Action Sequence Modeling for Tactical Training in Handball

Luke Wildman[1][0009−0003−8003−0401], Roland Nemes[2][0009−0006−1203−1834], and Zhe Hou[3][0000−0001−7164−0580]

[1] Brisbane Handball Club, Brisbane, Australia
[2] Hosei University, Tokyo, Japan
[3] Griffith University, Brisbane, Australia

Abstract. Handball is a highly dynamic and complex team sport, characterized by continuous player interactions, rapid transitions between attack and defense, and frequent decision-making under pressure. These factors create significant challenges for formal tactical modeling and performance analysis, as highlighted in previous systematic reviews of match analysis and action sequence complexity in handball. Unlike more discretized sports like baseball or even football, handball's fluidity demands advanced methods to capture and simulate strategic behaviors effectively. This study investigates a novel approach for analyzing handball tactical sequences by applying Probabilistic Model Checking (PMC) to model player actions, decisions, and outcomes. Using Markov Decision Processes (MDPs) and the Process Analysis Toolkit (PAT), we construct probabilistic simulations of handball attacks to evaluate how incremental improvements in player performance — such as passing accuracy, shooting effectiveness, or decision timing — impact overall team success rates.

Keywords: Handball· Sports analytics · Probabilistic model checking

1 Introduction

Handball (also known as European Handball and Team Handball) [16] has been an Olympic sport since 1972 and is estimated to be played by 30 million players by the International Handball Federation [8] and as such is considered to be one of the most popular team sports in the world. Handball action sequences have been studied by Tilp and others [22], [20], [21]. Action sequences are combinations of actions performed by players or teams during offensive or defensive phases of the game. Action sequences are influenced by tactical and situational variables such as the type of defence, the score difference, the game period, and the quality of the opponent. A systematic review of Handball Match Analysis was performed by Ferrari *et al* in 2019 [4].

This paper considers the use of model-checking technology (widely used in the analysis of mission critical systems) to take a probabilistic model (a Markov Decision Process) of handball play to calculate the probability of scoring. Model-checking technology can also be used for witness and counter-example generation to generate example action sequences for inspection. A Markov decision process (MDP) is a mathematical model for sequential decision-making under uncertainty. It consists of a set of states, a set of actions, and transition probabilities that depend on the current state and action. MDPs have been used to model and solve control problems for stochastic systems, such as robotics, planning, reinforcement learning, and so on. They have also been applied to model sports action sequences, such as possessions in football, baseball, tennis, and other sports where the goal is to measure the contribution of each action to the final outcome.

Our intended use of the model is to investigate the possible benefits of making incremental marginal improvements to the decision making and performance characteristics of the team players (through targetted training activities). We follow the approach of using MDP to model decisions and performance described in [13] and [9]. Building on initial work with a junior women's team preparing for a minor tournament, we have now expanded the model's validation by integrating datasets from matches played by two EHF Champions League teams (one male, one female) and a female Youth World Championship team. This expanded dataset allows comparison of tactical modeling across different age groups, genders, and performance levels, offering new insights into how decision-making patterns evolve with skill and experience.

The broader goal of the project is to develop methods that leverage machine learning and formal probabilistic modeling to help coaches identify strengths and weaknesses in both their own teams and their opponents. By simulating various tactical scenarios and player combinations, coaches can make informed decisions to maximize efficiency, adapt training activities, and increase the team's winning probability. Furthermore, the framework moves toward analyzing and understanding player decision-making processes — a major frontier in sports analytics — by making action sequences and tactical options explicitly visible and quantifiable. Our findings demonstrate that Probabilistic Model Checking offers a highly transparent, explainable alternative to conventional machine learning models. It enables simulation-driven strategy development without requiring massive datasets, making it practical even for teams with limited analytical resources. Future work will aim to incorporate dynamic defensive adjustments, expand the model's action space, and integrate automated video analysis pipelines to enable large-scale deployment.

2 Related Work in Decision Making in Handball

Decision-making in handball operates at the intersection of rapid perception, tactical cognition, and high-stakes interaction, presenting unique challenges for formal modeling. Unlike other team sports with broader spatial-temporal margins, handball compresses cognitive demands into milliseconds, requiring players to anticipate, decide, and act in dynamically shifting contexts [1, 6]. Defensive decisions, in particular, rely heavily on the real-time coupling between outfield defenders and goalkeepers, who must interpret subtle kinematic cues (e.g., shoulder tilt, arm movement) to predict shot direction and intent before execution [5,11]. These decision sequences are rarely isolated—they emerge from fluid, adaptive interplay between contextual variables such as game score, fatigue, and opponent tendencies [15,19]. While machine learning models can effectively capture structured game behavior and probabilistic state transitions (e.g., pass networks, spatial densities), the modeling of internal decision-making processes is hindered by a lack of labeled perceptual-cognitive data and ground truth for "intent" [3, 12]. Therefore, we defer the modeling of decision-making components until more granular multimodal datasets—such as eye-tracking, biomechanical markers, and real-time emotional state indicators—become available. In addition to the work on decision-making in handball, there is a large body of work on the application of machine learning to handball. Ichimura et al. [7] have applied machine learning to the prediction of handball player performance. Mizuno et al. [18] have applied random forest to the prediction of handball player performance. Marczinka et al. [17] consider technical elements in defence focusing on the differences between positions and genders.

