

Instructions for Authors
Coding with \LaTeX

\LaTeX 2 $_{\epsilon}$ Class
for Lecture Notes
in Computer Science

Version 2.4

Table des matières

1	Résumé	3
2	Introduction.....	3
3	Etat de l'art.....	3
4	modélisation des fluides	6
	4.1 forces hydrodynamiques	7
	4.2 poussée d'Archimède.....	7
5	Contrôle du mouvement	7
	5.1 modélisation du personnage.....	8
	5.2 générateur de mouvement	8
	5.3 conservation de l'équilibre	10
	5.4 contrôle de vitesse	11
	5.5 off-line optimisation	12
6	résultats	12
7	References	12

1 Résumé

A rédiger en dernier contexte 1 phrase présenter le vérou 1 phrase et comment il a été résolu 1/2 phrase résultat 1 phrase

2 Introduction

1/ 1.5 page (a faire à la fin) abstract en écrivant +, sans les résultat et snas présenter précisément la méthode contexte problématique finir par le plan du document

3 Etat de l'art

Animation de personnages humains Dans le cadre de l'animation de personnages on dispose d'un squelette composé de corps solides relié par des jointures permettant un certain nombre de degrés de libertés entre ces derniers. Il existe deux catégories de méthodes d'animation. La première, l'animation cinématique, décrit simplement les trajectoires des articulations au cours du temps. Bien que facile à réaliser cette méthode possède le défaut de rendre les interactions avec un environnement dynamique très compliquées. La seconde catégorie, l'animation basée physique, aborde le problème sous un tout autre angle. Cette fois –ci on ne déplacera pas directement les éléments du squelette mais on utilisera des interactions physiques pour les manipuler. On peut classer les interactions avec le squelette dans quatre catégories [5] :

- L'application de moments au niveau des jointures. Le principe est d'appliquer des moments au niveau des jointures pour faire se déplacer les éléments relativement entre eux suivant les degrés de liberté.
- La seconde consiste à appliquer des forces directement sur les différents corps solide pour les animer. Cependant, un personnage réel ne possédant pas la possibilité d'effectuer ce genre d'action, les résultats obtenus peuvent sembler surnaturels. Cette technique a d'ailleurs été surnommée "hand-of-god" [14].
- La dernière méthode consiste à considérer des force comme la seconde méthode à la différence que celle-ci seront simulée à l'aide de moments sur les différentes articulations concernées. Cette méthode permet donc de simuler une force tout en gardant un résultat réaliste. On la retrouve notamment de le cadre de contrôle de vitesse et d'équilibre [4].

Pour permettre l'animation du squelette de manière à obtenir les déplacements désirés, des systèmes hauts niveau sont créés [5]. Ces système prendront en compte des paramètres de haut niveau tel que la vitesse et la posture et les convertiront en interactions avec le squelette définies ci-dessus.

Contrôle dans l'espace des joints Ce type de méthode se base sur un système similaire à l'animation cinématique. L'utilisateur spécifie une série de positions désirées représentant le mouvement et le système tentera de suivre ces positions dans la limite du possible. A ceci s'ajoute des contrôleurs de feedback qui affecteront des moments supplémentaires suivant des règles intégrées au modèle.

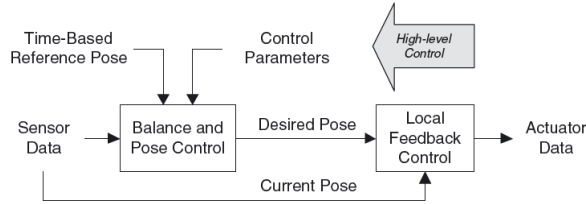


FIGURE 1. exemple de controleur dans l'espace des joints [5]

