**A Genetic Algorithm to Find Optimal Player Combinations for Budget Constrained Soccer Teams**

Charles F. Whorton

University of Wisconsin - Green Bay

DS 785

Dr. Tracy Bibelnieks

April 30, 2023

# **Abstract**

This project explored the use of a genetic algorithm to identify optimal combinations of players for a soccer club to sign. Additionally, this project set out to provide a user interface in R Shiny that allowed users to filter the data and execute the algorithm. Members of Orlando City Soccer Club’s analytics staff supported this project by providing subject matter expertise on soccer analytics use cases, particularly in player assessment and acquisition. This project discussed and analyzed multiple optimization approaches to determine the most effective mathematical approach to solve this problem. Research into combinatorial optimization methods suggested using a genetic algorithm developed in R to identify the most valuable combination of players within a soccer club’s budget. The algorithm utilized American Soccer Analysis’ goals added dataset for the 2022 season along with a combination of American Soccer Analysis’ 2022 salary data and the Major League Soccer Players Association's 2022 salary data. The algorithm underwent rigorous testing, identifying the best parameters for different outcomes. It identified a set of parameters that resulted in a very high convergence and a set of parameters that resulted in a lower convergence rate. The methodology and analysis of results discussed the benefits of each and how they meet the user’s needs.

Table of Contents

[Abstract 2](#_Toc133766776)

[List of Figures 5](#_Toc133766777)

[Chapter One: Introduction 6](#_Toc133766778)

[**Background** 7](#_Toc133766780)

[**Player Assessment** 8](#_Toc133766781)

[**Explanation of Optimization Algorithms** 9](#_Toc133766782)

[**Objectives** 9](#_Toc133766783)

[**Introduction** 10](#_Toc133766784)

[**Goals Added** 10](#_Toc133766785)

[**Possessions** 11](#_Toc133766786)

[**Opponent Expected Goals** 12](#_Toc133766787)

[**Expected Goals per Player** 12](#_Toc133766788)

[**Data Analysis in Player Recruitment** 13](#_Toc133766789)

[**Player Assessment** 13](#_Toc133766790)

[**Financial Assessment** 14](#_Toc133766791)

[**Offensive Metrics** 14](#_Toc133766792)

[**Ball Possession** 14](#_Toc133766793)

[**Ball Location** 15](#_Toc133766794)

[**Defensive Metrics** 16](#_Toc133766795)

[**Conclusion** 16](#_Toc133766796)

[Chapter Three: Research Methodology 17](#_Toc133766797)

[**Data Preparation** 17](#_Toc133766798)

[**Optimization Algorithms** 17](#_Toc133766799)

[**The Knapsack Problem** 18](#_Toc133766800)

[**Brute Force** 18](#_Toc133766801)

[**Dynamic Programming** 18](#_Toc133766802)

[**Genetic Algorithms** 19](#_Toc133766803)

[**Search Space** 20](#_Toc133766804)

[**Iterations** 21](#_Toc133766805)

[**Runs** 22](#_Toc133766806)

[**Population Size** 22](#_Toc133766807)

[**Mutation Probability** 23](#_Toc133766808)

[**Execution Time** 25](#_Toc133766809)

[**Equal Comparison of Algorithm Parameters** 25](#_Toc133766810)

[**Conclusion** 25](#_Toc133766811)

[Chapter Four: Analysis of Results 27](#_Toc133766812)

[**Test Results** 27](#_Toc133766813)

[**Test Result Conclusions** 29](#_Toc133766814)

[**Interpretation of Results** 30](#_Toc133766815)

[**Conclusion** 30](#_Toc133766816)

[Chapter Five: Summary, Recommendations, and Conclusion 32](#_Toc133766817)

[**Summary** 32](#_Toc133766818)

[**Recommendations** 33](#_Toc133766819)

[**Future Considerations** 34](#_Toc133766820)

[**Conclusion** 34](#_Toc133766821)

[References 35](#_Toc133766822)

[Appendix 38](#_Toc133766823)

# **List of Figures**

[Figure 1: *Average fitness value over max number of iterations* 20](#_Toc133756474)

[Figure 2: *Average fitness value per number of runs* 21](#_Toc133756475)

[Figure 3: *Average fitness value per population size* 22](#_Toc133756476)

[Figure 4: *Visualizing the mutation process* 23](#_Toc133756477)

[Figure 5: *Average fitness value per mutation probability value* 23](#_Toc133756478)

[Figure 6: *Best performing population size in a large search space* 26](#_Toc133756479)

[Figure 7: *Best performing maximum number of iterations in a large search space* 27](#_Toc133756480)

# **Chapter One**

# **A Genetic Algorithm to Find Optimal Player Combinations For Budget Constrained Soccer Teams**

In sports, teams use every edge they can find to outmatch their opponent. Losses are costly, and wins are valuable, so every decision matters. To make the stakes higher, teams invest tens of millions of dollars into player scouting and acquisition. Consequently, the decisions supporting player acquisitions have considerable financial implications.

Soccer is a challenging sport to quantify due to the game’s fluid, dynamic nature. Clubs measure players’ quality by their on-field performance and how they increase their team's overall performance. Ultimately, a club wants to optimize the team's performance by signing high performing players. Clubs measure the cost of a player as the player’s salary, the player’s transfer fee, or a combination of both. Additionally, soccer is a particularly expensive sport, given the mostly unconstrained nature of its transfer market. The average transfer fee in Major League Soccer (MLS) over the 2022 season was approximately $1.75m (Carlisle, 2022). The league’s record-setting transfer fee was $16m, while record-setting player acquisitions in larger European leagues cost well over $100 million. A team allocates approximately 15% of its budget to scouting and player transfers. The high cost of transferring players makes the analysis behind these acquisitions immensely valuable.

