RECONFIGURABLE PRODUCT ROUTING AND CONTROL FOR MASS CUSTOMISATION MANUFACTURING

A.J. Walker[1*] and G. Bright[2]

[1,2] Department of Mechanical Engineering
University of KwaZulu-Natal, South Africa
walkeraj@ukzn.ac.za, brightg@ukzn.ac.za

ABSTRACT

Reconfigurable control of flexible manufacturing systems can allow for the efficient and responsive production of customised product instances. This can aid in providing make-to-order business models for various small- to medium-sized enterprises in South Africa, and provide competitive advantage in a dynamic global marketplace. Reconfigurable control application requires an understanding of the modes of production variability in mass customisation manufacturing. Temporally uncorrelated workflow routings are considered as one of these production variability modes. In this light, this paper addresses the flexible material payload routing problem, and presents a mobile robot platform that has been developed to research and design reconfigurable routing systems.

OPSOMMING

Verstelbare beheer van aanpasbare vervaardigingstelsels kan die doeltreffende en reaktiewe produksie van doelgemaakte produkeksemplare toelaat. Hierdie beheer kan assisteer in die voorsiening van vervaardig-na-bestellingbesigheidsmodelle aan 'n aantal klein- tot medium-grootte ondernemings in Suid-Afrika, en aan hulle 'n kompeterende voorsprong bied in 'n dinamiese globale mark. Die instelling van verstelbare beheer vereis dat die modi van produkveranderlikheid in massa verbruikersaanpassingsvervaardiging verstaan word. Hierdie artikel spreek die probleem van aanpasbare roetebepaling van materiaalvragte in massa verbruikersaanpassingsvervaardiging aan. 'n Mobiele robot-platform, ontwikkel vir navorsing in verstelbare roetebepalingstelsels, word ook voorgelê.

---

[1] The author is a candidate for the PhD degree in Engineering in the Mechanical Engineering Department at the University of KwaZulu-Natal.
*Corresponding author.

1. INTRODUCTION

In order to facilitate production variability in mass customisation manufacturing (MCM), manufacturing control and execution systems must be able to adapt to ever-changing production plans, each of which is associated with a custom product instance. The adaptation in manufacturing control can be enabled through reconfiguring flexible production infrastructures so as to create required workflow through the production plant.

From a control perspective, fundamental decomposition of product variability in MCM provides two separate production variations. The first constitutes processing variations, in which reconfiguration of process control parameters and assembly procedures allows for slight modification in the manufacture and assembly of a part or parts. Considering that control of elementary processes performed by machine tool infrastructure remains somewhat constant through time [1], this variability generator is of limited interest. This is mainly due to the difficulty in implementing truly reconfigurable machine tools [2]. The second constitutes the 'on the fly' generation of unique product routes. This domain is less physically constrained, and requires only real-time point-to-point material payload transfer. By considering this property, the dynamic reconfiguration required by high product variability in MCM is mostly limited by the generation of flexible product routings [3].

The ability to implement sustained scheduling of unique routing plans would greatly benefit the realisation of batch-of-one production operations, and therefore MCM. This paper identifies and addresses the flexible routing control and execution problem in establishing flexible workflow for MCM. Section 2 characterises and defines the material payload routing environment of MCM. Section 3 presents a payload routing implementation architecture (IA) for flexible routing control and execution. Section 4 presents a physical instance of the IA in the form of a differential drive mobile robot, equipped with the materials handling infrastructure to carry out routing operations. Section 5 presents motion control and task execution results from a variety of motion control tasks carried out using the mobile platform. A discussion regarding this research is presented in Section 6, and Section 7 offers a conclusion and directions for future research.

2. UNDERSTANDING PRODUCT ROUTING IN MCM PRODUCTION STRUCTURES

In order to implement production structures capable of MCM, all product attributes that affect or influence the magnitude or frequency of customer-induced production rate variations must be collected and concurrently analysed. Concurrent analysis, and concurrent engineering for that matter, can provide insight into methods of quantifying and achieving production stability. Production stability is very important in implementing MCM, and even more so when avoiding manufacturing dead-locks during the reconfiguration of flexible production infrastructures.

There will always be a miscommunication of concepts across engineering disciplines. For this reason, the concepts of MCM production structures have been encapsulated in a control theoretic construct. A two degree-of-freedom Single Input Single Output (SISO) control loop model of MCM production operations highlights the required active control efforts (Figure 1).

Figure 1, moving from left to right in the direction of 'signal flow', highlights critical control points in MCM production. This characterises the materials handling and routing environment required for efficient and effective production of customised product instances. First, the pre-filter represents one of the main contributors to establishing effective and efficient MCM, i.e. Design for Mass Customisation (DFMC). As mentioned previously, the control loop has two degrees-of-freedom, one in the forward loop and another in the feedback loop. DFMC is of utmost importance in MCM as it ultimately affects the required active control efforts and production dynamics. Enabling modularity in custom product design creates better utilisation of manufacturing resources [4], and allows for well-defined product variability mappings.
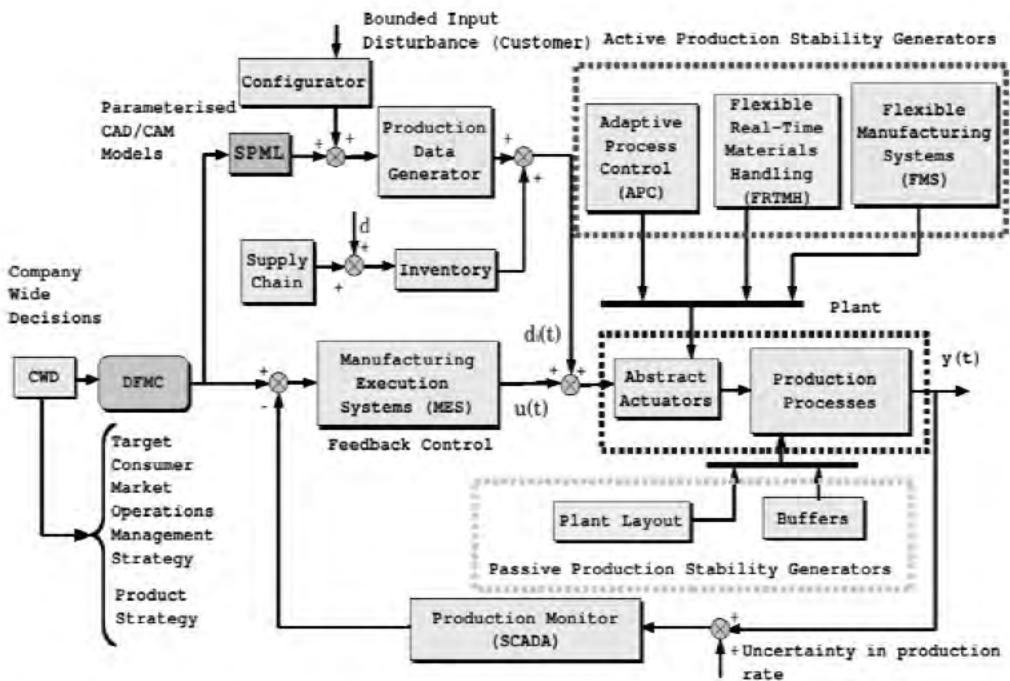
Figure 1: Control theoretic description of MCM

The SPML block represents a Standard Product Model Library, and represents customisable products in their standard, or basic, configuration.

