Poster at The 7th International Workshop on Intrinsically Motivated Open-ended Learning (IMOL 2025), 8–10 September 2025, University of Hertfordshire, Hatfield, UK.

Skill learning using qualitative models

Ruben Spolmink[1],Marco Van Cleemput[1], and Jan Lemeire[1]

***Abstract*— This work presents a structured approach to skill learning using qualitative models, applied to the task of throwing a ball into a basket. Unlike reinforcement learning, which learns a policy without capturing task structure, qualitative models describe how action parameters such as release velocity and angle influence task outcomes. These models abstract system behaviour into sequences of symbolic states, allowing the agent to reason about changes and branching possibilities during execution. Using the QSIM framework, key trajectory points are identified and used to build a graph-based model that represents causal influences between variables. This modular structure enables the agent to adapt the model across tasks by altering goal conditions without relearning the entire system. The result is a flexible and interpretable method for skill learning that supports generalisation across tasks such as ball throwing, can knocking, and bowling.**

I. Introduction

Skill learning is crucial for the autonomous system to interact with the world. In the state of the art, skill learning is done with reinforcement learning. In reinforcement learning, a policy will map the state space with the correct action to achieve the desired state rather than learning a world model. In the case of training an autonomous agent to throw a ball into a basket, the policy is mapping the basket's location to a velocity and angle the ball has to travel, rather than learning the complete skill. The velocity and angle are action parameters of the skill of throwing a ball, which is called a parametrised skill. The skill's abstraction to action parameters allows for more generalisation. [1][2] Unlike reinforcement learning agents, a person possesses an intuitive understanding of the problem and can reason about it in a structured way. When the ball misses the basket, a person can infer how to adjust the throwing motion. These adjustments are made based on the understanding of the world. These adjustments are, however, not made numerically, but rather qualitatively, lowering the angle or slightly throwing softer. This type of intuitive reasoning can be achieved through qualitative models, which abstract away from precise numerical values to capture the structural relationships between actions and outcomes. This is done by representing system dynamics symbolically, using sign-based relationships and discrete state abstractions rather than continuous equations. A qualitative model describes how a variable influences another variable through qualitative functions. These functions describe whether the influence between variables is increasing, decreasing, or constant. When this function is no longer monotonic or when the influenced variable changes its sign, it is marked as a landmark and signals a different operating region.

[1]Ruben Spolmink, Marco Van Cleemput, and Jan Lemeire are with the Dept. of Industrial Sciences (INDI) and the Dept. of Electronics and Informatics (ETRO) of the Vrije Universiteit Brussel (VUB) Pleinlaan 2, B-1050 Brussels. `ruben.spolmink@vub.be`

II. Qualitative Trajectories and State Transitions

To apply the qualitative models to real-world control tasks, qualitative trajectories are constructed as a sequence of states in which the relationship between the variables changes or when the sign of both variables changes. In the ball-throwing task, this sequence of state transitions can be divided into three different states. When the ball is released from the gripper, the ball is no longer tied to the movement of the arm, but will gain velocity and an angle concerning the ground. When the ball describes an arc, it will reach the highest point, at which point it will alter the angle and change the direction of the velocity, signalling another transition. The arc can be modelled using QSIM [3], where gravity is modelled as a constant negative acceleration. This results in a transition at the highest point of the arc, where the velocity changes direction. Finally, when the ball reaches or hits the basket, this marks another transition. In some cases, the ball may continue its motion without any change in direction or velocity, for instance, when overshooting the basket. Even though there is no qualitative change in the variables, this situation is still represented as a distinct state in the trajectory. This is because it introduces a branching point where different future outcomes can be reasoned. In qualitative control learning [4], such branches occur naturally in the transition graph when a symbolic state can lead to several qualitatively valid outcomes. These nondeterministic branches reflect uncertainty in system behaviour, allowing the agent to reason about alternative paths toward the goal. A worked-out state transition of the basketball throwing task can be found in Figure 1

III. Creating a qualitative model from trajectories

The QSIM model provides a sequence of state changes that identify key points along the ball's motion. At the release point, the ball gains a velocity and an angle relative to the ground. These values can be measured directly. At the highest point of the arc, the ball's height, horizontal distance, and velocity are observable. At the basket, a different strategy is needed. The position of the ball is not always clearly defined due to the possible absence of a sign change in the qualitative variables. Instead, a point is marked when the ball either hits the basket, falls below its height after overshooting, or crosses the basket's horizontal position without reaching sufficient height. Since this point varies across trajectories, it is expressed as the difference between the ball's position and the location of the basket. With these reference points, a qualitative model is built to capture how variables in one state influence those in the next. This model is shown as a graph in Figure 2.