This project aimed to build a tool that provides optimized transfer market suggestions. By doing this, soccer club analysts could identify the best combinations of players to sign under their team’s specific financial constraints. With well over 10,000 players in first division leagues worldwide, a soccer club has a massive pool of players to assess. Furthermore, individual player assessment requires significant effort, so the overall scouting process can take considerable time. By immediately identifying multiple optimized solutions, a club’s analytics staff can save time they would otherwise spend analyzing unfit players. Additionally, analytics staff members will have more time to perform in-depth analysis of player combinations to determine the best possible player combination to sign. This gives clubs more assurance that they identified the best players to sign. Ultimately, this tool equips analysts with reliable decision-making power to accelerate analytics functions and identify the highest-performing players the soccer club can afford.

There are many ways teams assess and quantify player performance. Some teams have proprietary methods, while other teams follow traditional approaches. A common metric is expected goals over replacement. This project did not have access to that metric, and building a model from the ground up that provided that metric did not fit into the scope of this project’s timeline. However, there were other reliable data sources that quantify player performance. The soccer analytics blog American Soccer Analysis (ASA) provides free soccer data and analysis resources. This project used their performance data and financial data for the 2022 Major League Soccer season. ASA built a metric called goals added that quantifies a player’s contribution to their team scoring a goal or preventing an opposition goal. This served as an excellent metric to optimize in the model. The Major League Soccer Players Association provided additional salary data that enriched the dataset.

## **Background**

Professional soccer players agree to play for a club by signing a contract. If a team wants to sign a player that is on contract with another team, the interested team must pay a transfer fee for the rights to sign the player. In a given season, it is likely that 65-75% of players in a league are still under contract. This means that most players' acquisition would require a transfer fee. While it is no secret that talented players will require a high transfer fee, there is immense value in identifying undervalued players. Many clubs cannot afford the transfer fees of world-class players, so clubs attempt to determine the greatest on-field value their transfer budget allows. This analytical approach earned the name “Moneyball” after Michael Lewis’ book of the same name.

MLS soccer clubs have salary cap constraints limiting how much they can pay players. Since this also limits which players a team can sign, clubs need to find the most talent to sign within their budget. By identifying high performing players within a price range, combinatorial optimization can save a club millions of dollars in salary cap space alone. Additionally, accurate player performance prediction can lead to improved overall team performance. This can result in increased revenue in a variety of the club’s operations. Overall, a mathematic approach to signing optimal players can have a multi-million dollar effect on the club.

## **Player Assessment**

Current scouting and analysis methods and technology allow soccer club analytics staff to analyze large volumes of data and assess large numbers of players in short periods of time. Data providers such as StatsBomb, Opta, and StatsPerform equip clubs with the information they need to perform advanced analysis. Clubs that take advantage of these services integrate them into their player recruitment strategies to improve their decision-making.

To start their data modernization journey, clubs use the vast data available to build a recruitment database. Within this database, clubs store information about players across the world. These data show various player metrics that clubs can use to assess players. With these metrics, clubs can build player assessment models that assign quantifiable scores to each player. These player performance scores vary between clubs as they are typically proprietary models developed in-house. While a player’s performance score can be a single value to indicate their overall quality, the score can also reflect a specific area of the game. With a player performance score, a soccer club’s analytics staff can compare players based on various factors and identify quality transfer targets. Clubs also use these vast data to create player profiles. The wide range of metrics available to clubs allows analytics staff to categorize players based on playing style, ability, and other traits. This enables clubs to narrow their search for transfer targets.

A club’s goal in the player assessment process is identifying a group of players as potential transfer targets. The combination of player profiles and performance scores allows soccer clubs to create a prioritized list of players the club wants to sign. This can be a manual and intensive process that may not mathematically guarantee good results. Technically, nothing can guarantee results for a decision that relies on human performance as it would be computationally infeasible to calculate a perfectly objective outcome. However, mathematical decision-making approaches increase the probability of identifying successful players.

## **Explanation of Optimization Algorithms**

A genetic algorithm is a computational model of biological evolution (Forrest, 1996). It is useful both as a search method for solving problems and for modeling evolutionary systems (Forrest, 1996). A genetic algorithm applied well to this project due to its ability to efficiently search through a large population. Different types of combinatorial optimization algorithms are more suited for different problems. However, the genetic algorithm provided immense value in identifying optimal combinations of players under financial and roster constraints.

## **Objectives**

To make effective decisions clubs must equip analysts with proper tools. The goal of this project was to create an application that provided a way for users to automate their player identification process. The project involved developing a combinatorial optimization algorithm and a supporting user interface.

First, the user interface allowed the user to specify input parameters for the application. The two input parameters were the number of players to search for and the budget threshold. The user interface passed these values to the optimization algorithm so the algorithm could build constraints. Next, the combinatorial optimization algorithm searched the provided dataset for the optimal combination of players based on the input received from the user interface. As a result, analysts used the application to provide data-driven recommendations of transfer targets for a club to sign.

**Chapter Two: Literature Review**

## **Introduction**

The primary objective of this study was to identify the optimal combination of players for a soccer club to sign based on player cost and quality. This project used a genetic algorithm to calculate the optimal combination of players but to do so, the algorithm needed a dataset to search. With too large of a dataset, the algorithm could not identify a good solution; in some cases, it could not identify any solution. The dependence on a reasonably sized dataset required thoughtful analysis before running the algorithm. Industry research provided a framework to help narrow down the large pool of players in the dataset so that the model had a suitable search space. The model used a metric developed by data scientists at American Soccer Analysis (ASA) called goals added to determine player quality. John Muller of ASA stated that goals added measures a player’s total on-ball contribution in attack and defense (Muller, 2020).

ASA provided data with goals added broken down by different game phases. Users could see goals added by dribbling, fouling, interrupting, passing, receiving, shooting, or an aggregation of all six game phases. While this project only used the aggregated goals added metric, there is a strong case to utilize game phase specific metrics depending on what type of player a user wants to see.

