8th International IFAC Symposium on
Dynamics and Control of Process Systems
Preprints Vol.1, June 6-8, 2007, Cancún, Mexico

MULTI-UNIT OPTIMIZATION WITH GRADIENT PROJECTION ON ACTIVE CONSTRAINTS

Lyne Woodward, Michel Perrier, Bala Srinivasan

*Department of Chemical Engineering,
École Polytechnique de Montréal,
Montréal, Canada, H3C 3A7*

Abstract: This paper addresses the problem of extremum-seeking (real-time steady state optimization of dynamic systems via control of the gradient) under constraints. Though gradient projection on active constraints has the advantage of providing better performance, it has not been utilized in the context of extremum-seeking due to difficulties in gradient estimation and drift from active constraints. In this paper, the gradient of the objective and the constraints are calculated simultaneously using the recently proposed multi-unit framework. Also, a correction term is added to handle the drift from active constraints. The theoretical concepts are illustrated on the optimization of a simple reactor.

Keywords: Real-time optimization, Extremum-seeking control, Inequality constraints, Gradient projection, Barrier function

1. INTRODUCTION

Process control methods typically deal with the problem of bringing a system to a desired set point and maintaining it therein. However, the real goal is often to optimize a certain performance criterion. For this, a model is built and is used to solve the optimization problem numerically. The model being just an approximation, the operating conditions computed numerically will only be sub-optimal for the real system. To handle this issue, the available measurements can be used to update the model, and the updated model can be optimized again. This model update can either mean adapting the parameters of a first-principles model (Marlin and Hrymak, 1997) or adding corrective terms (Desbiens and Shook, 2003).The two steps of model update and numerical optimization are repeated as often as needed.

When the objective function is convex, extremum-seeking control (Leblanc, 1922; Guay and Zhang, 2003) is another option for real-time optimization where controllers are designed so as to satisfy the necessary conditions of optimality. In the unconstrained case, this corresponds to pushing the gradient to zero.

The various extremum-seeking methods differ in the way in which the gradient is estimated. Perturbation methods (Leblanc, 1922; Krstic and Wang, 2000) use an input perturbation and compute the gradient using a correlation between the input and output variations. Adaptive extremum-seeking methods (Guay and Zhang, 2003) calculate the gradient based on a process model that is updated using available on-line measurements. In multi-unit optimization (Srinivasan, 2006), the gradient is computed as a finite difference between the outputs of multiple units with slightly different input values.

Though constraints play an important role in optimization, only a few papers consider constraints in the context of extremum-seeking (Dehaan and Guay, 2005). Therein, barrier functions are used to convert a constrained optimization problem into an unconstrained one. The current paper takes an alternate path through the projection on active constraints. This concept has the advantage of being on the active constraint, and thus leads to better performance. Though this idea is relatively old (Rosen, 1960), it has not been used in the context of extremum-seeking due to: (i) gradient with respect to the constraints are required, (ii) approximate gradients lead to deviation from the active constraint, and (iii) the resulting controller is hybrid in nature.

In this paper, the multi-unit technique is used for gradient computation, which can provide the gradients of the cost and the constraints simultaneously, with no extra effort. Secondly, the controller is designed such that the active constraints are met even if the gradients are approximate. Thirdly, the convergence could be established despite its hybrid nature.

The paper is organized as follows. Section 2 presents the extremum-seeking using perturbations and the barrier function approach as a point of comparison. In Section 3, multi-unit optimization is reviewed and the projection approach is elaborated in Section 4. Section 5 presents an example and Section 6 concludes the paper.

