2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Adaptive steady-state target optimization using iterative modified gradient-based methods in linear non-square MPC

L.R.E. Shead*, K.R. Muske**, J.A Rossiter*

Abstract— **An important feature of linear model predictive control (MPC) is the ability to provide offset-free control through integral action. Linear MPC can utilize a steady-state target optimizer (SSTO) in conjunction with a dynamic optimization in order to manage systems that are non-square, have integrating modes, or encounter infeasible setpoints. Integral action does not ensure that the feasible steady-state target is closest to the true optimum when the desired setpoint is infeasible. This paper describes the modifications necessary to linear state-space MPC algorithms in order to address this problem (assuming systems with no integrating modes). The solution employs features of the *Integrated System Optimization and Parameter Estimation (ISOPE)* algorithm: the SSTO cost is modified by a term that results in matching of the true plant and model conditions necessary for optimality. This work combines well with prior work [19], [20] which has determined the situations where the modification is actually necessary.**

I. Introduction

Model predictive control (MPC) refers to a control technique that makes use of the predicted evolution of a plant in determining an open-loop optimal set of future control trajectories. Usually the control law is computed using an optimization program which is solved on-line at every time-step. MPC has been employed widely in the process control industry, its popularity being largely attributed to its ability to consider both current and future constraints in the problem formulation and handle multivariable systems systematically.

Linear MPC is by now well understood [9], [17], [3], but there remain some issues with the well-established algorithms. Systems that are non-square, have integrating modes, or encounter infeasible setpoints require special management to obtain the true optimal behavior, and this has generally been achieved by a steady-state target optimizer (SSTO) that exists locally just above the dynamic MPC optimization in the system hierarchy [12]. Integral action is a key ingredient of MPC, which requires disturbance modeling for systems with a SSTO. Latter methods have preferred augmenting an estimator (required anyway for state-space MPC with unmeasured states) with disturbance states [11], [13]. However, even with all of these standard MPC ingredients, the established algorithms fail to find the correct target with infeasible setpoints, and uncertainty in the process model. Prior work [19], [20] has identified that very accurate information of the active process-dependent constraints is usually required for "constrained offset-free control" (although there are instances where this is not necessary). Disturbance modeling techniques can fail to provide constrained offset-free control through integral action with active steady-state constraints because they identify only a linear shift to reconcile plant with model once converged, and do not provide constraint gradient information.

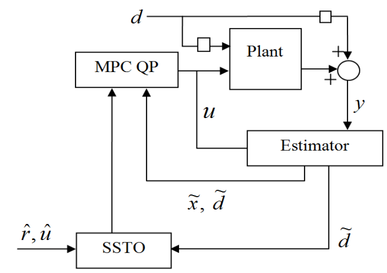

Fig. 1. Architecture diagram.

However, in the field of real-time process optimization (which may not assume MPC) adaptive methods have been developed to compensate for model uncertainty in the determination of optimal setpoints as reviewed in [4]. Two-step approaches utilize parameter estimation to determine the true plant information upon which the subsequent setpoint optimization is based. However, to avoid issues of interaction between the two steps, modifier-adaption algorithms such as the *Integrated System Optimization and Parameter Estimation (ISOPE)* approach [16] have been introduced which alter the setpoint optimization program by adding in true process derivative information. It can be proved that such a modified optimization program leads to the solution of the hypothetical true plant setpoint optimization [22]. A key component of modifier-adaption techniques is the algorithm used to determine true process model derivative information. A natural choice would be systems identification methods such as recursive (generalized/extended) least squares for estimating the gradients, requiring some sort of persistent excitation. Also applicable are simpler algorithms that make use of the past input/output setpoints, applicable because only the steady-state model is relevant. However, these algorithms still require a form of persistent excitation.

The contribution of this paper is to make sufficient adjustments to modern non-square MPC algorithms for ensuring constrained offset-free control, relying on developments made in the field of real-time process optimization. The objective of the work is to select from the various features of

This work is funded by the EPSRC.

*Department of Automatic Systems and Control Engineering, Mappin Street, University of Sheffield, S1 3JD, UK. l.shead@sheffield.ac.uk, j.a.rossiter@sheffield.ac.uk
**Villanova University, Villanova, PA, USA. kenneth.muske@villanova.edu)

the adaptive methods those which are required assuming that they are utilized in conjunction with a modern dynamic MPC algorithm, estimator, disturbance model and SSTO. A novel algorithm is proposed, which builds on the developments of offset-free MPC, making only sufficient adjustments to the standard MPC algorithm to ensure constrained offset-free control with infeasible setpoints. Integration of MPC and process optimization was developed in [1] where an adaptive scheme was implemented using systems identification methods. The method presented in this paper does not attempt to introduce adaptive methods unnecessarily, concentrating only on estimation of the steady-state process model equality constraint gradients needed to avoid constrained offset-free control, and relying otherwise on the integral action provided by the disturbance modeling.

