Such an approach may be helpful in some circumstances with distributed software, but certainly not all. There are many classes of errors which occur in multi-agent systems. There may be structural errors at the level of singular agents within the multi-agent system, such as wrong or missing aquaintance relationships between agents, missing resources, incorrectly specified (typically short) times to run tasks etc. There may be functional errors, ie errors which relate to the logic of the tasks that the agents are performing. These can be compounded by the fact individual agents may be functionally 'correct', but the emergent behaviour of the overall set up of distributed control agents may not be what was expected. This is typically due to what might be called co-ordination errors. In some cases, such behaviour can lead to incoherent multi-agent systems.

Some examples of the sort of undesired behaviours which emerge from such systems include the following:

- *Deadlock:* where agents may be contending for shared resources. An agent may grab a vital shared resource and fail to relinquish it for some reason, perhaps because of some failure for example at the level of the individual agent. But this resource is invaluable to other agents who 'hang up' waiting for this resource to be relinquished. Basically, deadlock refers to a state of affairs in which further action between two or more agents is impossible.
- *Livelock:* where agents continuously act (e.g. interact or exchange messages), but no progress is made in solving the problem at hand. This is common in cases where the agents co-ordinate their activities via decentralised planning. Here, the agents can indefinitely exchange subplans without necessarily progressing the co-ordination effort, especially where there are no structural checks to detect and prevent indefinite loops in the planning process.
- *Chaos*: Chaotic behaviour is always potentially possible in a distributed setting.

Such behaviours, in addition to standard 'incorrect' behaviours or bugs on the part of individual agents, frequently lead to uncoordinated behaviours. Naturally, a distributed control system should normally be coordinated. A correctly coordinated set-up fully exploits the capabilities of individual agents and minimises conflicts, resource contentions and redundancy between them. Clearly then, co-ordination is a desirable property of agent systems.