2024, Retos, 57, 462-472

Enhancing the Probability of Victor for Top Players via Knockout Tournament Fixture
Mejorando la Probabilidad de Victorias para los Mejores Jugadores a través del Emparejamiento de Torneos por Eliminación Directa

*Hegen Dadang Prayoga, *Tomoliyus,*Ria Lumintuarso, *Yudik prasetyo,*Endang Rini Sukamti, **Ari Tri Fitrianto, **Andi Kasanrawali, ***Ramadhan Arifin, **Ahmad Maulana, **Muhammad Habibie
*Yogyakarta State University (Indonesia), **Kalimantan Islamic University MAB (Indonesia), ***Lambung Mangkurat University (Indonesia)

Abstract. A tournament fixture is an integral part of the tournament rules. It determines the random pairing of contestants, with several matches played per round. The selection of fixture type that optimizes the top player's winning probability significantly affects the financial aspects for organizers, individuals, and participants while also addressing the interests of millions of fans. In addressing this challenge, this study designed a balanced tournament fixture and employed a labeling system to represent each fixture, utilizing a recursive function. By assigning a strength rating to each player, their rankings were established, leading to varied probabilities of winning. It was decided to represent these abilities with randomly selected integers ranging from 1 to 21, with 1 denoting minimum strength and 21 denoting maximum strength. We explore hierarchical knockout tournament fixtures in competitions to develop optimal tournaments that enhance their attractiveness. In this study, we also performed calculations to determine the probability of each player winning in each round, thereby deducing which tournament fixture minimizes or maximizes the likelihood of the strongest player winning. In cases where the number of players is a power of 2, the first half comprises $p/2$ matches, where p is the total number of players. However, if the number of players is not a power of 2, k matches are played in the first round, with $p = 2^r + k$, where $0 \leq k < 2^r$, followed by implementing a balanced tournament fixture. The findings underscore the effectiveness of employing a balanced tournament fixture to maximize the probability of winning in a single-elimination tournament.
Keywords: Single Elimination, Binary Tree, Dummy Players, Election Procedures, Hierarchically.

Resumen. Un fixture de torneo es una parte integral de las reglas del torneo. Determina el emparejamiento aleatorio de los concursantes, con varios enfrentamientos por ronda. La selección del tipo de fixture que optimiza la probabilidad de victoria del mejor jugador afecta significativamente a los aspectos financieros para los organizadores, los particulares y los participantes, al tiempo que atiende a los intereses de millones de aficionados. Para hacer frente a este reto, este estudio diseñó un dispositivo de torneo equilibrado y empleó un sistema de etiquetado para representar cada dispositivo, utilizando una función recursiva. Al asignar un índice de fuerza a cada jugador, se establecía su clasificación, lo que daba lugar a distintas probabilidades de ganar. Se decidió representar estas capacidades con números enteros elegidos al azar que van del 1 al 21, donde 1 denota la fuerza mínima y 21 la fuerza máxima. Exploramos los torneos por eliminatorias jerárquicas en competiciones para desarrollar torneos óptimos que aumenten su atractivo. En este estudio, también realizamos cálculos para determinar la probabilidad de que cada jugador gane en cada ronda, deduciendo así qué arreglo del torneo minimiza o maximiza la probabilidad de que gane el jugador más fuerte. En los casos en que el número de jugadores es una potencia de 2, la primera parte comprende $p/2$ partidos, siendo p el número total de jugadores. Sin embargo, si el número de jugadores no es una potencia de 2, se juegan k partidos en la primera ronda, con $p = 2^r + k$, donde $0 \leq k < 2^r$, seguido de la implementación de un fixture de torneo equilibrado. Los resultados subrayan la eficacia de emplear un sistema de torneo equilibrado para maximizar la probabilidad de ganar en un torneo de eliminación simple.
Palabras clave: Eliminación simple, Árbol binario, Jugadores ficticios, Procedimientos de elección, Jerárquicamente

Fecha recepción: 07-04-24. Fecha de aceptación: 24-05-24
Hegen Dadang Prayoga
hegendadang.2023@student.uny.ac.id

Introduction

A tournament is a rule that specifically regulates how teams or players will be contested to determine who will be the winner (Edwards, 1996). In sports activities, the terms of matches, competitions, and contests are usually used interchangeably to represent experimental units (Ekin et al., 2023). Similarly, the terms teams, players, composers, and contestants are used interchangeably and represent the treatments (Sobkowicz et al., 2020) (Bubna et al., 2023). It is assumed that games always result in either a win or a loss; ties are not allowed (Csató, 2023). In knockout tournaments, all players are contested except players who have been knocked out or eliminated (Přidal & Priklerová, 2018). Games must be played in a certain order because the results of previous games determine the contestants in subsequent games. A player who has been eliminated may not compete again (Musa et al., 2022). Tournament fixture is a part of the tournament rules (Brito De Souza et al., 2021) that determine how contestants will be paired by accidentally determining which contestants will be paired (Rojas-Valverde et al., 2020) and the number of matches played based on each round (Adler et al., 2017). According to (Bădică et al., 2021) the tournament is carried out in a series of rounds. If several X players are the power of 2, for example, $X = 4 = 2^2$, so the tournament fixture becomes as shown in Figure 1, with all players entering the tournament in the first round.

Tournament single-elimination (SE) is a knockout tournament fixture (P. Parande et al., 2023) where teams are eliminated after losing one game, and the winning team continues the match until, in the end, only the single team remains the winner (Hulett, 2019). In an SE tournament with an *r* round, the maximum number of players is $p_{max} = 2^r$, and it is obtained when in the first round, we have a maximum number of 2^{r-1} games; therefore, for a tournament with *r* rounds becomes as shown in Figure 2.

