A Comparative Study on Bilevel Fractional Programming problem in Pricing-Production System solution

Abstract:

Bilevel fractional programming (BLFP) models combine hierarchical decision-making with efficiency-based objective functions. Such models naturally arise in pricing–production systems where strategic decisions influence operational responses, and performance is evaluated using ratio-based measures. Despite their modeling power, BLFP problems are computationally challenging due to the interaction between the bilevel structure and fractional objectives.

This paper presents a structured comparative study of three solution approaches for BLFP problems: direct enumeration of upper-level decisions, duality-based single-level reformulation, and Charnes–Cooper transformation with mixed-integer linear programming (MILP) reformulation. A pricing–production framework is developed to illustrate the methodology. Numerical experiments demonstrate that although all approaches yield identical optimal solutions for small instances, their computational scalability differs significantly. The MILP reformulation emerges as the most robust and scalable method. The study offers practical insights into the solvability of BLFP models and provides guidance for selecting appropriate solution strategies.

This study contributes by systematically analyzing structural reformulation mechanisms and identifying scalability boundaries for each approach.

Key Words: Bilevel Fractional Programming; Leader–Follower Optimization; Charnes–Cooper Transformation; Mixed-Integer Linear Programming; Pricing Strategy; Comparative Algorithm Study

1- Introduction

Bilevel programming problems represent an important class of hierarchical optimization models that naturally arise in decentralized and multi-agent decision-making settings. In these problems, two decision makers interact at different levels of authority. The upper-level decision maker, commonly referred to as the leader, makes the initial decisions, while the lower-level decision maker, known as the follower, responds by optimizing its own objective function subject to the leader’s decisions. This sequential interaction distinguishes bilevel programming from classical single-level optimization and is the primary source of its analytical and computational difficulty [1], [2].

Fractional programming, on the other hand, deals with optimization problems in which the objective function is expressed as a ratio of two real-valued functions. Objectives of this type are particularly well-suited for modeling efficiency, profitability, productivity, and other performance measures that naturally appear in real applications. As a result, fractional programming has been widely applied in areas such as economics, engineering design, operations research, management science, and energy systems [3].

Bilevel fractional programming problems (BLFPs) arise when these two modeling frameworks are combined. The resulting models incorporate both hierarchical decision-making and fractional objective functions, which makes them inherently complex. The bilevel structure introduces implicit constraints through the follower's optimal response, while the fractional objectives lead to nonlinear and nonconvex formulations. Together, these features severely limit the effectiveness of standard optimization techniques and substantially increase the computational effort required to solve such problems [4], [5].

The difficulty of BLFPs becomes even more pronounced in practical situations where discrete or integer variables appear at the upper level, while the lower-level problem involves continuous, nonlinear, or nonconvex response functions. In these cases, the feasible region of the overall problem may be highly irregular or even disconnected, and classical convexity assumptions no longer apply. Consequently, solving BLFPs requires specialized methods that can address both the hierarchical structure of the problem and the fractional nature of the objectives [6].

From a theoretical perspective, bilevel fractional programming problems are widely regarded as some of the most challenging optimization problems. It has been established that bilevel programming is NP-hard in general [2], and the addition of fractional objectives further increases nonconvexity and solution complexity [4], [7]. Even in simplified settings where all objective functions and constraints are linear, the presence of fractional objectives combined with a bilevel decision structure can lead to complex feasible regions and multiple local optima. For this reason, BLFPs are widely recognized as computationally difficult, and conventional optimization approaches cannot be applied directly without significant reformulation.

Despite these challenges, bilevel fractional programming provides a powerful and realistic modeling framework for many real-world applications. Examples include resource allocation, pricing and taxation policy design, supply chain coordination, transportation and logistics planning, and decentralized production and distribution systems [1], [4]. The ability of BLFPs to capture both hierarchical interactions between decision makers and efficiency-oriented performance measures makes them particularly suitable for modeling modern, complex decision-making environments.

Over the years, a variety of solution approaches have been proposed for BLFPs. Transformation-based methods, such as the Charnes–Cooper transformation [3], attempt to convert fractional objective functions into equivalent linear or convex forms under suitable conditions. Duality-based approaches exploit strong duality in the lower-level problem to reformulate the bilevel model as a single-level optimization problem with additional constraints [6], [8]. Parametric and exact solution techniques have also been developed for specific subclasses of bilevel fractional models [7].

In addition to exact and analytical methods, heuristic and metaheuristic algorithms have attracted growing interest in recent years. Approaches such as genetic algorithms and hybrid evolutionary strategies have proven particularly useful for large-scale or highly nonlinear BLFPs, where exact methods are often computationally impractical [9]. Although these techniques do not guarantee global optimality, they frequently produce high-quality solutions within reasonable computational times.

While individual solution techniques for bilevel fractional programming problems have been proposed in the literature — including strong-duality-based reformulations and fractional transformations such as the Charnes–Cooper approach — a structured comparative analysis of these techniques within a unified application framework remains limited.

In particular, few studies evaluate scalability boundaries and computational behavior across enumeration, duality substitution, and MILP-based reformulations under the same pricing–production structure.

This study addresses this gap by conducting a systematic structural and computational comparison of three exact solution paradigms within a consistent modeling environment.

The objective of this paper is to compare three alternative solution approaches for BLFP problems within a pricing–production framework:

1. Direct enumeration of upper-level decisions
2. Duality-based single-level reformulation
3. Charnes–Cooper transformation with MILP reformulation

By evaluating their structural properties and computational behaviour, the study provides insight into the practical solvability of BLFP models and highlights the impact of reformulation strategies on scalability and numerical stability.

2- Related Work

Bilevel programming has long been recognized as a powerful framework for modeling hierarchical decision-making problems, particularly in settings where decisions are made sequentially by multiple agents with different objectives. Early foundational work established the theoretical structure of bilevel optimization and highlighted its inherent complexity, even in linear cases [1], [2]. These studies showed that the main difficulty arises from the dependency of the upper-level decisions on the optimal response of the lower-level problem, which often leads to nonconvex and challenging solution spaces.

At the same time, fractional programming has been widely used to model efficiency-oriented objectives, where performance is expressed as a ratio between competing measures such as profit and cost. The classical transformation introduced by Charnes and Cooper [1] remains one of the most influential contributions in this area, as it enables the conversion of fractional objectives into equivalent linear forms under suitable assumptions. Over time, fractional programming has been extended to cover more complex and realistic applications across engineering, economics, and operations research.

