Journï£¡es Franï£¡aises d'Informatique Graphique,
Rennes, 23 - 27 Octobre 2017

Physics-based control of virtual characters in low frequency simulations

Samuel Carensac[1,2] Nicolas Pronost[1,3] and Saïda Bouakaz[1,3]

[1]Univ Lyon, LIRIS, CNRS UMR5205
[1]INSA de Lyon
[2]Université Lyon 1

Rï£¡sumï£¡

Using a lower simulation frequency for physics-based control of virtual characters frees computation time that can be used for more complex environment. However, using low simulation frequency may introduce instabilities inside the simulation. In this paper, we demonstrate that even a simple control strategy can be used at a low simulation frequency by adapting the control parameters. Indeed we show that lower frequencies hold a more restrictive space of possible control parameters than higher ones. We propose a method to find optimized control parameters for frequencies as low as 200Hz. As using such low frequencies may introduce foot-ground contact instabilities, we also introduce an additional control feedback on the stance leg. Our controller shows similar robustness as high frequency controller while using 0.8ms per simulation step.

Mots clï£¡ : Physics Based animation, Motion control, Offline optimization

1. Introduction

Physically simulated locomotion of virtual characters as now been studied for some time as one of the solution to generate physically realist interactions between a character and its environment. Even though many control strategies allowing a real-time simulation have been proposed [GP12], very few could be used in a virtual environment containing even a relatively small number of characters. The main reason is that most of the proposed controllers use a really high control frequency, typically around 2000Hz. Simulating the physics world with such frequency prevent any possibility for real-time interactive simulation. Recent works propose controllers able to use lower frequencies but then require large computation times to simulate the character. To our knowledge the study of the implication of using low frequencies on simple control strategies has not yet been investigated. In this paper we present what happens when the control frequency is lowered and which components must be changed and added in order to obtain a much faster simulation producing as realist motions as with a high control frequency.

Our contributions are twofold : we show that by adapting control parameters, a simple controller such as the SIMBICON [YLVdP07] can be used at control frequencies lower than the ones supposed by previous works without any additional modification to the controller. We propose a process that allows the finding of these parameters for a specific control frequency. The use of low control frequencies producing instabilities in the ground-foot contacts, we also propose a novel online optimization component that reduces these instabilities.

The reminder of this paper is organized as follows : section 2 reviews previous works. Section 3 describes our low frequency controller and our online optimization. Section 4 introduces our method for the evaluation of the control parameters. We present our results and the limitations of our method in section 6.

2. Previous works

Recently some works have been able to design character controllers using low control frequencies. [MLPP09]'s work, later improved in [MPP11], use a prediction of the contact forces and a NQR tracking of a reference trajectory to allow the control of the character at a 120Hz frequency. Although this system achieves real-time control, the average computation time for one frame of a motion with numerous contacts with the ground is around 6ms. [HENo16] use a MPC combined with a guiding reference motion to make the movement more natural looking. Their system allows the use of frequencies down to 60Hz but the gained computation time is completely consumed for high energy motions. [LYvdP*10] proposed a sampling-based controller that have been used by [Gre16] to create a 60Hz controller and using only an average of 0.62ms for one frame.

Using low simulation frequencies can lead to numerical instabilities in particular in the reaction forces at ground contact. Several propositions have been made to help solving this problem. For instance [HENo16] chose to use the smoothed contact dynamics proposed by [TET12] resulting in more stable contacts. The main drawback of this solution is that this system is not implemented in commonly used engines such as ODE or PhysX. [MLPP09] integrated a prediction of the contact forces into the controller to compute torques compensating the instabilities. The main drawback of this solution is that it needs knowledge on the inner working of the engine which limits its use to open source engines. The use of soft models for the foot [JL11] can also be used to reduce the instabilities at the cost of the computation time required for soft contacts. On a similar note [dSNV*17] proposed to apply a supplementary torque on the stance foot that ensures that the torque applied on the stance foot is smaller than a user defined limit. Although physically unrealistic, their system prevents toppling of the foot even with imperfect key poses or perturbation with a very low computational cost.

