A reformulation strategy for mixed-integer linear bi-level programming problems

Sergio Medina-González[a,b], Lazaros G. Papageorgiou[a] and Vivek Dua[a] *

[a] *Centre for Process Systems Engineering, Department of Chemical Engineering, UCL (University College London), Torrington Place, London WC1E 7JE, UK*

[b] *Davidson School of Chemical Engineering, Purdue University, West Lafayette, IN 47907, United States*

*Corresponding author e-mail address: v.dua@ucl.ac.uk (V. Dua)

Abstract:

Bi-level programming has been used widely to model interactions between hierarchical decision-making problems, and their solution is challenging, especially when the lower-level problem contains discrete decisions. The solution of such mixed-integer linear bi-level problems typically need decomposition, approximation or heuristic-based strategies which either require high computational effort or cannot guarantee a global optimal solution. To overcome these issues, this paper proposes a two-step reformulation strategy in which the first part consists of reformulating the inner mixed-integer problem into a nonlinear one, while in the second step the well-known Karush-Kuhn-Tucker conditions for the nonlinear problem are formulated. This results in a mixed-integer nonlinear problem that can be solved with a global optimiser. The computational and numerical benefits of the proposed reformulation strategy are demonstrated by solving five examples from the literature.

Keywords:
Mixed Integer bi-level programming, Lower-level discrete variables, nonlinear reformulation.

1 Introduction

Optimisation problems involving two decision-makers in a hierarchical scheme are typically formulated as bi-level programming problems. A common notation defines the upper-level decision-maker as the *"leader"* who has to solve its associated optimisation problem sharing some constraints with the lower-level optimisation problem solved by a second decision-maker (also known as *"follower"*). Fig. 1 illustrates the leader-follower interaction. This type of problem formulation is particularly useful to address a wide variety of multi-level decision-making problems such as chemical process design (Clark and Westerberg, 1990), energy network systems (Motto et al., 2005), parameter estimation (Mitsos et al., 2009), and supply chain management (Ryu et al., 2004; Kuo and Han, 2011; Calvete et al., 2010; Gupta and Maranas, 2000; Roghanian et al., 2007; Gao et al., 2011; Garcia-Herreros et al., 2016), among others.

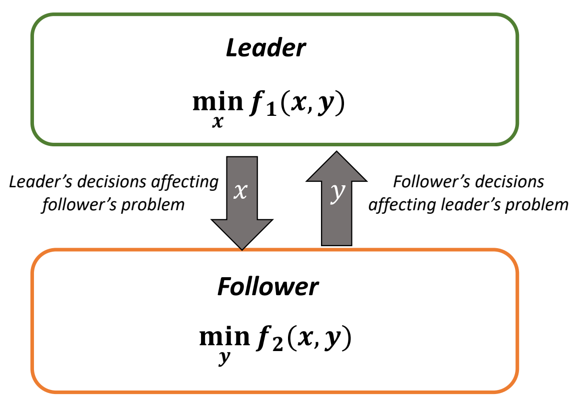

Figure 1: Leader-follower interaction scheme

A strategy used to solve bi-level problems consists of using the well-known Karush-Kuhn-Tucker (KKT) optimality conditions (Fortuny-Amat and McCarl, 1981; Bard and Falk, 1982; Bialas and Karwan, 1982; Motto et al., 2005; Garces et al., 2009) to reformulate the problem into a single-level problem. Essentially, KKT conditions depend on extracting the gradient information of the lower-level problem, thus, the inner problem should be strictly continuous (Marcotte and Savard, 2009; Sinha et al., 2018). In order to by-pass this limitation several approaches have been proposed. Particularly, branch-and-bound inspired iterative methods (Gumus and Floudas, 2005; Mitsos, 2010; Moore and Bard, 1990; Caramia and Mari, 2016; Xu and Wang, 2014; Kleniati and Adjiman, 2015), decomposition approaches (Saharidis and Ierapetritou, 2009) and genetic-inspired heuristics were proposed to guide the branching process and solve the problem faster (Nishizaki and Sakawa, 2005; Hecheng and Yuping, 2008; Arroyo and Fernandez, 2009). Alternatively, data-driven approaches have been studied in the past. Recently (Beykal et al., 2020) proposed the DOMINO-framework (that states from Data-driven Optimization of bi-level Mixed-Integer NOnlinear Problems) was proposed in which the bi-level optimization problems are approximated as single-level optimization problems by collecting samples of the upper-level objective and solving the lower-level problem to global optimality at those sampling points. Similarly, multi-parametric (mp) methods (Avraamidou and Pistikopoulos, 2017, 2018, 2019; Shokry et al., 2017) and multi-step reformulation/decomposition strategies Zeng and An (2014) have been used to solve bi-level mixed-integer programming (MIP) problems. Particularly, the inner problem is reformulated and decomposed into a subset of feasible LP problems. Yue and You (2016) have proposed a framework that combines a reformulation and a KKT-based cut strategy to generate feasible bounds.

Despite of the large number of alternatives, all of these either approximate the optimal solution or require a high pre/post-processing calculation effort. Therefore, solving a bi-level problem still represents a challenging task (Hansen et al., 1992; Deng, 1992; Colson et al., 2005). Almost two decades ago, Gumus and Floudas (2005) extended the polyhedral reformulation/linealization scheme originally proposed by Sherali and Adams (1990) to propose a generalized global optimization framework. Essentially the technique is based on the formulation of the mixed-integer inner problem as continuous and solving the resulting bi-level optimization problem using a tailor-made global optimization framework (Gumus and Floudas, 2005). Nevertheless, the complex reformulation strategy (that use of a large number of polynomial terms), the use of small examples and a lack of computational information hinders the evaluation of the time effectiveness of this strategy to be applied in large scale problems. Another more recent alternative was proposed by Fischetti et al. (2017) in which a branch-and-cut exact solution method considering several new classes of inequalities was proposed. Nevertheless, this approach is limited to address problems in which the continuous leader variables (if any) do not appear in the follower problem and the feasible set is a bounded polyhedron. More recently, Yue et al. (2019) combined a projection-based reformulation and a decomposition strategy in a single global optimisation framework. However, due to the iterative nature of the proposed strategy along with the inherent complexity in solving MIP-MIP bilevel problems, the computational performance hinders its application for large scale problems and particularly those with a relatively large number of low-level variables. Finally, Poirion et al. (2020) proposed a polyhedron-based approach to reformulate a binary mixed integer problem into a binary lineal problem. Despite its proven effectiveness, this approach is limited to consider the Type II (mixed-)integer bilevel problems (i.e. purely integer) with an additional assumptions that the inner binary variable is present in exactly one upper level constraint hindering its use for a variety of applications.

In this paper, we implement a reformulation strategy similar to Gumus and Floudas (2005) to solve MIP-MIP bilevel problems in a more straightforward fashion. The bilevel problem addressed in this paper involves mixed-integer linear programs at outer as well as inner levels. The work of Gumus and Floudas (2005) used convex-hull representation for the inner problem whereas we use a reformulation that converts integer variables to continuous variables, followed by formulating the KKT conditions. In section 2 a detailed description of the mathematical model reformulation is presented while the rest of the paper is organized as follows: Section 3 presents the results obtained for a set of examples while in section 4 the main conclusions are summarized along with the future research directions.

2 Mathematical reformulation

In this section the proposed mathematical reformulation is described in detail. Consider the mixed-integer linear bi-level programming problem with discrete variables in both levels (model B-MILP):

$$\min_x F(x,y) = c_1^T x + d_1^T y \qquad (B\text{-}MILP)$$

s.t.

$$g^{up}(x,y) = A_1^{ineq}x + B_1^{ineq}y - b_1^{ineq} \leq 0 \qquad (1)$$

$$h^{up}(x,y) = A_1^{eq}x + B_1^{eq}y - b_1^{eq} = 0 \qquad (2)$$

$$\min_y f(x,y) = c_2^T x + d_2^T y \qquad (3)$$

s.t.

$$g^{lo}(x,y) = A_2^{ineq}x + B_2^{ineq}y - b_2^{ineq} \leq 0 \qquad (4)$$

$$h^{lo}(x,y) = A_2^{eq}x + B_2^{eq}y - b_2^{eq} = 0 \qquad (5)$$

$$x_1, ..., x_i \in \mathbb{R}, \qquad y_1, ..., y_j \in \mathbb{R}, \qquad (6)$$

$$x_{i+1}, ..., x_{n1} \in \{0,1\}^l, \qquad y_{j+1}, ..., y_{n2} \in \{0,1\}^m \qquad (7)$$

where x and y are the mixed-integer variables for the upper and lower-level problem, respectively. $g(x,y)$ and $h(x,y)$ represent inequality and equality constraints respectively. c_1^T, d_1^T, c_2^T, d_2^T are constant coefficient vectors and $A_1^{ineq,eq}$, $A_2^{ineq,eq}$, $B_1^{ineq,eq}$, $B_2^{ineq,eq}$, $b_1^{ineq,eq}$ and $b_2^{ineq,eq}$ are constant coefficient matrices and vectors. Finally, Eq.6 corresponds to the continuous variable decisions, while eq. 7 the discrete ones.

