Computers & Chemical Engineering, volume 54, July 2013, pages 1-7

Accepted Manuscript

Title: Branch and Bound Method for Regression-Based Controlled Variable Selection

Author: <ce:author id="aut0005"> Vinay Kariwala<ce:author id="aut0010"> Lingjian Ye<ce:author id="aut0015"> Yi Cao

PII: S0098-1354(13)00071-9
DOI: http://dx.doi.org/doi:10.1016/j.compchemeng.2013.03.006
Reference: CACE 4663

To appear in: *Computers and Chemical Engineering*

Received date: 1-12-2012
Revised date: 22-2-2013
Accepted date: 3-3-2013

Please cite this article as: Vinay Kariwala, Lingjian Ye, Yi Cao, Branch and Bound Method for Regression-Based Controlled Variable Selection, *Computers and Chemical Engineering* (2013), http://dx.doi.org/10.1016/j.compchemeng.2013.03.006

Highlights

- A new Branch and Bound algorithm for regression based globally optimal controlled variable selection is proposed.
- Local shortcoming of existing self-optimizing control approach is overcome.
- New pruning algorithm makes the Branch and Bound much more efficient.
- Efficiency and effectiveness of the algorithm are demonstrated though numerical examples.

Branch and Bound Method for Regression-Based Controlled Variable Selection

Vinay Kariwala [a] Lingjian Ye [b] Yi Cao [c,*]

[a] *ABB Global Industries & Services Ltd, Mahadevpura, Bangalore 560048, India*

[b] *Ningbo Institute of Technology, Zhejiang University 315100, Ningbo, Zhejiang, China*

[c] *School of Engineering, Cranfield University, Cranfield, Bedford MK43 0AL, UK*

Abstract

Self-optimizing control is a promising method for selection of controlled variables (CVs) from available measurements. Recently, Ye et al. [2012] have proposed a globally optimal method for selection of self-optimizing CVs by converting the CV selection problem into a regression problem. In this approach, the necessary conditions of optimality (NCO) are approximated with linear combinations of available measurements over the entire operation region. In practice, it is desired that a subset of available measurements be combined as CVs to obtain a good trade-off between the economic performance and the complexity of control system. The subset selection problem, however, is combinatorial in nature, which makes the application of the globally optimal CV selection method to large-scale processes difficult. In this work, an efficient branch and bound (BAB) algorithm is developed to handle the computational complexity associated with the selection of globally optimal CVs. The proposed BAB algorithm identifies the best measurement subset such that the regression error in approximating NCO is minimized and is also applicable to the general regression problem. Numerical tests using randomly generated matrices and a binary distillation column case study demonstrate the computational efficiency of the proposed BAB algorithm.

Key words: Branch and bound, Control structure design, Controlled variables, Combinatorial optimization, Distillation, Self-optimizing control.

* Corresponding author: Tel: +44 1234 750111; Fax: +44 1234 754685, E-mail: y.cao@cranfield.ac.uk

1 Introduction

The selection of controlled variables (CVs) from available measurements is an important task during the design of control systems. For CV selection, several methods based on controllability and achievable performance as the selection criteria have been proposed in the literature; see *e.g.* Van De Wal and De Jager [2001] for an overview. Recently, Skogestad [2000] introduced the concept of self-optimizing control for selection of CVs based on process economics. In this approach, CVs are selected such that in presence of disturbances, as compared to the use of a real-time optimizer, the loss incurred in implementing the operational policy by holding the selected CVs at constant setpoints is minimal. The advantages of self-optimizing control approach for CV selection has been demonstrated through several case studies [Rangaiah and Kariwala, 2012].

The choice of CVs based on the general non-linear formulation of self-optimizing control requires solving large-dimensional non-convex optimization problems [Skogestad, 2000]. To quickly pre-screen alternatives, minimum singular value rule [Skogestad and Postlethwaite, 2005] as well as exact local methods with worst-case [Halvorsen et al., 2003] and average loss minimization [Kariwala et al., 2008] have been proposed. These local methods are useful for selecting a subset or linear combinations of available measurements as CVs, where the latter approach provides lower losses. Recently, explicit solutions to the problem of finding locally optimal measurement combinations have been proposed [Alstad et al., 2009; Heldt, 2010; Kariwala, 2007; Kariwala et al., 2008]. Hu et al. [2012] have proposed a local method to explicitly handle the input and output constraints during CV selection.

The available CV selection criteria are derived based on local linearization of the process model. Recently, a globally optimal CV selection framework has been proposed by Ye et al. [2012, 2013]. In this framework, the CV synthesis problem is converted into a regression problem using CVs as measurement combinations to approximate the Necessary Conditions of Optimality (NCO) globally over the entire operation region. It has been proven that the average loss is globally minimized when the regression error is minimal over the entire operation region and the measurement combinations as CVs are perfectly controlled at zero. A number of linear and nonlinear regression models have been adopted to approximate the NCO. Case studies showed that all these models are able to significantly reduce the average loss, as compared to those CVs designed using existing local methods.

It is known that the use of combinations of a few measurements as CVs often provide similar loss as the case where combinations of all available measurements are used [Alstad et al., 2009; Kariwala, 2007; Kariwala et al., 2008; Ye et al., 2013]. Though the former approach results in control structures with

lower complexity, it gives rise to a combinatorial optimization problem involving selection of measurements, whose combinations can be used as CVs. For local self-optimizing control methods, a number of efficient branch and bound (BAB) approaches, called bidirectional BAB (B^3) methods have been proposed to solve this combinatorial problem efficiently [Cao and Kariwala, 2008; Kariwala and Cao, 2009; Kariwala and Y.Cao, 2010]. These BAB algorithms are not required for the selection of individual measurements as globally optimal CVs, as approximations of individual gradients are not correlated and can be solved separately overcoming combinatorial issues. However, to select measurement combinations, the combinatorial difficulty still exists for the global CV selection problem.

With this motivation, the BAB framework is extended to measurement subset selection for synthesis of globally optimal CVs chosen as linear combinations of measurements. It is proven that the selection criterion is equivalent to a quadratic problem, for which a standard BAB algorithm [Narendra and Fukunaga, 1977] exists. The standard algorithm is improved into a downwards BAB algorithm. The efficiency and effectiveness of the proposed BAB algorithm is demonstrated through randomly generated matrices and a distillation case study [Skogestad, 1997].

The rest of the paper is organized as follows: the local and global methods for CV selection using the concept of self-optimizing control are discussed in Sections 2 and 3, respectively. The general principle of BAB approach and its adaptation for global CV selection are presented in Section 4. Numerical examples to demonstrate the efficiency of proposed BAB method are shown in Section 5. Finally, Section 6 concludes the paper.

