Enhancing Humanoids' Walking Skills through Morphogenesis Evolution Method

Nicolas Jouandeau[1] and Vincent Hugel[2]

[1] LIASD, Paris 8 University, France
n@ai.univ-paris8.fr
[2] Robotics Department, Toulon-Var University, France
vincent.hugel@univ-tln.fr

Abstract. This paper presents an evolution method used to modify the morphology of humanoids to make them more efficient in a specific direction of walking. Starting from the NAO's model used in the 3D Simulation Soccer League, the walking specializations are based on 5 to 8 parameters that are being evolved. A black-box optimization process is run and guided by a decision-making function that defines the outcome of the humanoid evolution process. The simulation results lead to four optimized morphological profiles, each of them specialized for either forward, or lateral, or diagonal walk, or in-place turn respectively. These results could be used to build heterogeneous humanoids inside a team of soccer players.

1 Introduction

The tuning of gait parameters thanks to automatic procedure has been widely studied. Mimicking humans and using machine learning algorithms are the most common ways to tune walking parameters. Starting from a set of parameters, short modifications are applied iteratively to improve the set of parameters, according to a fitness function. Hebbel *et al.* [1] successfully used different Evolution Strategies to design a fast forward walker. Following the process of mutation/selection, they proposed solutions to avoid local optima by only selecting children that differ from their parents and to explore more evolution branches over developing multiple parents. Niehaus *et al.* [2] used Particle Swarm Optimization to design an omni-directional walk. The omni-directional property is synthesized into five walking direction sets. As the gait is modelled with a 14 parameter set, the robot uses parameter values that are defined for the synthesized walking direction. Different speeds result from this approach. Moving left, backward and diagonally forward are equivalent. Compared to these first three directions, moving diagonally backward is slower and moving straight forward is faster. MacAlpine and Stone [3] proposed the use of a uniform-velocity omnidirectional walk. The optimization process is achieved with Covariance Matrix Adaptation Evolution Strategy [4] that is able to adapt the next generation according to previous generations' results. The evaluation of walking parameter sets is guided by goToTarget trials, that consist of a sequence of moves to different targets along different directions. Travelled distance, spent time and number of falls are taken into account to define a reward of the tested 14 parameters. To keep the advantage provided by fast forward walking and accurate positioning in the field, they added a Sprint parameter set and a Positioning parameter set. Results are developed over 5 reward policies that produce different forward/backward/sideways walking speeds. Farchy *et al.* [5] introduced Grounded Simulation Learning to reduce the gap between simulation and real application in humanoid walking optimization. The simulation process includes simulation server modification to fit simulation and application results, although models are slightly different. Simulation tests use the original NAO version and application tests use the last NAO version with longer legs.

The different techniques in the related work listed above lead to significant results on skill design optimization. However most of the optimization techniques depend on the optimization problem itself. As we look for a single uniform optimization technique without any preliminary step of parameters definition, particle swarm optimization techniques were discarded due to the definition of population size and selection operator that clearly influence the optimization effectiveness and thus require more optimization iterations. Previous work on learning methods also proposed to change reward weights over time to reduce the number of optimization iterations. As we study multiple skill optimization with different metrics, learning methods were also discarded. Therefore the main contribution of this paper is to present a unified humanoid enhancement process through an evolution method that is defined as generic, so that it allows enhancing different skills for different moves using different morphological models.

Jouandeau and Hugel [6] introduced an optimization process that is applied to both morphological characteristics and walking parameters [7, 8]. They improved the forward walking speed by tuning 2 morphological and 3 functional parameters using the Confident Local OPtimization [9] (CLOP) process. Two policies were finally proposed to define a best-first agent and a best-average agent. Results showed improvements of the morphological model as the optimization process produced faster humanoid walkers, with more realistic, safe and precise walk. The authors also used the same optimization framework to increase kicking skills of humanoids [10]. For a kick move, they applied a skill optimization process to different subsets of parameters of kicking parameters. Results shows that sequential sub-process optimization can lead to better results.