A widely used strategy to address a bi-level problem consists of transforming it into a single-level problem employing the KKT conditions. However, these conditions require in lower-level problem involving only continuous variables which hinders its direct application for models such as B-MILP. In this paper a reformulation of the lower-level MILP problem into an NLP is proposed. Such a reformulation consists of substituting the set of integer variables Y with a set of continuous ones Y_c and include the following additional equality constraint into the model B-MILP:

$$y_c(y_c - 1) = 0 \tag{8}$$

Note, Y_c should substitute the integer variables in both levels only if the lower-level binary variables appear also in the upper-level part of the problem. It is worth to mention that if the integer variables are not binaries, the reformulation may be still employed by increasing the number of terms. For example for $Y = \{0, 1, 2\}$ the reformulation term is $y_c(y_c - 1)(y_c - 2) = 0$. By applying such a reformulation the new model (*B-MILP2*) is set as follows:

$$\min_x F(x, y) = c_1^T x + d_1^T y \tag{B-MILP2}$$

s.t.

$$g^{up}(x, y) = A_1^{ineq} x + B_1^{ineq} y - b_1^{ineq} \leq 0 \tag{9}$$

$$h^{up}(x, y) = A_1^{eq} x + B_1^{eq} y - b_1^{eq} = 0 \tag{10}$$

$$\min_{y,yc} f(x, y, yc) = c_2^T x + d_2^T (y, yc) \tag{11}$$

s.t.

$$g^{lo}(x, y, yc) = A_2^{ineq} x + B_2^{ineq} (y, yc) - b_2^{ineq} \leq 0 \tag{12}$$

$$h_1^{lo}(x, y, yc) = A_2^{eq} x + B_2^{eq} (y, yc) - b_2^{eq} = 0 \tag{13}$$

$$h_2^{lo}(yc) = yc(yc - 1) = 0 \tag{14}$$

$$x_1, ..., x_i \in \mathbb{R}, \qquad y_1, ..., y_j \in \mathbb{R}, \qquad yc_{j+1}, ..., yc_{n2} \in \mathbb{R} \tag{15}$$

$$x_{i+1}, ..., x_{n1} \in \{0, 1\}^m, \tag{16}$$

By introducing the new vector y_c, the original MILP-MILP problem is transformed into an MILP-NLP one, where lower level problem in B-MILP has been replaced by an NLP in B-MILP2. At this point the KKT conditions can be derived for the lower part of the model (eqs. (11)-(14)). The necessary conditions, as well as the single level problem formulation are introduced as follows. The Lagrangian for the lower-level NLP problem in B-MILP2 is given by:

$$\mathcal{L}(x, y, yc) = c_2^T x + d_2^T(y, yc) + \lambda_1^T g^{lo}(x, y, yc) + \mu_1^T h_1^{lo}(x, y, yc) + \mu_2^T h_2^{lo}(yc) \quad (17)$$

Stationary conditions

$$\begin{aligned}\nabla\mathcal{L}(x^*, yc^*, \lambda^*, \mu^*) = \nabla_{y,yc} f(x, y, yc) + \lambda^T \nabla_{y,yc} g^{lo}(x, y, yc) + \mu_1^T \nabla_{y,yc} h_1^{lo}(x, y, yc) \\ + \mu_2^T \nabla_{yc} h_2^{lo}(yc) = 0\end{aligned} \quad (18)$$

Primal feasibility

$$g^{lo}(x, y, yc) \leq 0 \quad (19)$$

$$h_1^{lo}(x, y, yc) = 0 \quad (20)$$

$$h_2^{lo}(yc) = 0 \quad (21)$$

Complementary slackness

$$\lambda_p g_p^{lo}(x^*, yc^*) = 0 \qquad \forall\ p \quad (22)$$

Dual feasibility

$$\lambda_p \geq 0 \qquad \forall\ p \quad (23)$$

where λ and μ represents the Lagrangian multipliers for equality and inequality constraints respectively. By substituting constrains (12)-(14) by (18)-(23) in the lower-level problem the single-level MINLP is obtained as shown in model *B-MINLP*.

$$
\begin{aligned}
(B-MINLP) \qquad & \min_x F(x,y) = c_1^T x + d_1^T y \\
& s.t. \\
& \quad A_1^{ineq} x + B_1^{ineq} y \leq b_1^{ineq} \\
& \quad A_1^{eq} x + B_1^{eq} y = b_1^{eq} \\
& \quad \nabla_{y,yc}(c_2^T x + d_2^T (y,yc)) + \lambda^T \nabla_{y,yc}(A_2^{ineq} x + B_2^{ineq}(y,yc) - b_2^{ineq})... \\
& \qquad ... + \mu_1^T \nabla_{y,yc}(A_2^{eq} x + B_2^{eq}(y,yc) - b_2^{eq}) + \mu_2^T \nabla_{yc}(yc(yc-1)) = 0 \\
& \quad A_2^{ineq} x + B_2^{ineq}(y,yc) \leq b_2^{ineq} \\
& \quad A_2^{eq} x + B_2^{eq}(y,yc) = b_2^{eq} \\
& \quad yc_q(yc_q - 1) = 0 \qquad \forall\ q \\
& \quad \lambda_p (A_2^{ineq} x + B_2^{ineq}(y,yc) - b_2^{ineq})_p = 0 \qquad \forall\ p \\
& \quad \lambda_p \geq 0 \qquad \forall\ p \\
& x_1, ..., x_i \in \mathbb{R}, \qquad y_1, ..., y_j \in \mathbb{R}, \qquad yc_{j+1}, ..., yc_{n2} \in \mathbb{R} \\
& x_{i+1}, ..., x_{n1} \in \{0,1\}
\end{aligned}
$$

Note that the KKT conditions for the lower-level problem are local optimality conditions and hence global optimality can not be guaranteed. In this work, we compute the global optima, for five examples, by iteratively introducing cuts in the B-MINLP problem such that the optimal solution in the current iteration is better than the optimal solution in the previous iteration. This is achieved by including the cut: $f^k < f^{k-1}$, where f is lower-level objective function and k is the iteration counter. The iterations are carried out until B-MINLP is infeasible or a maximum number of pre-specified iterations is reached.

After solving the *B-MINLP* model the regularity conditions should be checked. In the next section the capabilities of the proposed reformulation strategy was tested by applying it to five different examples taken from the literature.

The summary of the reformulation algorithm is presented in table 1

Table 1: MILP-MILP reformulation algorithm

Step 1. Reformulate the lower-level problem from the original BMILP as an NLP using eq.8
Step 2. Write the KKT conditions for the resulting lower-level NLP.
Step 3. Solve the resulting B-MINLP problem iteratively by introducing cuts of the form $f^k < f^{k-1}$ and using a global optimisation solver

3 Examples

Five different examples were used in order to illustrate the capabilities of the presented reformulation. The first three examples consists of numerical problems including either integer and/or binary variables. The Fourth one consist of a two-level production-distribution problem and the last example describes a two-step sequential production system in which each step is controlled by a different decision-maker. The particular details for the used examples are presented in their respective subsection, however all the MINLP problems were implemented in GAMS 24.7 and solved using BARON18.11.12 to a global optimality. All the computations were done on a Dell workstation with Intel®Xeon®CPU E5-1650 v3@3.50 GHz and 32.00 GB RAM. For comparison purposes, Table 7 in appendix A displays the reformulated model statistics for all the examples used.

