Optimizing the Gait of a Humanoid Robot Towards Human-like Walking

Sven Wehner Maren Bennewitz
Department of Computer Science, University of Freiburg, Germany

***Abstract*— Achieving a stable, human-like gait with a humanoid robot is a challenging problem. While several rather simple as well as more complex techniques exist to generate stable walking patterns, only little attention has been paid towards the resemblance to the human gait. Popular gaits, for example, apply the strategy to bend the knees and to swing the torso in lateral direction in order to ensure stability. As a result, the walking patterns do not look very human-like. However, human resemblance is an important aspect whenever robots are designed to interact naturally with humans. In this paper, we present techniques to optimize an initial, stable gait of a humanoid robot with respect to human resemblance. To acquire walking data of a human, we use a full-body motion capture system. We propose four different optimizing algorithms that work at joint angle basis and use the joint angle difference as measure of similarity. Experimental results carried out with a *HOAP-2* robot in simulation demonstrate that we can adapt the robot's initial gait so that it is significantly more human-like.**

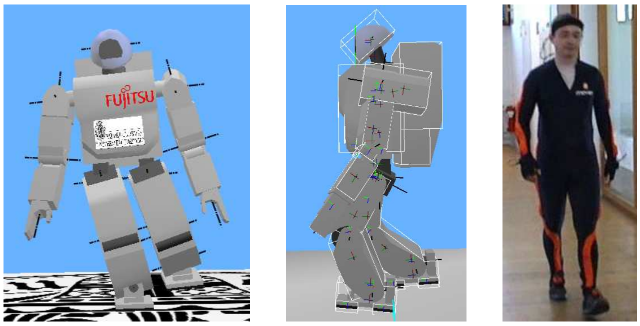

Fig. 1. The left and middle image show snapshots of the robot's initial gait. As can be seen, the torso of the robot moves heavily in lateral direction and the knees are bent which is not human-like. In the right image, a human equipped with the full-body motion capture system *MVN* is walking for the acquisition of human gait data.

I. Introduction

Recently, humanoid robots have become quite popular and are used as a research tool in many groups worldwide. One of the motivation behind the research area of humanoid robotics is to develop robots that are better adapted to environments designed for humans and that these robots are capable of natural interaction with humans.

Compared to wheeled robots, one challenge when working with humanoid robots is to design stable walking gaits for biped navigation. A frequently used approach to enabling humanoid robots to walk stably is to apply heuristics and to manually configure the walking patterns and their parameters. There exist techniques based on central pattern generators (CPGs) to generate joint trajectories using nonlinear oscillators [7, 1]. In these approaches, it is a challenging problem to find appropriate parameters to achieve a stable gait. More computational demanding methods use the concept of the zero moment point (ZMP) [22] and rely on joint angle trajectories which are computed considering dynamic motion of the robot. Here, an accurate model of the robot and its dynamics is needed.

Several approaches have been presented that aim at optimizing properties such as speed [8, 18, 13] or torso stability [14, 4] of a humanoid's walk. The resulting, optimized walking patterns often do not really resemble human gait. However, the resemblance to the human ideal should be taken into account when generating walking patterns for humanoid robots. These robots are designed to interact naturally with humans and therefore it is important that their motions look human-like.

The first two images of Fig. 1 shows snapshots of a typical humanoid's gait. First, the torso of the robot swings heavily in lateral direction and, second, the knees are bent. These concepts are often used in humanoid navigation to achieve stability. However, the resulting gait does not look very natural.

In this paper, we consider the problem of achieving a stable, human-like gait with a humanoid robot. We treat this as an optimization problem and develop four algorithms that work on joint angle basis. Our optimization starts from an initial, stable gait of the robot obtained via a CPG and uses as target gait the walk of a human recorded by a motion capture system (see right image of Fig. 1). Similarity between gaits is defined in terms of joint angle difference between the human's and the robot's joint angle trajectories. The advantage of working solely on the basis of joint angles is that we do not need to incorporate expert knowledge into the learning process, e.g., in form of a parameterized gait [14] or in form of a segmentation into different walking phases [19].