Other prominent work on unreachable setpoints in linear MPC [15] has focused on the dynamic advantages of posing the MPC cost in terms of deviation from the desired setpoint as opposed to deviation from an optimal steady-state target, yet an optimal steady state normally still exists (and is included in [15] as a terminal constraint for guaranteeing stability). The work in this paper is concerned with a different problem: ensuring that the steady-state target set by a SSTO based on an uncertain process model is actually optimal.

II. MPC Background

This section presents the MPC mathematical background necessary to discuss the main issues in this paper.

A. Modeling, feedback, and prediction

Consider the following discrete-time system with an unstructured disturbance model:

$$\begin{array}{ll} x_{k+1} = Ax_k + Bu_k + B_d d_k + w_k & \\ y_k = Cx_k + C_d d_k + v_k, & z_k = Hy_k \end{array} \tag{1}$$

where $u \in \mathbb{R}^m, (x, w) \in \mathbb{R}^n, (y, v) \in \mathbb{R}^l, z \in \mathbb{R}^p$, and d_k is a disturbance vector, $d \in \mathbb{R}^{n_d}$ and (w, v) are process and output noise vectors respectively. H selects linear combinations of measured outputs as controlled variables (CVs), z. Following the guidelines of [13] for the specification of the disturbance model in (1), an estimator is designed through the choice of (B_d, C_d) and noise weighting matrices (R_v, Q_w) based on augmenting (1) with disturbance states $\tilde{d}$ where $n_d = l$:

$$\begin{bmatrix} \tilde{x}_{k+1} \\ \tilde{d}_{k+1} \end{bmatrix} = \begin{bmatrix} A & B_d \\ 0 & I \end{bmatrix} \begin{bmatrix} \tilde{x}_k \\ \tilde{d}_k \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u_k \tag{2}$$

Use of estimates $(\tilde{x}, \tilde{d})$ are assumed henceforth. If (C, A) is detectable, and (B_d, C_d) are chosen such that the augmented system (2) is detectable, then a stable linear estimator exists. System (2) is detectable if:

$$\text{rank} \begin{bmatrix} I - A & -B_d \\ C & C_d \end{bmatrix} = n + n_d \tag{3}$$

Let the 'predicted' control law [18] for sample times k be:

$$(u_k - u_s) = \begin{cases} -K(\tilde{x}_k - x_s) + c_k & k \in [0, n_c - 1] \\ -K(\tilde{x}_k - x_s) & k \geq n_c \end{cases} \tag{4}$$

where c_k are the d.o.f. (or control moves) available for constraint handling and (x_s, u_s) are the expected steady-state input/state required to give offset-free tracking in the steady state. In order to determine (x_s, u_s), a separate SSTO is usually performed, such as that in [12]:

$$J^*(x_s^*, u_s^*) = \min_{x_s, u_s} \|r - Hy_s\|_{Q_s}^2 + \|\hat{u}_s - u_s\|_{R_s}^2 \tag{5}$$

$$\text{s. t.} \begin{bmatrix} (I - A) & -B \\ C & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} = \begin{bmatrix} B_d \tilde{d} \\ y_s - C_d \tilde{d} \end{bmatrix} \tag{6}$$

$$\begin{bmatrix} 0 & A_u \\ A_x & 0 \\ A_y C & 0 \end{bmatrix} \begin{bmatrix} x_s \\ u_s \end{bmatrix} - \begin{bmatrix} b_u \\ b_x \\ b_y - A_y C_d \tilde{d} \end{bmatrix} \leq -\text{col}(\epsilon) \tag{7}$$

where A_u, b_u, A_x, b_x and A_y, b_y represent the inequality constraints on the input, state and output respectively, and $\text{col}(x) = [x^T \ x^T \ldots]^T$. Note that the SSTO assumes that $\tilde{d}$ is constant for each execution.

B. Constraint handling and target optimization

Vectors of predictions $\underset{\rightarrow}{x}$, $\underset{\rightarrow}{y}$, $\underset{\rightarrow}{u}$ corresponding to simulating (1),(4) can then be written in concise form, e.g.:

$$\underset{\rightarrow}{x} = P_x \tilde{x}_k + P_c \underset{\rightarrow}{c}_k + P_s [x_s^T, u_s^T]^T + P_d d_k \tag{8}$$

for suitable P_x, P_s, P_d, P_c (note that more sophisticated steady-state parameterizations are possible, e.g. [21]). Let the constraints be linearly time invariant:

$$\begin{array}{l} \text{diag}(A_x) \underset{\rightarrow}{x} \leq \text{col}(b_x); \text{diag}(A_u) \underset{\rightarrow}{u} \leq \text{col}(b_u); \\ \text{diag}(A_y) \underset{\rightarrow}{y} \leq \text{col}(b_y); \text{diag}(X) = \begin{bmatrix} X & 0 & \dots \\ 0 & X & \dots \\ \vdots & \vdots & \ddots \end{bmatrix} \end{array} \tag{9}$$

Consequently the dynamic feasible region can be defined by substituting the predictions into (9), which together with the steady-state inequality constraints (7) takes the form:

$$\tilde{x} : \exists \underset{\rightarrow}{c} \text{ s. t. } M_x \tilde{x} + M_c \underset{\rightarrow}{c} + M_s [x_s^T, u_s^T]^T + M_d \tilde{d} \leq q \tag{10}$$