2 Local Methods for Self-Optimizing Control

Consider that the steady-state economics of the plant is characterized by the scalar objective function $J(\mathbf{u}, \mathbf{d})$, where $\mathbf{u} \in \mathbb{R}^{n_u}$ and $\mathbf{d} \in \mathbb{R}^{n_d}$ are inputs and disturbances, respectively. The optimal operation policy is to update $\mathbf{u}$ according to $\mathbf{d}$, which usually requires the use of an online optimizer. For this case, let the optimal value of the objective function be denoted as $J_{\text{opt}}(\mathbf{d})$. A simpler strategy involves indirect adjustment of $\mathbf{u}$ using a feedback controller. In this case, the feedback controller manipulates $\mathbf{u}$ to hold the CVs $\mathbf{c}$ close to their specified setpoints. Here, in addition to $\mathbf{d}$, J is also affected by the error $\mathbf{e}$ in implementing the constant setpoint policy, which results due to uncertainty and measurement noise. The suboptimal objective functional value under the second strategy is denoted as $J_c(\mathbf{n}, \mathbf{d})$. Then, the worst-case and average losses due to the use of the suboptimal strategy are given as

$$\text{Worst-case loss} = \max_{\mathbf{e}\in\mathcal{E}} \max_{\mathbf{d}\in\mathcal{D}} \left(J_{\text{opt}}(\mathbf{d}) - J_c(\mathbf{n},\mathbf{d})\right) \tag{1}$$

$$\text{Average loss} = E[J_{\text{opt}}(\mathbf{d}) - J_c(\mathbf{n},\mathbf{d})] \tag{2}$$

where $\mathcal{D}$ and $\mathcal{E}$ represent the sets of allowable disturbances and implementation errors, respectively, and E is the expectation operator. Self-optimizing control is said to occur, when we can achieve an acceptable loss by holding the CVs close to their setpoints without the need to reoptimize when disturbances occur [Skogestad, 2000]. Based on this concept, the appropriate CVs can be selected by comparing the losses for different alternatives.

As mentioned earlier, the use of nonlinear formulation of self-optimizing control is difficult. Hence, some local methods were developed to estimate the losses defined in (1) and (2) by linearising the process model around the normally optimal operating point as follows:

$$\mathbf{y} = \mathbf{M}^y\,\mathbf{u} + \mathbf{M}_d^y\,\mathbf{W}_d\,\mathbf{d} + \mathbf{W}_e\,\mathbf{e} \tag{3}$$

where $\mathbf{y} \in \mathbb{R}^{n_y}$ denotes the process measurements and $\mathbf{e} \in \mathbb{R}^{n_y}$ denotes the implementation error. Here, the diagonal matrices $\mathbf{W}_d$ and $\mathbf{W}_e$ contain the expected magnitudes of disturbances and implementation error, respectively. The CVs $\mathbf{c} \in \mathbb{R}^{n_u}$ are given as

$$\mathbf{c} = \mathbf{H}\,\mathbf{y} = \mathbf{M}\,\mathbf{u} + \mathbf{M}_d\,\mathbf{W}_d\,\mathbf{d} + \mathbf{H}\,\mathbf{W}_e\,\mathbf{e} \tag{4}$$

where $\mathbf{H}$ is a selection or combination matrix and

$$\mathbf{M} = \mathbf{H}\,\mathbf{M}^y, \quad \mathbf{M}_d = \mathbf{H}\,\mathbf{M}_d^y \tag{5}$$

It is assumed that $\mathbf{M} \in \mathbb{R}^{n_u \times n_u}$ is invertible. This assumption is necessary for integral control. When $\mathbf{d}$ and $\mathbf{e}$ are assumed to be uniformly distributed over the set

$$\left\| \begin{bmatrix} \mathbf{d}^T & \mathbf{e}^T \end{bmatrix}^T \right\|_2 \leq 1 \tag{6}$$

the local worst-case and average losses are given as [Halvorsen et al., 2003; Kariwala et al., 2008]:

$$L_{\text{worst}}(\mathbf{H}) = 0.5\bar{\sigma}^2 \left(J_{uu}^{1/2}(\mathbf{H}\,\mathbf{M}^y)^{-1}\mathbf{H}\,\mathbf{P}\right) \tag{7}$$

$$L_{\text{average}}(\mathbf{H}) = \frac{1}{6(n_y + n_d)} \left\| J_{uu}^{1/2}(\mathbf{H}\,\mathbf{M}^y)^{-1}\mathbf{H}\,\mathbf{P} \right\|_F^2 \tag{8}$$

where $\bar{\sigma}$ and $\|\cdot\|_F$ denote the maximum singular value and Frobenius norm, respectively, and

$$\mathbf{P} = \left[\left(\mathbf{M}^y \, J_{uu}^{-1} \, J_{ud} - \mathbf{M}_d^y\right) \mathbf{W}_d \quad \mathbf{W}_e \right] \tag{9}$$

with $J_{uu} = \frac{\partial^2 J}{\partial \mathbf{u}^2}$ and $J_{ud} = \frac{\partial^2 J}{\partial \mathbf{u} \partial \mathbf{d}}$, evaluated at the nominal operating point. In comparison with worst-case loss, the selection of CVs is preferred through minimization of average loss, as the worst-case may not occur frequently in practice [Kariwala et al., 2008].

When individual measurements are selected as CVs, $\mathbf{H}$ can be considered to be a selection matrix. Instead of using individual measurements, it is possible to use combinations of measurements as CVs. For this case, the following explicit expression for $\mathbf{H}$ can be derived, which minimizes the L_{average} in (8) [Alstad et al., 2009; Rangaiah and Kariwala, 2012]:

$$\mathbf{H}^T = (\mathbf{P}\mathbf{P}^T)^{-1}\mathbf{M}^y \tag{10}$$

As shown by Kariwala et al. [2008], the $\mathbf{H}$ in (10) also minimizes L_{worst} in (7). The locally optimal combinations of all the available measurements, which can be used as CVs can be found using (10).

3 Globally Optimal Method

The local methods [Halvorsen et al., 2003; Kariwala et al., 2008] are based on linearization around the nominally optimal operating point. Therefore, the identified CVs are only locally optimal. To derive globally optimal solution CVs, it is assumed that the NCO is approximated by CVs and the CVs are perfectly controlled at zero. Then, the loss, $L(\mathbf{d})$ for a particular disturbance $\mathbf{d}$, due to the approximation error, $\boldsymbol{\epsilon}(\mathbf{d})$ can be expressed as [Ye et al., 2012, 2013]:

$$L(\mathbf{d}) = 0.5\boldsymbol{\epsilon}^T(\mathbf{d})\mathbf{Q}^{-1}(\mathbf{d})\boldsymbol{\epsilon}(\mathbf{d}) \tag{11}$$

where $\mathbf{Q}(\mathbf{d})$ is the reduced Hessian of the cost function evaluated at point where the CV, $\mathbf{c}(\mathbf{d})$ is perfectly controlled corresponding to particular disturbance, $\mathbf{d}$, whilst $\boldsymbol{\epsilon}(\mathbf{d}) = \mathbf{g}(\mathbf{d}) - \mathbf{c}(\mathbf{d})$, where $\mathbf{g}(\mathbf{d})$ is the reduced gradient evaluated at the same point.