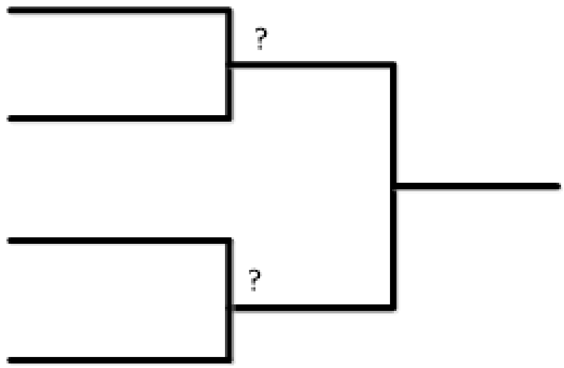

Figure 1. Tournament fixture with two rounds

$$2^{r-1} < p \leq 2^r$$

Figure 2. Tournament with *r* round

Observe that from Figure 2 implies by given Figure 3.

$$r = \left\lceil \log_2 p \right\rceil$$

Figure 3. Minimal number of rounds

To determine the maximum number of players entering the tournament in the first round, it can be obtained in a reverse way. that reverse way is shown in the Figure 4.

$$p = 2^r$$

Figure 4. Maximal number of players

Knowledge of binary trees has allowed some researchers to manipulate tournament fixtures and obtain different probabilities in the context of sporting tournaments (Arlegi & Dimitrov, 2020). A knockout tournament fixture is hierarchically structured as a binary tree (King & Rosenberg, 2023) (Prayoga et al., 2024) such that each leaf represents one player or team that is enrolled in the tournament, while each internal node represents a game of the tournament (Ikhwani et al., 2023). As shown in Figure 5, a 90-degree rotation to the right of a binary tree will result in a tournament fixture representation

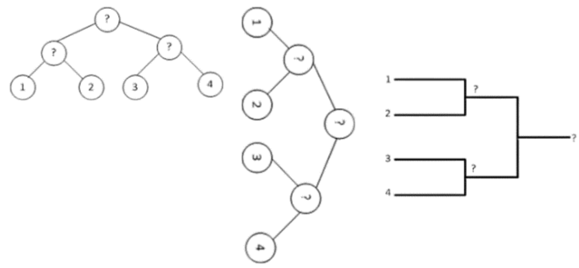

Figure 5. The binary tree is modeled as tournament fixture

In knockout tournaments, different types of tournament fixtures need to be distinguished (Guyon, 2022). The fixture indicates how players will be paired, but it without specifies which players will compete against each other. According to (Edwards, 1996) there are two tournament fixture types namely balanced and unbalanced as depicted graphically in Figure 6.

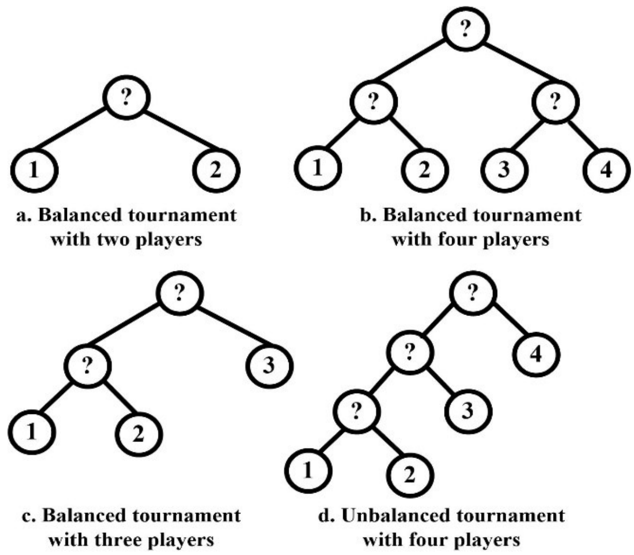

Figure 6. Tournaments of two and three players (first column) and four players (second column)

Observe that the tournaments on the first row (labeled "a" and "b") involve a number of elements that is a power of two ($2 = 2^1$ and $4 = 2^2$, respectively) and are fully balanced. However, the tournaments on the second row are not fully balanced, although the lower rightmost tournament involves $4 = 2^2$ players. However, intuitively, the tournament with three players (labeled "c") should be accepted, as player 3 will enter the tournament only 1 round after players 1 and 2, i.e., it has a sense of "balancing". However, the lower rightmost tournament with four players (labeled "d") is not acceptable, as player 4 received an exemption from playing in the first two rounds, and this is considered unfair. When the total count of *p* players is a power of 2, such as $p = 8 = 2^3$, the tournament fixture is a perfect binary tree, with all participants engaged in the initial round. To do so, perfect binary trees with 2^r leaves are used, and players planted into trees' leaves. To handle the number of players who are not the power of 2 (Prayoga et al., 2024) calculates the number of dummy players D with a equation 2^r - *p* where *r* is obtained from Figure 3 so that the number of dummy players can be calculated with a equation which is shown in the Figure 7.

$$D = 2^r - p$$

Figure 7. Number of dummy players

In this study, we used the same technique to deal with the number of players who were not power of two. According to (Adler et al., 2017), the analogy that an unbalanced fixture is a fixture of one match per round minimizes the strongest player's chances of winning. In addition, the authors provide an upper limit and a lower limit for each participant; the upper limit provides the probability that the weakest player will win the tournament, and the lower limit provides the probability that the strongest player will win it (Manurangsi & Suksompong, 2023). The driver explained more about seeding and its impact on the probability of calming a game in a knockout tournament. Driver and Hankin (Driver & Hankin, 2023) explain more detail about seeding and its impact on probability of winning a match in knockout tournament. They used Bradley-Terry model (Gao & Mahmoud, 2023) that returns the probability of participant *i* beating participant *j*, noted as V_{ij} in this equation.

$$V_{ij} \frac{strength(i)}{strength(i) + strength(j)}$$

Figure 8. Bradley-Terry model

A preference matrix P, which contains the probability that one player beats another, can be obtained from these rules. If it is applied to the eight-player tournament, it is denoted as Figure 9:

$$\begin{pmatrix} V_{11} & \cdots & V_{18} \\ \vdots & \ddots & \vdots \\ V_{81} & \cdots & V_{88} \end{pmatrix}$$