Another qualitative mode can be derived from this graph. For each of the connections, the influence of a variable is determined by changing one parameter while keeping others fixed. This allows the direction of influence to be isolated. In some cases, the relationship between variables is not monotonic. For example, increasing the release angle first increases the distance at the arc's peak. However, increasing the angle beyond 45 degrees, the distance begins to decrease. This marks a transition to a different operating region, defined by the angle's context.

Once the model is complete, the agent can adjust the throw based on the observed error. If the ball is too high and overshoots the basket, the height must be reduced. This can be done by lowering the release angle, decreasing the velocity, or adjusting the timing and speed of the arm joints.

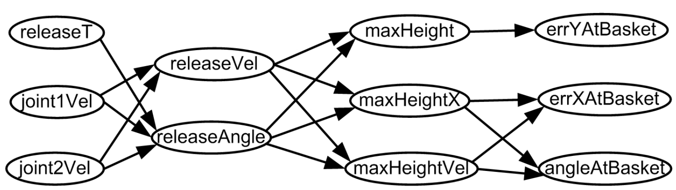

Fig. 2: Causal model for the basketball challenge with all variables involved.

IV. Adapting Model to Different Tasks

As the model is structured as a graph rather than a black box system, such as those used in reinforcement learning, it becomes possible to modify parts of the model without discarding what has already been learned. Consider the task of throwing cans off a bench, as is common at a fair. In this case, the goal is not to place the ball precisely but simply to reach a certain height with enough force. The final state representing the basket can therefore be omitted, and the model can instead treat the highest point of the trajectory as the goal. Since the graph captures how earlier variables, such as release velocity and angle, influence later outcomes like height, the structure remains consistent even if the goal changes. This idea extends to tasks where other variables are not relevant. In the case of bowling, for example, the outcome does not depend on height but only on the release speed and direction. The model can still be used, but with a reduced set of target variables. Because the relationships between the earlier variables remain valid, modifying the goal layer does not require relearning the entire model. This allows for the transfer of learned behaviour across tasks with different objectives, which is difficult to achieve using traditional reinforcement learning methods due to their tightly linked reward structures.

References

[1] B. Da Silva, G. Konidaris, and A. Barto, "Learning parameterized skills," *arXiv preprint arXiv:1206.6398*, 2012.

[2] J. Kober, A. Wilhelm, E. Oztop, and J. Peters, "Reinforcement learning to adjust parametrized motor primitives to new situations," *Autonomous Robots*, vol. 33, no. 4, pp. 361–379, 2012.

[3] B. Kuipers, "Qualitative simulation," *Artificial Intelligence*, vol. 29, no. 3, pp. 289–338, 1986.

[4] D. Šoberl and I. Bratko, "Qualitative control learning can be much faster than reinforcement learning," *Machine Learning*, vol. 114, no. 1, p. 4, 2025.

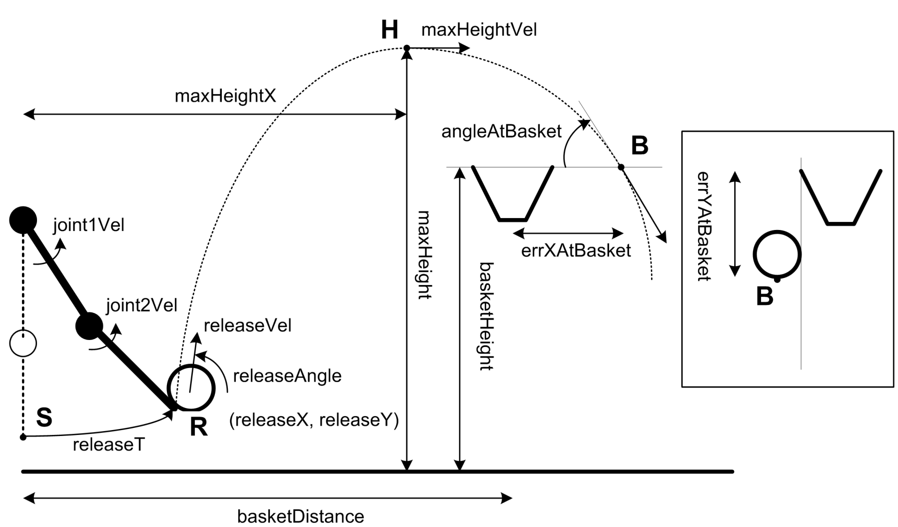

Fig. 1: A 2D basketball setting: an unsuccessful throw is presented – the ball flies above the basket. Special points of interest are: (S) the start point, (R) the location at which the ball is released, (H) the highest point of the ball's trajectory and (B) the height of the ball when it reaches the top of the basket.