The average loss over the entire operation region, $\mathcal{D}$ can be represented as,

$$\bar{L} = E_{\mathbf{d}\in\mathcal{D}} L(\mathbf{d}) \approx \frac{1}{2N}\sum_{i=1}^{N} \boldsymbol{\epsilon}(\mathbf{d}_i)^T \mathbf{Q}^{-1}(\mathbf{d}_i)\boldsymbol{\epsilon}(\mathbf{d}_i) \tag{12}$$

where $\mathbf{d}_i \in \mathcal{D}$, $i = 1, \ldots, N$ are N samples of disturbances in $\mathcal{D}$.

According to (12), the loss minimization is equivalent to a least squares regression problem to minimize the weighted residual, $\mathbf{Q}^{-1/2}\boldsymbol{\epsilon}$. However, due to the difficulty and reliability to evaluate the reduced Hessian for every $\mathbf{d}_i \in \mathcal{D}$, $\mathbf{Q}(\mathbf{d}_i)$ can be replaced by a constant matrix, *e.g.* the identity matrix or $\mathbf{Q}$ evaluated at nominal value of $\mathbf{d}$. To simplify discussion, in this work, we assume that $\mathbf{Q} = \mathbf{I}$. Then the regression problem can be set up as discussed next.

Let CVs $\mathbf{c} = \boldsymbol{\Theta}\hat{\mathbf{y}}$, where $\hat{\mathbf{y}} = \begin{bmatrix} 1 & \mathbf{y}^T \end{bmatrix}^T$ and $\boldsymbol{\Theta}$ is a $n_u \times (n_y + 1)$ measurement weight matrix, which needs to be determined. Let the entire operation region be sampled by N points for independent variables (input, $\mathbf{u}_1, \cdots, \mathbf{u}_N$ and disturbance, $\mathbf{d}_1, \cdots, \mathbf{d}_N$). The corresponding measurement values and the reduced gradient values are $\mathbf{y}_1, \cdots, \mathbf{y}_N$ and $\mathbf{g}_1, \cdots, \mathbf{g}_N$, respectively. Then the globally optimal CVs are determined by adjusting $\boldsymbol{\Theta}$ to minimize the regression error $\boldsymbol{\epsilon}_i = \boldsymbol{\Theta}\hat{\mathbf{y}}_i - \mathbf{g}_i$ as follows:

$$\begin{aligned} L &= \min_{\boldsymbol{\Theta}} \ \frac{1}{2N}\sum_{i=1}^{N} (\boldsymbol{\Theta}\hat{\mathbf{y}}_i - \mathbf{g}_i)^T(\boldsymbol{\Theta}\hat{\mathbf{y}}_i - \mathbf{g}_i) \\ &= \min_{\boldsymbol{\Theta}} \ \frac{1}{2N}\sum_{j=1}^{n_u} (\mathbf{Y}\boldsymbol{\theta}_j - \hat{\mathbf{g}}_j)^T(\mathbf{Y}\boldsymbol{\theta}_j - \hat{\mathbf{g}}_j) \\ &= \min_{\boldsymbol{\Theta}} \ \frac{1}{2N}\mathrm{trace}\left((\mathbf{Y}\boldsymbol{\Theta}^T - \mathbf{G}^T)^T(\mathbf{Y}\boldsymbol{\Theta}^T - \mathbf{G}^T)\right) \end{aligned} \tag{13}$$

where $\mathbf{Y} = \begin{bmatrix} \hat{\mathbf{y}}_1 & \cdots & \hat{\mathbf{y}}_N \end{bmatrix}^T$, $\boldsymbol{\theta}_j^T$ is the j^{th} row of $\boldsymbol{\Theta}$ and $\hat{\mathbf{g}}_j^T$ is the j^{th} row of the matrix,

$$\mathbf{G} = \begin{bmatrix} \mathbf{g}_1 & \cdots & \mathbf{g}_N \end{bmatrix}$$

Assume that $N > n_u$. Then, the least squares solution to the problem (13) is

$$\boldsymbol{\Theta}^T = (\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\mathbf{G}^T \tag{14}$$

The corresponding loss is

$$L = \frac{1}{2N}\mathrm{trace}\left(\mathbf{G}\left(\mathbf{I} - \mathbf{Y}(\mathbf{Y}^T\mathbf{Y})^{-1}\mathbf{Y}^T\right)\mathbf{G}^T\right) \tag{15}$$

In principle, it is possible to parametrize the CVs in terms of all the available

measurements. Control systems with lower complexity can be obtained by using a subset of available measurements to parametrize the CVs, which often provides similar loss as the case where CVs are chosen to be functions of all the available measurements [Alstad et al., 2009; Kariwala, 2007; Kariwala et al., 2008; Ye et al., 2013]. The selection of the subset of measurements is a combinatorial optimization problem, which makes the application of this method difficult to large-scale processes. The BAB framework used to overcome this difficulty is presented in the next section.

4 Branch and bound method

4.1 General principle

Let $X_r = \{x_i | i = 1, 2, \cdots, r\}$, be an r-element set. A subset selection problem with the selection criterion ϕ involves finding the optimal solution, X_n^*, such that

$$\phi(X_n^*) = \max_{X_n \subset X_r} \phi(X_n) \tag{16}$$

For this problem, the number of alternatives is $\mathcal{C}_r^n = \frac{r!}{(r-n)!n!}$, which grows very quickly with r and n rendering an exhaustive search unviable. A BAB approach can provide globally optimal solution for the subset selection problem in (16) without exhaustive search. In this approach, the original problem (node) is divided (branched) into several non-overlapping subproblems (sub-nodes). If any of the n-element solution of a sub-problem cannot lead to the optimal solution, the sub-problem is not evaluated further (pruned), else it is branched again. The pruning of sub-problems allows the BAB approach to gain efficiency in comparison with an exhaustive search.

The available BAB methods for subset selection can be classified as downwards [Cao and Saha, 2005; Chen, 2003; Narendra and Fukunaga, 1977; Somol et al., 2000; Yu and Yuan, 1993] and upwards [Cao and Kariwala, 2008; Kariwala and Cao, 2009; Kariwala and Y.Cao, 2010] BAB methods based on the search direction. For the regression problem associated with globally optimal CV selection, the downwards BAB approach is applicable and is discussed next.