3 PCSP# language overview

To model the handball play, we use the modelling language Probabilistic Communicating Sequential Programs (PCSP#) [14] and the Process Analysis Toolkit (PAT) [24] as the model checker. Given a model of the desired system expressed in *PCSP#*and its desired properties, PAT will automatically and exhaustively search all possible cases to verify if the system satisfies the desired property. When given a model which includes probabilistic choice and a probabilistic reachability property, PAT will calculate the probability of reaching the desired states [23]. A system is modelled using a set of variables and processes. A variable is usually an integer or an enumerable type within a certain range to ensure the number of states is countable and finite. When any of the variables is assigned a different value, the system is considered to have transited into a different state. More information on the use of model checkers and the algorithms used may be found in [14] and the references above. Another widely used alternative model-checking tool for probabilistic system is Prism [10]. We now introduce the relevant aspects of the modelling language via a handball example.

4 Basic model of a Handball Attack

Handball is played on a $20 * 40m$ court with $2 * 3m$ goals at both ends. Around each goal is a $6m$ line, in a D-shape marking the goal area (which is inclusive of the line). There is also a $9m$ dashed line which simply signifies the distance of $3m$ required for free throws, and a $7m$ mark used for taking penalties. See Figure 2. There are normally 7 players on court for each side, made up by 6 field players and 1 goal keeper. There are also several substitutes for each side with substitution over the side-line occurring any time and allowed multiple times per player. Only the goalkeeper is allowed to play in the goal area. Field players cannot play the ball while standing in the goal area, or gain advantage by moving through the goal area. Shots on goal must be taken from outside the goal area or while in the air over the goal area before touching down. Similarly defenders may not defend from inside the goal area. As for other invasion games, the usual modes of attack are fast breaks (in the case that the defence is disorganised) or attacks on an organised zone defence around the goal area. The attacking players are usually organised in a D-shape around the defence and the usual positions are labelled accordingly: *left wing*, *left back*, *centre back*, *right back*, *right wing* and lastly the *line player* or *pivot* who plays in the middle, among the defence on the goal area line.

A very simple model of a handball attack around the goal area (or 'D') is now provided for further introduction to the modelling language. We model the progress of the attack by the location of the players and ball relative to 14 zones. Zones 1 to 5 represent sequential positions around the goal area line from *left-wing* to *right-wing*, zones 6, 7, and 8 represent 3 back positions between the goal line and the 9-metre line, for *left back*, *centre back* and *right back* respectively, and zones 9, 10, and 11 represent three back positions behind the 9-metre line. Finally, zone 0 is inside the goal and represents scoring a goal. Zone 12 is used to represent an attacking position behind the backs to the half-way line from where it is very difficult to score by a direct shot on goal. Zone 13 represents the other half of the court, which from the point of view of the attacking team represents the defensive zone. Zone 14 is used to represent states where the attacking team loses the ball (causing a turn-over) by the ball being played outside of the playing court or missing a shot, or the ball being intercepted by the defence (including the goal keeper by a save), or otherwise being turned over because of an attacking foul or other reason. We use a single variable called *ball* to represent the zone occupied by the ball. The location of the players (in the order LW LB CB RB RW PV) is captured in the array variable *aloc*. For example, $aloc = [1, 6, 7, 8, 5, 3]$ models the normal positions occupied by the attacking players when facing a 6-0 defence where the defensive players are lined up around the goal area and $ball = 7$ represents it being held by the centre back, given the attacking team is in possession of the ball.

Fig. 1. Handball court with positions marked

We model the defence positions $d1$ to $d5$ representing the five zones around the goal area and the variable *gap* to capture whether there is gap in one of those zones. For example, $gap[d1] == 1$ represents a gap in zone 1. Note this model is adequate for a simple 6-0 defence but will need development for more complex defence arrangements. (See Section 7.2)

Finally, we keep track of the number of passes in the attack with the variable *pass*. This is due to the rules regarding *passive play* where the referees may force a turnover if it appears that the attacking team is not consistently attacking the goal. This call is made by the referees based on their read of the play, but as a general rule the number of passes in an attack is less than 15 [25]. We define a constant *maxpass* to represent this. The predicate *inplay* describes the states where the ball remains in play. Predicates are logical expressions involving the model variables and may be used in conditional expressions in the model. In this case *inplay* is true if the ball is not in the goal ($ball \mathrel{!}= 0$) or not lost/turned over ($ball \mathrel{!}= 14$), and that the referee has not called passive play ($pass < maxpass$).

Given the above very simple model of the state, we model the attacking play by describing each player as a process. The *CB* process below models the probabilistic choice of gameplay decisions made by the centre back faced with specific situations. The first line models the condition that the centre back may only play the ball if it is *inplay*. This is expressed as a conditional process **if** (*cond*) { P } **else** { Q } that behaves as P if the condition *cond* holds or otherwise behaves as Q. In this case Q is the special process *Skip* which just terminates because the ball is not in play.

$$
\begin{array}{l}
CB \;=\; if\ (inplay)\ \{\ case\ \{\ aloc[cb]\ ==\ 7\ \&\&\ gap[d3]\ ==\ 0\ :\ pcase\ \{\\
\quad 30\ :\ cb\_\ lb\ \rightarrow\ short\_\ pass(cb,\ lb);\ \ LB\\
\quad 30\ :\ cb\_\ rb\ \rightarrow\ short\_\ pass(cb,\ rb);\ \ RB\\
\quad 10\ :\ cb\_\ pv\ \rightarrow\ pivot\_\ pass(cb,\ pv);\ \ PV\\
\quad 10\ :\ cb\_\ lw\ \rightarrow\ long\_\ pass(cb,\ lw);\ \ LW\\
\quad 10\ :\ cb\_\ rw\ \rightarrow\ long\_\ pass(cb,\ rw);\ \ RW\\
\quad 5\ :\ cb\_\ m10\ \rightarrow\ move\_\ wball(cb,\ 10);\ \ CB\\
\quad 5\ :\ cb\_\ shot\ \rightarrow\ shot(cb)\\
\quad \}\ default\ :\ undef\ \rightarrow\ Skip\\
\}\}\ else\ \{Skip\};
\end{array}
$$