3.1 1^{st} Example

Consider the following mixed-integer nonlinear bilevel optimisation problem (*B-MILP-E1*) which was previously solved by Gumus and Floudas (2005) using a vertex polyheral convex hull representation and solving the resulting nonlinear bilevel optimization problem by a novel deterministic global optimization framework

$$\min_x (x-2)^2 + (y-2)^2 \tag{B-MILP-E1}$$

s.t.

$$\min_y y^2 \tag{24}$$

s.t.

$$-2x - 2y \leq -5 \tag{25}$$

$$x - y \leq 1 \tag{26}$$

$$3x + 2y \leq 8 \tag{27}$$

$$x \in \mathbb{R}; \qquad y \in \{0, 1, 2\}$$

Eqs. (24-3.1) represent the lower level constraints. The binary reformulation was applied here substituting y by the vector z_c, and adding eq.(28).

$$y_c(y_c - 1) = 0 \tag{28}$$

At this point the KKT conditions were derived for the lower-level problem (eqs.(24)-28) as presented next:

$$\begin{aligned} \mathcal{L}(yc) = (yc^2 + \lambda_1(-2x - 2yc + 5) + \lambda_2(x - yc - 1) \\ + \lambda_3(3x + 2yc - 8) + \mu_1(yc(yc-1)(yc-2)) \end{aligned} \tag{29}$$

Stationary conditions

$$\nabla_{yc}\mathcal{L}(yc, \lambda) = 2yc - 2\lambda_1 - \lambda_2 + 2\lambda_3 + \mu_1(3yc^2 - 2yc) = 0 \tag{30}$$

Complementary slackness

$$\lambda_1(-2x - 2yc + 5) = 0 \tag{31}$$

$$\lambda_2(x - yc - 1) = 0 \tag{32}$$

$$\lambda_3(3x + 2yc - 8) = 0 \tag{33}$$

Dual feasibility

$$\lambda_1 \geq 0 \tag{34}$$

$$\lambda_2 \geq 0 \tag{35}$$

$$\lambda_3 \geq 0 \tag{36}$$

The single-level model (*B-MINLP-E1*) including the associated inner problem reformulation is presented as follows:

$$
\begin{aligned}
(B-MINLP-E1) \qquad & \min_{x} (x-2)^2 + (yc-2)^2 \\
& s.t. \\
& x \geq 0 \\
\text{Primal feasibility,} & \begin{cases} -2x - 2yc + 5 \leq 0 \\ x - yc - 1 \leq 0 \\ 3x + 2yc - 8 \leq 0 \end{cases} \\
\text{Stationary conditions,} & \begin{cases} 2yc - 2\lambda_1 - \lambda_2 + 2\lambda_3 + \mu_1(3yc_1^2 - 2yc) = 0 \end{cases} \\
\text{Complementary slackness,} & \begin{cases} \lambda_1(-2x - 2yc + 5) = 0 \\ \lambda_2(x - yc - 1) = 0 \\ \lambda_3(3x + 2yc - 8) = 0 \end{cases} \\
\text{Dual feasibility,} & \begin{cases} \lambda_j \geq 0 \qquad \forall\ j = 1, 2, 3 \end{cases} \\
\text{Binary reformulation,} & \begin{cases} yc(yc-1)(yc-2) = 0 \end{cases} \\
& x \in \mathbb{R}_+; \qquad yc \in \mathbb{R}_+
\end{aligned}
$$

The model *B-MINLP-E1* consists of 10 equations, 8 single variables and 0 discrete variables and its solution takes around 0.840 seconds to achieve the global optimality accounting for all the cuts, including the last infeasible one. The optimal solution is obtained at (x^*, y^*)=(1.333, 2) with $F^* = 0.444$ which achieves the same objective function as obtained in previous studies (Gumus and Floudas, 2005).

3.2 2^{nd} Example

The following mixed-integer bilevel optimisation problem (*B-MILP-E2*) was taken from Yue et al. (2019) using a projection-based reformulation and decomposition strategy.

$$
\begin{aligned}
(B-MILP-E2) \qquad & \min_x (-x) - 10y \\
& \;\; s.t. \\
& \qquad \min_y y \\
& \;\; s.t. \\
& \qquad -25x - 20y \leq 30 \\
& \qquad x + 2y \leq 10 \\
& \qquad 2x - y \leq 15 \\
& \qquad 2x - 10y \leq -15 \\
& \qquad x \in \{0, 1, 2, 3, 4, 5, 6, 7, 8\}; \qquad y \in \{0, 1, 2, 4\}
\end{aligned}
$$

The KKT conditions are derived for the lower-level problem leading to the single-level model (*B-MINLP-E2*) while the full-details of the KKT conditions are presented in the appendix B.1.

$$(B-MINLP-E2) \qquad \min_{x}(-x)-10yc$$

$$s.t.$$

$$\text{Primal feasibility,}\begin{cases} -25x-20yc-30 \leq 0 \\ x+2yc-10 \leq 0 \\ 2x-yc-15 \leq 0 \\ -2x-10yc+15 \leq 0 \end{cases}$$

$$\text{Stationary conditions,}\begin{cases} 1+20\lambda_1+2\lambda_2-1\lambda_3-10\lambda_4+.. \\ ..+\mu_1(5yc^4-40yc^3+105yc^2-100yc+24)=0 \end{cases}$$

$$\text{Complementary slackness,}\begin{cases} \lambda_1(-25x+20yc-30)=0 \\ \lambda_2(x+2yc-10)=0 \\ \lambda_3(2x-yc-15)=0 \\ \lambda_4(-2x-10yc+15)=0 \end{cases}$$

$$\text{Dual feasibility,}\begin{cases} \lambda_j \geq 0 \qquad \forall\ j=1,2,3,4 \end{cases}$$

$$\text{Binary reformulation,}\begin{cases} yc(yc-1)(yc-2)(yc-3)(yc-4)=0 \end{cases}$$

$$x \in \{0,1,2,3,4,5,6,7,8\}; \qquad yc \in \mathbb{R}_+$$

Model *B-MINLP-E2* consist of 12 equations, 9 continuous variables and 1 discrete variable. Remarkably, three different optimal solutions were identified ((x^*, y^*)=$(2,4)$ with $F^*=-42$; (x^*, y^*)=$(2,3)$ with $F^*=-32$;(x^*, y^*)=$(2,2)$ with $F^*=-22$) which matches with those obtained in previous studies (Yue et al., 2019; Moore and Bard, 1990). The solution of *B-MINLP-E2* takes less than 0.83 seconds at each iteration (2.838 second in total).

3.2.1 Computational comparison

In order to perform a further comparison in the computational time required to solve the problems using the proposed reformulation and the projection-based reformulation another example from (Yue et al., 2019) was solved for different inner problems sizes/instances ranging from 36/36-144/144 in continuous/binary variables. This problem consist of a hierarchical planning problem formulated as an MIBLP problem which involves continuous and binary variables in both the upper- and lower-level programs. Note that due to the large number of binary variables the details of the KKT conditions were not included in this paper. The associated mathematical model formulation is presented as follows while more details can be found in (Yue et al., 2019).

$$
\begin{aligned}
\min z_1 = & \sum_i f_i Y_i + \sum_{i,j} g_{i,j} Z_{i,j} + \sum_i p_i (Cap_i - \sum_j d_j a_{i,j} X_{i,j}) \\
& s.t. \\
& \quad \sum_{i,j} d_j + e_{i,j} X_{i,j} \leq q \\
& \quad Cap_i \leq c_i^U \forall i \\
& \quad Y_i \in \{1,0\}, \qquad Cap_i \in \mathbb{R}_+ \\
& \quad \min z_2 = \sum_i w_i (\sum_j d_j a_{i,j} X_{i,j}) + \sum_{i,j} (s_{i,j} Z_{i,j} + d_j r_{i,j} X_{i,j}) \\
& \quad s.t. \\
& \qquad \sum_i X_{i,j} = 1 \qquad \forall j \\
& \qquad \sum_j d_j a_{i,j} X_{i,j} \leq Cap_i \qquad \forall i \\
& \qquad \sum_j X_{i,j} \leq n Y_i \qquad \forall i \\
& \qquad X_{i,j} \leq Z_{i,j} \qquad \forall i, j \\
& \quad X_{i,j} \in \mathbb{R}_+; \qquad Z_{i,j} \in \{1,0\}
\end{aligned}
$$

Where Y_i and $Z_{i,j}$ are the binary variables used to select the use or not of the plant i and production line for product j in plant i respectively. Similarly Cap_i and $X_{i,j}$ represent the continuous variables for the production capacity and the fraction of demand satisfied by production line i, j respectively. The rest of the elements are process parameters which are randomly generated as proposed in (Yue et al., 2019) and are defined in table 2.

Table 2: Parameter definition

Parameters	
a_{ij}	Capacity consumption ratio for processing product j in plant i
c_i^u	Upper bound of production capacity in plant i
d_j	Customer demand of product j
e_{ij}	Resource factor for processing product j in plant i
f_i	Opening cost for plant i
g_{ij}	Fixed cost for opening production line j in plant i
p_i	oportunity cost for unused production capacity of plant i after it is opened
q	Resource availability
r_{ij}	Transportation cost for transferring product j from plant i to the principal firm
s_{ij}	Fixed operation cost for processing product j in plant i
w_i	Cost to use production capacity in plant i

We test the proposed algorithm on a total of 5 instances varying the number of plants (i) and products (j) as follows: (6,6), (6,8), (8,8), (8,10), (10,10), (10,12), and (12,12). The proposed approach takes 3.2-10.3 seconds in average to reach global optima whereas the projection-based reformulation proposed by Yue et al. (2019) reported average solution times of 3-981 seconds. As the problem parameters were randomly generated the problems solved using our approach and those reported by Yue et al. (2019) may not be exactly the same.

