

Bold Hearts Team Description

RoboCup 2013 SSL 3D

Sander van Dijk, Drew Noakes, Valerio Lattarulo, Alex Metaxas, and Daniel Polani

School of Computer Science
University of Hertfordshire, UK

Abstract. We aim at systematically developing a battery of principled methods to generate behaviours useful to achieve a viable RoboCup 3D gameplay. This is particularly interesting and challenging in humanoids since any possible tactical, strategical or cooperative aspects can only be successfully tackled once the basic skills are in place. The construction of basic skills in humanoids is usually an intricate business that requires a large amount of hand-tuning. We aim to develop a systematic path towards reducing this amount of hand-tuning both on the perception as well as the actuation side. We approach this by combining principled methods, as well as hand-coded components. The long-term goal is to ultimately replace the hand-crafted structuring of the code by learnt frameworks. This Team Description Paper discusses the aspects the Bold Hearts team currently concentrates upon.

1 Base and Locomotion

Team Bold Hearts has competed in the RoboCup Soccer Simulation league since 2003. The first two years the team participated in the 2D competitions, in 2005 it joined the 3D community. At the beginning of 2009 a full restart of the team was initiated. To get the Bold Hearts up to steam quickly, the new code is based on the libbats library released by the Little Green BATS¹. Team Bold Hearts is now the maintainer of this library.

For the 2012 competitions we developed a new Inverse Kinematics based gait generation framework, which replaced the simple open loop oscillator model, based on [3]. This new framework significantly further increased locomotion speed, resulting in the third place in the world championship of 2012.

The gait generator works step-based: the properties of a full step are determined at the beginning of the step, and then kept constant until the step is finished. One step consist of a double support phase, and a swing phase, where the swing leg moves in a semi circle, while the stance foot moves horizontally (cf. Fig. 1). Table 1 lists the global gait parameters, which are fixed throughout the robot's lifetime.

¹ See <http://www.launchpad.net/libbats/>

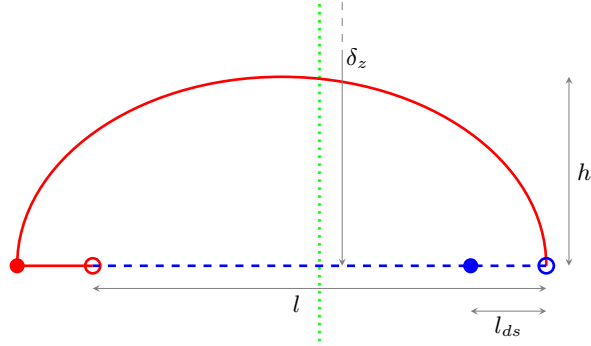


Fig. 1. Paths described by feet while walking. Red: swing leg, blue: stance leg. Open circles denote start of step, filled disks denote the end of double support phase/start of swing phase. The dotted green line shows the vertical axis that passes through the hip joint.

Parameter	Description
\bar{h}	Height of step (m)
l	Length of step (m, max)
w	Width of step (m, max)
l_{ds}	Length of double support (m)
$\delta_{x,y,z}$	Offset of step center, relative to hip joint (m)
T	Period (s)
α	Turn angle (rad, max)

Table 1. Gait parameters. Also see Fig. 1

At the beginning of a step, the start and end positions, in Cartesian coordinates relative to the hip, of both feet are determined. The start positions are equal to the end positions of the previous step, to ensure smooth transitions. The end positions are determined based on the parameters of Tab. 1, possibly scaled based on the target speed.

During the step, the x and y coordinates of the feet are calculated as combinations of the start and end positions. During the double support phase they are as follows:

$$\begin{bmatrix} x \\ y \end{bmatrix}_{st|sw} = \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}_{st|sw} + \frac{t}{T} \left(\begin{bmatrix} x_1 \\ y_1 \end{bmatrix}_{st} - \begin{bmatrix} x_0 \\ y_0 \end{bmatrix}_{st} \right), \quad (1)$$

where x_0 and x_1 are the coordinates at start, respectively end of step, t is time since start of step, and the subscripts st and sw denote 'stance foot' and 'swing foot'. The z coordinate is kept constant, at δ_z .

