Forward and Backward Reaching Inverse Kinematics (FABRIK) Solver for DHM: A Pilot Study

Maurice Lamb[1], Seunghun Lee[2], Erik Billing[1], Dan Högberg[1], and James Yang[2]

[1] *University of Skövde, Sweden*

[2] *Texas Tech University, United States*

Abstract

Posture/motion prediction is the basis of the human motion simulations that make up the core of many digital human modeling (DHM) tools and methods. With the goal of producing realistic postures and motions, a common element of posture/motion prediction methods involves applying some set of constraints to biomechanical models of humans on the positions and orientations of specified body parts. While many formulations of biomechanical constraints may produce valid predictions, they must overcome the challenges posed by the highly redundant nature of human biomechanical systems. DHM researchers and developers typically focus on optimization formulations to facilitate the identification and selection of valid solutions. While these approaches produce optimal behavior according to some, e.g., ergonomic, optimization criteria, these solutions require considerable computational power and appear vastly different from how humans produce motion. In this paper, we take a different approach and consider the Forward and Backward Reaching Inverse Kinematics (FABRIK) solver developed in the context of computer graphics for rigged character animation. This approach identifies postures quickly and efficiently, often requiring a fraction of the computation time involved in optimization-based methods. Critically, the FABRIK solver identifies posture predictions based on a lightweight heuristic approach. Specifically, the solver works in joint position space and identifies solutions according to a minimal joint displacement principle. We apply the FABRIK solver to a 7-degree of freedom human arm model during a reaching task from an initial to an end target location, fixing the shoulder position and providing the end effector (index fingertip) position and orientation from each frame of the motion capture data. In this preliminary study, predicted postures are compared to experimental data from a single human subject. Overall the predicted postures were very near the recorded data, with an average RMSE of 1.67°. Although more validation is necessary, we believe that the FABRIK solver has great potential for producing realistic human posture/motion in real-time, with applications in the area of DHM.

Keywords: Inverse Kinematics, Posture Prediction, IK validation, FABRIK

Introduction

Predicting human posture is one of the core functions of many digital human modeling (DHM) tools. However, because human biomechanical systems consist of many highly redundant degrees of freedom posture prediction is a particularly difficult challenge (Aristidou et al., 2018; D'Souza et al., 2001; Yang et al., 2004). Further, to make DHM tools easy and efficient to use, many tools provide control over manikin posture in the form of control points, typically located on end effectors (EE). While this simplifies control, it also means that the DHM tool must solve an inverse kinematics (IK) problem in which it must identify and select a plausible human posture given a possibly infinite set of posture solutions. Most modern approaches to IK posture prediction take inspiration from robotics where highly precise and optimized solutions are the focus (Aristidou et al., 2018; De Magistris et al., 2013; D'Souza et al., 2001; Yang et al., 2004). IK solvers in robotics are often guaranteed to find a plausible solution in finite time if there is one, but they can be slow and sometimes difficult to understand to end users. In this paper we introduce and explore one candidate IK solver developed in the context of computer graphics and recently extended to robotics. The Forward and Backward Reaching Inverse Kinematics (FABRIK) solver has been deployed in multiple computer game engines and 3D design contexts as a real-time posture prediction method (Aristidou et al., 2016; Aristidou & Lasenby, 2011). To our knowledge, the FABRIK solver has not been validated in a DHM context. Initial indications using human skeleton models for games animations are that it can produce plausible human postures (Aristidou et al., 2016). However, accurate predictions typically require appropriate biomechanical constraints and consistent coordinate conventions. The aim of this paper is to introduce the FABRIK solver to the DHM community, provide a pilot demonstration of its feasibility, and discuss some of the challenges in moving the solver from computer graphics to DHM contexts.