The situation described by this process is that the centre back occupies location 7 on the field and there is no gap in defensive position $d3$ directly in front of them ($gap[d3] == 0$). We only describe one situation, but in general many situations may be described where each situation is captured as a branch of a **case** process. In general the case process is written **case** { $b_1 : P_1\ b_2 : P_2$ ... *default* : P_d } where the combined case process behaves as a subprocess P_i if the associated condition b_i holds. The conditions are evaluated one by one until a true one is found and if no conditions hold, the case process behaves as the default P_d. In this basic situation we have estimated a probability of 30% that the *CB* passes the ball to the left back (*LB*), a 30% probability that they pass the ball to the right back (*RB*), and so on, with a 5% chance of shooting. This is modelled by a probabilistic choice **pcase** { $p_1 : P_1$ $p_2 : P_2$ ... $p_n : P_n$ } where the probabilities (p_i) are normalised to sum up to 1. Each branch of the probabilistic process (P_i) in the above example identifies events (e.g., *cb_lb*, *cb_rb*); an action to be executed; and then the next player taking control of the ball (and the overall attack process). For example, the process description states that there is a 10% chance of passing to the pivot, which is modelled by the event *cb_pv* followed by an action *pivot_pass* from the centre back to the pivot, and then if the pass is successful, the PV process takes over. The PV process models what happens when the ball is received by the pivot. The probabilities related to the pivot actions are different to the centre back. That is, in the situation where the pivot receives that ball on the line and there is a gap at that defensive position, then the pivot will almost certainly shoot and a very small percentage of the time they may do something else such as pass the ball to the centre back.

$$
\begin{array}{l}
PV \;=\; \mathit{if}\ (\mathit{inplay})\ \{\ \mathit{case}\ \{\ \mathit{loc}[pv] == \ \ \&\&\ \ \mathit{gap}[d3] == 1 : \\
\ \mathit{pcase}\ \{ \\
\quad 99\ :\ \mathit{pv\_\ shot}\ \rightarrow\ \mathit{shot}(pv) \\
\quad\ \ 1\ :\ \mathit{pv\_\ cb}\ \rightarrow\ \mathit{short\_\ pass}(pv, cb)\ ;\ \ CB \\
\} \\
\}\}\ \mathit{else}\ \{\mathit{Skip}\};
\end{array}
$$

Other player positions are described similarly. Decisions are dependent on the court situation: player position, ball position, defence position, and whether the attacker has an unhindered path to the goal line to shoot. They may also depend on the performance of defence (e.g., the goal keeper may be particularly good at saving goals from the wing making the winger less likely to take a shot from zones 1 or 5).

5 Player performance

We capture the performance of the players with arrays recording the probability that a short pass, long pass, or pass to the pivot succeeds (there are N players). This success rate is calculated based on statistics gathered in matches and scaled against elite player performance [25]. Values are elided for formatting purposes.

$$
\begin{array}{l}
\mathit{var}\ \mathit{short\_pass\_succ}[N] \;=\; [98, ..., 98]; \\
\mathit{var}\ \mathit{long\_pass\_succ}[N] \;=\; [85, ..., 84]; \\
\mathit{var}\ \mathit{pivot\_pass\_succ}[N] \;=\; [60, ..., 50];
\end{array}
$$

Process $\mathit{short\_pass}(pl, rp)$ below models the success of passing a ball from player pl to receiver rp. If the pass is successful (event $spass.pl.rp$), then the ball goes to the location of the receiving player and the number of passes increments by 1. If the pass fails (event $to.pl$), then the ball goes to location 14 representing a turn over. The probability of failure is 100 minus the probability of success. Processes *long_pass* and *pivot_pass* are defined similarly and are not shown in this paper.

$$
\begin{array}{l}
short\_pass(pl,\ rp)\ =\ pcase\ \{\\
\ short\_pass\_\ succ[pos[pl]]:\ spass.pl.rp\ \{ball = loc[rp];\ \ pass{+}{+}\}\ \rightarrow\ Skip\\
\ 100 - short\_pass\_succ[pos[pl]]:\ to.pl\ \{ball = zout\}\ \rightarrow\ Skip\};
\end{array}
$$

The effectiveness of a shot is captured in a similar way for each player and each shooting position in the array variable *shot_effect*. Values are elided for formatting purposes. The effect of the shot, if successful (event *goal*), is represented by setting the ball location to 0 Otherwise the event *miss* is represented by setting the ball location to zone 14.

$$
\begin{array}{l}
shot(n)\ =\ pcase\ \{\\
\ shot\_effect[ball-1][pos[n]] : goal\{ball = 0\}\ \rightarrow\ Skip\\
\ 100 - shot\_effect[ball-1][pos[n]] : miss\{ball = zout\}\ \rightarrow\ Skip\ \};
\end{array}
$$

6 Model Simulation

The handball model is simulated by providing an assertion to be checked and application of the PAT model checker. Assuming that an attack starts with the centre back, we define the process *Play* as *CB*. We define the proposition *scoregoal* as the state where the *goal* variable is equal to 0. The assertion to be checked is then defined as follows.

$$
\begin{array}{l}
Play\ =\ CB;\\
\#define\ scoregoal\ ball\ ==\ 0;\\
\#assert\ Play\ reaches\ scoregoal\ with\ prob;
\end{array}
$$

When executed, PAT returns the probability that the assertion is valid with minimum and maximum values. E.g. $[0, 0.4832]$. A minimum probability of 0 represents the fact that it is possible that the play fails to score.