During the swing phase, the stance foot follows the same equation as above. The swing foot on the other hand now moves according to:

$$\begin{bmatrix} x \\ y \end{bmatrix}_{sw} = \begin{bmatrix} x_{ds} \\ y_{ds} \end{bmatrix}_{sw} + \frac{t - t_{ds}}{T - t_{ds}} \left(\begin{bmatrix} x_1 \\ y_1 \end{bmatrix}_{sw} - \begin{bmatrix} x_{ds} \\ y_{ds} \end{bmatrix}_{sw} \right), \quad (2)$$

where x_{ds} and t_{ds} are a coordinate and the time at the end of the double support phase. The z coordinate of the swing foot follows $h \cdot \sin(\frac{t-t_{ds}}{T-t_{ds}}\pi)$.

Once the Cartesian coordinates of each foot are determined, inverse kinematics is used to determine the joint angles needed to reach these coordinates. Applying some simple geometry gives the following equations for the forward motion:

$$\theta_{hip_x} = \text{atan}\left(\frac{y}{-z}\right) + \text{acos}\left(\frac{l_1^2 + l_3^3 - l_2^2}{2l_1l_3}\right) + \theta_{h/k} \quad (3)$$

$$\theta_{knee_x} = \text{acos}\left(\frac{l_1^2 + l_3^3 - l_2^2}{2l_1l_2}\right) - \pi + \theta_{k/a} \quad (4)$$

$$\theta_{ankle_x} = \theta_{knee_x} - \theta_{hip_x} \quad (5)$$

Here, l_1 and l_2 are the lengths from hip joint to knee joint and from knee joint to ankle joint, respectively, and l_3 is the desired distance from hip to ankle, i.e. $\sqrt{x^2 + y^2 + z^2}$; the angles $\theta_{h/k}$ and $\theta_{k/a}$ are fixed offsets due to the knee joint not being placed on a vertical line with the hip and ankle joints in the zero position. The ankle angle is chosen to keep the foot flat at all time.

An important issue to handle is anterior/posterior stability. Currently, a basic method is used to lean the torso in the direction of movement to counteract angular momentum caused by the walking gait:

$$f = \min\left(\frac{v_y(t)}{v_{max}} + \alpha a_y(t), 1\right) \quad (6)$$

$$\theta(t) = f \cdot \theta_{max}, \quad (7)$$

where $v_y(t)$ is the agent's desired forward velocity at time t , v_{max} its maximum velocity, $a_y(t)$ its desired forward acceleration, θ_δ the added torso pitch and $\theta_{\delta_{max}}$ the maximum torso pitch. Efforts are under way to make this method more adaptive and effective, using reinforcement learning methods and stability measures like the ZRAM-point.

2 Gait Optimization

The parameters of the walking gait have been optimized with a new optimization algorithm called the Alliance Algorithm (AA) [1]. The AA is based on the metaphoric idea that a certain number of tribes struggle to conquer an environment, which offers resources to survive. Two features characterize each tribe: the strength and the resources necessary for its survival. In order to increase their strength, the tribes may enter into alliances. Also an alliance is characterized by a strength and an amount of necessary resources, these two values depend on the features that characterize the tribes inside the alliance. The execution of the algorithm is based on the formation of alliances. The algorithm will end when the strongest alliance will be created: it will be able to conquer the environment, to meet its resource demand and force other alliances out of the environment.

It is possible to perform several iterations of the algorithm starting with tribes that are influenced by the previous strongest alliance.

In this scenario there are different entities that play a fundamental role:

1. A tribe is a tuple composed of: a point in the solution space, a strength, a resource demand and an alliance
2. The set of alliances is a mutually disjoint partition of tribes. Each alliance represents a point in the solution space, defined by the tribes that compose the alliance
3. The strength of tribe/alliance is the value obtained with the objective function with the respective solution
4. The amount of resources needed by a tribe/alliance is the value obtained with the constraint function on the respective solution

The tribes that compose the strongest alliance represent the solution of the problem.

In experiments performed with this algorithm, the objective function is the distance traversed by an agent in a fixed amount of time. With the Alliance Algorithm a solution was found that achieved a walking speed of 0.93 m/s, an increase of 36% over the gait used by our team in 2010. Also, the algorithm found such solutions significantly faster than the genetic algorithm that was used and optimized by our team over the last few years. Full results have been submitted to the RoboCup 2011 Symposium. Currently work is done to develop new gait generators that can be optimized further with the AA.

3 Localization and Active Vision Through Infotaxis

One of the challenges introduced in 2009 for the Robocup 3D Simulation teams were the restrictions placed on the vision sensor. The previous two years the simulated robots were equipped with perfect 360-degrees omni vision cameras, making the environment fully accessible. From this year, however, a restricted vision sensor is introduced, similar to that used in the spheres version of the simulator until 2006. This sensor has a range of 120 degrees on both the horizontal as the vertical axis and supplies noisy data about the objects within its field of sight. The next sections describe methods and directions of research we use to handle this new challenge.