This set is an inner approximation to the Maximal Controlled Admissible Set (MCAS) [7][1], which can be thought of as the set corresponding to a finite horizon of constraints, beyond which only redundant constraints are added (as it is predicted that $\|\tilde{x}_k - x_s\|$ is vanishing). A typical dynamic optimization is the following QP:

$$\begin{array}{rl} J_c^*(\underset{\rightarrow}{c}^*) & = \arg\min_{\underset{\rightarrow}{c}} \|\underset{\rightarrow}{c}\|_{W_D}^2 \quad \text{s.t. } (\tilde{x}, \underset{\rightarrow}{c}) \in \text{MCAS} \\ \text{where:} & W_D = \text{diag}(W), \quad W = B^T \Sigma B + R \\ & \Sigma - \Phi^T \Sigma \Phi = C^T H^T Q H C + K^T R K \end{array} \tag{11}$$

Some selections of (x_s, u_s), as determined by the SSTO according to r and/or $\hat{u}$, may result in infeasibility of (11). Such a situation can be avoided by either reference governor approaches, incorporating the MCAS into the SSTO (more restrictive on the setpoint and dynamics), or combining the SSTO and dynamic optimizations [21], [8], [23]. Softening of output constraints may also be appropriate.

Even if the targets of the SSTO are feasible, and are achieved at steady-state, the SSTO may have failed in its task of determining the constrained target closest to an infeasible desired setpoint because of uncertainty in the process model. In the following sections we focus on the issue of modifying

[1] If is ϵ small but > 0 then the selected steady-state (effectively the origin being regulated to) is in the interior of the feasible region [7, Theorem 4.1], which is important for finite determination of the MCAS.

the SSTO such that it is able to determine a (feasible) target which is the true minimum weighted least-squares distance to the steady-state setpoint when nested within the feedback configuration of figure 1.

III. MODIFYING THE SSTO

Techniques for modifying the steady-state target in real-time optimization have been reviewed in [4], identifying the following approaches for overcoming issues with steady-state model uncertainty:

1) Model-adaption methods
2) Modifier-adaption methods
3) Direct-input-adaption methods

Model-adaption methods perform an on-line adaption of the process model by adjusting parameters of the process model that produce the same output as that measured. Once the model has been updated, the optimization is performed with the new model. Interaction between the model update and reoptimization steps could mean that the true optimum is not found when there is uncertainty in the process model, unless the model adaption leads to a matching of the KKT conditions for the model and the plant. Modifier-adaption methods are a more direct way of ensuring that the KKT conditions do match. Modifier-adaption methods were originally introduced in [16] with the standard ISOPE method. This method involves modifying the objective function of the model-based optimization program (OP) such that the modified objective function actually solves the problem relevant to the true plant. Direct-input-adaption methods select CVs which when tracked enforce optimal plant operation, thus providing a "self-optimizing control structure".

In general, a modifier-adaption method shall be tailored to the MPC algorithm in order to provide constrained offset-free control with infeasible setpoints, which involves expressing the SSTO in a suitable form. Assuming no integrating modes or state constraints and a full rank linear steady-state gain matrix (and omitting ϵ for the purposes of analysis), the SSTO can be simplified to the following:

$$J^*(u_s^*) = \min_{u_s} \|Hy_s - r\|_{Q_s}^2 + \|u_s - \hat{u}\|_{R_s}^2 \tag{12}$$

$$\text{s. t. } y_s = f(u_s, \tilde{d}) \text{ (linear steady-state model)} \tag{13}$$

$$= \underbrace{C(I-A)^{-1}B}_{G} u_s + \underbrace{C(I-A)^{-1}B_d + C_d}_{G_d} \tilde{d} \tag{14}$$

$$\begin{aligned} g_u(u_s) &= A_u u_s - b_u \leq 0 \\ g_y(y_s) &= A_y y_s - b_y \leq 0 \end{aligned} \tag{15}$$

Expressing the SSTO in this way emphasizes that the cost need not be considered a function of the process model, which is now explicitly expressed in the equality constraint, (14). Assuming choices have been made such that $\nabla_{u_s}^2(J) > 0$, u_s can be chosen that results in a unique x_s that is used as a feed-forward term in the dynamic MPC algorithm, and is eventually reached by x_k. Uncertainty in the steady-state process model (14) can be therefore represented as uncertainty in G for linear systems.

