Branch and bound method for multiobjective control structure design

Cao, Yi; Kariwala, Vinay

2008

Kariwala, V., & Cao, Y. (2008). Branch and bound method for multiobjective control structure design. Proceedings of the 4th IEEE Conference on Industrial Electronics and Applications, (pp. 2513-2518)

<https://hdl.handle.net/10356/91099>

<https://doi.org/10.1109/ICIEA.2009.5138660>

© 2008 IEEE. Personal use of this material is permitted. However, permission to reprint/republish this material for advertising or promotional purposes or for creating new collective works for resale or redistribution to servers or lists, or to reuse any copyrighted component of this work in other works must be obtained from the IEEE. This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder. <http://www.ieee.org/portal/site> This material is presented to ensure timely dissemination of scholarly and technical work. Copyright and all rights therein are retained by authors or by other copyright holders. All persons copying this information are expected to adhere to the terms and constraints invoked by each author's copyright. In most cases, these works may not be reposted without the explicit permission of the copyright holder.

Branch and Bound Method for Multiobjective Control Structure Design

Vinay Kariwala

Division of Chemical & Biomolecular Engineering
Nanyang Technological University
Singapore 637459
vinay@ntu.edu.sg

Yi Cao

School of Engineering
Cranfield University
Cranfield, Bedford MK43 0AL, UK
y.cao@cranfield.ac.uk

Abstract—Control structure design deals with the selection of controlled and manipulated variables, and the pairings interconnecting these variables. The available criteria for these tasks require enumeration of every alternative, which can be computationally forbidding for large-scale process. Owing to the computational complexity, variables and pairings are often selected sequentially, which may result in sub-optimal control structures. In this paper, an efficient branch and bound (BAB) method is proposed to select the variables and pairings together in a multiobjective optimization framework. As an illustration of the proposed multiobjective BAB framework, the minimum singular value rule and the μ -interaction measure are used as the criteria for selection of controlled variables and pairings, respectively. Numerical tests using randomly generated matrices and large-scale case study of HDA process show that the BAB method is able to reduce the solution time by several orders of magnitude in comparison with exhaustive search.

Index Terms—Combinatorial optimization, Control system design, Decentralized control, Interaction measures, Large-scale systems, Multiobjective optimization, Minimum singular value, Structured singular value.

I. INTRODUCTION

Control structure design (CSD) deals with the selection of controlled and manipulated variables (CVs and MVs), and the pairings interconnecting these variables [1]. Simply stating, the objective of CSD is to decide upon: *Given the process, where shall the controllers be placed?* This choice is often not obvious for processes encountered in practice. This problem is further complicated by the mass and energy integration among the different process units, which necessitates consideration of the whole plant together. CSD for complete chemical plants is also known as *plantwide control*.

With its practical implications, CSD has been studied by several researchers [1], [2], [3], [4]. A review of many of the available techniques is provided by Larsson and Skogestad [5]. While some of the available approaches involve extensive use of process knowledge and heuristics [4], [6], some model-based systematic methods have also been recently proposed [1]. In model-based systematic methods, the selection of variables and pairings is carried out sequentially, which often requires enumeration of every alternative. The rapid growth of alternatives with process dimensions makes CSD through an exhaustive search computationally forbidding for large-scale processes. Furthermore, the selection of CVs and

MVs followed by pairing selection may result in sub-optimal control structures. For example, the selected CVs and MVs may lead to highly interacting control loops for all possible pairings rendering decentralized control difficult. This necessitates the consideration of different tasks of CSD together in a multiobjective optimization framework such that a set of promising solutions (Pareto-optimal set) can be found. Then, the practicing engineer can select the control structure from the Pareto-optimal set by trading-off different selection criteria.

The CSD problem is a combinatorial optimization problem, where the binary decision variables are related to the selection of variables and pairings. Here, a variable or pairing can be considered to selected, if the decision variable is one and *vice versa*. Traditionally, multiobjective CSD problem has been solved by converting the multiobjective problem into an optimization problem with a single objective by weighing different objectives [7] or by converting all but one of the objectives to constraints (ϵ -constraint method) [8]. In these approaches, the binary decision variables are relaxed as continuous variables. Subsequently, the Pareto-optimal set is obtained by repeatedly solving the mixed integer linear or nonlinear program (MILP or MINLP) with different weights or constraint limits. A drawback of these approaches is that the choice of weights and constraint limits is non-trivial. Furthermore, the Pareto-optimal set obtained using weighted objective function approach is not necessarily complete [9]. Evolutionary algorithms [9] can directly handle the multiobjective nature of CSD problem, but do not guarantee global optimality of the solution.