Figure 9. Matrix probability

David (David, 1959) shows that this *P*-preference matrix follows strong stochastic transitivity. It results in $\forall_{ijk}$ which is shown in the Figure 10.

$$if\ V_{ij} \geq 0.5\ and\ V_{jk} \geq 0.5\ then\ V_{ij} \geq 0.5$$

Figure 10. Stochastic transitivity

It means that the probability of participant *i* defeats participant *j* is greater than or equal to .5, and the probability of participant *j* defeats participant k is also greater than or equal to .5, then the probability of participant *i* defeats participant *k*, based on the principle of transitivity, will be similarly greater than or equal to .5. This study considers only the strong stochastic transitivity matrix, and assumes that the players are sorted in order p_1 p_2, .., p_n from the strongest to the weakest. Therefore, since the *P* matrix follows a strong stochastic transitivity, players can be sorted by strength, which will be useful later when calculating each player's winning probability. The player p_i will be ranked above the player p_j, indicating that the player *i* is stronger than the player *j* if p_i has a better chance of winning from p_j. Edward (Edwards, 1996) used the probability that *i* win the round *r* as

$$W_{ir} = W_{k,r-1}\left[\sum_{k=v}^{u} P_{ik} W_{k,r-1}\right] where\, W_{i,0} = 1\, and\ r > 0$$

$$v = s(i,r) = 1 + 2^{r-1} + 2^{r-1}\left\lfloor \frac{i-1}{2^r} \right\rfloor - 2^{r-1}\left\lfloor \frac{i-1}{2^{r-1}} \right\rfloor\ and\ u = v + 2^{r-1} - 1$$

Figure 11. Edward's probability

In the Figure 11, *u* and *v* represent a possible opponent's upper and lower limits for *i*. These limits indicate all possible opponents of a player *i* in a round of *r*. For example, in an eight-player tournament with three rounds (R=3), as shown in Figure 12.

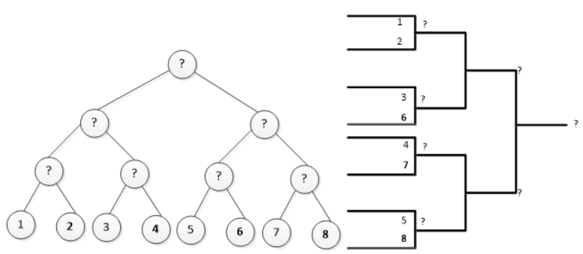

Figure 12. Tournament fixture with 8 players and three rounds

During the first round, players in the first leaf node can only meet players in the second leaf node. They can meet players on the 3rd or fourth leaf node in the second round. Finally, in the third round, they can meet players on the sixth, seventh, or eighth leaf nodes. Thus, for players in the first leaf node are as follows

$S(1,1) = 2$ as $v = 2$ and $u = 2$,
$S(1,2) = 3,4$ as $v = 3$ and $u = 4$,
$(1,3) = 5,6,7,8$ as $v = 5$ and $u = 8$.

With the same example, W_{11} is the probability that player 1 wins in the first round or the probability that player 1 beats player 2. W_{12} is the probability that player 1 wins in the second round or the probability that player 1 wins in round 1 and beats player 3 or player 4. Maurer (Maurer, 1975) proves that for random tournament fixtures, in certain cases where $v1 > v2 = v3 = ... = vp$, which means that one player has a higher strength while the other has the same strength, so the balanced tournament fixture that maximizes the probability of a stronger player winning the tournament. To prove it, a random tournament was used, and fixture for p players was a combination of a random fixture with fewer players. The number of different tournament fixtures for p players was also calculated. Figure 13 refers to the formula for calculating the number of different tournament fixtures for p players.

$$A_p = \sum_{i=1}^{p-1} A_i * (A_{p-1} + \delta_{i,p-1}) \; with \; A_1 = 1$$

Figure 13. Formula for number of different tournament

Where $\delta_{i,j}$ denotes the Kronecker's delta, a two-variables function that returns 1 if $i=j$ and 0 otherwise. Maurer also added a equation which is in all possible fixtures, is balanced denoted. (Bp). The Maurer equation is shown in the Figure 14

$$B_p = \sum_{i=1}^{u} B_i * (B_{p-i} + \delta_{i,p-1}) \; with \; A_1 = 1$$

Where, for $p = 2^r + k$ with $0 \le k < 2^r$:

$$1 = \max(2^{r-1}, k)$$

$$u = \min(2^{r-1} + k, 2^r)$$

Figure 14. Maurer equation

The results of these calculations are shown in Table 1.

Table 1.
The number of tournament balanced fixtures with 12 players

p	1	2	3	4	5	6	7	8	9	10	11	12
B_p	1	1	1	1	1	2	1	1	1	3	3	5

Edwards (Edwards, 1996) suggests that representation of each tournament fixture is labelled by a label. The approach consists of assigning a label with a unique sequence of digits representing each leaf node binary tree with $n+1$ leaf node external. Tournament players will be placed on each of the perfect binary tree's $n+1$ external leaf nodes. This label consists solely of the numbers 2s, 1s, and 0s. "2" indicates that two actual players, meaning not placeholders, are assigned to the tournament fixtures, "1" signifies that one real player competes against a dummy player, "0" indicates that two dummy players are allocated in tournament fixtures. For instance, in Figure 15, the sequence "2111" is illustrated for a five-player game, with three-dummy players blue dashed lines.