Signals flowing out of the DFMC block are in the form of parameterised CAD/CAM models and associated production plans. These production plans may not be absolutely and/or fully specified, as control reconfiguration and production execution may rely on active observation of current resource status [5]. Customers are seen by the production plant as bounded input disturbances. This assumption is reasonable, as the degree of customisation would always be bounded by the processing capability of the production plant. The objective of MCM production is to remain at nominal and economic production rates, under stochastic yet bounded input disturbances. The Manufacturing Execution System (feedback controller) in Figure 1, and therefore MCM control and execution, is designed to act as a disturbance regulator.

With absolute and certain understanding of the capabilities and dynamics of the production plant, i.e. the 'actuators' and production processes, stability information would be available and/or fully determined by the design of the pre-filter (DFMC). This is, however, seldom the case, as the complexity of modern manufacturing systems makes fully specified production operations intractable due to the state explosion problem [6]. Disturbances must be regulated via active process control and distributed integration, in order to maintain nominal production rates. Distributed process integration is enabled through material payload routing and materials handling operations. This accounts for a large amount of production variability absorption in the system, and is the focal point of this paper.

Distributed process integration can be considered to occur under utilisation of both passive and active measures. The passive measure is the Facility Layout Problem (FLP) in which known workflow volumes are mapped on to optimal or near optimal plant layouts [7]. The majority of metrics used in the optimisation take into account both the Euclidean distance between consecutive processing cells, and a relative weighting of payload flow volumes between processing cells. Multiple solutions to the FLP have been developed over many

years of production research [8], although all solutions assume steady or static product mixes and deterministic production plans. This makes these solutions less applicable to the implementation of MCM. Active measures are required to maintain production rate stability under customer-induced variations in standard process integration operations. This active measure relies on commissioning mobile payload routing platforms with the capability of executing a payload transfer operation between any two distributed processing cells in the production plant. This can be regarded as an enabling technology for MCM implementation.

Flexible material payload routing operations are those that fall outside and supersede the capabilities of standard fixed conveyor and gantry systems, in terms of spatial distribution. Of all material payload transfer infrastructure, conveyor systems have the largest capacity, as they are able to accommodate multiple payload instances along their path distributions. However, they lack the flexibility of mobile AGV systems in executing non-standard payload routings operations. For ease of reference, a flexible material payload routing operation will be defined below. The definition accounts for all aspects of task execution, including task allocation, handling, and status reporting. A Flexible Material Routing Primitive (FMRP) is an active instance of distributed process integration, and represents a subset of the 'abstract actuator' (Figure 2).
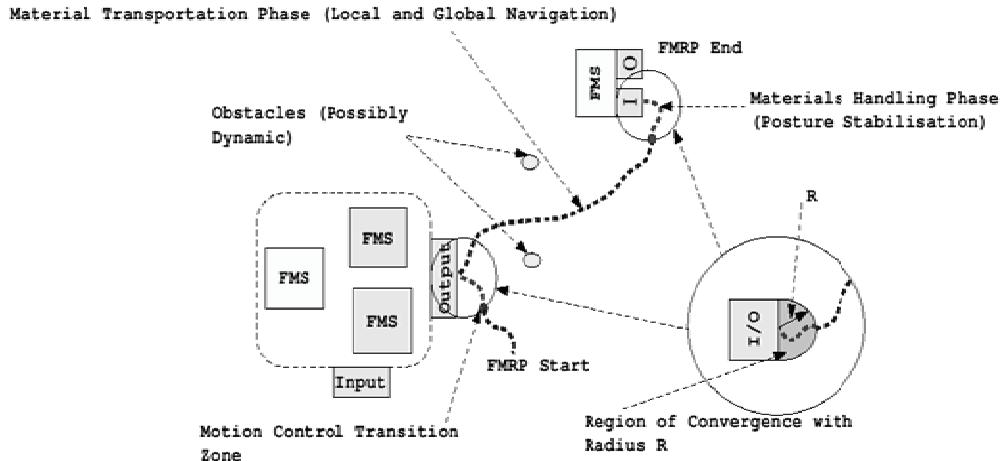
Figure 2: Disjoint Motion Control Regions of a FMRP
The Region of Convergence has been incorporated into the definition to account for mutual exclusivity in accessing production cell input/output (I/O) ports.

All physical point-to-point routing operations can be decomposed into three phases: a material loading phase, a transportation phase, and a material off-loading phase. The material loading and off-loading phases can be considered together as a single materials handling phase. The transportation phase is somewhat different in contributing to the routing task, in that absolute positioning of the payload is of limited interest. For this reason, a FMRP is composed of two disjoint motion control regions. Material payload transportation is handled through the motion outputs of local and global navigation functions. The objective of these motions is to provide required payload transportation between distributed manufacturing infrastructure subsets, while avoiding possibly dynamic obstacles.

Materials handling is handled via posture stabilisation on to a known goal configuration using full state feedback control. Posture stabilisation of differential drive mobile robots requires non-linear control laws, due to the differential constraints imposed by the 'rolling without slipping' condition exhibited by the drive wheels. This control implementation is covered in Section 4.

The Region of Convergence (RoC) has been conceptualized and included in the definition to serve two functions. First, a RoC acts as a restricted region, in which only one mobile payload routing platform may occupy a particular time instant. This ensures that higher level manufacturing management services can uphold mutually exclusive access to IO port infrastructures at processing cells. Second, the border of a region of convergence represents a transition zone for the type of motion control implementation. The full definition of a FMRP follows (Figure 3).
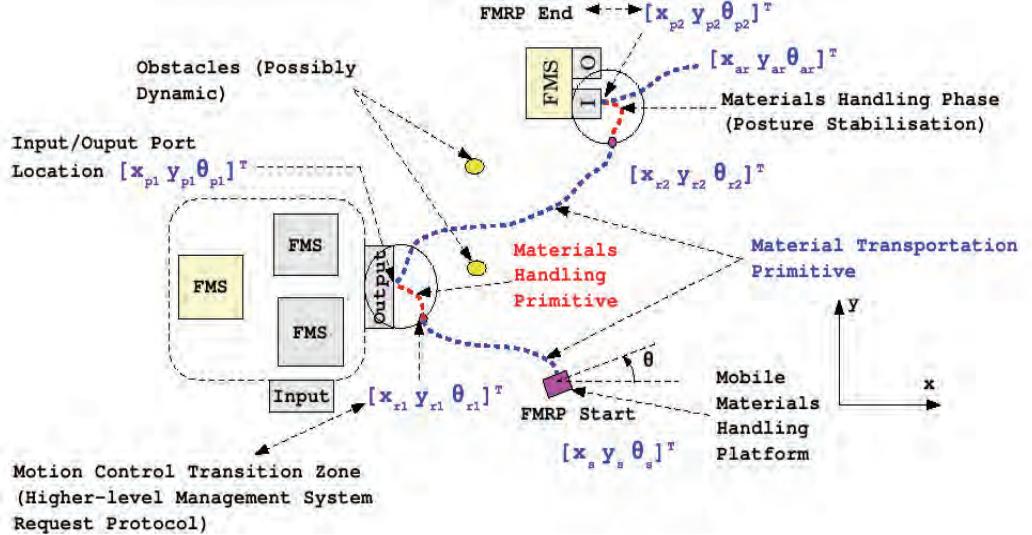
Figure 3: A Flexible Material Routing Primitive
Motion control when exiting a RoC is in the form of local and global navigation, as shown.

