Solvers Resolution Perspective to the Capacitated Centred Clustering Problem

Pablo Luis Fernandes Batista, Marcos Negreiros, Augusto Palhano

1 Universidade Estadual do Ceará (UECE)
Mestrado Profissional em Computação Aplicada
MPCOMP/UECE-IFCE
Av. Paranjana, 1700 – Campus do Itaperi
CEP: 60740-000 – Fortaleza – CE – Brazil

opablofernandes@hotmail.com, negreiro@graphvs.com.br

Resumo. Este trabalho apresenta um elaborado estudo sobre o processo de resolução com "solvers" das plataformas ARGONNE e LINDO Systems, de um modelo matemático proposto para o problema de geral de agrupamento capacitado em centro geométrico (PACCG). Investigam-se os resultados numéricos obtidos com a aplicação de relaxação lagrangeana sobre o modelo original, a seguir avalia-se o comportamento das soluções do modelo considerando uma resolução formal de algumas instâncias da literatura. Como resultado, mostra-se que os "solvers" escolhidos da plataforma ARGONNE indicaram falsos resultados ótimos para instâncias cujos limites superiores são conhecidos, enquanto que no LINGO os resultados apresentaram limites superiores de boa qualidade. O estudo mostra que foram obtidos poucos limites inferiores confiáveis para os casos avaliados, porém foram obtidos novos limites superiores, assim como soluções ótimas de instâncias da literatura foram provadas.

Palavras Chaves: Agrupamento, PACCG, Solvers.
Área principal : OC

Abstract. This work shows an elaborated study about the solvers resolution process under ARGONNE and LINDO Systems environments, to deal with the Capacitated Centered clustering problem (CCCP). It is considered the numerical results obtained with a lagrangean relaxation to the CCCP. It is also evaluated the full model resolution using selected instances from the literature. It is shown that the chosen ARGONNE solvers indicated false optimal results; meanwhile LINGO returns good upper bounds. This study shows that few confident lower bounds were obtained, although new upper bounds were achieved and optimal solutions were proved from the set of instances of the literature.

Keywords: Clustering, CCCP, MP Solvers.
Main area: OC

1 Introduction

Min-sum-clustering is a very well-known problem related to the process of assigning individuals to a number of disjoint partitions. The idea behind is to form groups where the most similar individuals are joined in its best corresponding group. A similarity measure
is a function of least-square distance between individuals to the geometric center of each partition. This problem is defined as unconstrained min-sum-of-squares clustering, it is NP-Hard and has a number of exact and approximate methods that can find optimal or very close to optimal solutions, [10], [5], [21], [22], [3].

Negreiros & Palhano (2006) proposed a new version of the problem, considering in the Euclidean plane, a constrained process of doing min-sum clustering of a set of individuals, [15]. Once clustering can be done in many different ways, this new problem searches for solutions where there are limits on the capacity of the clusters or even their size (maximum number of individuals per cluster). This new problem is also NP-Hard, and introduces different ways of exploring combinatorial optimization methods to solve it.

Two variations of the constrained clustering problem were proposed, the $p$-CCCP and the generalized CCCP or $g$CCCP. In the first version, the capacitated constrained min-sum clustering is bounded by a number of groups, where in the second generalized version there is no limit on the number of groups, but it is added to the objective function a fixed cost to open a new cluster.

Effort is being done for some researchers in the direction of solving the CCCP. The first direction was in heuristic field mainly explored by [15] which used constructive methods and VNS, [17] used constructive strategies with new moves called wave and fireworks, [6] proposed Simulated Annealing and Cluster Search, [7] proposed Genetic Algorithms and Cluster Search, and most recently [14] proposed a framework using Path Relinking, Tabu Search and multistart step.

A second direction was explored by [18], where they used column generation results obtained from $p$-median, to adapt the resulting partitions to calculate the clusters centers.

Tries in obtaining optimal solutions from direct models were done by [19]. Lately [16], explored a B&B combinatorial procedure to the problem, where they proved optimality to max-cut instances up to 20 vertices.

The use of solvers to deal with mathematical programming models is a constant actually. These resources are available in many free or proprietary platforms, where some are built in most advanced parallel environments of the world. Considering this possibility, most recently NEOS ARGONNE Labs are disposing their supercomputers (ARGONNE parallel architecture) to run mathematical programming solvers in their site, [12]. Basically they are state-of-the-art mathematical programming solvers, like: MINOS, FilMINT, Lancelot, and many others.