Bien qu'il existe plusieurs méthodes permettant de suivre les positions spécifiées (antagonist feedback [12], non-linear force field [11]), la méthode la plus commune est le "proportionnal-derivative control" (PD-control). Un PD-controlleur calcule un torque pour chaque articulation linéairement proportionnel à la différence entre l'état actuel et l'état désiré. Il prend en compte la différence entre les angles mais aussi la différence entre les vitesses angulaires

$$\tau = k_p(\theta_d - \theta) + k_v(\dot{\theta}_d - \dot{\theta})$$

Parmi les systèmes basés sur un PD-controller on trouve notamment le SIMBICON [19]. Le modèle est basé sur une machine à état finit. Chaque état est défini par une série de poses qui définiront les poses désirées au cours du mouvement. La transition d'un état à un autre peut s'effectuer après un certain temps ou bien lors d'un nouveau contact entre un pied et le sol. La spécification des poses clef possèdent quelques spécificités propres au SIMBICON. Les angles des articulations sont exprimé dans une repère local à l'exception de l'articulation entre le bassin et le dos et de celle de la hanche de balance sont exprimées dans le repère du monde. Enfin la hanche d'appuis ne possède pas de positions cibles. Les moments à appliquer sur celle-ci sont déterminés de manière à obtenir le moment désiré sur le pelvis sans avoir recours à des forces extérieures.

Le défaut du PD-controller est qu'il est nécessaire de connaître les bonnes valeurs pour les gains si l'on veut obtenir un résultat correct. Des gains trop faibles ne permettrait pas de suivre le mouvement défini. Des gains trop forts provoqueraient un mouvement saccadé et des possibles oscillations autour de la position désirée. On peut déterminer les bons paramètres par une série d'essais successifs mais cela ne permettrait pas d'obtenir un système robuste à des variations de géométrie dans le squelette. Pour pallier à ce problème le SIMBICON utilise un

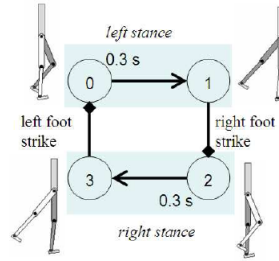


FIGURE 2. exemple de machine à état pour la marche [19]

système de feedforward [19] sur les moments des articulations permettant ainsi d'obtenir une partie des moments nécessaire sans avoir à utiliser des gains élevés. Il existe d'autres méthodes permettant de calculer automatiquement une partie des moments. On trouve notamment un système de compensation de gravité [4]. Ce système calcule des forces virtuelles compensant la gravité sur chaque partie du squelette.

Maîtrise de l'équilibre au cours de la marche Définit tel quel le système ne permettrait pas de supporter des interactions avec l'environnement qui résulteraient en un déséquilibre. Pour pallier à ce problème le SIMBICON ajoute un système de balance feedback sur la hanche de balance et le pied d'appuis. Le principe est de modifier les angles cibles en fonction de la vitesse du centre des masses et de la distance entre celui-ci et le pied d'appuis.

$$\theta_d = \theta_d0 + c_d * d + c_v * v$$

Ce système permet d'obtenir un placement du pied intelligent qui maintient le personnage dans un équilibre stable. Parmi les autres systèmes de placement intelligent du pied on trouve l'utilisation d'un inversed pendulum model (IPM) [7,4]. En plus de permettre l'équilibre l'IPM possède l'avantage de pouvoir complètement définir le mouvement de la marche humaine utilisant des paramètres plus haut niveau tel que la hauteur des pas. Un avantage majeur d'un tel système est que l'on obtient une définition indépendante des caractéristiques physique du squelette offrant ainsi un grande flexibilité. Bien que très performant l'utilisation d'un IPM limite le déplacement à de la marche. De plus l'utilisation d'un IPM pour générer le mouvement complet de la jambe de balance limite grandement les styles de déplacement possibles.

