

Regression and Mental Models for Decision Making on Robotic Biped Goalkeepers

Joseph G. Masterjohn¹(✉), Mihai Polceanu², Julian Jarrett¹,
Andreas Seekircher¹, Cédric Buche², and Ubbo Visser¹

¹ Department of Computer Science, University of Miami,
1365 Memorial Drive, Coral Gables, FL 33146, USA

{joe,julian,aseek,visser}@cs.miami.edu

² National Engineering School of Brest (ENIB),
LAB-STICC, 25 rue Claude Chappe, 29280 Plouzané, France
{polceanu,buche}@enib.fr

Abstract. We investigate the decision-making and behavior of robotic biped goalkeepers, applied to the RoboCup 3D Soccer Simulation League. We introduce two approaches to the goalkeeper’s behavior: first a heuristics-based approach that uses linear regression and Kalman filters for improved perception, and another based on mental models which uses nonlinear regression for ball trajectory filtering. Our experiments consist of 30,000 kick-and-save tests, using 100 random angle and distance kicks from six distance categories and four angle categories repeated 30 times. Our benchmark results show that both proposed approaches bring significant improvements for the goalkeeper’s save success rates (>200 %) and validate the applicability of the novel mental model based decision-making process.

1 Introduction

The RoboCup 3D simulation league soccer is an international contest promoting artificial intelligence and autonomous robotics. Virtual replicas of the Aldebaran NAO robot make teams of 11 players and compete against each other in a virtual game of soccer.

With the aim to improve our team’s performance in upcoming contests, the work presented in this paper focuses on an important challenge: the goalkeeper. The agent playing this role relies greatly on anticipation techniques. The first, most important step is to accurately track the ball and be able to predict its trajectory. This prediction should account for the time needed by the agent to perform a correct set of actions that would deflect the ball from the goal gate. Second, performance of attackers has improved such that the ball can be shot towards the goal with high velocities that force available reaction time below one second. This leaves little time to react and thus the goalkeeper must take advantage of every available millisecond in order to save the goal. Finally, noisy perception contributes to difficulties in accurate action planning. Therefore, the goalkeeper should begin moving to intercept the predicted ball trajectory while

also constantly improving its plan using recent perceptions. In the ideal case, it should also be able to perform last-moment adjustments using fast hand or foot movements to save goals.

The use of scripted behavior in the NAO goalkeeper, as well as in agents in general, can be highly efficient in many scenarios where a certain solution can be applied directly to the context at hand. For example, in this case, one can instruct the robot to position itself on the “goalkeeper’s arc” (an imaginary arc 2–6 yards from the goal) to maximize its range, or that it should move to a future location of the ball as soon as it is launched. However, in the cases where a decision has to be made along the way such as whether or not to dive, extend arms or feet in order to block the ball, scripted behavior becomes difficult, especially without the knowledge of whether or not it is necessary to do so and which action would give the best results.

The contributions of this paper consist in the description and evaluation of two approaches to improve the anticipation and decision performance of the goalkeeper agent, an important aspect of the RoboCup competition which has been relatively under-represented in research. We first discuss relevant work in the next section and describe the two approaches to goalie anticipation in Sect. 3. Our experimental setup and the conducted robot tests are explained in Sect. 4. Finally we discuss the pros and cons of our results and outline future work in the remaining Sect. 5.

2 Related Work

Multi-behavior goalkeepers have been studied in the realm of the RoboCup mid sized league. Although the Standard Platform League (SPL) and simulation league face unique challenges that are not present in the mid sized league, the study of behavior control within the mid sized league is still applicable. Menegatti *et al.* [13] were the first to create an *ad hoc* model for behavior and motion control. Their goalkeeper sweeps an arc in front of the goal and tries to intercept shots coming towards the goal. This work was furthered and refined by Lausen *et al.* [9]. Their goalkeeper is implemented by a 2-level hierarchical state machine that coordinates primitive tasks and behaviors. Complex motion within a task is carried out by a non-linear control algorithm.

Garcia *et al.* [7] implemented an ethological inspired architecture to generate autonomous behaviors, and focused on modeling a humanoid goalkeeper according to the rules of the SPL. The architecture implements the goalkeeper behavior using a deliberative structure for planning (self-positioning) and a reactive control mechanism for goal saving. They report being able to track the ball 100 % of the time, positioning itself properly 84 % of the time, and saving goals from 62 % of the trajectories tested.