## **Goals Added**

Matthias Kullowatz, the mathematician that created the goals added metric, described the metric as one that values each action in a soccer match in units of goals (Kullowatz, 2020). ASA built this metric to describe a player’s performance over a season. Kullowatz built the metric utilizing an expected goals (xG) model and event data. The mathematicians and data scientists at American Soccer Analysis built their own expected goals model. They describe xG as the number of goals expected to be scored based on where and how a shot was taken (Kullowatz, 2015). Soccer data analytics provider StatsBomb also has a version of expected goals similar to ASA’s. StatsBomb defined their expected goals metric as a model’s estimate of the number of goals scored, on average, from a situation or a collection of situations (Vatvani, 2022). StatsBomb’s definition of expected goals more clearly indicated that expected goals reflects the increase or decrease in probability of any event leading to a goal. In 2017 Alex Rathke provided one of the most comprehensive studies on expected goals. In his study he noted the variety of the expected goals metric between different researchers and data providers. He noted that some versions of the metric focused solely on the goal scorer’s position, distance to goal, and shot placement without considering any factors of the defensive team (Rathke, 2017). Fortunately, American Soccer Analysis’ expected goals metric had significantly evolved since the time of Rathke’s writing and it considered the context necessary to support Kullowatz’s goals added metric.

Prior research described event data as information that contained positional information of the players and the ball as well as the results of soccer actions (Richly et al., 2017). It was also more simply described as information describing the players' actions with the ball (Haaren, 2019). By valuing each action in units of goals, the metric captured the value of many actions that are difficult to assess visually (Kullowatz, 2020).

While the event data provide important information, Kullowatz needed to build a complex model to contextualize the events. Soccer is a fluid and dynamic game with few stopping points. While the game constantly changes based on many factors, it is difficult to quantify the circumstances that change it. Kullowatz provided a methodology to explain how the goals added metric works.

### ***Possessions***

In the metric’s methodology, Kullowatz noted that the model to create goals added created a series of distinct plays referred to as possessions or chains (Kullowatz, 2020). Kullowatz defined a possession as a chain of actions that end in either a shot or a turnover (2020). A possession gives context to events which can increase or decrease the value of a decision. For example, a backward pass may not appear to increase the probability of scoring a goal. However, if the pass enabled a more dangerous pass, the model should classify the backward pass as a valuable action. Without the context of the possession the backward pass a model could not appropriately value the pass.

Additionally, the model calculated how each event in the possession contributed to the probability of scoring a goal. Since a possession is a chain of events, each event in the chain can increase or decrease the xG of other events in the chain. Matthias Kullowatz further explains a possession chain as a sequence of possessing, on-ball actions by a team without allowing the opponent a possessing action (2020). Sarmento et al. noted that because the number of goals scored was the metric most closely correlated with winning, the sport needed a study on the factors that increased the number of goals scored (Sarmento et al., 2017). The study mentioned that possession sequences leading up to goal scoring opportunities should be the focus of more analysis (Sarmento et al., 2017). More specifically, fast, direct possessions led to higher-quality scoring opportunities (Sarmento et al., 2017). The observations from Sarmento et al. validate the usefulness of a team’s possession in the goals added metric.

### ***Opponent Expected Goals***

Goals added would be an incomplete metric without the penalty of a negative outcome. Goals added tracks xG for all events regardless of the team responsible for the event. The difference in xG for each team before and after an action occurred indicated that action's total value (Kullowatz, 2020).

### ***Expected Goals per Player***

Since American Soccer Analysis tracked player involvement in expected goals, Kullowatz could attribute goals added to the players involved in each event. Individual player goals added was an excellent representation of player performance in a single possession chain or aggregated across a whole season. As mentioned in the introduction, ASA also recorded goals added by different game phases. The metric’s detail enabled thorough analysis as well.

## **Data Analysis in Player Recruitment**

The wealth of information from the industry’s numerous data providers significantly changed player recruitment (Burn-Murdoch, 2018). Ted Knutson, founder and CEO of StatsBomb, one of the world’s leading soccer analytics providers, said that their data enable complex models to assess player skill (Burn-Murdoch, 2018). Numerous published studies evaluate player quality. Models such as Fernández’ (2019) Expected Possession Value (EPV) framework and the T-pattern analysis developed by Amatria et al. (2019) did an excellent job of identifying player attributes closely associated with higher goal-scoring probabilities. Additionally, the goals added metric developed by Matthias Kullowatz of American Soccer Analysis provided an objective and quantifiable analysis of player contribution. While many soccer clubs use proprietary models and data sources, research indicated that performance metrics primarily center around goal-scoring probability by actions throughout a game.

## **Player Assessment**

The data utilized in this project was somewhat limited in its detail of player assessment. The literature review previously noted that ASA provided goals added broken out by six phases of the game. There is also value in using specific game-phase-specific metrics when assessing certain players. When Fernandez-Navarro assessed the playing styles of elite teams in England and Spain, he highlighted nineteen performance indicators closely linked to successful outcomes (Fernandez-Navarro et al., 2016). In Fernandez-Navarro’s study, the offensive indicators centered around ball possession, crosses, and shots taken (Fernandez-Navarro et al., 2016). The findings from Fernandez-Navarro et al. support using goals added by dribbling, passing, receiving, or shooting when assessing attacking players. The defensive indicators were associated with the location of defensive dispossessions (Fernandez-Navarro et al., 2016). The research conducted by Fernandez-Navarro et al. on defensive player success supported using goals added by interrupting for defensive player assessment.

### ***Financial Assessment***

This study assumed the cost of a player was their salary. While that is a fair assumption, additional factors exist in the total cost of acquiring a player. Chapter One discussed the financial process of acquiring a player on contract at another club. Garcia-del-Barrio & Pujol (2020) note numerous factors to consider in the actual cost of acquiring a player. To assess the financial value of players, they utilized the “*Methodology for the Evaluation and Rating of Intangible Talent”* (MERIT) approach, which appraised on-field talent with off-field properties of players that teams or fans may find attractive (Garcia-del-Barrio & Pujol, 2020). They indicate that on-field performance was a significant majority of a player’s value, but scarcity in the transfer market and a player’s celebrity also contributed to their cost. While these factors were important for a thorough player analysis, they were out of the project’s scope. Since they were out of scope the project used player salary, a very accessible value, as the players’ cost.

