Implementation of Hybrid MILP/CP Decomposition from Jain and Grossmann (2001): A Case Study on Fantasy Premier League Team Management

Siu Hang IP

Assignment 4A - Project Proposal

Abstract

This project implements the hybrid MILP/CP decomposition from Jain and Grossmann (2001) as an open-source tool for Fantasy Premier League (FPL) team management. FPL involves over 10 million players optimizing squads under budget, formation, and transfer constraints. The tool applies MILP for lineup optimization (maximizing expected points) and CP for squad feasibility checks. This case study extends the paper's industrial scheduling to sports analytics, demonstrating potential performance over manual methods.

1 Introduction and Problem Context

1.1 Fantasy Premier League Overview

Fantasy Premier League (FPL) is a popular online game where participants assemble and manage virtual teams of 15 real Premier League players, earning points based on their real-world performances across 38 gameweeks. The core challenge for managers lies in navigating a series of complex decisions. These include the initial squad construction, which must be completed within a strict budget of £100 million, and the weekly selection of a starting lineup of 11 players that adheres to specific formations and positional quotas. Furthermore, managers must make strategic transfers throughout the season, constrained by a limited allowance of free transfers (1 free per gameweek, up to 5 saved, with extra transfers costing -4 points each). The problem is compounded by numerous constraints, such as fixed squad positions (2 goalkeepers, 5 defenders, 5 midfielders, 3 forwards), a limit of no more than three players from any single Premier League club, and the uncertainties of professional sports, including player injuries and suspensions.

Uncertainties like no-shows (e.g., injuries) will be handled via input probabilities to compute expected points, incorporating automatic substitution logic based on bench priorities (e.g., GK substitutes GK only, outfield by priority without breaking formation).

1.2 Discrete Optimization Problem Formulation

The FPL management problem is a discrete optimization task. It aligns with the class of problems described in the main reference paper by Jain and Grossmann in 2001. In that paper, the class of problems has the characteristic that only a subset of the binary variables have non-zero objective function coefficients if modeled as an MILP. The FPL game's structure requires combinatorial selection under constraints. As stated in the main reference paper, MILP provides a global perspective on constraints for optimization, while CP handles local constraint propagation. This makes FPL suitable for hybrid MILP/CP decomposition. The project applies this framework to maximize expected points of a fantasy team. It uses MILP for optimization bounds and CP for constraint propagation.

1.3 Motivation and Educational Value

The motivation for this project stems from a personal interest in FPL and observing fellow players make suboptimal decisions through manual, intuition-based methods. This project seeks to formally define the underlying mathematical structure of FPL and demonstrate how optimization techniques can yield superior results. From an educational perspective, it provides a practical and accessible application of the hybrid methods detailed in Jain and Grossmann's (2001) highly-cited paper, which recently received the 2024 INFORMS Journal on Computing Test of Time Award.

By developing an open-source tool, this project aims to create a valuable resource for students and enthusiasts to learn about hybrid optimization in a relatable domain, without requiring deep, domain-specific expertise. Documentation will include tutorials mapping code to theoretical concepts from the reference paper.

2 Related Work and State of the Art

The project builds on hybrid decomposition methods in discrete optimization. Roots trace to Benders Decomposition (Benders, 1962), which partitions a mixed-integer linear program into a continuous master problem and an integer subproblem. While effective, Benders decomposition is restricted to linear subproblems and struggles to represent combinatorial logic.

Breakthrough from Jain and Grossmann (2001) introduced hybrid MILP/CP decomposition that combines MILP relaxations with CP for feasibility checks. Work demonstrated that scheduling and planning problems—where subset of binary variables influences objective—can be solved by delegating global allocation to MILP and local feasibility to CP. Structure avoids big-M formulations and results in relaxations and search space.

Later, Hooker (2000–2007) formalized ideas as Logic-Based Benders Decomposition (LBBD), where Jain and Grossmann's framework is case of scheme with subproblem solved by CP, SAT, or constraint methods. Since then, LBBD is framework for integrating MILP and CP, with applications in workforce rostering, vehicle routing, and resource allocation. Solvers such as IBM ILOG CPLEX/CP Optimizer and Google OR-Tools adopted hybrid MILP/CP workflows in scheduling and logistics.