Contrôle de la vitesse Parmi les paramètres de haut niveau disponible à l'utilisateur on trouve le contrôle de la vitesse. Dans la version originale du SIMBICON la vitesse désirée est obtenue à l'aide d'une stratégie d'évolution. Le principe est de faire varier les positions cibles jusqu'à ce que l'on obtienne la vitesse voulue. Une amélioration du système [3] apporte la possibilité d'avoir

plusieurs sets d'état. Le principe est d'avoir des sets généraux (i.e. marche avant, marche arrière, ...) et de faire des combinaisons de ces états pour obtenir des états intermédiaires. Cela permet non seulement d'avoir une vitesse variable mais aussi de pouvoir changer de vitesse sans avoir à redémarrer le contrôleur. Plus récemment, un système appliquant une force virtuelle pour accélérer ou ralentir le personnage a été présentée [4]. La force à appliquer est calculée à l'aide d'un PD-contrôleur se basant sur la vitesse du personnage. Ce système permet d'avoir un contrôle fin de la vitesse. Cette méthode de contrôle a été utilisée de manière intensive dans le but de conserver l'équilibre dans le cadre de déplacement dans une position statique [6]. Cependant ce système ne permet pas de suivre correctement les vitesses si les poids du contrôleur sont incorrect. Particulièrement si l'on place le personnage dans un milieu entravant son déplacement il serait nécessaire de retrouver les gains adaptés, si donné que ces gains existent. De plus l'application d'une force virtuelle reste limité par les valeurs maximales des moments aux articulations. Ce qui veut dire que si le milieu gêne beaucoup le mouvement, cette stratégie de contrôle devient invalide.

Contrôle du mouvement en milieu aquatique La simulation d'interactions entre un personnage et un milieu liquide a déjà été effectuée. La plupart des travaux placent le personnage en milieu aqueux [18,8,17,15]. On trouve également des articles utilisant un fluide pour simuler l'effet du vent sur le personnage [9]. Cependant ces contrôleurs sont utilisés pour simuler de la nage et non de la marche. Dans les articles discutant des simulations de vent le personnage est entièrement immergé dans le fluide [9]. De plus dans ce genre de situation l'effet de la poussée d'Archimède est ignoré. Dans le cadre de la simulation de l'eau de nombreux articles se servent d'un modèle basé sur les équations de Navier Stokes [16] avec une représentation eulérienne [15]. Le défaut est que ces système ne permettent pas une simulation en temps réel. D'autres approches se contentent de modéliser l'eau à travers des forces extérieures calculées à l'aide d'équation simples [18].

Biomécanique de la marche en milieu aquatique La marche en milieu aquatique a été le sujet de plusieurs études dans le milieu de la bio-mécanique [1,2,13,10]. Certains de ces travaux travaillent sur les différences provoquées par la présence de l'eau [1]. Cependant ces travaux utilisent des niveau d'eau se situant au dessus du niveau du processus xiphoïde. C'est-à-dire que l'impact de l'eau sur le torse est toujours présent ce qui modifie fortement les résultat par rapport à notre situation.

4 modélisation des fluides

De manière à conserver une exécution en temps réel, le modèle utilisé se base sur les formules de base de dynamique des fluides. La poussée d'archimède est calculée à partir du volume immergé de la représentation physique du squelette.

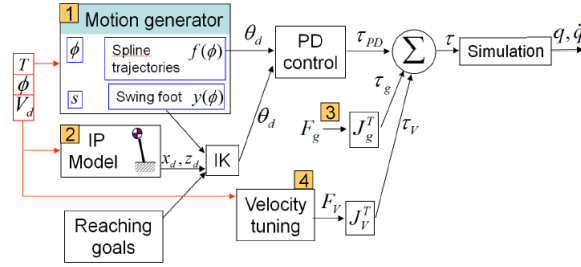


FIGURE 3. schéma récapitulatif du contrôleur //celui-ci est pris de coros 2010, je n'ai pas encore fait le mien

L'influence du déplacement du personnage à l'intérieur de fluide n'a pas été considérée.