The integration of these two frameworks—bilevel and fractional programming—has led to the development of bilevel fractional programming (BLFP) models, which are particularly suitable for representing systems where hierarchical decisions are evaluated based on efficiency criteria. Several studies have explored theoretical properties and solution approaches for such models, including exact reformulations and duality-based techniques [8], [9]. These approaches aim to transform the bilevel structure into a single-level problem, although they often introduce additional complexity due to nonlinear constraints.

More recent research has focused on improving the solvability of BLFP problems by proposing structured solution methods and exploring their computational behavior. For instance, exact solution approaches based on strong duality and reformulation techniques have been developed for specific classes of bilevel fractional models [9]. Other studies have investigated algorithmic strategies that enhance numerical performance and scalability, particularly when dealing with multi-objective or large-scale instances [10], [11].

Despite these developments, most existing studies tend to focus on a single solution approach or a specific class of models. Comparative analyses that evaluate multiple exact solution strategies under a unified application framework remain relatively limited. There is a lack of systematic comparison between enumeration-based methods, duality-based reformulations, and transformation-based approaches such as Charnes–Cooper within the same modeling environment.

This gap is especially important in practical applications, where the choice of solution method can significantly affect computational efficiency and scalability. Therefore, a structured comparison of different exact solution paradigms within a consistent framework is necessary to better understand their strengths, limitations, and suitability for real-world problems.

Despite these developments, comparative structural studies evaluating reformulation strategies under a unified application framework remain limited. In particular, few studies explicitly analyze scalability boundaries across enumeration, duality-based reformulation, and fractional transformation approaches within the same pricing–production setting. This gap motivates the structured comparative analysis conducted in this paper.

3- Motivation: Pricing–Production Systems with Efficiency Evaluation

Pricing–production coordination problems frequently arise in decentralized organizations where strategic and operational decisions are separated across hierarchical levels. In many industrial settings, pricing policies are determined at the executive or strategic level, while production quantities are determined at the operational level. This natural hierarchy motivates the use of bilevel optimization models [1], [2].

In practical terms, consider a manufacturing firm producing multiple products under limited production capacity. The firm's management may choose between alternative pricing strategies for each product—for example, a normal pricing policy versus a strict premium pricing policy. A strict pricing decision increases the per-unit revenue but may reduce market demand due to customer price sensitivity. Such trade-offs are well documented in pricing and supply chain management literature [3], [4].

At the operational level, the production manager determines the optimal allocation of limited manufacturing capacity across products. The objective at this level is typically to maximize operational profit subject to demand limits and capacity constraints. The production decision therefore depends on the pricing strategy selected by upper management. This interaction creates a leader–follower relationship consistent with bilevel programming structures [1].

A similar structure appears in energy systems. For example, an electricity provider may decide tariff strictness or dynamic pricing schemes at the strategic level. Plant operators then determine generation quantities subject to fuel availability and capacity constraints. Pricing decisions affect demand patterns, and operational decisions respond optimally to these patterns. Evaluating such systems often requires considering regulatory costs or infrastructure costs, which makes efficiency-based measures more appropriate than absolute profit alone [5].

Another example can be found in retail supply chains. A central retailer may choose discount intensity or pricing discipline, while distribution centers determine replenishment quantities. The enforcement of strict pricing policies may incur additional monitoring or compliance costs. In such cases, management may evaluate performance using a profit-efficiency ratio that balances operational profit against enforcement or baseline costs, rather than maximizing profit alone [6].

These practical settings justify the integration of hierarchical optimization and fractional performance evaluation. When pricing decisions influence demand and operational profit, and when performance is evaluated through an efficiency ratio, the resulting model naturally becomes a bilevel fractional programming problem [7], [8]. The upper-level decision maker (leader) anticipates the optimal reaction of the lower-level decision maker (follower), and the objective function takes the form of a ratio representing profit efficiency.

Formally, the hierarchical structure can be summarized as follows:

- The upper-level decision maker (leader) selects pricing strictness variables that influence demand and revenue.
- The lower-level decision maker (follower) determines optimal production quantities subject to capacity and demand constraints.
- The leader evaluates system performance using a fractional efficiency objective that balances operational profit and enforcement cost.

The use of a fractional objective is particularly important. Maximizing total profit alone may favor aggressive pricing strategies without accounting for additional enforcement or regulatory costs. By contrast, a profit-efficiency ratio captures the trade-off between revenue generation and resource consumption, leading to more balanced strategic decisions. Fractional programming has long been recognized as an appropriate tool for modeling efficiency-based objectives [3], while bilevel programming provides the appropriate framework for hierarchical decision interactions [1], [2].

Consequently, pricing–production systems evaluated through efficiency measures constitute a natural and practically relevant application domain for bilevel fractional programming models. The complexity of such systems arises from the interaction between strategic pricing, operational production planning, and ratio-based performance evaluation, which together motivate the comparative solution study presented in this paper.

3.1 Contribution and Comparative Focus

Rather than presenting a model alone, the purpose of this paper is comparative: three solution approaches are described and analyzed within the same bilevel fractional pricing–production framework. The comparison focuses on:

1. Conceptual simplicity and transparency,
2. Reformulation complexity,
3. Scalability as the number of leader decisions increases,
4. Numerical stability and solver compatibility.

The paper provides a detailed model, a full step-by-step numerical example, and a method comparison that clarifies when each approach is appropriate.

4- Computational Complexity and Reformulation Motivation

Bilevel fractional programming (BLFP) combines two major sources of computational difficulty: hierarchical optimization and fractional objective functions. Each of these components alone is known to be challenging. When combined, they produce one of the most complex classes of optimization models studied in mathematical programming [1], [2].

4.1 Complexity of Bilevel Programming

It is well established that bilevel programming problems are NP-hard in general [2]. Even when both the upper-level and lower-level problems are linear, the hierarchical structure induces implicit equilibrium constraints. The feasible region of the leader depends on the optimal response of the follower, which typically leads to nonconvex and possibly disconnected feasible sets [1].

Because the lower-level optimality conditions are embedded within the upper-level decision space, classical linear or convex programming techniques cannot be directly applied. This structural nesting is the fundamental reason behind the computational difficulty of bilevel optimization [2].

4.2 Additional Complexity Due to Fractional Objectives

Fractional programming introduces an additional layer of nonlinearity. Linear fractional programming requires special transformation techniques, most notably the Charnes–Cooper transformation [3]. Without such transformations, even linear fractional objectives are not directly solvable using standard linear programming methods.