The definition of a FMRP is based on the encapsulation of two payload handling operations. In the definition, a payload routing platform's pose is considered as the combination of its global Cartesian position $[x, y]^T$ and its local orientation θ. A Material Payload Handling Operation (MPHO) consists of the following (see Figure 3):

1. Posture stabilisation from a pose on the boundary of a RoC, $[x_r, y_r, \theta_r]^T$, on to a goal pose $[x_p, y_p, \theta_p]^T$, in a predetermined vicinity of an input/output port.
2. An absolute alignment of the materials handling infrastructure with the input/output port, using degrees-of-freedom above those of the underlying transportation device.
3. A material payload transfer operation, either loading or off-loading.

A Material Payload Transportation Operation (MPTO) consists of a global navigation operation from a location $[x_1, y_1]^T$ to a second location $[x_2, y_2]^T$, while avoiding obstacles in real time.

A FMRP consists of the following eight operations:
1. A materials handling and routing task assignment from higher-level manufacturing management services to a mobile materials handling and routing platform.
2. A MPTO (NULL)[3], with $[x_1, y_1]^T = [x_s, y_s]^T$, and $[x_2, y_2]^T = [x_{r1}, y_{r1}]^T$, a boundary point of a RoC.
3. A request to a higher-level manufacturing management service for access to an input/output port, followed by an outcome acknowledgement.
4. A MPHO (Loading).

---

[3] NULL specifies that no material payload is present.

5. A MPTO to the destination RoC, with $[x_1, y_1]^T = [x_{p1}, y_{p1}]^T$, and $[x_2, y_2]^T = [x_{r2}, y_{r2}]^T$.

6. A request to a higher-level manufacturing management service for access to an input/output port, followed by an outcome acknowledgement.

7. A MPHO (Off-loading).

8. A MPTO, (NULL), with $[x_1, y_1]^T = [x_{p2}, y_{p2}]^T$ and $[x_2, y_2]^T = [x_{ar}, y_{ar}]^T$.

Creating a full task specification via a FMRP definition allows for structured and axiomatic design principles to be applied in the design of flexible process integration systems for MCM. An Implementation Architecture (IA) has been developed to align with fundamental requirements of FMRP task execution in MCM production environments (Section 3).

3. A FMRP IMPLEMENTATION ARCHITECTURE

A physical materials handling and routing robot platform requires certain basic core capabilities in order to execute a FMRP task instance. In a functional sense, hardware is required to facilitate materials handling as well as provide low-level motion primitives in order to transport a material payload between manufacturing infrastructure subsets. Active sensory infrastructures are required to provide the mobile robot platform with environmental perception and information about its local working environment. This allows the platform to sense both static and dynamic obstacles in order to perform local navigation and obstacle avoidance during material payload transportation. Due to the possibly heterogeneous[4] materials handling and routing platforms that execute FMRPs in a MCM production plant, a software system is required to provide a Hardware Abstraction Layer (HAL) in order to provide scalability in control and management structures. Implementing such a software system allows generic communication standards with homogeneous semantics to be developed. A communication sub-system is required to allow for the passing of messages, such as FMRP task allocations, from higher-level manufacturing management services to the materials handling and routing robot platforms. At the highest level of abstraction, a materials handling and routing agent architecture is required to provide problem-solving and machine learning capabilities, to maintain optimal production rates.

Over many years of system development, engineering architectures have been developed to encapsulate concepts and specifications required to implement internationally recognized and scalable systems. In this section, an IA is developed that encapsulates the core capabilities for FMRP task execution, referred to as the Autonomous Material Transportation Specification (AMTS) (Figure 4).

The IA consists of four main layers, and provides specification from hardware levels through to task allocation and management. The Hardware Implementation Layer (HIL) is a specification on the physical configuration of the mobile payload routing platform. The HIL consists of three sub-systems, each of which enables a core capability required to implement a FMRP. The Mobility Hardware sub-system is a specification on the hardware configuration that provides required motions to transport a material payload between distributed manufacturing infrastructure subsets. The Mobility Hardware specification is based on providing the material payload with at least planar motions, i.e.

$$OS : \vec{q} = [x, y, \theta]^T \in R^2 \times SO^1$$

where OS represents the output specification, and states that the mobile payload routing platform must be able to achieve configurations on planar two-space and the special orthogonal. The Special Orthogonal SO^1 is the mathematical manifold that represents all planar rotations of a rigid body between $[0, 2\pi)$ radians.

---

[4] In terms of hardware implementation.
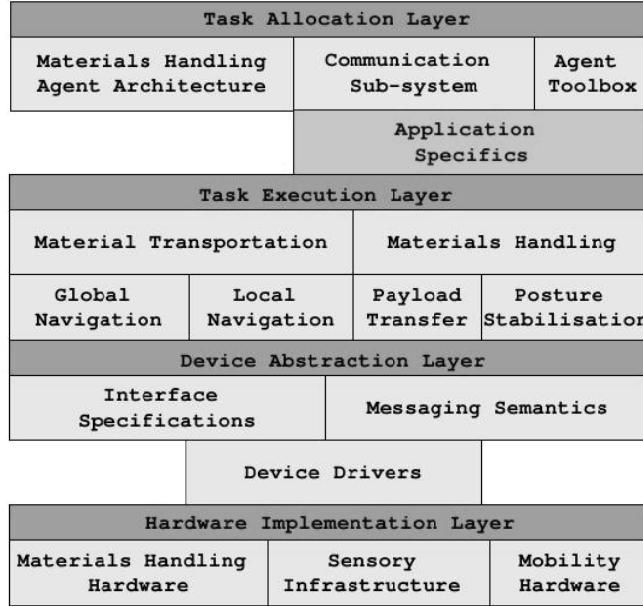
Figure 4: The Autonomous Material Transportation Specification.
The Application Specifics block is associated with the production requirements of each plant, such as facility layout configuration.

