HAL
open science

Cinématique guidée par les distances

Thibaut Le Naour, Nicolas Courty, Sylvie Gibet

To cite this version:

Thibaut Le Naour, Nicolas Courty, Sylvie Gibet. Cinématique guidée par les distances. Revue Electronique Francophone d'Informatique Graphique, 2012, Vol. 6 (N° 1), pp.15-25. hal-00762919

HAL Id: hal-00762919

https://hal.science/hal-00762919v1

Submitted on 9 Dec 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Revue Électronique Francophone d'Informatique Graphique, Volume 6, Numéro 1, pp. 15–25, 2012

Cinématique guidée par les distances

Thibaut Le Naour [1], Nicolas Courty [1], Sylvie Gibet[1]

[1]Laboratoire VALORIA-IRISA, Université de Bretagne Sud
Campus de Tohannic, Bât. ENSIbs, 56017 Vannes Cedex, France

Résumé
Ce papier décrit une nouvelle méthode de contrôle cinématique de systèmes articulés. Contrairement aux approches classiques, nous posons le problème dans un nouvel espace : celui des distances. Nous proposons une nouvelle formalisation de la cinématique guidée par les distances et une méthode d'optimisation qui permettent de résoudre un ensemble de problèmes classiques d'adaptation de mouvement et de cinématique inverse. L'originalité de la méthode réside également dans la possibilité d'introduire des contraintes de distance avec priorités. L'approche est validée par un ensemble d'expériences mettant en oeuvre plusieurs opérations classiques de cinématique appliquées au contrôle du mouvement de personnages et systèmes articulés, et comparée à d'autres approches au moyen de considérations de stabilité, de convergence et de performance.

This paper deals with a new method for controlling kinematically articulated systems. Unlike traditional approaches, we pose the problem in a new distance space. We propose a novel distance-based formalization of kinematic control, and an optimization method for solving a set of classical for motion adaptation and inverse kinematics problems. The originality of the method lies also in the possibility to introduce distance constraints with priorities. The approach is validated by a set of experiments using several classical kinematic operations applied to motion control of articulated figures, and compared to other approaches by means of stability, convergence and performance issues.

Mots clé : Cinématique inverse, contraintes géométriques

1. Introduction

Traditionnellement, l'animation d'un personnage est représentée par une suite de squelettes dans le temps. Un squelette est une hiérarchie d'articulations entre lesquelles les distances sont fixes. Dans un problème de cinématique inverse, on cherche à déplacer un effecteur (typiquement l'extrémité de la chaîne) vers un point $\boldsymbol{x}$. La chaîne du squelette d'animation peut-être représentée par un vecteur $\boldsymbol{q}$ décrivant les positions (axiales, angulaires) des différentes parties intervenant dans le mouvement. Dans un problème de cinématique directe, pour une chaîne à n articulations, on définit la fonction $f : SO(3)^n \rightarrow \mathbb{R}^{3n}$ telle que :

$$\boldsymbol{x} = f(\boldsymbol{q}). \tag{1}$$

Inversement dans un problème de cinématique inverse, on a :

$$\boldsymbol{q} = f^{-1}(\boldsymbol{x}), \tag{2}$$

où f représente l'ensemble des transformations qu'il faut appliquer à la chaîne $\boldsymbol{q}$ pour déplacer l'effecteur en $\boldsymbol{x}$. Nous pouvons faire un certain nombre de remarques sur la cinématique inverse traditionnelle :

- l'expression d'une tâche se fait par la définition d'un point $\boldsymbol{x}$ dans l'espace à atteindre,
- les équations permettant de trouver les positions et les orientations des articulations en fonction du point $\boldsymbol{x}$ sont hautement non linéaires,
- les systèmes sont généralement redondants vis à vis du problème à résoudre. En effet, le nombre de contraintes est souvent très fortement inférieur au nombre d'inconnues.
- bien que traité et résolu dans la littérature, la cinématique inverse pose le problème de la résolution dans le cas d'effecteurs multiples. Tel que posé, le problème atteint des limites en termes de complexité pour répondre à plusieurs contraintes de cibles. Ainsi la tâche à accomplir est rarement exprimée de manière globale sur l'ensemble du squelette mais plutôt introduite par un ensemble de tâches entre lesquelles il faut émettre des priorités.

Dans notre approche, nous cherchons à caractériser une posture clé non pas par $\boldsymbol{q}$, mais par des distances. Il s'agit alors

© REFIG 2012, Revue Électronique Francophone d'Informatique Graphique.
Publiée par l'Association Française d'Informatique Graphique

de contrôler le mouvement d'un système articulé en spécifiant des relations de distance à vérifier entre différents points du système. En caractérisant le mouvement dans l'espace des distances, la chaîne du squelette est ainsi représentée par un vecteur $\boldsymbol{w}$ décrivant un ensemble de positions associées à ces distances. Pour n articulations associées à m distances, nous pouvons exprimer la cinématique directe avec une nouvelle fonction $h : \mathbb{R}^m \rightarrow \mathbb{R}^{3n}$:

$$\boldsymbol{x} = h(\boldsymbol{w}). \tag{3}$$

Inversement pour la cinématique inverse, nous utilisons la notation :

$$\boldsymbol{w} = h^{-1}(\boldsymbol{x}). \tag{4}$$

Intérêt de la méthode. Tout comme la cinématique inverse traditionnelle, le problème formulé est également hautement non linéaire. Notre approche sera donc également de type variationnelle. En revanche, l'intérêt de la formulation proposée réside dans **la manière de définir les tâches à travers des distances**. Nous verrons à travers un jeu de tests que cette nouvelle façon d'exprimer les tâches simplifie grandement le travail de contrôle de l'animation, notamment en fournissant **une spécification implicite des contacts** (par opposition à une définition explicite spatiale des zones de contact). Ce dernier point fait l'objet de travaux récents [HK09, HKT10] qui montrent l'intérêt de la communauté pour cette problématique. Par ailleurs, notre système permet de traiter dans son ensemble le squelette, et la résolution d'une tâche ou de plusieurs tâches simultanées se fait simplement. Nous verrons enfin qu'à travers un découplage de l'information fixe (distances des segments du squelette) et de l'information qui est modifiée dans le temps (les autres distances), nous pouvons émettre des priorités entre les contraintes et ainsi donner un style à l'animation.

Plan de l'article. Dans un premier temps nous formalisons le problème à partir de l'expression de distances (section 3). Dans ce cadre, un squelette est caractérisé par un ensemble de points, chaque point s'exprimant par rapport aux autres en termes de distances. Nous proposons une méthode pour résoudre le problème de cinématique directe et mettons en évidence comment à travers cette formulation spatiale du squelette nous pouvons traiter le cas de la cinématique inverse (sections 4 et 5). Enfin, nous proposons un jeu de tests afin de valider notre méthode (section 6).

