

The Barnes-Hut Algorithm

Christian Swinehart

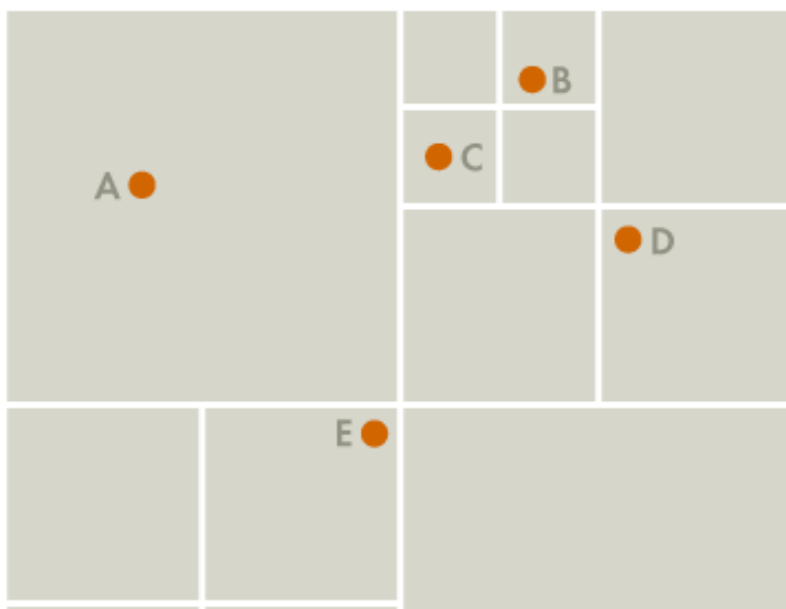
Tom Ventimiglia & Kevin Wayne

The crucial idea in speeding up the brute force n-body algorithm is to group nearby bodies and approximate them as a single body. If the group is sufficiently far away, we can approximate its gravitational effects by using its *center of mass*. The center of mass of a group of bodies is the average position of a body in that group, weighted by mass. Formally, if two bodies have positions (x_1, y_1) and (x_2, y_2) , and masses m_1 and m_2 , then their total mass and center of mass (x, y) are given by:

$$\begin{cases} m = m_1 + m_2 \\ x = (x_1 * m_1 + x_2 * m_2) / m \\ y = (y_1 * m_1 + y_2 * m_2) / m \end{cases}$$

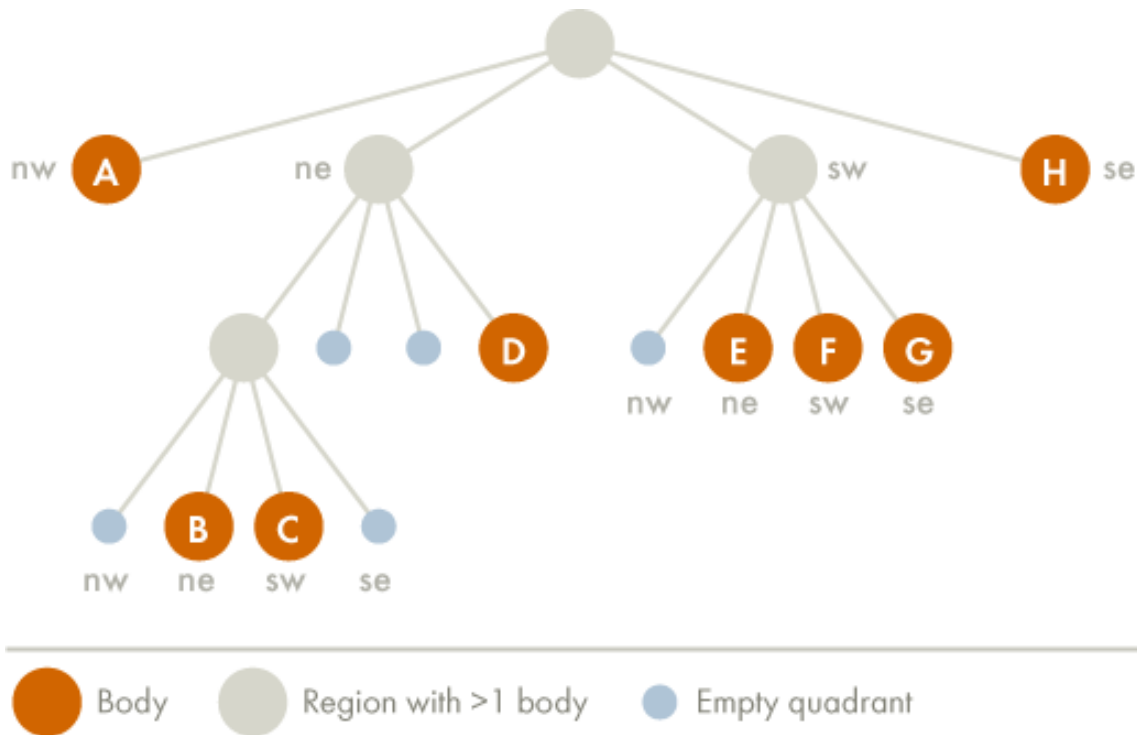
The *Barnes-Hut algorithm* is a clever scheme for grouping together bodies that are sufficiently nearby. It recursively divides the set of bodies into groups by storing them in a *quad-tree*. A quad-tree is similar to a binary tree, except that each node has 4 children (some of which may be empty). Each node represents a region of the two dimensional space.

