New Multi-Commodity Flow Formulations for the Generalized Pooling Problem*

Xin Cheng∗ Kai Tang∗∗ Xiang Li∗∗∗

∗ Department of Chemical Engineering, Queen’s University, Kingston, ON, K7L3N6, Canada (e-mail: x.cheng@queensu.ca).
** Department of Chemical Engineering, Queen’s University, Kingston, ON, K7L3N6, Canada (e-mail: 16kt16@queensu.ca)
*** Department of Chemical Engineering, Queen’s University, Kingston, ON, K7L3N6, Canada (e-mail: xiang.li@queensu.ca)

Abstract: The generalized pooling problem is involved in many planning and scheduling problems in the petrochemical industry. Compared to the standard pooling problem where the blenders (or pools) are not allowed to be connected to one another, the generalized pooling problem has a more complex network structure and allows more types of problem formulations. The state-of-the-art generalized pooling formulations adopt a multi-commodity flow (MCF) strategy that was first proposed by Alfaki and Haugland (2013a) and proved to be stronger than the classical p-formulation. This paper proposes two new MCF formulations for the generalized pooling problem, using mixing and split fractions of blenders rather than the commodity flow fractions. The case study results show that, for some cases, the proposed formulations perform better than the existing MCF formulations, but none of the formulations dominates others for all cases. The results also show that formulations which have similar sizes and similarly tight linear programming relaxations may have dramatically different performance.

Keywords: Pooling problem; Multi-commodity flow; Global optimization; Blending; Network flow optimization.

1. INTRODUCTION

The pooling problem is a special type of network flow optimization problem, which was originally studied for gasoline blending in oil refineries (Haverly (1978)). In a pooling network, flows from different supply tanks are blended at blenders (or pools) and then sent to demand tanks to form final products. Since blending operation changes the flow qualities and it needs to be described with bilinear functions, the pooling problem is a nonconvex nonlinear programming (NLP) problem. The generalized pooling problem is an extension of the standard pooling problem where at least two blenders are connected to each other. The pooling problem has been recognized as an important class of optimization problems in the petrochemical industry (Bodington and Baker (1990)), because blending appears in many petrochemical processes. Due to the nonconvexity of the pooling problem, the pooling problem is usually solved by a branch-and-bound based global optimization method, and the efficiency of the branch-and-bound search is known to be largely dependent on how tight the linear programming (LP) relaxation of the formulation is. Many pooling problem formulations have been studied in the literature (Gupte et al. (2017)). Well-known formulations for the standard pooling problem include P-, Q-, PQ-, and TP- formulations (Tawarmalani and Sahinidis (2002), Alfaki and Haugland (2013b)). The P-formulation models the blending operation using the flow rates and the flow qualities, while the Q-formulation replaces the flow qualities with the fractions of flows that come from the supply tanks. The PQ-formulation comprises the Q-formulation and extra strengthening constraints, and it is known to be stronger than the P- and Q-formulations. The TP-formulation is similar to the PQ-formulation but it uses the fractions of flows that go to the demand tanks. The TP-formulation sometimes performs better than the PQ-formulation and sometimes does not.

Recently, more attention has been paid to strong formulations of the generalized pooling problem. Alfaki and Haugland (2013a) proposed a multi-commodity flow (MCF) formulation, where the material in a supply tank is viewed as a monolithic commodity rather than a mixture of multiple components. They have shown that the MCF formulation reduces to the PQ-formulation for the standard pooling problem. Based on a similar idea, Boland et al. (2016) proposed a different MCF formulation that extends the TP-formulation for the generalized pooling problem, where a commodity is defined to be the product in a demand tank. They also proposed some other MCF formulations that combine the supply commodities, demand commodities, and commodity paths in different ways, and in their extensive case studies the supply commodity and the demand commodity based formulations performed better than the other formulations. On the other hand, a rather different modeling strategy has been used in the field of process sys-

* The authors are grateful to the Collaborative Research and Development Grants (CRDPJ 485798) from Natural Sciences and Engineering Research Council of Canada.
tems engineering (e.g., Quesada and Grossmann (1995)). In this strategy, the blending operation is modeled with individual components in the flows and the blender split fractions. Based on this strategy, Lotero et al. (2016) proposed a multi-commodity based split fraction formulation for multi-period blending problem, and they proved that this formulation leads to tighter LP relaxations than the component based split fraction formulation. Note that the multi-period blending problem in their work differs from the generalized pooling problem considered in this paper, because it considers the fractions of tank inventories rather than the fractions of tank throughputs.