The user can access the site and use for free all the resources of the platform, by providing a model written in AMPL, [9]. Some of the solvers may limit use; however NEOS does not limit the number of variables or constraints. The limit of 1.5 GB of memory for a job is imposed. The user must select the solver first, and send three files: one with the model, other with the data and the last with the displays and the programming part. Once they were sent to the environment, the job is launched concurrently with many others. NEOS will give to the user a number of the process which will be identified by a login and password given by the user. At the end, the solution is provided as its optimality status, [4].

Some of the solvers used in NEOS are commercial products that are available
for use for free through the NEOS Server. Others are NEOS Server implementations of academic codes or open source codes that may or may not be available for download (FilMINT). The daily use of this mathematical programming tool is impressive, [11], [2], [8].

LINDO Systems in counterpart provides a proprietary tool, LINGO, to deal with mathematical programming models, using their own solvers and algebraic tools of formulation and visualization of the results. This system is well known in the market for almost 30 years, and it is still considered as a state-of-the-art mathematical programming tool, [20].

Once those tools are available in some sense, it is of a great motion to explore their limits. And considering the CCCP as a challenge, it is a motivation to design better quality exact methods to hold to optimality bigger instances.

In this work, section 2 review the mathematical formulation for the studied version of the CCCP, in section 3 we show the aspects and results of the lagrangean relaxation used to deal the model which mainly obtain lower bounds to the CCCP instances, it is also presented the results achieved with the proposal LR by using NEOS ARGONNE and LINDO Systems solvers. In section 4 it is applied the full model for exact resolution by NEOS ARGONNE and LINDO System. In section 5 it is done the conclusion.

2 Description of the Problem

Let’s suppose that the general CCCP can be represented by using the following sets, parameters and variables:

Sets:
- \( d \) - is the dimension of the space (\( d = 2 \), in our case);
- \( I \) - is the set of individuals (\( m = |d| \));
- \( J \) - is the set of clusters centers;
- \(|J|\) - is the cardinality of the set \( J \), or a fixed number of clusters (\(|J| = p\));

Parameters:
- \( p \) - is the number of clusters;
- \( a_i \) - is a vector of \( d \) dimension with the coordinates of the individual \( i \);
- \( q_i \) - is the demand of an individual \( i \);
- \( Q \) - is the maximum capacity of a cluster;
- \( n_j \) - is the number of individuals in cluster \( j \);
- \( F \) - fixed cost to open a cluster;

Variables:
- \( \bar{x}_j = \begin{cases} 1, & \text{is a vector of dimension } d \text{ representing the center coordinates of the cluster } j \\ 0, & \text{otherwise} \end{cases} \)
- \( y_{ij} = \begin{cases} 1, & \text{if an individual } i \text{ is assigned to the cluster } j \\ 0, & \text{otherwise} \end{cases} \)
- \( z_j = \begin{cases} 1, & \text{If cluster } j \text{ is opened} \\ 0, & \text{otherwise} \end{cases} \)
The objective function (1) wants to minimize the Euclidean distance between clusters centers and individuals assigned to each cluster and the cost of opening a new cluster. The constraint (2) assign one individual to just one cluster. The constraint (3) one new cluster must be opened. The constraint (4) consider the number of individuals per cluster. The constraint (5) defines the geometric center of the clusters. The constraint (6) limits the assigned individuals to the maximum capacity of the opened cluster. The constraints (7) and (8) refer to the decision variables of the problem.

The gCCCP is non-linear and binary. It is NP-Hard, once its unconstrained version is also NP-Hard, [10]. Its main difficulty is related to the knapsack constraint, although it is also non trivial if we just consider the constraints to form the center of the clusters.