The topmost node represents the whole space, and its four children represent the four quadrants of the space. As shown in the diagram, the space is recursively subdivided into quadrants until each subdivision contains 0 or 1 bodies (some regions do not have bodies in all of their quadrants). Hence, some internal nodes have less than 4 non-empty children).





Each external node represents a single body. Each internal node represents the group of bodies beneath it, and stores the center-of-mass and the total mass of all its children bodies. Here is an example with 8 bodies:



To calculate the net force on a particular body, traverse the nodes of the tree, starting from the root. If the center-of-mass of an internal node is sufficiently far from the body, approximate the bodies contained in that part of the tree as a single body, whose position is the group's center of mass and whose mass is the group's total mass. The algorithm is fast because we don't need to individually examine any of the bodies in the group.

If the internal node is not sufficiently far from the body, recursively traverse each of its subtrees. To determine if a node is sufficiently far away, compute the quotient s/d , where s is the width of the region represented by the internal node, and d is the distance between the body and the node's center-of-mass. Then, compare this ratio against a threshold value θ . If $s/d < \theta$, then the internal node is sufficiently far away. By adjusting the θ parameter, we can change the speed and accuracy of the simulation. We will always use $\theta = 0.5$, a value commonly used in practice. Note that if $\theta = 0$, then no internal node is treated as a single body, and the algorithm degenerates to brute force.

Constructing the Barnes-Hut tree

To construct the Barnes-Hut tree, insert the bodies one after another. To insert a body b into the

tree rooted at node x , use the following recursive procedure:

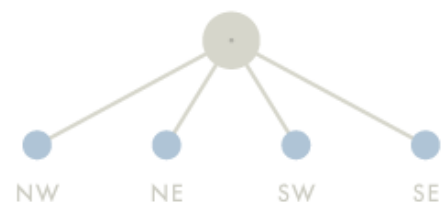
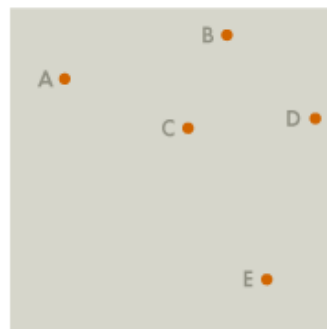
If node x does not contain a body, put the new body b here.

If node x is an internal node, update the center-of-mass and total mass of x . Recursively insert the body b in the appropriate quadrant.

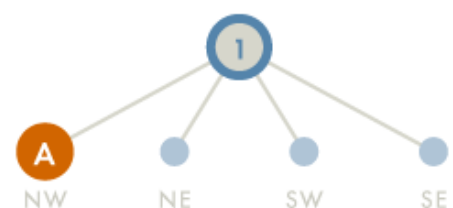
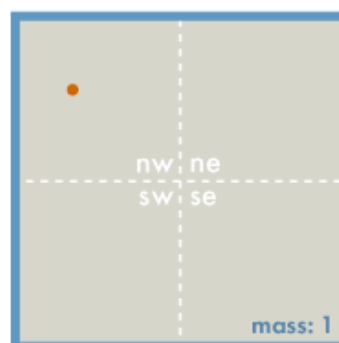
If node x is an external node, say containing a body named c , then there are two bodies b and c in the same region. Subdivide the region further by creating four children. Then, recursively insert both b and c into the appropriate quadrant(s). Since b and c may still end up in the same quadrant, there may be several subdivisions during a single insertion. Finally, update the center-of-mass and total mass of x .