The SIMBICON and associated works [YLVdP07, CBVdP10, CPB15] propose the possibility of having a system controllable by high level parameters such as orientation, velocity, and step positions. Those controllers typically use a frequency of 2000Hz and they can still easily obtain real-time interactive simulation thanks to their low computation time. Few works tried to study the impact of lower control frequencies on the SIMBICON. [GY11] study the system in various engines and show that below a frequency of 750Hz the system becomes unstable no matter the engine. [Gre16] implemented various improvements such as angular momentum regulation or fixing of the stance foot but they did not obtain successful results at low frequencies.

Therefore our work focuses on the following two points. First, we study the control parameters, here the gains of the PD controllers, that can be used in the SIMBICON for frequencies lower than 750Hz. Second, we present an online optimization feedback component correcting the contacts with the ground.

3. Control Framework

Our system is built on the version of SIMBICON presented by [CBVdP10]. We use the improved version of the velocity tuning process proposed by [CPB15]. We modified this system even further by normalizing the observed velocity curve before modifying the learning curve. This allows us to have a clean separation between the learning curve and the global offset. We also removed the linear balance feedback controllers. The character model is mainly the same as the one used by [YLVdP07] (28 DOFs). The only difference is that we use a cuboid to model the toes instead of a sphere.

3.1. Ground contact stabilizer

The goal of the ground contact stabilizer is to compensate the instabilities that can be observed at the contact between the stance foot and the ground without modifying the engine. In an ideal situation and for a foot modeled by a cuboid, the four lower corners should be in contact with the ground. In practice, it is common to observe only three or less corners in contact. Also it is possible to observe sudden variations of which corners are in contact with the ground between successive simulation steps.

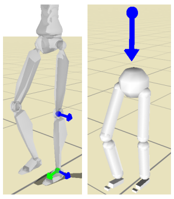

Figure 1: *left : joints affected by the ground contact stabilizer, right : example of reduced model where all the body parts above the pelvis have been simplified by a single force*

Following [LYvdP*10], we propose to search for a character pose that would result in the desired contacts. The found character pose will then be added to the current target pose to form the new target pose. Major differences with [LYvdP*10]'s work must be noted. Instead having the samples (character poses) define a pose displacement they directly define a supplementary torque to apply making the system independent from the gains of the PD controllers. That way, we will be able to limit the samples to the joints having an impact on the ground contacts. Following the same principle, we can use a simplified model of our character when evaluating the samples to diminish the computation time. Finally, we choose to not use a prevision window and handle each simulation step independently.

3.1.1. Pose samples

As mentioned above, our system focuses on the contacts between the stance foot and the ground. As such we can limit the samples to impactful joints which will reduce the search space of our sampling algorithm. We decided to focus our attention on the joints present in the stance leg as it is highly improbable that small variations of others joints have any significant impact on the contacts. We chose to exclude the toes joint as it only has an impact on the distribution of the forces between the toes and the foot. We excluded also the hip joint since it is used to control the pelvis and not the leg. Following these observations our samples are made of the ankle joint (2 DOFs) and knee joint (1 DOF), resulting in a 3-dimensional search space.

3.1.2. Simplified character model

As we limit our samples to the stance leg we can use a simplified character model during the evaluation of the samples. We chose to represent the removed body parts by a single force representing their weight. However, the stance leg is kept intact (see Fig.1).

3.1.3. Samples evaluation

Our evaluation of the contacts is done by evaluating the ground reaction forces (GRF) distribution. More precisely we will consider their ground tangent component, meaning we do not consider the components resulting from the friction. Also even if we represent the toes as a cuboid in our simulation we still consider the GRF on the foot as one force

(F_{toes}). We define the values $r_{right/left} = F_{right/left}/F_{all}$ and $r_{front/back} = F_{front/back}/F_{all}$ with $F_{right/left/front/back}$ being the sum of the forces on each side of the foot cuboid.

We define our evaluation function as follows :

$$f_{eval} = f_{quality} * 10 + f_{distance} \tag{1}$$

$$f_{quality} = \begin{cases} 1.0E15/10^{F_{all}/10+1} & F_{all} < 100 \\ 1.0E5 & F_{all} < 75 \\ (r_{cor} - limit_{cor})^2 + & \\ \quad (r_{sag} - limit_{sag})^2 & \text{otherwise} \end{cases} \tag{2}$$

$$\begin{aligned} f_{distance} =& (r_{cor} - max(limit_{cor}, r_{cor\_init}))^2 + \\ & (r_{sag} - max(limit_{sag}, r_{sag\_init}))^2 \end{aligned} \tag{3}$$

Distribution of the GRFs : $f_{quality}$ represents the quality of the GRFs distribution. $limit_{sag}$ and $limit_{cor}$ are parameters allowing the user to control the level of restriction imposed on respectively the sagital and coronal distribution. Lower values lead to faster convergence but higher values lead to more stable contacts. In our experiments, we use a value of 0.2 for both parameters.