Bozinovski et al. [3] have presented an approach for an evolving behavior model for the role of a goalkeeper. Using an emotion based self-reinforcement learning algorithm, they present a learning curve as result from training experiments.

Birbach *et al.* [2] have implemented a realtime perception system for catching flying balls with DLR’s humanoid *Rollin’ Justin*. The system tracks and predicts the trajectory of balls thrown towards the agent using a Multiple Hypothesis Tracker that uses Unscented Kalman Filters to track the different hypotheses for different balls. We expand upon the use of Kalman filters for state estimation and introduce regression methods for trajectory fitting.

Adorni *et al.* [1] used the current ball positions obtained from images frame by frame, then used a look-up table mechanism for an internal local representation of where the ball is relative to the agent and which eases computation and then compared two subsequent frames to estimate ball position, motion direction and speed. This straight-forward method worked on a Middle-Size League (MSL) agent in the beginning of RoboCup. Since then, research in soccer agents and robots has focused on several critical aspects of gameplay such as efficient omnidirectional movement [10, 11] and accurate perception [16].

Human goalkeepers have been shown to use visual cues in the shooter’s body posture to predict the direction of the ball [5], before the shot has been performed. However, in the context of the RoboCup 3D soccer competition, goalkeeper agents rely solely on the ball trajectory after shooting due to the constraints (ex. noisy perception, limited computational resources) posed by perceiving the opponent robot’s posture.

Despite abundant research into the various challenges posed by the robot soccer context, work on the goalkeeper agent is limited to simple rules that are acceptable for increasing chances of saving a goal. Moreover, goalkeepers use the same perception methods as field player agents, although they require a much more precise, three-dimensional ball localization for brief periods of time.

Goalkeepers also require special skills as they are the only player agents which are allowed to intentionally dive and use their hands to block the ball. Diving is the most popular goalkeeper skill present in the competition, but deciding to activate this skill relies on handcrafted rules. Such rules include instructing the goalie to move towards the position where the ball will be found before entering the goal and if the distance between the goalie and the ball is too large for a direct block, the goalie would perform a dive [12]. However, there are exceptions where after the dive the ball bounces over the fallen goalie resulting in a goal. Another example is when diving happens too slowly to block the ball and a slight leg movement would be more efficient instead.

3 Approach

We investigate two approaches to improve goalkeeper efficiency in the RoboCup 3D simulation competition. First, a rule-based approach which uses linear regression and Kalman filters to filter the ball trajectory. Second, a novel approach to decision-making that utilizes nonlinear regression for ball perception. In the following, we discuss the details of each approach and provide insight into their strengths and drawbacks.

3.1 Thiel-Sen and Kalman-Filter

In our first approach, we filter the ball position and velocity using linear regression, to estimate the location where the ball will pass the goal line. Based on this prediction, the goalie employs a deterministic behavior in order to improve its success in obstructing the ball. In the following we describe the logic behind this behavior and how the filtering has been implemented.

Behavior Logic. We describe the rational stages of behavior of the goalkeeper agent, first actions for the initial positioning of the agent, moving to obstruct the ball, final actions including decisions and diving and other auxiliary checks and behavior to maintain stable execution of the agent.

The agent acting as the goalkeeper first analyzes how to position itself in the goal based on the initial ball position of the free kick to be taken. The goalkeeper has one of five positions on the goal line to choose from. The rationale for making these options available for the goalkeeper are motivated from human soccer, watching various free kicks and observing the behavior of professional goalkeepers. The options in initial position are as follows: (a) the center of the goal if the kicker is directly in front of the goal. (b) between the center and the right goal post if the kick is coming from the right and is a short kick. (c) between the center and the left post if the kick is coming from the left and is a short kick. (d) close to the right post if the kick is coming from the right and is a long kick. (e) close to the left post if the kick is coming from the left and is a long kick. A “short” vs. “long” kick is determined by a distance threshold.