As an example, consider the 5 bodies in the diagram below. In our examples, we use the convention that the branches, from left to right, represent the northwest, northeast, southwest, and southeast quadrants, respectively. The tree goes through the following stages as the bodies are inserted:

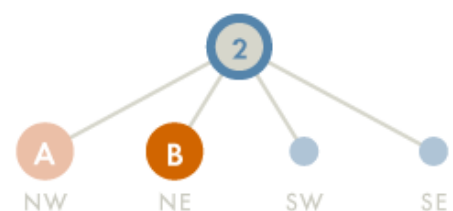
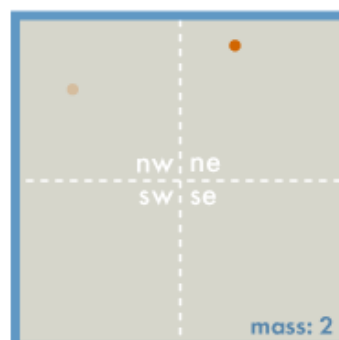
Initial Conditions
5 nodes to be added;
start with empty root node



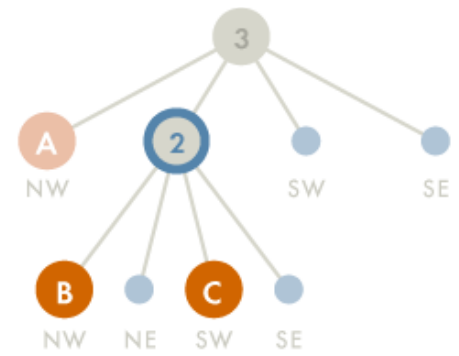
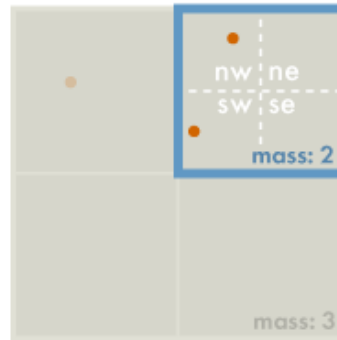
Add body A
fits in nw corner of root node



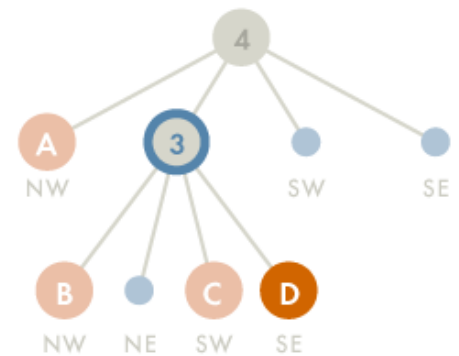
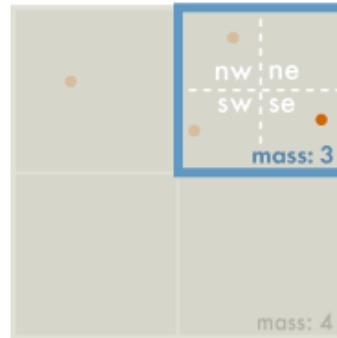
Add body B
fits in ne corner of root node



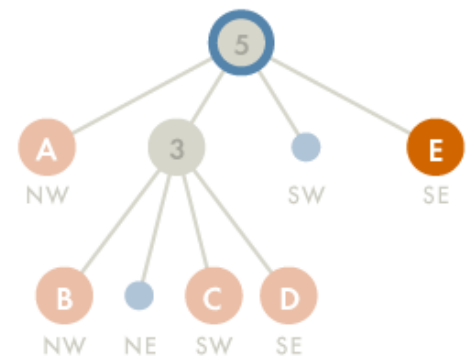
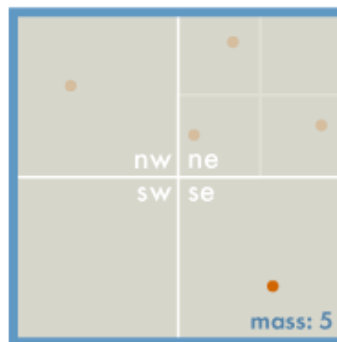
Add body C
ne corner occupied by B;
insert branch and add both
B and C as children



Add body D
fits in se corner of branch node



Add body E
fits in se corner of root node



The root node contains the center-of-mass and total mass of all five bodies. The two other internal nodes each contain the center-of-mass and total mass of the bodies *b*, *c*, and *d*.