In a downwards BAB approach, each node is represented by $X_s = F_f \cup C_c$, where $s = f + c$ and, F_f and C_c denote the fixed and candidate sets, respectively. Here, the subscript denote the size of the set. The relationship between

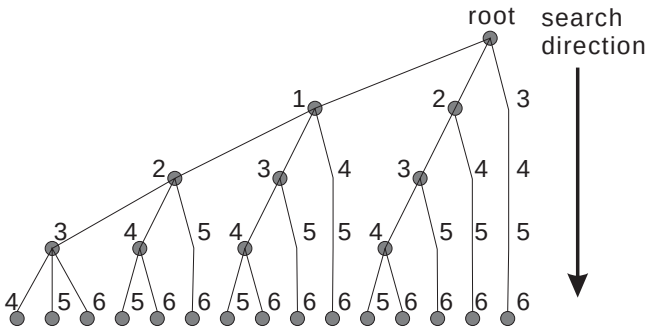

Fig. 1. Solution tree for selecting 2 out of 6 elements

the fixed and candidate sets of a node and its i^{th} sub-node (branching rule) is given as follows:

$$F_{f_i}^i = F_f \cup \{x_1, \cdots, x_{i-1}\};\ C_{c_i}^i = C_c \setminus \{x_1, \cdots, x_i\} \tag{17}$$

where $F_{f_i}^i$ and $C_{c_i}^i$ denote the fixed and candidate sets of the i^{th} sub-node and $i = 1, 2, \cdots, n-f+1$. An example of the solution tree obtained by recursively applying the branching rule in (17) is shown in Figure 1. For the root node in this solution tree, we have $F_f = \emptyset$ and $C_c = X_r$. The label of the nodes denote the element being removed from X_s. The solution tree has $\mathcal{C}_n^r$ terminal nodes (marked by grey circles), which represent different n-element subsets of X_r.

To describe the pruning principle, let $\mathcal{X}$ denote the ensemble of all n-element subsets, which can be obtained using (17), *i.e.*

$$\mathcal{X} = \{F_f \cup C_c \setminus X_{f+c-n} | X_{f+c-n} \in C_c\} \tag{18}$$

and $\underline{\phi}(F_f \cup C_c)$ be the upper bound on ϕ computed over all elements of $\mathcal{X}$, *i.e.*

$$\underline{\phi}(F_f \cup C_c) = \max_{X_n \in \mathcal{X}} \phi(X_n) \tag{19}$$

Assume that B is a lower bound of the globally optimal criterion, *i.e.* $B \leq \phi(X_n^*)$. Then,

$$\phi(X_n) < \phi(X_n^*)\, \forall X_n \in \mathcal{X}, \text{if } \underline{\phi}(F_f \cup C_c) < B \tag{20}$$

Hence, any $X_n \in \mathcal{X}$ cannot be optimal and can be pruned without further evaluation, if $\underline{\phi}(F_f \cup C_c) < B$.

Although pruning of nodes using (20) results in an efficient BAB algorithm, further efficiency can be gained by performing pruning on the sub-nodes directly. This happens as the lower bounds for different sub-nodes are related

and can be computed together from $\underline{\phi}(F_f \cup C_c)$ resulting in computational efficiency. For $x_i \in C_c$, the i^{th} sub-node can be pruned if

$$\underline{\phi}(F_f \cup C_c \setminus x_i) < B \tag{21}$$

For a BAB method involving pruning of sub-nodes, branching needs to be carried on sub-node level as well, which requires choosing a decision element to branch upon. Here, the decision element is selected as the element with largest $\underline{\phi}(F_f \cup C_c \setminus x_i)$ among all $x_i \in C_c$ (best-first search).

4.2 Application to CV Selection using Regression

According to (15), the globally average loss using a linear combination of measurements can be represented as

$$L = \frac{1}{2N}\left(\text{trace}(\mathbf{G}\mathbf{G}^T) - \text{trace}(\mathbf{B}^T\mathbf{C}^{-1}\mathbf{B})\right) \tag{22}$$

where $\mathbf{B} = (\mathbf{G}\mathbf{Y})^T$ and $\mathbf{C} = \mathbf{Y}^T\mathbf{Y}$.

As the first term in (22) is constant, measurements can be selected by minimizing the loss, L or equivalently maximizing the second term, $\text{trace}(\mathbf{B}^T\mathbf{C}^{-1}\mathbf{B})$. To select a subset of n measurements as CVs, X_n from a set of r available measurements, the criterion is:

$$\max_{X_n \subset X_r} \phi(X_n) = \text{trace}(\mathbf{B}_{X_n}^T \mathbf{C}_{X_n,X_n}^{-1} \mathbf{B}_{X_n}) \tag{23}$$

where $\mathbf{B}_{X_n}$ is a sub-matrix of $\mathbf{B}$ with row indices defined by X_n and $\mathbf{C}_{X_n,X_n}$ represents the principal submatrix of $\mathbf{C}$ with rows and columns indexed by X_n.

The use of BAB for solving the optimization problem in (23) requires an upper bound on the selection criteria, calculated over the ensemble $\mathcal{X}$ in (18). This upper bound is derived in the next proposition.

Proposition 1 (Pruning) *Consider a node with fixed set F_f and candidate set C_c. For $\mathcal{X}$ in* (18)*,*

$$\phi(F_f \cup C_c) \geq \max_{X_n \in \mathcal{X}} \phi(X_n) \tag{24}$$

Proof: For notational simplicity, let us define $S = F_f \cup C_c$. Let $\mathbf{R}$ and $\tilde{\mathbf{R}}$ be the Cholesky factors of $\mathbf{C}_{X_n,X_n}$ for some $X_n \in \mathcal{X}$ and $\mathbf{C}_{S,S}$, respectively, where $\mathcal{X}$ is given in (18). As $X_n \subset S$, $\mathbf{R}$ is a principal submatrix of $\tilde{\mathbf{R}}$, which implies that $\mathbf{R}^{-1}$ is a principal submatrix of $\tilde{\mathbf{R}}^{-1}$. Define $\mathbf{Z} = \mathbf{R}^{-T}\mathbf{B}_{X_n}$. Then,

$$\phi(X_n) = \text{trace}(\mathbf{B}_{X_n}^T \mathbf{R}^{-1}\mathbf{R}^{-T}\mathbf{B}_{X_n}) = \text{trace}(\mathbf{Z}^T\mathbf{Z}) \tag{25}$$

Similarly,

$$\phi(S) = \text{trace}(\mathbf{B}_S^T \tilde{\mathbf{R}}^{-1}\tilde{\mathbf{R}}^{-T}\mathbf{B}_S) = \text{trace}(\tilde{\mathbf{Z}}^T\tilde{\mathbf{Z}}) \tag{26}$$

where $\tilde{\mathbf{Z}} = \tilde{\mathbf{R}}^{-T}\mathbf{B}_S$. Since $\tilde{\mathbf{Z}}$ is a superset of $\mathbf{Z}$, $\phi(S) \geq \phi(X_n)$ and (24) follows. ■

Proposition 1 implies that the non-optimal nodes can be pruned using $\phi(F_f \cup C_c)$ as the upper bound. To gain further efficiency by pruning the sub-nodes directly, we relate the selection criteria of a node with its sub-nodes in the next proposition.

Proposition 2 (Subset pruning) *Consider a node with fixed set F_f and candidate set C_c. Let $S = F_f \cup C_c$. For $x_i \in C_c$, $i = 1, 2, \cdots, c$,*

$$\phi(S \setminus x_i) = \phi(S) - \alpha_i^2/\delta_i \tag{27}$$

where

$$\alpha_i = \sum_{j=1}^{n_u} \mathbf{z}_i^T \mathbf{B}_{S,j} \tag{28}$$

with $\mathbf{z}_i^T$ and δ_i being the ith row and (i,i)th element of $\mathbf{C}_{S,S}^{-1}$ respectively, whilst $\mathbf{B}_{S,j}$ is the j^{th} column of $\mathbf{B}_S$.

