Presently, most spacecraft are controlled from ground involving activities such as up-linking the schedule of daily operations and monitoring health parameters. These activities lead to a cognitive overload on human operators. Imaging/science opportunities are lost, if any discrepancies occur during the execution of pre-planned sequences. Consequently, advanced space exploration systems for future needs demand on-board intelligence and autonomy. This thesis attempts to solve the problem of providing an adequate degree of autonomy in future generation of spacecraft. The autonomous spacecraft accept high-level goals from users and make decisions on-board to generate detailed command schedules satisfying stringent constraints posed by the harsh environment of the space, visibility criteria and scarce on-board resources. They reconfigure themselves in case of any failure and re-plan when needed.

Autonomy concepts are derived in the context of complex systems by drawing analogy to living organisms and social organisations. A general autonomy framework may be defined with a six level structure comprising of the following capabilities - reflexes, awareness, self-regulation, self-healing, self-adaptation and self-evolution. A generic and reusable software architecture is proposed using hybrid multi-agent systems, which are arranged in a hierarchical manner using two types of decomposition viz. stratum and layer. The software architecture of the autonomous spacecraft is modeled as a stratified agent with a deliberative stratum, which achieves adaptive behaviour and a reactive stratum, which achieves reactive behaviour. Each individual agent has a generic structure comprising of perception, action, communication and knowledge components. It achieves the specialist capability through model-based reasoning. The knowledge models encompass: Planning knowledge describing higher-level goals, task structure and method of achieving the goals, Control knowledge encompassing the static and dynamic models of the spacecraft and Diagnostic knowledge incorporating the cause-effect relationships.

The deliberative stratum is capable of planning in different time horizons and is, in turn, organised into a hierarchical agent system with three layers corresponding to different time horizons. It is composed of a long-term, medium-term and short-term planning agents, focusing on strategic issues, spacecraft level resources and specific spacecraft states respectively.

The power of Petri nets is exploited for knowledge modeling as well as for plan representation. The ability of Petri nets to represent causality, concurrency and conflict relations explicitly makes it an excellent tool for representing the planning problem. Hierarchical Timed Petri Net is chosen for our modeling, since it captures the temporal requirements of the real-time spacecraft operations as well as facilitates the modeling of the system with multiple levels of abstraction. The necessary primitives for the plan representation are defined. In hierarchical modeling using Petri nets, refinement is done by a compound (high-level) transition. A compound transition models either a complex activity, which
corresponds to high-level operation on spacecraft or a method, which corresponds to the agent capability. At the lowest layer, a transition in the plan represents a primitive command to the spacecraft, such as ‘switch on camera’. The Petri net unfolding technique, which is a partial order approach, is applied to derive the plans from the dynamic knowledge models. This tackles the problem of combinatorial explosion. A hierarchical planning approach is followed, in which the abstract plan is recursively decomposed using the unfolding technique and refined by way of exercising the appropriate decisions in each layer. The reactive stratum is configured with three peer level agents. The control agent executes the command schedule and has the capability for reflex action. Structural properties of Petri nets are exploited by the execution-monitoring agent and the diagnostic agent for system level diagnosis. Fault tree method is applied for fine granularity diagnosis. The resultant architecture is a cost-effective solution, since it permits reusability of knowledge models across similar missions.

The knowledge models are formally verified for ensuring the absence of deadlocks, buffer overflows, recoverability and detection of unreachable modules using Petri net properties such as reachability, liveness, boundedness, safeness, reversibility and home state. The high-risk components are subjected to safety property verification, which makes the system rugged. The hierarchical composition of Petri net models (which are independently verified), preserves liveness and boundedness characteristics and thus ensuring the reliability of the integrated models. This, in turn, ensures that reliable plans are generated on-board using these good quality models. The models of the system components viz. partial order plan, conditional plan, dynamic world model, reflex model, resource model and the hierarchical models are developed and demonstrated using HPSIM and Moses Tool Suite, using examples from spacecraft domain. The long-term planning agent, with hierarchical world models, for handling high-level goals is developed and simulated using Moses Tool Suite. The plan generation using unfolding approach is demonstrated using VIPTool, which has the partial order analysis capability.

In summary, the main contributions include (a) Definition of a general framework for spacecraft autonomy; (b) design of a generic and reusable architecture for autonomous spacecraft using hybrid multi-agent concepts; (c) unified knowledge representation and reasoning using Petri nets across various strata/layers; (d) application of Petri net unfolding technique in a hierarchical manner for plan generation; (e) use of structural properties of Petri nets for fault identification and location; (f) verification and validation of Petri net models using Petri net properties and (g) simulation and demonstration of the system components viz. partial order plan, conditional plan, dynamic world model, reflex model, resource model and hierarchical models, by developing examples from spacecraft domain, using HPSIM and Moses Tool Suite and demonstration of plan generation using unfolding technique using VIPTool.