Recently, efficient branch and bound (BAB) methods have been developed for selection of CVs [10], [11], MVs [12], and pairings [13] by posing them as subset selection and permutation problems, respectively. These BAB methods guarantee globally optimal solution, while requiring several orders of magnitude lower computational times in comparison with exhaustive search. These methods, however, still need to be applied sequentially (CV and MV selection followed by pairing selection). Motivated by this drawback, we propose a BAB method to directly solve the multiobjective CSD problem in this paper.

The proposed BAB framework is general and can handle most of the available criteria for the selection of CVs, MVs and pairings. For illustration purposes, a biobjective CSD problem

is considered in this work, where the minimum singular value (MSV) rule [14] and μ -interaction measure (μ -IM) [15] are used for selection of CVs and pairings, respectively. The computational efficiency of the proposed method is demonstrated using randomly generated matrices and the large-scale case study of HDA process [16]. These numerical tests show that the BAB method is able to reduce the solution time by several orders of magnitude in comparison with exhaustive search.

The rest of this paper is organized as follows: the notation used in this paper is standardized in Section II. The general multiobjective BAB framework for CV and pairing selection is proposed in Section III. In Section IV, the BAB approach is used to solve the biobjective CSD problem with MSV rule and μ -IM as the criteria for selection of CVs and pairings, respectively. The efficiency of the BAB approaches is demonstrated in Section V through some numerical tests and the work is concluded in Section VI.

II. PRELIMINARIES

In this paper, $\{a, b\}$ denotes an unordered set, while (a, b) denotes an ordered set, both consisting of the elements a and b . Note that $\{a, b\} = \{b, a\}$, but $(a, b) \neq (b, a)$. We define N_n as the set of first n natural numbers, i.e. $N_n = \{1, 2, \dots, n\}$, where the subscript n is explicitly used to denote the size of the set. For an m -element set, X_m with $m > n$, $\mathbb{P}_n(X_m)$ represents the ensemble of all possible permutations of n -element subsets of X_m . For example, for $m = 3$ and $n = 2$, the 2-element subsets of $N_3 = \{1, 2, 3\}$ are $\{1, 2\}$, $\{1, 3\}$ and $\{2, 3\}$. Thus, $\mathbb{P}_2(N_3) = \{(1, 2), (2, 1), (1, 3), (3, 1), (2, 3), (3, 2)\}$.

We assume that the set of MVs or inputs is given; see Remark 1. Then, for a process with m measurements or outputs $y = \{y_1, y_2, \dots, y_m\}$ and n inputs $u = \{u_1, u_2, \dots, u_n\}$, $m > n$, the CSD problem consists of selecting n CVs among the available m outputs and their subsequent pairings with inputs. For $P_n = (p_1, \dots, p_n) \in \mathbb{P}_n(N_m)$, it is considered that $\{y_{p_1}, y_{p_2}, \dots, y_{p_n}\}$ are selected as CVs and $\{y_{p_1} - u_1, y_{p_2} - u_2, \dots, y_{p_n} - u_n\}$ as the pairings. For example, for $m = 3$ and $n = 2$, if $P_2 = (3, 2)$, the selected CVs are $\{y_3, y_2\}$ and the selected pairings are $\{y_3 - u_1, y_2 - u_2\}$.

$G(s)$ denotes the stable transfer function matrix relating the inputs and outputs of the process. The transfer function matrix $G(s)$ evaluated at the frequency ω is represented as $G(j\omega) \in \mathbb{C}^{m \times n}$ and the steady-state gain matrix as $G \in \mathbb{R}^{m \times n}$. For $P_n \in \mathbb{P}_n(N_m)$, G_{P_n} denotes the permuted $n \times n$ submatrix of G with rows indexed by P_n . Then, the selected CVs correspond to the row indices of G_{P_n} , while the pairings are selected on the diagonal elements of G_{P_n} .