The developments described in this paper extend the above concept of simultaneous tuning of leg morphology and walking parameters to build humanoids with enhanced walking skills in a specific direction. 5 to 8 parameters are being evolved to optimize 4 different walking specializations, that are forward walk, lateral walk, diagonal walk, and in-place turns.

This paper is organized as follows. Section 2 describes the walking gaits that were used for the proposed approach. Section 3 describes the proposed optimization process according to four desired walking specializations. The process produces simultaneous evolution of morphological parameters and walking parameters to enhance walking skills. Section 4 describes and discusses the obtained results. Section 5 concludes.

2 Walking gaits

In this study walking gaits were based on the Zero Moment Point (ZMP) technique [11] applied to the 3D-LIP model [12] that is represented by a single inverted pendulum with a massless telescopic leg that connects the supporting foot and the Center Of Mass (COM) of the entire robot. The height of the COM is kept constant, and there is no torque between the ground and the supporting foot. The robot is assumed to walk on a horizontal plane, with alternating single support phases. The double support phase is thus instantaneous.

The equations that govern the relationship between the position (x_G, y_G, z_G) of the COM and the ZMP – named P^* – are given by [12]:

$$\ddot{x}_G = \frac{g}{z_G}(x_G - x_{P^*}) \tag{1}$$

$$\ddot{y}_G = \frac{g}{z_G}(y_G - y_{P^*}) \tag{2}$$

where g is the gravity.
In our approach the ZMP is kept fixed for each single support phase. This leads to a hyperbolic shape of the COM trajectory for each step, also called *walking primitive*:

$$\begin{aligned} x_G(t) = {} & (x_G^{i(n)} - x_{P^*})\cosh(t/T_C) \\ & + T_C \dot{x}_G^{i(n)} \sinh(t/T_C) + x_{P^*} \end{aligned} \tag{3}$$

$$\begin{aligned} \dot{x}_G(t) = {} & (x_G^{i(n)} - x_{P^*})\sinh(t/T_C)/T_C \\ & + \dot{x}_G^{i(n)} \cosh(t/T_C) \end{aligned} \tag{4}$$

where n is the step number, $x_G^{i(n)}$ and $\dot{x}_G^{i(n)}$ are respectively the COM initial x-position and the COM initial x-velocity of step n, and $T_C = \sqrt{\frac{z_G}{g}}$. The same holds for $y_G(t)$ and $\dot{y}_G(t)$.

3 Gait and morphological optimizations

This section presents the evolution process proposed to improve displacement capabilities of humanoids. The optimization process modifies the morphological characteristics and the walking parameters. It is applied to 4 typical moves: forward translation, lateral translation, diagonal translation, and rotation, to create players with self-adapted morphologies. This section also details different scoring functions for these moves.

3.1 Evolution process

Using the black box optimizer CLOP, the evolution process is run with a list of input parameters, minimum and maximum values for each parameter and a pick-out function to qualify results. The set $\mathcal{L}$ defines input parameters with their minimum and maximum values. During the evolution process, results are collected in the history set $\mathcal{H}$. For each evolution iteration, a new set of parameter values are chosen from $\mathcal{L}$ according to $\mathcal{H}$. Each evolution iteration is processed over 10 trials, to produce average evaluation values. At the end of each evolution iteration, the fitness function `pickOut` states if the result is better, equivalent or worse than the best known result. Then parameters converge to best evolution values. This evolution process is presented in Alg. 1. It can be applied to different types T that defines the content of all sets.

Algorithm 1 $\texttt{evolution} < T >(n, \mathcal{L}, \texttt{pickOut})$

1: $(\nu', \mathcal{H}) \leftarrow (\emptyset, \emptyset)$
2: **for** $i = 0$ to n **do**
3: $\quad p \leftarrow \texttt{newParams} < T >(\mathcal{H}, \mathcal{L})$
4: $\quad (\nu) \leftarrow \texttt{multipleTrials} < T >(p)$
5: $\quad (\nu', h) \leftarrow \texttt{pickOut} < T >(\nu, \nu')$
6: $\quad \texttt{insert} < T >((p\ , h), \mathcal{H})$
7: **end for**
8: **return** $\texttt{paramsFrom} < T >(\nu')$

The `pickOut` function returns three possible values that correspond to better, equivalent and worse results. At each iteration, a new set of parameters p is chosen. ν' stands for best acceptable results. Each CLOP iteration implies that a new tuple (p,h) is inserted in $\mathcal{H}$. If n is too small, ν' could remain empty, which means that no solution is found over n iterations. As presented in Alg. 1 line `1`, the process is started from scratch. At the end, the best parameters that correspond to the results ν' are returned.