## **Offensive Metrics**

### ***Ball Possession***

The value of ball possession in soccer match analysis is widely disputed (Collet, 2012). Numerous studies linked ball possession to an increased likelihood of winning, but many fail to explain why. Due to various playing styles, the same measurements applied to different teams do not always produce the same outcome. In their 2007 study, Lago and Martin stated that standard notational analysis applied to possession did not consider the determinants of a team’s possession (Lago & Martín, 2007). As a result, this project included multiple possession-centric metrics during feature selection. The previously mentioned approach avoided bias toward specific systems of play.

While ball possession and passing require context, several passing metrics still increased win probability. Collet (2012) noted that pass accuracy added value to ball possession and indicated better player quality. While research did not dispute the value of pass accuracy, it did dispute the value of pass direction. Fernandez-Navarro et al. (2016) provides context to value of pass direction. In their study, Fernandez-Navarro et al. (2016) stated that more frequent backward passes indicated slower offensive progression. In contrast, they also mentioned that backward passes create new offensive opportunities that may result in more direct play (Fernandez-Navarro et al., 2016). Fernández’s (2019) Expected Possession Value (EPV) framework quantifies the theory stated by Fernandez-Navarro. He noted that backward passes created better conditions to further possession up the field (Fernández, 2019). Fernandez’ EPV framework supported this when it noted a positive EPV value for back passes (Fernández, 2019).

### ***Ball Location***

Studies also showed that pass location was another significant measure of team success (Fernandez-Navarro et al., 2016). Ball possession and movement closer to the opponent’s goal is more threatening than ball movement further away, regardless of the opposition’s defensive strategy. The pass recipient’s location, respective to the opposition’s formation, was also a significant metric. One of the most widely recognized tactical passing advantages comes from line-breaking passes. Line-breaking passes move the ball past a line of defense into a more threatening part of the field. In numerous meetings with analysts from the FC Barcelona youth academy, Fernández (2019) noted that they considered line-breaking passes one of the strongest indicators of win probability. Andreas Heuer’s (2020) research into packing rate, an alternate measure of line-breaking passes, affirmed the value of this metric.

Additionally, dribbling was a factor in ball location that increased goal-scoring probability (Amatria et al., 2019). While dribbling is a less efficient method of ball progression, the ability to do so in opposition territory proved to be a valuable skill. Dribbling is also an effective method to escape pressure and find open space. Since there is less open space closer to the goal, the ability to dribble well increases goal-scoring probability (Estêvão, 2017).

## **Defensive Metrics**

Similar to offensive strategies, there are a variety of defensive strategies in soccer. While players in different systems showcase different attributes, there are still common metrics that indicate player quality. The mechanisms used to recover ball possession are essential aspects of defense (Vogelbein et al., 2014). A 2012 European Football Championship study showed that regaining possession in the central or offensive third led to more goals than regaining possession in a team’s defensive third (Mitrotasios & Armatas, 2014). Regardless of defensive strategy, players that won the ball closer to the opponents’ goal increased their team’s win probability. Additionally, players increased their team’s win probability with a lower time between loss and regained control of possession (Vogelbein et al., 2014).

## **Conclusion**

Based on research into performance analysis, this project identified criteria to create an appropriate search space for an optimization model. Many of the player assessment studies shared common conclusions. Nearly all the studies noted the effect of passing and ball control on increased goal-scoring probability. Fernandez-Navarro et al. (2016) contextualized the value of different types of passes. Fernández (2019) pushed further and quantified the value of certain passes. Additionally, Heuer (2020) identified packing rate, or line-breaking passes, as the metric that has the strongest correlation to success. Lastly, Vogelbein et al. (2014) provided a compelling analysis of the location of defensive actions closely tied to success. American Soccer Analysis’ goals added metric represents the aforementioned elements of play, making it an excellent metric to optimize in this project’s optimization model.

# **Chapter Three: Research Methodology**

This section discusses the research methodology for identifying data science concepts and algorithms necessary to utilize optimization problems in the MLS transfer market. Since there are different use cases for optimization algorithms, this project required research to understand the appropriate algorithm. Additionally, this project required research to understand the proper tuning and modification of the selected algorithm to suit the problem best.

## **Data Preparation**

The data used in this project were easily consumable. The dataset provided aggregate metrics for each player from the 2022 season and the project did not need to perform significant calculations to derive meaning from the data. However, goals added was a summation of events, not a scaled rating. Without a reference point, goals added did not clearly indicate the quality of a player’s performance. The highest number of goals added was 5.43, which is unclear without context. This project took the log of the Goals\_Added column, which scaled the values between 0 and 1. After taking the log of Goals\_Added, the league’s lowest value of -3.42 appeared as 0.0043 and the highest value of 5.43 appeared as 0.9939. A log transformation to Goals\_Added contextualized performance and allowed the user to draw clearer comparisons.

## **Optimization Algorithms**

There are many optimization algorithms in mathematics, each fitting various real-world optimization scenarios. Linear programming and mixed integer linear programming problems are well-suited for problems attempting to optimize a linear objective function with a set of constraints. In addition, heuristic optimization algorithms perform well in complex search spaces by iteratively improving their search (Rego & Glover, 2002). Within the category of heuristic optimization algorithms there are a few different approaches. First, John Holland introduced the genetic algorithm in 1975. At the time the approach was called adaptation. Holland described the approach as a biological process whereby organisms evolve by rearranging genetic material to survive in environments confronting them (Holland, 1992). Later, in 1983, Kirkpatrick et al. found that the annealing process in statistical mechanics provides a framework for complex optimization problems. Their research supported the algorithmic approach of simulated annealing. These algorithms became two of the most popular heuristic combinatorial optimization algorithms.

## **The Knapsack Problem**

The knapsack problem is a classic combinatorial optimization problem. It aims to find a combination of objects that maximizes the total value while satisfying some resource constraints (Djannaty & Doostdar, 2007). The scenario of the knapsack problem fits the problem soccer clubs face in trying to identify the best combination of players available within a club’s budget. As it pertained to this project, the value the algorithm attempted to maximize was the players’ goals added. The resource constraints were the players' positions and the players' cost. While it was evident this project needed to solve the knapsack problem, there was still a question of how to solve it. There are numerous approaches to solving the knapsack problem that each benefit different scenarios.