Optimization-Based IK Solutions

A posture, for the current purposes, is defined by a biomechanical model (joints and spatial relations between them) and the states of the joints (specified in local angles). Often DHM tools are used to predict and evaluate postures given a particular task. If the user knows all the joint angles, then the posture can be specified using forward kinematics methods to apply the joint angles to each joint moving from a root joint out towards the EEs. While forward kinematic methods can be useful in some instances, it is rare for users to know all the joint angles needed to specify a posture used to accomplish a specific task. In the typical case, only the position and orientation of a few control points, typically EEs, can be known or reasonably anticipated and the DHM tool needs to predict a feasible posture that can meet those constraints. Given the expected EE states, some form of IK method can be applied to estimate a good solution or set of solutions. One method for solving IK problems, e.g., the possibility for infinite valid solutions, in DHM software is to identify a set of constraints on the biomechanical system that can be optimized (Howard et al., 2012; Yang et al., 2004, 2011; Yang & Ozsoy, 2020). Taking this approach, the set of solutions can be limited to those that meet the optimization criteria and search/selection criteria can be used to identify an optimal or nearly optimal solution. Once a solution is identified it can be used to specify the joint configuration of a multi-joint system where the EEs are positioned and oriented according to the specified goal. Optimization methods are highly effective and can produce results for arbitrarily complicated joint systems as well as balance multiple optimization objectives and constraints using multi-objective optimization methods. A challenge for optimization-based methods is that they can be computationally expensive and must be front loaded theoretically, that is they cannot identify a solution without first specifying which biomechanical constraints to optimize and then defining what is optimal. For many DHM applications, it is tricky to formulate optimization criteria that produce results similar to real human motion. While optimization may be ultimately indispensable, alternative approaches to identifying IK solutions may facilitate the formation of a cluster of tools which can quickly and accurately converge on a posture prediction with minimal theoretical front-loading. To this end, the FABRIK solver may provide a fast and minimally theory laden approach to identifying plausible IK solutions.

FABRIK

While the FABRIK solver was introduced to provide a fast and lightweight IK solver for computer graphics applications, it has been implemented in many domains for both pre-recorded and real-time solvers for human and non-human animations (Aristidou & Lasenby, 2011, Aristidou et al., 2016; Lansley et al., 2016). Recently, FABRIK has been extended for application in robotics for both fixed position and mobile multi-joint robots (M. Santos et al., 2021; P. C. Santos et al., 2020; Tao et al., 2021; Tenneti & Sarkar, 2019).

FABRIK is a heuristic IK solver meaning that it identifies a valid posture by applying a limited set of rules to transform an initial posture into a final posture where the EE(s) is in the specified final posture (See Figure 1 for algorithm sketch). An advantage of this approach is that very little information beyond the structure of the kinematic chain is needed, i.e., segment length, relative joint position, joint type, and joint range limits. This means that a solution can be identified quickly and with minimal theory leadenness. While FABRIK is correspondingly not guaranteed to provide a feasible or plausible solution, current applications of the solver in design and game development contexts suggest that it can provide good solutions in the vast majority of human motion prediction problems (Aristidou et al., 2016; Aristidou & Lasenby, 2011; M. Santos et al., 2021; Tao et al., 2021).

The FABRIK solver works in joint position space and is applied hierarchically to each joint and iteratively until a solution is identified. At its core the FABRIK solver involves moving each joint the shortest distance possible given the expected re-positioning of the previous joint in the kinematic chain and while adhering to the biomechanical constraints of the system (e.g., segment lengths and joint limits). A 2D application of the FABRIK solver is illustrated in FIGURE 1. A single iteration of the FABRIK solver starts at the EE and is applied to each joint in the kinematic chain moving towards a root joint. In the first step, the EE is repositioned and aligned to the new EE target (Figure 1A-C). Each joint moving from EE to root is then repositioned along the shortest line that can be drawn between the new position of the joint above and the current joint position while preserving bone length (Figure 1D). Joint limits are applied locally while repositioning joints to ensure that they are not violated in the predicted posture. Once the root node is reached the process is repeated in the reverse direction along the chain (Figure 1E). Assuming the root node is fixed, it is repositioned to its initial position and the process continues back to the EE. This process can be iterated until either the EE is within a specified tolerance of the goal, or a specified number of iterations has been completed. Several modifications and variations of the FABRIK solver have been developed that allow for multi-chain/branching systems, non-fixed root nodes, whole system repositioning, handling unsolvable targets, and obstacles (Aristidou et al., 2016; P. C. Santos et al., 2020; Tao et al., 2021). In the current project we used only the original algorithm along with a hinge constraint applied to the elbow as discussed in Aristidou and Lasenby (2011).