2. EXTREMUM-SEEKING CONTROL USING PERTURBATIONS

2.1 Optimization problem formulation

Consider a stable dynamic system with state $x \in \Re^n$, input $u \in \Re^m$ that has to be operated so as minimize a convex function $J(x,u)$:

$$\min_u \; J(x,u) \tag{1}$$

$$s.t. \quad \dot{x} = F(x,u) \equiv 0, \qquad S(x,u) \leq 0 \tag{2}$$

where $F(x,u)$ is the function describing the dynamics of the system and $S(x,u)$ the inequality constraints. The necessary conditions of optimality are:

$$\frac{dJ}{du} + \mu^T \frac{dS}{du} = 0 \tag{3}$$

$$\mu^T S = 0 \Rightarrow \mu_j = 0 \text{ or } S_j = 0. \tag{4}$$

where μ are the Lagrange multipliers and the total derivatives are given by,

$$\frac{dX}{du} = \left(\frac{\partial X}{\partial u} - \frac{\partial X}{\partial x} \left(\frac{\partial F}{\partial x} \right)^{-1} \frac{\partial F}{\partial u} \right) \tag{5}$$

where X is either J or S. Note that when a constraint is active, the corresponding Lagrange multiplier μ has a positive value, while it is zero when the constraint is inactive. This hybrid nature makes its resolution difficult.

An alternative is to transform the constrained problem into an unconstrained one using barrier functions (e.g. (Vassiliadis and Floudas, 1997)).

$$\min_u \; \bar{J}(x,u) = J(x,u) - \alpha \sum_i \log(-(S_i(x,u)) $$
$$s.t. \quad \dot{x} = F(x,u) \equiv 0 \tag{6}$$

The value of α will decide the distance from the constraints.

2.2 Extremum seeking

As in the steepest descent method for numerical optimization (Nocedal and Wright, 1999), extremum-seeking makes the process evolve in the opposite direction of the gradient. But instead of using the iteration index as in numerical methods, the real time is used here. The extremum-seeking control law is given by :

$$\dot{u} = -k \left(\frac{d\bar{J}}{du} \right)^T \tag{7}$$

2.3 Perturbation method

The key problem is the estimation of the gradient, which could be addressed using several methods. In the perturbation method (Leblanc, 1922; Krstic and Wang, 2000), a sinusoidal perturbation of amplitude a and frequency ω is added to the input (Fig. 1). The gradient is estimated using a high-pass filter, a modulation with the excitation signal and a low-pass filter. An integrator then implements the extremum-seeking control (7). The perturbation method can be used for processes with multiple inputs by choosing different frequencies/phase shift for each perturbation signals. Except for certain cases (Ariyur and Krstic, 2003),

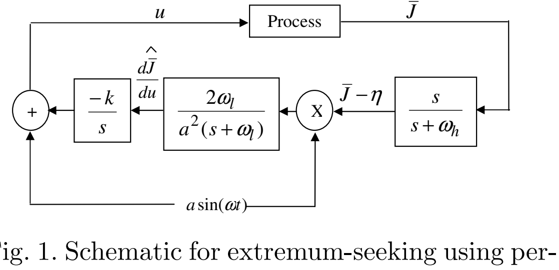

Fig. 1. Schematic for extremum-seeking using perturbations

the dynamics of the system should be relatively fast compared to the perturbation. Also, the filters must be slower than the perturbation, and the control of the gradient slower than the filters. This requires a three-fold time scale separation which slows down the optimization.

3. MULTI-UNIT OPTIMIZATION

Multi-unit optimization presents a faster alternative to gradient estimation, where a three-fold time-scale separation is not required. As shown in Fig. 2, the multi-unit optimization method (Srinivasan, 2006) requires a process with $(m+1)$ identical units, where m is the dimension of u. Examples of such processes are available in micro-array reactors and production lines.