Proof: To simplify notation without losing generality, we permute rows and columns of $\mathbf{C}_{S,S}^{-1}$ such that i^{th} row and column are the last row and column. With this re-ordering, $\mathbf{C}_{S,S}^{-1}$ can be partitioned as

$$\mathbf{C}_{S,S}^{-1} = \begin{bmatrix} \mathbf{M} & \boldsymbol{\eta}_i \\ \boldsymbol{\eta}_i^T & \delta_i \end{bmatrix}$$

where $\mathbf{z}_i^T = \begin{bmatrix} \boldsymbol{\eta}_i^T & \delta_i \end{bmatrix}$. Then,

$$\mathbf{C}_{S\setminus x_i, S\setminus x_i}^{-1} = \mathbf{M} - \boldsymbol{\eta}_i\boldsymbol{\eta}_i^T/\delta_i$$

Therefore,

$$\phi(S) = \text{trace}(\mathbf{B}_{S\setminus x_i}^T \mathbf{M} \mathbf{B}_{S\setminus x_i} + 2\mathbf{B}_{x_i}^T \boldsymbol{\eta}_i^T \mathbf{B}_{S\setminus x_i} + \mathbf{B}_{x_i}^T \mathbf{B}_{x_i} \delta_i)$$

This gives

$$\begin{aligned} &\phi(S) - \phi(S \setminus x_i) \\ =&\text{trace}(\mathbf{B}_{S\setminus x_i}^T \boldsymbol{\eta}_i \boldsymbol{\eta}_i^T \mathbf{B}_{S\setminus x_i}/\delta_i + 2\mathbf{B}_{x_i}^T \boldsymbol{\eta}_i^T \mathbf{B}_{S\setminus x_i} + \mathbf{B}_{x_i}^T \mathbf{B}_{x_i} \delta_i) \end{aligned}$$

Define $\mathbf{Q} = \boldsymbol{\eta}_i^T \mathbf{B}_{S\setminus x_i} + \mathbf{B}_{x_i}\delta_i = \mathbf{z}_i \mathbf{B}_S$. Then,

$$\phi(S) - \phi(S \setminus x_i) = \text{trace}(\mathbf{Q}^T\mathbf{Q})/\delta_i = \alpha_i^2/\delta_i$$

which leads to (27). ■

The evaluation of (27) requires inversion of only one matrix $\mathbf{C}_{S,S}$, which is the same for all $x_i \in C_c$. Thus, the use of (27) to obtain the selection criteria for all sub-nodes together is computationally more efficient than directly evaluating the selection criteria for every node. In summary, the following BAB algorithm can be used for subset selection for regression.

Algorithm 1 (BAB algorithm 1 [Cao et al., 2013]) *Initialize $f = 0$, $F_f = \emptyset$, $C_c = X_r$, $\phi(F_f) = 0$ and $B = 0$. Call the following recursive algorithm:*

(1) If $\phi(F_f \cup C_c) > B$, prune the current node and return, else perform Steps 2-4.
(2) Calculate α_i in (28) $\forall i \in C_c$. Prune the subsets with $\phi(F_f \cup C_c) - \alpha_i^2/\delta_i < B$.
(3) If $f = n$ or $f + c = n$, go to next step. Otherwise, generate the c sub-nodes according to the branching rule in (17) and call the recursive algorithm in Step 1 for each sub-node. Return to the caller after the execution of the loop finishes.
(4) Find $J_{\max} = \phi(F_f \cup C_c) - \max_{i\in C_c} \alpha_i^2/\delta_i$. If $J_{\max} > B$, update $B = J_{\max}$. Return to the caller.

4.3 Improved Pruning Algorithm

The main aim of measurement selection is to simplify the control structure, *i.e.* to select a measurement subset with size as small as possible. In this case, the upper bound based on monotonicity in Proposition 1 might be very loose if node size, $s = f + c \gg n$. To improve the efficiency, a tighter upper bound is derived by taking the multiple variables to be discarded into account.

Consider a node with a fixed set, F_f and a candidate set, C_c. Define $X_s = F_f \bigcup C_c$ with $s = f + c$, and $X_w = X_s \setminus X_n$ with $w = s - n$. According to

matrix inversion theorem, if C_{X_s,X_s} and $C^{-1}_{X_s,X_s}$ are partitioned according to a selection, X_n as follows.

$$\mathbf{C}_{X_s,X_s} = \begin{bmatrix} \mathbf{C}_{X_n,X_n} & \mathbf{C}_{X_n,X_w} \\ \mathbf{C}_{X_w,X_n} & \mathbf{C}_{X_w,X_w} \end{bmatrix}$$

$$\mathbf{C}^{-1}_{X_s,X_s} = \begin{bmatrix} \mathbf{Q}_{X_n,X_n} & \mathbf{Q}_{X_n,X_w} \\ \mathbf{Q}_{X_w,X_n} & \mathbf{Q}_{X_w,X_w} \end{bmatrix}$$

then

$$\mathbf{C}^{-1}_{X_s,X_s} - \begin{bmatrix} \mathbf{C}^{-1}_{X_n,X_n} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{Q}_{X_n,X_w} \\ \mathbf{Q}_{X_w,X_w} \end{bmatrix} \mathbf{Q}^{-1}_{X_w,X_w} \begin{bmatrix} \mathbf{Q}_{X_n,X_w} & \mathbf{Q}_{X_w,X_w} \end{bmatrix}$$

Therefore,

$$\phi(X_s) - \phi(X_n) = \text{trace}(\tilde{\mathbf{B}}^T_{X_w} \mathbf{Q}^{-1}_{X_w,X_w} \tilde{\mathbf{B}}_{X_w})$$

where $\tilde{\mathbf{B}}_{X_w} = \begin{bmatrix} \mathbf{Q}_{X_n,X_w} & \mathbf{Q}_{X_w,X_w} \end{bmatrix} \mathbf{B}_{X_s}$. Then an upper bound can be derived as follows.

$$\begin{aligned} \phi(X_n) &\leq \phi(X_s) - \text{trace}(\tilde{\mathbf{B}}_{X_w} \tilde{\mathbf{B}}^T_{X_w}) \underline{\lambda}(\mathbf{Q}^{-1}_{X_w,X_w}) \\ &\leq \phi(X_s) - \min_{X_w \in X_s} \|\tilde{\mathbf{B}}_{X_w}\|^2_F \overline{\lambda}^{-1}(\mathbf{Q}_{X_w,X_w}) \\ &\leq \phi(X_s) - \min_{X_w \in X_s} \|\tilde{\mathbf{B}}_{X_w}\|^2_F \overline{\lambda}^{-1}(\mathbf{C}^{-1}_{X_s,X_s}) \\ &\leq \phi(X_s) - \min_{X_w \in X_s} \|\tilde{\mathbf{B}}_{X_w}\|^2_F \underline{\lambda}(\mathbf{C}_{X_s,X_s}) \\ &\leq \phi(X_s) - \min_{X_w \in X_s} \|\tilde{\mathbf{B}}_{X_w}\|^2_F \underline{\lambda}(\mathbf{C}) \end{aligned}$$

In the above upper bound, the minimum eigenvalue of $\mathbf{C}$, $\underline{\lambda}(\mathbf{C})$ can be pre-calculated. Therefore, an improved downward pruning procedure can be as follows.