One aim of this project is investigating the extent to which the FABRIK solver can identify a plausible solution given minimal information about the system. As such, the shoulder and wrist joints in the 7 DOF arm model used below are unconstrained 3 DOF joints and the elbow is a 1 DOF hinge joint with a range of motion of 0-120°. An unconstrained version of the FABRIK solver treats all joints as unconstrained ball joints. The elbow joint requires introducing constraints to limit to a hinge joint and allows for a discussion of applying joint constraints in FABRIK. A 1 DOF hinge constraint can be implemented in the FABRIK solver by a method which limits the possible hinge joint positions to a plane partial defined by a line connecting the joint before and the joint after the hinge joint in the kinematic chain (Aristidou & Lasenby, 2011; M. Santos et al., 2021). The discussions of hinge joints in these papers center on general computer avatar or non-humanoid robots respectively, and do not consider how existing human biomechanical limits might affect the results. Notably while the primary plane axis is defined according to the line segment that intersects the wrist and shoulder joints for a human arm, the selection of a secondary and corresponding orthogonal axis must be specified. Given only the primary plane axis there are infinitely many planes to project the joint onto and the selection of plane can have a significant impact on predicted elbow position and orientation as well as the orientation of the other joints. For the study below we chose the secondary axis based on the fixed geometric relationships between the most recently predicted joint and the elbow. In this case, we could use the second axis of the wrist (joint 6 in Table 1) and the first axis of the shoulder (joint 1) respectively. The plane normal was then defined as the cross product of the selected secondary and the primary axes. This plane definition could be verified by testing a small set of samples and ensuring that the predicted rotational axis of the elbow was parallel to the second rotational axis of the shoulder (joint 2) as specified by the arm model in defined by Figure 2 and Table 1 and observed in the recorded data. Biomechanically this is the result of the fact that the rotational axis of the elbow is rigidly coupled to the shoulder joint. We acknowledge that the human shoulder is a relatively complicated joint and that the current model simplifies this joint greatly. However, even without a more representative shoulder joint, the initial testing of the hinge joint constraint allowed for plausible predictions.