4.1 forces hydrodynamiques

La résistance du fluide est calculée à partir de la formule suivante :

$$F_D = \frac{1}{2} \rho v^3 A C_d$$

Avec C_d le coefficient de résistance, ρ la densité du fluide, an la surface en vue du polygone et vn la vitesse du polygone. La vitesse n'étant pas constante en tout point d'un polygone nous utilisons un découpage en éléments finis de la surface des polygones pour effectuer nos calculs.

La friction du fluide est implémentée à travers un coefficient *mettre la formule*

Avec *signe grec* la viscosité du fluide.

4.2 poussée d'Archimède

La poussée d'archimède est calculé à l'aide de la formule du même nom : *formule* Le volume immergé (V_i) étant déterminé soit par l'utilisation des formules existantes (sphères, cylindres) soit par une représentation du polygone en voxels (prismes)

5 Contrôle du mouvement

Comme présenté par la figure (*lien figure*), le système de contrôle du mouvement est composé de 3 blocs principaux : suivit des poses clef, conservation de l'équilibre et suivit de la vitesse. Voici une vision globale du fonctionnement du système.

Chaque joint possède une trajectoire à suivre pour chaque degré de liberté. Ces trajectoires sont déterminées à partir d'un certain nombre de points clef et

les valeurs intermédiaires sont déterminées à l'aide de *calmut rom splines* (*chercher le terme exact*). Les joints ayant une influence sur la vitesse possèdent un set de trajectoires chacune correspondant à une vitesse. Suivant la vitesse le système calcule une combinaison des deux trajectoires des deux vitesses les plus proches. Les trajectoire de la jambe de balance sont déterminées dynamiquement à partir de la trajectoire désirée pour le pied de balance. Pour permettre l'utilisation de gains faibles dans le PD-controller nous utilisons le système de compensation de gravité dans *citation*. Ce système permet de calculer indépendamment une partie des moments nécessaire au mouvement d'alléger la charge du PD-contrôler.

La conservation de l'équilibre est réalisée par un placement du pied intelligent déterminé par un modèle de pendule inversé (IPM). Pour aider ce système les articulation de la hanche et du pied d'appuis possèdent des modèles de feedback corrigeant la position du personnage.

Le contrôle de la vitesse est obtenu à l'aide d'une force virtuelle similaire à celle utilisée par * citation*. Ce système a été amélioré pour prendre en compte les variations de vitesse interne à un pas. Ce système est aidé par une altération des résultats de l'IPM permettant d'accélérer ou de ralentir le personnage. Ces deux système sont contrôlés par du feedback et ne nécessitent aucun paramétrage.

5.1 modélisation du personnage

Le personnage utilisé est composé de corps solides simples (sphères, cylindres et prismes). Celui-ci est composé de 28 degrés de liberté interne présentés dans le figure (*mettre nbr figure*) plus 6 degrés de liberté externe (position et orientation). *mettre la figure*

5.2 générateur de mouvement

Le générateur de mouvement prend comme entrée les trajectoires définies par l'utilisateur pour chaque degré de liberté à l'exception de la jambe de balance et de la hanche d'appuis. Ces trajectoires indiquent les angles désirés à un instant donné du mouvement. Les angles sont définis comme les angles relatif entre le polygone parent et le polygone enfant de chaque articulation à l'exception des angles du pelvis et des hanches. Les angles du pelvis et de la hanche de balance sont définis dans le repère du personnage. Ce repère est similaire au repère du monde à l'exception d'une rotation sur l'axe y (axe vertical) permettant de garder l'axe z dans la direction que face le personnage. Cette rotation est définie par une variable de "heading" qui est utilisée pour contrôler la direction de marche du personnage. Les trajectoires sont des courbes de la phase ϕ d'un pas (ϕ appartient à $[0,1]$). L'alternation gauche droite des pas est gérée en utilisant la symétrie des trajectoires. Les courbes sont définies par un nombre arbitraire de points clefs choisis par l'utilisateur et les valeurs pour les positions intermédiaires sont déterminées à l'aide de *Calmull-Rom splines*. Le générateur de mouvement

convertit ces angles désirés en moments aux articulations en utilisant un PD-controller : *formule* Il est commun de voir que la position atteinte par le système soit très différente de la position demandée, particulièrement avec la présence d'un milieu aquatique.