Algorithm 2 (BAB Algorithm 2 [Cao et al., 2013]) *At each node,*

(1) Evaluate $\tilde{\mathbf{B}}_{X_s} = \mathbf{C}^{-1}_{X_s,X_s} \mathbf{B}_{X_s}$.
(2) For each $x_i \in C_c$, *calculate* $\alpha_{x_i} = \mathbf{b}^T_{x_i} \mathbf{b}_{x_i}$, *where* $\mathbf{b}^T_{x_i}$ *is a row of* $\tilde{\mathbf{B}}_{X_s}$ *corresponding to* x_i.
(3) Rank α_{x_i} *such that,* $\alpha_1 \leq \alpha_2 \cdots \leq \alpha_w$ *and sum the lowest* w *values as* $\beta = \sum_{k=1}^{w} \alpha_k$.
(4) If $\phi(X_s) - \beta \underline{\lambda}(\mathbf{C}) < B$, *then this node can be pruned.*

The above algorithm can further be applied to sub-node pruning if the pruning condition for the current node is not satisfied. If $x_i \in X_w$ is to be fixed, then, the corresponding α_{x_i} cannot be included in β. The $(w+1)^{\text{th}}$ lowest α_{w+1} is going to replace α_{x_i}. This will increase β by $\delta_i = \alpha_{w+1} - \alpha_{x_i}$. Therefore, if $\phi(X_s) - (\beta + \delta_i)\underline{\lambda}(\mathbf{C}) < B$ then x_i cannot be fixed. In other words, x_i must be discarded.

Furthermore, if $x_i \in C_c \backslash X_w$ is discarded, then β will increase by $\gamma_i = \alpha_{x_i} - \alpha_w$. Therefore, if $\phi(X_s) - (\beta + \gamma_i)\underline{\lambda}(\mathbf{C}) < B$, then x_i cannot be discarded, hence has to be fixed.

5 Numerical Examples

To evaluate the efficiency of the proposed BAB algorithm for selecting globally optimal CVs, we test the performance of BAB algorithm on random matrices and a binary distillation column [Skogestad, 1997]. All tests are conducted on a PC running Windows 7 SP1 with Intel Core i3-2100 3.10GHz processor, 8GB RAM using Matlab R2011a.

5.1 Random Matrices

Two full matrices are randomly generated for tests: $\mathbf{Y} \in \mathbb{R}^{N \times n_y}$ containing n_y candidate measurements with N samples for regression and $\mathbf{G} \in \mathbb{R}^{N \times n_u}$ containing N samples of regression objective for a n_u degrees of freedom problem. Elements of these matrices are normally distributed with zero mean and unit variance. Since the computation difficulty arises mainly due to the increasing n_y and n, the following two tests are conducted: In the first test, we let n_y vary from 10 to 100 while keeping $n = n_y - 5$. In the second test, we fix $n_y = 40$ and let n vary from 1 to 39. In both tests, average computation times and average numbers of nodes evaluated are computed for 100 experiments with $N = 1000$ and $n_u = 2$.

The results for the first test are summarized in Figure 2, BAB algorithms 1 and 2 show superior performance over brute force method. As shown in the figure, the required computation time and number of nodes to be evaluated for brute force algorithm increase dramatically as n_y increases. The brute force algorithm cannot solve the problem in reasonable time when n_y exceeds 35. Note that to simplify the problem, we have restricted $n = n_y - 5$. Nevertheless, using the proposed BAB algorithm, the problems can be efficiently solved. It takes about 20 s to complete the task when $n_y = 100$ for Algorithm 1 and only 2 s for Algorithm 2. By applying the improved pruning algorithm, the

computation time and number of nodes evaluated for Algorithm 2 grow only slightly as n_y increases.

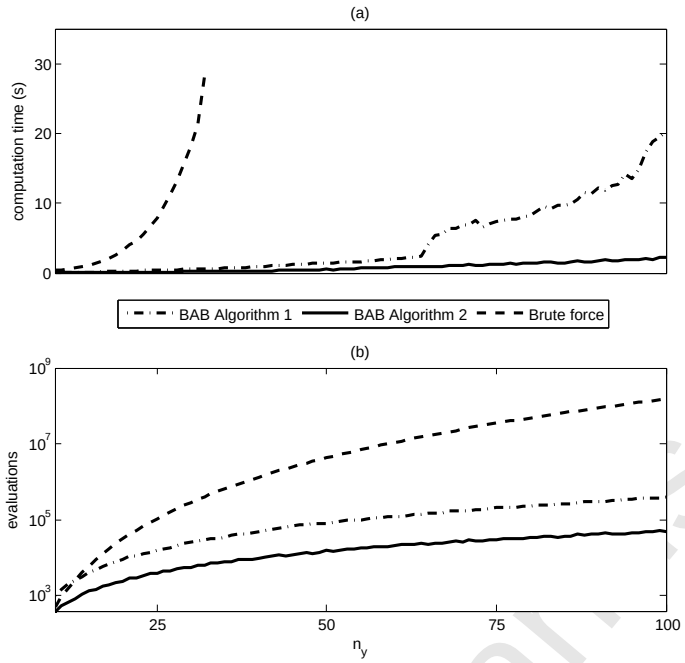

Fig. 2. Random test 1: (a) computation time and (b) number of nodes evaluated when n_y varies while keeping $n = n_y - 5$

The results for Test 2 are shown in Figure 3, which shows these problems can also be effectively solved by applying BAB algorithms 1 and 2 when n varies from 1 to 39, whereas a brute and force method cannot solve the problem when $6 < n < 35$. It takes at most 139 s for Algorithm 1 and, 40 s for Algorithm 2 to solve any of these problems. It can also be observed that computation times and number of nodes evaluated reach the peak when n is around 14 and not 20, which results in the largest number of combinations. This happens as downwards BAB is more efficient for problems where a few among many candidate variables need to be discarded [Cao and Kariwala, 2008].

