HoopEval: Individual Player Action Evaluation via Deep Reinforcement Learning

Basketball
62

1. Introduction

Evaluating and interpreting players' real-time performance and decision-making on the court requires a deep understanding of context-dependent actions and multi-agent coordination. Limited by the availability of data, prior approaches have primarily relied on the summarization of outcome-based metrics such as points scored, plus-minus ratings, or some all-in-one metrics to assess the technical and tactical capabilities of players from a macro-level perspective. The emergence of high-resolution tracking data fundamentally changes this landscape, making it possible to analyze the fine-grained, moment-to-moment decision dynamics that were previously invisible to traditional performance metrics. One of the most influential developments is the Expected Possession Value (EPV) framework [1]. EPV represents a breakthrough in basketball analytics: instead of evaluating a possession solely by its outcome, EPV treats a possession as a Markov Decision Process (MDP), quantizing the expected value of a possession at every moment in time, based on the spatial configuration and coordinated actions of all players on the court. Methodologically, EPV accomplishes *credit (value) assignment* on the temporal dimension through forward modeling.

Although EPV marks a foundational breakthrough beyond traditional outcome-based metrics, it remains inherently a team-level metric and reveals the team's overall value evolution over time. This implies that every offense player shares the same value at each moment without explicitly distinguishing individual contributions to actions taken at that instant. As a result, when EPV is applied to evaluate an individual player, the assigned value simply reflects the team's collective performance rather than the player's own contribution, making individual impact impossible to isolate. This is analogous to evaluating a player with game-level plus-minus [2], a measure that conflates individual impact with team performance and cannot disentangle the effects of teammates. This issue becomes even more pronounced when evaluating off-ball action, as the impact of off-ball actions often manifests with a delay and is therefore not immediately reflected in the team-level value assigned by EPV. To overcome this issue, we propose applying a second stage of assignment to EPV at the player dimension, so that the possession value can be decomposed across individual players instead of remaining at the team level.

However, the *forward-value-prediction* paradigm on which EPV is built fundamentally limits its ability to support player-level value decomposition. This is because EPV predicts forward to estimate possession value, rather than propagating outcome backward through state–action transitions, it lacks a mechanism for assigning credit to individual players' actions. In contrast, reinforcement learning (RL) provides a more principled foundation for

this task through its core mechanism of *backward-value-propagation*. Specifically, by propagating value backward through state-action sequences, the RL framework explicitly attributes change in expected value to the specific actions of individual agents. This backward credit-assignment mechanism thereby quantifies the marginal contribution of each player at every decision point—a capability that is structurally beyond the reach of the model like EPV.

In this paper, we propose HoopEval, a RL framework that addresses EPV's practical limitations in player-level value allocation, enabling fine-grained credit assignment and decision evaluation within multi-agent basketball possessions. Specifically, we adopt an offline reinforcement learning framework based on autoregressive Q-function [3], enabling both temporal dimension and player dimension credit assignment by sequentially propagating value through the ordered actions of the ball and the five offensive players. Based on the RL model, we estimate the value of all possible actions within the possession, including ball actions (pass, dribble, shot) and the spatial movement actions of the five offensive players.

2. Preliminaries

This section reviews the background most relevant to our work. Section 2.1 introduces EPV-based models widely used in sports analytics. Section 2.2 discusses multi-agent reinforcement learning in sports, which provides a conceptual foundation for modeling coordinated decision-making among multiple players.

2.1. EPV-based models

Expected Possession Value (EPV) models estimate the expected outcome of a possession from the current game state by forecasting future action sequences, such as passes, carries, shots, or other sport-specific events, and aggregating their expected returns.

In basketball, beyond the EPV framework introduced above, several other EPV-like models have been proposed to analyze player decision-making, quantify spatial value, and evaluate possession dynamics. Sicilia et al. [4] proposed DeepHoops which employs deep neural networks to encode past spatio-temporal trajectories and explicitly predict the probabilities of terminal events within a short future window, thereby enabling fine-grained evaluation of micro-actions and their impact on possession outcomes. Unlike EPV, which relies on a hand-crafted stochastic process model, DeepHoops offers an end-to-end trainable framework that implicitly captures the dynamics of player behavior through data-driven feature representations. In addition, Yanai et al. [5] introduced Q-ball, a deep reinforcement learning model that treats ball possession as a continuous decision-making process, combining deep neural networks to encode historical spatio-temporal context with a Q-learning framework that explicitly links state-action pairs to their expected long-term returns. To translate team-level value into player-specific insights, Q-ball incorporates a Shapley-value–based attribution mechanism that decomposes the learned possession value into individual players' marginal contributions.

However, we hold reservations about the use of Shapley values in RL framework, as they explain only a player's average contribution to model predictions rather than the logic of any specific decision made within a possession [6].