### ***Brute Force***

A brute force approach is a recursive programming style that evaluates each possible combination (Thelin, 2020). The approach is effective because it evaluates all possible solutions but is only suited for small problems since the time complexity is exponential (Thelin, 2020). This project faced search spaces of nearly two billion combinations, making a brute force approach to the knapsack problem infeasible.

### ***Dynamic Programming***

In the R programming language, the adagio package contains functions to solve the knapsack problem using a dynamic programming approach. A dynamic programming approach orders the combinations by their profit-to-weight ratio and building solutions with the most favorable ratios (Claßen et al., 2015). Dynamic programming eliminates solutions that do not meet the weight constraint, allowing the algorithm to search a larger space than the brute force approach. While this is an excellent approach to many knapsack problem scenarios, it still struggles to effectively manage large search spaces like the ones encountered in this project.

## **Genetic Algorithms**

One of the more popular heuristic optimization algorithms is the genetic algorithm. Stephanie Forrest’s 1996 paper described a genetic algorithm as a computational model of biological evolution. Genetic algorithms simulate the evolutionary processes by taking an initial population of individuals and applying the genetic algorithm in each reproduction (Djannaty & Doostdar, 2007).

A genetic algorithm starts by creating a population, which is a random sample of individuals from the search space (Djannaty & Doostdar, 2007). The algorithm encodes each individual in the population into a string or “chromosome” of binary values representing a possible solution to a given problem (Djannaty & Doostdar, 2007; Forrest, 1993). In the case of the knapsack problem, the possible solution is a binary value representing whether the individual is selected or not. An objective function then evaluates the selected individuals in the population. After each evaluation of its objective function an algorithm performs its evolutionary operations such as crossover and mutation. Crossover creates parent populations that “mate” to produce child populations (Djannaty & Doostdar, 2007). The mutation process randomly selects individuals to flip their binary representation. The processes of mutation and crossover increase the probability of a genetic algorithm finding a better solution by retaining individuals that push the solution toward a global maximum. A genetic algorithm repeats this process until it meets its execution parameters. Over each iteration it retains any solution that meets the objective function.

Another benefit of a genetic algorithm over brute force or dynamic programming approaches is its problem-specific flexibility (Blanchard et al., 2022). Multiple parameters and custom functions enable problem specific behavior. In soccer analytics, users want to use data to support decisions. An algorithm that always returns the global maximum may miss out on other feasible player combinations. While soccer clubs trust their analytics staff, pursuing only one combination of players in a transfer window would be unwise. A genetic algorithm’s flexibility enables it to search in a way that provides multiple valuable solutions for each scenario instead of a single solution.

Simulated annealing, another heuristic optimization approach, preferred continuous search spaces making it less suited for this project. Research concluded that a genetic algorithm was the best approach to the knapsack problem, particularly as it pertained to this project, largely due to its ability to search a large, discrete search space.

### ***Search Space***

One of the most valuable benefits of a genetic algorithm is its ability to search a large space effectively. A genetic algorithm preserves high-fitness individuals throughout selection which biases the algorithm toward high-fitness regions (Forrest, 1996). The ability to gravitate toward high-fitness regions allowed the algorithm to identify a feasible solution in this project without searching all possible combinations.

In most scenarios, a club narrows its search to approximately fifty players. If a club wants to sign three players from a pool of fifty, that creates over 19,600 combinations of players. While that is a reasonable number of combinations, the time required to perform that number of search iterations may be more than a user is willing to tolerate. Fortunately, the genetic algorithm’s ability to intelligently evolve in a search space allows the algorithm to identify the best solution quite rapidly. In the most extreme case, a club may look for up to five players pulling from a pool of nearly two hundred players. Such a search creates over two billion unique combinations. Even with this large search space, the genetic algorithm could identify an optimal solution. Although, the algorithm would require a set of parameters tailored to the search space to find a feasible solution.

The following sections show that testing the algorithm with a wide range of parameters for each selected argument revealed the optimal parameter values. The initial test ran on the most extensive search space the user may encounter. These tests produced very high parameters, which resulted in a fifty second average algorithm execution time. Since the large search space of five players from a pool of two hundred is not a common scenario, and the tests resulted in fewer optimal solutions, the research methodology used the more likely scenario of three players from a pool of fifty players.

### ***Iterations***

A genetic algorithm executes a set of procedures a number of times, allowing the algorithm to improve and move closer to an optimal solution. As expected, more iterations indicated more chances to find an optimal solution. However, when searching the smaller, more commonly encountered search spaces, the algorithm did not need a high number of iterations. The algorithm ran ten times for each possible value between 100 and 1000 in increments of 25 to identify the best value for the “max\_iter” parameter.

**Figure 1**

Average fitness value over max number of iterations

Chart, line chart

Description automatically generated

Tests were inconsistent as the same parameters yielded different results across multiple test executions. However, the tests often showed that the algorithm reached the highest fitness value at approximately 300 iterations. Figure 1 above shows that the algorithm found the optimal value at 300 iterations, and values 300 through 650 also produced suitable results.

### ***Runs***

Within each iteration, a genetic algorithm will run its evolutionary procedures a set number of times. The algorithm typically reached the highest fitness value with at least 125 runs. A lower number of runs didn’t take advantage of the mutation process which prevented the algorithm from finding the optimal solution.

**Figure 2**

*Average fitness value per number of runs*

Chart, line chart

Description automatically generated

### ***Population Size***

The population is the set of selected individuals to which the algorithm applies various evolutionary principles. A custom function created the population parameter to meet the user’s requirements. Since a user specified how many players the algorithm should return in the user interface, the population needed to produce that number of selected individuals each time. But due to crossover and mutation operations, the population should return more than the specified number of individuals to allow the evolutionary operations to search for better solutions. A population size of 30 was the lowest population to reach the optimal solution during testing.