This paper is concerned with strong formulations for the generalized pooling problem. There are two major contributions of the paper. One contribution is the proposal of two new MCF formulations, where the first one is similar to the one proposed by Lotero et al. (2016) for multi-period blending, and the second one is a new formulation based on mixing fractions and demand commodities. The other contribution is to show through case studies that, the two new MCF formulations sometimes perform better than the MCF formulations in the literature, but no formulation is always better than the others. In addition, the formulations can have significantly different performance even when their sizes are similar and their LP relaxations (at root nodes) are similarly tight.

The remaining part of the paper is organized as follows: Section 2 provides a descriptive problem statement with a list of symbols. Section 3 introduces two representative MCF formulations in the literature. Section 4 proposes two new MCF formulations that use blender mixing and split fractions, respectively. Section 5 compares the performance of the four MCF formulations through three case study problems. The paper ends with concluding remarks in Section 6.

2. PROBLEM STATEMENT

The generalized pooling network can be viewed as a acyclic graph $G = (N, A)$. The set of nodes $N$ consists of three subsets $S, B, D$, which include supply tanks, blenders, and demand tanks, respectively. The set of arcs $A$ includes all allowable connections between two tanks. In operation of the pooling network, material flows leave the supply tanks, and they are blended once or multiple times before entering the demand tanks. The goal of optimization is to determine the flow rates along all arcs of the network such that the total profit is maximized. Since the operation is assumed to be at a steady state, the inventory levels of the tanks are not considered in the problem. A general descriptive optimization formulation is given below:

$$\begin{align*}
\text{min.} \quad & \text{Negative profit} \\
\text{s.t.} \quad & (1) \text{ Flow quality change through blenders;} \\
& (2) \text{ Mass balance around blenders;} \\
& (3) \text{ Bounds on product qualities;} \\
& (4) \text{ Bounds on total flows going through tanks;} \\
& (5) \text{ Bounds on individual variables.}
\end{align*}$$

In the above formulation, constraint (1) restricts how flow qualities change through blending; constraint (2) enforces mass balance for the inlet flows and outlet flows of each blender; constraint (3) observes quality specifications of products in the demand tanks (such as concentration of a key chemical component); constraint (4) represents bounds on total flow rates through each tank, which result from the availability of materials, the blending capacity, and the costumer demands; constraint (5) imposes bounds on individual variables, such as non-negativity bounds of flow rates. Table 1 shows a list of symbols are will be used for the mathematical formulations in the subsequent sections.

The existing generalized pooling formulations differ primarily in the way to model constraint (1) (and accordingly constraint (2) as well). The classical P-formulation explicitly includes flow qualities as variables, and the change of flow qualities can be described by bilinear equations involving flow qualities and flow rates. The MCF formulations use a different idea, where the flow qualities are not defined as variables explicitly. In these formulations, each physical flow is logically disaggregated into several parts, and each part is called a commodity flow that is defined to originally come from a supply tank or eventually go to a demand tank. The quality of a physical flow can be calculated from the composition of the flow (in terms of the commodities). It has been proven that the MCF formulations lead to tighter LP relaxations than the P formulation and therefore favor branch-and-bound based global optimization (Alfaki and Haugland (2013a), Boland et al. (2016)).

3. TWO REPRESENTATIVE MCF FORMULATIONS FROM THE LITERATURE

This section introduces two representative MCF formulations, proposed by Alfaki and Haugland (2013a) and Boland et al. (2016) respectively. They are called supply based MCF formulation (SMCF) and demand based MCF formulation (DMCF) in this paper. According to the extensive simulations studies by Boland et al. (2016), SMCF and DMCF formulations usually outperform other MCF formulations in the literature.