The various units are operated with input values that are slightly different. The first unit, the reference, is operated at the input value u_0. The other units, $i = \{1, ..., m\}$ are operated at $u_i = u_0 + e_i\Delta$ with e_i the i^{th} unit vector. Then, the gradient can be estimated by :

$$\hat{\bar{g}}_i(u_o) = \frac{\bar{J}(x_i, u_i) - \bar{J}(x_0, u_0)}{\Delta} \tag{8}$$

where $\widehat{\left(\frac{d\bar{J}}{du}\right)} = \hat{\bar{g}}$. The extremum-seeking control law (7) is then used for all units:

$$\dot{u}_i = -k\hat{\bar{g}}^T(u_0) \; i \, \epsilon\{0, 1, ...m\} \tag{9}$$

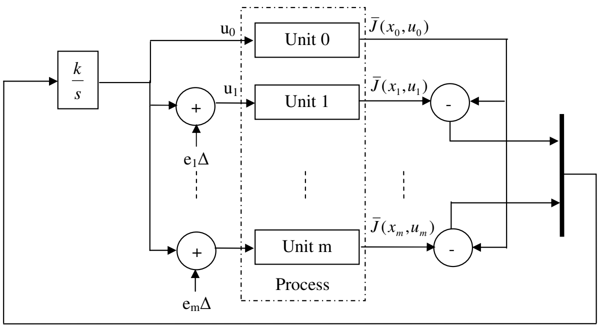

Fig. 2. Schematic for multi-unit unconstrained optimization

All units follow the same control law and always keep an input difference of Δ one from each other. The convergence of this scheme to a ball around the optimum has been proven despite the errors caused by the dynamics (which is assumed to be stable) and the error due to finite differences.

In contrast to extremum-seeking using perturbations, the perturbation here is not in the time-domain but in the dimension of the system units, which in turn implies that the time-scale separation is not needed. Thus, the optimum will be reached much faster with the multi-unit method than with the perturbation method.

4. MULTI-UNIT OPTIMIZATION AND GRADIENT PROJECTION

4.1 Constrained optimization using gradient projection

In the gradient projection method (Rosen, 1960; Rosen and Kreuser, 1972), the descent direction is the negative of the gradient projected onto the active constraints. To understand this, consider the necessary conditions of optimality (3). As the Lagrange multipliers associated with the inactive inequality constraints are zero, the optimality conditions can be expressed using the active inequality constraints only. Let $\bar{S}$ be the set of active constraints and $\bar{\mu}$ the Lagrange multipliers associated to these active constraints. Denoting $g = \frac{dJ}{du}$ and $M = \frac{d\bar{S}}{du}$, (3) becomes:

$$g + \bar{\mu}^T M = 0 \tag{10}$$

Then, the Lagrange multipliers of the active constraints are expressed by $\bar{\mu}^T = -gM^+$, where $M^+ = Q(MQ)^{-1}$ is the pseudo-inverse matrix of M, with any choice of Q such that (MQ) is invertible. Using $\bar{\mu}^T = -gM^+$ in (10) gives,

$$gP = 0, \qquad P = I - M^+M \tag{11}$$

The traditional gradient projection makes the choice $Q = M^T$ and uses $\dot{u} = -kP^Tg^T$. Note that from the definition of the projection $MP = 0$, but with this particular choice of $Q = M^T$, $MP^T = 0$ also. So, the variation in the active constraints is given by,

$$\dot{\bar{S}} = \frac{d\bar{S}}{du}\dot{u} = -kMP^Tg^T = 0 \tag{12}$$

Though this is a nice property, the major difficulty is that $\bar{S} = 0$ is not guaranteed if the gradients are in error or if there are disturbances.

4.2 Generalized gradient projection

Note that the optimality conditions have two parts: the active constraints, $\bar{S} = 0$, and the gradient projection condition, $gP = 0$. Modifying the extremum-seeking control (7) to include the two parts of the optimality conditions gives:

$$\dot{u} = -kP^Tg^T - \beta M^+\bar{S} \tag{13}$$

A non-zero β is required to ride along the active constraint. With $\beta > 0$, $\dot{\bar{S}} = -\beta\bar{S} - kMP^Tg^T$ corresponds to a stable system that forces $\bar{S} = 0$. The interesting aspect now is that due to the extra controller, any choice of Q would work and not necessarily $Q = M^T$.