We extensively evaluated and empirically compared the different optimization techniques in experiments carried out in the *Webots* simulator [5] with a *HOAP-2* [10] robot. The results show that the optimization methods based on hill climbing and on policy gradient estimation perform best. However, all techniques are able to improve the initial gait so that it is significantly more similar to the human gait.

The paper is organized as follow. After discussing related work in the next section, we present the humanoid robot used for the experiments in Sec. III and describe the collection and preprocessing of human gait data in Sec. IV. In Sec. V, we introduce the algorithms we developed to optimize the robot's walk. Finally, we discuss the experimental results in Sec. VI.

II. RELATED WORK

Several approaches exist to optimize the walking speed of a humanoid robot for a given, parameterized gait. For example, Faber and Behnke [8] applied an optimization based on policy gradient reinforcement learning and particle swarm optimization to increase the forward speed of their humanoid. The authors used eight parameters of the gait and developed two feedback mechanisms which were included into the optimization process. Niehaus *et al.* [18] also applied particle swarm optimization to speed up the performance of a humanoid. They considered 14 parameters and performed an optimization of the parameters for different walking directions to allow for omnidirectional walking. Hemker *et al.* [13] applied sequential surrogate optimization to searching for optimal values of the five chosen gait parameters of forward walking. Geng [11] *et al.* proposed a policy gradient reinforcement learning approach to optimize the parameters of a neuronal sensor-driven controller for a planar biped. Furthermore, several researches applied machine learning techniques to optimize the gait of quadruped robots (e.g. [17, 15]). Note that stability is not such a serious problem with quadruped robots which makes optimization easier. In contrast to all techniques presented above, our goal is not to optimize the speed of a humanoid robot which often leads to rather unnatural looking walking behaviors. Instead, we aim at generating a gait that looks more human-like than the initial walking behavior of the robot.

Chalodhorn *et al.* [4] used an imitation-based approach to teach a humanoid robot how to walk stably. Human data was recorded with an optical motion capture system and the search for appropriate actions of the robot leading to a stable gait was performed in a dimensionality-reduced space of the joint angles. In order to achieve a stable gait of the robot, the authors included an optimization of torso stability defined by using gyroscope signals. From the presented images, it seems that the human demonstrator moved in a rather unnatural way in order to facilitate the learning of the robot. Huang *et al.* [14] analyzed characteristics of human gait in terms of change of given walking parameters when choosing different step lengths and step cycles. They also used human motion data captured by an optical system. Considering the learned characteristics, parameters for a walking pattern were determined which resulted in a high upper-body stability of the humanoid. Serhan *et al.* [19] proposed to extract “critical angles” from human locomotion that influence the speed and step length. They used a segmentation of the walking cycle in eight phases and considered maximum angles for the different phases. The authors present experiments in which a simulated biped robot with only a small 4 DOF trunk achieved a dynamic walk. To generate a human-like walking behavior that can be adapted according to observations, e.g, barriers or stairs, Denk and Schmidt [6] propose to concatenate previously learned walking primitives.

Regarding other tasks than walking, several researchers have concentrated on generating motions for humanoids that are optimal with respect to specific criteria. For example, Harada *et al.* [12] optimized motion primitives for a humanoid robot in terms of joint torque, acceleration, or angular momentum.

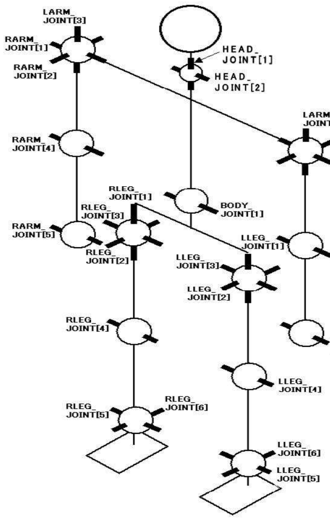

Fig. 2. An overview over the robot’s joints. There are six DOFs in each leg, five in each arm, two in the head and one in the hip. The bold lines represent the rotation axes of the joint. Illustration taken from [10].

The authors identified relevant variables for the individual tasks (e.g., a reaching motion, walking, or climbing). Faber *et al.* [9] proposed a strategy for gaze control of a humanoid during human-robot interaction. They consider different factors, i.e., tracking error, discomfort, and effort to control the pitch and yaw joints. Bobrow *et al.* [2] optimized the motion of robots performing different tasks with respect to minimum control effort. Svinin *et al.* [21] considered the problem of generating human-like reaching movements with a robotic arm. The authors use an objective function which is based on the minimization of hand jerk.