<table>
<thead>
<tr>
<th>Table 1. Notation for the pooling formulations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Sets</strong></td>
</tr>
<tr>
<td>$S$</td>
</tr>
<tr>
<td>$B$</td>
</tr>
<tr>
<td>$D$</td>
</tr>
<tr>
<td>$N$</td>
</tr>
<tr>
<td>$A$</td>
</tr>
<tr>
<td>$K$</td>
</tr>
<tr>
<td>$S_i$</td>
</tr>
<tr>
<td>$D_i$</td>
</tr>
<tr>
<td><strong>Parameters</strong></td>
</tr>
<tr>
<td>$\beta_s$</td>
</tr>
<tr>
<td>$\beta_d$</td>
</tr>
<tr>
<td>$\lambda_s^k$</td>
</tr>
<tr>
<td>$\lambda_d^k$</td>
</tr>
<tr>
<td>$\mu_i^b$</td>
</tr>
<tr>
<td><strong>Variables</strong></td>
</tr>
<tr>
<td>$f_{i,j}$</td>
</tr>
<tr>
<td>$f_{i,s}/z_{i,t}$</td>
</tr>
<tr>
<td>$z_{i,b,j,l}$</td>
</tr>
<tr>
<td>$x_{i,j}$</td>
</tr>
<tr>
<td>$x_{i,k}/x_{i,b}j$</td>
</tr>
</tbody>
</table>

163
Fig. 1. Illustration of SMCF and DMCF strategies

3.1 The Supply Based MCF Formulation

In the SMCF formulation, a commodity flow \( l \) in a physical flow is the part of the flow that originally comes from supply tank \( i \) (\( i \in S \)), and it is called supply commodity flow in the paper for convenience. The ratio of the supply commodity flows to the physical flow are defined as fractional variables, and the composition of a physical flow can be calculated from the supply commodity fractions in that flow. Obviously, the supply commodity fractions in flow \((i,j)\) equal to those in tank \(i\). For a blender, the inlet flows may have different supply commodity fractions but the outlet flows must have the same fractions to each other. This strategy to model the flow composition change is illustrated by Figure 1(a). The mathematical SMCF formulations is shown below:

Objective:

\[
\min \sum_{(s,j) \in A} \beta_s f_{sj} - \sum_{(i,d) \in A} \beta_d f_{id} \tag{SMCF-1}
\]

Supply commodity fraction:

\[
z_{ijl} = f_{ij} x_{il}, \quad \forall (i,j) \in A, l \in S_i \tag{SMCF-2}
\]

\[
\sum_{l \in S_b} x_{il} = 1, \quad \forall b \in B \tag{SMCF-3}
\]

Mass balance:

\[
\sum_{(i,b) \in A} z_{ibl} = \sum_{(b,l) \in A} z_{bjl}, \quad \forall b \in B, l \in S_b \tag{SMCF-4}
\]

Quality bounds:

\[
\sum_{(i,d) \in A} z_{ild} \lambda_{sk} \leq \sum_{(d,k) \in A \cup S_i} z_{ild} \lambda_{dk}^{U_l}, \quad \forall d \in D, k \in K \tag{SMCF-5}
\]

Node capacity:

\[
\sum_{(i,j) \in A} f_{ij} \leq \mu_i^U, \quad \forall i \in N \setminus D \tag{SMCF-6}
\]

\[
\sum_{(i,d) \in A} f_{id} \leq \mu_d^U, \quad \forall d \in D \tag{SMCF-7}
\]

Variable bounds:

\[
0 \leq x_{il} \leq 1, \quad \forall b \in B, l \in S_b \tag{SMCF-8}
\]

\[
f_{ij} \geq 0, \quad \forall (i,j) \in A \tag{SMCF-9}
\]

Strengthening constraints:

\[
\sum_{l \in S_b} z_{ibj} = f_{bj}, \quad \forall b \in B, (b,j) \in A \tag{SMCF-10}
\]

\[
\sum_{(b,j) \in A} z_{ibj} \leq \mu_i^U x_{bl}, \quad \forall b \in B, l \in S_b \tag{SMCF-11}
\]

