# Literature Review

This section builds, and expands, on material previously included in the project Initial Report (see Appendix B)

## Overview of Modular Spacecraft

Modular spacecraft are a design concept where the overall space system is composed of interchangeable modules / components, where each module is designed to serve a specific function such as propulsion, communication, power generation, or sensing. These modules are standardized allowing them to be easily connected to form a singular system, where modules can be moved or replaced to improve craft efficiency during operation and extend the overall lifetime of the system. Taking a modular approach to design offers several advantages over traditional designs such as flexibility, adaptability, and ease of maintenance.

Modules are equipped with standardised interfaces which define how the modules physically and electronically interact, enabling modules with different purposes or made from different manufacturers to seamlessly integrate into the overall system architecture. The size and shape of modules can vary in different modular designs however standardisation principles allow these components to be integrated with other modules regardless. The size of the final space system architecture is primarily limited by the type and number of modules it is comprised of, providing scalability which enhances the spacecrafts versatility and cost-effectiveness; As the system can therefore be tailored to meet specific needs of different missions without requiring a complete redesign.

## State-of-the-art in Spacecraft Modularity and Automated Reconfiguration

The following section explores existing cases of spacecraft modularity and reconfiguration technologies currently or previously in operation. Due to the challenges related to developing automated reconfiguration systems for operation in space, there are no relevant existing cases of automated reconfiguration other than the International Space Station (ISS), however the use of modular design principles has been present in the space industry for development of craft from as early as the 1980s with the development of the MMS.

### Multi-mission Modular Spacecraft (MMS)

The Multi-mission Modular Spacecraft (MMS) was designed and deployed by NASA in the 1980s and 1990s [%] with the intention of decreasing space mission costs. Intended to be recoverable/serviceable by the Space Shuttle Orbiter [%], It is one of the first cases of modular designs seen in the space industry and has paved the way for future innovations.

The MMS consisted of a small number of immobile modules, with the most basic deployed MMS containing only modules for altitude control, communications and data handling, and the power subsystems module [%].

The MMS flew only six missions through its lifetime which was vastly different from the thirty-one expected in the 1970s [%], it suffered limitation in the form of electronic technologies rather than mechanical restraints. NASA’s first Standard Spacecraft Computer (NSSC-1) [%] was developed to prevent requiring an entire redesign of onboard computers for each mission, requiring only a software redesign though this was still a heavy burden affecting the MMS’s mission flexibility. While no longer in operation as of 2006 [%], the system did show cost-savings in the range of 55% to 65% [%]. “The idea of a modular system serving many purposes was the pioneer of the leading systems within the space technology ecosystem today as it has left a lasting legacy” [%]. In the wake of the MMS’s legacy, new design techniques were developed such as the Modular, Adaptive, Reconfigurable Systems (MARS) system-level architecture [%] that has built the foundation for modern space systems.

### Modular Common Spacecraft Bus (MCSB)

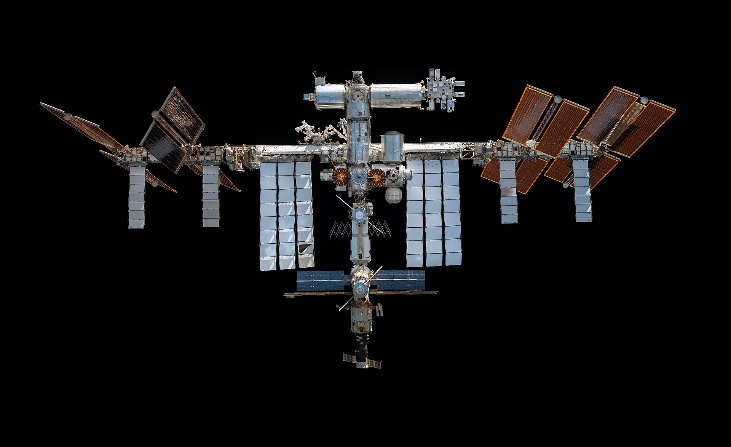
A diagram of a space module

Description automatically generatedThe MCSB is a fast-development, low-cost, general purpose spacecraft platform consisting of a series of 4-5 modules stacked on top of each other, each serving separate functionality [%]. According to NASA, “the spacecraft is roughly one tenth the price of a conventional unmanned mission and could be used to land on the Moon, orbit Earth, or rendezvous with near-Earth objects.” [%]

The MCSB system received the Popular Mechanics 2014 breakthrough Award for innovation in science and technology [%] and is proving to be at the forefront of existing modular space technologies, first deployed on the Lunar Atmosphere and Dust Environment Explorer (LADEE) mission in 2013 [%].

The MCSB system is an example of modularity being used to streamline and reduce costs of the initial development process of the craft, being able to carry up to 50kg of scientific equipment inside its payload module [%], though the end product is still a whole system that has limited in-operation service capabilities and is not capable of being reconfigured to adapt to mission requirements in-orbit.

### International Space Station (ISS)

The International Space Station (ISS) is the largest space platform ever built, created with the purpose of performing microgravity and space environment experiments. First launched in 1998 and expanded through the integration of additional modules and serviced by human occupants up until its planned de-orbit in 2031, it is a monument to advancements in the space industry.