The EPV framework has also been extended to the continuous spatial dynamics of soccer. EPV's original authors Luke Bornn and Daniel Cervone, together with Fernández [7] adapted the original EPV methodology to soccer, proposing a fine-grained model that estimates the instantaneous expected value of a possession using spatio-temporal tracking data. Additionally, several EPV-like models have been developed in soccer, including xT [8], VAEP [9], and OBV [10], all of which assess the value of actions or ball progressions through their expected influence on subsequent possession outcomes, as well as other related variants. Recently, several studies have explored reinforcement learning frameworks to model decision-making and learn action values directly from soccer tracking data, offering an alternative to forward-predictive possession-value models [11,12, 13]. However, these methods still lack the capability to quantify decision value at the level of individual players.

2.2. Multi-agent reinforcement learning

Recent advancements in multi-agent reinforcement learning (MARL) have demonstrated its application potential in modern games. Beginning with foundational work and progressing to landmark achievements such as AlphaStar [14] in StarCraft II and OpenAI Five [15] in Dota 2, MARL has proven capable of achieving superhuman performance across diverse game environments through techniques like self-play, supervised learning, and deep reinforcement learning. Among these, the one most closely aligned with real-world sports dynamics is the Google Research Football environment [16], which provides a realistic, physics-based simulation of soccer and supports multi-agent coordination, competitive play, and policy learning through reinforcement learning. This environment has become a widely used benchmark for studying multi-agent decision-making, teamwork, and strategy formation. Research in this domain has explored value decomposition architectures such as VDN [17] and QMIX [18], as well as communication and cooperation mechanisms designed to improve real-time coordination among agents. While these works provide valuable insights into multi-agent credit assignment and cooperative decision-making, they remain fundamentally limited by the constraints of an online simulated environment. As a result, many of the learned strategies exhibit a substantial gap from real-world sports behaviors and consequently cannot serve as reliable critics for evaluating the quality of actions observed in real-world tracking data. Empirically, reliable action evaluation requires learning effective policies or critics directly from offline datasets derived from real-world spatio-temporal player and ball tracking systems.

3. Methodology

3.1. Dataset

The dataset we used was collected by SportVU from 632 games in the 2015–2016 season of the NBA, which provides real-time positional data of the ball and players at 25 frames per second. According to basketball rules, the time sequence data for each game can be naturally segmented by offensive ball possession, and the time length of each offensive ball

possession is no more than 24 seconds. Furthermore, we used play-by-play data to trim each possession, ensuring that the last frame of every time sequence corresponds to the shot attempt. At the same time, we inferred the state of the ball, dribbled or passed, by calculating the distances between the ball and the players. To improve modeling efficiency, the tracking data were downsampled to 5 frames per second. Each offensive possession was modeled as a discrete-time Markov Decision Process (MDP), formally represented as a tuple {S, A_i, R, S'}, where *S* represents the set of states, A_i represents the set of actions of each offense players *i*, *R* represents the reward function formulating the reward, *S'* represents the set of next states.

3.1.1. State space

We discretized the basketball court into a hexagonal grid and used the grid cell indices of each player and the ball as node features in a graph-based state representation. In this graph structure, each player and the ball are represented as nodes, with their player embedding and spatial positions given by the corresponding grid indices. Since a transformer-based model is employed for state encoding, temporal features such as player movement speed and direction are implicitly captured by the model itself, without the need for explicit hand-crafted features. In addition, the graphs are fully connected; that is, for every pair of players, we will include the edge connecting them in the graph. Each of these edges encodes three binary features that indicates (1) whether the two players are on opposing teams or not; (2) who is the holding the ball. (3) who can be the pass receiver. We further include the rim as an additional node, which provides a stable spatial reference point that helps the model better capture offensive intent and positional relationships relative to the scoring target.

3.1.2. Action space

The action space in our model comprises two components. The first component involves on-ball action, specifically six bins: shooting or passing the ball to any of the five offensive players. It is worth noting that we treat the action of "dribbling" as a special case of passing, where the ball-handler is considered to pass the ball to himself. For example, if player 1 is the ball handler in the current state and performs a dribble during the possession, this is represented as choosing to pass the ball to himself in the state transition. The second component concerns player movement, which is defined over the hexagonally discretized state space. Specifically, the movement of each player is discretized into 19 possible actions,

corresponding to the nearest cells of the grid surrounding their current position.