3.2 Evaluation of evolution

A list of constant values are required for the evaluation of the evolution. The $SUCCESS\_RATE$ (that is equal to 0.75) defines the rate of trials achieved without falling. The XY_RATIO (that is equal to 0.1) represents the maximal lateral drift allowed, and YX_RATIO (that is equal to 0.1) the maximal longitudinal drift allowed. The $DIAG\_RATIO$ (that is equal to 1.1) qualifies the maximal drift allowed in a diagonal move between longitudinal and lateral distances travelled. α and β (that are equal to 3 and 1) are introduced to compare averages related to the normal distributions used in the process. γ (that is equal to 0.7) is a ratio to compare standard deviations and to quantify stability.

Our evolution process for humanoids is based on the optimization of 4 types T that correspond to 4 different basic moves : 3 translations and 1 rotation. For

Algorithm 2 `pickOut` $< T >((s,\, m,\, \sigma),\, (s',\, m',\, \sigma'))$

1: **if** $s < SUCCESS\_RATE$ **then return** $REJECT$;
2: **if** $type = FORWARD$ **then**
3:     **if** $m_y/m_x > XY\_RATIO$ **then return** $REJECT$;
4:     **return** `translationPickOut` $(m,\, \sigma_x,\, m',\, \sigma'_x)$;
5: **end if**
6: **if** $type = LATERAL$ **then**
7:     **if** $m_x/m_y > YX\_RATIO$ **then return** $REJECT$;
8:     **return** `translationPickOut` $(m,\, \sigma_y,\, m',\, \sigma'_y)$;
9: **end if**
10: **if** $type = DIAGONAL$ **then**
11:     **if** $m_y/m_x > DIAG\_RATIO$ **or** $m_x/m_y > DIAG\_RATIO$ **then**
12:         **return** $REJECT$;
13:     **end if**
14:     $e \leftarrow$ `sqrt` $(\sigma_x+\sigma_y)$
15:     $e' \leftarrow$ `sqrt` $(\sigma'_x+\sigma'_y)$
16:     **return** `translationPickOut` $(m,\, e,\, m',\, e')$;
17: **end if**
18: **if** $type = ROTATION$ **then**
19:     **if** $(m_x^2 + m_y^2) > MAX$ **then return** $REJECT$;
20:     **if** *optimize_accuracy* **then return** `RPO_accuracy` $(target,\, m,\, \sigma,\, m',\, \sigma')$;
21:     **if** *optimize_time* **then return** `RPO_time` $(m,\, \sigma,\, m',\, \sigma')$;
22: **end if**

each move, $< T >$ defines *type* value (used in Alg. 2 lines 4, 10, 16 and 24) and eventually a *target* value (used in Alg. 2 line 29 specially in an accuracy test). The 3 translations are selected among the 8 discrete walking directions : moving along the longitudinal axis (forward and backward), moving sideways (right or left) and moving diagonally (forward or backward, left or right). Since backward walking patterns are not frequently used, only the 5 forward translational moves, namely left, diagonal left, forward, diagonal right and right, were selected for the process. The rotation move is a self-rotating move on the spot. Each translation-move optimization is based on a single policy. Because the default rotation that was tuned manually is already effective – fast, large steps and no fall –, it appears to be close to its optimum parameter values. This is the reason why the optimization process for rotation moves is associated with two possible policies, the first policy fosters better accuracy, and the second policy fosters reduced execution time.

The main core of the evaluation function is presented in Alg. 2. Inside each set ν from Alg. 1 :

- the subset m defines average values of longitudinal step length, lateral step length, turning step angle and execution time.
- the subset σ defines standard deviations values related to the average values.
- the subset s defines the success rate of the experiments.