### 3 CCCP Lagrangean Relaxation

To deal with the difficulties related to the problem, it is considered that a lagrangean relaxation of the problem could be obtained to produce feasible cuts in short time, once the complexity related to the problem. A lagrangean model may be derived from (gCCCP) model, if it is considered the following relaxed version:
\[(RL - gCCCP)\text{Minimize} \sum_{j \in J} F z_j + \sum_{i \in I} \sum_{j \in J} \|a_i - \bar{x}_j\| y_{ij} \]
\[+ \sum_{j \in J} \lambda_j \sum_{i \in I} q_i y_{ij} - Q z_j \]
\[\text{such that :} \sum_{j \in J} y_{ij} = 1, \forall i \in I \]
\[\sum_{j \in J} z_j \geq 1 \]
\[\sum_{i \in I} y_{ij} \leq n_j, \forall j \in J \]
\[\sum_{i \in I} a_i y_{ij} \leq n_j \bar{x}_j, \forall j \in J \]
\[\bar{x}_j \in \mathbb{R}^d, n_j \in \mathbb{R}, z_j \in \{0, 1\}, \forall j \in J \]
\[y_{ij} \in \{0, 1\}, \forall i \in I, \forall j \in J \]
\[\lambda_j \geq 0, \forall j \in J \]

Considering, \(Z^*(gCCCP)\) and \(Z^*(RL-gCCCP)\) the optimal solution for the (gCCCP) and (RL-gCCCP) models respectively. It can be stated that \(Z^*(gCCCP) \geq Z^*(RL-gCCCP)\). Than any lower bound obtained from (RL-gCCCP) greater than a known upper bound obtained from a feasible solution of (gCCCP), may be considered as invalid.

The (RL-CCCP) model incorporates in the objective function the value of the multiple knapsack constraints. Applying a resolution procedure over the (RL-CCCP), mitigates the intrinsic complexity of the (gCCCP) version.

The (RL-CCCP1) model was applied to ARGONNE Labs site, using both FilMINT and MINOS solvers available versions. The classic TA instances were investigated, because the number of feasible solutions is very high, turning the solution process infeasible by combinatorial B&B methods. For these instances, it was considered the fixed cost equal to zero. In this case, the model has to be rewritten as:

\[(RL - gCCCP1)\text{Minimize} \sum_{i \in I} \sum_{j \in J} \|a_i - \bar{x}_j\| y_{ij} + \sum_{j \in J} \lambda_j \sum_{i \in I} q_i y_{ij} - Q z_j \]
\[\text{such that :} \sum_{j \in J} y_{ij} = 1, \forall i \in I \]
\[\sum_{i \in I} y_{ij} \leq n_j, \forall j \in J \]
\[\sum_{i \in I} a_i y_{ij} \leq n_j \bar{x}_j, \forall j \in J \]
\[\bar{x}_j \in \mathbb{R}^d, n_j \in \mathbb{R}, \forall j \in J \]
\[y_{ij} \in \{0, 1\}, \forall i \in I, \forall j \in J \]
\[\lambda_j \geq 0, \forall j \in J \]
Table 1. Characteristics of the TA and V## benchmark instances

<table>
<thead>
<tr>
<th>Name</th>
<th>n</th>
<th>p</th>
<th>Q</th>
<th>w_Avg</th>
<th>w_Dev</th>
<th>F</th>
</tr>
</thead>
<tbody>
<tr>
<td>TA25</td>
<td>25</td>
<td>5</td>
<td>6</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TA50</td>
<td>50</td>
<td>5</td>
<td>11</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TA60</td>
<td>60</td>
<td>5</td>
<td>13</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TA70</td>
<td>70</td>
<td>5</td>
<td>17</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TA80</td>
<td>80</td>
<td>7</td>
<td>12</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TA90</td>
<td>90</td>
<td>4</td>
<td>23</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>TA100</td>
<td>100</td>
<td>6</td>
<td>17</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>v10a30Z1</td>
<td>10</td>
<td>?</td>
<td>15</td>
<td>4.684</td>
<td>2.276</td>
<td>1000</td>
</tr>
<tr>
<td>v10a30Z2</td>
<td>10</td>
<td>?</td>
<td>15</td>
<td>4.512</td>
<td>2.270</td>
<td>1000</td>
</tr>
<tr>
<td>v15a45Z1</td>
<td>15</td>
<td>?</td>
<td>15</td>
<td>4.339</td>
<td>2.940</td>
<td>1000</td>
</tr>
<tr>
<td>v15a45Z2</td>
<td>15</td>
<td>?</td>
<td>15</td>
<td>5.527</td>
<td>2.232</td>
<td>1000</td>
</tr>
<tr>
<td>v20a60Z1</td>
<td>20</td>
<td>?</td>
<td>15</td>
<td>5.711</td>
<td>1.723</td>
<td>1000</td>
</tr>
<tr>
<td>v20a60Z2</td>
<td>20</td>
<td>?</td>
<td>15</td>
<td>5.718</td>
<td>2.829</td>
<td>1000</td>
</tr>
<tr>
<td>v25a75Z3</td>
<td>25</td>
<td>?</td>
<td>15</td>
<td>5.795</td>
<td>2.631</td>
<td>1000</td>
</tr>
<tr>
<td>v25a75Z4</td>
<td>25</td>
<td>?</td>
<td>15</td>
<td>5.390</td>
<td>2.406</td>
<td>1000</td>
</tr>
<tr>
<td>v30a90Z1</td>
<td>30</td>
<td>?</td>
<td>15</td>
<td>5.458</td>
<td>2.161</td>
<td>1000</td>
</tr>
<tr>
<td>v30a90Z4</td>
<td>30</td>
<td>?</td>
<td>15</td>
<td>6.055</td>
<td>2.401</td>
<td>1000</td>
</tr>
<tr>
<td>v30a47B1</td>
<td>30</td>
<td>?</td>
<td>450</td>
<td>83.233</td>
<td>48.978</td>
<td>2000</td>
</tr>
<tr>
<td>v30a47O1</td>
<td>30</td>
<td>?</td>
<td>512</td>
<td>83.233</td>
<td>48.978</td>
<td>2000</td>
</tr>
</tbody>
</table>