Calculating the force acting on a body

To calculate the net force acting on body *b*, use the following recursive procedure, starting with the root of the quad-tree:

If the current node is an external node (and it is not body *b*), calculate the force exerted by the current node on *b*, and add this amount to *b*'s net force.

Otherwise, calculate the ratio s/d . If $s/d < \theta$, treat this internal node as a single body, and calculate the force it exerts on body *b*, and add this amount to *b*'s net force.

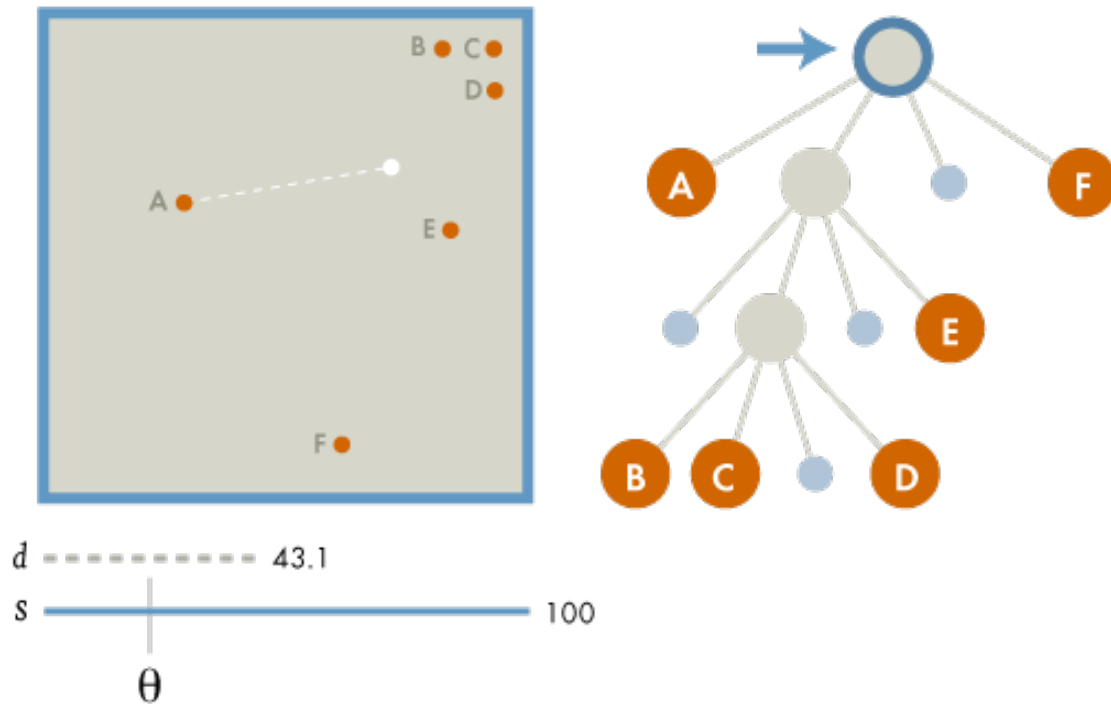
Otherwise, run the procedure recursively on each of the current node's children.

As an example, to calculate the net force acting on body *a*, we start at the root node, which is an internal node. It represents the center-of-mass of the six bodies *a*, *b*, *c*, *d*, *e*, and *f*, which have

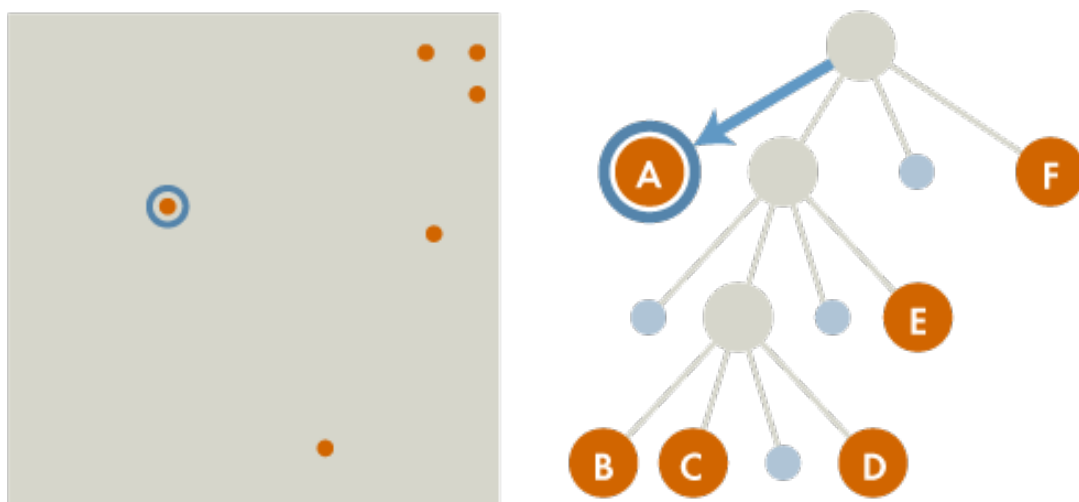
masses 1, 2, 3, 4, 5, and 6 kg, respectively.