The ISS is capable of reconfiguration using a robotic arm and automated docking with human oversight [%] unlike previous cases, though unsupervised automated reconfiguration is yet to be attempted due to the consequences of failure.

While the above is not necessarily a complete list of all cases of modularity in the space industry, it does cover the major cases presented over the history of space exploration. As of present, automated reconfiguration of spacecraft is yet to be introduced to the industry. This project therefore will endeavour to contribute towards the potential wide-spread introduction of automated modular reconfiguration in the future; By providing a system that can be compared to other systems under development to identify the implemented techniques which provide the greatest benefit. These techniques can then be utilised in tandem to create increasingly advanced reconfiguration systems.

## Challenges and Limitations of Automated Reconfiguration in Space

The lack of deployment of complex automated systems such as automated reconfiguration systems in space is not due to a lack of interest, but instead due to the difficulty of the technical challenges presented by such systems and the risk introduced to extremely expensive, and often critical missions, that cannot afford failure.

Space systems must be reliable and work in a wide range of conditions. As a systems complexity increases, so does the number of potential points of failure. Which makes the validation, verification, and deployment of complex systems in the space industry a lengthy and expensive process. Challenges that autonomous space systems face include:

* **Communication latency** – delays in communications from systems make it impossible for human controllers to react to unexpected situations in real-time, meaning any autonomous system must be capable of performing completely without human intervention. Simply having an autonomous system that is allowed to operate under human observation such as a self-driving car does not meet the reliability requirements for space applications.
* **Safety Requirements** – Systems will often be hosting valuable scientific equipment while operating in harsh, unpredictable environments where various hazards are present such as extreme temperature differences, radiation, space debris, ice and lack of gravity.
* **Limited Power Sources** – autonomous systems require power which depending on a crafts power source is not always guaranteed. For example, a craft relying on solar power may lose power during eclipses or due to unexpected collisions from space debris. Autonomous systems must be capable of recovering from temporary power loses or have reliable backup power sources to prevent mission failure.
* **Isolation** – unlike on land, it is usually not possible for a craft to quickly receive help or be viewed by an external observer. An autonomous system must have the sensing capabilities to self-diagnose problems or detect anomalies and halt standard operation, otherwise the system could cause further damage to itself.

Overcoming these challenges requires a level of technology that has only become available in recent years leading to the undertaking of research projects such as this one. It can be expected to see the number of autonomous systems present in the space industry drastically increase over the next few decades as computational power and materials sciences continue to advance.

## Emerging Advancements in Reconfiguration Technologies

### MOSAR Outcomes

Up till now the MOSAR project has produced several major outcomes:

* A standardised module framework making use of the HOTDOCK adapter.
* Design and fabrication of a walking manipulator arm.
* Related system architecture to control the arm remotely.
* Successful ground demonstration of the manipulator arms capabilities to move and connect modules.

At this stage, an in-orbit implementation of the MOSAR demonstrator would be theoretically capable of reconfiguration functionality, though requires reconfiguration instructions to be manually sent to the craft. Further work is required for automated functionality such as:

* Automatically find a desired module configuration for the craft to meet mission requirements.
* Automatically compute a set of manipulator instructions required to reconfigure the craft from one configuration to another.

The following review of literature will be focused on identifying the best methods to perform the latter.

### Automated Reconfiguration

Automatic planners, algorithms that find a solution for which sequence of operations must be accomplished to achieve a specified goal, have been an area of development attracting wide-spread interest since the earliest days of robotics. Currently there are many different types of automatic planning techniques available. They encompass a large set of algorithmic requirements which trend towards purely discrete or purely continuous search space characteristics. The development of “Hybrid” automated planning approaches with search space characteristics that are not purely discrete or continuous, especially Task and Motion Planning (TAMP) algorithms, represent an area of study of which solutions are considered the most computationally difficult in theory [%]. Consequently, the application of automated planning algorithms to robotic assembly of modular satellites is a very recent development in which little work has been published that implements automatic reconfiguration algorithms while fully considering the range of real-world physical restraints and limitations presented by usage of a mobile manipulator arm in a low-gravity environment.

A diagram of a hybrid planning

Description automatically generated

Taxonomy of automated planning approaches based on their search spaces’ characteristics. Image from [%1].

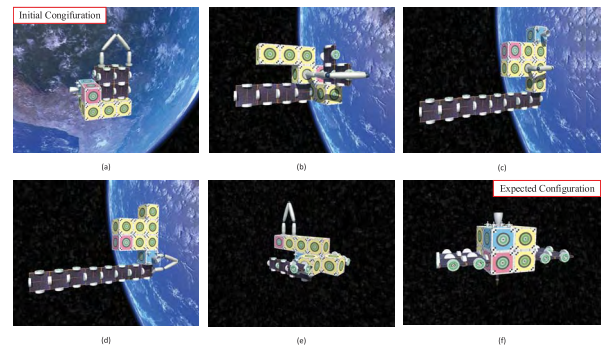
Motion Planning is finding solutions to move a robot “from one configuration to another configuration without colliding with the objects in the world” [%1]. It involves searching for paths within the robots reach which is a continuous configuration space limited by dimensions represented by the joints of the robot. These collision-free paths are important for robot motion but do not by themselves allow the robot to interact with the world. Further planning must be implemented to allow manipulation of objects through manipulation planning (known as Multi-Modal Motion Planning).