In the SMCF formulation, \( x_{il} \) denotes the fraction for supply commodity \( l \) in tank \(i\). The bilinear equation (SMCF-2) enforces the same \( x_{il} \) for the outlet flows of tank \(i\). When \(i = s\) (i.e., \(i\) is a supply tank), \( x_{il} \) is a parameter, which is 1 when \(l = s\) and 0 when \(l \neq s\); in this case, (SMCF-2) reduces to a linear equation. If the problem contains only one supply tank (i.e., \(|S| = 1\)), then the problem reduces to a LP problem. The unity equation (SMCF-3) states the fact that any part of a flow through a blender must come from one of the supply tanks.

The strengthening constraints (SMCF-10) and (SMCF-11) are obtained via the reformulation-linearization technique (RLT) (Sherali and Alameddine (1992)). They are redundant for modeling the problem but can tighten the LP relaxation of the formulation for efficient global optimization. Specifically, (SMCF-10) comes from multiplying both sides of (SMCF-3) by \( f_{bj} \), and (SMCF-11) from multiplying both sides of (SMCF-6) by \( x_{bl} \).

3.2 The Demand Based MCF Formulation

While SMCF is an extension of the PQ-formulation for the generalized pooling problem, DMCF is an extension of the TP-formulation. In the DMCF formulation, a commodity flow \( l \) is defined to be the part of a flow that eventually goes to demand tank \( l \) (\( l \in D \)), and for convenience, it is called a demand commodity flow in this paper. Consequently, the ratio of the demand commodity flows to the physical flow are expressed as fractional variables. In contrast to the supply commodity fractions, the demand commodity fractions in flow \((i,j)\) equal to those in tank \(j\). For a blender, the inlet flows must have the same demand commodity fractions but the outlet flows may have different fractions. This strategy is illustrated by Figure 1(b), and the DMCF formulation is shown below:

Objective:

\[
\min \sum_{(s,j) \in A} \beta_s f_{sj} - \sum_{(i,d) \in A} \beta_d f_{id} \tag{DMCF-1}
\]

Demand commodity fraction:

\[
z_{ijl} = f_{ij} x_{jl}, \quad \forall (i,j) \in A, l \in D_j \tag{DMCF-2}
\]

\[
\sum_{l \in D_b} x_{jl} = 1, \quad \forall b \in B \tag{DMCF-3}
\]

Mass balance:

\[
\sum_{(i,b) \in A} z_{ibl} = \sum_{(b,l) \in A} z_{bjl}, \quad \forall b \in B, l \in D_b \tag{DMCF-4}
\]

Quality bounds:

\[
\sum_{s \in S (s,j) \in A} z_{sji} \lambda_{sk} \leq \sum_{s \in S (s,j) \in A} z_{sji} \lambda_{dk}^{U_l}, \quad \forall l \in D, k \in K \tag{SMCF-5}
\]
Node capacity:
\[
\sum_{(i,j) \in A} f_{ij} \leq \mu^U_{ij}, \quad \forall i \in N \setminus D \quad \text{(DMCF-6)}
\]
\[
\sum_{(i,d) \in A} f_{id} \leq \mu^U_{id}, \quad \forall d \in D \quad \text{(DMCF-7)}
\]

Variable bounds:
\[
0 \leq x_{bl} \leq 1, \quad \forall b \in B, l \in D_b \quad \text{(DMCF-8)}
\]
\[
f_{ij} \geq 0, \quad \forall (i,j) \in A \quad \text{(DMCF-9)}
\]

Strengthening constraints:
\[
\sum_{b \in B} z_{ibl} = x_{bl}, \quad \forall b \in B, (i,b) \in A \quad \text{(DMCF-10)}
\]
\[
\sum_{b \in B} z_{ibl} \leq \mu^U_{ib}, \quad \forall b \in B, l \in D_b \quad \text{(DMCF-11)}
\]

Here \( x_{bl} \) stands for the fraction for demand commodity \( l \) in tank \( i \). When \( i = d \), \( x_{dl} \) is a parameter, which is 1 when \( l = d \) and 0 when \( l \neq d \), and the bilinear equation (DMCF-2) reduces to a linear equation. The strengthening constraints are obtained by the RLT technique from constraints (DMCF-3) and (DMCF-6).