The Sensory Infrastructure sub-system specifies active perception capabilities and sensor configuration. In the highly dynamic environment of MCM production operations, the geometrical sensor configuration is very important, as perception must be of equal potency in all directions in order to provide required responsiveness in local navigation operations. The Sensory Infrastructure sub-system specifies that the mobile material payload routing platform must be able at least to perceive its environment, in terms of spatial identification, on a planar disk on its periphery. This ensures that the mobile robot platform does not exhibit performance measures that are directionally biased.

The materials handling hardware sub-system is specified in terms of the number of degrees-of-freedom allocated to the material payload above those provided by the transportation hardware. The output specification allows for the material payload to be aligned absolutely with IO port infrastructures, via relative tracking control between the orientation of the IO port and materials handling hardware, using absolute sensor feedback. The output specification states that the material payload must have at least one extra degree-of-freedom after both the transportation and materials handling hardware are considered together. This allows the transportation and materials handling hardware to provide independent payload mobility.

The Device Abstraction Layer has been included to provide a HAL, such that control applications can operate on generic structures using homogeneous messaging semantics. This HAL is beneficial in both scalability and code re-use. The Task Execution Layer (TEL) is an encapsulation of a particular control architecture used in high accuracy orientation tracking devices, such as satellites. Referring back to the definition of a FMRP in Section 2, it may seem arbitrary to create two separate and disjoint motion control environments. However, these notions of a RoC and associated motion control primitives are analogous to the control architectures implemented on satellite attitude control systems. In such systems, the control infrastructure is distributed and performs a different control function based on its accuracy, repeatability, and power consumption. For example, in the attitude adjustment of a satellite's communications equipment with a receiver on earth, relatively powerful thrusters are used for large attitude adjustments, and smaller more accurate

magnetic torque generators for final alignment of the communications equipment. In much the same manner, the TEL encapsulates this control architecture, where the transportation phase provides material payloads with large initial displacements. These initial displacements are only specified for maintaining real-time obstacle avoidance, independent of absolute accuracy or repeatability, as long as such motions terminate on the order of a RoC. On the other hand, the materials handling phase is used to achieve final accurate alignment of the material payload with an IO port, to ensure robust and reliable payload transfer.

The Task Allocation Layer (TAL) is still under development, although research plans to incorporate agent communication protocols, as well as data representation, status reporting, and error recovery schemes into the TAL have been completed.

4. A PHYSICAL INSTANCE OF THE IMPLEMENTATION ARCHITECTURE

A mobile payload routing robot platform has been developed, based on a subset of the IA, for researching flexible process integration for MCM. The platform is fundamentally a differential drive mobile robot, and runs on an embedded PC consisting of a Mini-ITX form factor single board computer running Linux. Communication with the mobile robot occurs over a WIFI link using TCP/IP (Figure 5).
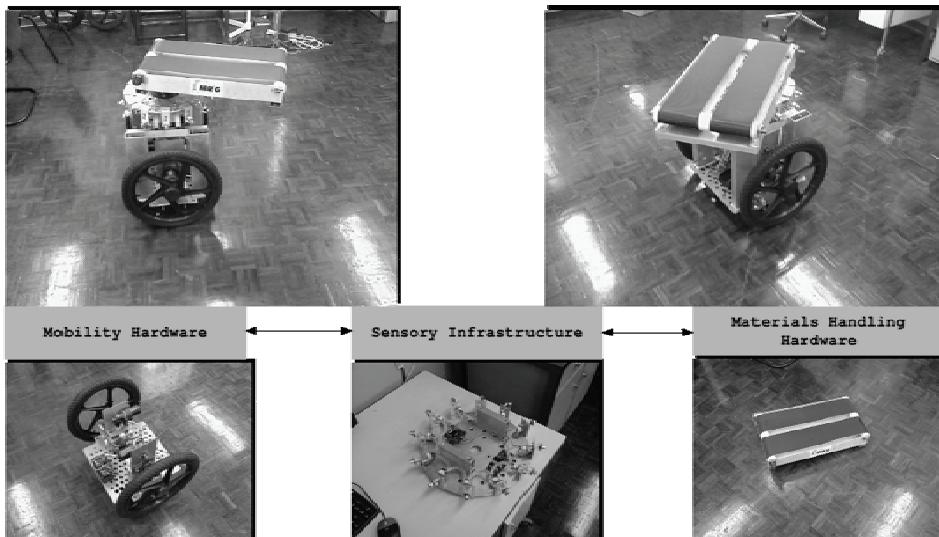
Figure 5: Mobile payload routing platform

4.1 Hardware implementation layer

4.1.1 Mobility hardware

The mobility hardware sub-system is in the form of a differential drive platform with embedded control and odometric capabilities. Embedded control is provided by a PIC18C252-based two-channel motor controller, which implements a PID loop around each drive wheel via quadrature encoder feedback and H-bridge motor drivers (Figure 6). Layers in the implementation architecture without shading have, to date, not been implemented and are under development.
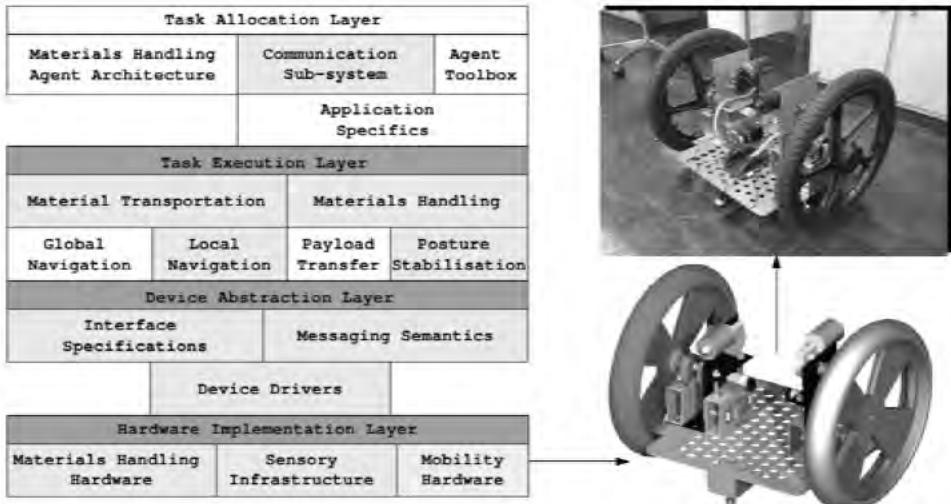
Figure 6: Mobility hardware implementation.
Each drive wheel is powered by a 12V DC motor. Power is transferred to the drive shaft via a chain and sprocket.

The PID control loops run on the embedded motor controller, which interfaces with host computing systems over a serial connection (RS232). Velocity control inputs are then set over a serial link to the embedded motor controller. Odometry is performed using runge-kutta second order numerical integration of the mobile platforms configuration velocity,

$$\dot{q} = [\dot{x}, \dot{y}, \dot{\theta}]^T = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}$$

$$q_{t+\Delta t} = q_t + \int_{t}^{t+\Delta t} \dot{q}\,dt$$

where v and ω are the translational and rotational velocities, respectively, of the mobile robot platform. Odometry is performed using algorithms embedded in the motor controller, which performs a discrete version of the integral shown above.