Le générateur de mouvement est composé d'une machine à été similaire à celle utilisée par *citation du simbicon original*. Le système change d'un état au suivant à chaque fois que le pied de balance touche le sol. L'utilisateur définit une durée de temps T pour chaque pas. La phase ϕ à un instant t après le changement d'état est déterminée par $\phi = t/T$.

Les trajectoires désirées pour les éléments composant la jambe de balance sont déterminées dynamiquement par l'utilisation d'un IPM (*mettre section*) et une définition de la trajectoire du pied de balance. Le système détermine la position du pied en choisissant soit la position définie par l'utilisateur soit celle indiquée par l'IPM. Le choix est effectué suivant des règles définies dans la section présentant l'IPM.

Le moment T_b à appliquer à la hanche d'appui est calculé de manière à ce que le moment total sur le pelvis corresponde au moment permettant d'obtenir l'orientation définie par l'utilisateur. Ce moment T^* désiré est déterminé par le PD-controller en prenant pour angle cible la somme de l'angle défini dans la trajectoire du pelvis plus l'angle de heading. $T_b = T - T_a - T_{torso}$ Avec T_a le moment appliqué sur la hanche de balance et T_{torso} le moment appliqué à l'articulation entre le pelvis et le torse.

Le générateur de mouvement possède également la capacité d'adapter les trajectoires des articulations en utilisant la combinaison de plusieurs trajectoires définies par l'utilisateur. Le principe est de faire la combinaison des états optimaux pour des environnements différents (vitesse différente, présence d'un liquide). Ce système est présenté dans la *mettre lien vers la section*

compensateur de gravité Le compensateur de gravité permet l'utilisation de gains faibles dans le PD-contrôleur et ainsi éviter la création de mouvements robotiques. Notre système est très similaire à celui présenté par *citation*. Le principe est de calculer pour chaque partie du personnage une force virtuelle compensant la gravité. Donc pour chaque polygone nous appliquons une force $F = -m \cdot g$ au centre des masses, le signe négatif indiquant une force vers le haut. Cette force est ensuite convertie en moment pour chaque articulation se trouvant dans la chaîne reliant le point d'application de la force et le pelvis $T = J_{trans} \cdot F$. J_{trans} est la transposée de la jacobienne de la chaîne reliant le pelvis et le point d'application de la force. La jacobienne d'une chaîne de k articulation contient les informations indiquant l'impact d'une rotation de chaque degré de liberté de la chaîne sur le point final de la chaîne *mettre une jacobienne* Chaque *formule du truc de une case* peut être calculé par le produit vectoriel entre l'axe de la rotation et le vecteur allant de l'articulation i et le point d'application *mettre la formule* Donc la force à appliquer à chaque articulation i correspond à : *mettre la formule*

Cette méthode de compensation de gravité est appliqué à tous les solides du squelette excepté ceux composant la jambe d'appuis.

Nous avons modifié légèrement ce système pour permettre la prise en compte d'un milieu liquide. Cette modification consiste à calculer les valeurs de la poussée d'Archimède sur chaque membre du personnage et de soustraire celle-ci à la force virtuelle à appliquer. Il paraît intéressant de noter que il ne suffit pas simplement de diminuer l'intensité de la force à appliquer. En effet suivant la modélisation des membres du personnage, il est possible, et même fortement probable, que la densité soit variable à l'intérieur de ceux-ci. De ce fait le point d'application de la force compensant la gravité et celui de la poussée d'Archimède seront différents, ce qui nous oblige à déterminer le point d'application de la force finale. *formule finale de la force*

5.3 conservation de l'équilibre

L'équilibre du personnage est obtenu par le placement intelligent du pied à l'aide d'un modèle de pendule inversé (IPM) et de fonctions de feedback sur certaines articulations (balance feedback).

