Best First Search for Solving the ILS Problem

The problem is defined as:

$$\min_x \|Rx - y\|_2^2.$$

Assume we have a tree of depth n where each edge has a cost to traverse.

Define a given nodes cost as the sum of the costs of all the edges traversed to reach the node, that is the length of the path between the node and the root. We would like to find the leaf with minimal cost. Define a nodes "next best child" as the child with the lowest cost that has not yet been visited.

In the application of sphere decoding we can quickly find a nodes "next best child", it corresponds to the next choice of x_k chosen in the SE fashion.

In the first step, we define the root node as having 0 cost and being at level n+1. The "next best child" of the root node will correspond to $x_n^1 = \lfloor y_n/R_{n,n} \rceil$ and the cost to visit this child will be equal to the partial residual $(R_{n,n}x_n^1 - y_n)^2$, we will call this cost the "next best child cost". The root node will store its current "next best child" and "next best child cost" and then be pushed into an empty priority queue, elements in this priority queue will always be sorted by their "next best child cost"

In the next step, we will "expand" the first child of the root, x_n^1.

First, pop the root back out of the priority queue and expand it by visiting its "next best child" x_n^1.

Now we must calculate the "next best child" and its cost for x_n^1. The "next best child" will be given by $x_{n-1}^1 = \lfloor (y_{n-1} - R_{n-1,n}x_n^1)/R_{n-1,n-1} \rceil$ and its cost will be the partial residual $(R_{n,n}x_n^1 - y_n)^2 + (R_{n-1,n}x_n^1 + R_{n-1,n-1}x_{n-1}^1 - y_{n-1})^2$. Notice that the first term in the cost is just the cost of the parent node. We then push x_n^1 into the priority queue.

Since we are now visiting the first child of the root, we must generate it's new "next best child", $x_n^2 = \lfloor y_n/R_{n,n} \rceil_{-}^{+} 1$ and the cost for this child (also the partial residual), $(R_{n,n}x_n^2 - y_n)^2$. We may now push the root back into the priority queue with the newly calculated "next best child cost"

Then, the node which will be expanded will be the one at the top of the priority queue with the smallest "next best child cost". We compare $(R_{n,n}x_n^1 - y_n)^2 + (R_{n-1,n}x_n^1 + R_{n-1,n-1}x_{n-1}^1 - y_{n-1})^2$ and $(R_{n,n}x_n^2 - y_n)^2$. If the former is smaller, we will expand x_{n-1}^1 next, otherwise we expand x_n^2.

By proceeding in this way, we always expand the nodes in the order of increasing partial residual. The next node we expand is always the one with the next smallest partial residual (in the whole tree) from the previous one, even if those 2 nodes are at different levels. In this way we can guarantee that the first time we find a leaf in the tree, it must be the leaf with the smallest squared residual and is therefore the solution.

Also, at the point where we find a leaf, we know we have explored only those nodes that have a partial residual within the radius of the optimal hyper-sphere.