2. Approches connexes

Cinématique inverse et applications Les techniques de cinématique inverse ont été largement étudiées au cours des dernières décennies, dans un contexte à la fois graphique [Wel93, ZB94] mais surtout robotique [WE84, NH86]. Ces méthodes restent un outil de choix, à la fois pour générer des mouvements a priori inconnus, mais aussi comme composant de méthodes plus spécifiques comme l'adaptation de mouvements à de nouvelles contraintes [Gle98, CK00, MBBT00, KMA05, CB06, HRE*08], l'inférence de poses basée sur des capteurs visuels [JiCVGHP10, HSP11], ou bien encore la reconstruction de mouvements partiellement acquis depuis des méthodes de capture [GMHP04, AL11]. Les approches les plus connues ont fait l'objet d'études comparatives dans le cadre de la reconstruction de différentes classes de mouvement [UPBS08].

Cependant, le mauvais conditionnement mathématique du problème, couplé à l'importante redondance des chaînes articulées considérées (que ce soit pour un bras ou une jambe, mais à plus forte raison pour un humanoïde complet), nécessite l'adjonction de tâches secondaires qui viennent surcontraindre le problème [BMT96, YN03, BB04, SK05]. Notons comme applications possibles de ces tâches secondaires la gestion des butées articulaires [CMA01], ou bien encore le maintien de l'équilibre d'une posture [BMT96]. Des travaux plus récents s'intéressent à l'intrication de plusieurs niveaux de contraintes [BB04, CB06, SK05] à l'aide de niveaux de priorité. La difficulté est ici de faire coexister des tâches potentiellement antagonistes sans idée a priori sur la réalisation globale de l'ensemble des consignes.

Du point de vue de la résolution du problème inverse, dans la grande majorité des cas, les algorithmes existants sont basés sur une descente de gradient exploitant une inversion de la Jacobienne évaluée numériquement. L'une des méthodes les plus efficaces consiste à calculer la pseudo-inverse de Moore-Penrose associée à une fonction d'amortissement [Mac90]. Cependant, il existe plusieurs alternatives à ce fonctionnement : approches sensori motrices [GM94], approches probabilistes [GMHP04] ou inférence Bayésienne [CA08]. Notons aussi la possibilité d'utiliser des informations issues de mouvements capturés comme poses de référence dans les processus d'optimisation [GMHP04, CBT07, TWC*09, RB09]. Ces approches travaillent non plus directement dans l'espace des poses mais plutôt dans des espaces latents (obtenus par méthodes de type analyse en composantes principales), sur-contraignant ainsi le problème et traduisant de manière implicite les corrélations pouvant exister entre certaines articulations. Notons ici que dans la méthode que nous proposons, l'utilisation de poses de référence, définies comme un ensemble de distances à respecter, joue un rôle similaire à ces approches basées données.

D'autres méthodes proposent de résoudre le problème de cinématique inverse en se plaçant, non plus dans l'espace des orientations mais dans celui des positions (espace euclidien). Ainsi, la méthode FABRIK [AL11] propose un nouvel algorithme de cinématique inverse qui permet de remonter de proche en proche aux différentes positions des articulations d'une chaîne, en parcourant la chaîne en avant et en arrière en un nombre fini d'itérations. L'intérêt principal de cette approche réside dans la possibilité de contrôler des systèmes multi-effecteurs mais également dans sa rapidité

© REFIG 2012.

d'exécution. Cette méthode, qui s'exprime sous la forme d'une séquence de constructions géométriques locales, ne garantit pas cependant l'optimalité de la solution obtenue. Dans la méthode intitulée « Particle IK Solver » [HRE*08], le personnage est représenté par un ensemble de particules 3D (positions des articulations). La méthode proposée se ramène à une méthode de résolution itérative sous contraintes, qui s'appuie sur un ensemble de préconditions ad hoc et opère en deux phases distinctes appliquées successivement aux différentes parties du corps à animer. Cette décomposition du problème, en permettant un contrôle local, apporte des solutions flexibles et paramétrables qui préservent le style des mouvements. Ho et al [HKT10] s'intéressent également à une représentation du squelette et du maillage sous la forme d'un nuage de points. Mais contrairement aux deux approches précédentes, le problème d'adaptation du mouvement (en particulier l'adaptation morphologique) est formalisé comme un problème global d'optimisation spatio-temporelle, les variations de position étant calculées à partir de la minimisation d'une somme de grandeurs énergétiques (déformation, accélération, contrainte énergétique). Enfin, une nouvelle méthode de résolution d'IK, basée sur des contraintes de distance est proposée en robotique [HR06]. Cette méthode géométrique consiste à paramétrer le système par des distances inter-articulations, et à reformuler le problème à partir de la résolution d'un système d'inégalités linéaires.

De manière similaire aux approches précédentes, notre méthode représente le personnage à animer à partir d'un ensemble de positions cartésiennes. Elle permet, comme dans [AL11] et [HR06] de gérer des effecteurs multiples et des graphes cycles. Cependant, notre méthode diffère essentiellement dans la méthode de résolution de l'IK. Il s'agit d'un processus d'optimisation itératif, global, et non linéaire. Un ensemble de distances (intra-squelette et distances avec l'environnement) est séparé en deux sous-ensembles distincts de contraintes (primaires et secondaires) qui sont directement exploitées dans le processus d'optimisation. De plus, les contraintes secondaires sont projetées dans le noyau de la tâche principale, ce qui constitue un apport intéressant de la méthode.

Distances, topologies L'originalité de la méthode que nous proposons provient de l'utilisation des informations de distance comme d'un espace de représentation pertinent du point de vue de la définition des tâches de la cinématique inverse. Cette formulation fait écho aux travaux existants dans le domaine de la biologie computationnelle, et plus particulièrement dans le calcul des conformations moléculaires [Hav98]. Dans ce contexte, seule l'information de distances entre les différents atomes des molécules est connue (notamment grâce aux techniques de résonance magnétique), et doit permettre de remonter à la topologie globale de la molécule, nécessaire à l'étude de sa fonction. De nombreuses méthodes existent. Elles reposent, soit sur une construction progressive de la molécule [DW03], soit sur des méthodes de recherche opérationnelle type séparation et évaluation [MLLT10]. Un état de l'art récent de ces méthodes est donné dans [LLMM10]. Notons ici cependant qu'à la différence d'un squelette humain, où le nombre de distances rigides entre deux articulations est très inférieur au nombre de couples d'articulations, la tendance générale est plutôt inversée dans le cas des molécules (problème fortement sur-contraint) et nécessite donc des procédures d'optimisation plus complexes intégrant l'incertitude sur la connaissance des distances.

Méthodes géométriques D'autres approches, telle que l'analyse spectrale de la matrice métrique du nuage de points [SS85], permettent de déterminer les coordonnées des points. Cette méthode nécessite cependant la connaissance de l'ensemble des distances exactes (respect des inégalités triangulaires). Enfin les méthodes introduites dans le domaine de la conception assistée par ordinateur proposent la résolution de système de contraintes géométriques telles que les distances entre deux points ou encore l'angle entre deux vecteurs. La résolution de tels systèmes a donné lieu à plusieurs méthodes présentées par Jermann et al [JMS07]. Nous notons en particulier celles utilisant les descentes de gradient (Newton, par homotopie). Elles présentent les avantages d'être rapides et proposent des résultats approchés en l'absence de solution.