Therefore, ν (respect. ν') set in Alg. 1 is replaced with its subsets (s,m,σ) (respect. (s',m',σ') in Alg .2).

The evaluation function is used for all moves, calling a more specialized sub-function if preliminary tests are passed successfully. First of all, as tested in Alg. 2 line `1`, a miminum success rate s is needed. Therefore, depending on the type of move, the function checks :

- The lateral drift while translating forward, in Alg. 2 line `3`.
- The longitudinal drift while translating sideways, in Alg. 2 line `7`.
- The shift between translation axes while translating in diagonal, in Alg. 2 line `11`.
- The drift while self rotating on the spot, in Alg. 2 line `19`.

For the evaluation of translations the same `translationPickOut` function is called with the same average results – named m and m' – but with different criteria:

- Translating forward uses σ_x and σ'_x that define the standard deviations of forward translation on the x-axis, in Alg. 2 line `4`.
- Translating sideways uses σ_y and σ'_y that define the standard deviations of sideways translation on the y-axis, in Alg. 2 line `8`.
- Translating in diagonal uses distances – named e and e' – that define the standard deviations of the achieved distance, in Alg. 2 line `16`.

As rotation is already very effective, the optimization of rotation moves in Alg. 2 is associated with two possible policies, the one fostering accuracy (line `20`) or the other one reduced execution time (line `21`). Accuracy optimization is achieved according to a specific *target* that defines the desired rotation angle.

Details of the respective evaluation functions are explained in the next section.

3.3 Specialized evaluations

Specialized evaluations regroup: The translation evaluation function, detailed in Alg. 3 ; The rotation evaluation function that checks the resulting accuracy, detailed in Alg. 4 ; The rotation evaluation function that checks the resulting speed, detailed in Alg. 5.

Algorithm 3 `translationPickOut` (m, e, m', e')

if $m' == UNDEFINED$ **then return** $ACCEPT$;
$d \leftarrow$ `sqrt` $(\ m_x^2 + m_y^2\ )$;
$d' \leftarrow$ `sqrt` $(\ m_x'^2 + m_y'^2\ )$;
if $d < d' - \alpha e'$ **then return** $REJECT$;
if $d < d' - \beta e'$ **then return** $EQUIVALENT$;
if $e < \gamma e'$ **then return** $ACCEPT$;
if $d < d'$ **then return** $EQUIVALENT$;
return $ACCEPT$;

All these evaluation functions make use of the three constant values α, β and γ: α is the nearness factor: if the new result is not close enough to the best known result, parameters are considered to lead to a worst instead of a best known result ; β is the equivalence factor: the test is similar to the nearness factor test with a different threshold. The result is now $EQUIVALENT$ with this factor whereas it is $REJECT$ with α. Because we compare averages and standard deviations resulting from two experiments, it is logical to ensure that $\beta < \alpha$; γ is the width factor: if the new result is more stable, then it is better.

Algorithm 4 `RPO_accuracy` ($target$, m, σ, m', σ')

1: **if** $m' == UNDEFINED$ **then return** $ACCEPT$;
2: **if** $\mid m_\theta - target \mid > \mid m'_\theta - target \mid + \alpha\sigma'_\theta$ **then return** $REJECT$;
3: **if** $\mid m_\theta - target \mid < \mid m'_\theta - target \mid - \beta\sigma'_\theta$ **then return** $EQUIVALENT$;
4: **if** $\sigma_\theta < \gamma\sigma'_\theta$ **then return** $ACCEPT$;
5: **if** $\mid m_\theta - target \mid > \mid m'_\theta - target \mid$ **then return** $EQUIVALENT$;
6: **return** $ACCEPT$;

As these functions are iteratively used, the constant values contribute to the convergence speed of the evolution.