The KKT conditions of optimality can be examined to determine whether uncertainty in these values results in a sub-optimal target. The SSTO lagrangian and necessary optimality conditions are:

$$\begin{aligned} \mathscr{L}(u_s, y_s, \lambda) = J(u_s, y_s) + \big(y_s - f(u_s, \tilde{d})\big)^T \lambda \\ + g_u(u_s)^T \tau_u + g_y(y_s)^T \tau_y \end{aligned} \tag{16}$$

$$\nabla_{u_s} \mathscr{L} = \nabla_{u_s} J - \nabla_{u_s} f^T \lambda + A_u^T \tau_u = 0 \tag{17}$$

$$\nabla_{y_s} \mathscr{L} = \nabla_{y_s} J + \lambda + A_y^T \tau_y = 0 \tag{18}$$

$$\nabla_{\lambda} \mathscr{L} = y_s - f(u_s, \tilde{d}) = 0, \lambda \geq 0 \tag{19}$$

$$\begin{aligned} \nabla_{\tau_u} \mathscr{L} = g_u \leq 0, \tau_u \geq 0, \operatorname{diag}(\tau_u) g_u = 0 \\ \nabla_{\tau_y} \mathscr{L} = g_y \leq 0, \tau_y \geq 0, \operatorname{diag}(\tau_y) g_y = 0 \end{aligned} \tag{20}$$

Substituting (18) into (17) by eliminating λ we have:

$$\nabla_{u_s} \mathscr{L} = \nabla_{u_s} J + [\nabla_{y_s} J + A_y^T \tau_y].\nabla_{u_s} f + A_u^T \tau_u = 0 \tag{21}$$

A. Situations with inactive constraints

If constraints (13-15) are inactive $\rightarrow$ $(\tau_u, \tau_y, \lambda) = 0$ then (20) can be ignored, and equation (18) shows that if $\lambda = 0, \nabla_{y_s} J = 0$, and then from (21) it can be seen that the model equality constraint gradients $\nabla_{u_s} f_m = G_m$ do not need to match the true plant gradients $\nabla_{u_s} f_p = G_p$ in order to satisfy the KKT optimality conditions. The equality constraint gradients are as follows for the particular SSTO that has been defined in (12-15):

$$\nabla_{y_s} J = 2(Hy_s - r)^T Q_s H \tag{22}$$

$$\nabla_{u_s} J = 2(u_s - \hat{u})^T R_s \tag{23}$$

These gradients are zero when the setpoints are feasible and compatible through the model equality constraint. The convergent estimator ensures that the model equality constraint is satisfied for both plant and model with the same values of u_s and y_s, even though d and $\tilde{d}$ may differ. This analysis reinforces the conclusion that integral action works for situations with inactive constraints. Papers such as [22] specify that it is necessary for the true plant and model equality constraint gradients to match, but this is not true in this case.

However, with non-square thin systems (when R_s is usually set to 0) or systems where $(\hat{u}, r)$ have not been chosen to be compatible with respect to (13), the gradients will be non-zero, some lagrange multipliers will be positive, and it may then be necessary for the gradients to match for optimality with respect to the true plant. In this case the least-squares deviation from the target to setpoint is only the same as that of the true plant if the gradients $\nabla_{u_s} f_m, \nabla_{u_s} f_p$ match.

B. Application of the ISOPE method to a standard SSTO QP

The ISOPE method traditionally consists of iterating the following 3 steps [22], [6], where $(\cdot)^i$ indexes iterations:

1) A modified model-based OP (MMOP):

$$\begin{aligned} J_{mod}^*(u_s, u_s^i, \tilde{d}, \alpha^i) = \arg\min_{u_s} J(u_s, y_s) \\ -\mu(u_s^i, \alpha^i)^T u_s \quad \text{s. t. } y_s = f_m(u_s, \tilde{d}, \alpha^i) \end{aligned} \tag{24}$$

$$\begin{aligned} \text{where: } \mu(u_s^i, \alpha^i)^T = [\nabla_{y_s} J(u_s^i, y_s) + A_y^T \tau_y]. \\ \Big[\nabla_{u_s} f_m(u_s^i, \tilde{d}, \alpha^i) - \nabla_{u_s} f_p(u_s^i, d)\Big] \end{aligned} \tag{25}$$

2) A model parameter (α) estimation:

$$f_m(u_s^i, \tilde{d}, \alpha^i) = f_p(u_s^i, d) = y_p \tag{26}$$

3) Dampening the setpoint changes:

$$u_s^{i+1} = u_s^i + K_s[u_s - u_s^i], \quad K_s \in (0,1] \tag{27}$$

Note that i is iterated at a slower rate than k, with each iteration corresponding to having attained a steady state. A key insight in reconciling process optimization with the standard MPC approach presented in Section II is that the disturbance modeling through the use of the estimator ensures that $f_m(u_s^i, \tilde{d}, \alpha) \rightarrow f_p(u_s^i, d)$ where α is fixed[2] This is a trivial extension to [13, Thm. 1]: with $n_d = l$ it follows that the estimator error is zero, and so predicted and corrected augmented state estimates are equal, leading to the conclusion that $\tilde{x}_k \rightarrow x_s, u_k \rightarrow u_s$ and critically that $y_k \rightarrow y_s$. Thus neither the parameter estimation step nor the function modification term proposed in [22] is needed, and the method is called an *iterative gradient-based modification optimization, IGMO* problem. The usual arguments regarding optimality of the modified objective function are still applicable without adaption (estimation) of the model parameters, which are that once converged, the KKT conditions of the modified OP match those of the original OP (17) objective function with the true plant:

$$\begin{aligned}
&\mathscr{L}_{mod}(u_s, u_s^i, \tilde{d}) = J(u_s, y_s) + \left(y_s - f_m(u_s, \tilde{d})\right)^T \lambda \\
&\quad + g_y^T \tau_y + g_u^T \tau_u - \mu(u_s^i)^T u_s
\end{aligned} \tag{28}$$

$$\begin{aligned}
&\nabla_{u_s}\mathscr{L}_{mod}(u_s, u_s^i, \tilde{d}) = \nabla_{u_s} J(u_s, y_s) - \nabla_{u_s} f_m(u_s, \tilde{d})^T \lambda \\
&\quad + A_u^T \tau_u - \left[\nabla_{y_s} J(u_s^i, y_s) + A_y^T \tau_y\right] . \\
&\left[\nabla_{u_s} f_m(u_s^i, \tilde{d}) - \nabla_{u_s} f_p(u_s^i, d)\right]
\end{aligned} \tag{29}$$

The cost modification does not change the identity in (18) and so $\nabla_{y_s}\mathscr{L}_{mod} = \nabla_{y_s}\mathscr{L}$. Hence substituting the expression for λ in (18) into (30):

$$\begin{aligned}
&\nabla_{u_s}\mathscr{L}_{mod}(u_s, u_s^i, \tilde{d}) = \nabla_{u_s} J(u_s, y_s) + \nabla_{u_s} f_m(u_s, \tilde{d}) . \\
&[\nabla_{y_s} J(u_s, y_s) + A_y^T \tau_y] + A_u^T \tau_u \\
&- \left[\nabla_{y_s} J(u_s^i, y_s) + A_y^T \tau_y\right] . \\
&\qquad \left[\nabla_{u_s} f_m(u_s^i, \tilde{d}, \alpha^i) - \nabla_{u_s} f_p(u_s^i, d)\right]
\end{aligned} \tag{30}$$

$$\begin{aligned}
&\text{As } u_s \rightarrow u_s^i : \nabla_{u_s}\mathscr{L}_{mod}(u_s^i, \tilde{d}) \rightarrow \nabla_{u_s} J(u_s^i, y_s) \\
&\quad + A_u^T \tau_u + \left[\nabla_{y_s} J(u_s^i, y_s) + A_y^T \tau_y\right] . \nabla_{u_s^i} f_p(u_s^i, d)
\end{aligned} \tag{31}$$

C. Other issues with active steady-state constraints

When steady-state inequality constraints become active, which is the case for infeasible setpoints, it is likely that the model equality constraints need to match. There are situations where they need not match, which have been investigated in [20], [5]. These conditions, based on the uncertainty in G and the current setpoint and steady-state target, should be checked to determine whether SSTO modification is actually necessary whenever a steady-state is reached. It is also possible to modify the model-based constraint function [6] if necessary. If the inequality constraints are not a function of the process model (requiring omission of the *state* constraints and MCAS), then only the model equality constraint gradients need match. In a typical SSTO formulation such as in [12], [14], the inequality constraints are a function of $\tilde{d}$ only, and so constraint gradient modification is not necessary.

[2]i.e. α is determined a priori as a decision of the process model.

In section II-B it was suggested that it may be suitable to incorporate MCAS constraints into the SSTO. MCAS constraints are not based on easily measurable quantities, and so the method for dealing with them proposed in [6] is not appropriate. An alternative way to approach this issue would be to employ model-adaption methods to improve the accuracy of many-step ahead predictions. However, a simpler approach is to incorporate the MCAS into the SSTO (subsuming (15)) and accept the loss of optimality in the target that may occur due to active MCAS constraints.

D. Determination of derivative information

The MMOP relies on determination of the derivative $\nabla_{u_s} f_p(u_s^i, d)$. There are a number of methods ranging from generalized recursive least squares (requiring some sort of excitation/perturbations) to simpler methods using past setpoint information. To avoid the necessity for persistent additional perturbations/excitation, the simple derivative estimation method proposed in [6] shall be considered, although it is noted that systems identification methods may provide faster convergence [10]. The gradient shall be approximated using the following formula:

$$\nabla_{u_s} f_p(u_s^i, d) \approx (S^{a(i)})^{\dagger} . \begin{bmatrix} y_p^{(i)} - y_p^{a(i)} \\ y_p^{(i)} - y_p^{(i-1)} \\ \vdots \\ y_p^{(i)} - y_p^{(i-n_w-1)} \end{bmatrix} \tag{32}$$

$$S^{a(i)} = [u_s^{(i)} - u_s^{a(i)} \quad u_s^{(i)} - u_s^{(i-1)} \ \dots \ u_s^{(i)} - u_s^{(i-n_w-1)}]^T \tag{33}$$

where $(\cdot)^{\dagger}$ is the matrix pseudo-inverse obtained from its singular value decomposition (SVD) and n_w is the size of the window of past inputs/outputs used in the calculation. More setpoints $u_s^{a(i)}$ are added if necessary (based on the condition number, γ, of S^i), in order to keep S^i well conditioned. It is recommended in [6] that this be chosen through use of the following non-linear OP:

$$\begin{aligned}
&u_s^{a(i)*} = \arg\max_{u_s^{a(i)}} \left\{ (\gamma^{-1})^{a(i)} = \tfrac{\sigma_{min}(S^{a(i)})}{\sigma_{max}(S^{a(i)})} \right\} \\
&\text{s. t.} \quad u_s^{(i-1)} - \Delta u_s \le u_s^{a(i)} \le u_s^{(i-1)} + \Delta u_s \\
&\qquad \text{Steady-state constraints: } (14, 15)
\end{aligned} \tag{34}$$