3.3 3^{rd} Example

This example consists of a mixed-integer linear programming problem at both levels (*B-MILP-E3*) which was previously solved by Avraamidou and Pistikopoulos (2019) using an mp-MILP approach.

$$
\begin{aligned}
(B-MILP-E3) \qquad & \min_{x_{1,2},y_3} 4x_1 - x_2 + x_3 + 5y_1 - 6y_3 \\
& s.t. \\
& \quad x_1 \leq 10 \\
& \quad x_2 \leq 10 \\
& \quad x_1 \geq -10 \\
& \quad x_2 \geq -10 \\
& \quad \min_{x_3,y_{1,2}} -x_1 + x_2 - 2x_3 - y_1 + 5y_2 + y_3 \\
& s.t. \\
& \quad 6.4x_1 + 7.2x_2 + 2.5x_3 \leq 11.5 \\
& \quad -8x_1 - 4.9x_2 - 3.2x_3 \leq 5 \\
& \quad 3.3x_1 + 4.1x_2 + 0.02x_3 + 4y_1 + 4.5y_2 + 0.5y_3 \leq 1 \\
& \quad x_1, x_2, x_3 \in \mathbb{R}; \qquad y_1, y_2, y_3 \in \{1, 0\}
\end{aligned}
$$

At this point the KKT conditions for the lower-level problem must be derived as presented in section 2. The complete derivation of the KKT conditions are presented in the appendix B.2 while the single-level model (*B-MINLP-E3*) including the associated inner problem reformulation is presented as follows:

$$
\begin{aligned}
(B-MINLP-E3) \qquad & \min_{x_{1,2},y_3} 4x_1 - x_2 + x_3 + 5y_1 - 6y_3 \\
& s.t. \\
& 10 \leq x_{1,2} \leq 10 \\
\text{Primal feasibility,} & \begin{cases} 6.4x_1 + 7.2x_2 + 2.5x_3 - 11.5 \leq 0 \\ -8x_1 - 4.9x_2 - 3.2x_3 - 5 \leq 0 \\ 3.3x_1 + 4.1x_2 + 0.02x_3 + 4yc_1 + 4.5yc_2 + 0.05y_3 - 1 \leq 0 \end{cases} \\
\text{Stationary conditions,} & \begin{cases} -2 + 2.5\lambda_1 - 3.2\lambda_2 + 0.02\lambda_3 = 0 \\ -1 + 4\lambda_3 + \mu_1(2yc_1 - 1) = 0 \\ 5 + 4.5\lambda_3 + \mu_2(2yc_2 - 1) = 0 \end{cases} \\
\text{Complementary slackness,} & \begin{cases} \lambda_1(6.4x_1 + 7.2x_2 + 2.5x_3 - 11.5) = 0 \\ \lambda_2(-8x_1 - 4.9x_2 - 3.2x_3 - 5) = 0 \\ \lambda_3(3.3x_1 + 4.1x_2 + 0.02x_3 + 4yc_1 + 4.5yc_2 + 0.05y_3 - 1) = 0 \end{cases} \\
\text{Dual feasibility,} & \begin{cases} \lambda_j \geq 0 \qquad \forall\ j = 1,2,3 \end{cases} \\
\text{Binary reformulation,} & \begin{cases} yc_1(yc_1 - 1) = 0 \\ yc_2(yc_2 - 1) = 0 \end{cases} \\
& x_1, x_2, x_3 \in \mathbb{R}_+; \qquad yc_1, yc_2 \in \mathbb{R}_+; \qquad y_3 \in \{1, 0\}
\end{aligned}
$$

3.3.1 Implementation and results

The *B-MINLP-E3* model consists of 21 equations, 16 single variables and 1 discrete variables. The model already includes the main parameters, and its solution takes 2.660 seconds to achieve the global optimality after 3 iterations. The optimal solution and its comparison with values reported in the literature are presented in Table 3.

Table 3: Single level optimal solution for different methodologies

	x_1	x_2	x_3	y_1	y_2	y_3	Objective Function
mp-MILP	-10	8.243	10.128	0	0	0	-38.115
Bi-level MINLP	-10	8.243	10.128	0	0	0	-38.115

Note that the proposed reformulation achieves the same objective function as obtained in the mp-MILP approach.

3.4 4^{th} Example

Consider the problem addressed by Avraamidou and Pistikopoulos (2017) consisting of a supply chain system operated by two different companies (Figure 2). One company produces one product through two production plants (p), while the other company distributes the product between two distribution centres (d) and three different costumers (c).

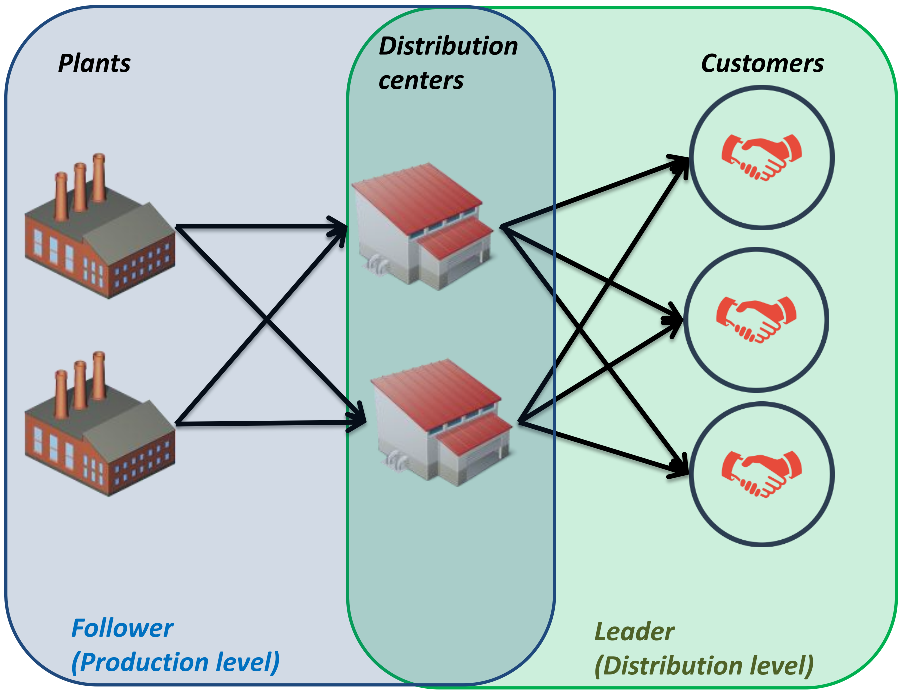

Figure 2: Schematic of the production-distribution planning problem case study

The distribution company aims to minimise the distributing and acquiring costs by deciding the existing routes connecting the centres to the customers, as well as the required inventory levels at each centre. The amount of product acquired by the distribution centres is used to calculate the production levels. Thus, the production company, after receiving the orders from each distribution centre, decides which processing plant will take that work load so as to minimize its production costs. Thus, the distribution problem can be considered as the leader which ignores the production plan that is defined by the follower seeking to minimise its own production cost.

The above problem can be formulated as an MILP-MILP bi-level problem as presented in model *B-MILP-E4*. The variables and parameters used are defined in Table 4:

Table 4: Variables and parameters definition

	Sets and indexes
P,p	Set of plants, plant index
D,d	Set of distribution centres, centres index
C,c	Set of customers, customers index
	Parameters
A_p	Maximum capacity of plant p
c^1_{dc}	Cost of route connecting centre d to customer c
c^2_{pd}	Cost of getting products from plant p to centre d
c^3_{pd}	Cost of manufacturing products from plant p for centre d
c^4_{pd}	Cost of route connecting plant p for centre d
b_c	Customer c demand
	Variables
s_{dc}	Amount of product sent from centre d to customer c
x_{pd}	Amount of product manufactured in plant p for centre d
	Binary variables
y_{dc}	Existence (or not) of route connecting centre d to customer c
z_{pd}	Existence of route connecting plant p for centre d

$$\min_{y_{dc},s_{dc}} \sum_{d,c} c^1_{dc} y_{dc} + \sum_{p,d} c^2_{pd} x_{pd} \qquad (B\text{-}MILP\text{-}E4)$$

s.t.

$$\sum_{d} s_{dc} \geq b_c \qquad \forall\ c \qquad (37)$$

$$s_{dc} \leq 35 y_{dc} \qquad \forall\ c,\ d \tag{38}$$

$$s_{dc} \geq 0 \qquad \forall\ c,\ d \tag{39}$$

$$\min_{x_{pd}, z_{pd}} \sum_{p,d} c^3_{pd} x_{pd} + c^4_{pd} z_{pd} \tag{40}$$

s.t.