With these preliminaries, a multiobjective CSD problem involves solving the following optimization problem

$$\begin{aligned} \min_{P_n \in \mathbb{P}_n(N_m)} \quad & \{J_1(P_n), J_2(P_n), \dots, J_q(P_n)\} \\ \text{s.t.} \quad & L(P_n) \leq 0 \end{aligned} \quad (1)$$

A solution $P_n \in \mathbb{P}_n(N_m)$ is considered feasible if it satisfies constraint (2). A feasible solution P_n^j is said to be dominated by another feasible solution P_n^i if

$$J_k(P_n^i) \leq J_k(P_n^j) \quad \forall k \in N_q \quad (3)$$

with strict inequality occurring for at least one $k \in N_q$. All non-dominated feasible solutions construct the Pareto-optimal set $\mathcal{P}$, where every pair of $P_n^i, P_n^j \in \mathcal{P}$ satisfy

$$\exists s, t \in N_q : J_s(P_n^i) < J_s(P_n^j), \quad J_t(P_n^i) > J_t(P_n^j) \quad (4)$$

Remark 1 (MV selection): CSD problems with more MVs than outputs require selection of a subset of available MVs and their subsequent pairings with outputs. Such problems can be handled in the same framework by interchanging the roles of MVs and CVs or in other words, using $G^T(s)$ as the transfer function matrix between inputs and outputs.

III. BRANCH AND BOUND METHOD FOR MULTIOBJECTIVE CONTROL STRUCTURE DESIGN

In this section, we present the general BAB method for multiobjective optimization problems involving subset selection (CV selection) together with permutation (pairing selection).

Principle. For the multiobjective CSD problem in (1)-(2), BAB *branches* the problem S into several non-overlapping sub-problems (partially selected CVs and assigned pairings). The sub-problem S_i is *pruned*, if a set of lower bounds on selection criteria over all complete solutions of S_i is dominated by a member of the current Pareto-optimal set $\mathcal{P}$, otherwise S_i is branched further. Here, a complete solution refers to $P_n \in \mathbb{P}_n(N_m)$, i.e. pairings assigned for n CVs selected from m candidate outputs. Whenever a complete control structure is reached, which is also Pareto-optimal, $\mathcal{P}$ is *updated*. Next, we provide further details of the three steps, namely branching, pruning and updating.

Branching. The implementation of BAB schemes requires a solution tree containing all possible alternatives. For combined CV and pairing selection problem, the solution tree for $m = 4$ and $n = 3$ is shown in Figure 1. The tree has $(n + 1)$ levels, where level i corresponds to i^{th} input u_i (except level 0, which corresponds to the empty root node). A node at level i represents a partially assigned pairing, where the label of the node denotes the output with which u_i is paired. At level i , a node has $(m - i)$ sub-nodes and the pairings assigned at a node are passed to all its sub-nodes. The tree has $n!C_n^m$ terminal nodes (marked by grey circles in Figure 1), which represent different pairing alternatives $P_n \in \mathbb{P}_n(N_m)$.

For ease of notation in the subsequent discussion, we introduce the concepts of fixed and candidate sets.

Definition 1 (Fixed set): The fixed set F_f is the ordered set of f output indices representing the partially assigned pairing.

Definition 2 (Candidate set): The candidate set C_c is an unordered set, whose elements can be chosen to append F_f .

Based on these definitions, a node in the solution tree is uniquely defined by its fixed and candidate sets, and is denoted as a 2-tuple, i.e. $S = (F_f, C_c)$. Furthermore, the solution tree is branched as follows:

Definition 3 (Branching rule): A node $S = (F_f, C_c)$ has $c = (m - f)$ branches. The fixed and candidate sets of the i^{th} sub-node $S^i = (F_{f+1}^i, C_{c-1}^i)$, $i = 1, 2, \dots, c$ are defined as

$$F_{f+1}^i = (F_f, c_i), \quad C_{c-1}^i = C_c \setminus c_i \quad (5)$$

Fig. 1. Solution tree for selecting control structure for a system with 4 outputs and 3 inputs

where c_i is the i^{th} element of C_c .