Where Δu_s is chosen to limit how far the new steady-state can deviate. There is a trade-off in choosing this value between the conditioning of S and deviating too far from the current best estimate of the optimal steady-state. In [6] S is chosen to be square, but this is not necessary: more than $n_w = m$ points can be used, making the necessity for perturbations less frequent. However, too long a window should not be used or it will become difficult to recover the condition number of S with a single additional point.

Model gradients could possibly be used as extra points for forming S initially. It should be noted that as the non-linear optimization (34) cost may be non-convex, local minima solutions could result in the new point not changing at all. An alternative to the non-linear OP would be to rely on the statistical properties of a feasible random perturbation bounded in the same way by Δu_s, or to use this random pertubation as the initial seed for the non-linear OP (recommended). Modification of the actual SSTO to include a constraint on condition number has not been pursued as this would make the SSTO itself non-linear and possibly non-convex (for higher dimension problems) [2].

E. Use of a transitory perturbation SSTO

There is a problem with the direct use of the method proposed in [6]. The determination of $u_s^{a(i)}$ by the condition number optimization in (34) assumes that $\tilde{d}$ will remain constant. However, one of the main roles of the SSTO is to provide integral action through $\tilde{d}$, and if there is a setpoint change, it is likely that $\tilde{d}$ will change in order to take into account the model uncertainty. Consequently $u_s^{a(i)}$ may not be feasible. One solution would be to make the new steady-state $u_s^{a(i)}$ the desired steady state, and to modify the SSTO cost to find a feasible u_s as close as possible to $u_s^{a(i)}$. The new perturbation SSTO would be as follows:

$$J^*(u_s^*, y_s^*) = \arg\min_{u_s} \|u_s - u_s^{a(i)}\|^2 \quad \text{s. t. (14, 15)} \tag{35}$$

This would be employed until a new steady state is achieved.

F. Damping

Because the SSTO determines values of (x_s, u_s) that reject unmeasured disturbances to the system, it is unsuitable to dampen u_s^i in the same way as with the standard ISOPE algorithm. Damping should be performed earlier in the algorithm where the damping of μ^i is the most appropriate:

$$\mu^{i+1} = \mu^i + K_s[\mu - \mu^i] \tag{36}$$

G. The proposed algorithm

The modified architecture is depicted in figure 2, illustrating the following proposed algorithm:

Algorithm 3.1: 1) Initialize all states, with μ^0 = 0; perturbation flag (PF) = 0; converged flag (CF) = 0;
2) If PF = 0 perform modified SSTO; .
3) If PF = 1 perform perturbation SSTO;
4) Perform dynamic MPC optimization;
5) Estimate $\tilde{x}, \tilde{d}$ using a Kalman filter or otherwise $\rightarrow \tilde{y}$;
6) If CF = 1 and $\|[r_k^T, \hat{u}_k^T, \tilde{d}_k^T] - [r_{con}^T, \hat{u}_{con}^T, \tilde{d}_{con}^T]\| < \epsilon_{restart}$ goto step 2;
7) Determine whether at steady-state using either/and/or $\tilde{y}, \tilde{x}, \tilde{u}$; If not steady goto (2);
8) If PF = 0, and sufficient conditions for zero offset [20] have been met, and the choice of H has not resulted in a non-square thin system, and r and $\hat{u}$ are compatible w.r.t. (13) then goto (2);
9) Calculate S^i using (32). If ill-conditioned then determine new steady-state using a feasible random bounded perturbation;
10) Use feasible random bounded perturbation as a seed for non-linear optimization (34) (optional);
11) If PF = 1 goto step 2;
12) Estimate derivatives and calculate μ using (33).
13) Dampen μ using μ^i giving μ^{i+1};
14) If $\|\mu - \mu^i\| < \epsilon_{con}$ then set CF = 1; Record $r_{con}, \hat{u}_{con}, \tilde{d}_{con}$;
15) Goto step 2;

If μ has stopped changing appreciably (determined by ϵ_{con}), IMGO is deactivated until there is a significant change in the setpoints or disturbances (determined by $\epsilon_{restart}$).