$$\sum_{d} x_{pd} \leq A_p \qquad \forall\ p \tag{41}$$

$$\sum_{p} x_{pd} \geq \sum_{c} s_{dc} \qquad \forall\ d \tag{42}$$

$$x_{pd} \leq 100 z_{pd} \qquad \forall\ d,\ p \tag{43}$$

$$x_{pd} \geq 0, \qquad \forall\ d,\ p \tag{44}$$

$$s_{dc}, x_{pd} \in \mathbb{R}; \qquad y_{dc}, z_{pd} \in \{0,1\}$$

Eqs. (37)-(39) represent the constrains for the distribution level problem while the rest of them are associated to the production part. The binary reformulation was applied here substituting z_{pd} by the vector zc_{pd}, and adding eq. (45).

$$zc_{pd}(zc_{pd} - 1) = 0 \tag{45}$$

At this point the KKT conditions were derived for the lower-level problem (eqs. (40)-(45) as presented next:

$$\mathcal{L}(x, zc) = \sum_{pd} c^3_{pd} x_{pd} + \sum_{pd} c^4_{pd} zc_{pd} + \sum_{p} \lambda 1_p (\sum_{d} x_{pd} - A_p) + \sum_{d} \lambda 2_d (\sum_{c} s_{dc} - \sum_{p} x_{pd})$$
$$+ \sum_{pd} \lambda 3_{pd} (x_{pd} - 100 zc_{pd}) - \sum_{pd} \lambda 4_{pd} x_{pd} + \sum_{pd} \mu_{pd} (zc_{pd}(zc_{pd} - 1)) \tag{46}$$

Stationary conditions

$$\nabla_x \mathcal{L}(x, \lambda) = \sum_{pd} c^3_{pd} + \sum_{p} \lambda 1_p - \sum_{d} \lambda 2_d + \sum_{pd} \lambda 3_{pd} - \sum_{pd} \lambda 4_{pd} = 0 \tag{47}$$

$$\nabla_{zc}\mathcal{L}(zc,\lambda) = \sum_{pd} c^4_{pd} - \sum_{pd} \lambda 3_{pd} 100 + \sum_{pd} \mu_{pd}(2zc_{pd} - 1) = 0 \tag{48}$$

Complementary slackness

$$\lambda 1_p(\sum_d x_{pd} - A_p) = 0 \qquad \forall\ p \tag{49}$$

$$\lambda 2_d(\sum_c s_{dc} - \sum_p x_{pd}) = 0 \qquad \forall\ d \tag{50}$$

$$\lambda 3_{p,d}(x_{pd} - 100zc_{pd}) = 0 \qquad \forall\ d,\ p \tag{51}$$

$$\lambda 4_{p,d}(-x_{pd}) = 0 \qquad \forall\ d,\ p \tag{52}$$

Dual feasibility

$$\lambda 1_p \geq 0 \qquad \forall\ p \tag{53}$$

$$\lambda 2_d \geq 0 \qquad \forall\ d \tag{54}$$

$$\lambda 3_{p,d} \geq 0 \qquad \forall\ d,\ p \tag{55}$$

$$\lambda 4_{p,d} \geq 0 \qquad \forall\ d,\ p \tag{56}$$

The single-level model *B-MINLP-E4* including the lower-level reformulation and the equivalent KKT optimality conditions is presented as follows:

$$
(B-MINLP-E4) \qquad \min_{y_{dc},s_{dc}} \sum_{d,c} c^1_{dc} y_{dc} + \sum_{p,d} c^2_{pd} x_{pd}
$$

$$
\begin{aligned}
& s.t. \\
& \sum_{d} s_{dc} \geq b_c \qquad \forall\ c \\
& s_{dc} \leq 35 y_{dc} \qquad \forall\ c,\ d \\
& s_{dc} \geq 0 \qquad \forall\ c,\ d
\end{aligned}
$$

$$
\text{Primal feasibility,}\begin{cases} \sum_d x_{pd} \leq A_p \qquad \forall\ p \\ \sum_c s_{dc} \leq \sum_p x_{pd} \qquad \forall\ d \\ x_{pd} \leq 100 zc_{pd} \qquad \forall\ p,\ d \\ -x_{pd} \leq 0 \qquad \forall\ p,\ d \end{cases}
$$

$$
\text{Stationary conditions,}\begin{cases} \sum_{pd} c^3_{pd} + \sum_p \lambda 1_p - \sum_d \lambda 2_d + \sum_{pd} \lambda 3_{pd} - \sum_{pd} \lambda 4_{pd} = 0 \\ \sum_{pd} c^4_{pd} - \sum_{pd} \lambda 3_{pd} 100 + \sum_{pd} \mu_{pd}(2zc_{pd} - 1) = 0 \end{cases}
$$

$$
\text{Complementary slackness,}\begin{cases} \lambda 1_p(\sum_d x_{pd} - A_p) = 0 \qquad \forall\ p \\ \lambda 2_d(\sum_c s_{dc} - \sum_p x_{pd}) = 0 \qquad \forall\ d \\ \lambda 3_{p,d}(x_{pd} - 100 zc_{pd}) = 0 \qquad \forall\ p,\ d \\ \lambda 4_{p,d}(-x_{pd}) = 0 \qquad \forall\ p,\ d \end{cases}
$$

$$
\text{Dual feasibility,}\begin{cases} \lambda 1_p \geq 0 \qquad \forall\ p \\ \lambda 2_d \geq 0 \qquad \forall\ d \\ \lambda 3_{p,d} \geq 0 \qquad \forall\ p,\ d \\ \lambda 4_{p,d} \geq 0 \qquad \forall\ p,\ d \end{cases}
$$

$$
\text{Binary reformulation,}\begin{cases} zc_{pd}(zc_{pd} - 1) = 0 \qquad \forall\ p,\ d \end{cases}
$$

3.4.1 Implementation and results

As commented before the model *B-MINLP-E1* was implemented in GAMS24.7.1. The model consists of 65 equations, 50 single variables and 6 discrete variables. The main parameters are included in Table 5. Baron18.11.12 was used as solver achieving a global optimal solution in 0.912 seconds after two iterations.

Table 5: Main parameters to be used in the model *B-MINLP-E4*

A	$p=1$	$p=2$	
	135	100	
c^1_{dc}	$c=1$	$c=2$	$c=3$
$d=1$	75	60	50
$d=2$	80	30	65
c^2_{dp}	$d=1$	$d=2$	
$p=1$	21	30	
$p=2$	26	25	
c^3_{dp}	$d=1$	$d=2$	
$p=1$	20	25	
$p=2$	20	25	
c^4_{dp}	$d=1$	$d=2$	
$p=1$	100	80	
$p=2$	110	70	
b_c	$c=1$	$c=2$	$c=3$
	55	65	15

The optimal solution obtained using the proposed reformulation approach is same as that reported in literature (Avraamidou and Pistikopoulos, 2017). The cuts of the form $f^k < f^{k-1}$, as explained in section 2, were iteratively included to identify all the optimal solutions. The product flow between plants and distribution centres are $x_{p1,d1} = 85$ and $x_{p2,d2} = 50$, while the flow between distribution centres and customers are $s_{d1,c1} = 35$, $s_{d1,c2} = 35$, $s_{d1,c3} = 15$, $s_{d2,c1} = 20$ and $s_{d2,c2} = 30$.

3.5 5^{th} Example

The second example was presented as a motivating example in Yue and You (2017) consisting of the profit maximisation of a chemical production supply chain system. As shown in Fig. 3, the leader and follower were defined for the supply chain. The leader is in charge of the process P, which has two production schemes namely P_1 and P_2. Scheme P_1 consumes chemical A as the raw material and produces platform chemical B. Scheme P_2 also consumes chemical A as the raw material but produces platform chemical C. In contrast, the downstream follower is in charge of the process Q, which has three production schemes, namely Q_1, Q_2, and Q_3. Scheme Q_1 consumes platform chemical B as the raw material and produces final product D. Scheme Q_2 also consumes platform chemical B as the raw material but produces final product E. Scheme Q_3 consumes platform chemical C as the raw material and produces final product E. Variable and fixed costs would occur during the production processes. Besides the revenue from selling platform chemicals, the leader has agreed with the follower to share some of the follower's product sales revenue.