To illustrate the concepts of fixed and candidate sets, consider the rightmost node on level 2 in the solution tree shown in Figure 1. The fixed and candidate sets of this node are $F_2 = (4, 3)$ and $C_2 = \{1, 2\}$, respectively. Similarly, the fixed and candidate sets of the adjacent node with label 2 are $F_2 = (4, 2)$ and $C_2 = \{1, 3\}$, respectively. It can be easily established that the solution tree branched based on Definition 3 is non-redundant and every terminal node belongs to one and only one branch.

Pruning. It is noticeable that the number of nodes in the solution tree shown in Figure 1 is much larger than the number of feasible control structures, *i.e.* $n!C_n^m$. A BAB method gains its efficiency by pruning branches of the tree, which cannot lead to the optimal solution. To illustrate this, consider a node $S = (F_f, C_c)$. Then, the ensemble of all n -element ordered sets that can be obtained by expanding F_f is given as

$$\mathcal{S} = \{(F_f, P_{n-f}) | P_{n-f} \in \mathbb{P}_{n-f}(C_c)\} \quad (6)$$

where $\mathcal{S} \subseteq \mathbb{P}_n(N_m)$. For example, for the rightmost node on level 2 and its adjacent node with label 2 in the solution tree shown in Figure 1, $\mathcal{S} = \{(4, 3, 1), (4, 3, 2)\}$ and $\mathcal{S} = \{(4, 2, 1), (4, 2, 3)\}$, respectively.

Let $\underline{J}_i(\mathcal{S})$ be a lower bound of J_i , $i = 1, 2, \dots, q$, calculated over all the elements of $\mathcal{S}$, *i.e.*

$$\underline{J}_i(\mathcal{S}) \leq J_i(P_n) \quad \forall P_n \in \mathcal{S} \quad (7)$$

Further, assume that the current Pareto-optimal set $\mathcal{P}$ contains a member P_n^j such that

$$\underline{J}_i(\mathcal{S}) \geq J_i(P_n^j) \quad (8)$$

with strict inequality occurring for at least one $i \in N_q$. Then the set $\mathcal{S} \subseteq \mathbb{P}_n(N_m)$ cannot contain any Pareto-optimal solutions and hence can be pruned, *i.e.* none of the solutions contained in $\mathcal{S}$ need to be evaluated. Pruning can also be performed, if for any constraint $L_k \leq 0$, $k = 1, 2, \dots, \ell$, $\underline{L}_k(\mathcal{S}) \geq 0$, where $\underline{L}_k(\mathcal{S})$ is a lower bound of L_k over the set $\mathcal{S}$.

Updating. Whenever the BAB method encounters a terminal node P_n , the current Pareto-optimal set $\mathcal{P}$ is updated. P_n is included in $\mathcal{P}$, if for every $P_n^j \in \mathcal{P}$,

$$\exists i \in N_q : J_i(P_n) < J_i(P_n^j) \quad (9)$$

On the other hand, P_n^j is removed from $\mathcal{P}$ if

$$J_k(P_n) \leq J_k(P_n^j) \quad (10)$$

with strict inequality occurring for at least one $k \in N_q$.

IV. A BIOBJECTIVE CSD PROBLEM

To illustrate the solution of multiobjective CSD problem using BAB method, we consider a biobjective CSD problem using the minimum singular value (MSV) [14] rule as the CV selection criterion together with the μ -interaction measure (μ -IM) [15] as the pairing criterion. In this section, these criteria and their lower bounds are discussed.

A. Minimum Singular Value

CV selection involves identifying a subset among the set of available measurements (subset selection). A survey of most of the methods available for CV selection is given in [17]. Recently, Skogestad [18] introduced the concept of self-optimizing control, which involves selecting CVs such that in presence of disturbances, the loss incurred in implementing the operational policy by holding these CVs at constant setpoints is minimal, as compared to the use of an online optimizer. The choice of CVs based on the general non-linear formulation of self-optimizing control can be time-consuming. To quickly pre-screen the alternatives, Skogestad and Postlethwaite [14] presented the approximate minimum singular value (MSV) rule, which is a local method based on linearized process model and second order accurate economic objective function. Some other local methods for selection of CVs in the self-optimizing control framework are available in [19], [20].