3. Cinématique guidée par les distances : formalisation

3.1. Notion de distance

Notre étude porte sur l'animation d'un squelette dans l'espace des distances. Chaque articulation peut être représentée géométriquement par un point dans $\mathbb{R}^3$. En terme de distance, nous cherchons à exprimer l'écart d'une articulation avec d'autres articulations du squelette ou encore avec des points de l'espace. Plus précisément, un squelette peut être représenté par un graphe $G = (V, E, d)$ où V est l'ensemble des points (centres des articulations), E l'ensemble des arêtes reliant les points de V et d l'ensemble des distances de $V \times V \rightarrow \mathbb{R}_+$. Nous noterons en particulier d_{ij} la distance euclidienne entre le point $\boldsymbol{x}_i$ et le point $\boldsymbol{x}_j$, pour tout $(\boldsymbol{x}_i, \boldsymbol{x}_j) \in E$. Nous pouvons exprimer la structure spatiale de ce graphe par l'ensemble des équations :

$$\forall (\boldsymbol{x}_i, \boldsymbol{x}_j) \in E, \|\boldsymbol{x}_i - \boldsymbol{x}_j\| = d_{ij}, \tag{5}$$

où $\|.\|$ est la norme euclidienne. Pour faciliter l'écriture, nous utilisons indifféremment la notation $\boldsymbol{x}_i$ pour représenter le point de V ou le vecteur $\overrightarrow{O\boldsymbol{x}_i}$.

Notre problème peut se formuler de la manière suivante : à partir d'un ensemble suffisant de distances, il s'agit de retrouver les positions des points (et donc des articulations) dans l'espace $\mathbb{R}^3$.

Chaque point de V est potentiellement associé à un certain nombre de distances dépendant de la topologie du graphe

© REFIG 2012.

en ce point. Remarquons dès à présent que pour inférer un point $\boldsymbol{x}_i$ dans l'espace il suffit de disposer de quatre points connus et deux à deux distincts, associés à leurs distances respectives au point $\boldsymbol{x}_i$. Ceci implique que suivant les cas nous pouvons simplifier le problème posé dans l'équation (5).

Numériquement, résoudre de telles équations s'avère difficile. Il existe plusieurs méthodes. Les méthodes par approche spectrale répondent bien au problème mais ne s'appliquent toutefois qu'à des situations où toutes les distances sont connues et exactes (respect des inégalités triangulaires). Nous avons choisi d'utiliser des méthodes d'optimisation qui permettent en particulier d'approcher les solutions dans le cas où les distances ne sont pas exactes, ou encore de réduire l'espace des solutions dans le cas où toutes les distances ne sont pas connues.

3.2. Formulation par un problème de minimisation

Résoudre le problème en passant par des méthodes d'optimisation revient à reformuler l'équation (5) sous la forme d'un problème de minimisation d'énergie W tel que :

$$W = arg \min_{\boldsymbol{X}} \sum_{E} (\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 - d_{ij}^2)^2, \tag{6}$$

où $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n)$ est le vecteur d'inconnues à déterminer, avec $n = |V|$ et $m = |E|$.

Remarquons que dans l'équation (6), nous effectuons une pénalisation quadratique de l'erreur. L'intérêt est d'assurer une erreur positive pour chaque équation, ce qui implique que plusieurs erreurs ne peuvent pas mutuellement s'annuler. Il en résulte une meilleure stabilité de l'algorithme.

Nous remarquons également que l'équation (6) est un système d'équations non linéaires.

Nous appellerons dans la suite de l'article *contraintes* les règles que doivent satisfaire les solutions réalisables du problème. Parmi celles-ci l'égalité de l'équation 5 (contrainte de distance) ou encore l'égalité d'un point avec un autre ($g(\boldsymbol{x}) = \boldsymbol{x}_T$) caractérisent les contraintes que nous pouvons donner à notre système. Appliqué à un squelette, l'objectif du problème revient alors à associer à chaque articulation un nombre suffisant de contraintes.

Afin de décrire plus précisément la méthode d'optimisation, nous traiterons uniquement dans la section suivante des contraintes de distance établies dans l'ensemble des points $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_n\}$. L'ensemble de ces contraintes peut être mis sous la forme d'un système d'équations F de $m.n$ fonctions objectifs f_{ij} où chaque fonction reprend l'égalité de distance entre deux points $\boldsymbol{x}_i$ et $\boldsymbol{x}_j$ telle que :

$$f_{ij}(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2 - d_{ij}^2)^2 = 0. \tag{7}$$

Dans ce cadre, le problème revient donc à résoudre le système d'équations $F(\boldsymbol{X}) = 0$, où $F = (f_{11}, ..., f_{1n}, ..., f_{mn})^T$ représente le vecteur des fonctions objectifs et $\boldsymbol{X} = (\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n)^T$ est le vecteur des variables inconnues.

4. Méthodes d'optimisation pour résoudre un système d'équations

Le système d'équations étant non linéaire, une approche par descente de gradient telle que la méthode de Newton est utilisée. Elle présente l'avantage d'être rapide et suffisamment stable pour notre problème. En effet une animation supposant une continuité temporelle, cela introduit une faible variation des postures du squelette entre chaque itération de l'algorithme. La formule de Newton appliquée à notre problème s'écrit :

$$\boldsymbol{X}_{k+1} = \boldsymbol{X}_k - \boldsymbol{J}_F(\boldsymbol{X}_k)^{-1} F(\boldsymbol{X}_k), \tag{8}$$

où $\boldsymbol{X}_k$ est le vecteur $\boldsymbol{X}$ obtenu à l'itération k de l'optimisation et $\boldsymbol{J}_F(\boldsymbol{X}_k)$ est la matrice jacobienne de $F(\boldsymbol{X}_k)$ au point $\boldsymbol{X}_k$.

La matrice jacobienne $\boldsymbol{J}$ rassemble les dérivées partielles de toutes les contraintes du problème F, c'est une matrice de dimension $m \times n$ ($m \times n$ contraintes pour $n \times 3$ inconnues) :

$$\boldsymbol{J}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n) = \begin{pmatrix} \frac{\partial f_{11}}{\partial \boldsymbol{x}_1} & \cdots & \frac{\partial f_{1n}}{\partial \boldsymbol{x}_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_{m1}}{\partial \boldsymbol{x}_1} & \cdots & \frac{\partial f_{mn}}{\partial \boldsymbol{x}_n} \end{pmatrix}. \tag{9}$$

avec $\frac{\partial f_{ij}}{\partial \boldsymbol{x}_j} = (\frac{\partial f_{ij}}{\partial x_j} \frac{\partial f_{ij}}{\partial y_j} \frac{\partial f_{ij}}{\partial z_j})^T$

x_j, y_j et z_j étant les coordonnées du point $\boldsymbol{x}_j$.