In the remainder of this paper, we present our approach to optimizing the walking motion of humanoid with respect to human resemblance. Our optimization works on joint angle basis and does not need to incorporate expert knowledge.

III. THE HUMANOID ROBOT

The humanoid robot, which is used in our experiments, is a simulated *HOAP-2* from *Fujitsu* [10]. The first two images of Fig. 1 show the simulated robot. The robot has a weight of 7 kg and is 50 cm tall. The total number of degrees of freedom (DOFs) is 25, but only 21 are considered as relevant for walking (excluding head and hand joints). Fig. 2 gives an detailed overview over the robot’s degrees of freedom.

There are some differences between the human anatomy and the robot’s one which can lead to different behavior. One of the major issues is the fact, that the robot wears a “backpack” (in which the processing unit is located), which shifts the robot’s center of mass. Furthermore, certain joints are not modeled in the robot, e.g., the rotation around the body’s yaw axis.

We use the model of the *HOAP-2* shipped with the *Webots* simulator. The walking motion, which we take as a basis for our experiments, was generated using a CPG. In this gait, the robot shows a lateral swing and walks with strongly bent knees (see Fig. 1).

The given simulation model provides only a single pressure sensor in each foot. We extended the simulated model so that it has a sensor in the front and another one in the back of each foot. We need this extension for our approach of stability estimation as we explain later.

IV. Data Acquisition

In this section, we describe how we acquire the joint angle trajectory data which is used as input to our optimization framework.

A. Recording Human Gait Data

For recording full-body motions of a human, we used the *Xsens MVN* [3] system. It uses sixteen sensors, which include accelerometers, gyroscopes, and magnetometers. The right picture in Fig. 1 shows the human demonstrator wearing the *MVN* body suit. The output of the system includes information about the position and rotation of the body segments. Since we use a joint angle based representation, we transform the rotation information into joint angles.

B. Data Preprocessing

After recording the data, we perform some data preprocessing steps in order to get the data which serves as input to our optimization framework.

1) Segmentation: We first extract a sequence of double-steps of the robot's and of the human's gait. We use the trajectories of both knees to find the exact start and end of segments containing five double-steps, which were sufficient for the analysis.

2) Uniform Trajectory Lengths: Extracted motion sequences containing five double-steps can have different lengths, e.g., a different number of data points. Therefore, we perform a linear interpolation to estimate data points and to achieve a uniform trajectory length (which corresponds to the length of the robot's trajectory).

3) Removing Temporal Distortion: Obviously, the captured human data contains some noise and temporal distortion. To reduce the influence of these effects, we compute the average over several human gait sequences. In particular, we use *Dynamic Time Warping* (*DTW*) [16]. The main idea of DTW is to match two sequences by warping the temporal position of data points in a way that the lowest overall distance between both is found. The distance of two joint angle trajectories $\boldsymbol{M}$ and $\boldsymbol{N}$ is defined as

$$\operatorname{dist}(\boldsymbol{M}, \boldsymbol{N}) = \sum_{\text{joint } j} \sum_{\text{time } t} \left(M_{j,t} - N_{j,t}\right)^2 . \tag{1}$$

The DTW algorithm finds the optimal matching between data points and minimizes the overall distance of the sequences. We then compute the average joint angle trajectories given the resulting time indices. For our experiments, we used eight sequences of five double-steps on which we perform DTW in a tree-like manner.

4) Symmetry: The trajectories of corresponding lateral joints (e.g., the left and the right knee) of the human appeared to slightly differ. Such small asymmetries can result in instable behavior of the robot. We therefore compute the joint angle trajectories as the average over the corresponding data points of the left and the right joints.

5) Uniform Gait Cycles: In the last preprocessing step, we eliminate irregularities within the different gait cycles of the given human's trajectories. We apply a Fourier transform to first get an amplitude representation. We call a frequency f *symmetric* if and only if

$$f \mod \#\text{double-steps} = 0. \tag{2}$$

Otherwise, it is called *asymmetric*. We then set the amplitudes of the asymmetric frequencies to zero since they correspond to irregularities in the different steps. Finally, we perform an inverse Fourier transform from the amplitude spectrum based representation into the data points representation one.