To present the MSV rule, let $\hat{G} = D_1 G D_2$ denote the scaled gain matrix, where D_1 and D_2 are the output and input scaling matrices, respectively; see [14], [18] for details. It is noted in [21] that D_1 and D_2 are independent of the selected CVs. Then, CVs can be selected using MSV rule by solving

$$\max_{X_n \subset N_m} \underline{\sigma}(\hat{G}_{X_n}) \quad (11)$$

where $\underline{\sigma}$ is the MSV and $\hat{G}_{X_n}$ is the submatrix of $\hat{G}$ with rows indexed by the unordered set X_n . We point out that for processes with more MVs than outputs, the set of MVs can also be selected by maximizing MSV of the unscaled gain matrix [14]. In either case, the goal is to square a matrix such that the MSV of the square matrix is maximized. Different BAB methods for solving this problem have been proposed in [21], [22]. Cao and Kariwala [10] have combined these approaches to propose a very efficient BAB method for CV selection based on MSV rule in (11).

B. μ -Interaction Measure

For appropriate input-output pairing selection, a number of criteria are available in the literature [14], [23]. In this paper, we use μ -IM [15] as the pairing selection criteria. The μ -IM is useful for assessing the feasibility of stabilizing the closed-loop system through independent design of individual loops of the controller. To present this method, let the rows of the $n \times n$ dimensional transfer matrix $G_{P_n}(s)$ be permuted such that the chosen pairings lie along the diagonal and $\tilde{G}_{P_n}(s)$ represent the matrix consisting of the diagonal elements of $G_{P_n}(s)$. Then, the diagonal controller $K(s)$ stabilizing $\tilde{G}_{P_n}(s)$ also stabilizes $G_{P_n}(s)$, if [15]

$$\bar{\sigma}(\tilde{T}(j\omega)) < \mu_{\Delta}^{-1}(E(j\omega)) \quad \forall \omega \in \mathbb{R} \quad (12)$$

where μ denotes the structured singular value [24] computed with a diagonally structured Δ , $\tilde{T}(s) = \tilde{G}_{P_n}K(s)(I + \tilde{G}_{P_n}K(s))^{-1}$ and

$$E(s) = (G_{P_n}(s) - \tilde{G}_{P_n}(s))\tilde{G}_{P_n}^{-1}(s) \quad (13)$$

The condition in (12) is called μ -IM. This powerful result allows the designer to impose restrictions on the individual controllers, but still design the controller solely based on $\tilde{G}_{P_n}(s)$ such that closed loop stability is ensured. Though originally proposed for stable systems, μ -IM has recently been extended to unstable systems as well [25]. Note that if the pairings are chosen such that $\mu_{\Delta}(E(j\omega))$ is small at all frequencies, the restrictions on decentralized controller synthesis using independent design method is minimum. $\mu_{\Delta}(E(j\omega))$ can also be seen as a measure of generalized diagonal dominance [26]. Thus, the use of BAB method is also useful to find the pairing for which $G_{P_n}(j\omega)$ is most diagonally dominant. As the exact computation of μ is computationally intractable, we instead minimize the upper bound on μ (denoted as $\bar{\mu}$) obtained through D -scaling method [24].

Another useful criteria for pairing selection is based on relative gain array (RGA) [27] defined as

$$\Lambda(G_{P_n}) = G_{P_n} \circ G_{P_n}^{-T} \quad (14)$$

where $\circ$ is the element-wise or Hadamard product. In particular, pairings are avoided on the negative elements of RGA, which is a necessary condition for integrity of the closed-loop system against loop failure [28], [29]. Then, pairing selection consists of solving the following optimization problem:

$$\begin{aligned} \min_{P_n \in \mathbb{P}_n(N_n)} \quad & \bar{\mu}_{\Delta}(G_{P_n}(I \circ G_{P_n})^{-1} - I) \\ \text{s.t.} \quad & [\Lambda(G_{P_n})]_{ii} > 0; \quad i = 1, 2, \dots, n \end{aligned} \quad (15)$$

A BAB method to efficiently solve the optimization problem in (15)-(16) is presented in [13].