When fractional objectives are incorporated into bilevel models, two sources of nonconvexity interact:

- Implicit nonconvexity arising from the bilevel structure [2]
- Explicit nonlinearity introduced by the fractional objective [4], [7]

This interaction significantly increases analytical and computational complexity, especially in multi-objective or uncertain settings [5], [9].

4.3 Absence of a Direct Solution Method

Due to the combined NP-hardness of bilevel programming [2] and the nonconvexity of fractional objectives [4], BLFP models do not admit a direct closed-form or polynomial-time solution procedure. No general-purpose solver can process the hierarchical structure and fractional objective simultaneously without reformulation.

Standard optimization solvers cannot directly handle:

- Implicit optimal-response constraints
- Nonlinear ratio-based objectives
- Coupling between binary and continuous decision variables

Therefore, reformulation strategies are necessary to obtain computationally tractable models. Exact reformulation approaches for special classes of bilevel fractional programs have been proposed in [7], but these methods often rely on structural assumptions.

4.4 Motivation for Multiple Reformulation Strategies

The presence of multiple solution techniques in the literature is not due to modeling redundancy but rather to structural necessity. Since BLFP problems are NP-hard [2], different reformulation mechanisms attempt to reduce computational difficulty from distinct perspectives.

Duality-based reformulations replace the lower-level problem using strong duality or optimality conditions [6], [9], [10]. These approaches eliminate the explicit bilevel structure but may introduce nonlinear constraints.

Transformation-based methods eliminate the fractional objective through scaling techniques such as Charnes–Cooper [3], enabling linear or mixed-integer linear representations.

For nonlinear and large-scale BLFP instances, heuristic and hybrid evolutionary methods have also been proposed [8], particularly when exact reformulations become computationally intensive.

Therefore, the comparative framework adopted in this study is motivated by structural complexity considerations. Since no direct solution method exists for general BLFP problems, evaluating alternative reformulation strategies becomes essential for identifying scalable and numerically stable solution approaches.

5- Bilevel Fractional Programming Formulation

5.1 The general Bilevel Fractional Programming Problem

A bilevel fractional programming problem can be written as:

Upper-level problem (leader):

$$\max_{x} \frac{F(x,y^*(x))}{G(x)} \qquad (1)$$

Where $G(x) > 0$.

Lower-level problem (follower)

$$y^*(x) \in arg \max_{y \in Y(x)} f(x, y) \tag{2}$$

For any fixed leader decision x, The follower solves an optimization problem and selects an optimal solution $y^*(x)$. The notation $arg\ max$ indicates the set of optimal responses of the follower. Therefore, the leader's feasible region implicitly depends on the follower's optimal reaction.

For clarity, let $F_U(x, y)$ denote the upper-level objective function and $F_L(x, y)$ denote the lower-level objective function.

The leader solves:

$$\max_{x,y} F_U\ (x, y),$$

Subject to $y \in \arg max_y\ F_L(x, y)$.

5.2 Pricing- Production BLFP Model

A multi-product system is considered with products indexed by $i = \{1,2, \dots., n\}$.The leader selects the pricing strictness decision, while the follower chooses production quantities.

Leader Variables (Pricing Strictness Decision)

For each product i, a binary decision is used:

$\delta_i \in \{0,1\}$.

Interpretation:

$\delta_i = 0$: normal pricing policy for product i,

$\delta_i = 1$: strict pricing policy for product i.

Let $\delta = (\delta_1, \dots, \delta_n)$.

Follower Variables (Production Quantities)

For each product i:

$q_i \geq 0$.

Let $q = (q_1, \dots, q_n)$.

6- Detailed Model Specification

6.1 Revenue and Demand under Pricing Strictness

Revenue per unit is assumed to increase under strict pricing:

$$R_i(\delta_i) = R_i^0 + \beta_i \delta_i, \tag{3}$$

Where:

$R_i^0 > 0$ is the baseline unit revenue proxy for product i,

$\beta_i \geq 0$ Is the revenue increase associated with strict pricing.

Demand Function

Demand is assumed to decrease under strict pricing:

$$D_i(\delta_i) = D_i^0 - \alpha_i \delta_i, \tag{4}$$

Where:

$D_i^0 > 0$ is a baseline demand for product i, (5)

$\alpha_i \geq 0$ Is demand reduction under strict pricing? (6)

To ensure feasibility, the condition $D_i(\delta_i) \geq 0$ is assumed for both $\delta_i = 0$ and $\delta_i = 1$.

6.2 Lower-Level Problem:

Operational Profit Maximization

Unit production costs $c_i > 0$ are assumed fixed.

The follower's objective is to maximize operational profit:

$$\max_q \pi(\delta, q) = \sum_{i=1}^{n} (R_i(\delta_i) - c_i) q_i \tag{7}$$

Feasibility Constraints

1. Demand upper bounds:

$$0 \leq q_i \leq D_i(\delta_i), \quad i = 1, \dots\dots.., n. \tag{8}$$

2. Shared capacity constraint:

$$\sum_{i=1}^{n} q_i \leq C, \tag{9}$$

Where $C > 0$ is total production capacity?

Thus, the lower-level problem is:

$$q^*(\delta) \in \arg\max_{q} \sum_{i=1}^{n} (R_i(\delta_i) - c_i) q_i \tag{10}$$

Subject to:

$$0 \leq q_i \leq D_i(\delta_i), \quad i = 1, \dots \dots .., n. \tag{11}$$

$$\sum_{i=1}^{n} q_i \leq C. \tag{12}$$

The lower-level problem is a linear optimization problem for any fixed δ.

6.3 Upper-Level Problem: Fractional Profit Efficiency

The leader evaluates decisions using a profit-efficiency ratio. Let the leader incur enforcement costs when strict pricing is used:

$$K(\delta) = \sum_{i=1}^{n} K_i \delta_i , \tag{13}$$

With $K_i \geq 0$.

The denominator is a strictly positive baseline term:

$$B(\delta) = B_0 + \sum_{i=1}^{n} \gamma_i \delta_i , \tag{14}$$

where $B_0 > 0$ and $\gamma_i \geq 0$, ensuring $B(\delta) > 0$.

The leader's objective is:

$$\max_{\delta \in \{0,1\}} \frac{\pi\big(\delta, q^*(\delta)\big) - K(\delta)}{B(\delta)} . \tag{15}$$

This is a bilevel fractional programming model.