Algorithm 5 `RPO_time` (m, σ, m', σ')

1: **if** $m' == UNDEFINED$ **then return** $ACCEPT$;
2: **if** $m_{time} < m'_{time} - \alpha\sigma'_{time}$ **then return** $REJECT$;
3: **if** $m_{time} < m'_{time} - \beta\sigma'_{time}$ **then return** $EQUIVALENT$;
4: **if** $\sigma_{time} < \gamma\sigma'_{time}$ **then return** $ACCEPT$;
5: **if** $m_{time} < m'_{time}$ **then return** $EQUIVALENT$;
6: **return** $ACCEPT$;

3.4 Parameters influence and trials

The optimization process can be seen as a nature-inspired growth since it makes morphological parameters of the legs and locomotion parameters evolve simultaneously. The first column of Tab. 1 and Tab.2 contains the morphological leg parameters and the walking parameters with upper/lower bounds.

The morphological leg parameters are listed first in these two tables. These parameters are related to the morphology of the leg:

- $ThighRelHip2\_Z$ stands for the semi-length of the femur. The change of this parameter value changes the crural index of the leg, which is the ratio of the tibia length with the femur length. The crural index is one of the key parameters in human morphology since it is useful for the comparison of the

different bipeds that colonized the Earth since the appearance of the first hominids. The tibia length is kept fixed.
- $Hip1RelTorso\_X$ is the half length between hips. This parameter can be tuned to build a larger or a narrower pelvis for the humanoid robot. A larger pelvis can increase the reachable space of the legs below the trunk and reduce collisions between legs. This parameter is expected to influence the quality of the walking patterns that involve sideways moves.
- $ratio\_flexion$ is the leg flexion ratio, which is defined as the ratio of the hip height from the ground over the total length of the leg when stretched. A change of the flexion ratio has an influence on the way the robot walks, *i.e* with knees more or less flexed.

The walking parameters are listed below the morphological parameters in Tab. 1 and Tab.2. The walking parameters can be varied to tune the walking skills of the robot in order to get a quick and well-balanced gait:

- $offset\_MidAnkles\_2\_Torso\_I$ stands for the horizontal distance between the middle of the ankles and the torso center. This parameter allows to balance the weight of the torso with respect to the flexed legs. The COM is considered to be fixed with respect to the torso, and its coordinates inside the torso coordinate frame are calculated automatically in the standing position as a function of the morphological parameters. This is an usual approximation in the case of the LIP-3D model.
- $height\_lift$ is the maximal height of leg lift-off.
- $xlength\_step\_max$ is the maximal forward step length.
- $ylength\_step\_max$ is the maximal sideways step length.
- $theta\_step\_max$ is the maximal turning step angle.
- $dist\_between\_feet\_p\_points$ is the distance between ankles in the rest position. This distance can be adjusted for the robot to walk with the feet more or less apart from each other. This walking parameter is expected to be influenced by the pelvis size.

4 Experiments and results

The simulation software is composed of 5 different parts, i.e. *rcssserver3d* [13, 14], the client agent *rcssagent3d-like*, a coach (that is responsible for starting trial), the CLOP framework [9] and utilities that link everything.

Table 1 indicates the parameters that were used for each walking gait optimization. Table 2 contains the optimized parameters resulting from each of the 5 experiments (that were run according to the 5 evaluation functions) :

- Moving straight ahead, called *Fwd.* trial.
- Moving sideways , called *Lat.* trial.
- Moving diagonally forward, called *Diag.* trial.
- Rotating accurately, called *Rot.* trial with *Opt.1 on Accuracy* parameters.
- Rotating fast, called *Rot.* trial with *Opt.1 on Time* parameters.

Table 3 contains the evaluation values, namely s, m_{time}, m_x, m_y, m_θ and the related deviations σ_x, σ_y and σ_θ for each experiment.

Tables 2 and 3 recall the default values of the parameters before optimization. These default values are related to the walking gaits that were tuned manually using expert's knowledge. The default values are useful to be compared to the optimized values obtained.

All experiments change morphological parameters and technical skills simultaneously to fit morphology and walking parameters to maximize values while minimizing other shifting values. As we aim at building a best-first agent, we only use the *Opt.1* policy [6], that defines the best last trial of the evaluation function. Each optimization is achieved with 500 CLOP iterations. Each CLOP iteration is performed over 10 trials. Final results are also calculated over 10 trials.