The goalkeeper, based on perceptions of his own position and the position of the ball, calculates the distance between himself and the ball as well as the angle between the vector from goalkeeper to ball and the center line. This first perception indicates the predicted direction of the free kick, the vector from the opponent to the center of the goal.

The goalkeeper uses the calculated angle to access a set of behaviors followed by the calculated distance to decide on a specific behavior within this set. (e.g. - an angle $\theta \in [\frac{\pi}{3}, \frac{\pi}{2}]$ is indicative of a kick coming from the left side, and a distance $d < threshold$ indicates a short kick.)

The goalkeeper receives filtered perceptions of the balls position and velocity from a provider module, the details of which are discussed later. The goalkeeper projects the ball along the velocity vector onto the goal line to determine the position at which it crosses the goal line. The goalkeeper characterizes a kick by observing the velocity and detecting when it accelerates past a certain threshold (i.e. - a kick has been performed). This sets off the rational actions to obstruct the path to goal. Once the kick has been taken, the goalkeeper uses the predicted goal line crossing to move towards that position.

Once the predicted ETA of the ball on the goal line becomes less than the estimated time needed to perform a dive action, the goalkeeper makes its final decisions. (1) If he is close enough to the predicted goal line crossing, he will remain in place to obstruct the ball (2) If he is too far from the crossing to obstruct with his feet, he will dive in the direction of the goal line crossing.

Diving is actuated by a predetermined and hand tuned motion defined by an interpolated sequence of joint angle values, for each actuated joint in the agent.

Methods and Implementation. Upon initial analysis of the accuracy of the existing ball position in the software, we decided that the first and most elementary method to predicting the ball velocity would be to compute a simple linear regression on the existing ball positions. This method operates under a few assumptions that are not accurate in the realm of the simulator:

1. $|v| \neq 0$, for v the velocity of the ball
2. $\forall \delta t \in \mathbb{R}, x'([t, t + \delta t]) = c, c \in \mathbb{R}$ (i.e. - constant velocity)

This model does not take into account the complex motion of the ball, but it does serve as a starting point for an efficient way to model the ball velocity. In order to do the initial regression, the Thiel-Sen algorithm was used. This method for calculating the linear trend of a set of points is not as sensitive to outliers as other methods [17]. The full algorithm for determining the velocity follows:

```

function LINEAR REGRESSION( $S \leftarrow [(x_1, y_1), \dots, (x_n, y_n)]$ )
   $M \leftarrow$  median of all unique slopes  $m = \frac{y_j - y_i}{x_j - x_i} \mid i \neq j$ 
   $v \leftarrow (1, M)$  or  $(-1, -M)$  depending on the rough direction of the samples.
   $v \leftarrow$  unitized  $v$  scaled by distance between  $p_n$  and  $p_1$  divided by  $t_n - t_1$ 
  return  $v$ 
end function
    
```

While the first method succeeded in giving an estimate of the ball velocity accurate enough to predict the goal line crossing and ETA, a second, more complex and contemporary method (Kalman Filter) was used for comparison. This decision was made with the additional foresight that the success of the linear regression model might be dependent on the magnitude of error realized in the simulation. Thus, a method robust enough to predict the state of the ball for a real world environment would then be integral. The Kalman Filter is a well known and studied method of estimating variables that cannot be directly measured through a series of noisy measurements, observed over time [6]. We formulate the Kalman Filter matrices as follows:

$$X = \begin{bmatrix} x \\ y \\ x' \\ y' \end{bmatrix} \quad F = \begin{bmatrix} 1 & 0 & \delta t & 0 \\ 0 & 1 & 0 & \delta t \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad P = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1000 & 0 \\ 0 & 0 & 0 & 1000 \end{bmatrix} \\
 Q = \begin{bmatrix} 0.01 & 0 & 0 & 0 \\ 0 & 0.01 & 0 & 0 \\ 0 & 0 & 0.01 & 0 \\ 0 & 0 & 0 & 0.01 \end{bmatrix} \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad R = \begin{bmatrix} 0.95 & 0 \\ 0 & 0.95 \end{bmatrix}$$

P is the a posteriori error covariance matrix (a measure of the estimated accuracy of the state estimate); F is the state transition matrix; Q is the covariance of the process noise; H is the observation model; R is the covariance of the observation noise; X is the state vector.