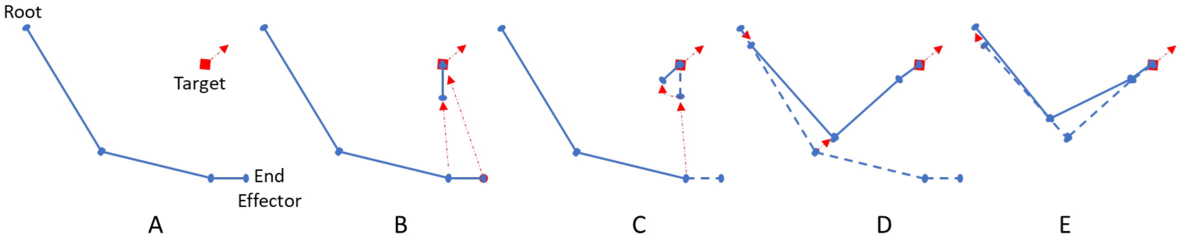


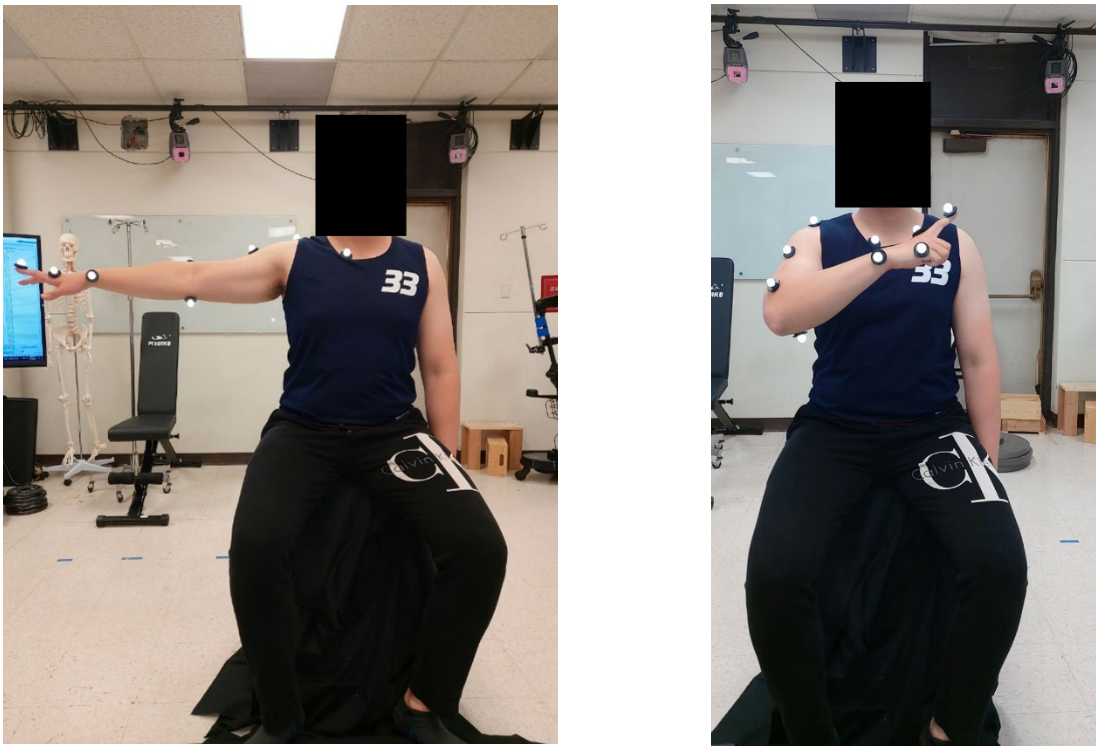
Figure 1. One iteration of the FABRIK solver. The target location and orientation are specified (A) along with an initial arm posture. EE is moved to the target and the other end of segment is moved along a line specified by its initial position and the new target position preserving bone segment length (B). The segment is aligned with the target orientation respecting local rotational constraints (C). Then the process repeated for each joint to the root (D). Since the root is likely moved by the final step, it is returned to its initial position and the process is repeated in reverse to the EE (E). This entire process is repeated until the EE is within the set tolerance or iteration limits.

Methods

In order to determine the initial feasibility of a FABRIK based DHM solution we collected motion capture data for a simple single arm reaching task from a single human actor. The aim of this pilot study is to apply FABRIK to a relatively simple 7 DOF arm model to see how well it can predict the actual arm postures during the reaching motions.

Motion Capture

The data was collected for 19 directed reaching motions performed by a single participant at Texas Tech University Human-Centric Design Research Lab. 7 Eagle-4 camera system (Motion Analysis Corporation, CA) were utilized in this experiment, and each of them has 40megapixel resolution with 500 frames per second. For one arm motion capture, 9 retroreflective markers were attached to the participant's right shoulder and arm. Each reaching motion consisted of sitting in a neutral position and then reaching to a predefined target position in front of the participant. Each reach to target from the initial posture is treated as a single task instance, starting with an initial neutral posture, and moving to an extended final posture. A total of 6 reach targets were used, with each reach cycle repeated 3 times. For data processing, Cortex (Motion Analysis Corporation, CA) was used to smooth the labeled marker movements at 60 Hz.


[redacted]
[redacted]

(A) (B)

Figure 2. Marker protocol: neutral pose (A) and reaching to a predefined target (B) observed from the front view

Figure 2 shows the marker locations. The markers were placed on the participant's sternum, back, and arm to track each relevant joint center (shoulder, elbow, and wrist) and the index fingertip. The marker data was converted to the joint angles using Visual 3D (C-Motion, MD), and a 7 DOF arm model (Figure 3) was defined with Deviant-Hartenberg (DH) parameters (See Table 1).

Simulations using the FABRIK method were carried out in Matlab 2021b. The shoulder was treated as the root node and fixed to the origin of the simulation space. The EE positions and orientations were extracted from the experimental data as the control signal for the simulated arm. All simulations were initialized using the configuration of the arm at the beginning of a simulated task instance, setting the arm joints to match the initial recorded joint configuration. After initialization FABRIK was used to solve the next recorded arm configuration based on the EE's configuration in the next recorded frame. For each subsequent step of the simulation the FABRIK solver used the previous simulated arm configuration as the initial arm configuration. Based on initial testing, the FABRIK solver was run until the predicted arm configuration placed the simulated EE <1mm of the target position or the FABRIK solver ran 500 iterations. The final simulated arm configuration was stored as 7 angles matching the conventions in Table 1 for each simulated frame. Timing and iteration counts were also stored, though because computational efficiency was not explicitly considered in the development of the test software, timing is not indicative of best-case performance.