4. THE NEW MCF FORMULATIONS

The new MCF formulations are motivated by the multicomponent flow strategy that was originally proposed in the process systems engineering community (Quesada and Grossmann (1995)). The main idea of this strategy is to disaggregate a physical flow into flows of the involved chemical components. At a blender, all inlet chemical component flows are split to different outlets with the same split fractions. Lotero et al. (2016) proposed a similar strategy that disaggregates the physical flows into supply commodity flows rather than chemical component flows, and applied the strategy to multi-period blending (where the split fractions are fractions of tank inventories rather than the tank throughputs). In this section, the formulation by Lotero et al. (2016) is modified for the generalized pooling problem, and the resulting formulation is called the supply and split fraction (SSF) based formulation. After that, a new formulation is proposed based on the demand commodity flow and the mixing fractions at the blenders. This formulation is called demand and mixing fraction (DMF) based formulation.

4.1 The Supply and Split Fraction Based Formulation

The SSF formulation involves the supply commodity flows along all arcs but not the physical flows (except for the supply tank outlet flows that contain only one commodity by definition). In order to model the flow composition change, the split fractions of the blenders \( x_{bj} \) are defined. This strategy is illustrated by Figure 2(a), and the SSF formulation is shown below:

Objective:
\[
\min \sum_{s \in S} \sum_{(s,j) \in A} \beta_s f_{sj} - \sum_{d \in D} \sum_{(i,d) \in A \cap S_d} \beta_d f_{idl} \quad \text{(SSF-1)}
\]

Blender split fraction:
\[
z_{ibjl} = x_{ibl} x_{bj}, \quad \forall b \in B, (i,b),(b,j) \in A, l \in S_b \quad \text{(SSF-2)}
\]

In the above formulation, the bilinear equation (SSF-2) enforces the same split fraction for all inlet commodity flows, and \( z_{ibjl} \) actually represents the rate of demand commodity \( l \) that comes from tank \( s \) to tank \( j \) through blender \( b \). (SSF-2) states that the sum of the split fractions is one. The strengthening constraints are obtained from constraints (SSF-3), (SSF-6).

4.2 The Demand and Mixing Fraction Based Formulation

The DMF formulation uses the demand commodity flow rather than the supply commodity flow. Note that at a blender, the fraction of an inlet demand commodity flow...
that goes to an outlet does not equal to the blender split fraction. For example, the fraction of an inlet demand commodity flow \( f_{ijl} \) that goes to an outlet is 0 if the outlet is not on a path to demand tank \( l \), no matter what the split fraction for the outlet is. On the other hand, the inlet demand commodity flows contribute to any outlet demand commodity flows with the same fractions, which equal to the mixing fractions of the inlets. For example, assume \( x_{ib} \) to be the fraction for the \( ith \) inlet flow in the total inlet flow. Then for any outlet demand commodity flow \( f_{bjl} \) (i.e., the part of outlet flow \( f_{bjl} \) that eventually goes to demand tank \( l \)), \( x_{ib} \) of it comes from the \( ith \) inlet. This strategy is illustrated in Figure 2(b), and the DMF formulation is as follows:

**Objective:**

\[
\begin{align*}
\min \quad & \sum_{s \in S} \sum_{(s,j) \in A} \sum_{l \in D_s} \beta_s f_{sjl} - \sum_{d \in D} \sum_{(i,d) \in A} \beta_d f_{idal} \\
\end{align*}
\]  

(DMF-1)

**Blender mixing fraction:**

\[
\begin{align*}
z_{ibjl} = f_{bjl} x_{ib}, \quad \forall b \in B, (i,b), (b,j) \in A, l \in D_b \\
\sum_{(i,b) \in A} x_{ib} = 1, \quad \forall b \in B
\end{align*}
\]  

(DMF-2)

**Mass balance:**

\[
\begin{align*}
f_{bl} &= \sum_{(b,j) \in A} z_{ibjl}, \quad \forall b \in B, \forall (i,b) \in A, l \in D_b
\end{align*}
\]  