The state transition matrix based on the kinematic equations:

$$\begin{array}{ll} x_{i+1} = x_i + v_{ix}\delta t & y_{i+1} = y_i + v_{iy}\delta t \\ v_{(i+1),x} = v_{ix} & v_{(i+1),y} = v_{iy} \end{array}$$

Here we see some simplifications that will inevitably affect the accuracy of this estimation. The velocity is assumed to be constant between small time steps, as in the regression model. If we assume deceleration due to friction is negligible, especially during a hard kick or a kick that spends a considerable amount of its trajectory in the air, the accuracy will be acceptable for the duration of a kick. Another simplification is that this model neglects the z movement of the ball. The dynamics of the ball are drastically different in the air vs. on the ground and if the goalie decides to dive when the ball enters the goal at a high position he will miss the ball.

3.2 Nonlinear Regression and Experimental Anticipation

Our second approach consists of enhancing the goalkeeper with a mental model of itself that it can simulate ahead of time to anticipate the efficiency of its current actions and eventual changes to current behavior. We first give a brief introduction to the concepts of this approach, the requirement of more precise velocity estimation and finally how the goalkeeper can use this cognitive layer to improve its results.

Orpheus, a Generic Mental Simulation Framework. To enable the goalkeeper agent to predict its environment, itself included, we use a generic open source framework for mental simulation entitled ORPHEUS¹ which has been successfully applied in different contexts requiring anticipative agents [14, 15].

The ORPHEUS architecture provides an agent with an “imaginary world” which can be used to evaluate the outcomes of its actions in accordance with other objects (e.g. ball) and agents that populate the environment it is placed in. To clarify, the goalkeeper has its own functional representation of itself and the ball, and is able to manipulate it towards improving its behavior in the game. Providing the agent with models of itself and its environment enables it to evolve and evaluate various courses of actions in its imagination, based on which it can make decisions. This cognitive layer functions in parallel to the main agent behaviors, which makes it well suited in the context of robot soccer agents where interruptions in sending motor instructions cause unwanted results.

Two important aspects lead to a successful use of the mental simulation paradigm: perception and mental model accuracy. Specifically, the position and velocity of the ball and robot as well as the models of environment physics and the robot’s own body motion must be accurate enough so that future events can be successfully predicted. Hence our focus is on improving velocity estimation and constructing the mental models for the robot.

¹ ORPHEUS source code: <https://bitbucket.org/polceanum/orpheus>.

Nonlinear Regression for Ball Position and Velocity. In the robot soccer simulation contest, as well as for real settings, the trajectory of the ball is shaped by damping factors. However, we can safely assume that, as long as the ball does not collide with any object as it’s traveling, it will describe a smooth non-linear trajectory. Unlike linear models, using a curve to approximate the trajectory, even on the X and Y axes where gravity does not intervene, can account for the damping factors.

In this approach, we use the Levenberg-Marquardt algorithm as implemented in Dlib-ml [8] to fit a curve on the raw online perception data. The choice of algorithm was based on its good performance in practice, efficiency on small input datasets and that it allows describing the curve model, as is the case for the goalie’s perception of the ball trajectory. We develop the damped motion equations $x_n = x_{n-1} + v_{n-1}\delta t$ and $v_n = \zeta v_{n-1}$ over multiple time steps, resulting in the model to be used in the algorithm:

$$\frac{p_1 \delta t (1 - \zeta^n)}{1 - \zeta} + p_0$$

where δt is the time difference between two perception frames, ζ is the damping factor, n is the step number, p_0 and p_1 are the position and velocity parameters respectively. With this approach, we store the ball perception frames after the kick and refit the model parameters at each step to gradually improve accuracy. For the Z axis, we extend the model to account for gravitational acceleration:

$$p_2 \delta t^2 \left(n - \frac{1 - \zeta^n}{1 - \zeta} \right) + \frac{p_1 \delta t (1 - \zeta^n)}{1 - \zeta} + p_0$$

The obtained velocity estimation enables more accurate prediction of the ball for default behavior such as walking towards the ball but also for prediction using mental simulations.

Learning Body Movement Models and Imaginary Reenactment. Once the ball can be predicted, the second step in this approach is to have the robot learn the effects of its actions and be able to reenact them in its imaginary world. This produces a set of mental models which are then used to perform mental simulations.