3.1 Localization

We supply the agents with a localization mechanism that maintains their global location in world coordinates. Many tasks can be achieved with only relative position information, for instance to kick the ball into a goal the relative position of the agent to the ball and to the goal is sufficient. Global coordinates however make it easier for the agent to deduce more about the world, like the trajectory of the ball and whether the team is in an attack or defence situation. In this section

we will describe the Kalman filter localization method, a traditional method used to solve prediction problems, as described in [2] and [6].

With this method, the agents estimated location is described by a multivariate normal distribution $N(\mathbf{x}, \mathbf{\Sigma})$ with means \mathbf{x} , here a 6-dimensional vector depicting the agents 3D coordinates and velocity, and covariance matrix $\mathbf{\Sigma}$. After each time step this estimate is refined in two steps: first a prediction is made based on the dynamics of the environment and the agents actions, secondly this prediction is updated by integrating observations.

Predict In the prediction step at timestep k the mean $\mathbf{x}_{k|k1}$ and covariance matrix $\mathbf{\Sigma}_{k|k1}$, where $(\cdot)_{k|l}$ means at timestep k , given all observations up to and including timestep l , are determined as follows:

$$\mathbf{x}_{k|k1} = \mathbf{A}\mathbf{x}_{k1|k1} + \mathbf{B}\mathbf{u}_{k1} \quad (8)$$

$$\mathbf{\Sigma}_{k|k1} = \mathbf{A}\mathbf{\Sigma}_{k1|k1}\mathbf{A}^T + \mathbf{Q} \quad (9)$$

where \mathbf{A} is the state transition model relating the state of the previous time step to that of the current timestep, \mathbf{u}_k is the control vector at time step k reflecting the agents actions, and \mathbf{Q} is the process noise.

Update The update step uses observations of landmarks at the current timestep, \mathbf{z}_k , to refine the estimate:

$$\mathbf{K}_k = \mathbf{\Sigma}_{k|k1}\mathbf{H}^T(\mathbf{H}\mathbf{\Sigma}_{k|k1}\mathbf{H}^T + \mathbf{R}_k)^{-1} \quad (10)$$

$$\mathbf{x}_{k|k} = \mathbf{x}_{k|k1} + \mathbf{K}_k(\mathbf{z}_k - \mathbf{H}\mathbf{x}_{k|k1}) \quad (11)$$

$$\mathbf{\Sigma}_{k|k} = (\mathbf{I} - \mathbf{K}_k\mathbf{H})\mathbf{\Sigma}_{k|k1} \quad (12)$$

where \mathbf{H} is the observation model relating an observation to a location world coordinates and \mathbf{R}_k is the observation noise covariance matrix. The matrix \mathbf{K}_k is the gain or blending factor that minimizes the posterior error covariance. Note that the observation noise model depends on the current time step, since the noise when observing far away landmarks is larger than with nearer objects.

4 Single-Landmark Observations

An observation consists of agent coordinates in the global coordinate system. These can be obtained through triangulation or trilateration of the observed locations of several landmarks. However, the gyroscopic sensor of the agent gives sufficient information to achieve an observation by using only a single landmark. To achieve this, we maintain the current rotation matrix \mathbf{T}_k of the agent relative to the field. After beaming we set $\mathbf{T}_k = \mathbf{I}$. In all subsequent timesteps we update the matrix using the angular velocity measurement $\dot{\theta}_k$, given by the gyroscopic sensor. Based on this measurement a rotation matrix $\mathbf{\Theta}_k$ is constructed, describing the rotation since the last time step in the global coordinate frame. Finally, this matrix is used to obtain the new global rotation matrix estimate:

$$\mathbf{T}_k = \mathbf{T}_{k-1} \cdot \mathbf{\Theta}_k \quad (13)$$

With this new estimate a local observation can be transformed into a global location observation, enabling accurate localization based on a single landmark.