A simple choice of Q would be $Q = [I \; 0]^T$, where I is an identity matrix of dimension $dim(\bar{S}) \times$

$dim(\bar{S})$. This results in a partitioning of the input space $u^T = [\bar{u}^T \ \tilde{u}^T]$ and in M, i.e. $M = [\bar{M} \ \tilde{M}]$. The adaptation laws for $\bar{u}$ and $\tilde{u}$ can be calculated from (13) by computing P with this choice of Q:

$$\dot{\bar{u}} = -\beta \bar{M}^{-1}\bar{S} \quad \dot{\tilde{u}} = -k\left[-\bar{M}^{-1}\tilde{M} \quad I\right] g^T \tag{14}$$

Note that with such a choice of Q, the two tasks of (13), i.e. (i) to keep the constraint active and (ii) force the reduced gradient to zero, are done by two parts of the input vector. The two tasks are essentially decoupled and any of the input variables can be assigned for keeping the constraints active as long as (MQ) is invertible.

4.3 Switching logic to determine the set of active constraints

The following switching logic can be used to determine the active set. This leads to a hybrid system and the transition is initiated by two events:

(1) A constraint is included in the active set when the constraint is hit.
(2) A constraint is removed from the active set when the Lagrange multiplier corresponding to the active constraint hits zero.

4.4 Multi-unit gradient estimation with gradient projection

The scheme with multi-unit gradient estimation and gradient projection is presented in Fig. 3. It includes two additional blocks corresponding to the switching logic to determine active constraints and gradient projection. Also, the controller is slightly more complex than the simple integral controller used earlier. The particularity of multi-

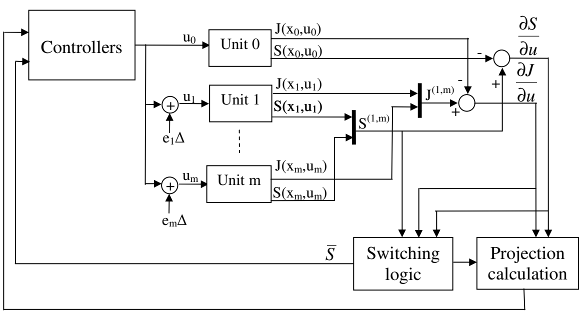

Fig. 3. Schematic for multi-unit constrained optimization with gradient projection

unit gradient estimation in this context needs to be stressed. Firstly, the objective and the constraints are assumed to be measured and so the estimation of the gradients of both the objective and constraints are obtained simultaneously. No extra units are required to calculate the constraint gradients. A similar finite difference formula is used to compute the gradient of the constraints.

$$\widehat{\frac{dS_j}{du_i}} = \frac{S_j(x_i, u_i) - S_j(x_0, u_0)}{\Delta} \tag{15}$$

5. ILLUSTRATIVE EXAMPLE

5.1 Description of the system

An isothermal continuous stirred-tank reactor with the two reactions $A + B \rightarrow C$ and $2B \rightarrow D$ is considered, with C being the desired product and D an undesired product. The reactor is fed by two streams with flow rates F_a and F_b and inlet concentrations $c_{A_{in}}$ and $c_{B_{in}}$, respectively.

The cost function is the amount of product C, $(F_a + F_b)c_C$, weighted by the yield factor $(F_a + F_b)c_C/F_a c_{A_{in}}$. The constraints of the problem include the upper bounds on the heat generated and on the total flow. The optimization problem is stated mathematically as follows:

$$\begin{aligned}
\max_{F_a,F_b} \ & J = \frac{(F_a+F_b)^2 c_C^2}{F_a c_{A_{in}}} \qquad (16)\\
s.t. \ & \dot{c}_A = \frac{F_a}{V}c_{A_{in}} - \frac{F_a+F_b}{V}c_A - r_1 \equiv 0\\
& \dot{c}_B = \frac{F_b}{V}c_{B_{in}} - \frac{F_a+F_b}{V}c_B - r_1 - r_2 \equiv 0\\
& \dot{c}_c = -\frac{F_a+F_b}{V}c_C + r_1 \equiv 0\\
& q - q_{max} \leq 0, \qquad (F_a+F_b) - F_{max} \leq 0\\
& r_1 = k_1 c_A c_B, \qquad r_2 = 2k_2 c_B^2\\
& q = r_1(-\Delta H_1)V + r_2(-\Delta H_2)V
\end{aligned}$$