**Figure 3**

*Average fitness value per population size*

Chart, line chart

Description automatically generated

### ***Mutation Probability***

This project previously mentioned that a genetic algorithm’s problem-specific flexibility was a significant benefit compared to other approaches. More specifically, the algorithm’s evolutionary operations provided flexibility, allowing its selection to evolve toward an optimal solution. The mutation process provided flexibility by switching, or “mutating”, an individual to its opposite binary value. The mutation process mimics the survival of the fittest concept, where unfit individuals mutate into potentially more fit individuals (Forrest, 1996). The genetic algorithm’s probability of mutation dictated how frequently the population mutated individuals to identify more fit individuals. A higher probability of mutation increased the likelihood of an individual mutating into another individual on the next iteration.

**Figure 4**

*Visualizing the mutation process*

**A picture containing text

Description automatically generated**

Mutation is a core part of a genetic algorithm’s ability to improve and “evolve iteratively” (Forrest, 1996). Testing showed that a mutation probability of .25 was the lowest value that returned the optimal solution though values near .5 also performed well.

**Figure 5**

*Average fitness value per mutation probability value*

Chart, line chart

Description automatically generated

### ***Execution Time***

As a test of the execution time, the algorithm ran 50 times with the previously mentioned parameters. Over those 50 executions, the algorithm’s average execution time was approximately three seconds. An execution time of three seconds was a reasonable execution time to consider this application for use in a business context.

### ***Equal Comparison of Algorithm Parameters***

To compare the algorithm with higher parameter values against the algorithm with lower parameter values they were both run on the same search space. The search space was the larger search space, a 5 player combination from a pool of 200 players. Across ten executions, the algorithm with higher parameter values produced an average fitness value of 3.85. Since the fitness value was a sum of the goals added value of each player in the solution it was useful to compare the values by a per player average. Since there were five players in the solution, the average goals added value for each player was .77. The algorithm with lower parameter values also ran ten times in the same search space and produced an average fitness value of 3.73. With five players in the solution the average goals added per player was .75. The test comparing the two sets of parameters indicated that the algorithm with lower parameter values performed very similarly to the algorithm with higher parameter values.

### **Conclusion**

Parameter testing indicated that lower values for the maximum number of iterations, the number of runs, the population size, and the mutation probability led to frequently finding the optimal solution with a low algorithm execution time. The algorithm performed best with high parameter values, particularly in the larger search space. Testing the large search space indicated that a run value near 225, a maximum iteration value near 950, and a population size of 200 would allow the algorithm to converge frequently. A final comparison of the algorithm running with two different sets of parameters on the same search space gave the results that confirmed which algorithm was best. The algorithm with lower parameter values performed similarly to the algorithm with larger parameter values. Considering the execution time of the two algorithms, research indicated that a run value near 100, a maximum iteration value near 300, a mutation probability near .25, and a population size near 30 provided the most value to the users.

# **Chapter Four: Analysis of Results**

This project effectively utilized a genetic algorithm to identify optimal player combinations for a soccer club to sign under certain salary limitations. The algorithm’s results provided quantifiable recommendations to users on the highest-performing players whose combined salaries fit within their budget. The genetic algorithm underwent a testing phase that identified the parameters producing the most valuable results. Testing the model parameters was an essential step in the project as the parameters dictated the model’s recommendation quality and execution time. The research considered two test scenarios: a large search space where the algorithm tried to find a combination of 5 players from a pool of 200 players, and a moderate search space with a combination of 3 players from a pool of 50 players.

## **Test Results**

Chapter three presented the test results as they dictated the decisions around the model’s development. To find the best combination of parameters, the model ran ten times for each value passed from lists of approximately twenty possible parameter values. Visualizations presented in the research methodology showed the average of the model’s best solution value over each parameter value tested.

Initially, tests ran on the large search space. The results indicated that very high model parameter values produced a higher convergence rate than lower parameter values. The parameters required for frequent convergence in a large search space were a population size of 200, 225 runs, and value of 950 for the maximum number of iterations. Figure 6 shows a linear relationship between the population size and the convergence rate in a large search space. A linear relationship between the population size and the convergence rate meant the convergence rate increased as the population size increased.

**Figure 6**

*Best performing population size in a large search space*

Chart, line chart

Description automatically generated

Since the population cannot be larger than the search space the maximum parameter value for the population was 200. This result was unsurprising as the evolutionary operations such as crossover and mutation provide a greater benefit in a large search space when applied to a large population. The algorithm required at least 225 for the number of runs and 950 for the maximum number of iterations. While the algorithm had its highest convergence rate when the maximum number of iterations was 950, it could afford to have a lower and still return valuable results. Figure 7 shows that the algorithm could find similar solutions with lower parameter values. Users would prefer the lower parameter values in scenarios where the user does not need a global optimum.

**Figure 7**

Best performing maximum number of iterations in a large search space

Chart, line chart

Description automatically generated

While larger parameter values improved the algorithm’s convergence rate, particularly in larger search spaces, they also significantly increased its execution time. When the algorithm ran with parameters that gave the highest convergence rate, the algorithm took an average of fifty seconds to execute. Given this algorithm provides results to a user interface, fifty seconds is a very long time for a user to wait for results. When the algorithm ran with the parameters recommended for the moderately sized dataset, it only took an average of three seconds, which is a much more reasonable time for a user to wait. The execution time was a significant factor in determining which model parameters to use.

## **Test Result Conclusions**

The research concluded that it was best to reassess the search space and determine how a faster-running algorithm could yield positive results since it had such a long execution time. Users did not often encounter the scenario containing a large search space. However, users frequently encountered the scenario that produced the moderately sized search space. The player evaluation analysis in the Chapter Two literature review provided the criteria to filter a dataset to a moderately sized search space. When testing the algorithm on the moderate search space of three players from a pool of fifty players the algorithm identified a set of parameters that produced a frequent convergence rate and an execution time of three seconds. The parameter values identified to perform well on the moderate search space was a population size of 30 with 125 runs and a maximum iteration value of 300. Additionally, model parameter testing showed that the algorithm with higher model parameters barely outperformed the algorithm with lower parameters even on the large search space. The algorithm with lower parameters was better suited for the project for multiple reasons. First, users indicated they rarely encountered the large search space. Second, the algorithm's convergence rate with lower parameter values was comparable to that of an algorithm with higher parameters. Lastly, a long execution time was unfavorable.