Recent advances use machine learning–augmented decomposition. For example, Tang et al. (2020) introduced reinforcement learning for cut selection in branch-and-cut. Learning-Guided Rolling Horizon Optimization (Li et al., 2025) uses neural networks in decomposition to predict which overlapping subproblems need not be re-solved, yielding 54% faster runtimes and solution quality over heuristic and methods. Works show ML-augmented LBBD, where framework remains stable but decisions guided by learned policies.

Hybrid MILP/CP decomposition of Jain and Grossmann (2001) is origin of LBBD, and influences state-of-the-art methods. Project applies framework to Fantasy Premier League team optimization, aligns with research and allows extensions involving learning-guided decomposition in field.

3 Technical Approach

3.1 Mathematical Model

Following the decomposition strategy proposed by Jain and Grossmann (2001), the FPL optimization problem is divided into a MILP master problem for optimizing lineup decisions that directly influence the objective and a CP subproblem for squad feasibility checks. Let P be the set of all available players, which is partitioned by position into G (Goalkeepers), D (Defenders), M (Midfielders), and F (Forwards). Let C be the set of all Premier League clubs.

Decision Variables:

$$\begin{aligned}
&y_i \in \{0,1\}, \forall i \in P \quad \text{(1 if player i is in the starting XI)}\\
&c_i \in \{0,1\}, \forall i \in P \quad \text{(1 if player i is captain)}\\
&v_i \in \{0,1\}, \forall i \in P \quad \text{(1 if player i is vice-captain)}\\
&b_{ij} \in \{0,1\}, \forall i \in P, j \in \{1,..,4\} \quad \text{(1 if player i is in bench position j)}\\
&x_i \in \{0,1\}, \forall i \in P \quad \text{(1 if player i is in the squad)}\\
&t_k \in \mathbb{Z}^+, \forall k \in K \quad \text{(number of transfers in gameweek k, for multi-gameweek extensions)}
\end{aligned}$$

Objective:

$$\text{maximize } \Phi(y,c,v,b,t) = \sum_i \text{expected\_points}_i \cdot (y_i + c_i + p_c v_i) - 4 \cdot \sum_k \max(0, t_k - \text{free\_transfers}_k)$$

, where expected_points_i is the projected points for player i, p_c is the probability of the chosen captain who does not show up and free_transfers_k is the number of free transfers available in gameweek k.

Constraints: The model is subject to the following constraints, partitioned between the MILP master problem and the CP subproblem.

MILP Master Problem Constraints (Lineup Optimization):

$$\sum_{i \in P} y_i = 11 \quad \text{(Starting XI size)}$$

$$\sum_{i \in G} y_i = 1 \quad \text{(One starting goalkeeper)}$$

$$3 \leq \sum_{i \in D} y_i \leq 5 \quad \text{(Defender formation limits)}$$

$$2 \leq \sum_{i \in M} y_i \leq 5 \quad \text{(Midfielder formation limits)}$$

$$1 \leq \sum_{i \in F} y_i \leq 3 \quad \text{(Forward formation limits)}$$

$$\sum_{i \in P} c_i = 1 \quad \text{(One captain)}$$

$$\sum_{i \in P} v_i = 1 \quad \text{(One vice-captain)}$$

$$c_i \leq y_i, \forall i \in P \quad \text{(Captain must be a starter)}$$

$$v_i \leq y_i, \forall i \in P \quad \text{(Vice-captain must be a starter)}$$

$$c_i + v_i \leq 1, \forall i \in P \quad \text{(Captain and vice-captain must be different)}$$

Surrogate Objective:

Maximize $\sum_i \text{expected\_points}_i \cdot (y_i + c_i + p_c v_i)$, relaxed for squad constraints.