To this end, we rely on querying the location of each of the robot’s body parts which are represented with primitives in the imaginary world. Various actions can be learned by having the robot perform it, while a model training module provided by the ORPHEUS framework distributes the data to the models which are being trained. For the goalkeeper’s actions, we use the K-Nearest Neighbors (KNN) algorithm as implemented in MLPack [4]. The dataset is constructed with the position, velocity and rotation in function of time, of each body part at each time step, relative to the goalie’s location at time zero.

Reenactment of each learned action with KNN uses the time and position of the robot’s body parts and finds the closest corresponding velocities and rotations. From this point forward, ORPHEUS can perform mental simulations

on demand, using the learned models through the process briefly described in the following. Perception data (robot, body part and ball position, velocity and rotation) at a given moment in time is submitted to the cognitive layer. From this data, the mental simulation process constructs subsequent mental images by employing the available mental models. This process leads to obtaining a future state of the environment, to a certain accuracy, ahead of time. By evaluating this future state, the robot can make a range of decisions, as detailed in the following.

Self Monitoring and Deciding to Change Behavior. There are limiting factors on how the goalkeeper agent can behave to deflect the ball. The main such factor is the high ball velocity achieved by other teams in the RoboCup 3D Simulation League, which gives little time to react. Moreover, the robot does not have the ability to move fast enough to reach the ball in many situations. There are however situations when extending its limbs or diving may help deflecting the ball.

The goalie can assess whether or not to perform an action and if so, which one is more favorable, by running mental simulations where it approximates the ball trajectory and its own movements and evaluate the results. Testing showed that simply moving towards the predicted position of the ball when it would reach the defense zone performs well, given satisfactory estimations of the ball position and velocity. Therefore, starting from a reactive decision to move towards this location of interest, we use mental simulations to predict whether or not simply walking would enable the robot to deflect the ball. As soon as the kick is detected, a coarse future location of the ball is computed and the robot starts moving towards that location. In parallel, it uses perceptual information to imagine the success or failure of this strategy.

We studied outcome prediction accuracy (Table 1) with a set of 100 penalty shots from random locations, 5 to 10 meters away from the goal gate. This resulted in an average of ~ 0.5 s in which the goalie could apply a different decision, after subtracting the time required to obtain reasonable ball velocity accuracy (~ 0.1 s) and the time required for mental simulations to finish; in our implementation, mental simulation speed ratio is ~ 3 x, i.e. the robot can simulate 1 s of real events in ~ 0.33 s.

As shown in Table 1, the accuracy of predicting the outcome of a set of actions rises above 50 % (random choice) as soon as valid data is acquired from the environment, and improves with time. Prediction failure is caused by the difference between mental models and reality, which could be later improved by learning more accurate models.

Concurrently with self monitoring, the goalkeeper also imagines and evaluates a set of actions that can be performed. This results in a set of solutions in the case in which it decides the current set of actions will not be effective. If a failure is predicted, the goalkeeper is able to choose one of the previously imagined solutions that may result in a last-moment save.

Table 1. Outcome prediction accuracy (% correct predictions) based on mental simulation, in function of the time before the ball passes the goalie’s location.

| <i>Time left</i> | 0.5 s | 0.4 s | 0.3 s | 0.2 s | 0.1 s |
|------------------------------------|--------|--------|--------|--------|--------|
| Correctly predicted outcome | 69.1 % | 80.1 % | 84.8 % | 89.5 % | 93.2 % |

This experimental approach is limited by the set of possible actions that the robot can perform, and the time available to evaluate the outcomes of these actions. However, more efficient actions can be later added to increase the success rate. We investigate whether this approach is feasible for full integration in the goalkeeper. Therefore, during the tests we enable the complete reasoning process to verify that no negative effects are introduced.

4 Results

We empirically evaluated the two approaches – with Linear Regression/Kalman-Filter (LR/KF) and Nonlinear Regression (NLR) – on a range of kick distance intervals. For the second approach, we developed two versions so that the experimental mental model based prediction (ORPHEUS) can be evaluated separately (Fig. 1).