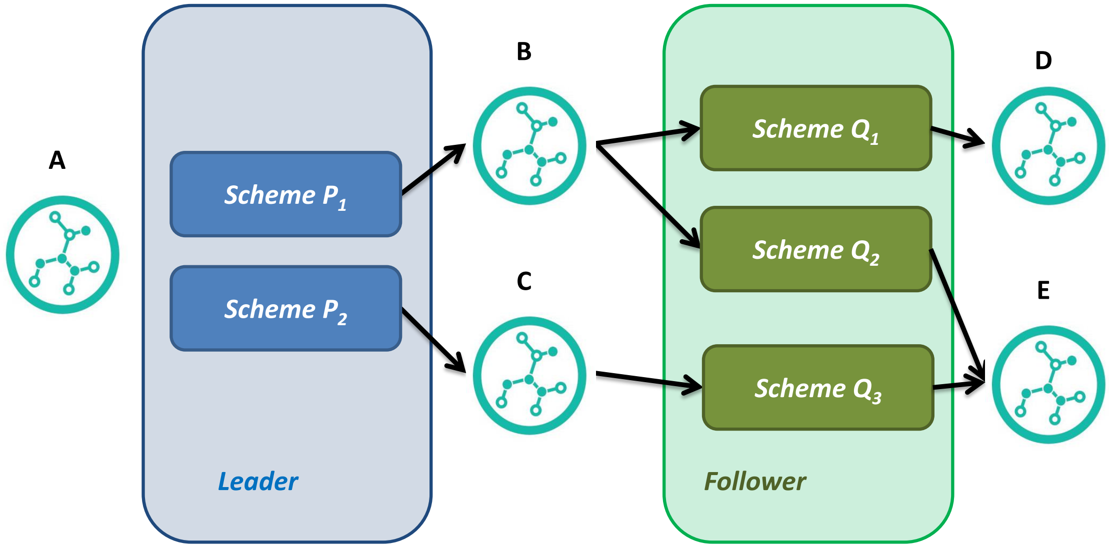

Figure 3: Supply chain superstructure of the example

The MILP-MILP bi-level optimisation model (*B-MILP-E5*) for this illustrative example is given below.

$$
\begin{aligned}
(B-MILP-E5) \qquad & \max P^L = 30x^L_{P1} + 20x^L_{P2} - 50y^L_{P1} - 100y^L_{P2} + 10x^F_{Q2} + 10x^F_{Q3} \\
& s.t. \\
& x^L_{P1} + x^L_{P2} \leq 10 \\
& x^L_{P1} \leq 10y^L_{P1} \\
& x^L_{P2} \leq 10y^L_{P2} \\
& y^L_{P1} + y^L_{P2} \leq 1 \\
& x^L_{P1}, x^L_{P2} \in \mathbb{R}_+; \qquad y^L_{P1}, y^L_{P2} \in \{1,0\}; \qquad P^L \in \mathbb{R} \\
& \max P^F = 40x^F_{Q1} + 40x^F_{Q2} + 65x^F_{Q3} - 100y^F_{Q1} - 250y^F_{Q2} - 100y^F_{Q3} \\
& s.t. \\
& \quad x^F_{Q1} + x^F_{Q2} + x^F_{Q3} \leq 10 \\
& \quad x^F_{Q1} \leq 10y^F_{Q1} \\
& \quad x^F_{Q2} \leq 10y^F_{Q2} \\
& \quad x^F_{Q3} \leq 10y^F_{Q3} \\
& \quad y^F_{Q1} + y^F_{Q2} + y^F_{Q3} \leq 1 \\
& \quad x^F_{Q1} + x^F_{Q2} \leq x^L_{P1} \\
& \quad x^F_{Q3} \leq x^L_{P2} \\
& x^F_{Q1}, x^F_{Q2}, x^F_{Q3} \in \mathbb{R}_+; \qquad y^F_{Q1}, y^F_{Q2}, y^F_{Q3} \in \{1,0\}; \qquad P^F \in \mathbb{R}
\end{aligned}
$$

where x represent the production rates for each production scheme while y are binary variables that indicate the schemes selection (i.e., 1 if selected; 0 otherwise). Superscripts L and F represents leader and follower, respectively. Similarly, the subscripts represent the corresponding production scheme. At this point the KKT conditions for the lower-level problem must be derived as presented in section 2. The complete derivation of the KKT conditions are presented in the appendix B.3 while the single-level model (*B-MINLP-E5*) including the associated inner problem reformulation is presented as follows:

$$(B - MINLP - E5) \qquad \max P^L = 30x^L_{P1} + 20x^L_{P2} - 50y^L_{P1} - 100y^L_{P2} + 10x^F_{Q2} + 10x^F_{Q3}$$

$$\begin{aligned}
& s.t. \\
& x^L_{P1} + x^L_{P2} \leq 10 \\
& x^L_{P1} \leq 10y^L_{P1} \\
& x^L_{P2} \leq 10y^L_{P2} \\
& y^L_{P1} + y^L_{P2} \leq 1 \\
& x^L_{P1}, x^L_{P2} \in \mathbb{R}_+; \qquad y^L_{P1}, y^L_{P2} \in \{1, 0\}; \qquad P^L \in \mathbb{R}
\end{aligned}$$

$$\text{Primal feasibility,} \begin{cases} x^F_{Q1} + x^F_{Q2} + x^F_{Q3} \leq 10 \\ x^F_{Q1} \leq 10yc^F_{Q1} \\ x^F_{Q2} \leq 10yc^F_{Q2} \\ x^F_{Q3} \leq 10yc^F_{Q3} \\ yc^F_{Q1} + yc^F_{Q2} + yc^F_{Q3} \leq 1 \\ x^F_{Q1} + x^F_{Q2} \leq x^L_{P1} \\ x^F_{Q3} \leq x^L_{P2} \end{cases}$$

$$\text{Stationary conditions,} \begin{cases} 40 - \lambda_1 - \lambda_2 - \lambda_6 = 0 \\ 40 - \lambda_1 - \lambda_3 - \lambda_6 = 0 \\ 65 - \lambda_1 - \lambda_4 - \lambda_7 = 0 \\ -100 + 10\lambda_2 - \lambda_5 - \mu_1(2yc^F_{Q1} - 1) = 0 \\ -250 + 10\lambda_3 - \lambda_5 - \mu_2(2yc^F_{Q2} - 1) = 0 \\ -100 + 10\lambda_4 - \lambda_5 - \mu_3(2yc^F_{Q3} - 1) = 0 \end{cases}$$

$$\text{Complementary slackness,} \begin{cases} \lambda_1(x^F_{Q1} + x^F_{Q2} + x^F_{Q3} - 10) = 0 \\ \lambda_2(x^F_{Q1} - 10yc^F_{Q1}) = 0 \\ \lambda_3(x^F_{Q2} - 10yc^F_{Q2}) = 0 \\ \lambda_4(x^F_{Q3} - 10yc^F_{Q3}) = 0 \\ \lambda_5(yc^F_{Q1} + yc^F_{Q2} + yc^F_{Q3} - 1) = 0 \\ \lambda_6(x^F_{Q1} + x^F_{Q2} - x^L_{P1}) = 0 \\ \lambda_7(x^F_{Q3} - x^L_{P2}) = 0 \end{cases}$$

$$
\text{Dual feasibility,} \begin{cases} \lambda_1 \geq 0 \\ \lambda_2 \geq 0 \\ \lambda_3 \geq 0 \\ \lambda_4 \geq 0 \\ \lambda_5 \geq 0 \\ \lambda_6 \geq 0 \\ \lambda_7 \geq 0 \end{cases}
$$

$$
\text{Binary reformulation,} \begin{cases} yc^F_{Q1}(yc^F_{Q1} - 1) = 0 \\ yc^F_{Q2}(yc^F_{Q2} - 1) = 0 \\ yc^F_{Q3}(yc^F_{Q3} - 1) = 0 \end{cases}
$$

$$
x^F_{Q1}, x^F_{Q2}, x^F_{Q3} \in \mathbb{R}_+; \qquad yc^F_{q1}, yc^F_{Q2}, yc^F_{Q3} \in \mathbb{R}_+
$$

$$
P^F \in \mathbb{R}
$$

3.5.1 Implementation and results

The model *B-MINLP-E5* consists of 35 equations, 30 single variables and 7 discrete variables. The main parameters were included in the model and the optimal solution was found in 3.476 seconds after three iterations.

Note that this example was solved in Yue and You (2017) using a pseudo bi-level (i.e. two-step approach) in which first, the upper-level problem is solved, then its decisions are fixed to solve the lower-level part. These steps are repeated fixing the lower level decisions until no further improvement is obtained. The results reported by Yue and You (2017) were compared with the ones obtained using the proposed reformulation (bi-level MINLP) (see Table 6).

