**Algorithm:** ResolveInterlinkConflict

**Input:** SystemState, FreeManipulatorsList, OccupiedCables, CurrentClampingCable, CurrentClampingManipulator, ViolatedLinkList

**Output:** Advanced SystemState, ResolutionSuccessFlag

1. CorrespondingCables.CableID = Corresponding Cable for every violated interlink
2. **For** selectedCable CorrespondingCables
   1. CorrespondingCables.ViolatedInterlinks
   2. CorrespondingCables.RelevantReferencePoints = Determine relevant reference points
   3. CorrespondingCables.GripPointsConsidered = Determine relevant grip points
   4. **If** (CorrespondingCable.CableID OccupiedCables.CableID
      1. CorrespondingCable.AssignedManipulator = OccupiedCables.ManipID
   5. **Else**
      1. CorrespondingCable.AssignedManipulator = pop(FreeManipulatorsList)
   6. **End**
   7. **if isEmpty(**CorrespondingCable.AssignedManipulator)
      1. ResolutionSuccessFlag = false
      2. Return
3. **End**
4. **If** (CorrespondingCable.CableID OccupiedCables.CableID)
   1. [SystemState SingleStepResolutionList RepositionCablesList] = AttemptSingleStepResolution(SystemState, CorrespondingCables)
5. **End**
6. ViolatedLinkList = CheckInterlink(SystemState)
7. **If** (isEmpty(ViolatedLinkList))
   1. ResolutionSuccessFlag = true
   2. Return
8. **End**
9. [SystemNode GeometricResolutionFlag] = AttemptGeometricResolution(CorrespondingCables, RepositionCablesList, SystemNode)
10. **If** (GeometricResolutionFlag = true)
    1. ResolutionSuccessFlag = true
    2. Return
11. **Else**
    1. ResolutionSuccessFlag = false
    2. Return
12. **End**

**Algorithm**: SingleStepAlignment

**Inputs**: SystemState, FreeManipulators, OccupiedManipulators,

**Output**: SystemState, RepositionCablesList, SingleStepAlignmentList

1. RepositionCablesList = OccupiedManipulators.CableID
2. **If** (n(Corresponding Cables not occupied)) < 0
   1. Return
3. **Else**
   1. **For** selectedCorrespondingCable,CableID (CorrespondingCables.CableID OccupiedManipulators.CableID)
      1. SortedReferencePointList = SortByPriorityValue(CorrespondingCable.RelevantReferencePoints)
      2. CableAlignFlag = false
      3. **For** (ReferencePoints SortedReferencePointList)
         1. [SystemState ParentState] = AlignReferencePointParallel(selectedCorrespondingCable.CableID, ReferencePoint, selectedCorrespondingCable.ManipID)
         2. **If** (AreResolved(selectedCorrespondingCable.ViolatedInterlink)
            1. newCorrespondingCables = DetermineCorrespondingCables(SystemState, selectedCorrespondingCable)
            2. **if** (newCorrespondingCables OccupiedCables.CableID)

SingleStepResolutionList.ReferencePoint.Add(current Reference point, Manipulator and Cable)

CableAlignFlag = true

Break

* + - * 1. Else

SystemState = ParentState;

* + - * 1. **End**
      1. **End**
    1. **End**
    2. **If** (CableAlignFlag = false)
       1. RepositionCables
    3. **End**
  1. **End** a

1. **End**

**Algorithm**: AttemptGeometricResolution

**Input**: RepositionCablesList, CorrespondingCables, SystemNode

**Output**: SystemState, GeometricResolutionSuccessFlag

1. If (Manipulator (FreeManipulators CorrespondingCables.AssignedManipulators))
   1. ReferencePointsToAlignList = DetermineReferencePointsToAlign(SystemState, CorrespondingCables, [])
2. **End if**
3. GeometricResolutionExitFlag = false; GeometricResolutionSuccessFlag = false;
4. **While** (ResolveFlag = false)
   1. **For** (Manipulators FreeManipulators)
      1. **If (**isEmpty(CorrespondingCable.GripPoint)
         1. Sample Grip Points from CorrespondingCable.GripPointsConsidered
         2. SystemState.GraspCableParallel()
      2. Else
         1. Increment ListEmptyCount
      3. **End if** a
      4. Delete selected GripPoint from array
   2. **End For**
   3. [SystemState SuccessFlag] = ComputeGeometricResolution(SystemState, RepositionCableList)
   4. **If** SuccessFlag = false
      1. Continue
   5. **Else**
      1. NewFreeManipulators = DetermineNewFreeManipulators(SystemState)
      2. NewOccupiedManipulators = DetermineOccupiedManipulators(SystemState)
      3. [SystemState AlignListFlag] = AlignRefList(SystemState, OccupiedManipulators, FreeManipulators, ReferencePointsToAlignList)
      4. **If** (AlignListFlag = true)
         1. GeometricResolutionExitFlag = true, GeometricResolutionSuccessFlag = true
      5. **Else**
      6. **End**
   6. **End if**
5. **End While**

