ISB 2013 BRAZIL
XXIV CONGRESS OF THE INTERNATIONAL SOCIETY OF BIOMECHANICS
XV BRAZILIAN CONGRESS OF BIOMECHANICS

A FAST GEOMETRICALLY-DRIVEN PRIORITIZED INVERSE KINEMATICS SOLVER

[1,2,3]Sébastien Moya and [1,2]Floren Colloud
[1]Université de Poitiers, Poitiers, France
[2]Institut Pprime UPR CNRS 3346, Futuroscope, France
[3]Corresponding author email: sebastien.moya@univ-poitiers.fr

SUMMARY

Based on the FABRIK solver (Aristidou and Lasenby 2011), this paper presents how such geometrically determinist Inverse Kinematics process could be advantageously adapted for biomechanical purposes. This study is inspired from the graphics research community which provides interesting computer methods for kinematical features. Two of them, the computation time efficiency and the management of the targets priority, are considered in the view to use them in a biomechanical context. Consequently, the initial FABRIK solver is modified to fit these new exigencies. The proposed solver is compared to the well-known Jacobian Transpose method in terms of the computation time and final posture solution. Based on our results, a physically realistic and real-time prediction of human movements could be based on such Inverse Kinematics process.

INTRODUCTION

Solving Inverse Kinematics (IK) problems is of main interest for biomechanical studies since it defines the segments configurations of the body subjected to some desired kinematical targets. IK analyses are prerequired for forward and inverse dynamics computations, actually widely used in the musculoskeletal modeling context (for a short overview on IK techniques, see [1]). The IK solvers currently used in biomechanics do not ensure both robust real-time applications and an accurate estimation of the kinematics at each step time. Two main problems can be identified: (1) The computation time of the actual IK solvers is far too important; (2) During marker occlusions, these solvers are not edited in a way to priority handle some given kinematical targets but to solve the best joints configurations which can globally fit all targets. Even if some researches in the graphics animation field could handle one of these two features [1-4], none of them handle both. Aristidou and Lasenby [1] recently introduced a geometrically-driven IK solver, called FABRIK, which can perform real-time tracking thank to its computation efficiency. Nevertheless, this new method suffers from a mechanical abnormality in its joints definition (Figure 1a) and could not manage the targets priorities as [4] does. The aim of the present study is to propose a new IK solver based on the work done by [1] and which can handle the two listed problems. This new IK solver is compared to the well-known Jacobian Transpose (JT) IK method in terms of the computation time and final posture solution.

METHODS

The first problem to solve is the parameterization of segments and joints within the FABRIK IK solver. Indeed, this solver can only model a segment with two joints. As a result, for a segment that would have more than two joints, an additional segment must be added in the model. From a mechanical point of view, this modeling is not correct. A joint is a mechanical link between two segments only. As an example, the pelvis is modeled using three segments linked via three joints (Figure 1a). We modified the FABRIK IK solver in order to handle segments with more than two joints based on a correct mechanics parameterization (Figure 1a). Let us note this segment as *subbase segment* and its parent joint (closer to the root) as *subbase joint*. Thus, during the IK process, the subbase segments are processed as shown in the Figure 1c. During the forward reaching, the positions of every child joints (further to the root: here P_2 and P_3) subjected to at least one further target (here t_4 and t_5) in the model are computed, inducing as many possible positions for their subbase joint as their number. The subbase segment (S_1) is then translated from the previous position of its subbase joint ($P_1(i)$) to the mean position ($P_1(f)$) of these possible positions. Let us note M(i) and M(f) the mean position of the child joints respectively before and after the displacement of this segment. This one is also rotated in the plane formed by P(f), $a = P \;\iota\; M(\iota)$ and $b = P \;f\; M(f)$ by the angle $\theta = (a; b)$. Then, during the backward reaching, the same procedure is done from the subbase joint to the child ones. The child segments finally start from the new positions of the child joints of the subbase segment and the iteration finishes when the end-effectors find their new positions with the classical procedure.

The second problem is to include the management of the targets priority in our proposed IK solver based on the work of [4] who proposed to insert a priority loop inside the convergence loop. Unlike the intuitive way, it appears that beginning by the weakest priority in this loop ensures the strongest one to be preferentially fulfilled. Indeed, every new run through this loop breaks the previous constraints to fit the new ones. In our proposed IK solver, the inputs are the architecture of the model (joint constraints structure, close-loop constraints), its geometry (segments lengths), the desired tasks (the targets with their associated joints and their given priorities) and an initial posture (positions of every joint). The iterative process generates a new posture based on the inputs (Figure 1b). A simple kinematic chain composed by six segments including one subbase segment is generated under the Matlab© software (Figure 2). This structure can be visually assimilated to the upper part of the body (from the pelvis to the wrists). The initial posture and tolerance are fixed. The test consists in positioning the two end-effectors the closest as possible of two unreachable targets. Results from our new IK solver implemented with and without loop priorities and those from the classical Jacobian Transpose (JT) IK method are compared in terms of computation time and final posture solution.