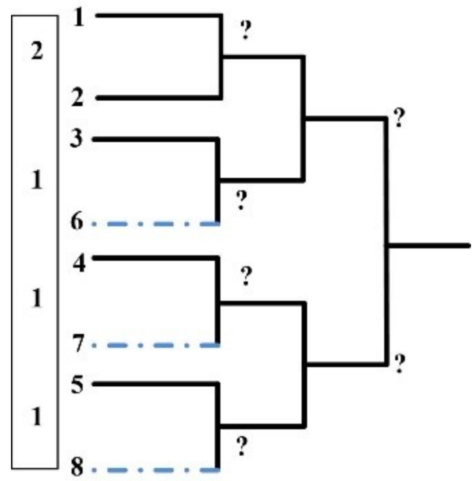

Figure 15. Tournament fixture 2111 with three-dummy players

Each digit of the label corresponds to a match; the length of the label depends on the number of rounds in the fixture tournament. Indeed, depending on the fixtures, $\lceil log_2 p \rceil$, and $p-1$ rounds can be different if *Rmin* represents the minimum rounds and *Rmax* denotes the maximum rounds in a tournament involving p players. Then, *Rmin* = $\lceil \log_2 p \rceil$, *Rmax* = p-1. Therefore, there will be several labels with lengths 2^r, where $Rmin \le r \le Rmax$ and $r \in N$. Lucas (Lucas et al., 1993) present a recursive algorithm that aims to produce different binary trees with only one side rotation. In this study, a recursive function is used to overcome the difficulty of generating binary trees computationally. For example, with six players, the maximum number of rounds is five, and the label's maximum length is $2^5 - 1 = 16$. It produces 316 = 43 054 721 possible labels to be generated.

The main objective of this research is to determine the type of fixture that optimizes the probability of the strongest player winning. In this study, the researchers explore knockout tournaments where two players face each other at most once. These are specific tournaments with potential applications in sports such as football (Paixao et al., 2021) and tennis (Cordellat Marzal & Valenciano, 2022), as well as muaythai (Bhumipol et al., 2023), eSports game (Sziklai et al., 2022) (Dong et al., 2023), and election procedures (Manurangsi & Suksompong, 2022). In the end, the probability of each player winning in every round was calculated to identify the highest likelihood. This process aimed to determine the optimal arrangement for the strongest player.

Methods

To facilitate understanding of the methods used, the researchers gather the main principles and explain each based on the literature review discussed earlier. The principles are as follows.

Table 2.
Different labels for tournament fixture up to 7 players

Players	Label
5	2111
	2210
	21101000
7	2221
	22111000
	21211000
	21101110
	22101010
	2111100010000000
	2110101010000000
	2110100010001000
	2210100010000000
	21101000100000001000000000000

Binary trees as tournament fixture

The first step is randomly generating labels containing the numbers 2, 1, and 0. After all possible order of labels, according to Edwards' naming rules, they need to be sorted to remove those labels that are not allowed. The result is all possible tournament fixtures with *p* players based on the Wedderburn-Etherington sequence (Lee et al., 2023). The algorithm randomly generates a sequence of numbers between 0 and 2. As a result, the likelihood of the number of rounds and their sequence length is exponentially increased. To solve this problem, we implemented a recursion function. An analysis of the tournament fixture shows that the left half of the label, which represents the top table of a fixture, is a sub-fixture of a label with fewer players. For example, Table 2 shows different labels for tournaments involving up to 7 players.

For a sequence of seven players and five rounds (i.e., length 16), the left part of the label, highlighted in gray, comes from a label with a length of eight players or fewer. Therefore, in the algorithm, from several loops greater than two, the recursive function "*use_previous_left*" is used to generate the left half of the label. The rules used in the function *use_previous_left (p,r)*, with *p* the number of players and the number of rounds with $3 < r$ are as follows: for any *p*, keep, in a list, the last official label of the labels of *r*-1. In the same way, to generate the right part of the label, the "*use_previous_right*" function is used. The rules used in the function *use_previous_right (p,r)*, where p is the number of players and is the number of rounds with $3 < r$ are as follows:

1. For each *p*, store the right part of the last official label for-1 in a list and add one label full of 0s.
2. Merge the right parts into one.
3. Skim, based on Edwards' rules.

The "*use_previous_full*" function will merge the left and right sides to get the full label and, after skimming by Edwards' rules, return the accepted label for *p* players and rounds. We take an example from Table 2 with a situation for 7 players and 5 rounds. The example can be seen in Table 3.

Table 3.
Seven players and five round

<table>
<tr><td colspan="4">use_previous_left (7.5) returns the accepted full labels for four rounds</td></tr>
<tr><td>[21101000]</td><td>[21101010]</td><td>[21111000]</td><td>[22101000]</td></tr>
<tr><td colspan="4">use_previous_right (7.5) returns the accepted right halves labels for four rounds and the label full of 0s</td></tr>
<tr><td>[1000]</td><td>[1010]</td><td>[1110]</td><td>[0000]</td></tr>
<tr><td colspan="4">After the concatenation between each one, and skimming by Edwards' rules</td></tr>
<tr><td>[10001000]</td><td>[10101000]</td><td colspan="2">[10000000]</td></tr>
<tr><td colspan="4">use_previous_full (7.5)</td></tr>
<tr><td>[2110100010001000]</td><td>[2110101010000000]</td><td>[2111100010000000]</td><td>[2210100010000000]</td></tr>
</table>

Using two recursive functions, *use_previous_left (p,r)* and *use_previous_right (p,r)*, allows labels of greater length to be generated. Indeed, with this function, the algorithm could produce labels with lengths greater than 8. With these improvements, labels up to 256 in length can be produced easily.

Add participant to tournament fixure

This study used a perfect binary tree, where all external nodes are at the same level, to represent the possibility of each tournament fixture. Dummy players are added to satisfy external nodes in the perfect binary tree with a relation $2^r = p + D$. Each player in this set should be given a power stored in a vector of size 2^r. Setting a strength allows the player to be ranked to determine the difference in the chance of victory.