## **Interpretation of Results**

Before testing, the initial theory was that higher model parameter values would help the algorithm achieve convergence more frequently. The results confirmed the theory, but the higher convergence rate came at a significant cost. This increased execution time encouraged research to explore alternative scenarios. When the algorithm ran with lower parameters on a moderately sized search space, the algorithm had a similar convergence rate in significantly less time. The results proved that users should perform the appropriate level of analysis to identify a moderate search space that an algorithm with lower parameters could manage.

## **Conclusion**

To summarize, the results did not support the initial hypothesis that higher parameters in the genetic algorithm lead to better solutions. Since execution and convergence rate were both a consideration when identifying a solution, the parameters that led to the most frequent convergence rate did not produce the best algorithm. Parameter testing favored the model with lower parameters because it had a high convergence rate in the most common search spaces, could still converge in a large search space, and had a fast execution time.

# **Chapter Five: Summary, Recommendations, and Conclusion**

This study explored using a genetic algorithm to suggest an optimal combination of soccer players given a constraint on a team’s budget. To start, client meetings and literature on soccer player analysis provided details for the contextual design of the algorithm. Then, thorough model parameter testing identified the most appropriate parameters to design a practical, value-added solution.

## **Summary**

The preliminary analysis determined that a genetic algorithm developed in R was the best approach to identify an optimal combination of soccer players in the MLS transfer market. This problem resembles the “knapsack problem” which a genetic algorithm solves very effectively. While numerous approaches to optimization problems exist, genetic algorithms are the best approach to non-linear problems. Standard “knapsack problem” packages did not have the features to constrain the number of players the algorithm returned. The flexibility of the “GA” package in R allowed a customized solution that fit the project’s needs.

Testing determined that there are limits to a genetic algorithm’s ability to optimize in a very large search space. The literature review discussed player evaluation criteria that could help users identify higher performing players. Users provided a manageable search space for the algorithm by first identifying higher performing players. The algorithm had a higher convergence rate with a low execution time on a manageable search space, making it significantly more useful to the user.

Client discussions revealed that a 100% convergence rate model was less useful than one with a lower convergence rate since it produced fewer solutions. Research indicated that teams preferred receiving multiple near-optimal solutions rather than one optimal solution. Since soccer is such a dynamic sport, it is challenging to predict a player’s future performance perfectly. An algorithm that produced multiple solutions allowed the client to assess player combinations in greater detail.

Model parameter testing helped identify the best parameters for different levels of performance. The model required high parameter values when prioritizing convergence in a large search space. In a moderately sized search space, the model still reached convergence with much lower parameter values, although with a lower convergence rate. The lower parameters values enabled a faster execution time, providing a more favorable user experience. Additionally, low parameter values in the model produced greater variance in results. Greater variance in results allowed users to see multiple near-optimal player combinations, improving analysis. Ultimately, the parameters identified as the best parameters maintained near convergence and allowed for rapid execution.

## **Recommendations**

The research strongly recommended proper analysis before the use of the algorithm. The algorithm was most effective in datasets of 200 rows or fewer and it excelled in datasets near 50 rows. The client indicated that it is common practice to search through datasets of players matching the desired profile rather than all available players. The client’s practice of filtering a dataset before assessing players supported the use case for the algorithm.

Additionally, prior research on the topic indicated that analysts prefer to have multiple player combinations rather than a single suggestion. A highly optimized algorithm had little variance in its solutions across different executions since it frequently converged. Therefore, testing recommended a model that does not always converge. The model's ability to generate varied solutions across executions proved more useful to analysts than a single, perfectly optimized recommendation.

Model parameter testing showed that modest parameter values achieved the previously mentioned varied set of solutions while maintaining a high degree of optimization and a low execution time. The recommended model parameters were a population near 30, 125 runs, and a maximum of 300 iterations per execution. These parameters best met the user’s needs of a high degree of optimization, somewhat varied solutions, and a low execution time.

## **Future Considerations**

At the conclusion of this project, the user could not change the algorithm's performance from the user interface. While this project followed the best practices indicated by research, there may be cases where a user would prefer a different approach. Enabling the user to alter the algorithm’s parameters would allow the user to choose the algorithm’s convergence rate. In cases where the user wants to identify the global optimum, the algorithm would perform best with a different set of parameters than what this project provided by default.

Additionally, the project could not prioritize certain player selections over others. When a soccer club searched for transfer targets the analytics staff prioritized their positional needs. The algorithm did not consider that one positional need may be a priority over another. To consider this, the algorithm should be able to favor a particular position as indicated by the user. The algorithm could allow player prioritization by applying a weight penalty to any individuals of a position other than the prioritized position.

## **Conclusion**

In conclusion, this project solved soccer clubs’ problem of identifying the optimal combination of players to sign under financial constraints. The results of this study guided the selection of a performant optimization model in the form of a genetic algorithm. Detailed research and testing indicated the appropriate parameter for the algorithm to provide the most valuable results. This project helped soccer clubs increase efficiency and reliability in the player identification and acquisition process by providing an efficient, data-driven approach to player identification.

# **References**

Amatria, M., Maneiro, R., & Anguera, M. T. (2019). Analysis of successful offensive play patterns by the Spanish soccer team. *Journal of Human Kinetics*, *69*(1), 191–200. https://doi.org/10.2478/hukin-2019-0011

Blanchard, A. E., Shekar, M. C., Gao, S., Gounley, J., Lyngaas, I., Glaser, J., & Bhowmik, D. (2022). Automating Genetic Algorithm Mutations for Molecules Using a Masked Language Model. IEEE Transactions on Evolutionary Computation, 26(4), 793–799. https://doi.org/10.1109/TEVC.2022.3144045

Burn-Murdoch, J. (2018, October 31). How data analysis helps football clubs make better signings. *Financial Times*. Retrieved March 12, 2023, from https://www.ft.com/content/84aa8b5e-c1a9-11e8-84cd-9e601db069b8.