4.1.2 Sensory infrastructure

A 360-degree sonar array of 12 ultrasonic transducers is used for the mobile robot platform's active sensory infrastructure. This satisfies the specifications on the sensory infrastructure sub-system, and provides the robot with unbiased perception of its local working environment within a 4-meter radius (Figure 7).

The sonar array is managed by an embedded microcontroller, and interfaces with the host computer (the Mini-ITX PC running Linux) over a serial connection. The sonar array has been implemented with a particular firing sequence that tries to minimise sensory cross-talk [9]. This is achieved by consecutive firing of geometrically opposite sensors.
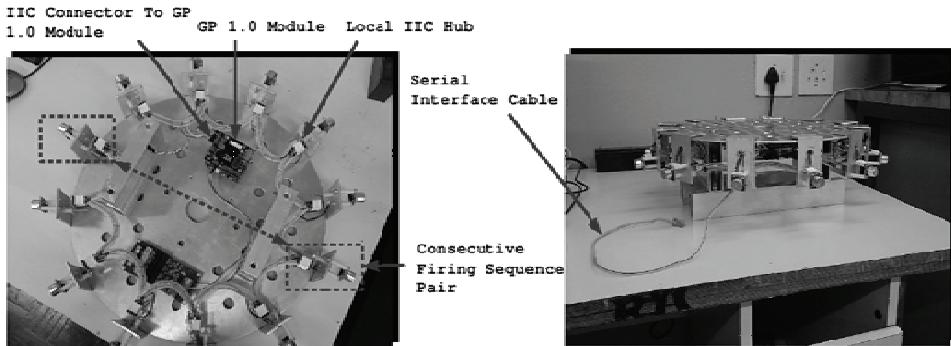
Figure 7: Sonar array sensor
The ultrasonic transducers operate at 40 KHz over an IIC bus interface.

4.1.3 Materials handling hardware

Material payload manipulation is provided by a rotary conveyor, which provides two extra degrees-of-freedom to a material payload: a translational and a rotational degree-of-freedom (Figure 8).
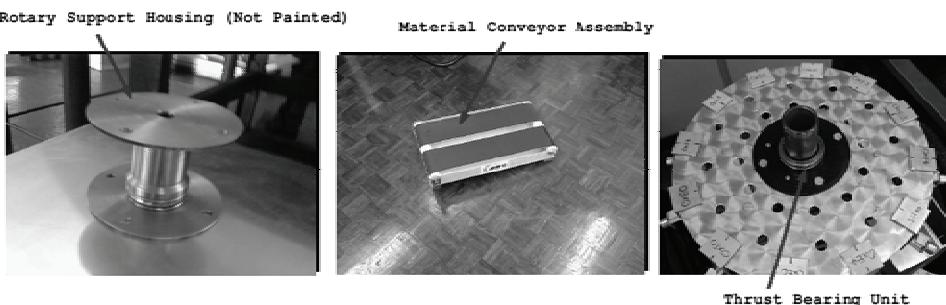
Figure 8: Materials handling rotary conveyor
Rotary support housing provides an angular degree-of-freedom, while the conveyor provides a translational degree-of-freedom.

The conveyor is driven via a DC motor, and an embedded microcontroller handles the alignment of the conveyor.

4.2 Device abstraction layer

To provide HAL functionality, the 'Player' robot device interface was used. Player is a C/C++ based software implementation analogous to the HAL in an operating system, and forms part of the Player/Stage/Gazebo project [10]. Player defines a set of interfaces, each specifying how user-based software applications can interact with a particular class of robotic device. Player has been designed and developed to run on an OS based on the POSIX[5] specification, such as Linux – both PC and embedded versions, Solaris and FreeBSD.

Player's most popular run-time implementation is in the form of a network scoped client/server model. Client applications communicate with the server, housed on the embedded PC on the mobile robot, over a TCP socket on an IP network. Client proxies are used to send and receive messages from the server, thus allowing client applications to control the mobile robot platform. This allows control code to be written in terms of generic device abstractions, thus increasing code re-use and scalability (Figure 9).

[5] Portable operating system based on Unix.
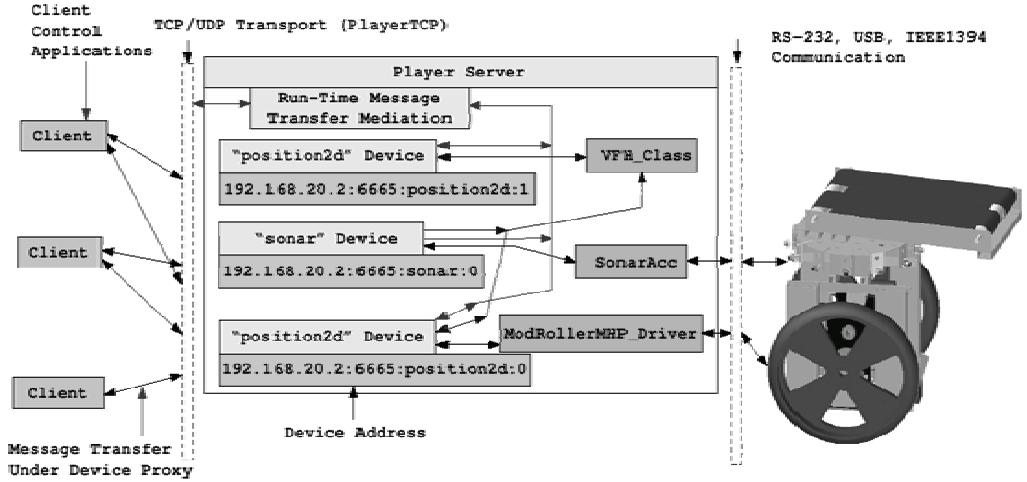
Figure 9: The run-time implementation of the 'Player' robot device interface in the form of a client/server model over a TCP/IP network

4.3 Task execution layer

Task execution is split according to RoC occupancy. Outside a RoC, motion is due to the outputs of global and local navigation functions. Currently, the mobile robot platform uses the Vector Field Histogram+ (VFH+) real-time obstacle avoidance algorithm [11]. This provides the platform with local navigation capabilities, and forms part of Player's code repository.

Inside a RoC, motion is in the form of posture stabilisation. Posture stabilisation describes the full state feedback control of a dynamic system, from an initial state to a goal state [13]. This can be regarded as regulating a state error via full state feedback control. A differential drive platform is under-actuated, having a state $[x, y, \theta]^T \in R^2 \times SO^1$, but having only two control inputs, $[v, \omega]^T \in R^2$. This makes differential drives nonholonomic and differentially constrained due to the 'rolling without slipping' condition exhibited by the active drive wheels. It is a known fact that, in order for a stabilise a controllable system via smooth time invariant full state feedback control, the closed loop vector fields of the resulting fully qualified system must be continuous around the goal state [12]. This restricts control application to nonlinear, time-varying, or piecewise continuous control laws.

