These methods (settle\_disputes and find\_parallels) are called after each iteration of construct\_tree. Each can be used with two trees (as in merge\_tree), in which case they will update the first tree with information from the second tree; they can also be used with only one tree (as in construct\_tree), in which case they will update the first tree with information from other parts of itself.

settle\_disputes attempts to determine which of its two possible parents a disputed node belongs to:

for each TreeNode that has disputed children in first tree:

for each TreeNode that does not have disputed children in the second tree:

if the nodes have the same event

(that is, their TMRs have the same concepts for EVENT, THEME,

AGENT, and INSTRUMENT if present) then:

let "question" be the disputed node, "answer" the non-disputed node,

and "other" the node that the disputed node is disputed with

(i.e. the other candidate parent for its children)

make a mapping from the children of answer to the children of other

so that, if one node is mapped to another, their TMRs have the same main event

(or they represent the same action, if they are leaf nodes)

for each child of answer that is mapped to a child of other,

remove that child from other

remove remaining children of other from question

find\_parallels method compares the order of children in nodes representing the same event, and determining whether those children are parallel or sequential.

for each non-disputed node in one tree:

for each non-disputed node in the other tree:

if their TMRs have the same event:

compare the orderings of their children and update their child relationship matrices accordingly:

The child relationship matrix is effectively an adjacency matrix, where each entry [i,j] is one of the following:

"1", representing that the ith child comes before the jth child in all known input,

"-1", representing that the ith child comes after the jth child in all known input,

"0", representing that the ith child and the jth child have been observed in either order or that i=j.

This updating is accomplished by multiplying a[i,j] and b[i,j] by (a[i,j]==b[i,j]) for all values of i and j,

so that if both a[i,j] and b[i,j] were nonzero they will stay unchanged, but otherwise they will both be set to zero.