V. Gait Optimization

The input to the optimization process are the robot's and the human's joint angle trajectories which are computed as described in the previous section. The goal of the optimization is to adapt the joint angle trajectories of the robot towards the one of the human. In an iterative fashion, we adapt the individual joint angle trajectories over the entire walking sequence towards the trajectories of the human walk.

The change of a joint angle trajectory at a certain time step is proportional to the difference between the robot's and the human's trajectory. For each time step t, the new angle of an individual joint i is computed proportional to the difference between the robot's original joint angle $\theta_{i,t}^{rob}$ and the human's joint angle $\theta_{i,t}^{hum}$. Using the adaption factor $0 \leq \alpha_i \leq 1$, the new angle $\hat{\theta}_i^{rob}$ for joint i is computed at all time steps as

$$\hat{\theta}_{i,t}^{rob} \leftarrow \theta_{i,t}^{rob} + \alpha_i(\theta_{i,t}^{hum} - \theta_{i,t}^{rob}). \tag{3}$$

Our optimization methods presented in the following aim at adapting α_i for each joint i so that a stable walk is achieved which is as similar as possible to the human's gait. We use a similarity measure which is based on the distance (see Eq. 1) to compute the similarity between the robot's current gait $\hat{\boldsymbol{M}}_{\boldsymbol{rob}}$ and the human's gait $\boldsymbol{M_{hum}}$ given the robot's initial gait $\boldsymbol{M_{rob}}$:

$$\operatorname{similarity}(\hat{\boldsymbol{M}}_{\boldsymbol{rob}}, \boldsymbol{M_{hum}}) = \frac{\operatorname{dist}(\boldsymbol{M_{rob}}, \hat{\boldsymbol{M}}_{\boldsymbol{rob}})}{\operatorname{dist}(\boldsymbol{M_{rob}}, \boldsymbol{M_{hum}})} \tag{4}$$

The stopping criterion for the optimization is that the gain in the similarity is below a certain threshold.

According to the DOFs of the *HOAP-2*, the search space is 11-dimensional since symmetric joints are treated alike and the hand as well as the two head joints are not considered.

A. Single Component Sampling

First, we considered the simple approach of sampling one joint i in each iteration whose adaption factor α_i is increased according to the change rate $\Delta\alpha_i > 0$:

$$\alpha_i \leftarrow \alpha_i + \Delta\alpha_i \tag{5}$$

When the gait resulting from the change of the adaption factor leads to a stable gait, we keep the increased adaption factor. If not, we refuse the change. If a certain number of attempts (i.e., in different iterations) to increase the adaption factor of joint i fail to generate a stable gait, we decrease the change rate $\Delta\alpha_i$.

Algorithm 1 Sampling with correlation-based Optimization. $correlationBasedOptimization(\boldsymbol{M}^{rob}, \boldsymbol{M}^{hum}, \Delta\boldsymbol{\alpha})$

Input: Joint angle trajectories of the robot's gait $\boldsymbol{M}^{rob}$ and of the human's gait $\boldsymbol{M}^{hum}$, change rate of the adaption factors for the N individual joint angles $\Delta\boldsymbol{\alpha} = \langle \Delta\alpha_1, \cdots, \Delta\alpha_N \rangle$.
Output: Joint angle trajectories $\boldsymbol{M}^{rob}_{opt}$ of the most human-like, stable gait found.

```
M^rob_opt ← M^rob; α ← 0;
Cor^hum ← computeCorrelation(M^hum)
Cor^rob ← computeCorrelation(M^rob)
while not converged do
  r ← random(1, ..., N)
  for all i ← 1 ... N do
    c ← (1 − α_i) · |Cor^rob_{r,i}| + α_i · |Cor^hum_{r,i}|
    α̂_i ← α_i + Δα_i · c
  end for
  M̂^rob ← generateJointAngleTrj(M^rob, M^hum, α̂)
  if evaluate(M̂^rob) = stable then
    M^rob_opt ← M̂^rob; α ← α̂
  end if
end while
return M^rob_opt.
```