4.1 Experiment Setup

The evaluation consisted in running a goalkeeper with an attacker that will kick the ball from a random distance within a specified interval, at a given angle. As a benchmark, we tested the naive goalkeeper behavior, used so far by the RoboCane team in previous competitions.

There would be a large variance in the kicks of the striker, even if the striker kicks from the same distance and angle. Therefore, we reproduce the exact same kicks by assigning a velocity to the ball in the beginning of the kick. Thus, we are able to create a random sequence of kicks and repeat the exact same kicks multiple times using different goalies. Variances in the results are only caused by variance in the goalkeepers behavior, but not by variances in the kicks.

To better estimate the relationship between distance, shot angle and performance, we devised 2 m intervals starting from 3 m away (which is roughly the margin of the goalkeeper’s area) to 15 m away (midfield). The obtained results

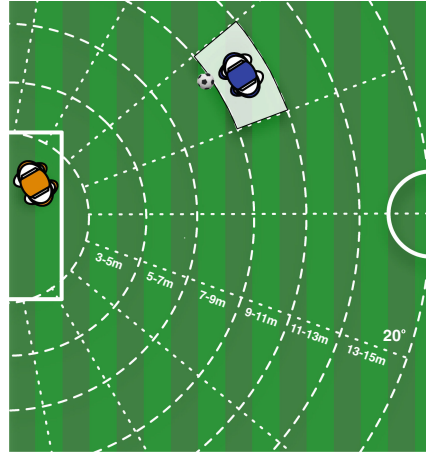


Fig. 1. Setup for our experiments: six distance categories, 4×2 angle categories, 100 random kicks repeated 30 times on five agents.

for each interval (Table 2) are the success rates for 100 different kicks repeated 30 times and averaged out (3,000 kicks per interval).

The experiment has been performed on an Intel Core i7-5930K machine with 6 cores. Results were also verified on two less powerful machines – a Quad Core Intel Xeon and a Dual Core Intel Pentium – and replicate the main experiment.

4.2 Goalkeeper Performance

Results of distance-based evaluation (Table 2) show that both approaches bring significant improvements over Naive goalkeeper which was used by the RoboCane team in previous competitions.

Table 2. Goalkeeper success rates for 100 random kicks from different distances. The kick angles are chosen randomly from -45 to 45° . Each rate is averaged over 30 repetitions of the 100 kicks (standard deviation shown in parentheses).

| Distance | Naive goalie | Kalman Filter | LR | NLR | ORPHEUS |
|----------|--------------|---------------|--------------|--------------|--------------|
| 3–5 m | 29.8 % (2.2) | 35.8 % (3.6) | 35.6 % (3.4) | 46.1 % (2.8) | 45.8 % (2.3) |
| 5–7 m | 22.0 % (1.9) | 31.8 % (3.4) | 31.7 % (4.0) | 50.4 % (2.9) | 50.2 % (2.3) |
| 7–9 m | 19.5 % (1.9) | 31.8 % (4.6) | 30.1 % (3.0) | 48.6 % (3.0) | 48.0 % (3.0) |
| 9–11 m | 17.8 % (1.9) | 23.9 % (3.5) | 23.4 % (2.9) | 47.5 % (3.4) | 46.9 % (3.6) |
| 11–13 m | 17.0 % (2.4) | 21.5 % (3.3) | 22.1 % (2.9) | 45.0 % (3.5) | 45.1 % (3.8) |
| 13–15 m | 14.8 % (2.0) | 18.6 % (2.7) | 19.5 % (2.8) | 43.9 % (3.5) | 44.8 % (3.4) |

The first important difference between the naive goalkeeper and the two approaches presented in this paper is that previously, the naive keeper was instructed to place itself at the middle between the ball position and the goal, while the proposed goalkeeper’s behavior uses the goalkeeper’s arc. Midway positioning can be exploited by shooting the ball over the goalkeeper, resulting in easy scoring, illustrated by the constant drop in success rates with respect to distance. This effect is significantly less prominent in the LR/KF and NLR approaches, due to superior initial positioning of the goalkeeper.