C. Combined CV and pairing selection

When both CV selection using MSV rule and pairing selection using μ -IM are considered together, the multiobjective

optimization problem is given as

$$\min_{P \in \mathbb{P}_n(N_m)} \{J_{\sigma}(P_n), J_{\mu}(P_n)\} \quad (17)$$

$$\text{s.t.} \quad [\Lambda(G_{P_n})]_{ii} > 0; \quad i = 1, 2, \dots, n \quad (18)$$

where

$$J_{\sigma}(P_n) = \underline{\sigma}^{-1}(\hat{G}_{P_n}) \quad (19)$$

$$J_{\mu}(P_n) = \bar{\mu}_{\Delta}(G_{P_n}(I \circ G_{P_n})^{-1} - I) \quad (20)$$

Note that the maximization of $\underline{\sigma}(\hat{G}_{X_n})$ in (11) is equivalent to minimization of $\underline{\sigma}^{-1}(\hat{G}_{X_n})$. Furthermore, for any $X_n \subset N_m$, $X_n \in \mathbb{P}_n(N_m)$. Thus, the CVs can be selected through minimization of $\underline{\sigma}^{-1}(\hat{G}_{P_n})$ over $P_n \in \mathbb{P}_n(N_m)$. We point out that for given X_n , $J_{\sigma}(P_n)$ is the same for all $P_n \in \mathbb{P}_n(X_n)$, where $\mathbb{P}_n(X_n) \subset \mathbb{P}_n(N_m)$. Thus, a BAB method based on $J_{\sigma}(P_n)$ in (19) will involve redundant computations. As the computation of $J_{\mu}(P_n)$ is much more expensive than computation of $J_{\sigma}(P_n)$, it is expected that such redundancy will not significantly affect the efficiency of the BAB method.

The application of a BAB method requires lower bounds on selection criteria for pruning purposes. To present these lower bounds, let $\mathcal{S}$ in (6) represent the ensemble of all possible solutions of the node $S = (F_f, C_c)$. Then, a lower bound on $J_{\sigma}(P_n)$ and $J_{\mu}(P_n)$ for all $P_n \in \mathcal{S}$ are given as [10], [13]

$$\underline{J}_{\sigma}(\mathcal{S}) = \max(\underline{\sigma}^{-1}(\hat{G}_{F_f}), \underline{\sigma}^{-1}(\hat{G}_{F_f \cup C_c})) \quad (21)$$

$$\underline{J}_{\mu}(\mathcal{S}) = \rho(G_{F_f}\tilde{G}_{F_f}^{-1} - I) \quad (22)$$

where ρ is the spectral radius. The lower bounds in (21) and (22) can be used together to check whether the node is dominated by a member of the current Pareto optimal set and hence can be pruned. The constraint in (18) is checked at the terminal nodes before updating the Pareto-optimal set $\mathcal{P}$.

A simplified flowchart for recursive implementation of the proposed BAB algorithm is shown in Figure 2.

V. NUMERICAL TESTS

In this section, we demonstrate the efficiency of the developed biobjective BAB method through numerical examples. First, we use randomly generated matrices to check the average performance of the BAB algorithm. Subsequently, the BAB method is applied for combined CV and pairing selection for the HDA process [16]. All these tests are carried out on a notebook with Intel® Core™ Duo Processor T2400 (1.83 GHz, 2MB RAM) using MATLAB® 2007b.

A. Random Tests

Firstly, the efficiency of the biobjective BAB algorithm is examined through some randomly generated matrices. The first test consists of selecting n CVs out of $2n$ measurements and their pairings with inputs. The BAB algorithm is applied to 100 randomly generated $2n \times n$ matrices for every n ranging from 2 to 8. The average computation time and number of node evaluations required by the BAB algorithm are shown in Figure 3. In the second test, the BAB algorithm is applied to 100 $4n \times n$ random matrices and the average computation time and number of node evaluations are shown in Figure 4.

Fig. 2. Flow chart of branch and bound algorithm

Fig. 3. Random tests for CSD for $2n \times n$ systems: (a) computation time and; (b) number of nodes evaluated

For comparison purposes, the estimated computation time and the number of node evaluations ($n!C_n^{2n}$ and $n!C_n^{4n}$) required by a brute-force approach are also shown in Figures 3 and 4. For each n , the computation time required for a brute-force approach is estimated by multiplying the number of alternatives with the time required for evaluating the MSV and $\bar{\mu}(E)$ of an $n \times n$ matrix (averaged over 1000 instances).