The current mobile robot platform utilises a nonlinear control law that provides polynomial convergence of state errors. The control law is based on a polar coordinate transformation from Cartesian coordinates. The control law provides repeatable posture stabilisation from all initial conditions, and can be tuned online with ease [13] (Figure 10).
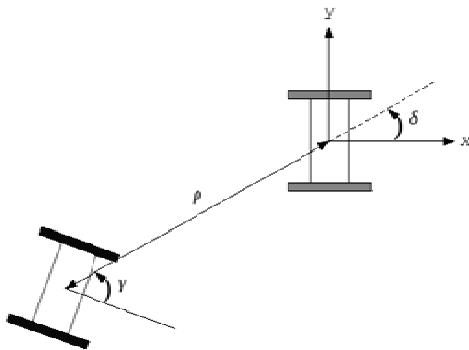
Figure 10: Polar coordinate transformation used in nonlinear control implementation

The coordinate transformation is set up as follows:

$$\rho = \sqrt{x^2 + y^2}$$
$$\gamma = \tan^{-1}\left(\frac{y}{x}\right) - \theta + \pi$$
$$\delta = \gamma + \theta$$

Using this new coordinate system, the kinematic model for a differential drive mobile robot has smooth vector fields with a singularity at the origin. In practical implementation of this control law, care must be taken to avoid instability as the state converges on the goal point. This can be handled by locking the control inputs upon entering a prescribed insignificantly small region around the goal point. The new system is characterised as follows:

$$\dot{\rho} = -\cos(\gamma)v$$
$$\dot{\gamma} = \frac{\sin(\gamma)}{\rho}v - \omega$$
$$\dot{\delta} = \frac{\sin(\gamma)}{\rho}v$$

The control law is implemented as follows:

$$v = k_1 \rho \cos(\gamma)$$
$$\omega = k_2 \gamma + k_1 \frac{\sin(\gamma)\cos(\gamma)}{\gamma}(\gamma + k_3)$$

where k_1, k_2, k_3, are the tuning parameters.

4.4 Task allocation layer

The task allocation layer (TAL) is currently under development and is not described here. Management of flexible payload routing systems in MCM is a complex subject, and requires research in all operational aspects of MCM.

5. MOTION CONTROL PERFORMANCE AND CHARACTERISTICS

Multiple tests were carried out to characterise and evaluate the performance of the nonlinear posture stabilisation controller. The controller performed well and showed good repeatability characteristics. Tests were set up by marking out point locations on the

laboratory floor at known absolute displacements from a goal point. The mobile payload routing platform was placed at these locations in a known configuration, and tasked with travelling to the goal location via full state feedback control. The endpoint of each convergence trajectory was marked, and the orientation of the platform was measured in order to quantify the steady state and repeatability performance of the controller.

Online tuning of the control law allowed for optimised convergence. This was achieved by tuning the controller to input maximum control input values on all locations on the border of a hypothetical RoC. The tuning parameters were first set at the following values: $k_1 = 0.4$, $k_2 = 0.3$, and $k_3 = 0.7$, and provided a response to an initial condition of [-3.00 -1.50 0.0]^T as shown in Figure 11.
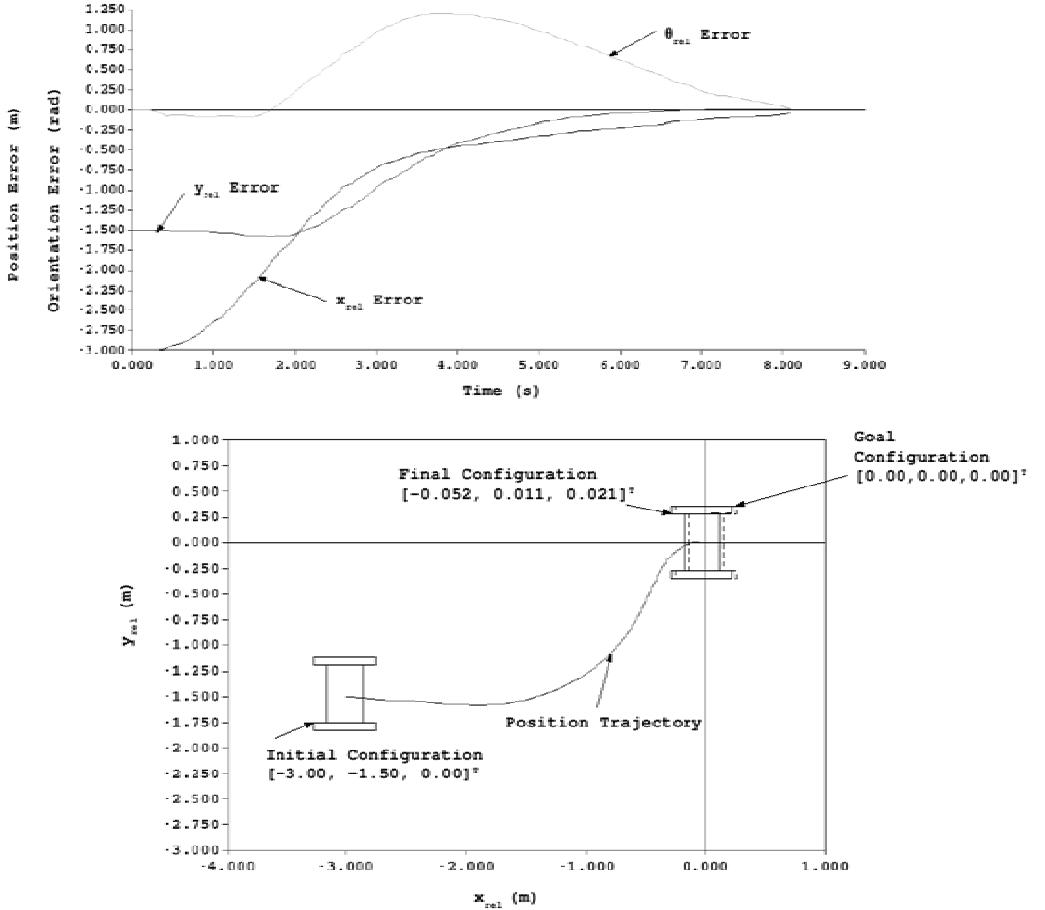

Figure 11: State convergence and position trajectory for k_1=0.4, k_2=0.3, k_3=0.7