Distance : the goal of the function $f_{distance}$ is to evaluate the distance between the distribution resulting from the use of the sample and the one we would have obtained without the optimization. r_{cor_init} and r_{sag_init} correspond to the distribution observed with the original torques. This allows us to use a target distribution that would be the closest to the no sample distribution.

By looking at these two functions, we can see that they will have similar order of magnitude. We decided to elevate by 10 the order of magnitude for $f_{quality}$ as the goal of $f_{distance}$ is only to differentiate between already acceptable distributions. We used the CMA algorithm to generate the samples.

4. Gains study

Usually the values of the gains used in the PD controllers are obtained by running an offline optimization which also determines the target poses. The evaluation function used in such optimization usually do not try to evaluate the quality of the gains themselves but are often evaluating the quality and stability of the motion. Therefore such optimization may not be able to differentiate between multiple sets of gains resulting in an acceptable motion. This leads to two issues. First, if a large number of combinations are possible then we have no way of favoring one. Secondly, if the range of possible gain values is very small then we have nothing helping our optimization to reach them, therefore failing to find a solution where there is one. If we look at the possibilities studied in [Gre16], we can see that they only considered modifications to the controller itself and never questioned the validity of the inputs (trajectories and gains).

Intuitively we can imagine that correct gain values for high frequencies have a high chance of being incorrect for lower frequencies. Indeed, when using lower frequencies the tracking of the desired pose will most likely be less precise as we correct the current position and velocity errors less often. If the gains are kept the same these larger errors will result in larger torques which may introduce vibrations around the target and possibly a failure of the controller or the physics simulation. This observation makes us formulate the following hypothesis : lower frequencies have more restrictive ranges for the possible gain values. Following this idea, we used an offline optimization to find the range of gain values for which the resulting motion is similar to the one obtained with the original frequency (2000Hz) and gain values. That offline optimization will always use identical key poses and we initiate the gain values with the ones from the high frequency.

4.1. Objective function

The optimization process uses the following objective function :

$$f_{eval} = \sum_{t<k} (f_{gains}) + f_{head} + f_{hands} + f_{speed} + f_{balance} \tag{4}$$

where k is the duration of the evaluation in seconds. This function is made of three parts. The first one numerically evaluates the set of gains :

— *evaluation of gains* (f_{gains}). Instead of simply using the sum of the gain values we chose to use their sum normalized by their initial values. This will prevent the algorithm from favoring joints.

The second part makes sure the new motion is similar to the original one. For that we look at the displacements of the end effectors, namely the head and the hands :

— *evaluation of hands* (f_{hands}). To evaluate the quality of the hands positions we used inverse kinematics to produce key poses giving us a constant target position relative to the pelvis allowing us to evaluate any deviation from this target.

— *evaluation of head* (f_{head}). This function will penalize a solution for which the head position does not stay in an elevation close to the original one.

The last part contains two terms ensuring that global characteristics are conserved. f_{speed} discards any solution resulting in an error of more than 5% on the desired character speed and $f_{balance}$ verifies that the balance is kept through the motion, i.e. it does not fall.

4.2. Optimization strategy

As already mentioned, we do not directly evaluate the sum of the gains but the sum of the gains normalized by their initial values. Such calculation heavily depends on the initial values. To find correct initial values, we apply successive optimizations where each one of them uses the result of the previous one as the starting point. To choose our starting points we made two assumptions :

— the variation in the best set of gains is continuous, ie. the best gains for a frequency should be close to the best gains for a slightly different frequency

— lower frequencies have more restrictive correct ranges of gain values

With these assumptions we are likely to be able to use the solution obtained for a frequency as the starting point for a slightly lower frequency. The optimization procedure can be seen in the algorithm 1.