4.1 Information Gathering

During a match an agent that focusses solely on the ball will receive sufficient observations of landmarks to be able to localize effectively using the method described in the previous section. However, there are more interesting objects in the field. Especially with the current increase in team size, coordination and keeping track of the opponents players becomes an important issue, hindered by the fact that an agent can only pay attention to a small part of the field at the same time. Therefore we search for active vision strategies which optimize the amount of useful information gathered by the agent. To do this we will use the infotaxis strategy which locally maximizes the expected rate of information gain[5]. The information gain resulting from an observation can be measured by the decrease of the entropy $H(f)$ of the distribution $f(\mathbf{x})$. In our case of multivariate normal distribution we have:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{N/2}|\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{x} - \mu)^T \Sigma^{-1}(\mathbf{x} - \mu)\right) \quad (14)$$

$$H(f) = \int_{-\infty}^{\infty} f(\mathbf{x}) \log f(\mathbf{x}) d\mathbf{x} \quad (15)$$

$$= \log\left(\sqrt{(2\pi e)^N |\Sigma|}\right), \quad (16)$$

where N is the number of dimensions, in our case $N = 6$. The problem we need to solve is which action $a \in \mathcal{A}$ of the possible actions \mathcal{A} to take to maximize the decrease in entropy:

$$a_k = \arg \max_a E[\Delta H(f)|a] \quad (17)$$

$$= \arg \min_a E[H(f)_{k+1} H(f)_k | a] \quad (18)$$

$$= \arg \min_a E[H(f)_{k+1} | a] \quad (19)$$

$$= \arg \min_a E[\log\left(\sqrt{(2\pi e)^N |\Sigma_{k+1|k+1}|}\right) | a] \quad (20)$$

$$= \arg \min_a E[|\Sigma_{k+1|k+1}| | a] \quad (21)$$

$$= \arg \min_a E[\mathbf{R}_{\mathbf{k}+1|\mathbf{a}} | a], \quad (22)$$

where $E[X|y]$ denotes the expected value of X given y . This basically means that the agent should take the action that results in an observation with the highest accuracy, which can be estimated with the believe of current state.

4.2 Ongoing Research

There are several ways to continue from here and multiple problems we are or will be researching. Firstly, the choice of set of actions A is important to get the best results. If for instance it consists of turn head n degrees left/right the agent may focus on a single set of landmarks, unwilling to sweep over empty areas, even though that may lead to observing better landmarks. Secondly, the current model of infotaxis assumes that all information is equally valuable. However, in football this is not the case; information about the location of the ball may be worth more than where your keeper is. Moreover, the relative value of different types of information could change during a game. A trade-off has to be made to decide on which target to focus, e.g. by limiting A to actions that will not lose sight of the ball or by alternating between the targets based on the current value of the information about each to the agent. To optimize the latter case we will use another information theoretical principle, relevant information, which measures the amount of information present in a random variable that is relevant for an agent's optimal strategy [4]. This amount gives an indication which variable should get more attention.

5 Team Coordination

With growing team sizes, the importance of the coordination of the agents' positions increases. Due to its simplicity we have opted for a centralised system with the keeper responsible for coordinating the remainder of the team.

Initially, each agent is assigned a position according to his uniform number. As the game evolves, agents may drift from their assigned position when executing certain tasks, such as attacking the opponents goal. Our system allows the keeper to reassign each agent so that it assumes the position closest to it. This allows agent to swap positions dynamically in response to changing game developments, closing gaps in the formation that would have otherwise been left open until the departed agent returns.

Further work is planned to explore other models of coordination, such as distributed systems, and to develop responsiveness to game situations (e.g. winning/losing, corner-kicks).

References

1. V. Calderaro, V. Galdi, V. Lattarulo, and P. Siano. A new algorithm for steady state load-shedding strategy. In *12th International Conference on Optimization of Electrical and Electronic Equipment (OPTIM)*, pages 48–53, 2010.
2. Kalman, Rudolph, and Emil. A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, 82(Series D):35–45, 1960.
3. J. Morimoto, G. Endo, J. Nakanishi, S. Hyon, G. Cheng, and D. Bentevegna. Modulation of simple sinusoidal patterns by a coupled oscillator model for biped walking. In *Proceedings of the 2006 IEEE international conference on robotics and automation*, pages 1579–1584, 2006.

4. Daniel Polani, Thomas Martinetz, and Jan T. Kim. An information-theoretic approach for the quantification of relevance. In *ECAL '01: Proceedings of the 6th European Conference on Advances in Artificial Life*, pages 704–713, London, UK, 2001. Springer-Verlag.
5. Massimo Vergassola, Emmanuel Villermaux, and Boris I. Shraiman. 'infotaxis' as a strategy for searching without gradients. *Nature*, 445:406–409, 2007.
6. Greg Welch and Gary Bishop. An introduction to the kalman filter. Technical report, Chapel Hill, NC, USA, 1995.