The increased success rates obtained by the LR/KF and NLR approaches are also attributed to the improvements in ball tracking (Fig. 2). In the context of the RoboCup 3D soccer simulation league, the ball does not have an ideal trajectory due to the realistic simulation performed by the physics engine which includes friction and damping. This leads to better results with NLR caused by a better fit of the damped trajectory.

Results of the experimental ORPHEUS version of the goalie do not vary in a statistically significant manner from the NLR version, which shows that it is feasible to integrate prediction in the goalkeeper without suffering a trade-off.

In order to more extensively evaluate our goalies’ performance, we also varied the angle from which the ball is shot and measured success rates (Table 3).

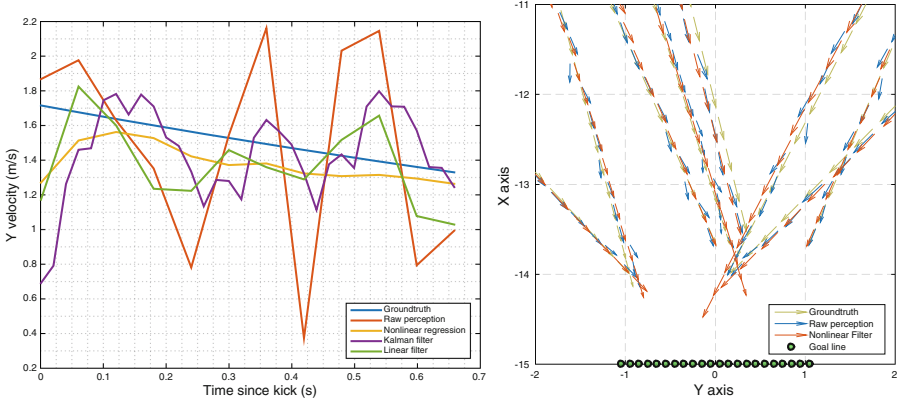


Fig. 2. Comparison of ball velocity values for Y axis over time since the kick (left) and 2D velocity perception during kicks (right).

Table 3. Average goalkeeper success rates for 100 random kicks as in Table 2, here grouped by angle (using the distance 5–7 m)

| Angle ° | Naive goalie | Kalman Filter | LR | NLR | ORPHEUS |
|---------|--------------|---------------|--------------|--------------|--------------|
| 0–20 | 18.3 % (2.1) | 26.8 % (3.7) | 26.1 % (3.8) | 43.6 % (3.3) | 43.4 % (2.8) |
| 20–40 | 21.5 % (2.0) | 31.9 % (3.5) | 30.0 % (3.1) | 50.0 % (2.9) | 49.8 % (2.7) |
| 40–60 | 33.1 % (2.7) | 38.4 % (3.7) | 38.7 % (4.5) | 65.0 % (3.3) | 64.7 % (3.1) |
| 60–80 | 63.3 % (3.6) | 32.1 % (2.9) | 37.0 % (3.3) | 76.0 % (3.5) | 78.1 % (3.5) |

Results show that the increase in success rates is directly proportional with the shooting angle for the Naive, LR, NLR and ORPHEUS versions.

5 Conclusion and Future Work

We have presented two approaches to improving the goalkeeper agent of the RoboCane team: one based on linear regression, Kalman Filter and scripted decision making, and another based on nonlinear regression and an experimental prediction capability based on internal mental models. The proposed approaches have been extensively evaluated along with the naive version used in previous competitions as benchmark, showing significant improvements over all kicking conditions (distance intervals and angles from the goal).

Design features of each approach were correlated with the results they obtained, and concluded the importance of accurate perception and fast reaction in the goalkeeper’s efficiency. We have also provided a novel mechanism for the goalkeeper to perform auto-evaluation which has the potential to further improvement of our goalkeeper’s performance.

We intend to continue the improvement of the goalkeeper agent by further studying the behavior of the proposed approaches in exceptional cases such as collisions of the ball with the ground or other objects. Future work will also include efforts on improving the prediction ability of the ORPHEUS version by learning better models and integrating it into the decision process of the goalkeeper. The aim of this approach is to enable the goalkeeper to adjust its scripted motion to save more seldom and more difficult shots.