For the tests, the hardware specifications are as follows, once the used solvers are hosted at UW-Madison, [4]:

- neos-2 and neos-4 are Dell PowerEdge R410 servers with the following configuration:
  - CPU - 2x Intel Xeon X5660 @ 2.8GHz (12 cores total), HT Enabled
  - Memory - 64GB RAM
  - Disk - 2x 500GB SATA drives setup in RAID1
  - Network - 1Gb/s Ethernet

- neos-3 and neos-5 are Dell PowerEdge R420 servers with the following configuration:
  - CPU - 2x Intel Xeon E5-2430 @ 2.2GHz (12 cores total), HT Enabled
  - Memory - 64GB RAM
  - Disk - 2x 2TB SATA drives setup in RAID1
  - Network - 1Gb/s Ethernet

The TA instances here used were extracted from [1], their details is shown in table 1, where \(w_{\text{avg}}\) is the weight average per individual, \(w_{\text{Dev}}\) is the standard deviation of the individuals demand per instance. The results can be seen in table 2. The columns cost refers to the objective cost of the model, Gap% is the GAP to the best upper bound known (best-known cost) for the problem at the related instance, and T(s) is the time spent in seconds by the solver in the platform to obtain the solution. The column UB inform the best upper bound (best-known cost) obtained for the instance by the literature.

The FilMINT followed the (RL-gCCCP1) model as proposed, but MINOS consider all the variables as continuous. In this case, the lower bound produced by MINOS may be expected to be lower than what is expected for FilMINT.

From table 2 the majority of the instances the cuts were generated without returning a feasible lower bound, once it is known a better upper bound by the heuristics methods already published. For all the instances MINOS returned an unfeasible upper bound, but
Table 2. Computational results for some instances with gCCCp lagrangean relaxation resolution.

<table>
<thead>
<tr>
<th>Name</th>
<th>Cost</th>
<th>Gap%</th>
<th>T(s)</th>
<th>Cost</th>
<th>Gap%</th>
<th>T(s)</th>
<th>UB</th>
</tr>
</thead>
<tbody>
<tr>
<td>TA25</td>
<td>1239.88</td>
<td>-0.923</td>
<td>6</td>
<td>1340.22</td>
<td>7.094</td>
<td>6</td>
<td>1251.44</td>
</tr>
<tr>
<td>TA50</td>
<td>4498.72</td>
<td>0.541</td>
<td>5</td>
<td>4492.34</td>
<td>0.398</td>
<td>5</td>
<td>4474.52</td>
</tr>
<tr>
<td>TA60</td>
<td>5329.81</td>
<td>-0.499</td>
<td>6</td>
<td>6087.48</td>
<td>0.136</td>
<td>5</td>
<td>5356.58</td>
</tr>
<tr>
<td>TA70</td>
<td>6228.13</td>
<td>-0.200</td>
<td>15</td>
<td>6322.87</td>
<td>1.317</td>
<td>5</td>
<td>6240.67</td>
</tr>
<tr>
<td>TA80</td>
<td>5706.46</td>
<td>3.462</td>
<td>20</td>
<td>5727.40</td>
<td>3.842</td>
<td>6</td>
<td>5515.46</td>
</tr>
<tr>
<td>TA90</td>
<td>9064.25</td>
<td>1.856</td>
<td>16</td>
<td>9085.89</td>
<td>2.099</td>
<td>15</td>
<td>8899.05</td>
</tr>
<tr>
<td>TA100</td>
<td>8384.29</td>
<td>3.483</td>
<td>15</td>
<td>8102.74</td>
<td>0.008</td>
<td>26</td>
<td>8102.04</td>
</tr>
</tbody>
</table>