Another useful aspect of model checking is the generation of witness sequences and counter examples. Consider the following reachability assertion:

$$\#assert\ Play\ reaches\ scoregoal;$$

PAT returns that the assertion is valid and provides the following trace (with some simplifications) as a witness.

$$
\begin{array}{l}
<init\ \rightarrow\ [\ if\ (inplay)]\ \rightarrow\ [(loc[cb]\ ==\ 7)]\ \rightarrow\ [(gap[d3]\ ==\ 1)]\ \rightarrow\\
\quad 0.5\ \rightarrow\ cb\_shot\ \rightarrow\ 0.4\ \rightarrow\ goal>
\end{array}
$$

The above example illustrates some of the basic concepts used in the model of the handball attack. However, there are many limitations to the model and we progressively deal with these in the following sections.

7 Extensions

In our previous work [26], we focused on modeling simple attacks such as fast breaks. The current paper extends this by considering more complex attack patterns and basic defensive structures. The attack patterns analyzed are derived from matches in the EHF Champions League and Youth World Championship, while the defensive modeling is based on observations from the EHF Champions League. Importantly, the model framework we have developed is readily extensible to incorporate additional attack and defence patterns of increasing complexity.

7.1 Attack

We extend the model by incorporating several additional attack patterns: the simple cross, the empty cross /"Jugo" style attack, the second pivot attack pattern, and pivot crossing movements. These represent common tactical variations seen in high-level handball matches (see [7] for more details on common tactical variations in handball). The attack patterns are introduced into the model by encoding the behaviour as in *PCSP#*. We have modified the process description for Center Back behaviour for the *CB* when they are in zone 10 as follows. First it is checked that the *CB*, pivot and left and right backs are in the correct positions. That is, the *CB* is in zone 10, the pivot is in zone 4, the left back is in zone 9 and the right back is in zone 11. The process then encodes the decision to initiate the different styles of attacks with probabilities derived from the frequencies observed in the EHF Champions League.

The event *pvxr* encodes the decision to initiate the pivot cross. Assumming it starts 17% of the time (embedded in a pcase) it starts by the pivot moving to zone 10 followed by the *CB* passing to the pivot. Note that this represents movement of the pivot without the ball. In practice, the pivot usually actually starts moving prior to *CB* receiving the ball, anticipating the pass from the *LB*, and the pivot is in zone 10 when the *CB* receives the ball. However we have simplified the process for the purpose of this explanation. The process then evolves the the *PVXR* process which is described further below. Similarly, if the empty cross on the left (or Jugo left) (*cbexl*) is chosen then the *CB* moves to zone 11 and passes to the *LB*. The process then evolves the the *CBEXL* process which is also described further below. The *CB*10 process also includes the *CBEXR* process which is the same as *CBEXL* but with the *CB* moving to zone 9 and passing to the *RB*, as well as other possible passes, movements and shots as described earlier for basic model.

$$
\begin{aligned}
&CB10 \;=\; [aloc[cb] \;==\; z10 \;\&\&\; ball \;==\; z10] \\
&\quad gap[d3] \;==\; 0 \;\&\&\; aloc[pv] \;==\; z4 \;\&\&\; aloc[lb] \;==\; z9 \\
&\quad\;\; \&\&\; aloc[rb] \;==\; z11 \;:\; pcase\;\{ \\
&\ldots \\
&17:\; pvxr \;\rightarrow\; move(pv, z10);\;\; cb\_\,pv \;\rightarrow\; short\_pass(cb,\; pv);\;\; PVXR
\end{aligned}
$$

The $PVXR$ process is then defined as follows. First the state is checked to see if the ball is in play. This checks that the previous pass was successful and did not result in a turn over due to a bad pass or other attacker error. (Note this is very unlikely, however it is a possibility must be accounted for). Next, we implement the role switch between the *cb* and the *rb*. This represents that the *CB* role is now being played by player who was previously the RB. By describing the process in terms of roles we are able to reuse the same process decriptions for the *CB* and the *RB* without having to wirte player-specific processes. This is done by calling the *switch* process with the *cb* and *rb* roles as arguments. This is followed by the new *cb* moving to zone 10 (completing the empty cross) and evolves into the $PVL10$ process.

$$PVXR \ = \ \mathit{if}(\mathit{inplay}) \ \{sw.cb.rb\{\mathit{call}(\mathit{switch}, cb, rb)\} \\ \quad \rightarrow \ cbm10 \ \rightarrow \ \mathit{move}(cb, z10) \ ; \ \ PVL10\};$$

The switch function is defined as follows. It simply swaps the players in the two roles.

$$\#\mathit{define}\ \mathit{switch}(pl,\ rv)\ \{\mathit{var}\ t = \mathit{pos}[pl];\ \mathit{pos}[pl] = \mathit{pos}[rv];\ \mathit{pos}[rv] = t\};$$

The $PVL10$ process is defined similarly to PV in the basic model, but with the pivot being in zone 10 and Pivot moving to the left back *lb* with the ball. In this case it is mostly likley that the pivot will pass to the *LB* but the description allows for other options such as when there is an open shooting opportunity or the PV decides to pass it elsewhere. (Note, there is obviously a number of possible other variation, but we have only modelled those observed in practice.)