The force calculation proceeds as follows:

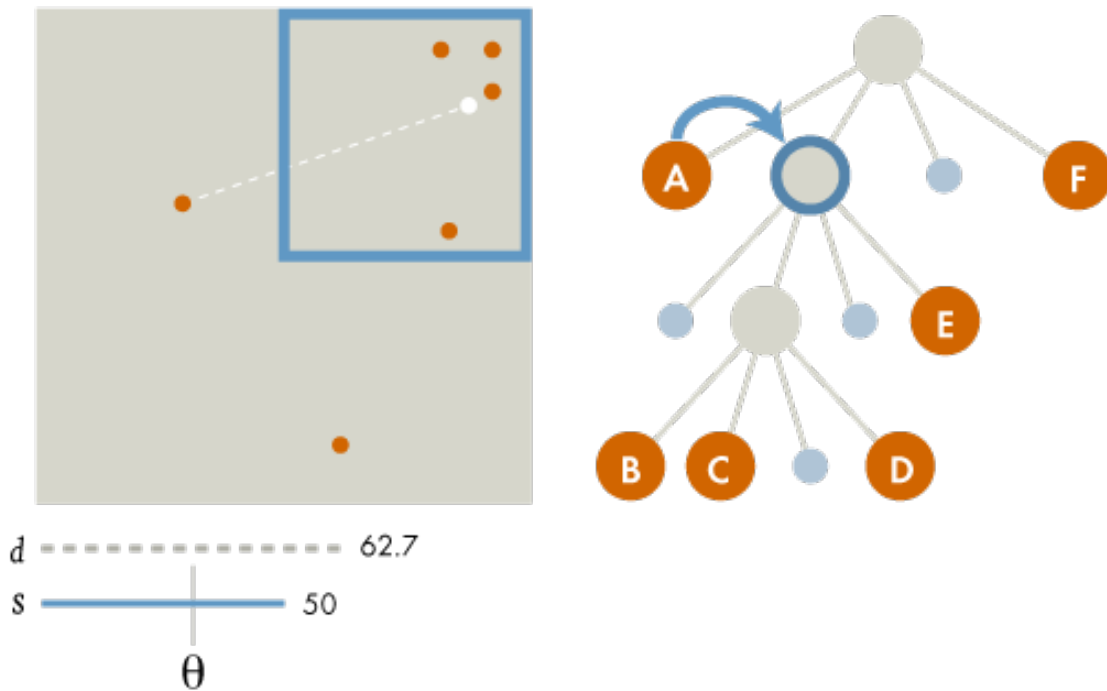
The first node examined is the root. Comparing body A to the node's center of mass (white dot), the quotient $s/d = 100/43.1 > \theta = 0.5$, so we perform the process recursively on each of the root's children.