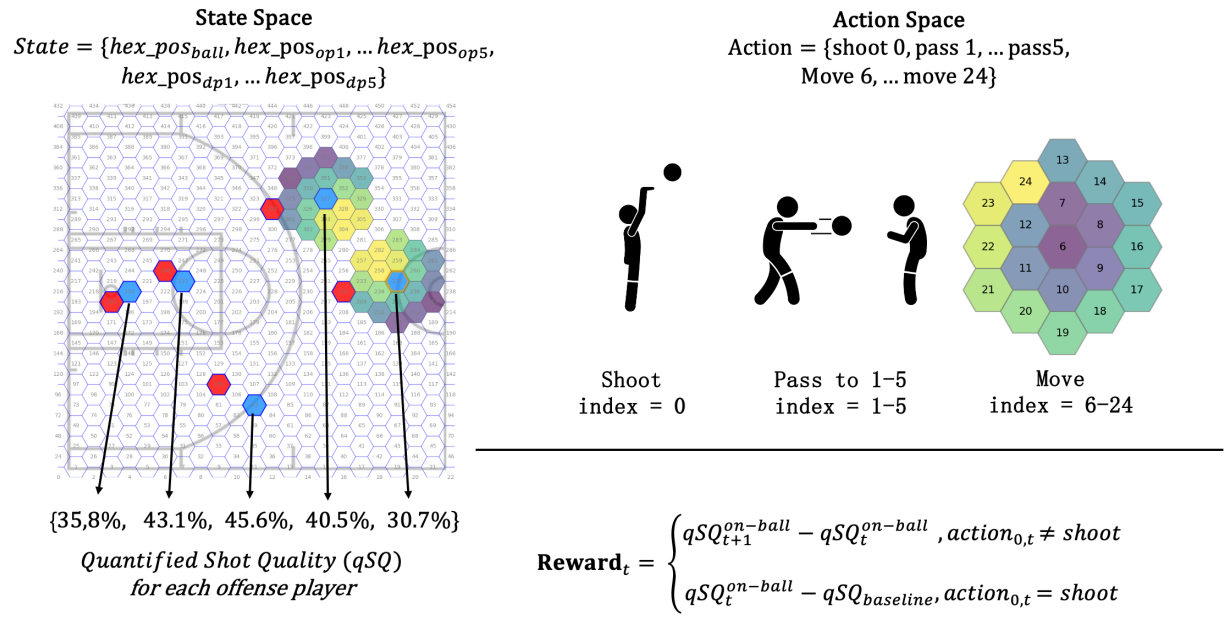


Figure 1 Illustration of the state space, action space, and reward model. The state space was represented using a hexagonal discretization of the court, where the positions of each player and the ball are encoded as node features in a fully connected graph, along with player embeddings and edge features that capture team assignment and ball possession status. The action space included ball-related actions (shooting, passing, and dribbling as self-pass) as well as discretized movement actions to the 19 nearest grid cells for each player. The reward function is defined as the change in the ball handler's qSQ value, computed as the difference between the qSQ at time t+1and time t when the action is not a shot. For shooting actions, the reward is given by the ball handler's qSQ minus a predefined baseline value (0.40).

3.1.3. Reward model

The most straightforward reward signal in basketball possessions is whether the possession outcome is a score. However, we believe this reward is too sparse to effectively guide agents learning optimal action. Thus, it is necessary to handcraft a denser reward function by leveraging domain knowledge, in order to provide denser feedback and encourage behaviors relevant to successful performance. For offensive players, the fundamental purpose of tactical behaviors is to generate the easiest and highest-quality shot opportunities. One such domain-specific metric is Quantified Shot Quality (qSQ) [19], which provides a continuous measure of the quality of shot opportunities available to each offensive player at every time step. By calculating the qSQ at each timeframe, we obtain a dense, temporally aligned reward signal that reflects the incremental value of each action in contributing to high-quality scoring opportunities. For non-shooting actions, the reward is defined as the change in the ball handler's qSQ between consecutive timesteps, capturing how passes, dribbles, and movements improve offensive positioning. For shot attempts, the reward is given by the ball handler's qSQ relative to a predefined baseline, allowing the model to distinguish between high- and low-quality shots.

3.2. Model architecture

Our model employs an encoder–decoder architecture to model basketball possessions and fully leverages the advantages of the spatiotemporal attention mechanism, which has been widely adopted in human–motion analysis, and multi-agent trajectory modeling for its ability to capture complex spatial interactions and their temporal evolution [20].