Modèle du pendule inversé *mettre un schéma* Pouvoir déterminer dynamiquement la position optimale pour conserver l'équilibre est un très grand avantage pour obtenir un contrôleur robuste aux interactions avec l'environnement. Pour déterminer la position $p(x,y)$ du prochain pas, nous utilisons un modèle de pendule inversé. Nous utilisons le même modèle que *citation* et considérons une longueur de jambe constante. Le principe de l'IPM est de partir du constat que la somme de l'énergie potentielle de pesanteur et de l'énergie cinétique reste constante peu importe la position du pendule. *formule* Si l'on considère que lors de la position d'équilibre (position verticale) nous désirons une vitesse nulle nous pouvons simplifier l'équation avec les variables suivantes $v=0$ et formule de h . On peut ainsi déterminer la valeur de d . *équation de d * Pour permettre d'avoir un résultat conservant une vitesse proche de la vitesse désirée une altération des résultats de l'IPM est réalisée $\Delta = \alpha \cdot V_d$. Ainsi une vitesse positive apporte une diminution de la longueur des pas provoquant une accélération et inversement une vitesse négative ralentit le personnage. Ce système est étendu à un système de feedback pour le contrôle précis de la vitesse (voir *section*). Selon la vitesse de personnage il est possible que l'IPM nous donne des résultats impossibles à atteindre c'est pourquoi la longueur des pas est limitée à $d=0.6L$ avec L la longueur de la jambe. La position effective du pied est calculée à l'aide d'une interpolation entre la position de départ du pied et la valeur retournée par l'IPM. La loi d'interpolation utilisée est une loi cubique : $P_f = (1 - \phi^2) \cdot p_{ini} + \phi^2 \cdot p(x,y)$ * La hauteur du pied au cours du pas est définie par l'utilisateur à l'aide d'une trajectoire. Les angles désirés pour la hanche et le genou sont ensuite déterminés à l'aide de la cinématique inversée (*figure expliquant comment on trouve les angles*). Il reste un degré de liberté qui est fixé pour que la position du genou se trouve dans le plan de rotation de la hanche.

Bien que l'IPM soit intéressant il limite les possibilités de mouvement. Par exemple il rend impossible un mouvement définissant une trajectoire rectangulaire pour le pied comme l'on voit sur la figure *mettre nbr figure*. Ce genre de déplacement est important dans le cadre de notre étude car elles sont typiques de lieu où la progression est beaucoup entravée (par exemple de la neige). Pour permettre l'observation de tels déplacements nous avons rajouté la possibilité de manuellement définir la position du pied au cours du pas. Cette définition manuelle est considérée temps que le personnage est dans la phase ascendante du pas $v(y) > 0$ et qu'il n'est pas dans un état instable. Nous déterminons une instabilité du personnage si nous détectons une variation irrégulière du pattern de vitesse observé au cours d'un pas (voir *mettre le numéro de la section*). Si un état instable est détecté nous utilisons les résultats de l'IPM pour la totalité du pas jusqu'à ce que nous retournions dans un état stable.

balance feedback *mettre le schéma pr le d et le v* le mettre avec le modèle du pendule inversé et les degrés de liberté Dans le cadre de ce feedback nous utilisons une loi linéaire fonction de d et de v pour altérer les angles désirés : *mettre la loi*. Avec θ l'angle qui sera utilisé par le PD-contrôleur, θ_0 l'angle spécifié par les trajectoires, d la distance entre le centre des masses et le pied d'appui et v (*figure*) la vitesse de centre des masses. Les gains C_d et C_v sont déterminés pendant la phase d'optimisation hors-ligne.