B. Sampling with Correlation-based Optimization

The next approach considers the correlation between the trajectories of the individual joints and allows for changes of all joints simultaneously. The idea is to take into account that the movements of certain joints are related to each other. During the optimization process, we therefore change the trajectory of each joint according to the correlation to the sampled joint. The correlation between joints of the robot $\boldsymbol{Cor}^{rob}$ and of the human $\boldsymbol{Cor}^{hum}$ are learned from the initial gait and the recorded trajectories, respectively. The pseudocode of the correlation-based optimization can be found in Alg. 1. Since the correlation of the robot's and the human's joints can be different, we perform a linear interpolation between these two values according to the current adaption factor. As before, we start with the same change rate $\Delta\alpha_i$ for all joints, but decrease this value when there is no success in several attempts in which the same joint was sampled (this is omitted in the code to ensure readability).

C. Hill Climbing Considering all Joints Simultaneously

Next, we approach the optimization problem using a hill climbing algorithm which tries in each iteration D different adaption vectors $\boldsymbol{\alpha}$ and takes then the one yielding the best result as starting point for the next iteration. In contrast to the previous methods, here we allow for three different change rates. For each joint i, α_i is changed randomly by either $+\Delta\alpha$, 0, or $-\Delta\alpha$. This way, there is a higher flexibility in finding configurations (i.e., changes in the opposite direction of the human are possible now) and we hope to find good solutions since still dependencies between joints are considered. The similarity between the trajectories can be computed before the evaluation of the resulting gait in terms of stability. Therefore, we sort the resulting gaits according to their similarity and take the first gait evaluated as stable as starting point for the next iteration. A precondition is of course, that this gait has a higher similarity than the starting gait of this iteration, otherwise we sample new change rates for the individual joints. If a certain number of attempts is not successful, we decrease the change rate. Alg. 2 depicts pseudocode of the hill climbing method. Here, we use the same absolute value of the change rate $\Delta\alpha$ for all joints, which is decreased as described above (this is omitted in the code).

Algorithm 2 Hill climbing, all joints change simultaneously. $hillClimbing(\boldsymbol{M}^{rob}, \boldsymbol{M}^{hum}, \Delta\alpha, D)$

Input: Joint angle trajectories of the robot's gait $\boldsymbol{M}^{rob}$ and of the human's gait $\boldsymbol{M}^{hum}$, change rate $\Delta\alpha$, D number of candidates evaluated in each iteration.
Output: Joint angle trajectories $\boldsymbol{M}^{rob}_{opt}$ of the most human-like, stable gait found.

```
M^rob_opt ← M^rob; α ← 0; P ← ∅; prev_sim ← 0
while not converged do
  for all j ← 1 ... D do
    insertWithPriority(P, createCandidate(α, Δα))
  end for
  while P ≠ ∅ do
    α̂ ← dequeue(P)
    M̂^rob ← generateJointAngleTrj(M^rob, M^hum, α̂)
    if similarity(M̂^rob, M^hum) > prev_sim then
      if evaluate(M̂^rob) = stable then
        M^rob_opt ← M̂^rob; α ← α̂; P ← ∅
        prev_sim ← similarity(M̂^rob, M^hum)
      end if
    else
      P ← ∅
    end if
  end while
end while
return M^rob_opt.
```

D. Policy Gradient Optimization

Finally, we developed a method based on policy gradient estimation, which is a modification of standard policy gradient reinforcement learning [20]. The idea is to evaluate a number of adaption vectors $\boldsymbol{\alpha}$ in order to approximate the gradient of the similarity for each dimension of the search space and explore in the direction of the optimum. Since stability is a precondition, it also has to be taken into account during the computation of the gradient.