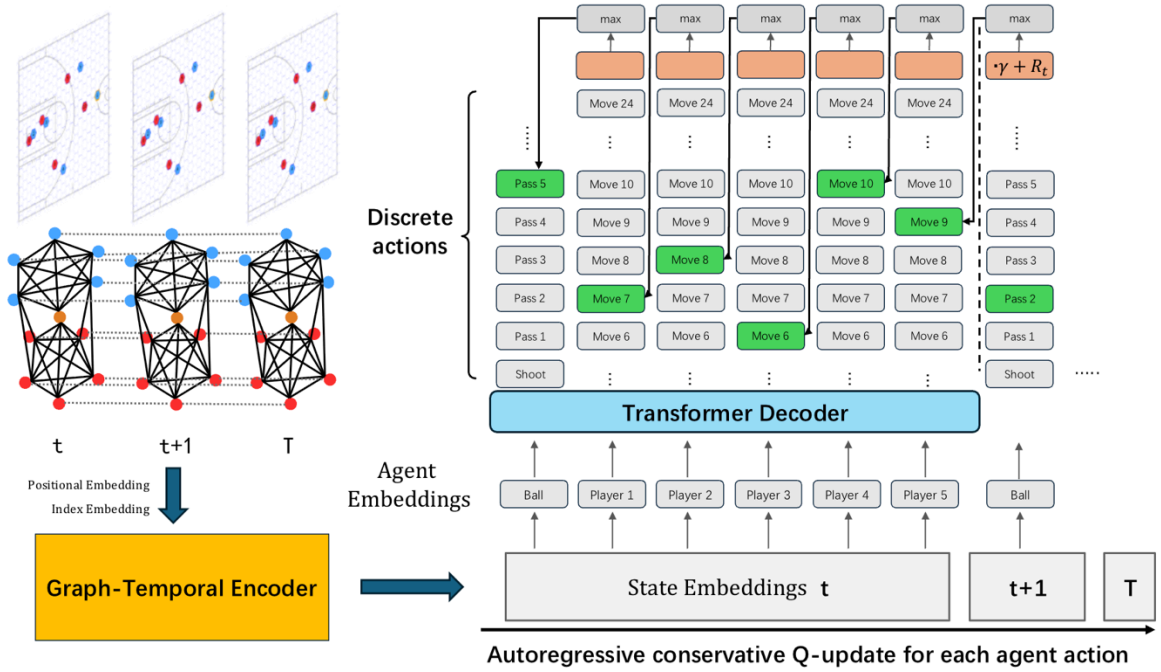


Figure 2 Overview of the model framework. Given a sequence of states in a possession, each state is encoded as a graph including the ball and five players. For each state, the model autoregressively predicts the Q-values for the actions of the ball and all players, one by one. The Q-values of the discrete action bins corresponding to dataset actions (green boxes) are trained via the Bellman update. The Q-values of unobserved action bins (gray boxes) are minimized toward zero to enforce conservativeness. The Q-targets for all agent dimensions except the last are computed using the maximum Q-value across the next agent's actions at the same timestep (pink boxes). The Q-target of the last agent is computed as the discounted maximum Q-value of the first agent in the next state plus the current reward. We also incorporate Monte Carlo returns by taking the maximum of the computed Q-targets and the return-to-go.

3.2.1. Graph-based Spatial-temporal Encoder

In this work, we utilize a Graph Attention Network to encode spatial information, capturing the dynamic interactions among all players and the ball within each frame [21]. This graph-based encoder treats players and the ball as nodes and models their relationships as edges, effectively representing both team structure and contextual positioning. Recent research, most notably the TacticAI study [22], has demonstrated that graph attention neural networks are particularly well-suited for modeling multi-agent interactions in sports tracking data, especially in domains such as football and basketball where the spatial dependencies between agents are complex and highly dynamic. To capture the temporal dependencies and causal structure inherent in sequential decision-making, we adopt a causal self-attention mechanism within the transformer architecture. This allows the model

to attend to relevant historical states and actions in an autoregressive manner, ensuring that each prediction is conditioned only on past information and thus preserving the correct temporal order.

3.2.2. Autoregressive Q-function Decoder

The autoregressive Q-function decoder is the core module responsible for sequential value assignment to each agent's action in a basketball possession. At every timestep, the decoder receives the encoded state representation, which integrates the spatial and temporal context for all agents on the court. It then predicts Q-value distributions for the discrete action space of each agent, processing agents in a fixed order (for example, starting with the ball action, then proceeding through the action of each player). Let $\tau = (s_1, a_1, \cdots, s_T, a_T)$ be a trajectory of basketball experience of length T from an offline dataset D. For a given timestep t and the corresponding joint action a_t, we define a per-agent view of the action at time t. Let $a_t^{1:i}$ denote the vector of actions from the first agent up to the $i-th$ agent at time t, where i can range from 1 to the total number of agents, denoted as d_{A}. We define the Q-value of the $i-th$ agent's action a_t^i using an autoregressive Q-function conditioned on the previous state sequence and the previous agents' actions $a_t^{1:i-1}$ at the same time step. To train the Q-function, we define a per-agent Bellman update. For all agent dimensions $i \in 1, \cdots, d_{\mathrm{A}}$:

$$Q(s_{t-w:t}, a_t^{1:i-1}, a_t^i) \leftarrow \begin{cases} \max_{a_t^{i+1}} Q(s_{t-w:t}, a_t^{1:i}, a_t^{i+1}) & \text{if } i \in \{1, \ldots, d_{\mathcal{A}} - 1\} \\ R(s_t, a_t) + \gamma \max_{a_{t+1}^1} Q(s_{t-w+1:t+1}, a_{t+1}^1) & \text{if } i = d_{\mathcal{A}} \end{cases}$$