FilMINT returned 3 feasible lower bounds, indicating that the actual upper bound values are possibly optimal for the instances: TA25, TA60 and TA70. All the negative gaps means the lower bound is below the upper bound, what is expected to be for this relaxation.

The surprisingly time spent to solve the TA instances, pointed to the good choice of the capacity constraints to be the lagrangean constraints. Although, both solvers showed to be not capable of obtaining confident lower bounds at all.

4 Direct Model Resolution

The time spent to obtain lagrangean solutions and the complexity related to the (RL-CCCp1) model, came to this research as a new question. Is it possible to solve to optimality the CCCp instances, by using the general CCCp model with these solvers?

Now, it was selected as benchmark instances TA and v###a##. Here we now include LINGO, from LINDO Systems, to evaluate the gCCCp model behavior in the environments. For TA it was maintained the fixed cost to zero for opening new clusters, but for v###a## instances all the parameters were considered. The instances v###a##z#, or v###a##B# or v###a##O#, were used by [16] for evaluating their combinatorial B&B version. In these instances the capacity is a float value, as the demand of each individual.

LINGO was used under a PC Intel Core 2 Duo T5555-1.83GHz with 3Gbytes of RAM. The ARGONNE Labs platform was available during all the investigation process, and the traffic highly varied during the period of the evaluations. Here it is included a new solver from ARGONNE (Lancelot), because of the curiosity about its performance.

FilMINT and LINGO are the only solvers that consider the model as it is, i.e with integer, binary and continuos variables. The other solvers, Lancelot and MINOS, their results of the direct model returns continuos variables. Then for Lancelot and MINOS it is considered the results as relaxations of the direct model.

It can be seen in table 3 that most v###a## instances were not solved by the ARGONNE solvers. LINGO returns there is no feasible solution to TA25 and v20a60z2. The ARGONNE FilMINT and MINOS solvers tilted as optimal solutions all their results. Once the upper bound is known, it is not possible to trust in their optimality criterion for this problem. LINGO only says as optimal instance v15a45z1, with a Gap for the true proved optimal of 0.63%. Lancelot solver returns two new upper bounds to TA50 and TA60 that cannot be trusted, and LINGO returns four new upper bounds to v###a## instances. In over-
all, it is possible to trust in the results produced by LINGO and FilMINT (desconsidering its optimality proof).

The LINGO solver remains robust in all tested cases, but the time spent to solve an instance is still very high. The time to obtain a local optimal in LINGO is reasonable for all evaluated instances, but it is not here reported because it is not returned by reports of the solver. LINGO returns new upper bounds for the instances v30. Special attention may be given for v30a37O1, where the reduced gap% reached 16.47%.

Using table 3 it can be seen that TA25, TA60 and TA100 can be stated as optimal values. For TA50 LINGO returns the optimal value, agreed closely with FilMINT, but Lancelot returns a new lower bound for this instance as for TA60. Trusting in Lancelot is a problem. It does not return confident values for the set of instances investigated in this research.

5 Conclusions

This study considered the possibility to solve different kind of instances of the CCCP using the state-of-the-art mathematical programming solvers from NEOS ARGONNE and LINDO Systems. The instances proved to be very difficult to be solved and to obtain the optimality. False optimal solutions were titled by the solvers and also false lower bounds were achieved. False optimality signature is a severe error.

Although the negative results, the study proved the optimality of 5 TA instances, and obtained new lower bounds for 4 of them.

It is important to say that the CCCP is a very difficult problem. The solvers are in fact very good, but they can fail in problems of great complexity like CCCP, [13].

