Recursive Algorithm for completing the alignment of a list of Reference points

We will frame the entire cable installation as a problem of attaching a list of specified reference points. To attach those reference points, we provide the algorithm with the starting system state which includes the configuration of the cables, certain free manipulators and certain manipulators already grasping cables.

A feasible intermediate step is reached when along with one of the specified reference points being aligned, the free manipulators, that the list started with have been freed, no cables are stretched and the interlinks are not stretched.

However, in some instances the resolution of an interlink constraint creates its own set of reference points that need to be aligned, thus invoking the need to call the list alignment function within the resolution function and vice versa. Thus the overall installation algorithm ends up being a recursive algorithm.

This document outlines the pseudo code for both these functions and attempts to mark the entry, and exit points (both successful and unsuccessful) of the algorithm to ensure completeness of the algorithm.

Pseudo Code for Resolve:

Input: CurrentlyClampedCable, OccupiedManipulators, FreeManipulators, ClampedCableManipulator

Determine the violated interlinks

Classify the reference points or Interest and Gripping points to be considered

Assign Free Manipulators to corresponding cables.

Declare unfeasibility if corresponding cables > free manipulators

Try the one step alignment of all reference points of interest on the cable

If successful

Exit

Else

Determine the reference points that need to be aligned on corresponding cables till all the initial violated interlinks are resolved.

Determine the reference points that need to be aligned on other cables based on the links violated here and so on.

Create a list of reference point alignments that must be achieved

While ~IsEmpty(GrippingPointList)

Select a gripping

Delete GrippingPoint

Compute geometrical resolution for given gripping point.

Set success flag

AlignList and set Alignmentflag

If alignment successful

Exit

End if

EndWhile

Pseudo Code for align list

Inputs: SystemNode, OccupiedManipulators, FreeManipulators, RequiredReferencePointsList

Create Array: RemainingReferencePointsList.

Format of Array: RemaningReferencePointsList.CableID

RemainingReferencePointsList.RefID

RemaningReferencePointsList.failedAttempts

Set exit flag to notExit

While exitFlag set to notExit

If isEmpty(RemainingReferencePointsList)

Set exitFlag to exit

Set SuccessFlag to success

break

End if

Select one element of the RemainingReferencePointsList. Delete that element from the list

If

Use AlignRefPoint function to reposition and clamp the cable.

Check for interlink violation.

If no interlinks are violated

Set ResolveFlag to success

Else

ResolveInterlinkConstraint (OccupiedManipulator, ClampedManipulator, SystemNode, FreeManipulators)

Set ResolveFlag to success or failure

End IF

If ResolveFlag is set to success

Set TriesSinceSuccess = 0;

Continue

Else

Add the current reference point to the end of queue,

Increment TriesSinceSuccess;

End if

End while

If FailFlag set to Success

Declare success

Else

Declare failure

End if

Pseudo Code for determine reference points required to be aligned

Inputs: SystemNode, CorrespondingCablesStructure, currentlyViolatedInterlinks, currentClampedCable

Start aligning reference point in order of total distance from all violated interlinks with that corresponding cable

Check for more violated interlinks with the current corresponding cable and determine the secondary corresponding cables

Recall the function for each of the corresponding cables