For each agent, the decoder outputs a vector of Q-values corresponding to all possible actions, such as passing, shooting, or moving to neighboring locations. Each agent's Q-value prediction is conditioned on the current state embedding and the actions already selected by previous agents in the sequence. This autoregressive design allows the model to capture dependencies and coordinated behaviors among multiple agents, which are essential for representing complex basketball tactics.

During training process, Q-values for actions observed in the dataset are updated using the Bellman equation, while penalizing the Q-values of actions not performed to ensure the conservatism of the learning objective. For all agents except the last in the sequence, Q-targets are computed as the maximum Q-value over the next agent's action dimension at the same timestep, effectively propagating value through the agent sequence. For the last agent, the target is computed as the discounted maximum Q-value of the first agent at the next timestep plus the immediate reward. To further improve stability, the model incorporates Monte Carlo returns by taking the maximum between the computed Q-targets

and the empirical return-to-go.

3.3. Training Objective

Offline reinforcement learning is prone to Q-value overestimation for actions that are rarely or never observed in the dataset due to distributional shift. To address this issue, we incorporate Conservative Q-Learning (CQL) to explicitly penalize the Q-values of out-of-distribution (OOD) actions and improve the robustness of value estimation. Based on the empirical reward distribution in our basketball dataset, we set a conservative lower bound for OOD Q-values and introduce action-dependent penalty weights.

To further stabilize training and align learned representations with basketball-specific decision semantics, we introduce two auxiliary alignment tasks that are optimized jointly with the Q-learning objective. The first task predicts the qSQ values of all five offensive players from the encoder output, encouraging the model to capture spatial and contextual cues relevant to offensive value. The second task is a supervised ball-action classification objective under teacher forcing, which distinguishes between pass and dribble actions and regularizes the structure of the ball-action space.

The overall training objective is defined as a weighted combination of the temporal-difference loss for in-dataset actions, the conservative regularization term for OOD actions, and the auxiliary alignment losses. Joint optimization of these components yields more stable training and results in Q-values that are both value-consistent and interpretable at the player and action levels. Formally, the total loss is defined as:

$$
\begin{aligned}
J = & \underbrace{\mathbb{E}_{s\sim\mathcal{D},\, a\sim\pi_\beta(a|s)}\left[Q(s,a) - \mathcal{B}^* Q^k(s,a)\right]^2}_{\text{(i) TD loss}} \\
& + \alpha \cdot \underbrace{\mathbb{E}_{s\sim\mathcal{D},\, a'\sim\nu(\cdot|s)}\left[w(a') \cdot (Q(s,a') - \text{Mini\_reward})^2\right]}_{\text{(ii) Conservative regularization } \mathcal{L}_{\text{CQL}}} \\
& + \beta \cdot \underbrace{\mathbb{E}_t\left[\frac{1}{5}\sum_{i=1}^{5}\left(\hat{q}_i^t - q_i^t\right)^2\right]}_{\text{(iii) qSQ prediction loss } \mathcal{L}_{\text{qSQ}}} \\
& + \gamma \cdot \underbrace{\mathbb{E}_t\left[-y_t \log \hat{p}_t - (1-y_t)\log(1-\hat{p}_t)\right]}_{\text{(iv) Ball action CE loss } \mathcal{L}_{\text{ball}}}
\end{aligned}
$$

4. Result

Figure 3 illustrates the temporal evolution of the estimated team value over a single offensive possession, along with the corresponding player spatial positions at frame 1, frame 20, frame 36, and frame 58.

As shown in frame 1, the middle offensive player has the ball at the top of the penalty arc. One teammate is positioned at the right bottom corner, two teammates are clustered around the left bottom corner, and another teammate is positioned on the left sideline. When considering only the ball handler's qSQ, this moment would be regarded as a favorable open-shot opportunity. However, our model assigns a relatively lower team value

to this state because the ball-handler at this point is Rudy Gobert, whose career three-point shooting percentage is zero. As the possession progresses to Frame 20, the estimated team value reaches its lowest point over the entire possession. At this moment, although the offense remains in control of the ball, defensive players are well-aligned to restrict driving and passing options, resulting in a configuration that is tactically unfavorable at the team level. Consequently, the model assigns the minimum team value to this state, reflecting a momentary breakdown in collective offensive advantage.

