, 63–72.

(32) López-Negrete, R.; Biegler, L. T. A moving horizon estimator for processes with multi-rate measurements: A nonlinear programming sensitivity approach. *Journal of Process Control* **2012**, *22*, 677–688.

(33) Mesbah, A.; Huesman, A. E.; Kramer, H. J.; Van den Hof, P. M. A comparison of nonlinear observers for output feedback model-based control of seeded batch crystallization processes. *Journal of Process Control* **2011**, *21*, 652–666.

(34) Shyamal, S.; Swartz, C. L. E. A Multi-rate Moving Horizon Estimation Framework for Electric Arc Furnace Operation. *IFAC-PapersOnLine* **2016**, *49*, 1175–1180.

(35) Steel Statistical Yearbook of World Steel Association, http://www.worldsteel.org/steel-by-topic/statistics/steel-statistical-yearbook-.html. 2016.

(36) Fruehan, R. J. *The Making, Shaping, and Treating of Steel: Ironmaking volume*; AISE Steel Foundation, 1999; Vol. 2.

(37) Billings, S.; Boland, F.; Nicholson, H. Electric arc furnace modelling and control. *Automatica* **1979**, *15*, 137–148.

(38) Swartz, C. L. E. An algorithm for hierarchical supervisory control. *Computers & chemical engineering* **1995**, *19*, 1173–1180.

(39) Chong, Z.; Swartz, C. L. E. Optimal operation of process plants under partial shutdown conditions. *AIChE Journal* **2013**, *59*, 4151–4168.

(40) Chong, Z.; Swartz, C. L. E. Optimal response under partial plant shutdown with discontinuous dynamic models. *Computers & Chemical Engineering* **2016**, *86*, 120–135.

(41) Chong, Z.; Swartz, C. L. E. Discontinuous Modeling Formulations for the Optimal Control of Partial Shutdowns. *IFAC Proceedings Volumes* **2011**, *44*, 3659–3664.

(42) Chong, Z.; Swartz, C. L. E. Model-based control of multi-unit systems under partial shutdown conditions. American Control Conference, 2009. ACC'09. 2009; pp 160–165.

(43) Biegler, L. T. New directions for nonlinear process optimization. *Current Opinion in Chemical Engineering* **2018**, *21*, 32–40.

(44) Magnusson, F. Numerical and symbolic methods for dynamic optimization. **2016**,

(45) Åkesson, J.; Årzén, K.-E.; Gäfvert, M.; Bergdahl, T.; Tummescheit, H. Modeling and optimization with Optimica and JModelica. org?Languages and tools for solving large-scale dynamic optimization problems. *Computers & Chemical Engineering* **2010**, *34*, 1737–1749.

(46) Robertson, D. G.; Lee, J. H. On the use of constraints in least squares estimation and control. *Automatica* **2002**, *38*, 1113–1123.

(47) Alessandri, A.; Baglietto, M.; Battistelli, G. Moving-horizon state estimation for nonlinear discrete-time systems: New stability results and approximation schemes. *Automatica* **2008**, *44*, 1753–1765.

(48) Küpper, A.; Diehl, M.; Schlöder, J. P.; Bock, H. G.; Engell, S. Efficient moving horizon state and parameter estimation for SMB processes. *Journal of Process Control* **2009**, *19*, 785–802.

(49) Zavala, V. M.; Biegler, L. T. Optimization-based strategies for the operation of low-density polyethylene tubular reactors: Moving horizon estimation. *Computers & Chemical Engineering* **2009**, *33*, 379–390.

(50) Magnusson, F.; Åkesson, J. Dynamic optimization in JModelica.org. *Processes* **2015**, *3*, 471–496.

(51) Kramer, S.; Gesthuisen, R.; Engell, S. Fixed structure multirate state estimation. American Control Conference (ACC). 2005; pp 4613–4618.

(52) Krämer, S.; Gesthuisen, R. Multirate state estimation using moving horizon estimation. *16th IFAC World Congress, IFAC Proceedings Volumes* **2005**, *38*, 1–6.

(53) Simon, D. *Optimal state estimation: Kalman, H infinity, and nonlinear approaches*; John Wiley & Sons, 2006.