where c_X is the concentration of species X, k_i the rate constants, r_i the reaction rates, $(-\Delta H_i)$ the enthalpies of the two reactions, q the heat produced by the reactions, V the reactor volume, and q_{max} and F_{max}, the bounds on the heat production and the total flow rate, respectively. The numerical values used in this study are given in Table 1. c_{A_0}, c_{B_0}, c_{C_0}, F_{a_0} and F_{b_0} are the initial steady-state values of the system. The

Table 1. Parameter values and initial conditions

k_1	1.5	l mol/h	$(-\Delta H_1)$	7×10^4	J/mol
k_2	0.014	l mol/h	$(-\Delta H_2)$	5×10^4	J/mol
V	500	l	$c_{A_{in}}$	2	mol/l
q_{max}	10^6	J/h	$c_{B_{in}}$	1.5	mol/l
F_{max}	22	l/h	c_{A_0}	0.085	mol/l
F_{a_0}	7	l/h	c_{B_0}	0.195	mol/l
F_{b_0}	11	l/h	c_{C_0}	0.692	mol/l

optimal operating conditions correspond to $F_a = 7.62\ l/h$ and $F_b = 13.1\ l/h$. The constraint on the

heat produced is active i.e. $q = q_{max}$ and the valu
of the objective function is 12.3 mol/h.

5.2 *Perturbation method and barrier function*

Perturbation method with the logarithmic barri function was used with the following parameter

- Excitation: $a = 0.1$, $\omega = 1\ rad/h$, $\sin(\omega$ and $\cos(\omega t)$ for the two inputs.
- Filters: $\omega_l = \omega_h = 0.2\ rad/h$
- Adaptation gains: $k_{F_a} = k_{F_b} = 1/8$
- Log barrier gain: $\alpha = 0.1$

Since the settling time is around 0.6 h, the fr quency of the perturbation was chosen to be (1/ h). The lower the log barrier gain, the closer the system is to the constraint. However the excitation amplitude limits how close we can get to the constraint, thereby providing a lower limit on the log barrier gain. The maximum adaptation gains values with which the system still converges were used. The optimum is reached in about 150 hours as shown on Fig. 4 and the value was 12.15 mol/h.

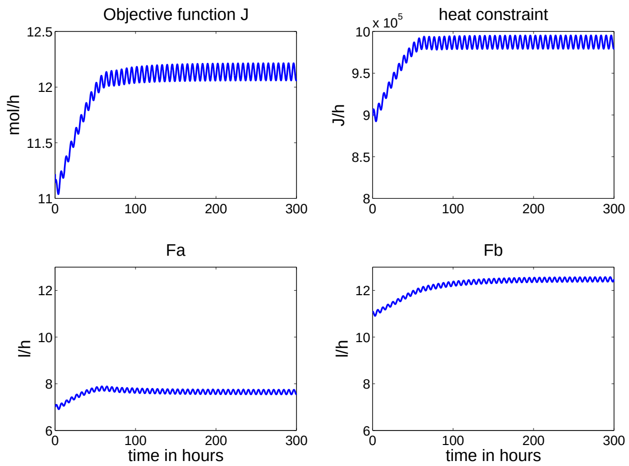

Fig. 4. Results of the perturbation method with a logarithmic barrier function

5.3 *Multi-unit method and barrier function*

Next, the multi-unit method with the logarithmic barrier function was used with the following parameters:

- Excitation: $\Delta_{F_a} = \Delta_{F_b} = 0.1$
- Adaptation gains: $k_{F_a} = k_{F_b} = 100$
- Log barrier gain: $\alpha = 0.1$

As shown in Fig. 5, the optimum is reached in about 1.5 hours, 100 times faster than the perturbation method, thanks to a higher adaptation gain. The final values are provided in Table 2.