The *CBEXL* (Jugo to the left) process is defined similarly to the Pivot cross process. As part of the initiation desribed above in $CB10$, the centerback moves with the ball to zone 11. This is continued in *CBEXL* which after checking that this has not resulted in a turn over due to an attacker error, the ball is passed to the *LB* (this is a long pass given that the *CB* is in zone 11), switches roles with the rightback *rb*, the new *CB* moves into zone 10, and evolves into the *LB* process, noting that the *LB* has the ball.

$$CBEXL \ = \ \mathit{if}\ \ (\mathit{inplay})\ \{\mathit{long}_{-}\ \mathit{pass}(cb,\ lb); \\ \quad sw.cb.rb\{\mathit{call}(\mathit{switch}, cb, rb)\} \ \rightarrow \ cbm10 \\ \qquad \rightarrow \ \mathit{move}(cb, z10) \ ; \ \ LB\};$$

A slightly different move is the 2nd pivot attack pattern. This starts similalry to the pivot cross but instead of pivot, the wing (left wing in this case) moves out to zone 10 and the *CB* passes the ball to the *LW* the *LW* passes to the *RB* while the CB executes an empty cross with the *LB* (switching roles.

$$LW2PV \ = \ \mathit{if}(\mathit{inplay}) \ \{sw.cb.lb\{\mathit{call}(\mathit{switch}, cb, lb)\} \\ \quad \rightarrow \ cbm10 \ \rightarrow \ \mathit{move}(cb, z10) \ ; \ \ LWR10\};$$

What is different here is that the *LW* changes into a second pivot and moves to zone 4. That is, there is a change of role for that player from a wing role to a second pivot (PV2) role. This is encoded in the following event sequence. Following the LW passing to the RB, the LW transforms to the 2nd pivot role and moves to zone 4. The process then evolves into the RB process as the RB now has the ball.

$$lw\_\,rb \;\rightarrow\; short\_\,pass(lw, rb)\;;\;\; tr.lw.pv2\;\{call(trans, lw, pv2)\} \;\rightarrow$$
$$pv2m4 \;\rightarrow\; move(pv2, z4);\;\; RB$$

The *trans* function transforms the player at role *pl* to the role *rv* as well as their location. The first role (left wing in this case) is no longer being used so we set the role to *pnull* and the player's location to *zout* (out of play).

$$\#define\; trans(pl,\; rv)\; \{pos[rv] = pos[pl];\; pos[pl] = pnull;$$
$$aloc[rv] = aloc[pl];\; aloc[pl] = zout\};$$

7.2 Defence

In this section, we extend the model to include defensive formations and behaviors. Specifically, we incorporate two common defensive systems: the 6-0 defense (where all defenders are positioned along the 6-meter line) and the 5-1 defense (with five players on the 6-meter line and one advanced defender).

6-0 Defence The 6-0 defence is modeled using a simplified representation focused on defensive gaps rather than explicit defender positions. We consider six potential gaps in the defensive formation: five gaps around the goal area (D) and one gap in front for the 5-1 formation. Each gap is represented as a binary state (0 or 1) where 1 indicates the presence of a defensive gap.

The five gaps around the goal area are positioned as follows:

- d1: A gap on the left wing, occurring when either the wing defender is out of position or the adjacent defender fails to provide coverage
- d2: A gap on the left side, arising when the second defender is displaced and neither the first nor third defender compensates
- d3: A gap in the center, representing a significant area where the middle defenders have failed to maintain their formation
- d4: A gap on the right side, following similar principles to d2
- d5: A gap on the right wing, analogous to d1

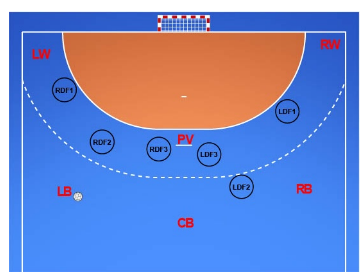

Fig. 2. Handball court with positions marked

Additionally, $d6$ represents a gap at zones 7 and 10, which would allow the center back to attempt shots from these positions. This abstraction captures a key defensive principle: the defence typically only exposes gaps on the side opposite to the ball's location. However, when defenders step out to challenge shots from the backs, gaps may emerge if other defenders fail to adjust their coverage appropriately. The probabilities for gap formation can be calibrated using empirical data collected from different variations of the 6-0 defense (designated as 6-0A, 6-0B, and 6-0C), combined with tactical analysis of defensive behavior.

$$var\ gap[6] \ = \ [0,0,0,0,0,0];$$

The function *CloseGap* models a simplified version of a sliding 6-0 defence with four possible outcomes with the following probabilities:

- 0.1: The defence closes all gaps.
- 0.5: The defence closes a gap in front of the player and leaves a gap on the furthest wing.
- 0.2: The defence opens a gap in front of the player and closes a gap on the furthest wing.
- 0.2: The defence opens a gap in front of the player and leaves a gap on the furthest wing.

$$\begin{array}{l}
CloseGap(x) \ = \ pcase\ \{ \\
\quad [0.1] \ : \ gca.x\ \{resgap5\} \ \rightarrow \ DEF60 \\
\quad [0.5] \ : \ gcwo.x\ \{resgap5; \ \ call(wgap, x)\} \rightarrow \ DEF60 \\
\quad [0.2] \ : \ go.x\ \{resgap5; \ \ gap[admap[aloc[x]]] \ = \ 1\} \rightarrow \ DEF60 \\
\quad [0.2] \ : \ gowo.x \ \ \{resgap5; \ call(wgap, x) \rightarrow \ DEF60\ \};
\end{array}$$

Defence Process The defense is implemented as a separate process that interacts with the attacking processes through defined communication channels. These channels facilitate the exchange of information about player movements, and passing actions. The defensive process takes inputs from the attacking processes, produces appropriate responses through its outputs, and is composed with the attacking processes to create the complete game model.