The controller was tuned further, and was found to provide optimal performance at $k_1 = 0.5$, $k_2 = 0.4$, $k_3 = 0.6$. The response to the same initial conditions under the new tuning parameters is shown in Figure 12. As can be seen, convergence occurs 1.5 seconds earlier, and is time optimal in the sense that the platform starts moving at its maximum translational velocity of 1.5 m/s (Figure 13). Due to higher inertia, however, the steady state performance is slightly lower than the previous control parameters.
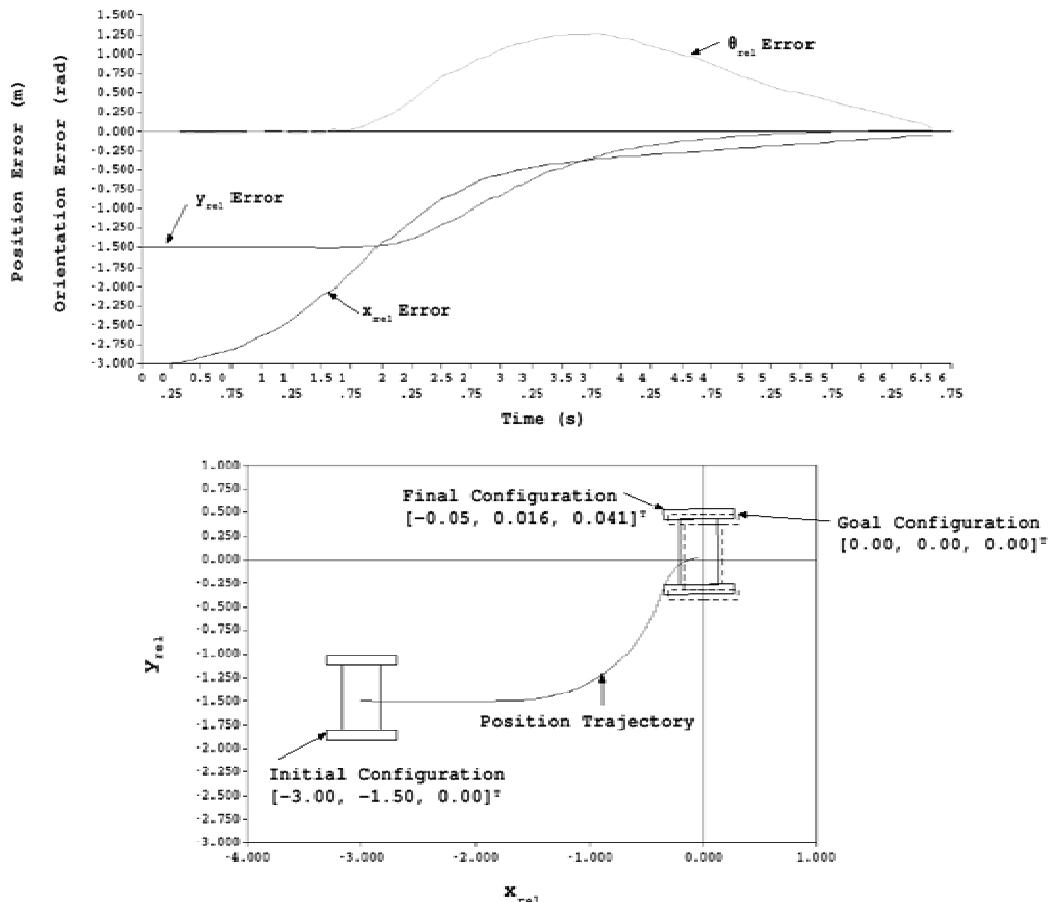
Figure 12: State convergence and position trajectory for k_1=0.5, k_2=0.4, k_3=0.6
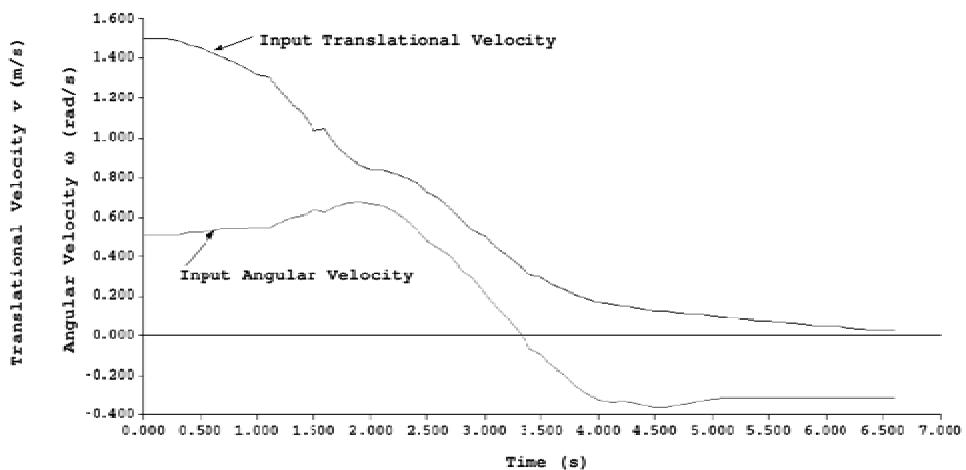

Figure 13: Control input convergence

In order to avoid the singularity at the origin of the polar system's state space, the angular and translational velocities are held constant during final approach to the goal point – i.e. the relative origin.

In order to prove the applicability and performance of this nonlinear control law, the mobile robot platform was requested to perform a parallel parking manoeuvre from an initial condition of $[0.00 \, -3.00 \, 0.00]^T$. This is the greatest test for posture stabilisation of a differential drive mobile robot platform. The response is shown in Figure 14.
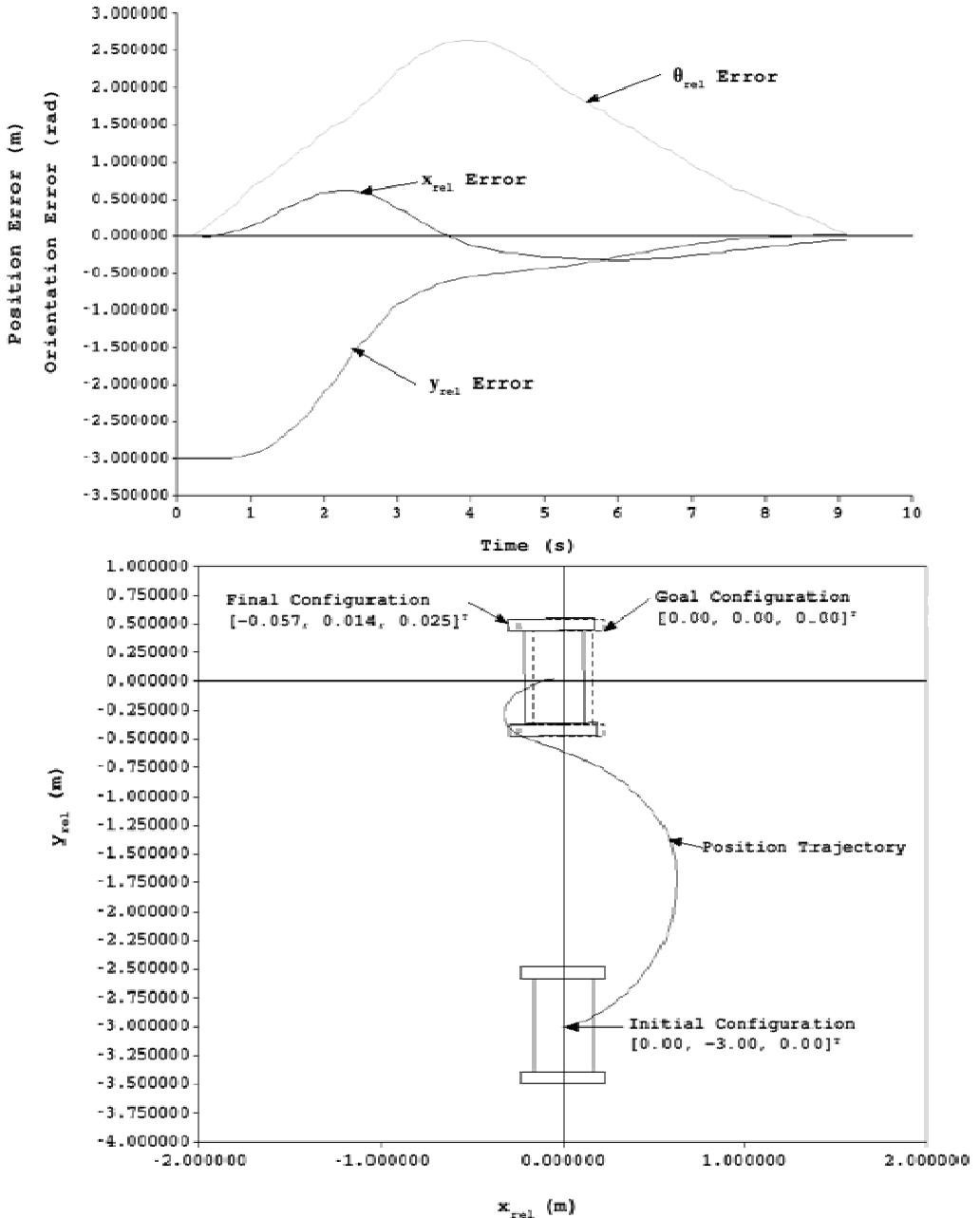