Algorithm 1 Offline optimization procedure

```
Require: a set of gains $S_{ini}$ resulting in a stable motion and the associated frequency $F_{ini}$
procedure LowerFrequency($S_{ini}$,$F_{ini}$)
    ($S_i$,$F_i$) ← ($S_{ini}$,$F_{ini}$)
    do
        ($S_{min}$,$S_{max}$,$is\_valid$) ← OptimizeFrequency($S_i$,$F_i$)
        $S_i$ ← $S_{min}$
        $F_i$ ← $F_{new}$ with $F_{new} < F_i$
    while $is\_valid$

function OptimizeFrequency($S_{start}$,F)
    $iter$ ← 0
    ($S_{min}$,$is\_valid$) ← ($S_{start}$,true)
    while $iter$ < 15 && $is\_valid$ do
        ($S_{min}$,$is\_valid$) ← run_cma($S_{min}$,F,min)
        $iter$ ← $iter$+1
    ($S_{max}$,$is\_valid\_max$) ← ($S_{min}$,$is\_valid$)
    while $iter$ < 30 && $is\_valid\_max$ do
        ($S_{max}$,$is\_valid\_max$) ← run_cma($S_{max}$,F,max)
        $iter$ ← $iter$+1
    return ($S_{min}$,$S_{max}$,$is\_valid$)
```

5. Results

Our controller was implemented using the ODE physics engine. Results on the global controller (response to perturbations, orientation change, ...) are presented in the companion video.

5.1. Ground contact stabilizer

The following results have been obtained over a simulation of 200 character steps of a forward walk motion at a frequency of 300Hz. The desired coronal speed cycled between $0m.s^{-1}$, $0.2m.s^{-1}$ and $-0.2m.s^{-1}$ every 5 character steps to introduce a continuous perturbation.

We present a comparison showing the error induced in the evaluation of the future time step by using various reduced models. The body parts removed in the six different reduced models are : arms ($M1$), arms and legs ($M2$), everything above the pelvis ($M3$), everything above the pelvis and the swing leg ($M4$), arms, head and swing leg ($M5$). We also studied one special configuration ($M6$) : using $M3$ when both feet touch the ground and $M4$ if only the stance foot touches the ground. As we can see in figure 2 (left), simplifying the arms and the head does not result in an important degradation of the result. Also $M6$ shows that the swing leg has an important impact on the result even when not touching the ground. Finally comparing $M3$ and $M5$ shows that simplifying the swing leg has a less negative impact than simplifying the torso.

Our second evaluation consists in comparing the quality of the gains resulting from the use of our system and its computation time. We also ran this test with the full character ($M0$). We can see in Figure 2 (middle) that the number of simulation steps presenting a disequilibrium is greatly lowered by our system for every reduced model. The comparison of $M3$ and $M5$ confirms the previous result by showing a far lower computation time with less steps showing a disequilibrium.

5.2. Gains study

For the offline gain optimization, we started from [YLVdP07]'s values for 750Hz. We placed the target position for the hands just in front of the torso. We tested our optimization process with two different simulation scenarios : One while introducing perturbations by modifying the desired coronal seed, the other without any perturbation The results are presented in Figure 2 (right). We observe that, as assumed, lower frequencies have a smaller range of correct values for the gains. In particular we can see that for high frequencies the range of possible values for some gains is very large, as for the pelvis joint for which the position gain stands between 1800 and more than 10000. This explains why there is no need to give any special attention to the gains values if the controller is to be used only at high frequencies. Our optimization generated a stable motion for frequency down to 250Hz with the perturbations and down to 200Hz without the perturbations.

5.3. Limitations

Our ground contact stabilizer is not capable of correcting larges disequilibrium of the GRFs. In particular, correcting a distribution where only one corner for the foot touches the ground is currently impossible. This situation may frequently appear at low frequencies and most of the time results in the failure of the control.

The controllers obtained with frequencies lower than 300Hz are not very robust to external perturbations even though they are not as bad as the ones obtained in some previous works. Also we noted that with lower frequencies the controllers using the minimum gains had to use larger forces inside the system of velocity control. This could mean that some of the joints need to keep higher gain values for some joints (most likely the ones between the pelvis and the torso).

6. Conclusion

In conclusion, we have successfully obtained a SIMBICON-based controller working with control frequencies down to 250Hz which is lower than the ones presented in previous works for this type of controller. We also proposed a system reducing the perturbations caused by the interactions between the stance foot and the ground at low frequencies. The results were obtained without having to be dependent on any particular physics engine.