Nous remarquons que le problème précédemment posé est plus général et qu'il n'amène pas forcément à un nombre de fonctions égal au nombre de variables inconnues. En effet, en fonction du nombre de contraintes définies par l'utilisateur, nous pouvons nous ramener à des problèmes sous-déterminés similaires à ceux de la cinématique inverse standard, avec plus d'inconnues que de contraintes à respecter. A contrario, notre méthode permet simplement d'ajouter des contraintes de distance réduisant l'espace de solution, jusqu'à se retrouver dans des problèmes sur-contraints (plus de contraintes que d'inconnues). Mathématiquement, ces différentes situations conduisent à des matrices jacobiennes qui ne sont pas forcément carrées, ni de rang plein. Nous utilisons donc la matrice pseudo-inverse (ou inverse de Moore-Penrose) $\boldsymbol{J}^+$ qui est une généralisation de l'inverse caractérisant toute matrice $m \times n (m > n)$ telle que $\boldsymbol{J}^+ = (\boldsymbol{J}^T \boldsymbol{J})^{-1} \boldsymbol{J}^T$. Le calcul de la pseudo-inverse est réalisé par la méthode des valeurs singulières (SVD), et l'évaluation de la matrice jacobienne est effectuée à travers l'expression analytique du gradient de la contrainte de distance entre deux points. Ainsi pour un élément de la matrice $\boldsymbol{J}$ évalué aux points $\boldsymbol{x}_i$ et $\boldsymbol{x}_j$ dans $\mathbb{R}^3$, le gradient s'écrit :

© REFIG 2012.

$$\frac{\partial f_{ij}}{\partial \boldsymbol{x}_j}(\boldsymbol{x}_i,\boldsymbol{x}_j) = -4.(\boldsymbol{x}_i - \boldsymbol{x}_j).f_{ij}(\boldsymbol{x}_i,\boldsymbol{x}_j) \qquad (10)$$

5. Adaptation de mouvement et cinématique inverse

Pour résoudre des problèmes de cinématique inverse pure, le modèle décrit ci-dessus est pleinement satisfaisant. Si l'on considère une chaîne articulée fixée à une extrémité, connaissant la posture initiale et une cible à atteindre par l'autre extrémité, il suffit en effet de donner au système les contraintes de distance entre les articulations adjacentes du squelette.

Le problème ainsi formulé nous permet de rejouer un mouvement ou encore de résoudre la cinématique inverse en minimisant itérativement à chaque posture clé de l'animation les différentes contraintes de distance. Dans l'objectif d'adapter un mouvement, il est nécessaire d'introduire de nouveaux concepts et d'améliorer la méthode d'optimisation précédente. Nous pouvons différencier deux types de contraintes de distance (figure 1) :

- **Contraintes dures :** ce sont les contraintes qui doivent être impérativement respectées. Il s'agit par exemple des distances invariantes dans le temps qui sont les distances entre les articulations adjacentes dans la hiérarchie du squelette. Elles correspondent aux segments. Il s'agit également des contraintes cibles de la cinématique inverse, comme par exemple le contact d'un pied avec le sol représenté par une distance nulle entre le pied et un plan. Pour ces contraintes les fonctions doivent être nulles à l'issue de l'itération de la boucle d'optimisation.
- **Contraintes souples :** ce sont les autres distances du graphe. Dans le cadre de la cinématique directe, elles évoluent dans le temps et ce sont finalement elles qui caractérisent l'animation.

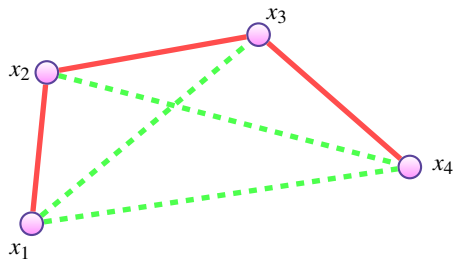

Figure 1: *Les segments en pointillés représentent les contraintes de distance souples alors que les autres segments représentent les contraintes de distance dures.*

L'introduction de ces contraintes est motivée par la nécessité de donner une priorité à certaines contraintes par rapport à d'autres. Dans le cadre de l'adaptation de mouvement, nous partons d'un mouvement référence calculé dans l'espace des distances auquel on ajoute une tâche via des contraintes dures. Ceci induit un conflit entre les contraintes : la longueur des segments ainsi que la tâche ajoutée sont des contraintes à satisfaire en priorité (contraintes dures), alors que les distances qui caractérisent le positionnement spatial des articulations de la posture de référence constituent des propriétés que l'on souhaite conserver au mieux (contraintes souples). Le système ainsi défini est finalement sur-contraint. Nous proposons donc une modification de l'algorithme de Newton qui consiste à ajouter une tâche secondaire (contraintes souples) qui est projetée dans l'espace des solutions de la tâche principale (contraintes dures). Cela permet d'incorporer les contraintes dures et souples dans un même algorithme en tenant compte éventuellement des priorités entre les contraintes, comme c'est le cas dans la formulation proposée par Boulic et al [CB06]. Notre algorithme devient :

$$\boldsymbol{X}_{k+1} = \boldsymbol{X}_k - \boldsymbol{J}_0^+ F(\boldsymbol{X}_k) + (\boldsymbol{I} - \boldsymbol{J}_0^+ \boldsymbol{J}_0)\boldsymbol{J}_1^+ G(\boldsymbol{X}_k) \qquad (11)$$

où $\boldsymbol{X}_k$ est le vecteur des positions des articulations à déterminer. Par souci de simplification des notations, $\boldsymbol{J}_{F_0}^+(\boldsymbol{X}_k)$ et $\boldsymbol{J}_{F_1}^+(\boldsymbol{X}_k)$ sont notées $\boldsymbol{J}_0^+$ et $\boldsymbol{J}_1^+$. $F(\boldsymbol{X}_k)$ et $G(\boldsymbol{X}_k)$ représentent respectivement les fonctions objectifs associées aux contraintes dures et souples, et $\boldsymbol{J}_0^+$ et $\boldsymbol{J}_1^+$ sont les pseudo-inverses des jacobiennes associées. $(\boldsymbol{I} - \boldsymbol{J}_0^+ \boldsymbol{J}_0)$ est un opérateur de projection qui garantit que la réalisation de $G(\boldsymbol{X}_k)$ n'affectera pas celle de $F(\boldsymbol{X}_k)$.

6. Résultats

Afin de valider notre méthode, nous l'expérimentons sur plusieurs cas d'études. Nous mettons ainsi en évidence la convergence de la méthode et l'importance de la définition des tâches à accomplir. Tout d'abord nous nous intéressons à la tâche de suivi de mouvement qui a pour objectif de rejouer une animation. Dans un second temps, nous expérimentons des situations de cinématique inverse, notamment en comparant les résultats à ceux obtenus par une méthode de cinématique traditionnelle. Enfin nous testons notre méthode sur une expérience d'adaptation de mouvement. Nos tests sont tous réalisés en temps réels (10 images par seconde pour l'animation la moins rapide). Ils sont effectués par un seul processus avec un processeur cadencé à 2.3 GHz. Nous avons testé et utilisé deux bibliothèques mathématique ([Dav06]), [GJ*10]) pour valider nos résultats.