Both tests show that the developed BAB algorithm is able

Fig. 4. Random tests for CSD for $4n \times n$ systems: (a) computation time and; (b) number of nodes evaluated

to reduce the computation time by several orders of magnitude in comparison with brute force search. For $2n \times n$ and $4n \times n$ processes, the BAB algorithm is able to handle cases until $n = 8$ and $n = 6$ in 1 minute, respectively. The brute force approach is only able to handle processes until $n = 5$ and $n = 4$, respectively, in the same time limit.

B. HDA case study

The developed algorithm is also applied to select CVs and pairings for the HDA process, which was used as a case study in [10]. Details of self-optimizing control of this process can be found in [16]. This process has 8 MVs, hence, 8 CVs are to be selected from 129 available measurements, and to be paired with MVs. It is interesting to note that this problem has $8!C_{129}^8 \approx 6.15 \times 10^{16}$ alternatives, whereas MATLAB® requires about 2 ms to calculate the μ -IM of an 8×8 matrix. Therefore, if this problem were to be solved using the brute force approach, it would require 4 million years to find the globally optimal solution.

In comparison, the developed BAB algorithm only requires about 9 minutes to solve the CSD problem. The Pareto optimal set has 92 members, whose objective function values are shown in Figure 5. The Pareto-optimal set gives a clear overall picture for a designer to select the CVs and pairings which provide a good trade-off between the MSV rule and the μ -IM criteria. The results indicate that such a solution corresponds to a point in the low-left corner of the plot in Figure 5, which has nearly the largest MSV and the nearly the smallest $\bar{\mu}(E)$. However, it would be difficult to identify this control structure by selecting the CVs and pairings sequentially.

VI. CONCLUSIONS

The sequential selection of controlled or manipulated variables followed by pairing selection may lead to non-optimal control structures. A multiobjective branch and bound (BAB)

Fig. 5. Selection criteria values of the Pareto-optimal set for the HDA process.

method is developed to find the Pareto-optimal set of control structures, which provides a clear picture of the trade-off between different selection criteria for the practicing engineer. The application of the proposed method is illustrated using a biobjective optimization problem with minimum singular value (MSV) rule and μ -interaction measure (μ -IM) as the selection criteria for controlled variables and pairings, respectively. Numerical tests are used to demonstrate the computational efficiency of the BAB method over brute force search. In future, the computational efficiency of the BAB framework will be tested using more than two selection criteria and the method will also be extended to selection criteria other than MSV rule and μ -IM.

ACKNOWLEDGEMENTS

The first author gratefully acknowledges the financial support from Office of Finance, Nanyang Technological University, Singapore through grant no. M52120046.