Table 6: Optimal solutions from different modelling and optimisation approaches

	$P_1(t/d)$	$P_2(t/d)$	$Q_1(t/d)$	$Q_2(t/d)$	$Q_3(t/d)$	P^L (\$)	P^F (\$)	Global Profit (\$)
pseudo bi-level	10	0	10	0	0	250	300	550
Bi-level MINLP								
Iter-1	10	0	0	10	0	350	150	500
Iter-2	10	0	10	0	0	250	300	550
Iter-3	0	10	0	0	10	200	550	750

From Table 6 it can be seen that the proposed bi-level MINLP approach is capable of identifying different optimal solution (including the global optimal). Note that, the optimal solution was obtained in less than 4 seconds.

4 Concluding remarks

This paper introduces a reformulation strategy for the solution of mixed-integer bi-level programming problems with integer variables in both optimisation levels. The capabilities of the proposed strategy to give the global optimal solution of such problems was validated using three different MILP-MILP examples. Numerical results prove that the proposed reformulation is a promising alternative to the current approximation algorithms. The main advantage of the proposed approach is that after applying the proposed reformulation, any appropriate global optimiser can be used. Additionally, the results demonstrate that the proposed reformulation is computationally efficient, hence applicable to a wide variety of problems such as hierarchical model predictive controllers, scheduling and control integration or planning and scheduling integration. Future work will include testing the proposed approach on large scale problems.

Acknowledgment

The authors would like to thank the financial support received from the UK Engineering and Physical Sciences Research Council (under the project EP/M028240/1).

A Model statistics for comparison

Table 7: Reformulated models statistics

Examples	Continuous Variables	Discrete Variables	Equations	CPU (s)
1^{st}	8	0	10	0.840
2^{nd}	9	1	12	2.838
3^{rd}	16	1	21	2.660
4^{th}	50	6	65	0.912
5^{th}	30	7	35	3.476

B Derivation of KKT conditions

B.1 Case study 2

From the original bi-level problem (model *B-MILP-E2*) the KKT conditions were derived as follows:

$$
\begin{aligned}
\mathcal{L}(yc) = (yc &+ \lambda_1(-25x + 20yc - 30) + \lambda_2(x + 2yc - 10) \\
&+ \lambda_3(2x - yc - 15) + \lambda_4(-2x - 10yc + 15) \\
&+ \mu_1(yc(yc-1)(yc-2)(yc-3)(yc-4))
\end{aligned}
\tag{B.1}
$$

Stationary conditions

$$
\begin{aligned}
\nabla_{yc}\mathcal{L}(yc, \lambda) = 1 &+ 20\lambda_1 + 2\lambda_2 - \lambda_3 - 10\lambda_4 \\
&+ \mu_1(5yc^4 - 40yc^3 + 105yc^2 - 100yc + 24) = 0
\end{aligned}
\tag{B.2}
$$

Complementary slackness

$$\lambda_1(-25x + 20yc - 30) = 0 \tag{B.3}$$

$$\lambda_2(x + 2yc - 10) = 0 \tag{B.4}$$

$$\lambda_3(2x - yc - 15) = 0 \tag{B.5}$$

$$\lambda_4(-2x - 10yc + 15) = 0 \tag{B.6}$$

Dual feasibility

$$\lambda_1 \geq 0 \tag{B.7}$$

$$\lambda_2 \geq 0 \tag{B.8}$$

$$\lambda_3 \geq 0 \tag{B.9}$$

$$\lambda_4 \geq 0 \tag{B.10}$$

The single-level model *B-MINLP-E2* is included in section 3.2.

B.2 Case study 3

From the original bi-level problem (model *B-MILP-E3*) the KKT conditions were derived as follows:

$$\begin{aligned}\mathcal{L}(x_3, y_1, y_2) = &(-x_1 + x_2 - 2x_3 - yc_1 + 5yc_2 + y_3)\\ &+ \lambda_1(6.4x_1 + 7.2x_2 + 2.5x_3 - 11.5) + \lambda_2(-8x_1 - 4.9x_2 - 3.2x_3 - 5)\\ &+ \lambda_3(3.3x_1 + 4.1x_2 + 0.02x_3 + 4yc_1 + 4.5yc_2 + 0.5y_3 - 1)\\ &+ \mu_1(yc_1(yc_1 - 1)) + \mu_2(yc_2(yc_2 - 1))\end{aligned} \tag{B.11}$$

Stationary conditions

$$\nabla_{x_3}\mathcal{L}(x_3, \lambda) = -2 + 2.5\lambda_1 - 3.2\lambda_2 + 0.02\lambda_3 = 0 \tag{B.12}$$

$$\nabla_{yc_1}\mathcal{L}(yc_1, \lambda) = -1 + 4\lambda_3 + \mu_1(2yc_1 - 1) = 0 \tag{B.13}$$

$$\nabla_{yc_2}\mathcal{L}(yc_2, \lambda) = 5 + 4.5\lambda_3 + \mu_2(2yc_2 - 1) = 0 \tag{B.14}$$

Complementary slackness

$$\lambda_1(6.4x_1 + 7.2x_2 + 2.5x_3 - 11.5) = 0 \tag{B.15}$$

$$\lambda_2(-8x_1 - 4.9x_2 - 3.2x_3 - 5) = 0 \tag{B.16}$$

$$\lambda_3(3.3x_1 + 4.1x_2 + 0.02x_3 + 4yc_1 + 4.5yc_2 + 0.5y_3 - 1) = 0 \tag{B.17}$$

Dual feasibility

$$\lambda_1 \geq 0 \tag{B.18}$$

$$\lambda_2 \geq 0 \tag{B.19}$$

$$\lambda_3 \geq 0 \tag{B.20}$$

The single-level model *B-MINLP-E3* is included in section 3.3.

B.3 Case study 5

From the original bi-level problem (model *B-MILP-E5*) the KKT conditions were derived as follows:

$$\begin{aligned}
\mathcal{L}(x^F_{Q1}, x^F_{Q2}, x^F_{Q3}, yc^F_{Q1}, yc^F_{Q2}, yc^F_{Q3}) = \\
&(40x^F_{Q1} + 40x^F_{Q2} + 65x^F_{Q3} - 100yc^F_{Q1} - 250yc^F_{Q2} - 100yc^F_{Q3}) \\
&- \lambda_1(x^F_{Q1} + x^F_{Q2} + x^F_{Q3} - 10) - \lambda_2(x^F_{Q1} - 10yc^F_{Q1}) \\
&- \lambda_3(x^F_{Q2} - 10yc^F_{Q2}) - \lambda_4(x^F_{Q3} - 10yc^F_{Q3}) \\
&- \lambda_5(yc^F_{Q1} + yc^F_{Q2} + yc^F_{Q3} - 1) - \lambda_6(x^F_{Q1} + x^F_{Q2} - x^L_{P1}) \\
&- \lambda_7(x^F_{Q3} - x^L_{P2}) - \mu_1(yc^F_{Q1}(yc^F_{Q1} - 1)) \\
&- \mu_2(yc^F_{Q2}(yc^F_{Q2} - 1)) - \mu_3(yc^F_{Q3}(yc^F_{Q3} - 1))
\end{aligned} \tag{B.21}$$

Stationary conditions

$$\nabla_{x^F_{Q1}} \mathcal{L}(x^F_{Q1}, \lambda) = 40 - \lambda_1 - \lambda_2 - \lambda_6 = 0 \tag{B.22}$$

$$\nabla_{x^F_{Q2}} \mathcal{L}(x^F_{Q2}, \lambda) = 40 - \lambda_1 - \lambda_3 - \lambda_6 = 0 \tag{B.23}$$

$$\nabla_{x^F_{Q3}} \mathcal{L}(x^F_{Q3}, \lambda) = 65 - \lambda_1 - \lambda_4 - \lambda_7 = 0 \tag{B.24}$$

$$\nabla_{yc^F_{Q1}} \mathcal{L}(yc^F_{Q1}, \lambda) = -100 + 10\lambda_2 - \lambda_5 - \mu_1(2yc^F_{Q1} - 1) = 0 \tag{B.25}$$

$$\nabla_{yc^F_{Q2}} \mathcal{L}(yc^F_{Q2}, \lambda) = -250 + 10\lambda_3 - \lambda_5 - \mu_2(2yc^F_{Q2} - 1) = 0 \tag{B.26}$$

$$\nabla_{yc^F_{Q3}} \mathcal{L}(yc^F_{Q3}, \lambda) = -100 + 10\lambda_4 - \lambda_5 - \mu_3(2yc^F_{Q3} - 1) = 0 \tag{B.27}$$

Complementary slackness

$$\lambda_1(x^F_{Q1} + x^F_{Q2} + x^F_{Q3} - 10) = 0 \tag{B.28}$$

$$\lambda_2(x^F_{Q1} - 10yc^F_{Q1}) = 0 \tag{B.29}$$

$$\lambda_3(x^F_{Q2} - 10yc^F_{Q2}) = 0 \tag{B.30}$$

$$\lambda_4(x^F_{Q3} - 10yc^F_{Q3}) = 0 \tag{B.31}$$

$$\lambda_5(yc^F_{Q1} + yc^F_{Q2} + yc^F_{Q3} - 1) = 0 \tag{B.32}$$

$$\lambda_6(x^F_{Q1} + x^F_{Q2} - x^L_{P1}) = 0 \tag{B.33}$$

$$\lambda_7(x^F_{Q3} - x^L_{P2}) = 0 \tag{B.34}$$

Dual feasibility

$$\lambda_1 \geq 0 \tag{B.35}$$

$$\lambda_2 \geq 0 \tag{B.36}$$

$$\lambda_3 \geq 0 \tag{B.37}$$

$$\lambda_4 \geq 0 \tag{B.38}$$

$$\lambda_5 \geq 0 \tag{B.39}$$

$$\lambda_6 \geq 0 \tag{B.40}$$

$$\lambda_7 \geq 0 \tag{B.41}$$

The single-level model *B-MINLP-E5* is included in section 3.5.