For future works we would like to experiment further to determine if there are gain values that are critical to follow correctly a desired velocity. Moreover we would like to investigate a method to help the ground contact controller to keep a stable motion when extreme contact conditions are detected.

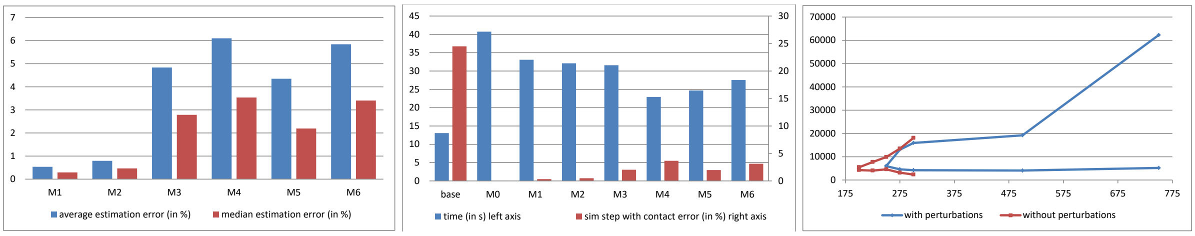

Figure 2: *Left : Estimation error observed when using each reduced model (in % of the original value) ; Middle : red : percentage of the simulation steps presenting a disequilibrium of the GRFs, blue : time consumed for the evaluation ; Right : representation of the sum of the gains* $\sum K_p + K_d * 10$ *corresponding to the minimum and the maximum possible for each frequency. frequencies lower than 250*

Rï£¡fï£¡rences

[CBVdP10] Coros S., Beaudoin P., Van de Panne M. : Generalized biped walking control. *ACM Transactions on Graphics (TOG). Vol. 29*, Num. 4 (2010), 130.

[CPB15] Carensac S., Pronost N., Bouakaz S. : Real-time gait control for partially immersed bipeds. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games* (2015), ACM, pp. 177–182.

[dSNV*17] da Silva D. B., Nunes R. F., Vidal C. A., Cavalcante-Neto J. B., Kry P. G., Zordan V. B. : Tunable Robustness : An Artificial Contact Strategy with Virtual Actuator Control for Balance : Tunable Robustness. *Computer Graphics Forum* (mars 2017).

[GP12] Geijtenbeek T., Pronost N. : Interactive character animation using simulated physics : A state-of-the-art review. *Computer Graphics Forum. Vol. 31*, Num. 8 (2012), 2492–2515.

[Gre16] Greer D. A. : *Physics-based character locomotion control with large simulation time steps.* PhD thesis, Bournemouth University, 2016.

[GY11] Giovanni S., Yin K. : LocoTest : deploying and evaluating physics-based locomotion on multiple simulation platforms. *Motion in Games* (2011), 227–241.

[HENo16] Han D., Eom H., Noh J., others : Data-guided Model Predictive Control Based on Smoothed Contact Dynamics. In *Computer Graphics Forum* (2016), vol. 35, Wiley Online Library, pp. 533–543.

[JL11] Jain S., Liu C. K. : Controlling physics-based characters using soft contacts. In *ACM Transactions on Graphics (TOG)* (2011), vol. 30, ACM, p. 163.

[LYvdP*10] Liu L., Yin K., van de Panne M., Shao T., Xu W. : Sampling-based contact-rich motion control. In *ACM Transactions on Graphics (TOG)* (2010), vol. 29, ACM, p. 128.

[MLPP09] Muico U., Lee Y., PopoviÄ J., PopoviÄ Z. : Contact-aware nonlinear control of dynamic characters. In *ACM Transactions on Graphics (TOG)* (2009), vol. 28, ACM, p. 81.

[MPP11] Muico U., PopoviÄ J., PopoviÄ Z. : Composite control of physically simulated characters. *ACM Transactions on Graphics. Vol. 30*, Num. 3 (mai 2011), 1–11.

[TET12] Todorov E., Erez T., Tassa Y. : MuJoCo : A physics engine for model-based control. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on* (2012), IEEE, pp. 5026–5033.

[YLVdP07] Yin K., Loken K., Van de Panne M. : Simbicon : Simple biped locomotion control. In *ACM Transactions on Graphics (TOG)* (2007), vol. 26, ACM, p. 105.