Due to the increased complexity of the problem presented by manipulation planning, the problem is best broken down into a hybrid discrete-continuous search problem of “selecting a finite sequence of discrete action types (e.g. which objects to pick and place), continuous action parameters (such as object poses to place and grasps), and continuous motion paths” [%1].

While Motion and Manipulation planning are seen as problems mainly within the robotics field, planning within large discrete domains such as in problems presented by task planning has been more deeply researched within the artificial intelligence (AI) community [%]. Task planning (also known as Action planning) referring to deducing a composition of symbolic actions to achieve a high-level goal (e.g. computing a sequence of actions required to stack boxes in a specified order). The discrete nature of the problem makes it particularly suitable for many machine learning techniques which have particularly advanced in recent years.

Current research in task and motion planning (TAMP) primarily aims to combine the robotics solutions for manipulation planning under physical constraints with the usually unrestricted AI approach to task planning. With the goal of deriving automated planning systems capable of reasoning symbolically with discrete “high-level” robotic action sets while geometrically taking into account continuous “low-level” robotic motion planning and restrictions. To date, several papers have developed algorithms for similar TAMP problems to the scenario of modular satellite reconfiguration that unfortunately are not compatible due to the method of module mobility, but act as a proof of concept that a solution is possible [%][%][%].

The 2010 Intelligent Building Blocks for On-Orbit Satellite Servicing and Assembly (IBOSS) project [%] by DLR provided many advances in the area of satellite modularization with the development of standardised building blocks and interfaces [%]. Simple task planning techniques were implemented using Hierarchical task network (HTN) planning to produce high-level mobile arm instruction sets to then be verified through inverse kinematic checks and motion planning. This implementation solved the discrete and continuous planning problems separately, which simplified the problem however does not allow the separate systems to properly integrate. The system was not capable of efficiently solving more difficult tasks of identifying were solutions where not feasible.



Alternatively, another approach was taken here [%] through the implementation of the melt-grow algorithm [%]. The physical restraints of the robot were not including in the reconfiguration planning stage of the system, effectively reducing the problem to task planning. This reduces complexity though can only be achieved due to the behaviour of the melt-grow algorithm, which deconstructs (melts) the initial module configuration into chains of modules defined as the intermediate configuration, seen in configuration d in figure [%], before then reconstructing (growing) the modules into the expected configuration. The system then does not need to consider whether a move is possible for the mobile arm through manipulation planning as due to the algorithms inclusion of an intermediate state between the melting and growing operations, the algorithm essentially reconstructs the satellite instead of modifying the current state, all required moves are possible for the mobile manipulator and simply require motion planning. While proven to work, this method is shown to be highly inefficient for the mobile manipulator, especially as the number of modules increases in the system. Though, the paper [%] suggests this could be offset by the inclusion of additional manipulators which would consequently increase construction and operational costs.

More recent research has taken inspiration from these previous works to propose a comprehensive Task and Motion Planning (TAMP) problem solver [%] to intrinsically include the robot constraints into the system. The system, seen in figure [%], includes a logic layer, a physical layer, and a feedback system. Where the logic layer acts as a task planner finding a semantic solution by considering the solution as a sequence of states, with module movements defining the transition between states. A graph is developed to represent the possible states where nodes are system states and edges represent module movements which are verified by the physical layer which provides manipulation planning results through the feedback system. Using this graph, the shortest and hence most efficient set of operations to reconfigure the system into the desired state can be identified. The removal of the intermediate configuration present in the melt-grow algorithm improves the efficiency of the solution set of operations, especially as the number of modules in the system increases, requiring less movement from the mobile manipulator.

Diagram of a machine

Description automatically generated

“Architecture of the autonomous robot planning system. The system receives as inputs the start and goal satellite configurations, and iterates between the logic and physical layer until a solution is found.” [%]

The paper notes “the goal of this work was not to set a baseline for planning problems in terms of absolute times, but to demonstrate the usefulness of integrating feedback from the physical layer on the logic layer.” [%], suggesting that there is an opportunity for further research into the components of the planning system and the related feedback strategies to further advance the system towards space applications.

## Gaps and Opportunities

Modular reconfiguration defines a subclass of the generic planning problems usually addressed by TAMP. Although research has previously demonstrated effective systems that can handle both symbolic and geometric reasoning, their application to robotic assembly and in particular robotic re-assembly is currently limited. There is additionally a distinct lack of discriminating modular blocks by type in existing algorithms which could potentially be easily implemented without a substantial hit to system performance.

The system proposed in figure [%] is promising due to the robustness of solutions and flexibility of the logic layer, however, there lacks the extensive performance testing required to recognise weaknesses and future improvements and identify why this system could not be used in real-world application currently.

Need to add:

* Identifying key technologies, I will use