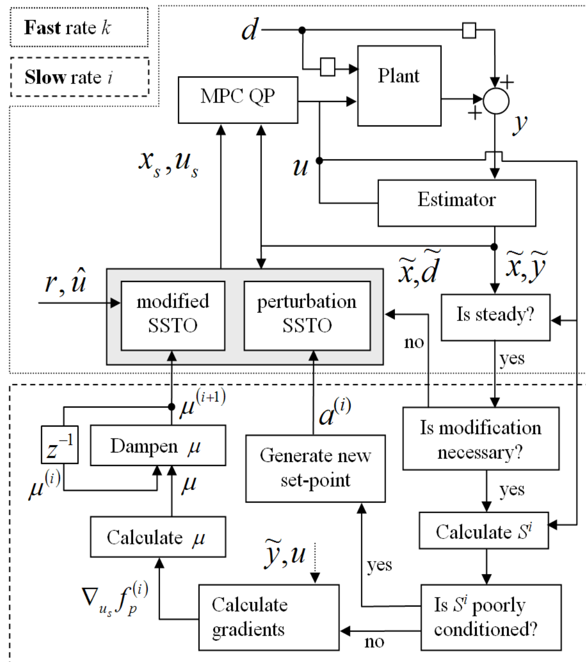

Fig. 2. Modified architecture diagram.

Remark 3.1: The IMGO technique can easily be applied to single-stage algorithms [21] if necessary simply by combining steps 2/3 and 4 with the cost being switched as necessary. The cost is only switched once a steady-state has been achieved, and so will not affect stability.

IV. Simulations

A motivating example for the problem of constrained steady-state offset with the MPC formulation of Section II is given in [19]. This system was contrived with simple input bounds to demonstrate constrained offset for infeasible setpoints. Replication of the results of the standard SSTO (5 - 7) + MPC algorithm (11) applied to this problem are given in figure 3. $(y_\infty =) y_{sm}^*$ is the (achieved) target set by the SSTO (unmodified in this case), whereas y_{sp}^* is the true optimal target. The IGMO algorithm has then been applied to remove the constrained offset. It was found that it was necessary to have a damping value for $K_s = \text{diag}(k_s), k_s \leq 0.5$ or the setpoints would not converge. n_w was chosen to be 10, and inclusion of the MCAS as opposed to (15) only did not affect the results (as the MCAS fully spanned the steady-state solution space). The results are shown in figure 4, depicting convergence to the true optimum.

V. Conclusions and future work

This paper has made clear the changes required to a modern MPC algorithm for achieving constrained offset-free control. Existing literature has not tended to discuss setpoint optimization in the context of modern MPC algorithms; hence other IMGO algorithms may include features (such as parameter estimation) which are not necessary. This paper therefore shows how IMGO approaches may be adapted to the context of modern MPC.

It has been identified in this work that provided that process-dependent constraints are not incorporated in the standard SSTO (which assumes a linear model with no integrating modes), an IMGO approach comprising on-line estimation of the process gradient and a suitable adjustment to the SSTO cost are all that is needed for constrained offset-free control. The cost adjustment proposed

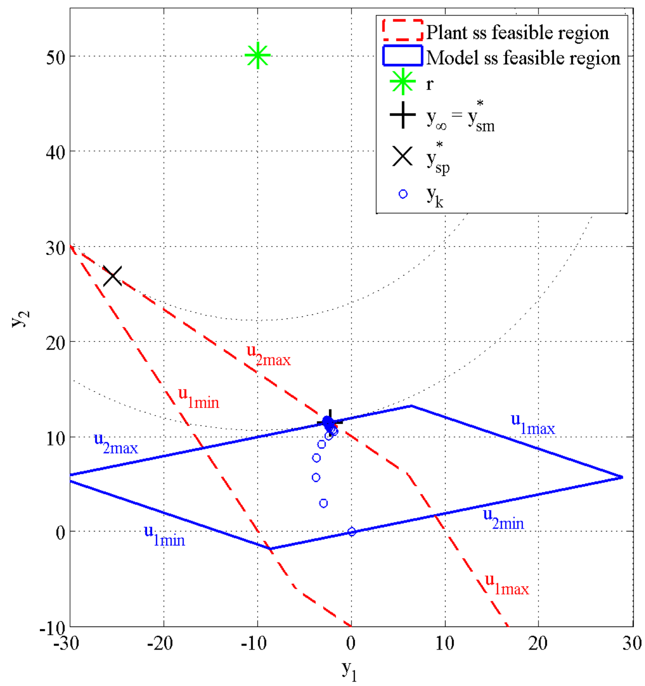

Fig. 3. Motivating example, with setpoint of (-10, 50), showing discrepancy between resulting simulated steady state and setpoint.

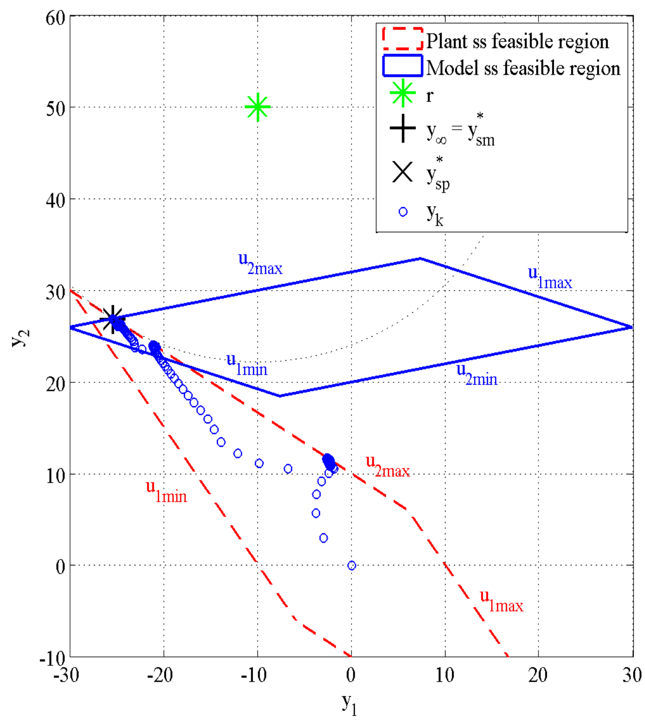

Fig. 4. Application of the IMGO algorithm, showing convergence to the constrained true optimum.