As in our hill climbing approach, D different adaption vectors $\boldsymbol{\alpha}$ are sampled in each iteration to generate test gaits around the currently best gait. Again, for each joint i, the change rate $\Delta\alpha_i$ is set randomly to $+\Delta\alpha$, 0, or $-\Delta\alpha$. In each iteration, all D resulting gaits are executed and the result is for each dimension i added to one of the categories S_i^+, S_i^0, and S_i^- depending on the value of the sampled change rate for this joint. To be more specific, for joint i, the progress in similarity – i.e., the difference in distance – is added to

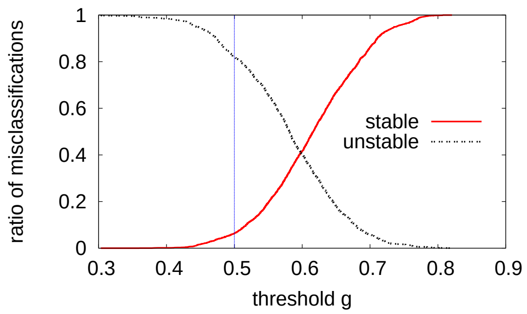

Fig. 3. Relative number of misclassifications according to a given threshold g. The red (solid) line shows the amount of unstable motions which are classified as stable according to g. The black (dashed) line shows the amount of stable runs, which are classified as unstable. We chose a value of 0.5 for g.

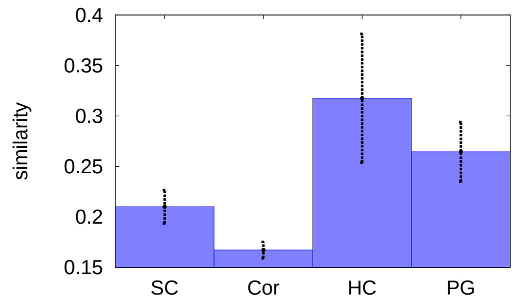

Fig. 4. Comparison of the different optimization methods. Hill climbing (HC) and policy gradient estimation (PG) perform significantly better than the single component (SC) and correlation-based (Cor) methods. All methods are able to greatly improve the similarity to the human gait. Shown are the mean and the 95% confidence interval.

S_i^+ if $\Delta\alpha_i$ is positive, otherwise it is added to S_i^- or S_i^0, respectively. If the resulting gait is unstable, 0 is added to the respective category. After evaluating each of the test gaits, the average scores Avg_i^+, Avg_i^0, and Avg_i^- of S_i^+, S_i^0, and S_i^- are computed. The corresponding values give an estimate of the gain in changing the joint angle θ_i in this specific direction and indicate how θ_i should be changed to improve the result. For each joint i, an adjustment factor δ_i is in each iteration computed as follows:

$$\delta_i \leftarrow \begin{cases} 0 & \text{if } Avg_i^0 > Avg_i^+ \text{ and} \\ & \quad Avg_i^0 > Avg_i^- \\ Avg_i^+ - Avg_i^- & \text{otherwise} \end{cases} \tag{6}$$

Afterwards, δ_i is normalized and multiplied by a scalar η. The final adaption factor α_i for joint i is then determined as:

$$\alpha_i \leftarrow \alpha_i + \eta \cdot \frac{\delta_i}{|\boldsymbol{\delta}|} \tag{7}$$

Thus, each joint is “drawn” towards the direction yielding the highest improvement with η determining the amount of change (the step size). Also in this optimization approach, the change rate $\Delta\alpha$ is decreased in case of no success.

VI. Experiments

We performed extensive experiments to evaluate the different optimization methods. We carried out ten experiments for each of the learning methods in order to perform a significance analysis. In the beginning of each run, the robot started with the same stable motions which were smoothly transformed to the desired movements resulting from the optimization process. Then, the robot executed five gait cycles with identical joint angle trajectories. At the end of each run, the trajectories were smoothly adapted to stand still stably.

A. Stability Evaluation

Since the execution of the trajectories is noisy, it cannot be guaranteed that a gait is stable if the robot does not fall in a run. To assess whether executed joint angle trajectories result in stable motions of the robot, we therefore use two criteria. First, we carry out five repetitions of executions of the same motions. We consider a gait as instable, as soon as the robot falls in a run.

To further analyze the gaits which were not classified as instable so far, we use a second criterion. This is based on the assumption, that a stable motion should be regular. To measure the degree of regularity, we use the pressure sensors in the feet of the robot and processed it in the following way: First, we transform the sensor data to the amplitude spectrum using the Fourier transform. In this representation, we compute the ratio of the sums over the amplitudes of the symmetric (as defined in Eq. 2) versus those of the asymmetric frequencies. Since we use two sensors in each foot, we get a total of four ratios. We use the worst, i.e., the smallest value to classify the regularity of the run. If the value drops below a given threshold g, we consider the run as irregular and thus as instable. Obviously, using this criterion, stable motions can possibly be classified as instable. However, it also decreases the number of instable or irregular results which are falsely classified as stable. Fig. 3 shows the classification result depending on g. While a low threshold filters out less unstable motions, it misclassifies only few stable gaits. For our experiments, we chose a value of 0.5 for g which we found out to yield good results.