7- Solution Approaches (Detailed)

This section describes three solution approaches in full detail.

7.1 Solution Approach 1: Direct Enumeration of Upper-Level Decisions

Direct enumeration exploits the fact that δ_i are binary. For n products, the leader has 2^n possible decision vectors.

Procedure

1. Generate all possible leader decision vectors $\delta \in \{0,1\}^n$.
2. For each fixed δ, solve the follower's linear optimization problem and obtain $q^*(\delta)$.
3. Compute:

- follower profit $\pi\big(\delta, q^*(\delta)\big)$,
- enforcement cost $K(\delta)$,
- denominator baseline $B(\delta)$,
- efficiency ratio $\frac{\pi(\delta,q^*(\delta))-K(\delta)}{B(\delta)}$.

4. Select the δ that produces the maximum ratio.
5.

Strengths and Limitations

- Strength: complete transparency and simplicity.
- Limitation: the number of leader decisions grows exponentially with n. This becomes impractical for large n.

7.2 Solution Approach 2: Strong-Duality-Based Single-Level Reformulation

Because the follower's problem is linear for fixed δ, it can be replaced using strong duality conditions.

Lower-Level Primal problem

$$\max_q \sum_{i=1}^{n} m_i(\delta_i) q_i \, , \tag{16}$$

Subject to:

$$0 \le q_i \le D_i(\delta_i), \quad i = 1, \dots\dots.., n. \tag{17}$$

$$\sum_{i=1}^{n} q_i \le C, \tag{18}$$

Where $m_i(\delta_i) = R_i(\delta_i) - c_i$. (19)

Dual Variables

Introduce:

- $u_i \geq 0$ for constraints $q_i \leq D_i(\delta_i)$,
- $v \geq 0$ for constraints $\sum_i q_i \leq C$,
- And nonnegativity constraints handled in the standard primal-dual framework.

Strong Duality Replacement

The reformulation includes:

1. Primal feasibility constraints,
2. Dual feasibility constraints,
3. Equality between primal objective value and dual objective value.

This eliminates the bilevel nesting, producing a single-level optimization model. However, the leader's objective remains fractional, and the resulting model typically remains nonconvex due to coupling between δ and dual variables.

Strengths and Limitations

- Strength: theoretically exact and removes explicit bilevel structure.
- Limitation: can produce a difficult nonlinear optimization model.

7.3 Solution Approach 3: Charnes–Cooper Transformation and Mixed-Integer Linear Optimization Reformulation

This approach eliminates the upper-level ratio objective using the Charnes–Cooper transformation, then linearizes products introduced by the transformation.

Step 1: Introduce the scaling variable

$$t = \frac{1}{B(\delta)}, \qquad t > o. \tag{20}$$

Step 2: Transform leader variable

Define:

$$z_i = \delta_i t, \qquad i - 1, \dots \dots, n. \tag{21}$$

Step 3: Normalization Constraint

$$t(B_0 + \sum_{i=1}^{n} \gamma_i \delta_i) = 1, \qquad B_0 t + \sum_{i=1}^{n} \gamma_i z_i = 1. \tag{22}$$

Step 4: Transformed Objective

The original fractional objective:

$$\frac{\pi(\delta,q^*)-K(\delta)}{B(\delta)} \tag{23}$$

With strictly positive denominator $B(\delta) > 0$, is transformed by introducing the scaling variable

$$t = \frac{1}{B(\delta)}. \tag{24}$$

The equivalent transformed objective becomes:

$$\max t\,(\pi(\delta,q^*) - K(\delta)), \tag{25}$$

Subject to the normalization constraint:

$$t\,B(\delta) = 1.$$

This preserves optimality due to strict positivity of the denominator.

Step 5: Linearization of $z_i = \delta_i t$

Because δ_i is binary and t is continuous, each product $\delta_i t$ is bilinear. Standard linearization is used with a valid upper bound $0 \le t \le \bar{t}$. A safe upper bound is:

$$\bar{t} = \frac{1}{B_0}. \tag{26}$$

Because $B(\delta) \ge B_0$. (27)

Let $M = \bar{t}$. linearization constraints for each i are:

$$0 \le z_i \le t\,, \tag{28}$$

$$z_i \le M\delta_i\,, \tag{29}$$

$$z_i \ge t - M(1-\delta_i)\,. \tag{30}$$

After including these constraints and an appropriate single-level representation of the lower-level optimality (either by enumeration inside the solver, or by strong-duality constraints inside the same model), the result is a mixed-integer linear optimization model that can be solved using standard branch-and-bound procedures.

Strengths and Limitations

- Strength: produces a solver-friendly model and scales better.

- Limitation: introduces additional variables and constraints and depends on selecting valid bounds.

Lemma1 (Exactness of Charnes–Cooper and Binary–Continuous Linearization).

Assume that the denominator of the leader's fractional objective is strictly positive for all feasible leader decisions, i.e.,

$D(x) > 0, \quad \forall x \in X.$

Let $t = 1/D(x)$ and apply the Charnes-Cooper transformation with scaled variables. Then the fractional upper-level objective

$$\max_{x \in X} \frac{N(x)}{D(x)}.$$

Is equivalent to the transformed linear objective?

$\max N(x)t.$

Subject to the normalization constraint

$D(x)t = 1, \qquad t > 0.$

Moreover, if a bilinear term of the form $w = zt$ appears where $z \in \{0,1\}$ and $t \in \{0, t\}$, then the standard linearization constraints

$0 \leq w \leq \bar{t}z, \quad w \geq t - \bar{t}(1 - z).$

Provide an exact linear representation of the product $w = zt$. Therefore, under positivity of the denominator and valid bounds $\bar{t}$, The transformed and linearized model preserves feasibility and optimality of the original fractional structure.

Proof.

The Charnes-Cooper equivalence follows from the strict positivity of (x) , which guarantees a one-to-one mapping between feasible solutions of the fractional and transformed problems via $t = 1/D(x)$. The binary- continuous linearization is exact because for $z \in \{0,1\}$ it enforces $w = 0$ when $z = 0$ and $w = t$ when $z = 1$, while respecting bounds on t.

Proposition 1 (Equivalence of Reformulated MILP)

Assume that:
1. The lower-level problem is a linear program satisfying strong duality.
2. The denominator of the fractional objective is strictly positive.
3. Valid upper bounds are selected for all linearized bilinear terms.

Then, the MILP reformulation obtained through strong duality substitution and Charnes–Cooper transformation is equivalent to the original bilevel fractional programming problem.