5.2 Binary Distillation Column

Next, we consider self-optimizing control of a binary distillation column with 41 trays [Jäschke, 2011]. The distillation column has 4 manipulated variables: reflux flow rate (L), vapor boilup V, distillate flow rate (D) and bottoms flow rate (B). The levels of top condenser and bottom reboiler need to be stabilized, which consumes two degrees of freedom. The top composition of the product is required to be actively controlled at 0.99. D, B and L are selected to control the two levels and the product composition, respectively. Therefore, only one degree of freedom V remains for optimization purpose. The objective is to

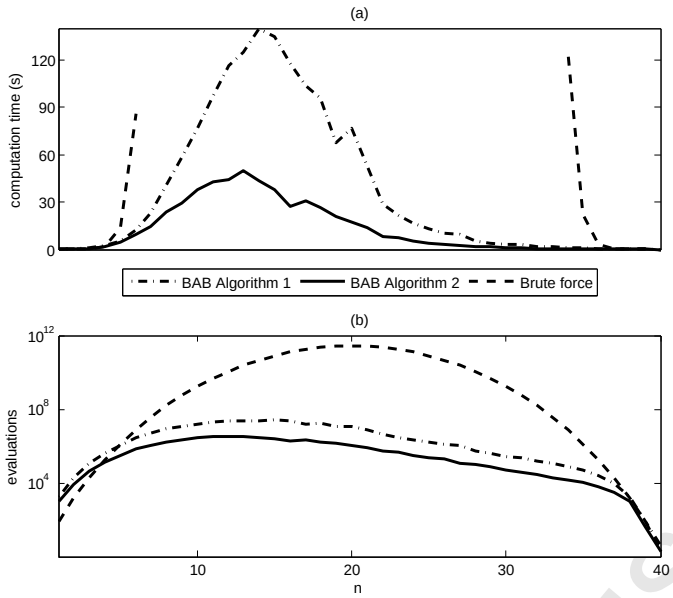

Fig. 3. Random test 2: (a) computation time and (b) number of nodes evaluated when n varies

minimize the cost function which can be equivalently formulated as

$$J = V - 64D \tag{29}$$

where the first term V indicates the energy cost and the second term of the cost function is the negative profit for selling the product with an assumed price 64 currency unit. The reader is referred to [Jäschke, 2011] for more detailed description of the column.

The main disturbances are feed flow rate (F), feed composition (z_F) and vapor fraction of feed (q_F), which are here defined to vary between $[1, 1.5]$, $[0.5, 0.75]$ and $[1, 1.5]$, respectively. Besides the two levels and the top composition, one additional CV needs to be identified for self-optimizing control of the column. The combination of temperatures on 41 trays ($y_1, \ldots, y_{41}$, counting from bottom to top), which are measured with an accuracy of ± 0.5°C, are considered as CV for implementation of self-optimizing control strategy.

Data samples for NCO regression are generated as follows: each independent variable is sampled with 6 points within appropriate ranges. The variation range for disturbances are defined earlier, whilst variation range for vapor boilup V is bounded within $(3.4, 5.2)$. For each scenario, temperatures at each tray are calculated and the NCO component J_V, which refer to the gradient of J with respect to V, is also obtained using finite difference method. Therefore, $6^4 = 1296$ samples are collected for regression. Because the number of candidate measurements for regression is large, we apply BAB algorithm 2 to implement globally optimal CV selection and the results are summarized in

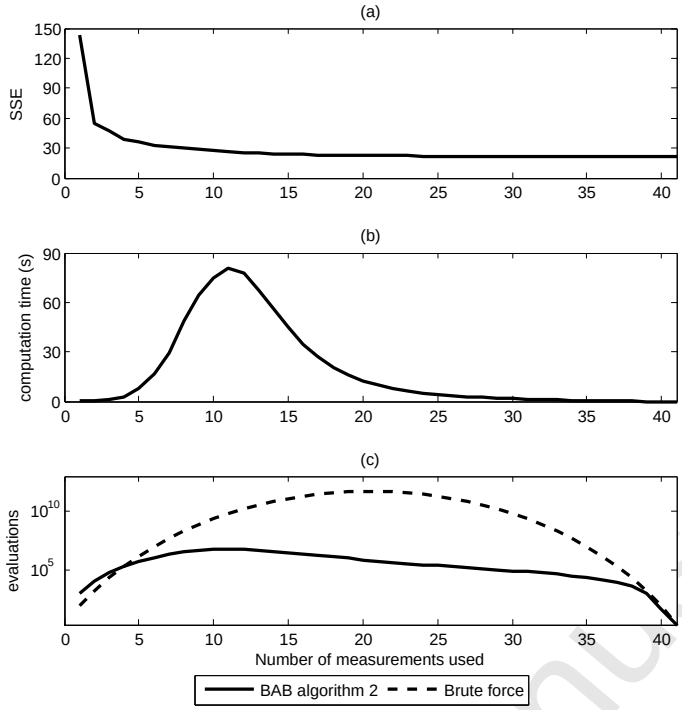

Fig. 4. BAB performance for the column case study: (a) regression SSE; (b) computation time; (c) Evaluations

Table 1
Average losses for global and local approaches with best subset measurements

n	Global approach		Local approach	
	Best subset	Average loss	Best subset	Average loss
1	$[T_{12}]$	0.0232	$[T_9]$	0.0338
2	$[T_{14}, T_{23}]$	0.0018	$[T_{12}, T_{21}]$	0.0056
3	$[T_{13}, T_{21}, T_{29}]$	0.0015	$[T_{11}, T_{12}, T_{21}]$	0.0048
4	$[T_{12}, T_{13}, T_{21}, T_{31}]$	0.000599	$[T_{10}, T_{11}, T_{12}, T_{21}]$	0.0043
5	$[T_{12}, T_{13}, T_{21}, T_{22}, T_{36}]$	0.000606	$[T_{10}, T_{11}, T_{12}, T_{13}, T_{21}]$	0.0047
6	$[T_{12}, T_{13}, T_{15}, T_{21}, T_{22}, T_{36}]$	0.000115	$[T_{10}, T_{11}, T_{12}, T_{13}, T_{21}, T_{22}]$	0.0047

Figure 4.

Figure 4 (a) shows that when $n > 6$, the SSE can only be slightly reduced as n increases. Therefore, using full set of measurements as predictors for the gradient J_V is not necessary and a trade-off between economics and the number of measurements can be made. Figure 4 (b) and (c) show the computation time and number of node evaluations and demonstrate the usefulness and effectiveness of proposed BAB algorithm. Brute force cannot handle such a large problem, whereas proposed BAB algorithm solves it successfully. It takes

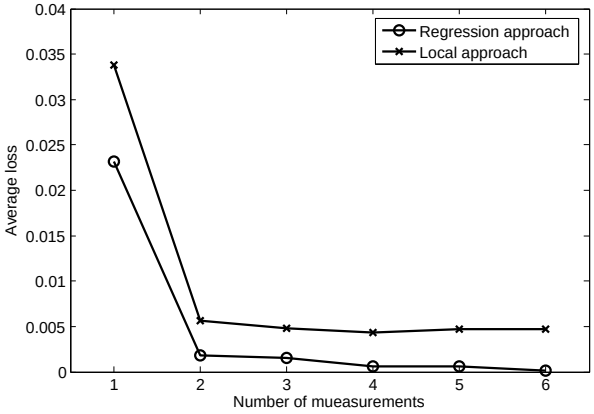

Fig. 5. Trends of average loss through nonlinear model validation against the number of measurements with n varying from 1 to 6

about 731 s to complete all the selection tasks, which is acceptable for off-line computing. Largest computation time occurs at $n = 11$, which takes about 80 s. Overall, Figure 4 can be used as a guide for selecting appropriate number of measurements to be used for self-optimizing control. Therefore, proposed algorithm is practically appealing, because the algorithm makes it possible to reduce the overall operation cost and meanwhile, reduces the investment for hardware sensors, (*e.g.* temperature sensors for the column).

