

This UCP is a 3-phase sequence of activities "proposal", "counter-proposal" and "acceptance/confirmation", with possible iterations as shown. Each activity is represented by a "node" (represented here by a high level process description) and the nodes are connected by "arcs" (shown here as arrows) which
5 each indicate a transition condition for moving from node to node. The Coordination Software Module 210 for each agent must be capable of supporting that agent's role in the UCP.

2.3.1 Co-ordination Software Module 210

10

The co-ordination software module 210 is designed to interpret co-ordination graphs when given an initial (problem) state. The initial state specifies the initial conditions of the problems and the necessary data.

Execution of a graph by the coordination software module 210 proceeds
15 as follows: the engine selects the first node of the graph and instantiates a process identified by the label of the node. This process is run by calling its *exec()* which returns one of three possible results: FAIL, OK or WAIT. If *exec()* returns OK, a process identified by the label of the first arc leaving the node is instantiated and executed. If the arc test succeeds then the node pointed to by the arc is scheduled
20 for execution. The graph will be executed in this way until a final node is reached from which there are no more arcs.

The co-ordination software module 210 continuously cycles through graphs in the sense that it monitors for events and exceptions occurring at any time.

25 If an arc test fails at a node, the next arc from the node is tried. If a node's *exec()* method returns FAIL or all arcs leaving the node fail then the node is backtracked over (by calling the node's *backtrack()* method which undoes any changes made by the *exec()* method) to the previous node of the chain.

If a node's *exec()* method returns WAIT, then the node is placed on a wait
30 queue until one of two conditions become true: either a new external event is received by the engine (e.g. a reply message is received from another agent) or a timeout specified by the node is exceeded. In either case, the node is scheduled for execution again. In summary, the engine performs a depth-first execution of the graph with backtracking allowed.