(54) Lopez-Negrete, R.; Patwardhan, S. C.; Biegler, L. T. Constrained particle filter approach to approximate the arrival cost in moving horizon estimation. *Journal of Process Control* **2011**, *21*, 909–919.

(55) Qu, C. C.; Hahn, J. Computation of arrival cost for moving horizon estimation via unscented Kalman filtering. *Journal of Process Control* **2009**, *19*, 358–363.

(56) Diehl, M.; Ferreau, H. J.; Haverbeke, N. Efficient numerical methods for nonlinear MPC and moving horizon estimation. In *Nonlinear Model Predictive Control*; Springer, 2009; pp 391–417.

(57) Huang, R.; Zavala, V. M.; Biegler, L. T. Advanced step nonlinear model predictive control for air separation units. *Journal of Process Control* **2009**, *19*, 678–685.

(58) Kühl, P.; Diehl, M.; Kraus, T.; Schlöder, J. P.; Bock, H. G. A real-time algorithm for moving horizon state and parameter estimation. *Computers & Chemical Engineering* **2011**, *35*, 71–83.

(59) Biegler, L.; Yang, X.; Fischer, G. Advances in sensitivity-based nonlinear model predictive control and dynamic real-time optimization. *Journal of Process Control* **2015**, *30*, 104–116.

(60) Li, Y.; Fruehan, R. J. Computational fluid-dynamics simulation of postcombustion in the electric-arc furnace. *Metallurgical and Materials Transactions B* **2003**, *34*, 333–343.

(61) Irons, G. A. Developments in electric arc furnace steelmaking. AISTECH-Conference Proceedings-. 2005; p 3.

(62) Matson, S.; Ramirez, W. F. Optimal operation of an electric arc furnace. 57 th Electric Furnace Conference. 1999; pp 719–730.

(63) Bekker, J. G.; Craig, I. K.; Pistorius, P. C. Modeling and simulation of an electric arc furnace process. *ISIJ international* **1999**, *39*, 23–32.

(64) Fathi, A.; Saboohi, Y.; Škrjanc, I.; Logar, V. Comprehensive Electric Arc Furnace Model for Simulation Purposes and Model-Based Control. *Steel Research International* **2017**, *88*, 1600083.

(65) Process Systems Enterprise Ltd., gPROMS, www.psenterprise.com/gproms, 1997-2015. 2015.

(66) Ghobara, Y. Modeling, Optimization and Estimation in Electric Arc Furnace (EAF) Operation. M.Sc. thesis, McMaster University, 2013.

(67) Andersson, J. A General-Purpose Software Framework for Dynamic Optimization. PhD thesis, Arenberg Doctoral School, KU Leuven, Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium, 2013.

(68) Shyamal, S.; Swartz, C. L. E. Optimization-based Online Decision Support Tool for Electric Arc Furnace Operation. *IFAC-PapersOnLine* **2017**, *50*, 10784–10789.

(69) Woodside, C.; Pagurek, B.; Pauksens, J.; Ogale, A. Singular arcs occurring in optimal electric steel refining. *IEEE transactions on automatic control* **1970**, *15*, 549–556.

(70) Oosthuizen, D.; Craig, I.; Pistorius, P. Model predictive control of an electric arc furnace off-gas procedure combined with temperature control. Africon, 1999 IEEE. 1999; pp 415–420.

(71) Matson, S.; Ramirez, W. F. Optimal operation of an electric arc furnace. 57 th Electric Furnace Conference. 1999; pp 719–730.

(72) MacRosty, R. D.; Swartz, C. L. E. Nonlinear predictive control of an electric arc furnace. *IFAC Proceedings Volumes* **2007**, *40*, 285–290.

(73) Biegler, L. T. An overview of simultaneous strategies for dynamic optimization. *Chemical Engineering and Processing: Process Intensification* **2007**, *46*, 1043–1053.

(74) Wächter, A.; Biegler, L. T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical Programming* **2006**, *106*, 25–57.

(75) Hindmarsh, A. C.; Brown, P. N.; Grant, K. E.; Lee, S. L.; Serban, R.; Shumaker, D. E.; Woodward, C. S. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software (TOMS)* **2005**, *31*, 363–396.

(76) Nocedal, J.; Wright, S. Numerical optimization, series in operations research and financial engineering. *Springer, New York, USA, 2006* **2006**,

List of Figures