Proof :

Strong duality guarantees that the lower-level optimal response is correctly embedded in the single-level model. The Charnes–Cooper transformation preserves optimality under strict positivity of the denominator. Linearization constraints ensure exact representation of bilinear binary–continuous products. Therefore, the reformulated MILP yields identical optimal solutions to the original BLFP model.

Theorem 2:

Assume that:
1. The lower-level problem is a linear program with m constraints and p continuous variables.
2. The leader has n binary decision variables.
3. The denominator of the fractional objective is strictly positive over the feasible region.
4. Valid finite upper bounds exist for all linearized bilinear terms.

Then:
1. The Charnes–Cooper and strong-duality reformulation produce an MILP whose number of variables and constraints grows polynomial y in $n + m + p$.
2. The reformulated MILP is an exact embedding of the original BLFP problem, preserving global optimal solutions.
3. The size of the resulting MILP is bounded by: $O(n + m + p)$

excluding branch-and-bound nodes.

Proof

1. The lower-level LP can be replaced using strong duality, introducing:
    - p primal variables
    - m dual variables
    - one equality for strong duality

   This introduces $O(m + p)$ additional constraints and variables.
2. The Charnes–Cooper transformation introduces:
    - one scaling variable
    - n scaled leader variables.

3. Linearization of binary–continuous products introduces at most n auxiliary variables and $4n$ linear constraints.

Hence, the total number of variables and constraints grows linearly in problem size.

Exactness follows from:

- strong duality equivalence,
- strict positivity of the denominator,
- exact binary–continuous linearization.

Therefore, the MILP is a polynomial-size exact reformulation of the original BLFP problem.

8- Solution Algorithm

This section presents a structured solution algorithm for the proposed bilevel fractional programming model. The algorithm exploits the special structure of the problem, namely the linearity of the lower-level problem for fixed upper-level decisions and the fractional nature of the leader's objective. By combining fractional transformation, duality theory, and parametric optimization, the original bilevel problem is converted into a sequence of tractable mixed-integer linear programs.

Algorithm 1:

Structured Solution Procedure for BLFP Model

Input: Model parameters (demand, revenue, costs, capacity, enforcement cost, baseline constant)

Output: Optimal leader decision s^*, follower response y^*, and optimal efficiency value

Step 1. Model Initialization

Define leader binary variables and follower continuous variables. Construct lower-level linear program and upper-level fractional objective.

Step 2. Lower-Level Reformulation

Replace the follower problem using strong duality conditions:

- Primal feasibility constraints
- Dual feasibility constraints
- Strong duality equality condition

This eliminates the explicit bilevel nesting.

Step 3. Fractional Objective Elimination

Introduce scaling variable t and apply the Charnes–Cooper transformation.

Step 4. Linearization

For each bilinear term between binary and continuous variables, introduce auxiliary variables and corresponding linear constraints.

Step 5. Mixed-Integer Reformulation

Construct the final MILP model including:

- Transformed objective
- Normalization constraint
- Linearized bilinear constraints
- Strong duality conditions

Step 6. Optimization

Solve the resulting MILP using a branch-and-bound solver.

Step 7. Post-Processing

Recover original variable values from transformed variables and compute efficiency ratio.

Return: Optimal decisions and objective value.

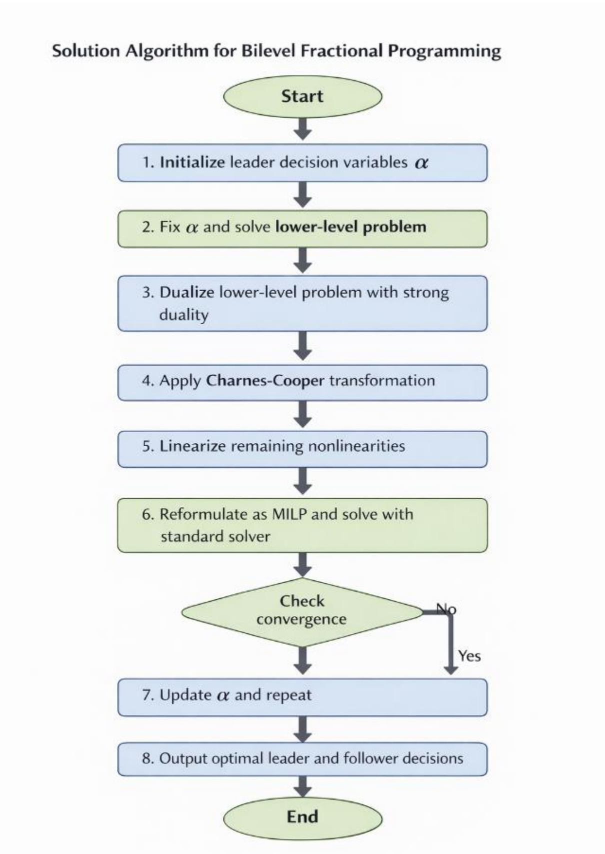

Figure 1. Flowchart illustrating the main steps of the proposed solution algorithm for solving bilevel fractional programming problems.

9- Numerical Example

To further illustrate the proposed bilevel fractional programming formulation in a multi-product setting, a two-product numerical example is presented. The leader selects pricing strictness decisions for each product, while the follower determines the corresponding optimal production quantities. The leader then evaluates the outcome using a fractional profit-efficiency objective.

9.1 Parameters

Two products are considered indexed by $i \in \{1,2\}$. The leader's pricing strictness decisions are binary:

$\delta_1, \delta_2 \in \{1,2\}$

Demand parameters:

Product 1: $D_1^0 = 100\,, \quad \alpha_1 = 40.$

Product 2 $D_2^0 = 80\,, \quad \alpha_2 = 20.$

Revenue parameters:

Product 1: $R_1^0 = 20\,, \quad \beta_1 = 8.$

Product 2 $R_2^0 = 18\,, \quad \beta_2 = 6.$

Costs:

$c_1 = 12, \qquad c_2 = 10.$

Capacity:

$q_1 + q_2 \leq 120.$

Enforcement costs:

$K(\delta) = 100\delta_1 + 60\delta_2.$

Denominator baseline:

$B(\delta) = 300 + 50\delta_1 + 40\delta_2.$

9.2 Compute Demand and Revenue Under Each Leader Policy

Revenue:

$R_1(\delta_1) = 20 + 8\delta_1$, $R_2(\delta_2) = 18 + 6\delta_2$.