6.1. Suivi de mouvement

De façon à valider la capacité de notre méthode à calculer une animation dans l'espace des distances, notre premier test porte sur un exemple rejoué, qui représente la reproduction d'un mouvement source S (données capturées) vers un mouvement cible C pour des squelettes identiques. Nous définissons quatre points deux à deux distincts dans l'espace (soit la base canonique et l'origine). Pour chaque pas de temps, après avoir calculé les positions des articulations de

© REFIG 2012.

S, nous extrayons pour chaque articulation du squelette S les distances avec ces quatre points. Enfin nous incorporons ces distances dans notre système. Plus précisément, pour un squelette à n articulations, nous définissons un problème de $4n$ contraintes pour $3n$ inconnues. Notre test est effectué sur un mouvement de marche stocké au format *bvh* provenant de la base CMU.

Afin de valider notre méthode, nous évaluons la convergence de l'algorithme, et nous montrons (figure 3) le suivi en temps réel de l'animation cible. La figure 2 montre bien le succès de la convergence. Plus précisément, nous pouvons en déduire que la minimisation des erreurs pour chaque contrainte à l'issue de 10 itérations est égale à zéro (rappelons que la minimisation a été élevée au carré). Cela se traduit par une distance nulle entre les articulations de S et C, et visuellement par un suivi à l'identique de l'animation C sur S (figure 3).

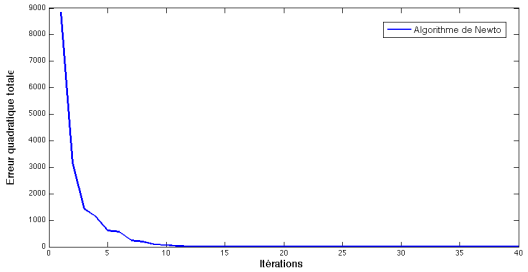

Figure 2: *Résultats de convergence par l'itération de Newton (passage de la pose de référence à la première pose du mouvement)*

Figure 3: *Illustration de la reproduction de mouvement par tracking. Figure de gauche : posture de référence (à gauche), posture initiale(à droite). Figure de droite : le mouvement de référence (à gauche) et le mouvement calculé (à droite).*

Cette expérimentation montre également la robustesse de la méthode : en effet, comme l'illustre la figure 3, l'algorithme d'optimisation peut s'adapter à des variations importantes de valeurs initiales (posture très différente de la posture de référence). L'animation de personnage présente l'avantage d'avoir une pause de référence (bind pose) qui introduit un vecteur initial dans l'itération de Newton déjà très proche de la solution.

6.2. Cinématique inverse

6.2.1. Atteinte d'une cible par une chaîne articulée

Notre second test (figure 4) porte sur un exemple simple de cinématique inverse. L'objectif est de faire atteindre une cible $\boldsymbol{c}$ à l'extrémité d'une chaîne articulée $\boldsymbol{e}_1$, l'autre extrémité $\boldsymbol{e}_2$ étant attachée à un autre point $\boldsymbol{p}$. Pour cette expérience, nous construisons une chaîne articulée formée de 10 articulations, chacune espacée de la même distance de l'articulation adjacente. Le problème se pose ainsi : nous donnons au système les 9 contraintes propres à la chaîne articulée plus 2 contraintes, de telle façon que les distances entre $\boldsymbol{e}_1$ et $\boldsymbol{c}$ et entre $\boldsymbol{e}_2$ et $\boldsymbol{p}$ soient nulles. Il s'agit ici de mettre en évidence le bon respect des contraintes (11 contraintes pour 3×10 inconnues).

Les résultats montrent en premier lieu le succès de l'expérience avec des contraintes respectées (l'erreur est à zéro après 18 itérations). Au delà du respect de ces contraintes, l'expérience montre que notre méthode est capable sur un exemple simple de réaliser les même tâches que celles résolues par la cinématique inverse traditionnelle.

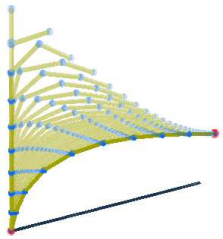

Figure 4: *Illustration du test du bras articulé. Les contraintes ajoutées sont illustrées par les sphères rouges.*

Notre méthode ainsi résolue présente cependant l'inconvénient d'être moins rapide que les méthodes standard de cinématique inverse. En effet, comme illustré dans la figure 5, nous avons comparé les temps moyens de calcul d'une itération en fonction du nombre d'articulations d'une chaîne pour notre méthode et pour la cinématique inverse standard. Les différences de temps entre les méthodes s'expliquent par le fait que notre matrice jacobienne basée distance est dans le cadre de cette implémentation de dimension bien supérieure à la matrice jacobienne classique (basée orientation), conduisant par conséquent à un temps de calcul de l'inversion plus important. Remarquons toutefois que notre matrice est fortement creuse et que l'on peut envisager, pour accélérer l'inversion de ce type de matrice, des méthodes telles que celle proposée par M. Berry [Ber92]. L'accélération de la méthode dans ce contexte est implémentée et discutée dans la section 6.3. Notons également quelques irrégularités dans la courbe basée distance, dues à des variations de la charge du processeur.

© REFIG 2012.

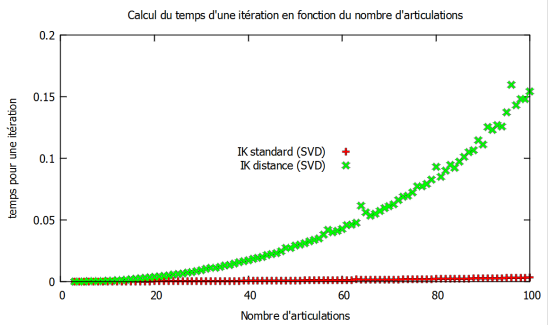

Figure 5: *Temps moyen de calcul d'une itération en fonction du nombre d'articulations pour les méthodes basées distance (en vert) et les méthodes de cinématique inverse traditionnelle (en rouge).*

Dans le cas de la cinématique inverse standard, les problèmes de singularité amènent à utiliser une matrice d'amortissement. Les résultats obtenus sont fortement dépendants de la bonne estimation des coefficients de cette matrice. Il est intéressant de noter que notre méthode donne une configuration régulière de la chaîne sans paramètre d'amortissement (voir figure 6). Cela s'explique par le fait que la minimisation est calculée dans l'espace des distances, ce qui est équivalent à exprimer les variations à travers la jacobienne le long de l'axe entre deux points contraints, contrairement à la cinématique inverse standard qui applique ces variations sur les 3 axes de rotation de chaque articulation.