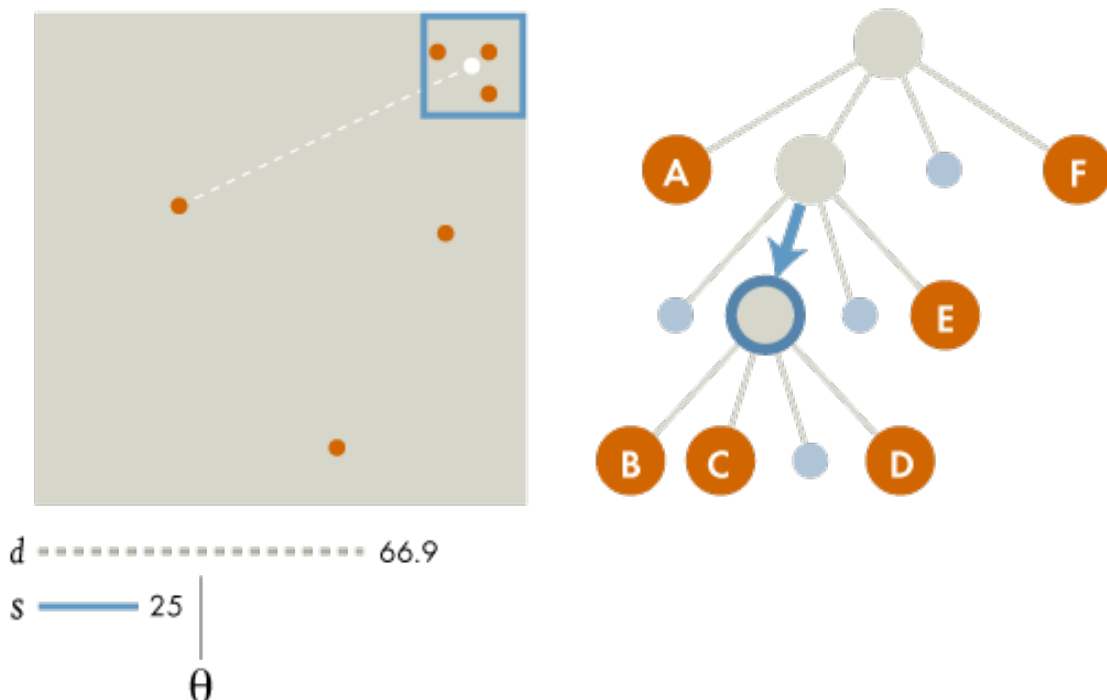
The first child is body *a* itself. A node does not exert force on itself, so we don't do anything.



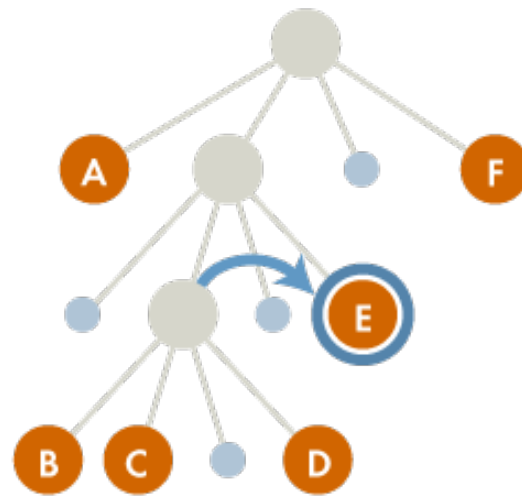
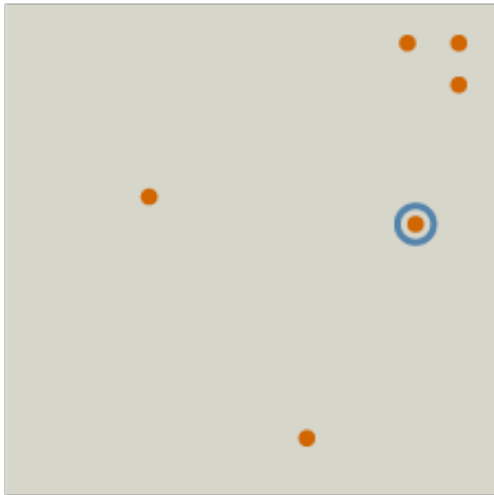
This child represents the northeast quadrant of the space, and contains the center-of-mass of bodies *b*, *c*, *d*, and *e*. Now $s/d = 50/62.7 > \theta$ so we recursively calculate the force exerted by the node's first non-empty child.



This is also an internal node, representing the northeast quadrant of its parent, and containing the center-of-mass of bodies b , c , and d . Now $s/d = 25/66.9 < \theta$. Treating the internal node as a single body whose mass is the sum of the masses of b , c , and d , we calculate the force exerted on body a , and add this value to the net force exerted on a . Since the parent of this node has no more children, we continue examining the other children of the root.



The next child is the one containing body e . This is an external node, so we calculate the pairwise force between a and e , and add this to a 's net force.



Having examined all the siblings at this level, we move on to the next sibling of the parent. This brings us to the node containing body f . Since it is an external node we calculate the pairwise force between a and f , and add this to a 's net force.