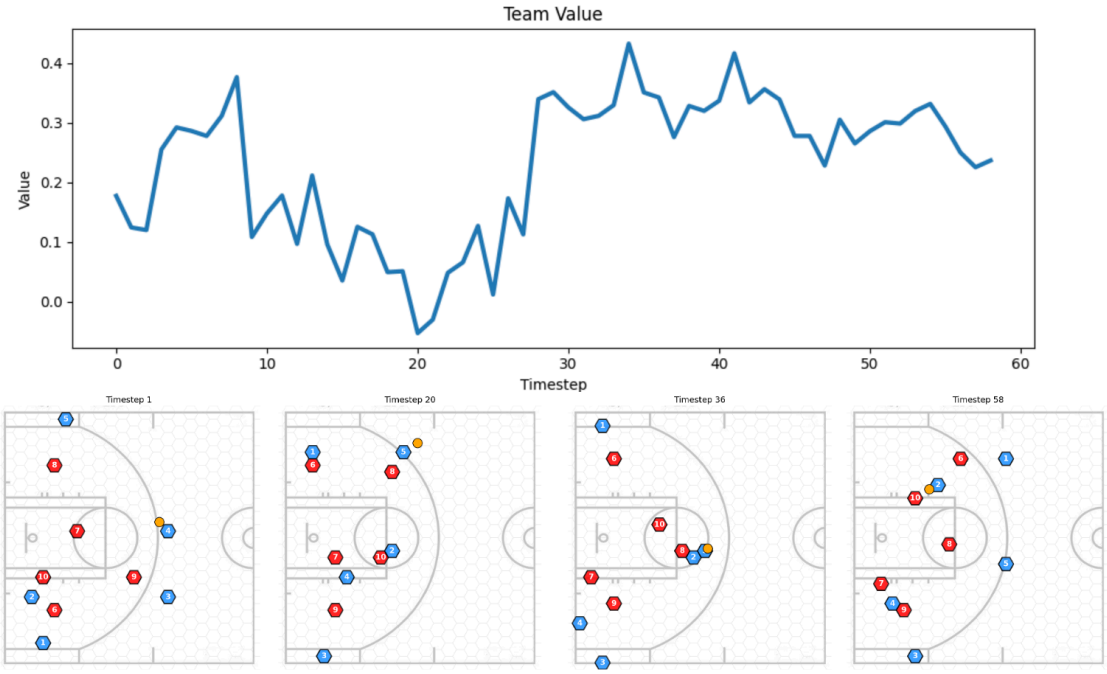
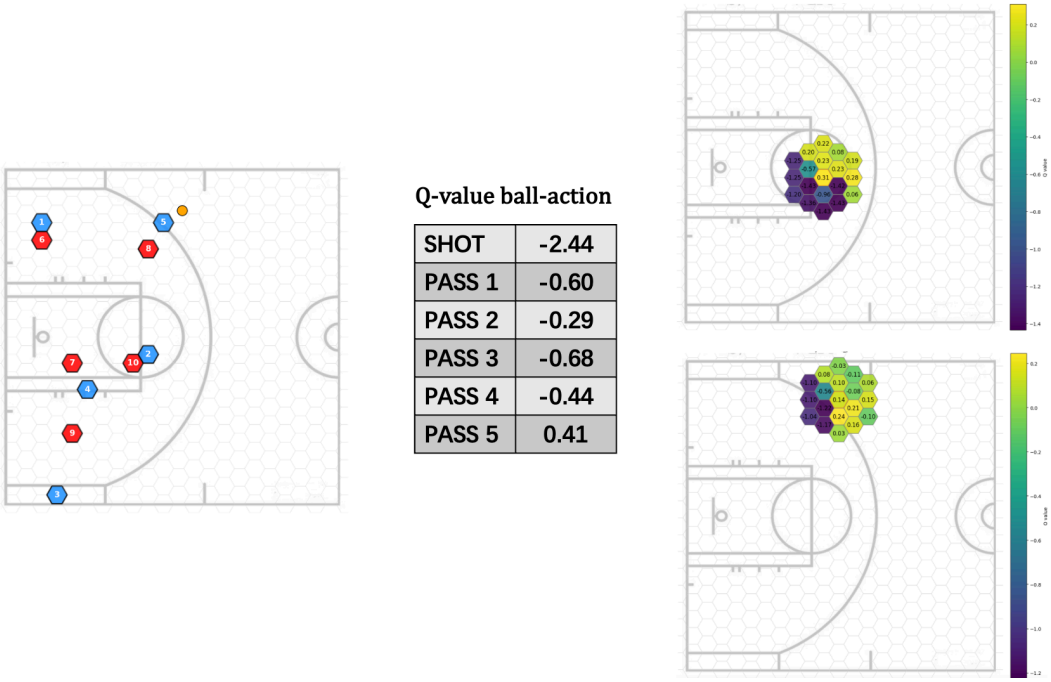


Figure 3 Temporal evolution of estimated team value and corresponding spatial configurations. The top panel shows the estimated team value across all timesteps within the possession. The team value reaches its minimum at timestep 20, indicating the least favorable offensive state as evaluated by the model. The bottom panel visualizes the spatial configurations of all players and the ball at four representative timesteps (1, 20, 36, and 58) using the hexagonal court discretization.