Channels are a *PCSP#* feature that allows processes to communicate with each other. We use channels of length 0 to indicate that the communication is synchronous, as illustrated by the following.

$$channel\ spass\ 0;$$

Channels are defined for passing and movement events.

We synchronize the defence actions with the events modelling the performance of an action rather than the event describing the decision to perform the action. That is, the defence process is triggered by the observation of the action rather than having read the mind of the attacking player.

Input to the channels is defined using the ! operator. For example, the values *pl* and *rp* are input to the short pass channel *spass* in the following update of the short pass process. The values of *pl* and *rp* are composed to form the value *pl.rp* which is passed to the defence process via the channel.

$$\begin{aligned}&\mathit{short\_pass}(pl,\ rp)\ =\ \mathit{pcase}\ \{\\&\quad \mathit{short\_pass\_\ succ}[pos[pl]] : \mathit{spass}!pl.rp \rightarrow \{\mathit{ball} = \mathit{aloc}[rp];\ \mathit{pass}{+}{+}\} \rightarrow\ \mathit{Skip}\\&\quad 1000 - \mathit{short\_pass\_succ}[pos[pl]] : \mathit{to}.pl\ \{\mathit{ball} = \mathit{zout}\} \rightarrow \mathit{Skip}\ \};\end{aligned}$$

The defence procees uses the output of the channel. The value is read form the channel using the ? operator. In the following example, the value $y.x$ is read from the short pass channel *spass*.

This example is for the 6-0 defence. The process DEF60 calls the process CloseGap with the value x which is the player that received the pass. The CloseGap process is defined further below.

$$\begin{aligned}\mathit{DEF}60\ =\ &\mathit{spass}?y.x\ \rightarrow\ \mathit{CloseGap}(x)\ \Box\\&\mathit{lpass}?y.x\ \rightarrow\ \mathit{CloseGap}(x)\ \Box\\&\mathit{ppass}?y.x\ \rightarrow\ \mathit{CloseGap}(x)\ \Box\\&\mathit{movb}?y.x\ \rightarrow\ \mathit{CloseGap}(y)\ \Box\\&\mathit{mov}?y.x\ \rightarrow\ \mathit{CloseGap}(y);\end{aligned}$$

5-1 Defence The 5-1 defence is a simple extention of the same idea. The difference is that additional defensive zone is added in front of the 6-0 line.

Composition of Attack and Defence The attack and defence processes are composed using the ||| operator. This is a *PCSP#* operator that allows the composition of processes that communicate via channels. The syntax is as follows.

$$AD\ =\ CB\ |||\ \mathit{DEF}60;$$

The attack begins with the process *CB* and evolves as described in the process description. The defence process interleaves with the atttack process by synchronising on the channel events.

8 Application

We have applied the model to the analysis of handball matches from the European Handball Federation (EHF) Champions League and the IHF U-18 World Championships. The four matches cover sex, age and tier. The European Handball Federation (EHF) Champions League, established in 1956, is Europe's premier club competition in handball, encompassing both men's and women's tournaments. Hungarian clubs have been prominent in both competitions. Telekom Veszprém HC has reached the men's final four times, each time finishing as runner-up, while MOL-Pick Szeged is the reigning Hungarian Cup holder. In women's handball, Győri Audi ETO KC dominates with six Champions League trophies, and FTC-Rail Cargo Hungaria, current Hungarian champions and cup winners, have also achieved runner-up status in the competition. These high-stakes encounters not only fuel intense rivalries but also provide valuable data for analyzing tactical sequences and player decision-making.

Event logging Two analysts watched match in a media-player displaying and entered every attacking episode into spreadsheets. Each atomic or combination event was logged with time, zone and outcome.

Atomic actions (28)	– Ball control: receive, dribble, pick-up, fake-shot, fake-pass; – Passes: parallel, cross/X-over, long diagonal, return, pivot feed, bounce, wing; – Shots: wing, back-court, pivot, 6 m breakthrough, fast-break, lob, jump, standing; – Defence/keeper: block, steal/intercept, goalkeeper save, foul earned, 7m earned; – Outcomes: goal, miss, turnover, ball-out.
Multi-action combinations (18)	– Wing-Yugo, Yugo, Pivot cross, PvSlide, Rb/Lb -Pv 2-2, Lw/Rw -Rover, Cb-Rb/LbX, RunP, LongP, RetP, PvBlock, Lw/Rw 2ndPv, Ten Pv- Lw/RwX

We also recorded the occurences of passing and movement events and shooting locations and success rates. The results are provided in the appendix.

