15-464/15-664 Reference List for January 25

We began today by reviewing the CCD algorithm for inverse kinematics and talking about some of the ways in which the basic algorithm can be expanded to work with practical systems and to achieve better results.

We began by having a look at this slide deck, which may have been put together by Aryel Beck (if someone knows for sure, let me know), which does a great job of portraying CCD IK in pictures.
http://www.cs.cmu.edu/~15464-s13/lectures/lecture6/InverseKinematicsBeck.ppt

The following references detail the mathematics for the 2D CCD case, including equations and code and are an easy introduction to the topic.

Lander, Jeff. "Oh my god, I inverted kine." *Game Developer Magazine* 9 (1998): 9-14.
http://www.cs.cmu.edu/~15464-s13/lectures/lecture6/jlander_gamedev_sept98.pdf

Lander, Jeff. "Making kine more flexible." *Game Developer Magazine* 1, no. 15-22 (1998): 2.
http://graphics.cs.cmu.edu/nsp/course/15464-s15/www/lectures/lec06/jlander_gamedev_nov98.pdf

The lengthier paper that I showed in class, which discusses many interesting variations on the CCD algorithm, is this one:

Kenwright, Ben. "Inverse kinematics–cyclic coordinate descent (CCD)." *Journal of Graphics Tools* 16, no. 4 (2012): 177-217.
https://alogicalmind.com/res/inverse_kinematics_ccd/paper.pdf

We then went on to discuss a number of Jacobian related techniques, following the discussion in the following paper. This paper is so far my favorite reference for Jacobian related techniques for solving inverse kinematics. I find its mathematical explanations to be very clear:

Buss, Samuel R. "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods." *IEEE Journal of Robotics and Automation* 17, no. 1-19 (2004): 16. http://math.ucsd.edu/~sbuss/ResearchWeb/ikmethods/iksurvey.pdf

You may also find the following survey useful. It collects a tremendous number of references organized by method of solution and is relatively up to date on current research related to Inverse Kinematics.

Aristidou, Andreas, Joan Lasenby, Yiorgos Chrysanthou, and Ariel Shamir. "Inverse Kinematics Techniques in Computer Graphics: A Survey." In *Computer Graphics Forum*, vol. 37, no. 6, pp. 35-58. 2018. https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.13310

My go-to reference for explaining Jacobian based IK techniques and how to put them together into a character animation system is the following paper. This paper uses the Jacobian damped pseudoinverse with nullspace projection to do character posing with a two-level priority system. The point which the user clicks and drags is given top priority as the IK target, and secondary constraints include satisfying constraint points, moving the character towards a reference pose, and resolving joint limits. These secondary constraints are satisfied as well as possible with secondary priority by cleverly making use of the opportunity to project desired goals into the nullspace associated with the top priority motion.

Yamane, Katsu, and Yoshihiko Nakamura. "Natural motion animation through constraining and deconstraining at will." *Visualization and Computer Graphics, IEEE Transactions on* 9, no. 3 (2003): 352-360. http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1207443

The classic Jacobian based approach to IK has been parallelized to work quickly for many degree of freedom characters on a modern GPU. For reference, I suggest this paper, which is also worth reading for one more description of the classic Jacobian based approach to inverse kinematics.

Harish P, Mahmudi M, Callennec BL, Boulic R. Parallel inverse kinematics for multithreaded architectures. ACM Transactions on Graphics (TOG). 2016 May 25;35(2):19. http://dl.acm.org/citation.cfm?id=2887740

In some applications, even faster results may be desired. For fast IK, CCD is the standby approach. However, CCD can create artifacts such as the end effector curling in on itself. The following approach is also very fast and in the same style, but seems to create more visually pleasing and consistent results, at least under some circumstances. We had a quick look at the algorithm behind this approach today.

Aristidou, Andreas, and Joan Lasenby. "FABRIK: a fast, iterative solver for the inverse kinematics problem." *Graphical Models* 73, no. 5 (2011): 243-260.
http://www.andreasaristidou.com/FABRIK.html
https://www.youtube.com/watch?v=tN6RQ4yrNPU