According to the results listed in Tab. 3 :

- While moving forward straight ahead : *Fwd. Opt.1* morphology is 1.62 times faster than the *Dflt* one (from ratio of m_x values in Tab. 3). *Fwd. Opt.1* morphology is 1.29 times more stable than the *Dflt* one (see parameter s in Tab. 3).
- While rotating : *Rot. Opt.1 on Accuracy* morphology is 3.44 times slower than the *Dflt* one (from ratio of m_{time} values in Tab. 3). *Rot. Opt.1 on Time* morphology is equivalent to the *Dflt* one (similar values for m_{time} in Tab. 3).
- While moving sideways : *Lat. Opt.1* morphology is 1.78 times faster than the *Dflt* one (from ratio of m_y values in Tab. 3).
- While moving on diagonal : *Diag. Opt.1* morphology is 1.51 times faster than the *Dflt* one (from ratio of m_x and m_y values in Tab. 3).

Table 1. All parameters bounds and trial policies

Bounds and trials	Min	Max	Fwd.	Rot.	Lat.	Diag.
Morphological parameters :						
$ThighRelHip2\_Z$	-0.09	-0.02	X	X	X	X
$Hip1RelTorso\_X$	-0.01	-0.10	X	X	X	X
$ratio\_flexion$	0.60	0.95	X	X	X	X
Walking skills parameters :						
$offset\_MidAnkles\_2\_Torso\_I$	0.001	0.030	X	X	X	X
$height\_lift$	0.025	0.080	X	X	X	X
$xlength\_step\_max$	0.020	0.150	X			X
$ylength\_step\_max$	0.020	0.150			X	X
$theta\_step\_max$	0.020	0.785		X		
$dist\_between\_feet\_p\_points$	0.020	0.200		X	X	X

Table 3 shows that the new two rotation morphologies (*Rot. Opt.1 on Accuracy* and *Rot. Opt.1 on Time*) are not better than the *Dflt* one. This results is not surprising because the rotation gait was carefully designed in the original morphology as explained in section 2, and because the rotation gait is less sensitive to dynamical effects along the longitudinal direction that is more prone to

Table 2. Resulting parameter values

Trial		Fwd.	Rot.		Lat.	Diag.
Parameters	Dflt	Opt.1	Opt.1 on Accuracy	Opt.1 on Time	Opt.1	Opt.1
Morphological parameters :						
$ThighRelHip2\_Z$	-0.040	-0.079	-0.082	0.080	-0.061	-0.065
$Hip1RelTorso\_X$	-0.055	-0.022	-0.096	-0.058	-0.083	-0.062
$ratio\_flexion$	0.728	0.902	0.657	0.775	0.809	0.857
Walking skills parameters :						
$offset\_MidAnkles\_2\_Torso\_I$	0.011	0.020	0.006	0.024	0.013	0.009
$height\_lift$	0.030	0.063	0.066	0.069	0.044	0.036
$xlength\_step\_max$	0.080	0.125				0.050
$ylength\_step\_max$	0.060				0.121	0.094
$theta\_step\_max$	1.047		0.205	0.757		
$dist\_between\_feet\_p\_points$	0.110		0.079	0.027	0.123	0.146

Table 3. Results for each trial

Trial	Fwd.		Rot.			Lat.		Diag.	
Param.	Dflt	Opt.1	Dflt	Opt.1 on Accuracy	Opt.1 on Time	Dflt	Opt.1	Dflt	Opt.1
s	0.70	0.90	1.00	1.00	1.00	1.00	1.00	1.00	1.00
m_{time}			2.807	9.643	3.058				
m_x	4.422	**7.168**	0.018	0.023	0.012	0.084	0.078	1.840	**2.843**
m_y	0.304	0.553	0.041	0.053	0.025	2.021	**3.588**	1.942	**2.860**
m_θ	0.123	0.187	5.973	6.161	6.030	0.062	0.060	0.083	0.248
σ_x	0.046	0.075	0.010	0.023	0.015	0.031	0.048	0.064	0.250
σ_y	0.234	0.534	0.032	0.036	0.015	0.053	0.046	0.106	0.287
σ_θ	0.066	0.173	0.068	0.125	0.159	0.021	0.048	0.047	0.131