Probabilistic verification The data was hand-coded into the *PCSP#* model and simulated in the Process Analysis Toolkit (PAT). Four reachability queries were run for every team similar to the following examples. The first example is the probability of eventually scoring. The second example is the probability of eventually scoring given that the attack is a pivot cross (left or right). Due to the size of the model, the reachability queries were run to maximum depth of 5 passes. This is not considered a significant limitation due the fact that the number passes that actually impact the outcome of an attack is often less than 5.

$$\#assert\ AD\ reaches\ scoregoal\ with\ prob;$$
$$\#assert\ AD\ \models\ F(pvxl\ \ \|\ \ pvxr)\ \&\&\ X\ F(scoregoal)\ with\ prob;$$

Results Table 1 shows the results of the matches with the fast breaks excluded and a summary of the predicted number of goals and success rate. We have excluded the fast breaks from the analysis as they are not currently part of the model. (Note we covered gast breaks in the earlier paper.)

The maximum probability of the complex combinations are presented in Table 2.

Table 1. Analysis results (A = Actual, P = Predicted)

Tier	Competition	Sex	A (attacks/goals/success%)	P (goals/success%)
T1	EHF Champions League	♂	Szeged.A (46/23/50) Veszprém.A (57/22/39)	(19/42) (26/45)
T1	EHF Champions League	♂	Szeged.B (63/32/52) Veszprém.B (52/39/59)	(21/33) (17/33)
T2	EHF Champions League	♀	Győri ETO (40/21/53) FTC (39/15/38.5)	(16/40) (15/38)
T4	IHF U-18 Worlds	♀	Netherlands (46/17/37.0) China (42/8/19.0)	(12/26) (11/26)

Table 2. Success Rates by Play Pattern and Team (A = Actual, P = Predicted)

Play	SzA		VeB		Ch		Ne		FTC		ETO		SzB		VeB	
	A	P	A	P	A	P	A	P	A	P	A	P	A	P	A	P
P2	73	40	33	24	50	19	-	-	-	-	-	-	80	27	100	33
PX	43	75	0	26	20	21	60	20	50	29	-	-	100	38	100	35
EX	-	-	29	20	-	-	33	20	20	29	38	34	50	22	37	27
X	60	22	57	19	0	23	40	20	50	30	75	30	80	26	60	29

8.1 Discussion

Key findings Across four elite and youth matches the probabilistic model correctly identified the winning team in two cases and produced a mean absolute error of 6.8 percentage points in goal-probability prediction.

Interpreting discrepancies The youth mismatch (Netherlands vs China) exposes two boundaries of the current framework. First, player-level parameters were calibrated on senior data; youth error rates in passing and shooting are higher and more variable, amplifying divergence. Second, the model assumes a standard 6-0 or 5-1 defensive structure with fixed gap-closing probabilities. Video review shows that China defended with a high-pressure 6-0 variant (6-0C), stepping aggressively into first-line gaps and contesting most passes. Those pressure cues induce state transitions—especially forced turnovers—outside the calibrated range, inflating forecast error. These observations confirm that model fidelity may depend less on sample size than on defensive intensity and style representativeness.

The 2nd (EHF Champions League) match between Szeged and Veszprem predicts the opposite outcome to the actual result (Szeged 32, Veszprem 39). This is due to the fact that the model does not cover fast breaks. In this particular match, Szeged suffered an extrordinarly high number of turnovers (18) resulting in 16 fast breaks for Veszprem whereas there were only 2 fast break for

Szeged from 7 Veszprem turnovers. However the model does correctly predict the outcome of the non-fastbreak play.

The maximum probabilities of the complex combinations are not indicative of the actual success rate. Further investigation is required to understand the reasons for this.

Practical value for coaches Despite sparse data, the framework already yields actionable insights. PAT counter-examples reveal that Szeged's most profitable sequence was the Wing-to-Second-Pivot (success = 73 %), yet it accounted for only 20 % of attacks. Conversely, Veszprem's Pivot-Cross produced zero goals in seven attempts; reallocating those possessions to X-cross would have yielded an estimated +0.9 goals per game. Such diagnostics let coaches target training time without exhaustive video coding.

Limitations The manual analysis of each game takes significant time and is error prone, and as such we were only able to analyse a small number of matches. As such the Data is confined to 6 teams and and only a small number of atomic labels; rare tactics (e.g., 7 v 6 empty-goal) are absent. Because we did not retain a systematic double-coded sample, we cannot report a formal inter-rater statistic (e.g., Cohen's k); quantifying annotation uncertainty is therefore left to future work. Defensive reactions are currently modelled as static zone gaps; dynamic elements such as stepped-out first-line pressure and adaptive goalkeeper positioning remain abstracted away. The present implementation is also episodic and ignores fatigue, score effects, and stochastic time-outs. Davis *et al* [2] discuss these and other limitations of sports analytics models.

9 Further Work and Conclusion

The details presented above define a framework for the consideration of playing conditions; description of decision options (as probabilistic choices); and the description of some performance characteristics such as passing and shooting (also probabilistic).

While our initial results are promising, there are several key areas for future development of this work. A primary limitation is the relatively small dataset currently available for analysis. The model would also benefit from including goalkeeper success rates to better reflect defensive capabilities. Player fatigue effects on performance metrics should be considered, as these can significantly impact game outcomes over time. Additionally, analyzing defensive adjustment patterns across multiple games would provide insight into tactical adaptations, especially in the case of superiority or inferiority due to exclusions. The impact of timeouts and substitutions on attack effectiveness represents another important avenue of investigation. These extensions would create a more comprehensive framework for analyzing handball tactics and performance.