(DMF-3)

**Quality bounds:**

\[
\begin{align*}
\sum_{s \in S} \sum_{(s,j) \in A} f_{sjl} \lambda_{sk}^U \leq \sum_{s \in S} \sum_{(s,j) \in A} f_{sjl} \lambda_{sk}^L, \quad \forall l \in D, k \in K
\end{align*}
\]  

(DMF-5)

**Node capacity:**

\[
\begin{align*}
\sum_{(i,j) \in A \in D_j} f_{ijl} \leq \mu_{ij}^U, \quad \forall i \in N \setminus D \\
\sum_{(i,d) \in A \in D_d} f_{idal} \leq \mu_{idal}^U, \quad \forall d \in D
\end{align*}
\]  

(DMF-6)

(DMF-7)

**Variable bounds:**

\[
\begin{align*}
0 \leq x_{ib} \leq 1, \quad \forall b \in B, (i,b) \in A \\
f_{ijl} \geq 0, \quad \forall (i,j) \in A, l \in D
\end{align*}
\]  

(DMF-8)

(DMF-9)

**Strengthening constraints:**

\[
\begin{align*}
\sum_{(i,b) \in A} z_{ibjl} = f_{bjl}, \quad \forall b \in B, (i,b) \in A, l \in D_b \\
\sum_{(b,j) \in A} z_{ibjl} \leq \mu_{b,j}^U x_{ib}, \quad \forall b \in B, (i,b) \in A
\end{align*}
\]  

(DMF-10)

(DMF-11)

The above strengthening constraints are obtained from constraints (DMF-3), (DMF-6). Note that the left-hand-side of quality constraint (DMF-5) represents the total quality \( k \) (contributed by demand commodities from all supply thanks) entering demand tank \( l \), and the right-hand-side of the constraint represents the maximum quality \( k \) allowed to enter the same tank.

### 5. SIMULATION STUDIES

The purpose of the simulation study is to demonstrate that the four MCF formulations, especially the last three that have not attracted much attention form the process systems engineering community, have significant computational advantages for certain problem instances. Due to the page limit of the paper, only three examples are presented and discussed here. Examples 1 and 2 are depicted in Figures 3 and 4 respectively, where the values of parameters used are also labeled. Example 3 is adapted from a natural gas production network operation problem that was originally developed in Setlot (2009). The basic problem information can be found in Li et al. (2011), but in addition to CO\(_2\), five more components are considered, including N\(_2\), C\(_2\), C\(_3\), C\(_4\), C\(_5\). The quality constraints are the upper bounds of the six components, given in Table 4.1 in Setlot (2009). In addition, gas price (i.e., \( \beta_d \)) is changed into 0.00536417 ($/mol) and the gas costs (i.e., \( \beta_s \)) are also updated and shown in Table 2.

The case studies were performed on a virtual machine with a 3.40 GHz CPU, 4GB memory, and Ubuntu 16.02 operating system. The case study problems were formulated on GAMS 24.8.5 (Bussieck and Meeraus (2004)) and solved by BARON 17.4.1 Tawarmalani and Sahinidis (2005). For all cases, the relative termination tolerance is 10^{-3}.

Table 3 provides the topological information, number of qualities, and the optimal objective values of the example problems (where \(|·|\) denotes the cardinality of a set). The computational results with the four MCF formulations are shown in Table 4, from which we can have the following...
Table 2. Cost (βs) information for Example 3

<table>
<thead>
<tr>
<th>Gas Field</th>
<th>D35</th>
<th>BY</th>
<th>SC</th>
<th>E11</th>
<th>F6</th>
<th>F23SW</th>
</tr>
</thead>
<tbody>
<tr>
<td>βs ($/mol)</td>
<td>0.001</td>
<td>0.001</td>
<td>0.001</td>
<td>0.003</td>
<td>0.001</td>
<td>0.002</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Gas Field</th>
<th>M4</th>
<th>M1</th>
<th>JN</th>
</tr>
</thead>
<tbody>
<tr>
<td>βs ($/mol)</td>
<td>0.001</td>
<td>0.002</td>
<td>0.001</td>
</tr>
</tbody>
</table>