B. Comparison of the Optimization Methods

The parameters used for the presented experiments are the following: The single component method used $\Delta\alpha = 0.05$, the correlation based method $\Delta\alpha = 0.1$, the hill climbing method $\Delta\alpha = 0.1$ and $D = 10$, and the policy gradient approach $\Delta\alpha = 0.1$, $D = 25$, and $\eta = 0.2$.

Fig. 4 shows the results obtained with the different optimization techniques. As can be seen, all methods are able to improve the initial robot’s gait (note that a similarity of 0 corresponds to the robot’s initial gait). Hill climbing and policy gradient optimization significantly outperform the two other methods. In comparison to single component sampling, the advantage of these methods is that several joints are adapted simultaneously. In policy gradient optimization, the gradient can only be approximated inaccurately and, therefore, it does not perform better than hill climbing.

Fig. 5 depicts the evolution of the adaption factor for different joints over time. The results are shown for the most human-like, stable gait found with hill climbing. As can be seen, especially the knee joint can be highly adapted. With hill climbing, the lateral swing can be reduced up to 12.23%.

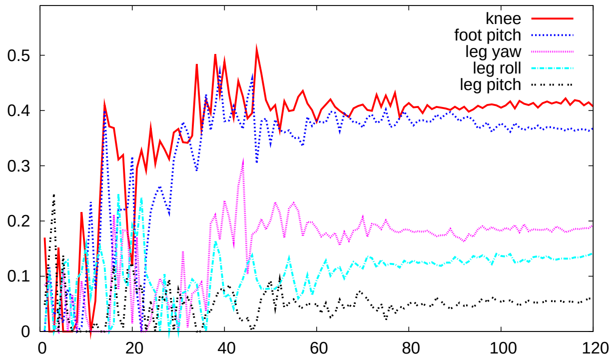

Fig. 5. Evolution of the adaption factor of selected joints over time for the best gait found with hill climbing.

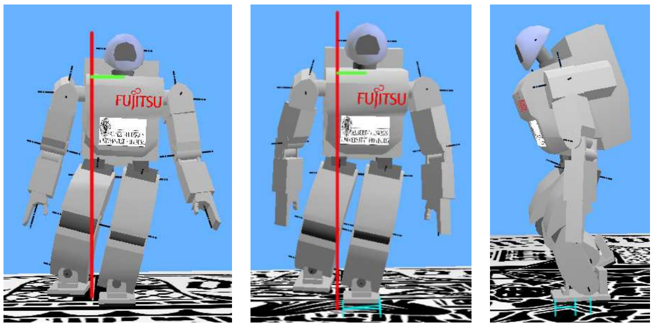

Fig. 6. Comparison of the lateral swing of the initial (left) and the optimized gait (center). The robot learned to lean forward such as humans do when carrying heavy loads (right).

In Fig. 6, snapshots of the initial (left image) and the optimized gait (center image) are depicted.

Interestingly, in several experiments with the hill climbing and the policy gradient estimation, it came out that the robot is leaned forward (see right image of Fig. 6). This is similar to the behavior of humans when carrying heavy loads. Thus, the robot learned how to walk stably and human-like despite the fact that it does not explicitly know about the different weight distribution.

VII. Conclusions

In this paper, we presented an approach to generate walking patterns for humanoid robots that are more human-like. We investigated four optimization techniques which work on joint angle basis and do not need any further knowledge. The optimization starts with a gait for the humanoid obtained by a central pattern generator and tries to optimize it with respect to human resemblance in terms of joint angle difference. We recorded human data using a full-body motion capture system.

We demonstrated in simulation experiments with a *HOAP-2* robot that all of the presented optimization techniques are able to generate joint angle trajectories that are significantly more human-like than the original gait. The best performance was achieved by methods based on hill climbing and policy gradient estimation. We achieved these results despite different anatomy and weight distribution of the robot and the human.