References

1. G. Bonnet et al. Toward a better understanding of decision-making in handball: A systematic review. *Movement & Sport Sciences*, (108):5–20, 2020.
2. Jesse Davis, Lotte Bransen, Laurens Devos, Wannes Meert, Pieter Robberechts, Jan Van Haaren, and Maaike Van Roy. Evaluating sports analytics models: Challenges, approaches, and lessons learned, 2022-07-18.
3. S. Espoz-Lazo and C. Hinojosa-Torres. Modern handball as a dynamic system: Orderly chaotic. *Applied Sciences*, 15(7):3541, 2025.
4. Willian Ferrari, Hugo Sarmento, and Vasco Vaz. Match analysis in handball: A systematic review. *Montenegrin Journal of Sports Science and Medicine*, 8:63–76, 09 2019.
5. J. Fontaine. The role of emotions in intuitive decision-making: A study with expert handball coaches. In *Naturalistic Decision-Making Conference Papers*, 2024.
6. M. Hinz et al. Differences in decision-making behavior between elite and amateur team-handball players in a near-game test situation. *Frontiers in Psychology*, 13:854208, 2022.
7. Shiro Ichimura, Tetsushi Moriguchi, and Roland Nemes. Development of data model for attacking behavior in handball. In *7th European Handball Federation Scientific Conference "Sustainability in Handball – Circle of a Handball Life"*, Porto, May 2024. Conference Coordinators: João Monteiro, Beata Kozlowska, Noémi Szécsényi.
8. IHF. International handball federation, 2023. Accessed on August 6, 2023.
9. Jiang Kan. Sports strategy analytics using probabilistic model checking and machine learning, 2023.
10. M. Kwiatkowska, G. Norman, and D. Parker. PRISM: Probabilistic symbolic model checker. In *Proc. Tools Session of Aachen 2001 International Multiconference on Measurement, Modelling and Evaluation of Computer-Communication Systems*, 2001.
11. M. Le Menn et al. Handball goalkeeper intuitive decision-making: A naturalistic case study. *Journal of Human Kinetics*, 70:249–260, 2019.
12. B. Lenzen et al. Situated analysis of team handball players' decisions: An exploratory study. *Journal of Teaching in Physical Education*, 28(3):327–343, 2009.
13. Zhaoyu Liu, Kan Jiang, Zhe Hou, Yun Lin, and Jin Song Dong. Insight Analysis for Tennis Strategy and Tactics . In *2023 IEEE International Conference on Data Mining (ICDM)*, pages 1169–1174, Los Alamitos, CA, USA, December 2023. IEEE Computer Society.
14. Zhaoyu Liu, Murong Ma, Kan Jiang, Jin Dong, Zhe Hou, and Ling Shi. *PCSP# Denotational Semantics with an Application in Sports Analytics*, volume 14900, pages 71–102. Springer Nature Switzerland, Cham, 09 2024.
15. L. Magnaguagno et al. Cognitive representations of handball tactic actions in athletes: The function of expertise and age. *PLOS One*, 18(4):e0284941, 2023.
16. Zoltan Marczinka. *Playing handball: A comprehensive study of the game.* Trio Budapest, 1993.
17. Zoltan Marczinka. Analysis of technical elements in defence: Focusing on the differences between positions and genders. In *7th European Handball Federation Scientific Conference "Sustainability in Handball – Circle of a Handball Life"*, Porto, May 2024. Conference Coordinators: João Monteiro, Beata Kozlowska, Noémi Szécsényi.

18. Hiromu Mizuno and Roland Nemes. Examination of scoring efficiency in men's handball using random forest. In *7th European Handball Federation Scientific Conference "Sustainability in Handball – Circle of a Handball Life"*, Porto, May 2024. Conference Coordinators: João Monteiro, Beata Kozlowska, Noémi Szécsényi.
19. S. Nicolosi et al. Situational analysis and tactical decision-making in elite handball players. *Applied Sciences*, 13(15):8920, 2023.
20. J. Perl, Markus Tilp, Arnold Baca, and Daniel Memmert. Neural networks for analysing sports games. *Routledge Handbook of Sports Performance Analysis*, pages 237–247, 01 2013.
21. Paul Rudelsdorfer, Norbert Schrapf, Horst Possegger, Thomas Mauthner, Horst Bischof, and Markus Tilp. A novel method for the analysis of sequential actions in team handball. *International Journal of Computer Science in Sport (IJCSS)*, 13:69–84, 01 2014.
22. Norbert Schrapf and Marcus Tilp. Action sequence analysis in team handball. *Journal of Human Sport and Exercise*, 8(3proc):S615–S621, Dec. 2013.
23. Songzheng Song, Lin Gui, Jun Sun, Yang Liu, and Jin Song Dong. Improved reachability analysis in dtmc via divide and conquer. In *IFM*, pages 162–176, 2013.
24. Jun Sun, Yang Liu, Jin Song Dong, and Jun Pang. Pat: Towards flexible verification under fairness. volume 5643 of *Lecture Notes in Computer Science*, pages 709–714. Springer, 2009.
25. Nilgün Vurgun, Murat Bilge, Serdar Eler, Nebahat Eler, and Aydın Şentürk. Current developments in handball game analysis. *Journal of Population Therapeutics and Clinical Pharmacology*, 30(12):421–435, 05 2023.
26. Luke Wildman. Probabilistic model checking of handball action sequences. In *2023 IEEE 28th Pacific Rim International Symposium on Dependable Computing (PRDC)*, pages 332–336, 2023.