Table 3. Case study problem characteristics

<table>
<thead>
<tr>
<th>Cases</th>
<th>Formulations</th>
<th>Bilinear equ. terms</th>
<th>Time (s)</th>
<th>Rel. gap at root node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ex. 1</td>
<td>SMCF</td>
<td>25 8 10 x</td>
<td>24.2</td>
<td>55.67%</td>
</tr>
<tr>
<td></td>
<td>DMCF</td>
<td>19 7 8 x</td>
<td>1.1</td>
<td>55.74%</td>
</tr>
<tr>
<td></td>
<td>SSF</td>
<td>59 23 8 x</td>
<td>3.5</td>
<td>55.74%</td>
</tr>
<tr>
<td></td>
<td>DMF</td>
<td>53 22 7 x</td>
<td>1.2</td>
<td>55.74%</td>
</tr>
</tbody>
</table>

| Ex. 2 | SMCF         | 48 10 15 x           | 4.7      | 96.73%                |
|       | DMCF         | 28 8 10 x            | 20.6     | 98.22%                |
|       | SSF          | 132 39 10 x          | 314.5    | 98.22%                |
|       | DMF          | 104 36 8 x           | 0.3      | 98.11%                |

| Ex. 3 | SMCF         | 79 14 58 x           | 31.8     | 4.46%                 |
|       | DMCF         | 60 30 23 x           | 2.1      | 1.11%                 |
|       | SSF          | 186 93 8 x           | 1.0      | 5.32%                 |
|       | DMF          | 81 29 28 x           | 0.3      | 5.32%                 |

Table 4. Results for the four MCF formulations

<table>
<thead>
<tr>
<th>Cases</th>
<th>Formulations</th>
<th>Time (s)</th>
<th>Rel. gap at root node</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ex. 1</td>
<td>SMCF</td>
<td>24.2</td>
<td>55.67%</td>
</tr>
<tr>
<td></td>
<td>DMCF</td>
<td>1.1</td>
<td>55.74%</td>
</tr>
<tr>
<td></td>
<td>SSF</td>
<td>3.5</td>
<td>55.74%</td>
</tr>
<tr>
<td></td>
<td>DMF</td>
<td>1.2</td>
<td>55.74%</td>
</tr>
<tr>
<td>Ex. 2</td>
<td>SMCF</td>
<td>4.7</td>
<td>96.73%</td>
</tr>
<tr>
<td></td>
<td>DMCF</td>
<td>20.6</td>
<td>98.22%</td>
</tr>
<tr>
<td></td>
<td>SSF</td>
<td>314.5</td>
<td>98.22%</td>
</tr>
<tr>
<td></td>
<td>DMF</td>
<td>0.3</td>
<td>98.11%</td>
</tr>
<tr>
<td>Ex. 3</td>
<td>SMCF</td>
<td>31.8</td>
<td>4.46%</td>
</tr>
<tr>
<td></td>
<td>DMCF</td>
<td>2.1</td>
<td>1.11%</td>
</tr>
<tr>
<td></td>
<td>SSF</td>
<td>1.0</td>
<td>5.32%</td>
</tr>
<tr>
<td></td>
<td>DMF</td>
<td>0.3</td>
<td>5.32%</td>
</tr>
</tbody>
</table>

Table 2. Cost (βs) information for Example 3

Gas Field D35 BY SC E11 F6 F23SW
βs ($/mol) 0.001 0.001 0.001 0.003 0.001 0.002

Gas Field M4 M1 JN
βs ($/mol) 0.001 0.002 0.001

6. CONCLUDING REMARKS

According to the best of our knowledge, the DMCF, SSF, DMF formulations have not attracted much attention for the generalized pooling problem within the process systems engineering community. The case study results indicate that one may consider using these formulations when the classical P-formulation and the SMCF formulation do not work well. The case study results also show that, the LP relaxation and the problem size, which are widely accepted criteria to assess the generalized pooling formulation, are not the only factors determining the performance of the formulation. A rigorous theoretical explanation for our findings is an open question for the future research.

REFERENCES