Acknowledgment

This work has partly been supported by the German Research Foundation (DFG) under contract number SFB/TR-8. We would like to thank Christian Liedtke and Slawomir Grzonka for their effort in collecting data with the MVN motion capture system.

References

[1] S. Aoi and K. Tsuchiya. Locomotion control of a biped robot using nonlinear oscillators. *Journal of Autonomous Robots*, 19(3):219–232, 2005.

[2] J. E. Bobrow, B. Martin, G. Sohl, E. C. Wang, J. Kim, and F. C. Park. Optimal robot motions for physical criteria. *Journal of Robotic Systems*, 18(12):785–795, 2001.

[3] XSENS Technologies B.V. Moven. http://xsens.com/en/entertainment/mvn, 2009.

[4] R. Chalodhorn, D. Grimes, K. Grochow, and R. Rao. Learning to walk through imitation. In *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, 2007.

[5] Cyberbotics. Webots 6. http://www.cyberbotics.com/, 2009.

[6] J. Denk and G. Schmidt. Walking primitive databases for perception-based guidance control of biped robots. *European Journal of Control*, 13(2-3):171–188, 2007.

[7] G. Endo, J. Morimoto, J. Nakanishi, and G. Cheng. An empirical exploration of a neural oscillator for biped locomotion control. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, 2004.

[8] F. Faber and S. Behnke. Stochastic optimization of bipedal walking using gyro feedback and phase resetting. In *Proc. of the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2007.

[9] F. Faber, M. Bennewitz, and S. Behnke. Controlling the gaze direction of a humanoid robot with redundant joints. In *Proc. of the 17th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*, 2008.

[10] Ltd Fujitsu Automation Co. HOAP-2 instruction manual. http://robita.iiita.ac.in/hoap2instruction03e.pdf, 2004.

[11] T. Geng, B. Porr, and F. Wörgötter. Fast biped walking with a sensor-driven neuronal controller and real-time online learning. *Int. Journal of Robotics Research*, 25(3):243–259, 2006.

[12] K. Harada, K. Hauser, T. Bretl, and J.-C. Latombe. Natural motion generation for humanoid robots. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2006.

[13] T. Hemker, M. Stelzer, O. von Stryk, and H. Sakamoto. Efficient walking speed optimization of a humanoid robot. *Int. Journal of Robotics Research*, 28(2):303–314, 2009.

[14] Q. Huang, Z. Yu, W. Zhang, X. Duan, Y. Huang, and K. Li. Generation of humanoid walking pattern based on human walking measurement. In *Proc. of the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2008.

[15] N. Kohl and P. Stone. Machine learning for fast quadrupedal locomotion. In *Proc. of the National Conference on Artificial Intelligence (AAAI)*, 2004.

[16] J. Kruskall and M. Liberman. The symmetric time warping problem: From continuous to discrete. In *Time Warps, String Edits and Macromolecules*, pages 125–161. Addison-Wesley Publishing Co., Reading, USA, 1983.

[17] D. Lizotte, T. Wang, M. Bowling, and D. Schuurmans. Automatic gait optimization with gaussian process regression. In *Proc. of the Int. Conf. on Artificial Intelligence (IJCAI)*, 2007.

[18] C. Niehaus, T. Röfer, and T. Laue. Gait-optimization on a humanoid robot using particle swarm optimization. In *2nd Workshop on Humanoid Soccer Robots at the IEEE-RAS Int. Conf. on Humanoid Robots (Humanoids)*, 2007.

[19] H. Serhan, C. Nasr, Hénaff, P., and F. Ouezdou. A new control strategy for ROBIAN biped robot inspired from human walking. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2008.

[20] R. Sutton, D. Mcallester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in Neural Information Processing Systems*, 12:1057–1063, 2000.

[21] M.M. Svinin, Y. Masui, Z.W. Luo, and S. Hosoe. On the dynamic version of the minimum hand jerk criterion. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2004.

[22] M. Vukobratovic and B. Borovac. Zero-moment point – Thirty five years of its life. *Int. Journal of Humanoid Robotics*, 1(1):157–173, 2004.