Table 4. Results of 3 parameter sets for each trial

Trials	Fwd.				Lat.				Diag.			
Param.	Fwd.	Rot.*	Lat.	Dgn.	Fwd.	Rot.*	Lat.	Dgn.	Fwd.	Rot.*	Lat.	Dgn.
s	0.90	1.00	1.00	1.00	1.00	1.00	1.00		1.00	1.00	1.00	1.00
m_{time}		1.339				1.380				1.398		
m_x	**7.168**	0.035	0.428	1.091	4.737	0.045	0.078	2.423	2.918	0.026	0.418	**2.843**
m_y	0.553	0.042	1.961	2.465	0.540	0.049	**3.588**	2.634	0.201	0.053	3.023	**2.860**
m_θ	0.187	2.232	0.397	0.747	0.270	1.789	0.060	0.119	0.112	2.182	0.237	0.248
σ_{time}		0.001				0.003				0.001		
σ_x	0.075	0.020	0.071	0.046	0.094	0.014	0.048	0.155	0.033	0.015	0.144	0.250
σ_y	0.534	0.033	0.067	0.053	0.407	0.029	0.046	0.158	0.074	0.039	0.054	0.287
σ_θ	0.173	0.003	0.052	0.028	0.236	0.010	0.048	0.072	0.064	0.004	0.009	0.131

falling. Therefore none of these new rotation morphologies were selected. the heterogeneous team. The three remaining morphologies (Fig. 1) present interesting properties because they enhance the displacement capabilities, either forward, sideways or diagonally (see m_x and m_y values in bold font in Tab. 3).

However it is necessary to check that each of the three selected morphologies is still compatible with the walking gaits in the other directions, i.e. other than the *enhanced* direction. Table 4 summarizes such results. It shows that the three optimized morphologies are compatible with the gaits in the other directions. In the checking experiments the angle in the rotation gait was limited to $2\pi/3$. In addition since the lateral morphology increased the lateral step up to the maximal bound, it was necessary to limit the diagonal step to maintain the foot

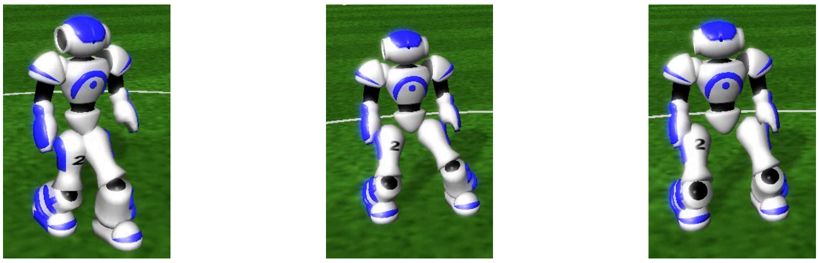

Fig. 1. *Fwd.*, *Diag.* and *Lateral* morphologies.

trajectory inside the working volume of the leg. Therefore the diagonal step was limited to $0.08m$.

If we observe the values of the morphological parameters listed in Tab. 2, we can notice that the *Lat.* and *Diag.* morphologies have a wider pelvis compared to the *Dflt* morphology and the *Fwd. Opt.1* morphology (see $Hip1RelTorso\_X$ parameter). The pelvis is even wider in the *Lat.* morphology to enable a larger sideways step, i.e. $0.121m$ compared to $0.94m$ with the *Diag.* values. In addition we can notice that all three morphologies have longer thighs (see $ThighRelHip2\_Z$ parameter), and that the flexion ratio is also larger, which means that all new robots will walk in a more high-legged way. This way of walking reminds the human walk where legs get stretched and flexed alternately. The third observation concerns the height of foot lift off. This height is reduced for the *Lat.* and the *Diag.* morphologies. Actually it was noticed that the lateral walk was more sensitive to lift-off height. This is due to ground impacts of the swinging leg that cause oscillations of the torso in the frontal plane, and these oscillations can be dangerous and make the robot fall if the amplitude increases too much. The reduction of the lift-off height is useful to prevent leg impacts from triggering undesired oscillations.