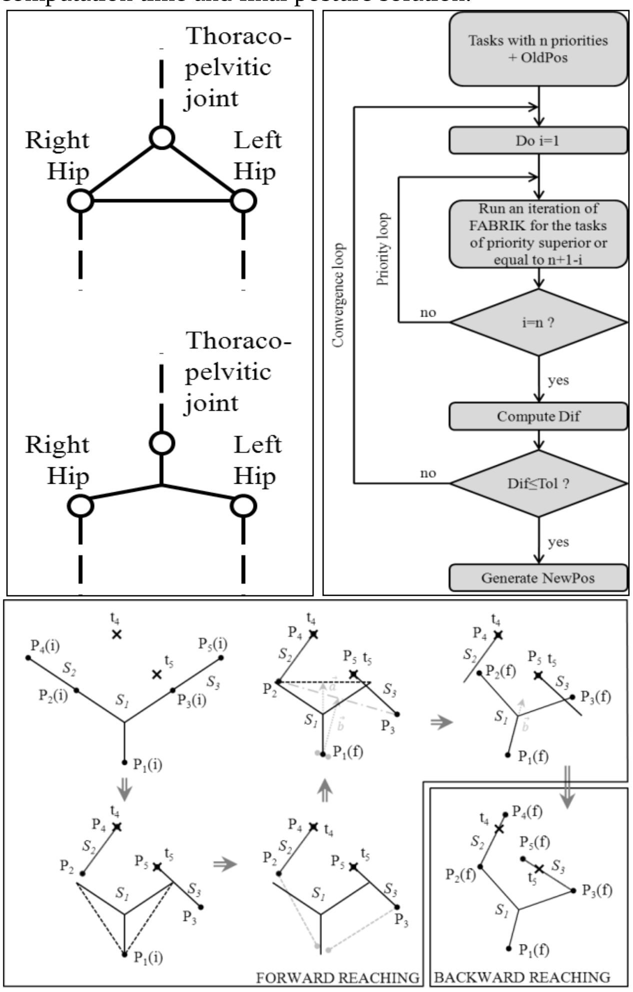

Figure 1: An overview of the proposed IK solver. *(a: Top left)* Two different modelizations of the pelvis. The top one, used in [1,5], restricts every segment to two joints and does not respect the mechanical principle of what are joints. The bottom one is its modelization in the proposed IK solver based on rigorous mechanical parameterization. *(b: Top right)* Structure of the proposed IK solver with priorities. *(c: Bottom)* A full iteration of the proposed IK solver with a model containing a subbase segment.

RESULTS AND DISCUSSION

Results of the proposed IK solver compared to the JT ones, give the results recorded in the Figure 2. The tests concerned the structure previously described. In terms of computation time (non-compiled Matlab© environment), the JT method is more than 25 times slower than the modified FABRIK IK solver without its priority loop. Even when the priority loop was added, the computation time is still 7 times faster. Moreover, the JT method solved the IK problem in 2D while our proposed method did it in 3D. Concerning the solutions, *i.e.* the final postures, the JT does not converge to the best natural posture. In comparison, our proposed IK solver implemented without priorities shows a final posture centered between the two targets. With priorities, the final posture is bent to the highest priority target quite in a straight line while trying to satisfy the lowest priority target at best.

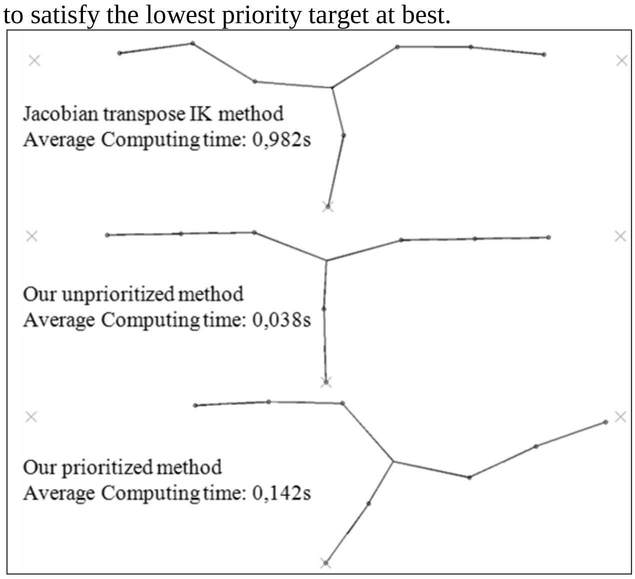

Figure 2: Final postures and computation times for the three Inverse Kinematics algorithms tested.

Our results show that intuitive and natural postures could be found in a real-time environment with this class of IK solver. The joints ranges of motion (RoM) were not implemented inside our proposed IK solver yet. In other words, unrealistic postures from an anatomic point of view can be found. However, since the principle of the FABRIK solver is geometrically-driven in Cartesian coordinates, the direction vectors of every segment are already generated within each iteration. Hence, RoM bounds can be easily implemented. These joint constraints can be test at each step time by computing the Cartesian relative orientation of two adjacent segments. If the solution find is over the RoM bounds, the reorientation procedure is realized. Further test should also be undertaken to characterize our IK solver using orientation targets and limiting by measured RoM. Moreover, when this IK solver is validated, it will be possible to gradually decrease the number of markers needed to be tracked in the view to find a set which is significantly poor but which provides accurate results. The purpose behind this work is to estimate real-time three-dimensional kinematics using a minimal instrumentation and time of post-processing to analyze human movements performed in natural environments.

CONCLUSION

Even with many possible improvements, this paper presents a promising IK algorithm based on the mix of the graphics community results and the needs of the biomechanics one.

ACKNOWLEDGEMENTS

The main author warmly thanks JC for his constant help.

REFERENCES

1. Aristidou A, Lasenby J, *Graphical Models*. **73**:243-260, 2011.
2. Engell-Nørregard M, Erleben K, *Computer & Graphics*. **35:2**:288-298, 2011.
3. Tolani D, et al., *Graphical Models*. **62**:353-388, 2000.
4. Baerlocher P, Boulic R, *The Visual Computer*. **20**:402-417, 2004.
5. Huang J, Pelachaud C, Motion In Games Conference MIG, Rennes, France, 2012.