To compare the actual self-optimizing performances, the CVs with best measurement subsets for n varying from 1 to 6 are tested through nonlinear model validations. A Monte Carlo experiment with 100 sets of randomly generated disturbances within their allowable ranges is carried out. The local approach with average loss criterion [Kariwala et al., 2008] is also applied, where the locally optimal subset is found using PB^3 algorithm [Cao and Kariwala, 2009]. The results of average losses are summarized in Table 1 and the trends of average loss against the number of measurements are shown in Figure 5. It can be seen that the global approach performs much better than local approach. For local approach, there is no significant improvement of economic loss by increasing n when $n > 3$. For global approach, even with only 2 measurements, the performance is still much better than local approach with 6-subset measurements. In summary, with the use of proposed BAB algorithm, the best subset measurements can be identified to find CVs with global self-optimizing performances efficiently.

6 Conclusions

In the context of self-optimizing control, a novel branch and bound (BAB) algorithm is proposed for efficiently selecting globally optimal controlled variables (CVs) as a function of the subset of available measurements. The BAB

algorithm identifies the best measurement subset such that the sums of squares of errors between the selected CVs and necessary conditions of optimality is minimized. Numerical tests using randomly generated matrices and a practical binary distillation column case study show the efficiency and effectiveness of the proposed algorithm. For distillation column case, the self-optimizing performance of regression approach is also compared with local SOC approach where the PB^3 algorithm is used for best subset selection. Results show that using global approach combined with proposed BAB algorithm, not only is the loss reduced, the required number of measurements to achieve self-optimizing properties is also decreased, which makes the control system simpler with reduced hardware investment.

It is also pointed out that the proposed algorithm is applicable to the general linear regression problem as well as other statistical problems. The proposed downwards algorithm is most efficient for problems where a few from many candidate variables need to be discarded. In this paper, an improved algorithm is presented for problems where a few variables need to be selected from many candidate variables. To further reduce the incurred computational expense, the use of upwards and bidirectional BAB algorithms is promising and will be explored in future.

References

Alstad, V., Skogestad, S., Hori, E. S., 2009. Optimal measurement combinations as controlled variables. J. Proc. Control 19 (1), 138–148.

Cao, Y., Kariwala, V., 2008. Bidirectional branch and bound for controlled variable selection: Part I. Principles and minimum singular value criterion. Comput. Chem. Engng. 32 (10), 2306–2319.

Cao, Y., Kariwala, V., November 2009. Bidirectional Branch and Bound for Average Loss Minimization. MATLAB File Exchange, available at `http://www.mathworks.com/matlabcentral/fileexchange/25870`.

Cao, Y., Saha, P., 2005. Improved branch and bound method for control structure screening. Chem. Engg. Sci. 60 (6), 1555–1564.

Cao, Y., Ye, L., Kariwala, V., February 2013. Improvd downward branch and bound algorithm for regression variable selection. MATLAB File Exchange, available at `http://www.mathworks.co.uk/matlabcentral/fileexchange/40357`.

Chen, X.-W., 2003. An improved branch and bound algorithm for feature selection. Pattern Recognition Letters 24, 1925?1933.

Halvorsen, I. J., Skogestad, S., Morud, J. C., Alstad, V., 2003. Optimal selection of controlled variables. Ind. Eng. Chem. Res. 42 (14), 3273–3284.

Heldt, S., 2010. Dealing with structural constraints in self-optimizing control engineering. Journal of Process Control 20 (9), 1049–1058.

Hu, W., Umar, L. M., Xiao, G., Kariwala, V., 2012. Local self-optimizing control of constrained processes. J. Proc. Contr. 22, 488–493.

Jäschke, J., 2011. Invariants for optimal operation of process systems. Ph.D. thesis, Ph. D. thesis, Norwegian University of Science and Technology.

Kariwala, V., 2007. Optimal measurement combination for local self-optimizing control. Ind. Eng. Chem. Res. 46 (11), 3629–3634.

Kariwala, V., Cao, Y., 2009. Bidirectional branch and bound for controlled variable selection: Part II. Exact local method for self-optimizing control. Comput. Chem. Eng. 33 (8), 1402–1412.

Kariwala, V., Cao, Y., Janardhanan, S., 2008. Local self-optimizing control with average loss minimization. Ind. Eng. Chem. Res. 47 (4), 1150–1158.

Kariwala, V., Y.Cao, 2010. Bidirectional branch and bound for controlled variable selection: Part III. Local average loss minimization. IEEE Trans. Ind. Informat. 6 (1), 54–61.

Narendra, P., Fukunaga, K., 1977. A branch and bound algorithm for feature subset selection. IEEE Trans. Comput. C-26, 917–922.

Rangaiah, G. P., Kariwala, V., 2012. Plantwide Control: Recent Developments and Applications. John Wiley & Sons, Chichester, UK.

Skogestad, S., 1997. Dynamics and control of distillation columns - A tutorial introduction. Trans. IChemE Part A 75, 539–562.

Skogestad, S., 2000. Plantwide control: The search for the self-optimizing control structure. J. Proc. Control 10 (5), 487–507.

Skogestad, S., Postlethwaite, I., 2005. Multivariable Feedback Control: Analysis and Design, 2nd Edition. John Wiley & Sons, Chichester, UK.

Somol, P., Pudil, P., Ferri, F., Kittler, J., 2000. Fast branch and bound algorithm in feature selection. In: Sanchez, B., Pineda, J., Wolfmann, J., Bellahsense, Z., Ferri, F. (Eds.), Proceedings of World Multiconference on Systemics, Cybernetics and Informatics. Vol. VII. Orlando, Florida, USA, pp. 1646–651.

Van De Wal, M., De Jager, B., 2001. A review of methods for input/output selection. Automatica 37 (4), 487–510.

Ye, L., Cao, Y., Li, Y., Song, Z., 2012. A data-driven approach for selecting controlled variables. In: Proceedings of 8th Intl. Symposium on ADCHEM. Singapore.

Ye, L., Cao, Y., Li, Y., Song, Z., 2013. Approximating necessary conditions of optimality as controlled variables. Ind. Eng. Chem. Res. 52 (2), 798–808.

Yu, B., Yuan, B., 1993. A more efficient branch and bound algorithm for feature selection. Pattern Recognition 26, 883–889.

2013-03-27

Branch and bound method for regression-based controlled variable selection

Kariwala, Vinay

Elsevier

Kariwala V, Ye L, Cao Y. (2013) Branch and bound method for regression-based controlled variable selection. Computers & Chemical Engineering, Volume 54, July 2013, pp. 1-7
http://dx.doi.org/10.1016/j.compchemeng.2013.03.006