Forward Kinematics

In 2D, a *hierarchical skeleton* is a directed tree $T = (V, E)$ together with a map $\theta : V \to \mathbb{R}$ describing the angle of rotation of each vertex relative to its parent around a center $c : V \to \mathbb{C}$. Let $p : V \to \mathbb{C}$ denote the position of the skeleton induced by r and c, and for convenience let

$$u_i := c_i - c_{i-1}.$$

Consider any vertex $v \in V$; it is connected to the root node $v_0 \in V$ via a sequence of vertices $v_0, v_1, \ldots, v_k = v$. Let c_i and r_i be the corresponding rotations and centers in this sequence (respectively). Assuming v_0 sits at the origin (i.e., $p(v_0) = 0$), the configuration of any other vertex v_i in this sequence can be expressed as

$$p(v_i) = p(v_{i-1}) + R(\Theta_i)u_i.$$

where $R(\Theta_i)$ denotes a counter-clockwise rotation by the angle

$$\Theta_i := \sum_{m=0}^{i-1} \theta_m$$

and

$$p(v_0) := p_0$$

for some translation $p_0 \in \mathbb{C}$ of the root. In other words we get the next position in the sequence by starting at the previous position, and adding the cumulative rotation applied to the difference between consecutive centers. This recursive expression of course implies an efficient algorithm for incrementally evaluating the positions for all vertices in the tree.

Inverse Kinematics

Suppose we want to find angles θ that place vertex i at a target position $\tilde{p}_i$. We can formulate this problem as a variational problem

$$\min_{\theta \in \mathbb{R}^{|V|}} \tfrac{1}{2}|p(v_i) - \tilde{p}_i|^2 =: E(\theta).$$

In practice, we can search for a solution to this problem using gradient descent. In particular, we want to numerically integrate the ODE

$$\dot{\theta} = -\nabla_\theta E.$$

The gradient is given explicitly by components

$$\tfrac{\partial}{\partial \theta_j} \tfrac{1}{2}|\tilde{p}_i - p(v_i)|^2 = (\tilde{p}_i - p(v_i)) \cdot \tfrac{\partial}{\partial \theta_j} p(v_i),$$

where

$$\tfrac{\partial}{\partial \theta_j} p(v_i) = \begin{cases} R(\pi/2)(p(v_i) - p(v_j)), & j < i, \\ 0, & \text{otherwise}, \end{cases}$$

and $R(\pi/2)$ denotes a rotation by $\pi/2$ in the counter-clockwise direction. How do you see this? Well, $p(v_j)$ is the center of rotation, and if we change the angle θ_j, the point $p(v_i)$ must move in a circle around this center. In other words, the velocity must be tangent to the circle; in particular, increasing the angle will result in a counter-clockwise motion. And how fast is that motion? Well, the circumference of any circle is proportional to its radius r. So the distance along the circumference covered per angle of rotation should also be proportional to the radius. In fact, the constant of proportionality is just 1, since for a change in angle of 2π we cover a distance $2\pi r$. In summary: the derivative should be (i) tangent to the circle of rotation, and (ii) equal in magnitude to the radius. But since in this case $r = |p(v_i) - p(v_j)|$, that's the same as saying that the derivative is just the radius vector $p(v_i) - p(v_j)$ rotated by a quarter turn in the counter-clockwise direction.

Overall, then, we have a very simple algorithm for inverse kinematics:

1. Compute the current positions p of all joints.
2. Plug these values into the simple closed-form expression for the gradient.
3. Take a small gradient step.
4. Repeat.