REFERENCES

- [1] S. Skogestad, "Control structure design for complete chemical plants," *Comp. Chem. Engng.*, vol. 28, no. 1-2, pp. 219-234, 2004.
- [2] M. Morari, G. Stephanopoulos, and Y. Arkun, "Studies in the synthesis of control structures for chemical processes - Part I: Formulation of the problem. process decomposition and the classification of the control task. analysis of the optimizing control structures," *AIChE J.*, vol. 26, no. 2, pp. 220-232, 1980.
- [3] T. J. McAvoy and N. Ye, "Base control for the Tennessee Eastman problem," *Comput. Chem. Engng.*, vol. 18, no. 5, pp. 383-413, 1994.
- [4] W. Luyben, B. Tyreus, and M. Luyben, *Plantwide Process Control*. New York, USA: McGraw-Hill, 1998.
- [5] T. Larsson and S. Skogestad, "Plantwide control: A review and a new design procedure," *Modeling, Identification and Control*, vol. 21, no. 4, pp. 209-240, 2000.
- [6] N. V. S. N. Murthy Konda, G. P. Rangaiah, and P. R. Krishnaswamy, "Plant-wide control of industrial processes: An integrated framework of simulation and heuristics," *Ind. Eng. Chem. Res.*, vol. 44, no. 22, pp. 8300-8313, 2005.
- [7] I. K. Kookos and J. D. Perkins, "Heuristic-based mathematical programming framework for control structure selection," *Ind. Eng. Chem. Res.*, vol. 40, pp. 2079-2088, 2001.
- [8] M. L. Luyben and C. A. Floudas, "Analyzing the interaction of design and control - 1. A multiobjective framework and application to binary distillation column," *Comput. Chem. Engng.*, vol. 18, no. 10, pp. 933-969, 1994.
- [9] K. Deb, *Multi-Objective Optimization using Evolutionary Algorithms*. Chichester, UK: John Wiley & Sons, 2001.
- [10] Y. Cao and V. Kariwala, "Bidirectional branch and bound for controlled variable selection Part I. Principles and minimum singular value criterion," *Comput. Chem. Engng.*, vol. 32, no. 10, pp. 2306-2319, 2008.
- [11] V. Kariwala and Y. Cao, "Bidirectional branch and bound for controlled variable selection: Part II. Exact local method for self-optimizing control," *Comput. Chem. Eng.*, vol. In press, 2009.
- [12] Y. Cao and P. Saha, "Improved branch and bound method for control structure screening," *Chem. Engg. Sci.*, vol. 60, no. 6, pp. 1555-1564, 2005.
- [13] V. Kariwala and Y. Cao, "Efficient branch and bound methods for pairing selection," in *Proc. 17th IFAC World Congress*, Seoul, Korea, 2008.
- [14] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 1st ed. Chichester, UK: John Wiley & sons, 1996.
- [15] P. Grosdidier and M. Morari, "Interaction measures for systems under decentralized control," *Automatica*, vol. 22, no. 3, pp. 309-319, 1986.
- [16] A. C. B. D. Araujo, E. S. Hori, and S. Skogestad, "Application of plantwide control to the HDA process. II Regulatory control," *Ind. Eng. Chem. Res.*, vol. 46, pp. 5159-5174, 2007.
- [17] M. Van de Wal and B. de Jager, "A review of methods for input/output selection," *Automatica*, vol. 37, no. 4, pp. 487-510, 2001.
- [18] S. Skogestad, "Plantwide control: The search for the self-optimizing control structure," *J. Proc. Control*, vol. 10, no. 5, pp. 487-507, 2000.
- [19] I. J. Halvorsen, S. Skogestad, J. C. Morud, and V. Alstad, "Optimal selection of controlled variables," *Ind. Eng. Chem. Res.*, vol. 42, no. 14, pp. 3273-3284, 2003.
- [20] V. Kariwala, Y. Cao, and S. Janardhanan, "Local self-optimizing control with average loss minimization," *Ind. Eng. Chem. Res.*, vol. 47, no. 4, pp. 1150-1158, 2008.
- [21] V. Kariwala and S. Skogestad, "Branch and bound methods for control structure design," in *Proc. 16th ESCAPE and 9th Intl. Symposium on PSE*, Garmisch-Partenkirchen, Germany, 2006.
- [22] Y. Cao, D. Rossiter, and D. H. Owens, "Globally optimal control structure selection using branch and bound method," in *Proc. IFAC symposium on DYCOPS 5*, Corfu, Greece, 1998, pp. 183-188.
- [23] M. Van de wal, "Control structure design for dynamic systems: A review," Eindhoven University of Technology, Eindhoven, Netherlands, Technical Report WFW-94-084, September 1994.
- [24] J. C. Doyle, J. E. Wall, and G. Stein, "Performance and robustness analysis for structured uncertainty," in *Proc. of IEEE Conf. on Decision and Control*, Orlando, FL, 1982.
- [25] V. Kariwala, J. Forbes, and S. Skogestad, " μ -Interaction measure for unstable systems," *Int. J. of Automation and Control*, vol. 1, no. 4, pp. 295-313, 2007.
- [26] V. Kariwala, J. F. Forbes, and E. S. Meadows, "Block relative gain: Properties and pairing rules," *Ind. Eng. Chem. Res.*, vol. 42, no. 20, pp. 4564-4574, 2003.
- [27] E. H. Bristol, "On a new measure of interaction for multivariable process control," *IEEE Trans. Automat. Contr.*, vol. 11, 1966.
- [28] P. Grosdidier, M. Morari, and B. R. Holt, "Closed-loop properties from steady-state gain information," *Ind. Eng. Chem. Fundam.*, vol. 24, pp. 221-235, 1985.
- [29] V. Kariwala, J. F. Forbes, and E. S. Meadows, "Integrity of systems under decentralized integral control," *Automatica*, vol. 41, no. 9, pp. 1575-1581, 2005.