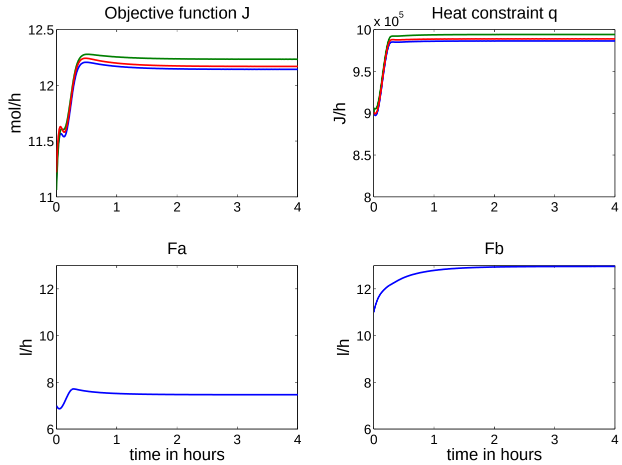

Fig. 5. Results of the multi-unit optimization method with a logarithmic barrier function

5.4 *Multi-unit method and gradient projection*

The multi-unit method with projection of the gradient on active constraints (14) was also implemented. These control laws for various hybrid states are given below :

- No active constraints
  $\dot{F}_a = k_{F_a}\frac{(J_1 - J_0)}{\Delta F} \qquad \dot{F}_b = k_{F_b}\frac{(J_2 - J_0)}{\Delta F}$
- Heat constraint of unit j is active
  $\dot{F}_a = \beta_q(q_{max} - q_j)$
  $\dot{F}_b = k_{F_b}\left(\frac{(J_2 - J_0)}{\Delta F} - \frac{(q_2 - q_0)(J_1 - J_0)}{(q_1 - q_0)\Delta F}\right)$
- Flow constraint of unit $\bar{j}$ is active
  $\dot{F}_a = k_{F_a}\left(\frac{(J_1 - J_0)}{\Delta F} - \frac{(F_1 - F_0)(J_2 - J_0)}{(F_2 - F_0)\Delta F}\right)$
  $\dot{F}_b = \beta_F(F_{max} - F_{\bar{j}})$
- Both constraints active
  $\dot{F}_a = \beta_q(q_{max} - q_j) \qquad \dot{F}_b = \beta_F(F_{max} - F_{\bar{j}})$

where J_i and q_i are the objective function measurement and the heat measurement of unit i respectively and $F_i = F_a + F_b$. The pairing was chosen arbitrarily, i.e F_a with the q_{max} constraint and F_b with the F_{max} constraint. The switching logic is shown in Fig. 6. The following parameters were used:

- Excitation: $\Delta_{F_a} = \Delta_{F_b} = 0.1$
- Adaptation gains: $k_{F_a} = k_{F_b} = 100$
- Constraint gains : $\beta_q = 0.006$, $\beta_F = 8$

The evolution of the input values is shown in Fig. 7. First, the system evolves without any constraint being active. Then, the unit 1 hits the heat constraint and from then on, F_a is controlled to keep this constraint active. The other flow helps to reach the optimum by following the gradient projected on this active constraint. Table 2 provides a comparison of the results obtained using a barrier function with those obtained using the gradient projection. It can be seen that a better cost is obtained by projection as compared to the barrier approach by being closer to the constraint.