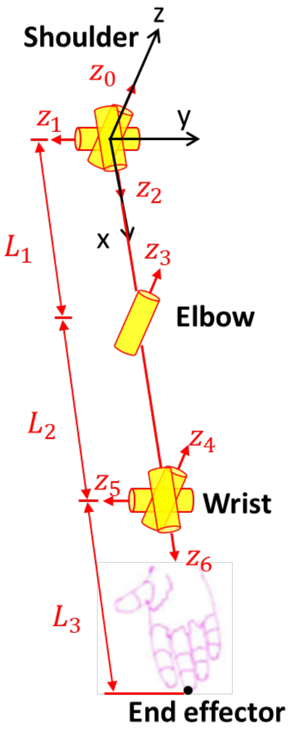


Figure 3. Arm Model (See Table 1).

Table 1

DH-Parameters for arm model

Joint (i)	θ_i (rad)	d_i (cm)	α_i (rad)	a_i (cm)
1	$0 + q_1$	0	$\pi/2$	0
2	$\pi/2 + q_2$	0	$\pi/2$	0
3	$\pi/2 + q_3$	L_1	$\pi/2$	0
4	$\pi/2 + q_4$	0	0	L_2
5	$0 + q_5$	0	$\pi/2$	0
6	$\pi/2 + q_6$	0	$\pi/2$	0
7	$0 + q_7$	L_3	0	0

After simulations were completed, the experimental and simulated arm configurations at each frame were compared for each task instance. Comparisons were quantified by calculating root mean square error (RMSE) for each joint angle (θ_n) that specified the arm configuration,

$$err(\theta_n) = \sqrt{\sum_{i=2}^{len}\left(\theta_i^{en} - \theta_i^{sn}\right)^2} \tag{1}$$

where $\theta_i^{en}$is the experimental angle on the n^{th} joint on the i^{th} frame and θ_i^{sn} is the corresponding simulated angle during that frame. Each task consisted of len number of frames. The first frame is excluded from the calculation of θ_n because it is taken from the recorded data.

Results

The FABRIK solver was able to identify a solution for every target in the experimental data. The provided solutions placed the EE within an average of 0.007 mm of the target position with a max distance to EE goal of 0.013 cm. RMSE for each joint was calculated between predicted and recorded joint angles for each trial and the average and standard deviation RMSE across trials for each joint DOF is presented in Table 2. Across all joints the average RMSE was 1.67°. While the code used for the current study was not optimized for speed, on average the FABRIK solver converged on a solution at each frame in 0.03 seconds and with an average number of 24 iterations.