Between Frames 20 and 36, the offensive team initiates a middle pick-and-roll action. The ball handler uses the screen set near the middle of the court to probe the defense, forcing defensive rotations and momentarily disrupting on-ball containment. This coordinated two-player action improves spacing and creates downstream advantages for off-ball players, which is reflected in a steady increase in the estimated team value during this interval. It's worth noting that although the ball handler's instantaneous qSQ value in frame 36 was still relatively low, the model assigned the highest value to this moment, indicating that model predicts this pick-and-roll play will generate greater value in subsequent moments.

In addition, our model outputs action-level Q-values for each agent at Frame 20. At this moment, the model assigns relatively low Q-values to the available pass options for the ball handler, indicating the absence of favorable passing opportunities. Similarly, the predicted

Q-value for a shot is also low, reflecting the high difficulty of an immediate shooting attempt. As a result, the model suggests that the ball handler should continue dribbling to maintain possession and facilitate the development of a more advantageous offensive configuration. Moreover, the model indicates that the ball handler (Player 2) and an off-ball teammate (Player 5) should move toward each other, as this movement pattern is likely to initiate a pick-and-roll action. Such coordination is expected to improve spatial structure and create advantageous conditions for subsequent offensive decisions.



Q-value ball-action

SHOT	-2.44
PASS 1	-0.60
PASS 2	-0.29
PASS 3	-0.68
PASS 4	-0.44
PASS 5	0.41

Figure 4 presents an action-level interpretation at Frame 20. While the predicted Q-value for an immediate shot is the lowest among all ball actions and passing options are generally unfavorable, the model favors continuing to dribble to maintain possession. The player movement action analysis further shows higher Q-values for coordinated movement between the ball handler (Player 2) and an off-ball teammate (Player 5), indicating that the model anticipates a pick-and-roll action that is expected to create more advantageous offensive states in subsequent moments.

5. Conclusion

This work addresses a fundamental limitation of existing possession-value frameworks in basketball analytics: the inability to attribute value to individual players' decisions within a possession. While EPV and related models represent a major advance over outcome-based metrics by modeling the temporal evolution of team-level value, their forward-predictive formulation constrains them to collective evaluation and prevents principled player-level credit assignment, particularly for off-ball actions whose effects emerge with delay.

To overcome this limitation, we propose HoopEval**,** an offline reinforcement learning framework based on an autoregressive Q-function that enables fine-grained decision evaluation in multi-agent basketball possessions. By propagating value backward through

ordered state–action transitions, HoopEval explicitly quantifies the marginal contribution of each agent's action—both on-ball and off-ball—at every moment in time. In contrast to forward modeling methods, this approach integrates credit assignment directly into the learning objective and produces action-level value estimates that are temporally grounded and decision-aware.

Beyond methodological novelty, our results demonstrate that the learned Q-values align with interpretable basketball tactics, capturing how coordinated actions such as spacing adjustments and pick-and-roll interactions create future offensive advantage even when instantaneous shot-quality indicators remain low. This illustrates the potential of reinforcement-learning–based critics to serve not only as evaluative tools, but also as explanatory models for understanding decision dynamics in real games.

More broadly, this work highlights the necessity of learning value functions directly from offline, real-world tracking data in order to reliably evaluate player decisions. By bridging the gap between EPV-style possession modeling and agent-level decision analysis, HoopEval provides a principled foundation for next-generation performance evaluation systems and opens new avenues for interpreting multi-agent decision making in team sports.

References

[1] Cervone, D., D'Amour, A., Bornn, L., & Goldsberry, K. (2016). A multiresolution stochastic process model for predicting basketball possession outcomes. Journal of the American Statistical Association, 111(514), 585-599.

[2] Grassetti, L., Bellio, R., Di Gaspero, L., Fonseca, G., & Vidoni, P. (2021). An extended regularized adjusted plus-minus analysis for lineup management in basketball using play-by-play data. IMA Journal of Management Mathematics, 32(4), 385-409.