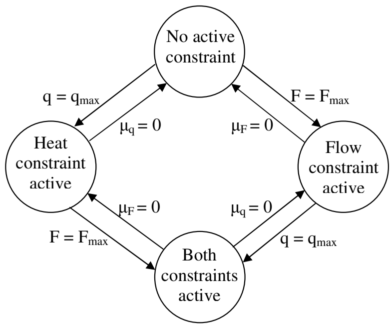

Fig. 6. Switching logic for multi-unit optimization with gradient projection

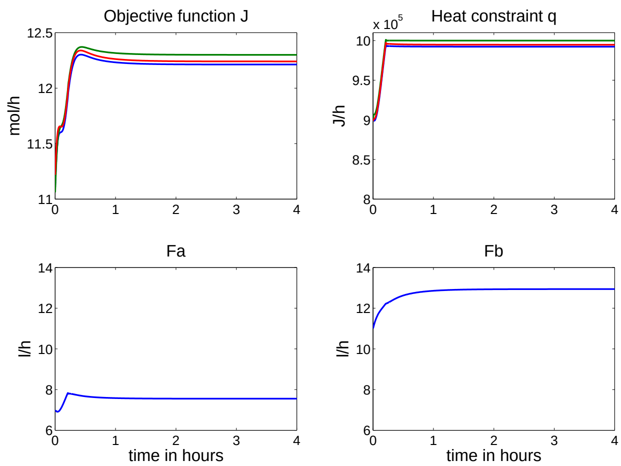

Fig. 7. Results of the multi-unit optimization method with projection

However, since the inputs to the different units in the multi-unit method are non-identical, only one of the units is on the constraint, while the others are away from the constraint by a distance determine by Δ.

Table 2. Cost, constraint and the input values for various units and optimization strategies

	unit	$F_a(l/h)$	$F_b(l/h)$	$J(mol/h)$	$q(kJ/h)$
M-U	0	7.47	12.96	12.14	986
with	1	7.57	12.96	12.23	994
barr.	2	7.47	13.06	12.17	989
M-U	0	7.55	12.94	12.21	992
with	1	**7.65**	**12.94**	**12.30**	**1000**
proj.	2	7.55	13.04	12.24	995

6. CONCLUSION

In this paper, real-time optimization under inequality constraints was addressed using the projection of gradient on the active constraints. The projection was generalized to take into account the drift from the active constraints due to potential error in the gradient estimation. Also, a hybrid control law was proposed to take care of the changes in active constraints. Multi-unit gradient estimation scheme was used to provide the gradients of the cost and constraints.

Despite a major improvement in terms of rate of convergence to the optimum, the main disadvantage of the multi-unit method is the strong assumption that the units are perfectly identical. Future work will try to extend the application of this method to process with multiple units which are similar but not exactly identical.

REFERENCES

Ariyur, K.B. and M. Krstic (2003). *Real-time Optimization by Extremum-Seeking Control*. John Wiley and Sons.

Dehaan, D. and M. Guay (2005). Extremum-seeking control of state-constrained nonlinear systems. *Automatica* **41**(9), 1567–74.

Desbiens, A. and A.A. Shook (2003). IMC-optimization of a direct reduced iron phenomenological simulator. *4th International Conference on Control and Automation* pp. 446–450.

Guay, M. and T. Zhang (2003). Adaptive extremum seeking control of nonlinear dynamic systems with parametric uncertainties. *Automatica* **39**(7), 1283–93.

Krstic, M. and H.-H. Wang (2000). Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica* **36**(4), 595–601.

Leblanc, M. (1922). Sur l'électrification des chemins de fer au moyen de courants alternatifs de fréquence élevée. *Revue générale de l'électricité* **12**(8), 275–277.

Marlin, T.E. and A.N. Hrymak (1997). Real-time operations optimization of continuous processes. Vol. 316 of *AIChE Symposium Series*. p. 156.

Nocedal, J. and S. Wright (1999). *Numerical optimization*. Springer. New York.

Rosen, J.B. (1960). Gradient projection method for nonlinear programming. *Society for Industrial and Applied Mathematics Journal* **8**(1), 181–217.

Rosen, J.B. and J. Kreuser (1972). A gradient projection algorithm for non-linear constraints. *Conference on numerical methods for nonlinear optimization* pp. 297–300.

Srinivasan, B. (2006). Real-time optimisation of dynamic systems using multiple units. *International Journal of Robust and Nonlinear Control*. DOI: 10.1002/rnc.1165.

Vassiliadis, V. S. and C. A. Floudas (1997). Modified barrier function approach for large-scale optimization. *Computers and Chemical Engineering* **21**(8), 855–874.