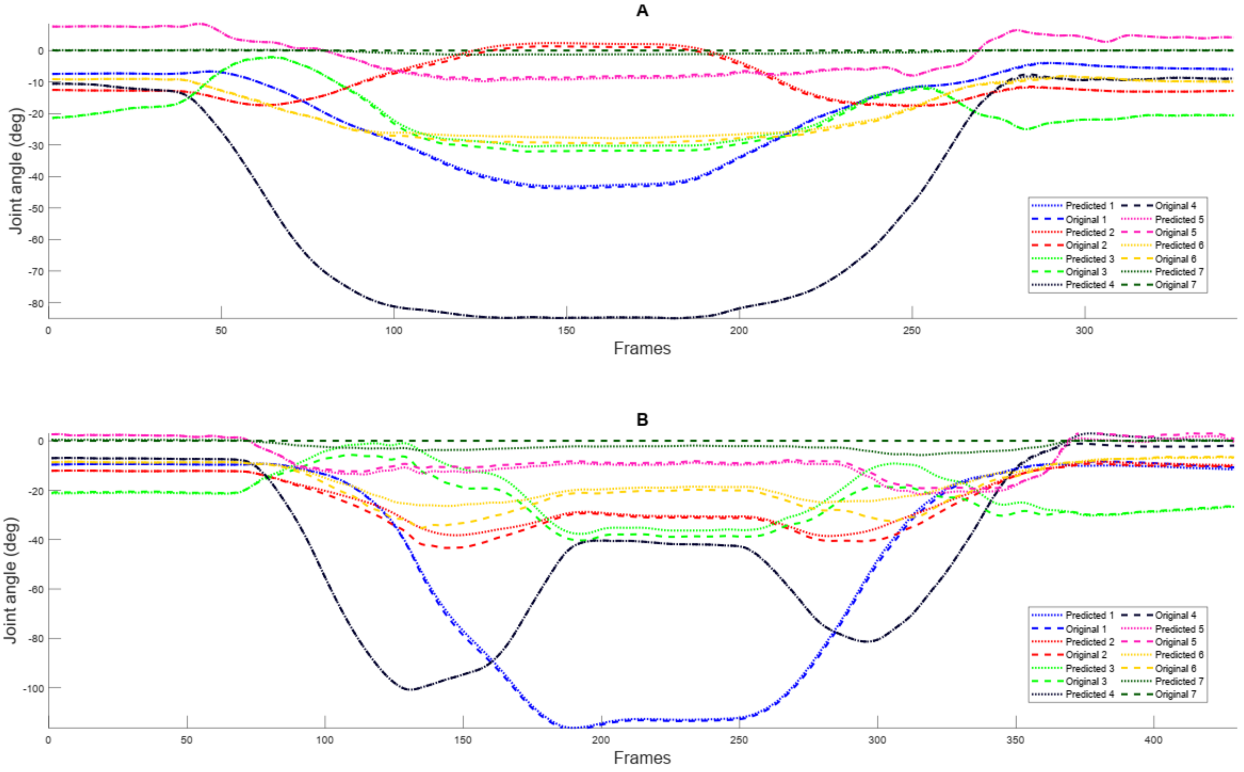


Figure 4. Joint angle plots for 2 trials. Dashed lines are the data from the motion-capture and dotted lines are the predictions. When run through the forward kinematics model, the top plot, A, was visually indistinguishable from the original, the bottom plot, B, showed some visible separation in the elbow joint around frame 275.

The predicted values were run through a forward kinematic solver and visualized as animations in Matlab for visual inspection. Overall the predicted postures were very near the recorded solutions with only a few trials where there was clear seperation between the predicted and recorded postures. Two representtive plots of the predicted and recorded joint angles are presented in Figure 4 to illustrate the predictions for a case with no easily visible seperation (Figure 4A) and a case with a clear moment of seperation (Figure 4B).

Table 2

RMSE between predicted and experimental joint angles^{ab}

Joint 1	Joint 2	Joint 3	Joint 4	Joint 5	Joint 6	Joint 7
0.99 (2.13)	1.38 (1.73)	2.42 (3.46)	0.33 (0.33)	2.00 (4.65)	2.20 (2.99)	2.34 (4.51)

a. mean (standard deviation) b. all values in degrees.

Discussion and Conclusions

In this pilot study, the FABRIK solver was able to provide plausible predictions of the recorded human arm postures across the entire motion path. While this is only an initial exploration of the feasibility of the FABRIK solver for DHM purposes, we believe it is a compelling indication of its possible application. Implementations of FABRIK in modern game engines can converge on a solution for a full humanoid kinematic chain within 4 iterations and continuously provide real-time solutions at 90 Hz or faster. However, while these implementations of the FABRIK solver have proven effective in entertainment contexts, further work needs to be done to validate and optimize the FABRIK solver for DHM applications to ensure fidelity to real-world postures and motions. For the FABRIK solver to provide results consistently and to be a useful tool, the underlying kinematic chain and biomechanical assumptions for joint ranges and limits must be appropriately implemented. Further, it is unclear how far the FABRIK solver may diverge from plausible or valid solutions in more complicated DHM use cases. Systems and checks to ensure plausibility and validity for DHM applications need to be explored. The speed and flexibility of FABRIK also opens for potentially very powerful solutions when combined with traditional optimization based approaches. Once the initial validity of a lightweight version of FABRIK is determined, we believe that additional insights from optimization methods and approaches may synergistically benefit both and contribute to new insights into human postures and behavior prediction. FABRIK provides static IK solutions which may be valid for DHM when correctly implemented. The minimal theory-ladenness, fast convergence to a solution, and relative simplicity makes it an ideal candidate for testing motion planning and control insights from cognitive science research in DHM tools.