CP Subproblem Constraints (Squad Feasibility and Logic):

$$\sum_{i \in P} x_i = 15 \quad \text{(Total squad size)}$$

$$\sum_{i \in G} x_i = 2 \quad \text{(Goalkeepers)}$$

$$\sum_{i \in D} x_i = 5 \quad \text{(Defenders)}$$

$$\sum_{i \in M} x_i = 5 \quad \text{(Midfielders)}$$

$$\sum_{i \in F} x_i = 3 \quad \text{(Forwards)}$$

$$\sum_{i \in P} \text{price}_i \cdot x_i \leq 100 \quad \text{(Budget limit)}$$

$$\sum_{i \in P_c} x_i \leq 3, \forall c \in C \quad \text{(Club limit)}$$

$$y_i \leq x_i, \forall i \in P \quad \text{(Starters must be in the squad)}$$

$$\sum_{j=1}^{4} b_{ij} = x_i - y_i, \forall i \in P \quad \text{(Assign benched players to one slot)}$$

$$\sum_{i \in P} b_{ij} = 1, \forall j \in \{1, .., 4\} \quad \text{(Each bench slot has one player)}$$

Transfer Constraints (in extensions): $t_k \leq 20$ (assumed no chips and wild cards); free_transfers_k up to 5 saved; integrated in objective for penalties.

3.2 Why Hybrid MILP/CP is Ideal for FPL

The hybrid MILP/CP approach is particularly well-suited for the FPL problem due to the natural partitioning of its constraints. MILP provides strong Linear Programming (LP) relaxations for the lineup aspects directly affecting the objective, such as expected points and roles, which yields tight bounds on the optimal solution. Simultaneously, CP excels at efficiently propagating the logical constraints related to squad selection and formations, avoiding the need for "big-M" formulations that often weaken the LP relaxation in pure MILP models. As demonstrated by Jain and Grossmann (2001) in their scheduling examples, this decomposition significantly reduces the search space and empirically outperforms pure MILP or CP methods. For FPL, this means that the

MILP component can effectively handle the points optimization across a large player pool (approximately 700 players), while the CP component can compactly and efficiently manage the complex combinatorial problem of squad feasibility. This partitioning aligns with the reference by placing objective-influencing variables in the master. To handle scalability with 700 players, heuristics such as player pre-filtering based on expected points will be applied in the MILP master.

3.3 Algorithmic Framework

The project will implement the decomposition algorithm proposed by Jain and Grossmann (2001). The iterative process begins by solving the MILP master problem to generate a lineup proposal (y, c, v). This proposal is then passed to the CP subproblem, which checks its feasibility with respect to the squad constraints. If the squad is not feasible, the CP subproblem generates cuts that are added to the MILP master problem to eliminate the infeasible solution and guide the subsequent search. Cuts will be "no-good" type (e.g., forbidding specific combinations of y that lead to infeasible x, such as $\sum_{i \in \text{infeasible\_set}} y_i \leq |\text{set}| - 1$) or problem-specific (e.g., based on budget violations), explicitly derived from the reference paper. This loop continues until a convergent solution is found.

The framework will be designed in phases, initially focusing on squad selection and then extending to handle weekly lineup decisions, transfers (modeling 1 free transfer per gameweek, up to 5 saved, as additional variables in multi-gameweek extensions with penalties in objective).

4 Evaluation Plan

To validate the tool, it will be tested on historical FPL datasets. The evaluation will leverage publicly available player performance predictions, as the primary focus of this project is on the optimization methodology rather than on developing novel forecasting techniques. The optimized squads and lineups will be used to calculate actual scores based on real outcomes from those seasons. These scores will then be compared to the overall population of FPL participants in those leagues to determine the ranking. Metrics include average percentile across test seasons, solution time, and expected points vs. realized points.

5 Deliverables

- Open-source Python codebase on GitHub, implementing hybrid MILP/CP using PuLP (MILP) and OR-Tools (CP).
- Documentation
- Historical FPL datasets and evaluation scripts.

References

[1] Benders, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, 4(1), 238–252.

[2] Jain, V., & Grossmann, I. E. (2001). Algorithms for hybrid MILP/CP models for a class of optimization problems. *INFORMS Journal on Computing*, 13(4), 258–276.

[3] Hooker, J. N. (2007). Planning and scheduling by logic-based Benders decomposition. *Operations Research*, 55(3), 588–602.

[4] Tang, Y., Cappart, Q., Rousseau, L. M., & Bengio, Y. (2020). Reinforcement learning for integer programming: Learning to cut. *Proceedings of the 37th International Conference on Machine Learning (ICML).*

[5] Li, S., Ouyang, W., Ma, Y., & Wu, C. (2025). Learning-guided rolling horizon optimization. *arXiv preprint arXiv:2502.15791.*