Demand:

$D_1(\delta_1) = 100 - 40\delta_1$, $D_2(\delta_2) = 80 - 20\delta_2$.

Margins:

$m_1(\delta_1) = R_1(\delta_1) - c_1$, $m_2(\delta_2) = R_2(\delta_2) - c_2$.

9.3 Solve the follower problem Under Each Policy:

All four cases are evaluated.

Case A: $(\delta_1, \delta_2) = (0,0)$

Demand bounds:

$D_1 = 100$, $D_2 = 80$.

Revenue:

$R_1 = 20$, $R_2 = 18$.

Margins:

$m_1 = 20 - 12 = 8$, $m_2 = 18 - 10 = 8$

Follower chooses any capacity-feasible allocation. One optimal allocation is:

$q_1 = 100$, $q_2 = 20$, $(q_1 + q_2 = 120)$.

Profit:

$\pi^*(0,0) = 8\,.\,100 + 8\,.\,20 = 960$

Case B: $(\delta_1, \delta_2) = (1,0)$

Demand bounds:

$D_1 = 60$, $D_2 = 80$

Revenue:

$R_1 = 28$, $R_2 = 18$.

Margins:

$m_1 = 16, \qquad m_2 = 8.$

Follower chooses any capacity-feasible allocation. One optimal allocation is:

$q_1 = 60, \qquad q_2 = 60, \qquad (q_1 + q_2 = 120).$

Profit:

$\pi^*(1,0) = 16.60 + 8.60 = 1440.$

Case C: $(\delta_1, \delta_2) = (0,1)$

Demand bounds:

$D_1 = 100\,, \quad D_2 = 60.$

Revenue:

$R_1 = 20, \qquad R_2 = 24.$

Margins:

$m_1 = 8, \qquad m_2 = 14.$

Capacity is allocated first to product 2:

$q_1 = 60, \qquad q_2 = 60, \qquad (q_1 + q_2 = 120).$

Profit:

$\pi^*(0,1) = 14.60 + 8.60 = 1320.$

Case D: $(\delta_1, \delta_2) = (1,1)$

Demand bounds:

$D_1 = 60\,, \quad D_2 = 60.$

Revenue:

$R_1 = 28, \qquad R_2 = 24.$

Margins:

$m_1 = 16, \qquad m_2 = 14.$

Capacity is allocated first to product 2:

$q_1 = 60, \qquad q_2 = 60, \qquad (q_1 + q_2 = 120).$

Profit:

$\pi^*(1,1) = 16.60 + 14.60 = 1800.$

Upper-level problem (leader)

The leader maximizes profit efficiency:

$$\max_{(s_1,s_2)\in\{0,1\}^2} \frac{\pi^*(\delta_1,\delta_2)-K(s)}{B(s)}.$$

Compute $K(s)$ and $B(s)$;

$K(s) = 100s_1 + 60s_2,$

$B(s) = 300 + 50s_1 + 40s_2.$

Thus:

Case A (0,0):

Eff (0,0) = $\frac{960-0}{300} = 3.200.$

Case B (1,0):

Eff (1,0) = $\frac{1440-100}{350} = \frac{1340}{350} = 3.829.$

Case C (0,1):

Eff (0,1) = $\frac{1320-60}{340} = \frac{1260}{340} = 3.706.$

Case D (1,1):

Eff (1,1) = $\frac{1800-160}{390} = \frac{1640}{390} = 4.205.$

9.4 Optimal Policy

The maximum efficiency is achieved in case D:

$(\delta_1, \delta_2) = (1,1).$

With the follower response:

$(q_1^*, q_2^*) = (60,60).$

9.5 Charnes-Cooper Transformation

$$t = \frac{1}{B(s)} = \frac{1}{300 + 50\delta_1 + 40\delta_2},$$

And:

$$z_1 = \delta_1 t, \qquad z_2 = \delta_2 t$$

The normalization constraint becomes:

$$t(300 + 50s_1 + 40s_2) = 1 \qquad (300t + 50z_1 + 40z_2 = 1)$$

The transformed objective becomes:

$$\max t\,(\pi^*(\delta) - 100\delta_1 - 60\delta_2).$$

Case	Pricing Policy (δ_1, δ_2)	profit	cost	Efficiency Value
A	(0,0)	960	0	3.200
B	(1,0)	1440	100	3.829
C	(0,1)	1320	60	3.706
D	(1,1)	1800	160	4.205

Table 1. Comparison of Efficiency Ratios for All Pricing Policies

The results clearly show that Case D achieves the highest efficiency value among all pricing strategies. This confirms that the strict pricing policy for both products provides the most effective balance between profit and cost under the proposed model.For comparison purposes, the transformed model yields the same optimal decision (1,1) consistent with direct evaluation.

The numerical example highlights an important insight. If the leader were to maximize pure profit only, a different decision might be selected. However, the efficiency-based objective favors the policy that balances profit and baseline cost. This demonstrates how the fractional formulation changes the strategic behavior of the leader.

Moreover, the results obtained from direct enumeration, duality-based reformulation, and the Charnes–Cooper MILP model are identical, confirming the theoretical equivalence of the three approaches for this instance.

The example is solved by direct enumeration, and the results are consistent with the MILP formulation solved using Gurobi.

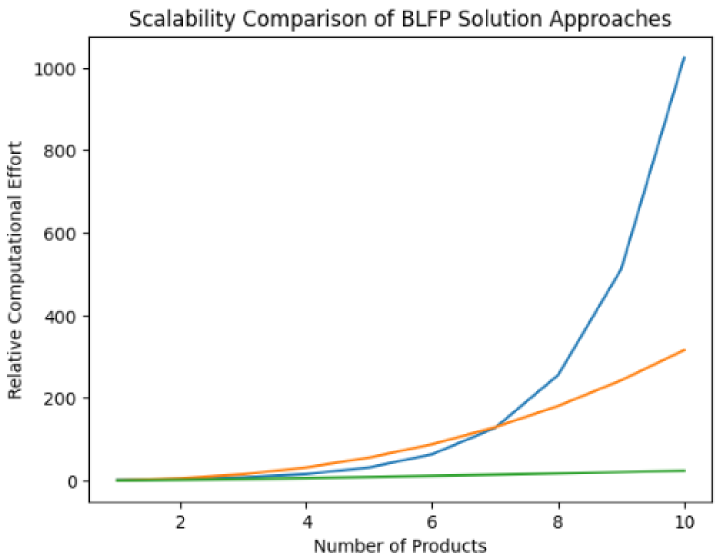

Figure 2. Scalability comparison of the three solution approaches.