References

Aristidou, A., Chrysanthou, Y., & Lasenby, J. (2016). Extending FABRIK with model constraints. *Computer Animation and Virtual Worlds*, *27*(1), 35–57. https://doi.org/10.1002/cav.1630

Aristidou, A., & Lasenby, J. (2011). FABRIK: A fast, iterative solver for the Inverse Kinematics problem. *Graphical Models*, *73*(5), 243–260. https://doi.org/10.1016/j.gmod.2011.05.003

Aristidou, A., Lasenby, J., Chrysanthou, Y., & Shamir, A. (2018). Inverse Kinematics Techniques in Computer Graphics: A Survey. *Computer Graphics Forum*, *37*(6), 35–58. https://doi.org/10.1111/cgf.13310

De Magistris, G., Micaelli, A., Evrard, P., Andriot, C., Savin, J., Gaudez, C., & Marsot, J. (2013). Dynamic control of DHM for ergonomic assessments. *International Journal of Industrial Ergonomics*, *43*(2), 170–180. https://doi.org/10.1016/j.ergon.2013.01.003

D'Souza, A., Vijayakumar, S., & Schaal, S. (2001). Learning inverse kinematics. *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*, *1*, 298–303 vol.1. https://doi.org/10.1109/IROS.2001.973374

Howard, B., Cloutier, A., & Yang, J. J. (2012). Physics-based seated posture prediction for pregnant women and validation considering ground and seat pan contacts. *Journal of Biomechanical Engineering*, *134*(7). https://doi.org/10.1115/1.4007006

Lansley, A., Vamplew, P., Smith, P., & Foale, C. (2016). Caliko: An Inverse Kinematics Software Library Implementation of the FABRIK Algorithm. *Journal of Open Research Software*, *4*(1), e36. https://doi.org/10.5334/jors.116

Santos, M., Molina, L., Carvalho, E. A. N., Freire, E. O., Carvalho, J. G. N., & Santos, P. (2021). FABRIK-R: An Extension Developed Based on FABRIK for Robotics Manipulators. *IEEE Access*, *9*, 53423–53435. https://doi.org/10.1109/ACCESS.2021.3070693

Santos, P. C., Freire, R. C. S., Carvalho, E. A. N., Molina, L., & Freire, E. O. (2020). M-FABRIK: A New Inverse Kinematics Approach to Mobile Manipulator Robots Based on FABRIK. *IEEE Access*, *8*, 208836–208849. https://doi.org/10.1109/ACCESS.2020.3038424

Tao, S., Tao, H., & Yumeng, Y. (2021). Extending FABRIK with Obstacle Avoidance for Solving the Inverse Kinematics Problem. *Journal of Robotics*, *2021*, 1–10. https://doi.org/10.1155/2021/5568702

Tenneti, R. A., & Sarkar, A. (2019). Implementation of modified FABRIK for robot manipulators. *Proceedings of the Advances in Robotics 2019*, 1–6. https://doi.org/10.1145/3352593.3352605

Yang, J., Marler, R. T., Kim, H., Arora, J. S., & Abdel-Malek, K. (2004). Multi-objective optimization for upper body posture prediction. *Collection of Technical Papers - 10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, 2288–2305. https://scholars.ttu.edu/en/publications/multi-objective-optimization-for-upper-body-posture-prediction

Yang, J., Marler, T., & Rahmatalla, S. (2011). Multi-objective optimization-based method for kinematic posture prediction: Development and validation. *Robotica*, *29*(2), 245–253. https://doi.org/10.1017/S026357471000010X

Yang, J., & Ozsoy, B. (2020). Three dimensional unassisted sit-to-stand prediction for virtual healthy young and elderly individuals. *Multibody System Dynamics*, *49*(1), 33–52. https://doi.org/10.1007/s11044-019-09699-9