This research randomly chose to express these abilities with a random integer ranging from 1 to 21, where 1 signifies the minimum strength and 21 represents the maximum. All dummy players were assigned a strength value of zero. To determine in which position the dummy

2024, Retos, 57, 462-472

player will be placed on the external node of a perfect binary, a function is employed to convert the structural label, made up of the digits 2, 1, and 0, into a sequence consisting solely of 1s and 0s. This sequence represens each external node in the perfect binary tree. An external node, indicated by the digit 1, contains an actual player, whereas those represented by the digit 0 contain dummy players.

The tournament fixture "11111112" is a potential configuration for a nine-players match with seven dummy players. When interpreted in relation to external nodes, it generates the sequence "1010101010101011", as illustrated in Figure 16. In this context, *X* represents the assignment of a real player to the node, while D in blue denotes a dummy player.

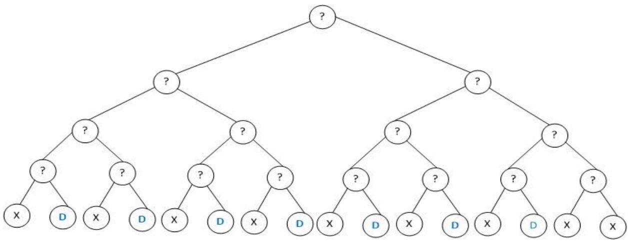

Figure 16. Representation of the external nodes sequence "1010101010101011."

Generating players strengths

Then, we perform crossover process on the external node sequence and the vector containing the players' strengths using a two-cut-point crossover to obtain new offspring (Hassanat et al., 2019) . Hence, the strength of a dummy player is set to zero. The crossover then brings up a new offspring based on the exchange point chosen with particular parts of the vectors (Yao et al., 2020). Figure 17 depicts a crossover process for the production of two offspring.

Cut point / Cut point

external nodes sequence	1 0 1 0 1 0 1 0 | 1 0 1 0 | 1 0 1 1
Players strength	1 2 3 4 5 6 7 8 | 9 10 11 12 | 13 14 15 16
Offspring1	1 0 1 0 1 0 1 0 9 10 11 12 1 0 1 1
Offspring2	1 2 3 4 5 6 7 8 1 0 1 0 13 14 15 16

Figure 17. Crossover method process for produces two offspring

Next, to obtain the final player strengths vector, we multiply both offspring as shown in Figure 18.

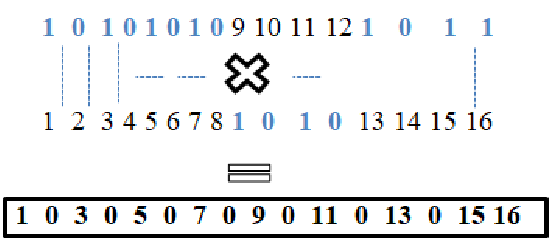

Figure 18. Approach for acquiring the ultimate player strengths vector

This last player's strength vector represents one possible draw for various strengths. However, there are many different possible draws, which is why permutations will be performed inside the actual player- that is, a non-zero force. The permutation vector is depicted in Figure 19.

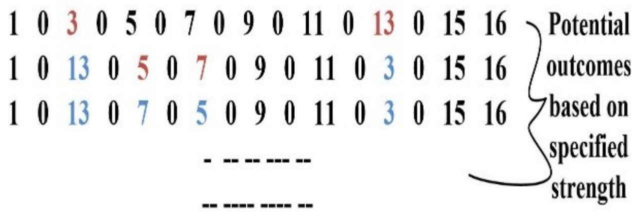

Figure 19. Permutation within the ultimate player strengths vector

All of these final power vectors represent possible draws.

Calculate the probability

To calculate the probability of each player winning in each round, it is used formula in Figure 11 that has been explained in the intoduction. Firstly, calculate the average probability of the strongest player to win for all possible draws with a given strength. Secondly, calculate it for many different strengths. Finally, the result is the strongest player's average probability of winning for a given fixture.

For each possible tournament fixture of player *p*, the probability that player *i* wins the tournament can be calculated. Therefore, by taking the probability of the strongest player to win in each structure and by choosing the maximum, the optimal type of tournament fixture can be concluded. In the same way, selecting a minimum probability reveals a tournament fixture that minimizes the probability of the strongest player winning.

Result

This part displays the outcomes achieved through the algorithm for competitions involving a maximum of 12 players. The pairing process can greatly influence the results of a tournament. As an illustration, suppose the algorithm was executed just once, and the random draw led to the top player facing the second-strongest player in the initial match of the first round. In such a scenario, the probability would be skewed due to an exceptional outcome. Hence, to generate meaningful and valuable outcomes, 100 permutations were applied to the final strengths vector (refer to Figure 19) to create various draws with the same specified strengths. The algorithm was then executed over 1000 iterations. As depicted in Figure 20, illustrating the distribution of the average likelihood of the strongest player winning for the 2211 fixture, the results demonstrate stability, with a marginal standard deviation (s) 0.002.

2024, Retos, 57, 462-472

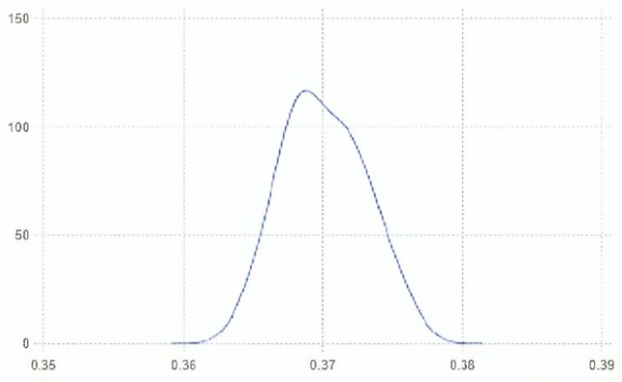

Figure 20. Probability distribution for 2211 fixture

As shown in Table 4, the number of tournament structures and label lengths increased rapidly. The probability calculation becomes computationally expensive for more than eight rounds (i.e., label length 12). However, as can be observed, as the length of the label increases, the probability of the strongest player winning decreases. Therefore, it can be assumed that the optimal label will be visible by simply calculating the "short" label.