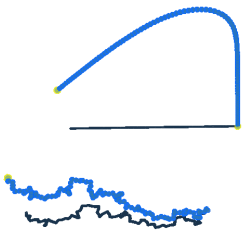

Figure 6: *Illustration pour deux chaînes identiques. En bas : calculé par la cinématique inverse standard avec un mauvais facteur d'amortissement. En haut : calculé par notre méthode.*

6.2.2. Application au contact d'une main avec un objet

A travers cette expérience, nous cherchons à mettre en évidence la capacité de la méthode à exprimer simplement des tâches sous la forme de distances, ainsi qu'à gérer implicitement un nombre important de contraintes associées à un certain niveau de priorité. Dans le cadre de la cinématique inverse traditionnelle, il est possible également de spécifier des tâches complexes sous la forme de buts multiples à atteindre, caractérisés par des contraintes géométriques de différentes sortes [PZB90]. Ce type d'approche peut se décomposer en deux phases :

- Il s'agit d'abord de convertir les contraintes en termes de buts à atteindre,
- puis de résoudre la cinématique inverse dans l'espace des coordonnées angulaires.

Notre méthode s'affranchit de la phase de conversion. Elle permet de spécifier directement la tâche sous la forme d'un ensemble de propriétés à respecter. Ces propriétés sont implicitement intégrées dans la matrice jacobienne, et l'algorithme d'optimisation cherche à trouver une solution convergente dans l'espace des coordonnées cartésiennes.

Afin d'illustrer cette approche, nous avons mis en oeuvre une expérience (voir figure 7) d'atteinte d'une sphère par la main droite, à partir de la simple spécification d'un ensemble de distances, sans connaître précisément les points à atteindre. L'expérience est réalisée avec un squelette détaillé de la main. A partir d'une posture clé de référence, l'objectif est d'amener l'ensemble des articulations de la main droite à toucher la sphère placée dans la scène 3D. Au delà des longueurs des segments, nous avons considéré des contraintes dures qui expriment la tâche à travers les contacts entre les 26 articulations de la main et la sphère. Ainsi, toute articulation de la main en contact avec la sphère conduit à une contrainte de distance qui s'exprime sous la forme d'une fonction objectif du même type que (7).

Cet exemple montre l'intérêt d'incorporer des priorités entre les contraintes, sous la forme de contraintes dures et de contraintes souples (tâches secondaires). Ne pas spécifier de priorité surcontraint le problème, avec le risque qu'il n'y ait pas de solution. Il est possible ainsi, en ajoutant ou enlevant des contraintes, d'influer sur les solutions du problème d'inversion. Si l'on enlève par exemple la tâche secondaire, cela donne plus de latitude au système d'atteindre la cible, mais peut conduire à une posture de la main potentiellement irréaliste.

6.3. Modification de mouvement (cible à atteindre pendant la marche)

Nous mettons ici en évidence la capacité de la méthode à modifier des mouvements d'origine, en introduisant des tâches supplémentaires. Nous montrons également deux phases dans le déroulement de l'animation. La première concerne l'atteinte de l'objectif (positionnement d'une contrainte), la seconde le relâchement de la contrainte qui s'accompagne du retour du squelette vers la posture initiale. Cette expérimentation soulève le problème de la concurrence entre les tâches, ce qui nécessite une gestion de la priorité entre contraintes.

Pour cet exemple, nous reprenons le mouvement de marche de l'expérience de suivi de mouvement. Durant sa marche, nous donnons comme directive au personnage d'atteindre successivement par la main droite, la main gauche

© REFIG 2012.

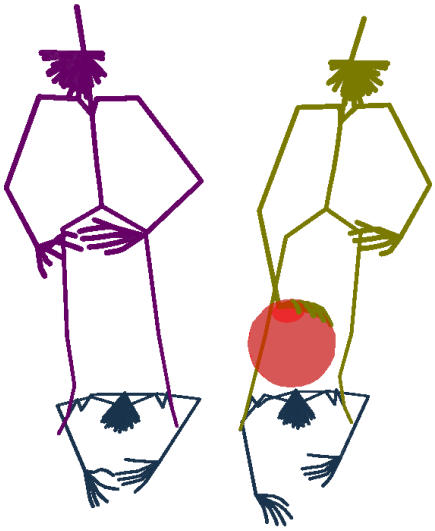

Figure 7: *Sphère à atteindre avec la main droite. A gauche la pose de référence. A droite le mouvement composé.*

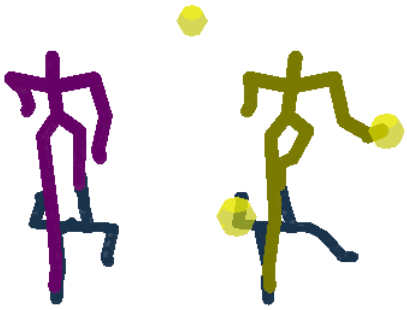

Figure 8: *Illustration de l'adaptation de mouvement. A droite le mouvement référence, à gauche le nouveau mouvement.*

puis à nouveau la droite des cibles qui sont sur son itinéraire. En reprenant les mêmes contraintes que celles de la première expérience, nous considérons cette fois-ci les longueurs des segments comme des contraintes dures et les autres comme des contraintes souples (tâche secondaire). L'objectif d'atteindre des cibles durant l'animation se traduit par l'ajout temporaire de contraintes dures prioritaires sur la tâche secondaire.

La figure 9 met en évidence le conflit entre les contraintes à travers une augmentation de l'erreur de la tâche secondaire lors de l'atteinte des cibles et le bon respect des contraintes dures tout au long de l'animation.

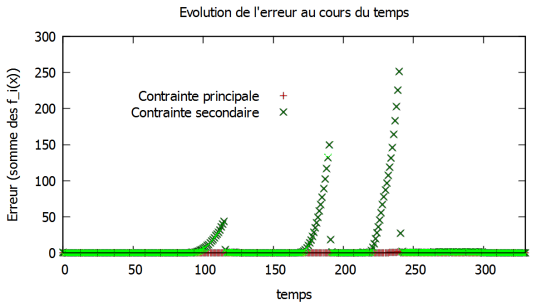

Figure 9: *Evolution des erreurs des tâches principale et secondaire au cours de l'animation. En bleu, l'erreur générée par la tâche principale. En rouge, celle pour la tâche secondaire.*

Stabilité Dans un premier temps, il est important de noter que, contrairement à la cinématique inverse standard où l'orientation d'une articulation est dépendante de l'ensemble des articulations parentes, travailler sur un nuage de points permet d'exprimer un point par rapport à deux voisins dans le cas le plus simple et rarement à plus de quatre ou cinq points pour un squelette (six pour un maillage régulier). Cela implique une très faible dépendance entre les points et donc une répartition plus locale et plus spécifique à la fonction objectif de l'erreur, ce qui rend notre algorithme plus stable. Notons cependant que cette méthode conduit à une convergence plus lente. Dans un second temps, le fait de sur-contraindre le problème entraîne la possibilité d'absence de solution et peut donc conduire à une non continuité entre deux itérations. Notons enfin que nous avons ajouté un paramètre à l'itération de Newton visant à limiter l'implication de la tâche secondaire en fin d'itération, la tâche secondaire servant ainsi de guide vers une solution durant l'itération.