References

Arroyo, J. M. and Fernandez, F. J. (2009). A genetic algorithm approach for the analysis of electric grid interdiction with line switching. In *15th International Conference on Intelligent System Applications to Power Systems*, volume 15, pages 1–6.

Avraamidou, S. and Pistikopoulos, E. N. (2017). A multiparametric mixed-integer bi-level optimization strategy for supply chain planning under demand uncertainty. *IFAC Papers Online*, 50-1:10178–10183.

Avraamidou, S. and Pistikopoulos, E. N. (2018). A novel algorithm for the global solution of mixed-integer bi-level multi-follower problems and its application to planning & scheduling integration. *2018 European Control Conference*, 1:1056–1061.

Avraamidou, S. and Pistikopoulos, E. N. (2019). A multi-parametric optimizationapproach for bilevel mixed-integer linear and quadratic programming problems. *Computers and Chemical Engineering*, 125:98–113.

Bard, J. and Falk, James, E. (1982). An explicit solution to the multi-level programming problem. *Computers & Operations Reserach*, 9:77–100.

Beykal, B., Avraamidou, S., Pistikopoulos, I., Onel, M., and Pistikopoulos, E. N. (2020). Domino: Data-driven optimization of bi-level mixed-integer nonlinear problems. *Journal of Global Optimization*, pages 1–36.

Bialas, W. and Karwan, M. (1982). On two-level optimization. *IEEE Transactions on Automatic Control*, 27:211–214.

Calvete, H., Gal, C., and Oliveros, M. J. (2010). Bilevel model for production distribution planning solved by using ant colony optimization. *Computers and Operations Research*, 38(1):320–327.

Caramia, M. and Mari, R. (2016). A decomposition approach to solve a bilevel capacitated facility location problem with equity constraints. *operations Letters*, 10(5):997–1019.

Clark, P. A. and Westerberg, A. N. (1990). Bilevel programming for steady-state chemical process design—i. fundamentals and algorithms. *Computers & Chemical Engineering*, 14:87–97.

Colson, B., Marcotte, P., and Savard, G. (2005). Bilevel programming: A survey. *A Quarterly Journal of Operations Research*, 3:87–107.

Deng, X. (1992). Complexity issues in bilevel linear programming. *Society for Industrial and Applied Mathematics*, 13(5):1194–1217.

Fischetti, M., Ljubie, I., Monaci, M., and Sinni, M. (2017). A new general-purpose algorithm for mixed-integer bilevel linear programs. *Operations Research*, 65(6):1615–1637.

Fortuny-Amat, J. and McCarl, B. (1981). A representation and economic interpretation of a two-level programming problem. *The Journal of the Operational Research Society*, 32:783–792.

Gao, Y., Zhang, G., Lu, J., and Wee, H. M. (2011). Particle swarm optimization for bi-level pricing problems in supply chains. *Journal of Global Optimization*, 51(7):245–254.

Garces, L. P., Conejo, A. J., Garcia-Bertrand, R., and Romero, R. (2009). A bilevel approach to transmission expansion planning within a market environment. *IEEE Transactions on Power Systems*, 24:1513–1522.

Garcia-Herreros, P., Zhang, L., Arslan, E., Mehta, S., and Grossmann, I. E. (2016). Mixed-integer bilevel optimization for capacity planning with rational markets. *Computers & Chemical Engineering*, 86:33–47.

Gumus, Z. H. and Floudas, C. A. (2005). Global optimization of mixed-integer bilevel programming problems. *Computational Management Science*, 2:181–212.

Gupta, A. and Maranas, C. D. (2000). A two-stage modeling and solution framework for multisite midterm planning under demand uncertainty. *Industrial and Engineering Chemistry Research*, 39(10):3799–3813.

Hansen, P., Jaumard, B., and Savard, G. (1992). New branch-and-bound rules for linear bilevel programming. *Society for Industrial and Applied Mathematics*, 13(5):1194–1217.

Hecheng, L. and Yuping, W. (2008). Exponential distribution-based genetic algorithm for solving mixed-integer bilevel programming problems. *Journal of Systems Engineering and Electronics*, 19(6):1157–1164.

Kleniati, P. M. and Adjiman, C. S. (2015). A generalization of the branch-and-sandwich algorithm: From continuous to mixed-integer nonlinear bilevel problems. *Computers & Chemical Engineering*, 72:373–386.

Kuo, R. and Han, Y. (2011). A hybrid of genetic algorithm and particle swarm optimization for solving bi-level linear programming problem- a case study on supply chain model. *Applied Mathematical Modelling*, 35(8):3905–3917.

Marcotte, P. and Savard, G. (2009). *Bilevel Programming: Applications*. Sringer.

Mitsos, A. (2010). Global solution of nonlinear mixed-integer bilevel programs. *Journal of Global Optimization*, 47(4):557–582.

Mitsos, A., Bollas, G. M., and Barton, P. I. (2009). Bilevel optimization formulation for parameter estimation in liquid–liquid phase equilibrium problems. *Chemical Engineering Science*, 64(3):548–559.

Moore, J. and Bard, J. F. (1990). The mixed integer linear bilevel programming problem. *operations Research*, 38(5):911–921.

Motto, A. L., Arroyo, J. M., and Galiana, Francisco, D. (2005). A mixed-integer lp procedure for the analysis of electric grid security under disruptive threat. *IEEE Transactions on Power Systems*, 20:1357–1365.

Nishizaki, I. and Sakawa, M. (2005). Computational methods through genetic algorithms for obtaining stackelberg solutions to two-level integer programming problems. *Cybernetics and Systems*, 36(6):565–579.

Poirion, P. L., Toubaline, S., D'Ambrosio, C., and Liberti, L. (2020). Algorithms and applications for a class of bilevel milps. *Discrete Applied Mathematics*, 272:75–89.

Roghanian, E., Sadjadi, S., and Aryanezhad, M. (2007). A probabilistic bi-level linear multi-objective programming problem to supply chain planning. *Applied Mathematics and Computation*, 188(1):786–800.

Ryu, J. H., Dua, V., and Pistikopoulos, E. N. (2004). A bilevel programming framework for enterprise-wide process networks under uncertainty. *Computers & Chemical Engineering*, 28:1121–1129.

Saharidis, G. K. and Ierapetritou, M. G. (2009). Resolution method for mixed integer bi-level linear problems based on decomposition technique. *Journal of Global Optimization*, 44(1):29–51.

Sherali, H. D. and Adams, W. P. (1990). A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. *SIAM Journal on Discrete Mathematics*, 3(3):411–430.

Shokry, A., Medina-González, S. A., and Espuña, A. (2017). Mixed-integer multiparametric approach based on machine learning techniques. *Computer Aided Chemical Engineering*, 40:451–456.

Sinha, A., Malo, P., and Deb, K. (2018). A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on evolutionary computation*, 22(2):276–295.

Xu, P. and Wang, L. (2014). An exact algorithm for the bilevel mixed integer linear programming problem under three simplifying assumptions. *Computers & Operations Research*, 41:309–318.

Yue, D., Gao, J., and You, F. (2019). A projection-based reformulation and decomposition algorithm for global optimization of a class of mixed integer bilevel linear programs. *Journal of Global Optimization*, 73:27–57.

Yue, D. and You, F. (2016). Projection-based reformulation and decomposition algorithm for a class of mixed-integer bilevel linear programs. *Computer Aided Chemical Engineering*, 36:481–486.

Yue, D. and You, F. (2017). Stackelberg-game-based modeling and optimization for supply chain design and operations: A mixed integer bilevel programming framework. *Computers & Chemical Engineering*, 102:81–95.

Zeng, B. and An, Y. (2014). Solving bilevel mixed integer program by reformulations and decomposition. *Optim online*, 1:1–34.