[3] Chebotar, Y., Vuong, Q., Hausman, K., Xia, F., Lu, Y., Irpan, A., ... & Levine, S. (2023, December). Q-transformer: Scalable offline reinforcement learning via autoregressive q-functions. In Conference on Robot Learning (pp. 3909-3928). PMLR.

[4] Sicilia, A., Pelechrinis, K., & Goldsberry, K. (2019, July). Deephoops: Evaluating micro-actions in basketball using deep feature representations of spatio-temporal data. In Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining (pp. 2096-2104).

[5] Yanai, C., Solomon, A., Katz, G., Shapira, B., & Rokach, L. (2022, June). Q-ball: Modeling basketball games using deep reinforcement learning. In Proceedings of the AAAI conference on artificial intelligence (Vol. 36, No. 8, pp. 8806-8813).

[6] Heuillet, A., Couthouis, F., & Díaz-Rodríguez, N. (2022). Collective explainable AI: Explaining cooperative strategies and agent contribution in multiagent reinforcement learning with shapley values. IEEE Computational Intelligence Magazine, 17(1), 59-71.

[7] Fernández, J., Bornn, L., & Cervone, D. (2021). A framework for the fine-grained evaluation of the instantaneous expected value of soccer possessions. Machine Learning, 110(6), 1389-1427.

[8] Introducing expected threat. https://karun. in/blog/expected-threat.html. Accessed: 2019-06-21.

[9] Decroos, T., Bransen, L., Van Haaren, J., Davis, J., & Bessiere, C. (2020, January). VAEP: An objective approach to valuing on-the-ball actions in soccer. In Proceedings of the twenty-ninth international joint conference on artificial intelligence, IJCAI-20 (pp. 4696-4700). International Joint Conferences on Artificial Intelligence Organization.
[10] StatsBomb: Introducing on-ball value (OBV). https://statsbomb.com/articles/soccer/introducing-on-ball-value-obv/ (2021).
[11] Nakahara, H., Tsutsui, K., Takeda, K., & Fujii, K. (2023). Action valuation of on-and off-ball soccer players based on multi-agent deep reinforcement learning. IEEE Access, 11, 131237-131244.
[12] Pulis, M., & Bajada, J. (2022). Reinforcement learning for football player decision making analysis.
[13] Rahimian, P., Van Haaren, J., Abzhanova, T., & Toka, L. (2022). Beyond action valuation: A deep reinforcement learning framework for optimizing player decisions in soccer. In 16th MIT sloan sports analytics conference (Vol. 3).
[14] Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., ... & Silver, D. (2019). Grandmaster level in StarCraft II using multi-agent reinforcement learning. nature, 575(7782), 350-354.
[15] OpenAI et al., "Dota 2 with large scale deep reinforcement learning," arXiv:1912.06680 [cs, stat], Dec. 2019.
[16] Kurach, K., Raichuk, A., Stańczyk, P., Zając, M., Bachem, O., Espeholt, L., ... & Gelly, S. (2020, April). Google research football: A novel reinforcement learning environment. In Proceedings of the AAAI conference on artificial intelligence (Vol. 34, No. 04, pp. 4501-4510).
[17] P. Sunehag et al., "Value-decomposition networks for cooperative multiagent learning," in Proc. 17th Int. Conf. Auto. Agents MultiAgent Syst., Stockholm, Sweden, Jul. 2018, pp. 2085–2087.
[18] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. N. Foerster, and S. Whiteson, "Monotonic value function factorisation for deep multiagent reinforcement learning," J. Mach. Learn. Res., vol. 21, pp. 1–51, Aug. 2020.
[19] Chang, Y. H., Maheswaran, R., Su, J., Kwok, S., Levy, T., Wexler, A., & Squire, K. (2014, February). Quantifying shot quality in the NBA. In Proceedings of the 8th Annual MIT Sloan Sports Analytics Conference. MIT, Boston, MA.
[20] Wang, X., Tang, Z., Shao, J., Robertson, S., Gómez, M. Á., & Zhang, S. (2024). Hooptransformer: advancing NBA offensive play recognition with self-supervised learning from player trajectories. Sports Medicine, 54(10), 2663-2673.
[21] Brody, S., Alon, U., & Yahav, E. (2021). How attentive are graph attention networks?. arXiv preprint arXiv:2105.14491.
[22] Wang, Z., Veličković, P., Hennes, D., Tomašev, N., Prince, L., Kaisers, M., ... & Tuyls, K. (2024). TacticAI: an AI assistant for football tactics. Nature communications, 15(1), 1906.
[23] Kumar, A., Zhou, A., Tucker, G., & Levine, S. (2020). Conservative q-learning for offline reinforcement learning. Advances in neural information processing systems, 33, 1179-1191.

Appendix

An appendix is not required, but if you have one please include it here.