Ce système est appliqué sur la cheville d'appui. Il est utilisé pour obtenir une partie du moment nécessaire à appliquer ce qui permet de conserver des gains faibles. Il est important de signaler que ce n'est qu'une stratégie d'équilibre secondaire car la nécessité d'avoir des gains optimaux la rend très peu flexible.

5.4 contrôle de vitesse

Dans le cadre de ce projet, la capacité à marcher à une vitesse donnée est un critère primordial. De plus nous souhaitons obtenir un système qui sera robuste à la variation du milieu (présence ou non de liquide). Pour obtenir un tel résultat, nous utilisons une force virtuelle ainsi qu'un système d'altération des résultats de l'IPM.

contrôle de vitesse par force virtuelle *mettre la figure montrant la chaîne affectée* Pour obtenir une vitesse désirée nous allons appliquer une force virtuelle horizontale qui accélérera ou ralentira le personnage en déplaçant le centre des masses. Ce système est basé sur le système utilisé par *citation*. Le principe est d'utiliser un PD-contrôleur se basant sur la différence entre la vitesse actuelle du personnage et la vitesse désirée pour déterminer l'intensité de la force à appliquer. La force sera ensuite appliquée au centre des masses du personnage en calculant les torques nécessaires à l'aide de la méthode de la transposée jacobienne (*mettre le num de section de la jacobienne*). La chaîne de polygone considérée pour l'application de la force est la chaîne reliant le torse au pied d'appui (voir *figure*). Les valeurs utilisées pour la jacobienne sont légèrement

différentes de celles-ci utilisées précédemment. Cette fois nous utilisons la somme des vecteurs séparant les articulations pondérés par le poids du membre présent entre ces deux articulations. *mettre formule* Avec M la somme des poids de tous les membres considérés. Ceci nous permet d'accorder une importance plus grande aux articulations provoquant de grands changements dans la position de centre des masses.

Il est facile d'observer que la vitesse de déplacement du centre des masses n'est pas constante au cours d'un pas. Cependant lorsque nous sommes dans le cas d'un déplacement stable, ces variations de vitesse sont constantes entre chaque pas. C'est pourquoi nous avons construit un système permettant d'apprendre cette courbe de variation de vitesse de manière à appliquer une force virtuelle plus adaptée. De manière similaire aux trajectoires des articulations cette courbe est définie comme une fonction de ϕ et possède un nombre de points de référence arbitraire. Les valeurs entre les points de références sont également déterminées à l'aide de Catmull-Rom splines. Un nombre de 10 points de référence s'est montré être une valeur correcte. Ce paragraphe explique de manière plus précise l'apprentissage de la courbe. Aux cours d'un pas nous relevons pour chaque point de référence la différence entre la valeur stockée dans la courbe et la valeur observée réellement. Si la variation moyenne observée est inférieure à une heuristique, nous adaptons les valeurs stockées pour les rapprocher des valeurs observées. Pour finir nous déterminons le ratio entre la vitesse moyenne demandée et la vitesse moyenne et nous modifions les points de la courbe de façon à ce que la nouvelle courbe définisse les valeurs à demander pour obtenir la vitesse voulue.

La présence de l'heuristique nous permet de détecter les variations irrégulières de la courbe de vitesse. Une fois la situation d'équilibre atteinte nous obtenons une courbe stable qui est caractéristique au déplacement constaté. Si l'on observe une variation de cette courbe cela veut dire qu'un élément est venu perturber le déplacement. Dans ce cas, nous déclarons au système que nous ne sommes plus dans un état stable ce qui adaptera la stratégie de déplacement jusqu'à ce que le mouvement redevienne stable. Une fois le mouvement redevenu stable nous recommençons à adapter la courbe de vitesses.

altération des résultats de l'IPM

5.5 off-line optimisation

6 résultats

7 References

Références

1. Barela, A.M., Stolf, S.F., Duarte, M. : Biomechanical characteristics of adults walking in shallow water and on land. *Journal of Electromyography and Kinesiology* 16(3), 250–256 (2006)