Performance Les contraintes souples sont toutes exprimées vis-à-vis de quatre points de référence statiques (qui ne font pas partie du vecteur d'inconnues), ce qui signifie également qu'il n'y a pas de dépendance entre les trois coordonnées d'un point avec toutes les autres coordonnées du vecteur d'inconnues. Cela induit que notre matrice jacobienne des contraintes souples est une matrice diagonale par blocs. Or l'inverse d'une telle matrice est une matrice diagonale par blocs des inverses des blocs. On peut donc simplifier et finalement accélérer la vitesse de l'inversion de notre jacobienne en calculant indépendamment les pseudo-inverses de chaque sous matrice. Si $\boldsymbol{x}_a$, $\boldsymbol{x}_b$, $\boldsymbol{x}_c$ et $\boldsymbol{x}_d$ sont les points de référence et $\boldsymbol{x}_i$ un point inconnu à déterminer en fonction de ces quatre points de référence, l'expression de la jacobienne devient :

© REFIG 2012.

$$\boldsymbol{J}(\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n) = \begin{pmatrix} \boldsymbol{J}_1 & 0 & \ldots & 0 \\ 0 & \boldsymbol{J}_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \boldsymbol{J}_n \end{pmatrix}, \quad (12)$$

avec les matrices diagonales (4×3)

$$\boldsymbol{J}_i(\boldsymbol{x}_i,\boldsymbol{x}_a,\boldsymbol{x}_b,\boldsymbol{x}_c,\boldsymbol{x}_d) = 4 \begin{pmatrix} f_{ia}(\boldsymbol{x}_i,\boldsymbol{x}_a)(\boldsymbol{x}_i-\boldsymbol{x}_a)^T \\ f_{ib}(\boldsymbol{x}_i,\boldsymbol{x}_b)(\boldsymbol{x}_i-\boldsymbol{x}_b)^T \\ f_{ic}(\boldsymbol{x}_i,\boldsymbol{x}_c)(\boldsymbol{x}_i-\boldsymbol{x}_c)^T \\ f_{id}(\boldsymbol{x}_i,\boldsymbol{x}_d)(\boldsymbol{x}_i-\boldsymbol{x}_d)^T \end{pmatrix}. \quad (13)$$

Cette optimisation du code nous permet d'obtenir des résultats significatifs : pour 31 points à déterminer (soit 93 inconnues) associés à 124 (4×31) contraintes souples et 30 contraintes dures, nous obtenons une fréquence d'affichage de 110 images par seconde dans le cas optimisé pour 51 images par seconde dans le cas non optimisé.

6.4. Application à un graphe cycle

La dernière expérience propose d'étendre la résolution à des problèmes plus larges tels que des graphes contenant des cycles. Pour cette catégorie d'applications nous avons utilisé un graphe constitué de vingt sommets, soit vingt distances dures correspondant aux segments entre les sommets adjacents. Nous avons également ajouté deux distances dures représentant le contact entre deux cibles et deux sommets de notre graphe. Enfin, nous avons mis en opposition deux expériences avec et sans contraintes souples afin de visualiser l'importance de celles-ci dans la déformation spatiale de notre graphe. Comme illustré par la figure 10, la méthode fonctionne bien pour des graphes cycles et nous permet d'opposer les résultats avec et sans contraintes souples. Cette démonstration met en avant le fait qu'à travers la manière de définir les contraintes souples, en plus de garantir des états de référence au cours de la manipulation, il nous est possible de donner un style à la déformation de la chaîne. Notamment il est possible de définir un style relatif à la rigidité de la structure considérée.

Figure 10: *Application à un graphe cycle. Sans contrainte souple (figure de gauche) et avec contraintes souples (figure de droite) : ces contraintes sont représentées par les pointillés en bleu*

7. Discussion et perspectives

Les résultats obtenus démontrent la capacité de la méthode basée distances à résoudre des problèmes de contrôle cinématique du mouvement, dans des tâches diversifiées allant du simple suivi de mouvements capturés à la résolution de problèmes de cinématique inverse et à la modification de mouvements. L'originalité de la méthode réside essentiellement dans la possibilité de définir des tâches sous la forme d'un ensemble de contraintes géométriques qui traduisent de manière simple et pertinente la fonction à réaliser (par exemple adapter la main à l'objet à saisir). Ces contraintes s'expriment en particulier sous la forme de distances qui caractérisent la proximité de points du squelette (distances inter-articulations), ou d'effecteurs à effecteurs ou encore à d'autres points de l'environnement. La formulation du contrôle du mouvement que nous proposons permet ainsi de prendre en compte de manière implicite un ensemble de contraintes qui incluent notamment les contraintes liées à la tâche. L'incorporation de ces contraintes dans les méthodes d'optimisation fournit des solutions optimales et régulières, relativement robustes aux variations de conditions initiales et de tâches. La méthode générale est étendue, afin de séparer la contribution des distances dures (tâche principale) et des distances souples (tâche secondaire), permettant ainsi d'introduire une forme de priorité entre contraintes. De plus, la projection des solutions de la tâche secondaire sur le noyau de la tâche principale permet de garantir le respect de la tâche principale tout en réduisant l'espace des solutions. La méthode permet de résoudre toute une gamme de problèmes, incluant la gestion d'effecteurs multiples et de graphes cycles.

La méthode s'adapte facilement à des tâches d'adaptation du mouvement à partir de données capturées. Cependant, on observe des limitations de la méthode lorsque les contraintes spécifiées sont trop différentes de celles du mouvement original, ce qui peut conduire à une non convergence de l'algorithme, voire à des instabilités. La méthode présentée ne traite pas non plus les aspects temporels des animations. Des améliorations significatives seraient envisageables en couplant aux fonctions spatiales des fonctions temporelles modélisant les synergies articulaires [AJG09]. La méthode pourrait également être étendue afin de traiter des nuages de points issus de maillages ou de données capturées.

Références

[AL11] ARISTIDOU A., LASENBY J. : Fabrik : A fast, iterative solver for the inverse kinematics problem. *Graphical Models. Vol. 73*, Num. 5 (2011), 243 – 260.

[BB04] BAERLOCHER P., BOULIC R. : An inverse kinematics architecture enforcing an arbitrary number of strict priority levels. *The Visual Computer. Vol. 20*, Num. 6 (2004), 402–417.

[Ber92] BERRY M. W. : Large scale sparse singular va-

© REFIG 2012.

lue computations. *International Journal of Supercomputer Applications. Vol. 6* (1992), 13–49.

[BMT96] BOULIC R., MAS R., THALMANN D. : A robust approach for the control of the center of mass with inverse kinetics. *Computers & Graphics. Vol. 20*, Num. 5 (1996).

[CA08] COURTY N., ARNAUD E. : Inverse kinematics using sequential Monte Carlo methods. In *International Conference on Articulated Motion and Deformable Object* (Port d'andratx, Espagne, 2008), pp. 1–11.

[CB06] CALLENNEC B. L., BOULIC R. : Interactive motion deformation with prioritized constraints. *Graphical Models. Vol. 68*, Num. 2 (2006), 175–193.

[CBT07] CARVALHO S., BOULIC R., THALMANN D. : Interactive Low-Dimensional Human Motion Synthesis by Combining Motion Models and PIK. *Computer Animation & Virtual Worlds. Vol. 18* (2007). Special Issue of Computer Animation and Social Agents (CASA2007), to appear.