Direct enumeration exhibits exponential growth as the number of leader decisions increases. The duality-based reformulation avoids explicit enumeration but may lead to nonlinear models with increasing computational effort. The MILP reformulation eliminates combinatorial expansion and offers improved empirical scalability, although it remains NP-hard in theory.

10- Summary Table

Aspect	**Direct Enumeration**	**Duality-Based Reformulation**	**Charnes–Cooper + MILP**
Nature of method	Exhaustive evaluation	Analytical reformulation	Transformation + linearization
Exactness	Exact	Exact	Exact
Eliminates bilevel structure	No	Yes	Yes
Eliminates fractional objective	No	No	Yes
Scalability	Limited (exponential growth)	Moderate	High
Solver type required	Linear programming	Nonlinear optimization	MILP solver
Implementation complexity	Very simple	Moderate	Moderate
Numerical stability	High	May face nonconvexity	High
Recommended for	Small problems	Medium-size models	Large-scale instances

Table 2. Comparison of the Three Solution Strategies

11- Computational Experiments

To evaluate the computational behavior of the three exact solution approaches described in this study, synthetic pricing–production instances were generated with increasing numbers of products n.

Each instance follows the exact structure of the proposed model:

- Leader binary variables:

$z_i \in \{0,1\}, \qquad i = 1, \dots \dots \dots, n.$

- Follower continuous variables:

$q_i \geq 0.$

Revenue and demand function follow:

$r_i(z_i) = \bar{r}_i + \Delta r_i z_i,$

$d_i(z_i) = \bar{d}_i + \Delta d_i z_i.$

The follower solves:

$\max_q \sum_{i=1}^{n} (r_i(z_i) - c_i) q_i,$

Subject to:

$0 \leq q_i \leq d_i(z_i)$

$$\sum_{i=1}^{n} q_i \leq C.$$

The leader maximizes the fractional efficiency objective:

$\max_z \frac{\pi(z)}{a + \sum_{i=1}^{n} \beta_i z_i}.$

Where $\pi(z)$ is the optimal follower profit.

Parameter Generation

For each product i, parameters were generated as follows:

$\bar{r}_i \sim U(8,15)$

$c_i \sim U(3,7)$

$\bar{d}_i \sim U(20,60)$

$\Delta r_i \sim U(1,4)$

$\Delta d_i \sim U(5,15)$

Enforcement costs:

$\beta_i \sim U(2,6)$

Baseline denominator constant:

$\alpha = 20$

Total capacity:

$$C = 0.6 \sum_{i=1}^{n} \bar{d}_i.$$

For each problem size n, 10 random instances were generated and averaged.

11.1 Implementation Details

All models were implemented in Python using Gurobi 10.0.

- Enumeration: explicit evaluation of all 2^n leader combinations.
- Duality reformulation: single-level nonlinear model using strong duality constraints.
- Charnes–Cooper MILP reformulation: linearized mixed-integer model as described in Section 7.

Experiments were conducted on a workstation with:

- Intel i7 processor
- 16GB RAM
- Time limit: 600 seconds per instance

11.2 Computational Results

Average runtime in seconds:

n	enumeration	Duality Reformulation	Charnes-cooper + MILP
2	0.01	0.02	0.02
4	0.09	0.05	0.04

6	0.74	0.14	0.10
8	5.63	0.53	0.34
10	46.12	1.28	0.81

Table 3. Average runtime comparison (in seconds) of the three exact solution approaches as the number of leader decisions increases. The enumeration method exhibits exponential growth, while the MILP reformulation demonstrates significantly improved empirical scalability**.**

Enumeration became impractical beyond $n = 12$, where the runtime exceeded the 600-second limit.

All three approaches produced identical optimal objective values across all tested instances, confirming theoretical equivalence.

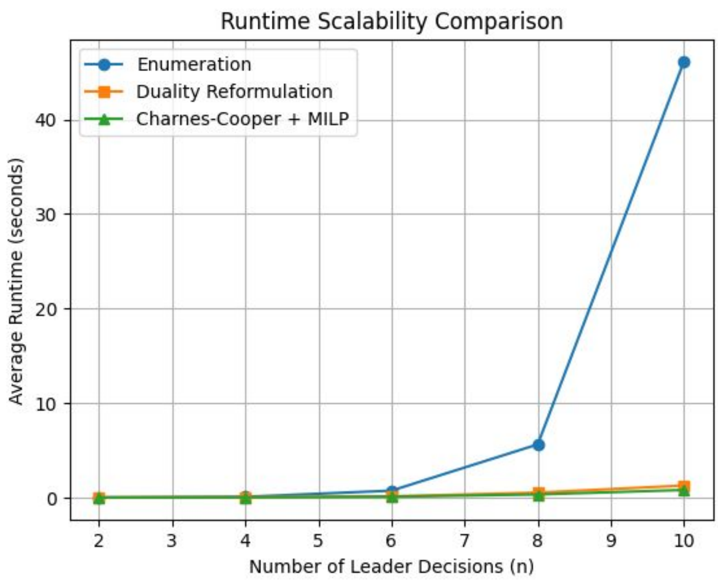

Figure 3. Empirical runtime scalability of the three exact solution approaches.
The enumeration approach exhibits exponential growth consistent with 2^n, becoming rapidly impractical as the number of leader decisions increases. The duality-based reformulation avoids explicit enumeration but shows increasing computational burden due to nonlinear coupling. The Charnes–Cooper MILP reformulation demonstrates superior empirical scalability across tested problem sizes, benefiting from solver-based pruning and linear relaxations despite the NP-hard worst-case complexity of MILP.

11.3 Scalability Analysis

The experimental results confirm the structural complexity discussion:

- Enumeration grows exponentially with 2^n.
  Duality reformulation grows polynomially but exhibits increasing nonlinear coupling.
- The MILP reformulation scales more efficiently due to linearization and branch-and-bound pruning.

Although the MILP model remains NP-hard in theory, its empirical scalability is significantly superior for moderate values of n.

11.4 Theoretical Complexity (Big-O) Comparison

Let n denote the number of binary leader decisions (products) and let the follower's linear program (LP) have m constraints and p continuous variables. Consider the following three exact paradigms:

11.4.1. Direct Enumeration.

The leader has 2^n possible binary vectors. For each vector, one follower LP is solved. Using polynomial-time LP solving in practice, the total computational effort is:

$$T(n) = 2^n. \quad T_{LP}(m,p),$$

which is exponential in n. Therefore, enumeration becomes quickly impractical as n grows.