Thanks to this study it is possible to build a new team with heterogeneous players where:

- Strikers built on the *Fwd.* parameter set could be both faster sprinters and nicely reactive due to their fast rotation speed.
- Midfields built on the *Diag.* parameter set could be good at breaking opponents attack.
- Defenders and goalie built on the *Lat.* parameter set) could be good at intercepting the ball or the opponent trajectory.

The next developments will aim at testing teams of heterogeneous soccer players during soccer game plays.

5 Conclusion

We introduced an optimization process that is especially designed for the simultaneous evolution of humanoids' morphological characteristics and walking parameters. The optimization process is essentially guided by a fitness function that distinguishes among better, equivalent and worse results. Three morphological profiles have been produced to create three agents that appear to be more effective than the previous agents that were tuned manually by expert users. Actually the process leads to morphologies well suited for forward walk, lateral walk and diagonal walk. This process will be applied to the building of a heterogeneous humanoid team with new models designed according to specific skills.

References

1. M. Hebbel, R. Kosse and W. Nistico, *Modeling and Learning Walking Gaits of Biped Robots*, In IEEE-RAS Int. Conf. on Humanoid Robots 2013, 1st Workshop on Humanoid Soccer Robots (HSR2006).
2. C. Niehaus, T. Röfer and T. Laue, *Gait Optimization on a Humanoid Robot using Particle Swarm Optimization*, In Second Workshop on Humanoid Soccer Robots, IEEE-RAS 7th Int. Conf. on Humanoid Robots, 2007, (HSR2007).
3. P. MacAlpine and P. Stone, *Using Dynamic Rewards to Learn a Fully Holonomic Bipedal Walk*, In Adaptive Learning Agents Workshop, 2012, (ALA 2012).
4. N. Hansen, *The CMA Evolution Strategy: A Tutorial*, `https://www.lri.fr/~hansen/cmatutorial.pdf`, 2011.
5. A. Farchy, S. Barrett, P. MacAlpine and P. Stone, *Humanoid Robots Learning to Walk Faster: From the Real World to Simulation and Back*, In Int. Conf. on Autonomous Agents and Multiagent Systems, 2013, (AAMAS 2013).
6. N. Jouandeau and V. Hugel, *Simultaneous evolution of leg morphology and walking skills to build the best humanoid walker*, In IEEE-RAS Int. Conf. on Humanoid Robots 2013, 8th Workshop on Humanoid Soccer Robots (HSR2013).
7. V. Hugel and N. Jouandeau, *Walking Patterns for Real Time Path Planning Simulation of Humanoids*, In 21st IEEE Int. Symposium on Robot and Human Interactive Communication, (IEEE RO-MAN 2012).
8. V. Hugel and N. Jouandeau, *Automatic generation of humanoid s geometric model parameters*, In 17th annual RoboCup Int. Symposium 2013 (RCUP-2013).
9. R. Coulom, *CLOP: Confident Local Optimization for Noisy Black-Box Parameter Tuning*, In Advanced Computer Games (ACG), volume 7168 of Lecture Notes in Computer Science, page 146-157. Springer, 2011.
10. N. Jouandeau and V. Hugel, *Optimization of Parametrised Kicking Motion for Humanoid Soccer Player*, In IEEE-RAS Int. Conf. on Autonomous Robot Systems and Competitions 2014 (ICARSC2014).
11. M. Vukobratovic, B. Borovac, *Zero-moment point - thirty five years of its life*, in: Int. J. of Humanoid Robotics, Vol. 1(1), 2004, pp. 157–173.
12. S. Kajita, *Humanoid Robot*, Ohmsha Ltd, 3-1 Kanda Nishikicho, Chiyodaku, Tokyo, Japan, 2005.
13. O. Obst and M. Rollmann, *Spark - A Generic Simulator for Physical Multi-Agent Simulations*, In Multiagent System Technologies, Lecture Notes in Computer Science, vol. 3187, pp. 243-257, 2004.
14. *SimSpark, a generic physical multiagent simulator system for agents in three-dimensional environments*, `http://simspark.sourceforge.net/`.