[CK00] CHOI K., KO H. : Online motion retargetting. *Journal of Visualization and Computer Animation. Vol. 11* (2000), 223–235.

[CMA01] COURTY N., MARCHAND E., ARNALDI B. : Through-the-eyes control of a virtual humanoïd. In *Proc. of Computer Animation 2001* (Seoul, South Korea, novembre 2001), pp. 74–83.

[Dav06] DAVIES R. B. : Newmat c++ matrix library. http ://www.robertnz.net/nm_intro.htm, 2006.

[DW03] DONG Q., WU Z. : A geometric build-up algorithm for solving the molecular distance geometry problem with sparse distance data. *Journal of Global Optimization. Vol. 26* (July 2003), 321–333.

[GJ*10] GUENNEBAUD G., JACOB B., ET AL. : Eigen v3. http ://eigen.tuxfamily.org, 2010.

[Gle98] GLEICHER M. : Retargetting motion to new characters. In *Proc. of the 25th annual conference on Computer graphics and interactive techniques (Siggraph)* (1998), pp. 33–42.

[GM94] GIBET S., MARTEAU P. : A self-organised model for the control, planning and learning of nonlinear multivariable systems using a sensori- feedback. *Journal of Applied Intelligence. Vol. 4* (1994), 337–349.

[GMHP04] GROCHOW K., MARTIN S., HERTZMANN A., POPOVIC Z. : Style-based inverse kinematics. *ACM Tra. on Graphics (Proc. SIGGRAPH). Vol. 23*, Num. 3 (août 2004), 522–531.

[Hav98] HAVEL T. F. : Distance geometry : Theory, algorithms, and chemical applications. *Encyclopedia of Computational Chemistry. Vol. 120* (1998).

[HK09] HO E. S., KOMURA T. : Character motion synthesis by topology coordinates. In *Computer Graphics Forum* (2009), vol. 28, pp. 299–308.

[HKT10] HO E. S., KOMURA T., TAI C. L. : Spatial relationship preserving character motion adaptation. *ACM Transactions on Graphics (TOG). Vol. 29*, Num. 4 (2010), 1–8.

[HR06] HAN L., RUDOLPH L. : Inverse kinematics for a serial chain with joints under distance constraints. In *Proc. of Robotics, Science, and Systems* (2006).

[HRE*08] HECKER C., RAABE B., ENSLOW R. W., DEWEESE J., MAYNARD J., VAN PROOIJEN K. : Real-time motion retargeting to highly varied user-created morphologies. *ACM Transactions on Graphics (TOG). Vol. 27*, Num. 3 (2008), 1–11.

[HSP11] HAUBERG S., STEENSTRUP-PEDERSEN K. : Predicting articulated human motion from spatial processes. *International Journal of Computer Vision. Vol. 94* (2011), 317–334.

[JiCVGHP10] JAUME-I CAPÓ A., VARONA J., GONZÁLEZ-HIDALGO M., PERALES F. : Adding image constraints to inverse kinematics for human motion capture. *EURASIP J. Adv. Signal Process. Vol. 2010* (January 2010), 4 :1–4 :13.

[JMS07] JERMANN C., MICHELUCCI D., SCHRECK P. : Modélisation géométrique par contraintes. In *Informatique graphique, modélisation géométrique et animation*, Bechmann D., Péroche B., (Eds.). Hermés Science, 2007, pp. 185–210.

[KMA05] KULPA R., MULTON F., ARNALDI B. : Morphology-independent representation of motions for interactive human-like animation. *Computer Graphics Forum, Eurographics 2005 special issue. Vol. 24*, Num. 3 (2005), 343–352.

[LLMM10] LIBERTI L., LAVOR C., MUCHERINO A., MACULAN N. : Molecular distance geometry methods : from continuous to discrete. *International Transactions in Operational Research. Vol. 18* (2010), 33–51.

[Mac90] MACIEJEWSKI A. : Dealing with the ill-conditioned equations of motion for articulated figures. *IEEE Computer Graphics and Applications. Vol. 10*, Num. 3 (1990), 63–71.

[MBBT00] MONZANI J.-S., BAERLOCHER P., BOULIC R., THALMANN D. : Using an intermediate skeleton and inverse kinematics for motion retargetting. *Computer Graphics Forum. Vol. 19*, Num. 3 (2000). ISSN 1067-7055.

[MLLT10] MUCHERINO A., LAVOR C., LIBERTI L., TALBI E.-G. : A parallel version of the branch & prune algorithm for the molecular distance geometry problem. In *AICCSA* (2010), pp. 1–6.

[NH86] NAKAMURA Y., HANAFUSA H. : Inverse kinematics solutions with singularity robustness for robot manipulator control. *Journal of Dynamic Systems, Measures and Control. Vol. 108* (septembre 1986), 163–171.

© REFIG 2012.

[PZB90] PHILLIPS C., ZHAO J., BADLER N. : Interactive real-time articulated figure manipulation using multiple kinematic constraints. In *Proc. of the 1990 symposium on Interactive 3D graphics (I3D'90)* (1990), ACM, New York, USA, pp. 245–250.

[RB09] RAUNHARDT D., BOULIC R. : Motion constraint. *The Visual Computer. Vol. 25*, Num. 5 (May 2009), 509 – 518.

[SK05] SENTIS L., KHATIB O. : Synthesis of whole-body behaviors through hierarchical control of behavioral primitives. *International Journal of Humanoid Robotics. Vol. 2*, Num. 4 (2005).

[SS85] SIPPL M. J., SCHERAGA H. A. : Solution of the embedding problem and decomposition of symmetric matrices. *Proceedings of the National Academy of Sciences. Vol. 82*, Num. 8 (1985), 2197–2201.

[TWC*09] TOURNIER M., WU X., COURTY N., ARNAUD E., REVERET L. : Motion compression using Principal Geodesic Analysis. *Computer Graphics Forum. Vol. 28*, Num. 2 (mars 2009).

[UPBS08] UNZUETA L., PEINADO M., BOULIC R., SUESCUN Ã. : Full-body performance animation with sequential inverse kinematics. *Graphical Models. Vol. 70*, Num. 5 (2008), 87–104.

[WE84] WOLOVICH W. A., ELLIOT H. : A computational technique for inverse kinematics. In *Proc. of 23rd IEEE Conf. on Decision and Control* (1984), pp. 1359–1363.

[Wel93] WELMAN C. : *Inverse Kinematics and Geometric Constraints for Articulated Figure Manipulation*. Master's thesis, Simon Frasier University, septembre 1993.

[YN03] YAMANE K., NAKAMURA Y. : Natural motion animation through constraining and deconstraining at will. *IEEE Tra. on Visualization and Computer Graphics. Vol. 09*, Num. 3 (2003), 352–360.

[ZB94] ZHAO J., BADLER N. : Inverse kinematics positioning using nonlinear programming for highly articulated figures. *ACM Tra. on Graphics (Proc. SIGGRAPH). Vol. 13*, Num. 4 (1994), 313–336.

© REFIG 2012.