2. Chevutschi, A., Alberty, M., Lensel, G., Pardessus, V., Thevenon, A. : Comparison of maximal and spontaneous speeds during walking on dry land and water. *Gait & posture* 29(3), 403–407 (2009)
3. Coros, S., Beaudoin, P., van de Panne, M. : Robust task-based control policies for physics-based characters. In : *ACM Transactions on Graphics (TOG)*. vol. 28, p. 170. ACM (2009)
4. Coros, S., Beaudoin, P., van de Panne, M. : Generalized biped walking control. In : *ACM Transactions on Graphics (TOG)*. vol. 29, p. 130. ACM (2010)
5. Geijtenbeek, T., Pronost, N. : Interactive character animation using simulated physics : A state-of-the-art review. In : *Computer Graphics Forum*. vol. 31, pp. 2492–2515. Wiley Online Library (2012)
6. Geijtenbeek, T., Pronost, N., van der Stappen, A.F. : Simple data-driven control for simulated bipeds. In : *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. pp. 211–219. Eurographics Association (2012)
7. Kajita, S., Kanehiro, F., Kaneko, K., Yokoi, K., Hirukawa, H. : The 3d linear inverted pendulum mode : A simple modeling for a biped walking pattern generation. In : *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*. vol. 1, pp. 239–246. IEEE (2001)
8. Kwatra, N., Wojtan, C., Carlson, M., Essa, I., Mucha, P.J., Turk, G. : Fluid simulation with articulated bodies. *Visualization and Computer Graphics, IEEE Transactions on* 16(1), 70–80 (2010)
9. Lentine, M., Gretarsson, J.T., Schroeder, C., Robinson-Mosher, A., Fedkiw, R. : Creature control in a fluid environment. *Visualization and Computer Graphics, IEEE Transactions on* 17(5), 682–693 (2011)
10. Miyoshi, T., Shirota, T., Yamamoto, S.I., Nakazawa, K., Akai, M. : Functional roles of lower-limb joint moments while walking in water. *Clinical Biomechanics* 20(2), 194–201 (2005)
11. Mussa-Ivaldi, F.A. : Nonlinear force fields : a distributed system of control primitives for representing and learning movements. In : *Computational Intelligence in Robotics and Automation, 1997. CIRA'97., Proceedings., 1997 IEEE International Symposium on*. pp. 84–90. IEEE (1997)
12. Neff, M., Fiume, E. : Modeling tension and relaxation for computer animation. In : *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*. pp. 81–88. ACM (2002)
13. Orselli, M.I.V., Duarte, M. : Joint forces and torques when walking in shallow water. *Journal of biomechanics* 44(6), 1170–1175 (2011)
14. Van de Panne, M., Lamouret, A. : *Guided optimization for balanced locomotion*. Springer (1995)
15. Si, W., Lee, S.H., Sifakis, E., Terzopoulos, D. : Realistic biomechanical simulation and control of human swimming. *ACM Transactions on Graphics (TOG)* 34(1), 10 (2014)
16. Stam, J. : Stable fluids. In : *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. pp. 121–128. ACM Press/Addison-Wesley Publishing Co. (1999)
17. Tan, J., Gu, Y., Turk, G., Liu, C.K. : Articulated swimming creatures. In : *ACM Transactions on Graphics (TOG)*. vol. 30, p. 58. ACM (2011)

18. Yang, P.F., Laszlo, J., Singh, K. : Layered dynamic control for interactive character swimming. In : Proceedings of the 2004 ACM SIGGRAPH/Eurographics symposium on Computer animation. pp. 39–47. Eurographics Association (2004)
19. Yin, K., Loken, K., van de Panne, M. : Simbicon : Simple biped locomotion control. In : ACM Transactions on Graphics (TOG). vol. 26, p. 105. ACM (2007)