**Algorithm**: Compute Geometric Resolution

**Input**: RepositionCableList, SystemState

**Output**: SystemState, GeometricResolutionComputationSuccessFlag

1. **For** (All RepositionCablesList)
   1. Form CurrentPosition and DesiredPosition vectors from list of interlinks violated on Cable
2. **End for**
3. **While** (exitFlag == false)
   1. **For** (All Reposition Cables)
      1. **If** (Interlinks on given cable are stretched)
         1. Add new violated Interlinks to CurrentPosition and DesiredPosition arrays
         2. ComputeJacobian(CurrentPosition, Interlink Locations on the cable)
         3. Propogate for (
         4. SystemState.RepositionManipulators(AssignedManipulator, CurrentPosition+)
      2. **End if**
      3. **If** (All Interlinks Resolved)
         1. exitFlag = true
      4. **End if**
   2. **End for**
4. **End** **While**

**Algorithm:** Determine Reference Points to Align

Input: SystemState, CorrespondingCables, InitialList, FreeManipulators

Output: FinalList

1. FinalAlignmentList = InitialList
2. **For** (All CorrespondingCables)
   1. Sort ReferencePoints by priority
   2. j = 0
   3. **While** (exitFlag == false)
      1. Align CorrespondingCable.AlignReferencePoint[j]
      2. Add the aligned Reference point to FinalAlignmentList
      3. Increment j
      4. **If** (All violated interlinks on currentCorrespondingCable resolved)
         1. exitFlag = true
      5. **end if** a
   4. **End While**
   5. **If** (New Interlinks violated)
      1. Create NewCorrespondingCables with currentCable as clamped cable
      2. Identify Relevant ReferencePoints for NewCorrespondingCables
      3. [Final Alignment List] = DetermineReferencePointsToAlign(SystemState, newCorrespondingCables, FinalAlignmentList)
   6. **End if**
3. **End For**

**Algorithm**: AlignReferecenPointList

**Input**: SystemState, ReferencePointList, FreeManipulators, OccupiedManipulators

**Output**: SystemState, SuccessFlag

1. RemainingReferencePointsList = ReferencePointsList
2. exitFlag = false; successFlag = false
3. AttemptsSinceLastSuccess = 0;
4. CurrentListSize = Size(RemainingReferencePointsList)
5. **While** exitFlag == false
   1. CurrentListSize = Size(RemainingReferencePointsList)
   2. **If** (isEmpty(RemainingReferencePointsList)
      1. exitFlag = true
      2. SuccessFlag = true
   3. **End if**
   4. currentReferencePoint = Pop(RemainingReferencePointList)
   5. SystemState.AlignReferencePoint(currentReferencePoint, AssignedManipulator)
   6. Determine ViolatedInterlinks
   7. **If** (isEmpty(ViolatedInterlinks)
      1. AttemptsSinceLastSuccess = 0;
   8. **Else**
      1. Determine FreeManipulators and OccupiedManipulators
      2. [SystemState ResolveSuccessFlag] = ResolveInterlinkConflict(SystemState, OccupiedManipulators, FreeManipulators, ClampingManipulator, ClampingCable)
      3. **If** (ResolveSuccessFlag == true)
         1. AttemptsSinceLastSuccess = 0
      4. **Else**
         1. Increment AttemptsSinceLastSuccess
         2. Backtrack SystemState to Before alignment
      5. **End if**
      6. **If** AttemptsSinceLastSuccess >= CurrentListSize
         1. exitFlag = true
         2. SuccessFlag = false
      7. **End If**
   9. **End If** a
6. **End While** a