in [22] has been generally adopted in the literature, but this is not necessary for MPC algorithms that utilize an estimator. The proposed algorithm includes novel adjustments to those proposed previously, and includes valuable discussion of how modern MPC features (such as the MCAS) combine with the IMGO algorithms. An essential element that has been included is the conditions for constrained offset-free control in the problem formulation, which depend on the general accuracy of the process model: given a level of uncertainty, certain setpoints do not require activation of the IMGO algorithm, and so the necessary sub-optimal perturbations for numerical conditioning are then avoided. This advantage relies on full development of the conditions for offset-free control in a wide variety of scenarios, which has begun in [19], [20], and will be the main focus of future work.

References

[1] V. Becerra, P. Roberts, and G. Griffiths, "Novel developments in process optimisation using predictive control," *J Process Contr*, vol. 8, no. 2, pp. 117–138, 1998.

[2] M. Brdys and P. Tajewski, "An algorithm for steady-state optimising dual control of uncertain plants," in *Proc IFAC Workshop on New Trends in Des of Cont Sys*, 1994, pp. 249–254.

[3] E. Camacho and C. Bordons, *Model predictive control*. Springer, London, 2003.

[4] B. Chachuat, B. Srinivasan, and D. Bonvin, "Model parameterization tailored to real-time optimization," in *Proc ESCAPE 18*, 2008.

[5] J. Forbes and T. Marlin, "Model accuracy for economic optimizing controllers: the bias update case," *Ind Eng Chem Res*, vol. 33, pp. 1919–1929, 1994.

[6] W. Gao and S. Engell, "Iterative set-point optimization of batch chromatography," *Comput Chem Eng*, vol. 29, pp. 1401–1409, 2005.

[7] E. Gilbert and K. Tan, "Linear systems with state and control constraints: The theory and application of maximal output admissable sets," *IEEE T Automat Contr*, vol. 36, pp. 1008–1020, 1991.

[8] D. Limon, I. Alvarado, T. Alamo, and E. Camacho, "MPC for tracking of piece-wise constant references for constrained linear systems," in *Proc IFAC World Congress*, Prague, Czech Republic, 2005.

[9] J. Maciejowski, *Predictive control with constraints*. Prentice Hall Int, 2002.

[10] M. Mansour and J. Ellis, "Comparison of methods for estimating real process derivatives in on-line optimization," *Appl Math Model*, vol. 27, pp. 275–291, 2003.

[11] K. Muske and T. Badgwell, "Disturbance modelling for offset-free linear model predictive control," *J Process Contr*, vol. 12, pp. 617–632, 2002.

[12] K. Muske and J. Rawlings, "Model predictive control with linear models," *AIChE J*, vol. 39, no. 2, pp. 262–287, 1993.

[13] G. Pannocchia and J. Rawlings, "Disturbance models for offset-free model-predictive control," *AIChE J*, vol. 49, no. 2, pp. 426–437, 2003.

[14] C. Rao and J. Rawlings, "Steady states and constraints in model predictive control," *AIChE J*, vol. 45, no. 6, pp. 1266–1278, 1999.

[15] J. Rawlings, D. Bonne, J. Jorgensen, A. Venkat, and S. Jorgensen, "Unreachable set-points in model predictive control," 2007.

[16] P. Roberts, "An algorithm for steady-state system optimization and parameter estimation," *Int J Syst Sci*, vol. 10, pp. 719–734, 1979.

[17] J. Rossiter, *Model-based predictive control, a practical approach*. Prentice Hall Int, 2003.

[18] J. Rossiter, B. Kouvaritakis, and M. Rice, "A numerically robust state-space approach to stable predictive control strategies," *Automatica*, vol. 34, pp. 65–73, 1998.

[19] L. Shead, K. Muske, and J. Rossiter, "Conditions for which MPC fails to converge to the correct target," in *Proc IFAC World Congress*, 2008.

[20] ——, "Sufficient conditions for which MPC converges to the correct target," in *Proc Int Work Ass And Fut Dir NMPC*, 2008.

[21] L. Shead and J. Rossiter, "Feasibility for non-square linear MPC," in *Proc ACC*, 2007, pp. 4464–4469.

[22] P. Tajewski, "Iterative optimizing set-point control - the basic principle redesigned," in *Proc IFAC World Congress*, 2002.

[23] ——, "Advanced control and on-line process optimisation in multi-layer structures," *Annual Reviews in Control*, vol. 32, pp. 71–85, 2008.