Table 4.
Strongest player's probability of winning in tournament fixture

Players	Possible fixture (labels)	Average strongest player's probability of winning
Two players	2	0.577
Three players	21	0.511
Four players	22	0.442
	2110	0.410
Five players	2111	0.365
	2210	0.354
	21101000	0.334
Six players	2121	0.330
	2211	0.328
	22101000	0.291
	21111000	0.301
	21101010	0.314
	2110100010000000	0.288
Seven players	2221	0.330
	22111000	0.261
	21211000	0.271
	21111010	0.282
	21101110	0.281
	22101010	0.274
	2111100010000000	0.251
	2110101010000000	0.261
	2210100010000000	0.271
	2110100010001000	0.252
	21101000100000001000000000000000	0.241

Without showing all the different labels, the optimal fixture for tournaments with more than eight players can be seen in Table 5.

As can be observed, for each tournament, the balanced fixture provides the highest winning probability for the strongest players. Indeed, for tournaments with $p = 2^r + k$ players, a structure with k matches in the first round followed by 2^r in the second round maximizes the probability of the strongest play winning in the tournament.

However, as calculated earlier in Table 1, by using formula in Figure 14, different balance fixtures may appear for some tournaments. For example, when the number of players equals six, two possible tournament fixtures are equally depicted in Figure 21.

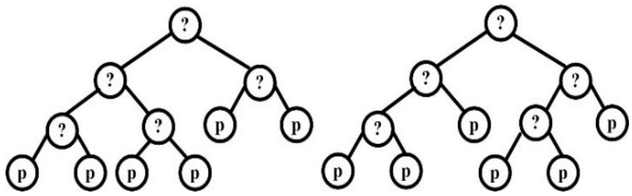

Figure 21. Possible balanced tournament fixture for a six-player tournament

In the left fixture tournament of Figure 21, two first-round matches are at the top of the structure. In contrast, in the right structure, two first-round matches are separated, one at the top, one at the bottom of the top, and one at the bottom of the fixture tournament. As a result of the calculated probability, the tournament fixture on the right provides a slightly higher probability of victory.

One last algorithm was developed to get the optimal label directly for the strongest player. For each p player, this algorithm returns a balanced tournament fixture that will maximize the strongest player's chances of winning the tournament. The same recursion principle used in the previous algorithm is used to do this. For the number of rounds smaller than 3, possible labels with lengths 2^{r-1} with $r = \lceil \log_2 p \rceil$. The only change made is that the label can only contain 2s or 1s. Indeed, since only a small number of dummy players were introduced to achieve a perfect binary tree, the label would never contain 0.

The number 2s represents the pre-round match, and the number 2s on the optimal label equals the k match. This implies that the sum of the number 1s will be equal to $2^r - k$:

Recall: $p = 2^r + , 0 \leq k < 2^r$

Therefore, the only difficulty is knowing where to place the 2s.

By observation, it can be concluded that if the number of players is even, which also implies that k is zero or even, the number of pre-round matches, i.e., 2s, should be evenly distributed between the top and bottom tables, i.e., evenly distributed between the left and right parts of the label. On the other hand, when the number of players is odd, the k is odd, and thus, the match $\lfloor k/2 \rfloor + 1$ must be established in the top sub-fixture (the left part of the label) and the match $\lfloor k/2 \rfloor$ in the lower sub-fixture (the right part of the label).

When the number of turns is r higher than or equal to 3, the same recursion principle used in the previous algorithm is used. Indeed, the left side of the label is optimal for tournaments with fewer players. The right side of the label is the optimal label for tournaments with fewer players or a sequence of only 1s. The output of all these rules gives a unique label representing a tournament fixture balanced for *p* players can be seen in Table 6.

Table 5.
Strongest player's probability of winning in tournament fixture

Players	Possible fixture (labels)	Average strongest player's probability of winning
Eight players	2222	0.280
	21101111	0.261
	21111110	0.251
	21102110	0.252
Nine players	21211010	0.252
	22211000	0.231
	22111010	0.253
	22101110	0.254
	21112110	0.240
	21111111	0.244
	22102110	0.235
	22101111	0.241
Ten players	21211110	0.236
	22111110	0.234
	22211010	0.228
	22221000	0.222
Eleven players	21112111	0.226
	22102210	0.218
	22102111	0.222
	21212110	0.221
	21211111	0.224
	22112110	0.218
	22111111	0.221
	22211110	0.215
	22221010	0.210
Twelve players	21212111	0.201
	21212210	0.206
	22112111	0.201
	22112210	0.202
	22212110	0.201
	22211111	0.204
	22221110	0.203
	21212121	0.198
	22112211	0.195
	22112121	0.198
	22212210	0.191
	22212111	0.192
	22222110	0.191
	22221111	0.194

Table 6.
Balanced Tournament Fixture based on Label

Number of players, p	Balanced tournament fixture
2	2
3	21
4	22
5	2111
6	2121
7	2221

Discussion

In essence, our research delved into the hierarchical arrangements within knockout tournaments. Our aim was to devise a method for generating all potential fixtures for a knockout tournament involving p players and identifying the most optimal arrangement. Given the exponential growth in possibilities as the player count rises, we developed an algorithm to systematically determine various fixtures and calculate the likelihood of victory for the strongest participant. As the absent structures consistently lack balance and only emerge in tournaments with eleven or more players during specific rounds, our findings remain robust. It's unlikely that any of these absent structures would favor the strongest player's chances of winning. The same approach used to assess the probability of the strongest player winning was employed to evaluate the probability of the weakest player winning. Review and analysis of similar studies that have been conducted by Ikhwani (Ikhwani et al., 2023) and (Prayoga et al., 2024). However, in their work they did not use the crossover process to generating players strengths and speed up the computation of large knockout tournaments. In our work, we perform crossover process to recombination of the set of players strength that will be randomly selected from the external node squence that has been formed from the selection process. Crossover process can speed up the computation of large knockout tournaments and will produce a set of offspring becomes more optimal (Koohestani, 2020) whose diversity will be maintained by the next process, multiply both offspring.