The results indicate that the users may take great attention with the responses given by these solvers, to complex non-linear mixed-binary problems like CCCP.
<table>
<thead>
<tr>
<th>Nome</th>
<th>Custo</th>
<th>Gap%</th>
<th>T(s)</th>
<th>Custo</th>
<th>Gap%</th>
<th>T(s)</th>
<th>Custo</th>
<th>Gap%</th>
<th>T(s)</th>
<th>Custo</th>
<th>Gap%</th>
<th>T(s)</th>
<th>UB</th>
</tr>
</thead>
<tbody>
<tr>
<td>TA25</td>
<td>1528.24</td>
<td>22.119</td>
<td>15</td>
<td>1343.33</td>
<td>7.359</td>
<td>10</td>
<td>1251.45</td>
<td>0.000</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td>1251.44</td>
</tr>
<tr>
<td>TA50</td>
<td>4326.85</td>
<td>-3.300</td>
<td>21</td>
<td>4492.34</td>
<td>0.398</td>
<td>15</td>
<td>4498.72</td>
<td>0.540</td>
<td>2</td>
<td>4478.83</td>
<td>0.096</td>
<td>20580.20</td>
<td>4474.52</td>
</tr>
<tr>
<td>TA60</td>
<td>4999.51</td>
<td>-6.666</td>
<td>21</td>
<td>6106.59</td>
<td>14.002</td>
<td>6</td>
<td>5356.58</td>
<td>0.000</td>
<td>2</td>
<td>5370.047</td>
<td>0.251</td>
<td>4640</td>
<td>5356.58</td>
</tr>
<tr>
<td>TA70</td>
<td>9709.25</td>
<td>55.580</td>
<td>41</td>
<td>6352.35</td>
<td>1.790</td>
<td>5</td>
<td>6250.61</td>
<td>0.159</td>
<td>2</td>
<td>6250.61</td>
<td>0.159</td>
<td>3600</td>
<td>6240.67</td>
</tr>
<tr>
<td>TA80</td>
<td>7163.52</td>
<td>-29.881</td>
<td>60</td>
<td>5755.82</td>
<td>4.358</td>
<td>6</td>
<td>5736.36</td>
<td>4.005</td>
<td>5</td>
<td>5746.51</td>
<td>4.189</td>
<td>3600</td>
<td>5515.46</td>
</tr>
<tr>
<td>TA90</td>
<td>18809.40</td>
<td>111.364</td>
<td>20</td>
<td>9103.22</td>
<td>2.294</td>
<td>26</td>
<td>9069.85</td>
<td>1.919</td>
<td>4</td>
<td>9069.85</td>
<td>1.919</td>
<td>3600</td>
<td>8899.05</td>
</tr>
<tr>
<td>TA100</td>
<td>12928.71</td>
<td>59.574</td>
<td>15</td>
<td>8122.68</td>
<td>0.255</td>
<td>5</td>
<td>8387.42</td>
<td>3.522</td>
<td>2</td>
<td>8102.044</td>
<td>0.000</td>
<td>3600</td>
<td>8102.04</td>
</tr>
<tr>
<td>v10a30z1</td>
<td>3912.64</td>
<td>-18.87</td>
<td>25</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4822.43</td>
</tr>
<tr>
<td>v10a30z2</td>
<td>5333.13</td>
<td>-3.011</td>
<td>20</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>5498.74</td>
</tr>
<tr>
<td>v15a45z1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>6261.89 (*)</td>
</tr>
<tr>
<td>v15a45z2</td>
<td>8550.81</td>
<td>9.23</td>
<td>31</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>7828.17</td>
</tr>
<tr>
<td>v20a60z1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>10341.28</td>
</tr>
<tr>
<td>v20a60z2</td>
<td>12041.23</td>
<td>7.91</td>
<td>40</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>4373</td>
</tr>
<tr>
<td>v25a75z3</td>
<td>14235.30</td>
<td>5.76</td>
<td>60</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>13956.01</td>
</tr>
<tr>
<td>v25a75z4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>13987.64</td>
</tr>
<tr>
<td>v30a47B1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>12109.43</td>
</tr>
<tr>
<td>v30a37O1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>12138.31</td>
</tr>
<tr>
<td>v30a90z1</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>12170.09</td>
</tr>
<tr>
<td>v30a90z4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>12188.31</td>
</tr>
<tr>
<td>v30a90z4</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>12188.31</td>
</tr>
</tbody>
</table>

Table 3. Computational results with gCCCP direct model resolution.
References