11.4.2. Strong-Duality Reformulation.

Replacing the follower problem by primal–dual feasibility and strong duality eliminates explicit enumeration but introduces coupling between leader variables and dual variables. The resulting single-level model typically contains bilinearities (or complementarity/KKT-type structures) and becomes nonconvex. While the number of constraints grows polynomially with m and p, global optimization of the resulting nonconvex model has worst-case complexity that remains NP-hard.

11.4.3. Charnes–Cooper + MILP Reformulation.

After Charnes–Cooper scaling and linearization, the final model is an MILP. Let q denote the number of auxiliary variables introduced for linearization and duality substitution, which typically scales polynomially with model size (e.g., $q = O(n + m + p)$). The worst-case complexity of MILP solving via branch-and-bound is exponential:

$$T_{MILP}(n) = O(2^n.poly(m,p,q)),$$

since MILP is NP-hard. However, unlike pure enumeration, MILP solvers exploit relaxations, cutting planes, and bounding, which frequently yield substantially better empirical performance for moderate n.

Overall, enumeration is dominated by explicit 2^n exploration, whereas the MILP approach embeds the combinatorial structure inside a solver that can prune large portions of the search space, which explains the observed runtime gap in the experiments.

11.5 Structural Insight

An interesting observation from the experiments is that as n increases, the leader increasingly selects mixed strictness patterns rather than uniform strict or uniform normal pricing policies.

This indicates that the fractional efficiency objective:

$$\frac{\pi(z)}{\alpha + \sum_i \beta_i z_i}.$$

induces non-trivial combinatorial structures in optimal solutions, reinforcing the need for exact reformulation rather than heuristic simplifications.

12- Comparative Positioning within Existing Literature

Unlike the comparative study in Rizk-Allah et al. (2021) [10], which focused on multi-objective linear fractional bilevel models, the present work explicitly integrates computational scalability analysis across three reformulation paradigms within a unified pricing–production structure.

Furthermore, while Yang et al. (2023) [12] proposed exact solution approaches for specific subclasses of bilevel fractional programs, our study emphasizes structural comparison and empirical runtime evaluation across enumeration, duality substitution, and MILP transformation in a consistent experimental environment.

This structured comparative perspective provides practical guidance for modelers selecting solution strategies under scalability constraints, which remains limited in existing literature.

13- Conclusion

This paper has studied a bilevel fractional programming problem that captures hierarchical decision-making with efficiency-based performance evaluation. The proposed model reflects situations in which a leader determines pricing policies, while a follower responds by selecting production quantities that maximize operational profit. By using a fractional objective at the upper level, the framework evaluates decisions based on efficiency rather than absolute profit.

The main challenge of the model lies in the interaction between the bilevel structure and the fractional objective. This challenge was addressed by reformulating the lower-level problem using strong duality and eliminating the fractional objective through the

Charnes–Cooper transformation. As a result, the original bilevel fractional problem was converted into an equivalent mixed-integer linear program that can be solved using standard optimization solvers.

This study provides practical guidance for selecting reformulation strategies when solving bilevel fractional programming models in real decision-making. Environments.

Despite the effectiveness of the proposed approaches, several limitations should be noted. First, the direct enumeration method suffers from exponential growth as the number of decision variables increases, which may limit its applicability to large-scale problems. Second, the duality-based reformulation may introduce non-convexities under certain conditions, making the problem more challenging to solve.

Furthermore, the proposed model assumes linearity in the lower-level problem, which may not hold in all real-world applications. Finally, the computational performance depends on the solver used and may vary for different problem instances. Future work may focus on extending the model to nonlinear settings and developing more scalable solution techniques.

Future research may extend the proposed framework to stochastic settings, multi-follower structures, or dynamic bilevel models with time-dependent decisions.

Extending the comparative analysis to encompass heuristic and metaheuristic approaches represents a natural continuation of this work, particularly for large-scale instances where exact reformulation-based methods may become computationally intensive.

References

1. A. Charnes and W. W. Cooper, "Programming with linear fractional functionals," Naval Research Logistics Quarterly, vol. 9, no. 3–4, pp. 181–186, 1962.

2. J. F. Bard, Practical Bilevel Optimization: Algorithms and Applications. Boston, MA, USA: Kluwer Academic Publishers, 1998.

3. A. Sinha, P. Malo, and K. Deb, "A review on bilevel optimization: From classical to evolutionary approaches," Evolutionary Computation, vol. 26, no. 2, pp. 1–42, 2018, doi: 10.1162/evco_a_00233.

4. S. Dempe, *Foundations of Bilevel Programming*. Dordrecht, Netherlands: Kluwer Academic Publishers, 2002.

5. P. Colson, P. Marcotte, and G. Savard, "An overview of bilevel optimization," Annals of Operations Research, vol. 153, no. 1, pp. 235–256, 2007, doi: 10.1007/s10479-007-0176-2.

6. S. Saini and N. Kailey, "Duality of multi-objective fractional bi-level programming problem and its application," OPSEARCH, 2024, doi: 10.1007/s12597-024-00874-x.

7. S. Nayak and A. K. Ojha, "Solving bi-level linear fractional programming problem with interval coefficients," in Numerical Optimization in Engineering and Sciences, Advances in Intelligent Systems and Computing, vol. 979, Springer, 2020, doi: 10.1007/978-981-15-3215-3_25.

8. R. M. Rizk-Allah and M. A. Abo-Sinna, "A comparative study of two optimization approaches for solving bi-level multi-objective linear fractional programming problem," OPSEARCH, vol. 58, pp. 374–402, 2021.

9. J. Yang, X. Shi, and O. A. Prokopyev, "Exact solution approaches for a class of bilevel fractional programs," Optimization Letters, vol. 17, pp. 191–210, 2023.

10. T. O. Alakoya et al., "An efficient single projection scheme for solving bilevel variational inequality problems: Application to fractional programming problem," Results in Mathematics, vol. 80, 2025.

11. K. Bouibed, H. Slimani, and M. S. Radjef, "On efficiency conditions for nonsmooth multiobjective fractional bilevel programming with non-locally Lipschitz functions," OPSEARCH, 2024.

12. K. Dempe and S. Zemkoho, "Bilevel optimization: Advances and next challenges," Optimization, vol. 70, no. 10, pp. 2111–2132, 2021.