Conclusion

The only type of tournament fixture that often considered is the balance, where there are $p/2$ matches in the first round, with p number of players, in cases where the number of players is the power of 2. If the number of players is not a power of 2, k matches are played in the first round, with $p = 2^r + k$ dan di mana $0 \leq k < 2^r$, followed by a balanced tournament fixture. Based on this experiment, the research question in knockout tournaments is the type of tournament fixture that can optimize players' probability of winning the match. Therefore, the goal is to generate all possible tournament fixtures that differ from knockout tournaments with p players and find the optimal one. Since the number of possibilities increases sharply with the number of players, an algorithm is developed to determine the different structures and calculate the probability of victory for the strongest players. Based on the results, it can be concluded that a balanced tournament fixture can effectively minimize the probability of winning the game in a single-elimination tournament. Furthermore, for a tournament with $p = 2^r + k$ players, k matches played in the first round must be evenly placed within the structure to maximize the probability that the player will win. If k is even, k/2 matches should be placed in the top sub-fixture, and k/2 matches in the bottom sub-fixture. If k is odd, $\lfloor k/2 \rfloor + 1$ matches must be placed in the top sub-fixture and $\lfloor k/2 \rfloor$ matches in the bottom sub-fixture. We also conclude, even in more general cases, that the probability of the weakest player winning is maximized under a totally unbalanced fixture and minimized under a balanced structure with *k* matches in the first round.

Conflict of Interest

The authors declare no potential conflicts of interest related to the research, authorship, and publication of this article.

References

Adler, I., Cao, Y., Karp, R., Peköz, E. A., & Ross, S. M. (2017). Random Knockout Tournaments. Operations Research, 65(6), 1589–1596. https://doi.org/10.1287/opre.2017.1657

Arlegi, R., & Dimitrov, D. (2020). Fair elimination-type competitions. European Journal of Operational Research, 287(2), 528–535. https://doi.org/10.1016/j.ejor.2020.03.025

Bădică, A., Bădică, C., Buligiu, I., Ciora, L. I., & Logofătu, D. (2021). Dynamic Programming Algorithms for Computing Optimal Knockout Tournaments. Mathematics, 9(19), 2480. https://doi.org/10.3390/math9192480

Bhumipol, P., Makaje, N., Kawjaratwilai, T., & Ruangthai, R. (2023). Match analysis of professional Muay Thai fighter between winner and loser. Journal of Human Sport and Exercise, 18(3). https://doi.org/10.14198/jhse.2023.183.12

Brito De Souza, D., López-Del Campo, R., Resta, R., Moreno-Perez, V., & Del Coso, J. (2021). Running Patterns in LaLiga Before and After Suspension of the Competition Due to COVID-19. Frontiers in Physiology, 12, 666593. https://doi.org/10.3389/fphys.2021.666593

Bubna, K., Trotter, M. G., Polman, R., & Poulus, D. R. (2023). Terminology matters: Defining the esports athlete. Frontiers in Sports and Active Living, 5, 1232028. https://doi.org/10.3389/fspor.2023.1232028

Cordellat Marzal, A., & Valenciano, R. (2022). Estudio descriptivo sobre el uso del auto-habla en tenistas profesionales (Descriptive study on the use of self-talk in professional tennis players). Retos, 45, 996–1001. https://doi.org/10.47197/retos.v45i0.93132

Csató, L. (2023). A paradox of tournament seeding. International Journal of Sports Science & Coaching, 18(4), 1277–1284. https://doi.org/10.1177/17479541221141617

David, H. A. (1959). Tournaments and Paired Comparisons. Biometrika, 46(1/2), 139. https://doi.org/10.2307/2332816

Dong, Z.-L., Ribeiro, C. C., Xu, F., Zamora, A., Ma, Y., & Jing, K. (2023). Dynamic scheduling of e-sports tournaments. Transportation Research Part E: Logistics and Transportation Review, 169, 102988. https://doi.org/10.1016/j.tre.2022.102988

Driver, T. C., & Hankin, R. K. S. (2023). Analysis of competitive surfing tournaments with generalized Bradley-Terry likelihoods. Journal of Sports Analytics, 9(2), 133–140. https://doi.org/10.3233/JSA-220596

Edwards, C. T. (1996). Double-Elimination Tournaments: Counting and Calculating. The American Statistician, 50(1), 27–33. https://doi.org/10.1080/00031305.1996.10473538

Ekin, C. C., Polat, E., & Hopcan, S. (2023). Drawing the big picture of games in education: A topic modeling-based review of past 55 years. Computers & Education, 194, 104700. https://doi.org/10.1016/j.compedu.2022.104700

Gao, S., & Mahmoud, H. (2023). Winning a Tournament According to Bradley-Terry Probability Model. Statistics, Optimization & Information Computing, 11(2), 332–344. https://doi.org/10.19139/soic-2310-5070-1490

Guyon, J. (2022). “Choose your opponent”: A new knockout design for hybrid tournaments†. Journal of Sports Analytics, 8(1), 9–29. https://doi.org/10.3233/JSA-200527

Hassanat, A., Almohammadi, K., Alkafaween, E., Abunawas, E., Hammouri, A., & Prasath, V. B. S. (2019). Choosing Mutation and Crossover Ratios for Genetic Algorithms—A Review with a New Dynamic Approach. Information, 10(12), 390. https://doi.org/10.3390/info10120390

Hulett, R. (2019). Single-Elimination Brackets Fail to Approximate Copeland Winner [Application/pdf]. 20 pages, 484510 bytes. https://doi.org/10.4230/LIPICS.APPROX-RANDOM.2019.13