Carlisle, J. (2022, August 12). MLS is now a transfer-Market Destination League, and investment in youth has played a big role in that. ESPN. Retrieved from https://www.espn.com/soccer/major-league-soccer/story/4718910/mls-is-now-a-transfer-market-destination-leagueand-investment-in-youth-has-played-a-big-role-in-that

Claßen, G., Koster, A. M. C. A., & Schmeink, A. (2015). The multi-band robust Knapsack Problem—a dynamic programming approach. Discrete Optimization, 18, 123–149. https://doi.org/10.1016/j.disopt.2015.09.007

Collet, C. (2012). The Possession Game? A comparative analysis of ball retention and team success in European and International Football, 2007–2010. *Journal of Sports Sciences*, *31*(2), 123–136. https://doi.org/10.1080/02640414.2012.727455

Djannaty, F., & Doostdar, S. (2007). A Hybrid Genetic Algorithm for the Multidimensional Knapsack Problem. International Journal Of Contemporary Mathematical Sciences, 3(2008), 443–456.

Estêvão, T. (2017, December 28). *Why is dribbling useful in football?* Medium. Retrieved March 11, 2023, from https://tiagoestv.medium.com/why-is-dribbling-useful-in-football-3087be9fdd8b

Fernández, J., Bornn, L., & Cervone, D. (2019, March). Decomposing the immeasurable sport: A deep learning expected possession value framework for soccer. In 13th MIT Sloan Sports Analytics Conference.

Fernandez-Navarro, J., Fradua, L., Zubillaga, A., Ford, P. R., & McRobert, A. P. (2016). Attacking and defensive styles of play in soccer: Analysis of Spanish and English elite teams. *Journal of Sports Sciences*, *34*(24), 2195–2204. https://doi.org/10.1080/02640414.2016.1169309

Forrest, S. (1993). Genetic algorithms: principles of natural selection applied to computation. Science, 261(5123), 872+. https://link-gale-com.ezproxy.uwgb.edu/apps/doc/A14217688/AONE?u=greenbay&sid=bookmark-AONE&xid=a4d459fa

Forrest, S. (1996). Genetic algorithms. *ACM computing surveys (CSUR)*, *28*(1), 77-80.

Garcia-del-Barrio, P., & Pujol, F. (2020). Recruiting talent in a global sports market: Appraisals of soccer players' transfer fees. Managerial Finance, 47(6), 789–811. https://doi.org/10.1108/mf-04-2020-0213

Haaren, J. V., Robberechts, P., Decroos, T., Bransen, L., & Davis, J. (2019). Analyzing Performance and Playing Style Using Ball Event Data. Football Analytics: Now and Beyond. A Deep Dive into the Current State of Advanced Data Analytics, 36–47.

Heuer, A. (2020, March 8). *Identification of relevant performance indicators in round-robin tournaments*. arXiv.org. Retrieved March 12, 2023, from https://arxiv.org/abs/2003.03774v1

Holland, J. H. (1992). Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and Artificial Intelligence. MIT Press.

Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. Science, 220(4598), 671–680. http://www.jstor.org/stable/1690046

Kullowatz, M. (2015). What are expected goals (XG)? American Soccer Analysis. Retrieved March 7, 2023, from https://www.americansocceranalysis.com/explanation

Kullowatz, M. (2020, July 1). Goals added: Deep Dive Methodology. American Soccer Analysis. Retrieved March 7, 2023, from https://www.americansocceranalysis.com/home/2020/5/4/goals-added-deep-dive-methodology

Lago, C., & Martín, R. (2007). Determinants of possession of the ball in soccer. *Journal of Sports Sciences*, *25*(9), 969–974. https://doi.org/10.1080/02640410600944626

Mitrotasios, M., & Armatas, V. (2014). Analysis of Goal Scoring Patterns in the 2012 European Football Championship. The Sport Journal, 47, np–np.

Muller, J. (2020). What are goals added (G+)? American Soccer Analysis. Retrieved March 7, 2023, from https://www.americansocceranalysis.com/what-are-goals-added

Rathke, A. (2017). An examination of expected goals and shot efficiency in soccer. Journal of Human Sport and Exercise, 12(2), 514-529.

Rego, C., Glover, F. (2002). The Traveling Salesman Problem and Its Variations, edited by G. Gutin, and A. P. Punnen, Springer, 2002. ProQuest Ebook Central, http://ebookcentral.proquest.com/lib/uwgb/detail.action?docID=197850.

Richly, K., Moritz, F., & Schwarz, C. (2017). *Utilizing Artificial Neural Networks to Detect Compound Events in Spatio-Temporal Soccer Data* [Paper presentation]. 3rd SIGKDD Workshop on Mining and Learning from Time Series. Halifax, Nova Scotia, Canada. 13-17 August 2017.

Sarmento, H., Figueiredo, A., Lago-Peñas, C., Milanovic, Z., Barbosa, A., Tadeu, P., & Bradley, P. S. (2018). Influence of tactical and situational variables on offensive sequences during elite football matches. *Journal of Strength and Conditioning Research*, *32*(8), 2331–2339. https://doi.org/10.1519/jsc.0000000000002147

Thelin, R. (2020). Demystifying the 0-1 Knapsack problem: Top solutions explained. Educative. Retrieved March 27, 2023, from https://www.educative.io/blog/0-1-knapsack-problem-dynamic-solution

Vatvani, D. (2022, September 28). *Upgrading expected goals*. StatsBomb. Retrieved March 10, 2023, from https://statsbomb.com/articles/soccer/upgrading-expected-goals/

Vogelbein, M., Nopp, S., & Hökelmann, A. (2014). Defensive transition in soccer – are prompt possession regains a measure of success? A quantitative analysis of German Fußball-Bundesliga 2010/2011. *Journal of Sports Sciences*, *32*(11), 1076–1083. https://doi.org/10.1080/02640414.2013.879671

# **Appendix**

To view the codebase please visit the following URL: **https://github.com/cfw412/capstone**

To use the application please visit the following URL:

**https://cfw-capstone-playeroptimizationapp.shinyapps.io/capstone/**