References

1. Adorni, G., Cagnoni, S., Mordonini, M.: Landmark-based robot self-localization: a case study for the RoboCup goal-keeper. In: Proceedings of the 1999 International Conference on Information Intelligence and Systems, pp. 164–171. IEEE (1999)
2. Birsch, O., Frese, U., Bäuml, B.: Realtime perception for catching a flying ball with a mobile humanoid. In: 2011 IEEE International Conference on Robotics and Automation, pp. 5955–5962 (2011)
3. Bozinovski, S., Jäger, H., Schöl, P.: Engineering goalkeeper behavior using an emotion learning method. In: Proceedings of the RoboCup Workshop, KI 1999: Deutsche Jahrestagung für Künstliche Intelligenz, pp. 48–56 (1999)
4. Curtin, R.R., Cline, J.R., Slagle, N.P., March, W.B., Ram, P., Mehta, N.A., Gray, A.G.: MLPACK: a scalable C++ machine learning library. *J. Mach. Learn. Res.* **14**, 801–805 (2013)
5. Diaz, G.J., Fajen, B.R., Phillips, F.: Anticipation from biological motion: the goalkeeper problem. *J. Exp. Psychol. Hum. Percept. Perform.* **38**(4), 848 (2012)
6. Faragher, R., et al.: Understanding the basis of the Kalman filter via a simple and intuitive derivation. *IEEE Sig. Process. Mag.* **29**(5), 128–132 (2012)
7. García, J.F., Rodríguez, F.J., Fernández, C., Matellán, V.: Design an evaluation of RoboCup humanoid goalie. *J. Phys. Agents* **4**(2), 19–26 (2010)
8. King, D.E.: Dlib-ml: a machine learning toolkit. *J. Mach. Learn. Res.* **10**, 1755–1758 (2009)
9. Lausen, H., Nielsen, J., Nielsen, M., Lima, P.: Model and behavior-based robotic goalkeeper. In: Polani, D., Browning, B., Bonarini, A., Yoshida, K. (eds.) RoboCup 2003. LNCS (LNAI), vol. 3020, pp. 169–180. Springer, Heidelberg (2004)
10. Liu, J., Liang, Z., Shen, P., Hao, Y., Zhao, H.: The walking skill of Apollo3D – the champion team in the RoboCup2013 3D soccer simulation competition. In: Behnke, S., Veloso, M., Visser, A., Xiong, R. (eds.) RoboCup 2013. LNCS, vol. 8371, pp. 104–113. Springer, Heidelberg (2014)
11. MacAlpine, P., Barrett, S., Urieli, D., Vu, V., Stone, P.: Design and optimization of an omnidirectional humanoid walk: a winning approach at the RoboCup 2011 3D simulation competition. In: AAAI (2012)
12. MacAlpine, P., Collins, N., Lopez-Mobilia, A., Stone, P.: UT Austin Villa: RoboCup 2012 3D simulation league champion. In: Chen, X., Stone, P., Sucar, L.E., van der Zant, T. (eds.) RoboCup 2012. LNCS, vol. 7500, pp. 77–88. Springer, Heidelberg (2013)
13. Menegatti, E., Nori, F., Pagello, E., Pellizzari, C., Spagnoli, D.: Designing an omnidirectional vision system for a goalkeeper robot. In: Birk, A., Coradeschi, S., Tadokoro, S. (eds.) RoboCup 2001. LNCS (LNAI), vol. 2377, pp. 81–91. Springer, Heidelberg (2002)

14. Polceanu, M., Buche, C.: Towards a theory-of-mind-inspired generic decision-making framework. In: IJCAI 2013 Symposium on AI in Angry Birds (2013)
15. Polceanu, M., Parenthoen, M., Buche, C.: ORPHEUS: mental simulation as support for decision-making in a virtual agent. In: 28th International Florida Artificial Intelligence Research Society Conference (FLAIRS-28), pp. 73–78. AAAI Press (2015)
16. Seekircher, A., Abeyruwan, S., Visser, U.: Accurate ball tracking with extended kalman filters as a prerequisite for a high-level behavior with reinforcement learning. In: The 6th Workshop on Humanoid Soccer Robots at Humanoid Conference, Bled (Slovenia) (2011)
17. Sen, P.K.: Estimates of the regression coefficient based on Kendall's tau. *J. Am. Stat. Assoc.* **63**(324), 1379–1389 (1968)