Ikhwani, Y., Ramadhan, A., Bahit, M., & Faesal, T. H. (2023). Single elimination tournament design using dynamic programming algorithm. MATRIK : Jurnal Manajemen, Teknik Informatika Dan Rekayasa Komputer, 23(1), 113–130. https://doi.org/10.30812/matrik.v23i1.3290

King, M. C., & Rosenberg, N. A. (2023). A Mathematical Connection Between Single-Elimination Sports Tournaments and Evolutionary Trees. Mathematics Magazine, 96(5), 484–497. https://doi.org/10.1080/0025570X.2023.2266389

Koohestani, B. (2020). A crossover operator for improving the efficiency of permutation-based genetic algorithms. Expert Systems with Applications, 151, 113381. https://doi.org/10.1016/j.eswa.2020.113381

Lee, E., Masuda, M., & Park, S. (2023). Toric Richardson varieties of Catalan type and Wedderburn–Etherington numbers. European Journal of Combinatorics, 108, 103617. https://doi.org/10.1016/j.ejc.2022.103617

Lucas, J. M., Vanbaronaigien, D. R., & Ruskey, F. (1993). On Rotations and the Generation of Binary Trees.

Journal of Algorithms, 15(3), 343–366. https://doi.org/10.1006/jagm.1993.1045

Manurangsi, P., & Suksompong, W. (2022). Generalized kings and single-elimination winners in random tournaments. Autonomous Agents and Multi-Agent Systems, 36(2), 28. https://doi.org/10.1007/s10458-022-09557-7

Manurangsi, P., & Suksompong, W. (2023). Fixing knockout tournaments with seeds. Discrete Applied Mathematics, 339, 21–35. https://doi.org/10.1016/j.dam.2023.06.012

Maurer, W. (1975). On Most Effective Tournament Plans With Fewer Games than Competitors. The Annals of Statistics, 3(3), 717–727. JSTOR.

Musa, R. M., K.Suppiah, P., Abdullah, M. R., Majeed, A. P. P., & Razmaan, M. A. M. (2022). Positional differences in the performance of volleyball players for anthropometric and psychological readiness in a congested fixture tournament. Journal of Physical Education and Sport, 22(4). https://doi.org/DOI:10.7752/jpes.2022.04127

P. Parande, N. Sorthiya, M. Lanje, N. Dudhuke, & P. Maidamwar. (2023). Dynamic Grouping of Players and Analysis for Regional Tournaments. 2023 5th International Conference on Smart Systems and Inventive Technology (ICSSIT), 1451–1456. https://doi.org/10.1109/ICSSIT55814.2023.10061151

Paixao, P., Giménez Fuentes-Guerra, Fco. J., Navarro Domínguez, B., Cerrada Nogales, J. A., Robles Rodríguez, J., & Abad Robles, M. T. (2021). Perfil y concepción de la enseñanza del entrenador de fútbol base de la región de Beja (Portugal) (Profile and conception of the teaching of the basic football coach in the region of Beja (Portugal)). Retos, 42, 344–352. https://doi.org/10.47197/retos.v42i0.87365

Prayoga, H. D., Ramadhan, A., Kasandrawali, A., & Setiawan, K. (2024). PEMBUATAN DAN PELATIHAN APLIKASI BRACKET PERTANDINGAN MUAYTHAI DI PENGURUS PROVINSI MUAYTHAI KALIMANTAN SELATAN. RESWARA: Jurnal Pengabdian Kepada Masyarakat, 5(1), 25–32. https://doi.org/10.46576/rjpkm.v5i1.3515

Pridal, V., & Priklerova, S. (2018). Analysis of relation between team placing in tournament and selected indicators of playing performance in top-level volleyball. Journal of Physical Education and Sport, 2018(03), 1501–1505. https://doi.org/DOI:10.7752/jpes.2018.03221

Rojas-Valverde, D., Gómez-Carmona, C. D., Oliva-Lozano, J. M., Ibáñez, S. J., & Pino-Ortega, J. (2020). Quarter's external workload demands of basketball referees during a European youth congested-fixture tournament. International Journal of Performance Analysis in Sport, 20(3), 432–444. https://doi.org/10.1080/24748668.2020.1759299

Sobkowicz, P., Frank, R. H., Biondo, A. E., Pluchino, A., & Rapisarda, A. (2020). Inequalities, chance and success in sport competitions: Simulations vs empirical data. Physica A: Statistical Mechanics and Its Applications, 557, 124899. https://doi.org/10.1016/j.physa.2020.124899

Sziklai, B. R., Biró, P., & Csató, L. (2022). The efficacy of tournament designs. Computers & Operations Research, 144, 105821. https://doi.org/10.1016/j.cor.2022.105821

Yao, J., Shi, H., & Liu, C. (2020). Optimising the configuration of green supply chains under mass personalisation. International Journal of Production Research, 58(24), 7420–7438. https://doi.org/10.1080/00207543.2020.1723814

Datos de los/as autores/as y traductor/a:

Hegen Dadang Prayoga	hegendadang.2023@student.uny.ac.id	Autor/a – Traductor/a
Tomoliyus	tomoliyus@uny.ac.id	Autor/a
Ria Lumintuarso	ria_l@uny.ac.id	Autor/a
Yudik prasetyo	yudik@uny.ac.id	Autor/a
Endang Rini Sukamti	endang_fik@uny.ac.id	Autor/a
Ari Tri Fitrianto	aritri.fitrianto@uniska-bjm.ac.id	Autor/a
Andi Kasanrawali	kasandrawali89@gmail.com	Autor/a
Ramadhan Arifin	ramadhan.arifin@ulm.ac.id	Autor/a
Ahmad Maulana	ahmadmaulana@uniska-bjm.ac.id	Autor/a
Muhammad Habibie	muhammadhabibie@uniska-bjm.ac.id	Autor/a