Figure 14: Parallel parking manoeuvre performed through full state feedback control

As can be seen in Figure 14, the sinusoidal descriptions in the governing equations that describe the platform's polar coordinate state-space produce smooth convergence. This convergence is smooth even under such difficult initial conditions. This parallel parking task

is the most difficult for differential drive platforms, as it is in direct conflict with their differential constraints.

6. DISCUSSION

In the context of facilitating modern niche markets and establishing first mover market share, the successful implementation of MCM production structures relies on a manufacturing firm's ability to integrate all available manufacturing resources constructively and concurrently. The control theoretic model and description of the production structure involved in MCM (recall Figure 1) provides semantic homogeneity for engineers across multiple disciplines. The authors consider this approach to be a way of establishing Common Model Development (CMD) in the modern research community. The standardisation of concepts brought on by CMD can provide the necessary infrastructure to establish international research partnerships, thus aiding the development of third world nations through constructive and relevant research and development practices.

It has been established in Section 2 that plant layout, as a passive measure of decreasing required materials handling, is just as important as the development of highly engineered autonomous materials handling and routing robots, in facilitating the materials handling requirements of customer-induced variations in production requirements. The uncertain, yet bounded, nature of customer-induced variations in production requirements plays a crucial role in establishing insight into the concurrent design and application of plant layout structures and active flexible materials handling infrastructure. This aspect of production implementation once again highlights the need for concurrent engineering. Papers such as [14], describing methods of applying probabilistic models to develop metrics that describe customer preference in product choice, are evidence that there is a research community that is involved in the research required to achieve efficient MCM production operations.

As this research was concerned with materials handling and routing in MCM production environments, the problem space associated with performing a materials handling and routing task between distributed, unconnected, manufacturing infrastructure required quantification in order to establish a well-defined problem space. The FMRP task definition provided in Section 2 takes into account the various forms of motion control required to provide robust material payload transfer to and from materials handling infrastructure via correct alignment, through posture stabilisation, of the mobile robot platform with the input/output port infrastructure of processing cells.

A material transportation primitive of a FMRP task essentially decomposes into a global path planning and local navigation and real-time obstacle avoidance problem. There is much literature on this subject in the mobile robotics research community – although, in the majority of the literature, the application scope covers unstructured environments, unlike those utilising structured plant layouts for production implementation. There is a need to develop path planning algorithms for mobile materials handling and routing robot platforms that include optimisations based on the knowledge of scheduling outputs in the production plant.

7. CONCLUSION

In this paper, the problem of establishing reconfigurable control and execution systems for flexible payload routing in MCM production environments was addressed. Some core concepts of MCM production were encapsulated in a control theoretic construct. This allowed for the characterisation of the particular material payload routing environment in MCM. A definition was developed that encapsulates a flexible routing and materials handling task. This allowed for the structured and axiomatic development of an implementation architecture (IA). A mobile robot platform was introduced, whose design was based on a subset of the IA, to research reconfigurable routing systems. Multiple motion control experiments were performed on the mobile robot platform to characterise the nonlinear posture stabiliser implemented on the mobile robot. The nonlinear posture

stabiliser behaved well, and was able to stabilise the mobile robot's pose on to a goal pose, including one that was in direct conflict with the mobile robot's differential constraints.

8. REFERENCES

[1]   **Petin, J.F., Gouyon, D. & Morel, G.** 2007. Supervisory synthesis for product-driven automation and its application to a flexible assembly cell, *Control Engineering Practice*, 15(5), pp 595-614.

[2]   **Michaloski, J., Birla, S., Weinert, G. & Yen, C.J.** 1998. A framework for component based CNC machines, in *Proceedings of the SPIE* (Sensors and Controls for Intelligent Machining, Agile Manufacturing, and Mechatronics), pp 132-143.

[3]   **Qui, R., Wysk, R. & Xu, Q.** 2003. Extended structured adaptive supervisory control of shop floor controls for an e-manufacturing system, *International Journal of Production Research*, 41(8), pp 1605-1620.

[4]   **Mikkola, J.H.** 2006**.** Capturing the degree of modularity embedded in product architectures, *Journal of Product Innovation Management*, 23, pp 128-146.

[5]   **Brennan, R.W., Zhang, X., Xu, Y. & Norrie, D.H.** 2002. A reconfigurable concurrent function block model and its implementation in real-time java, *Journal of Integrated Computer Aided Engineering*, 9.

[6]   **Muhl, E.**, **Charpentier, P. & Chaxel, F.** 2003. Optimization of physical flows in an automotive manufacturing plant: Some experiments and issues, *Engineering Application of Artificial Intelligence*, 16, pp 293-305.

[7]   **Balakrishnan, J.** & **Cheng, C.H.** 1998. Dynamic plant layout algorithms: A state of the art survey, *Omega*, 26(4), pp 507-521.

[8]   **Huang, H.** & **Irani, S.A.,** 2000. Custom design of facility layouts for multi-product facilities using layout modules, *IEEE Transactions on Robotics and Automation*, 16, pp 259-267.

[9]   **Shoval, J.** & **Borenstein, S.** 2001. Using coded signals to benefit from ultrasonic sensor cross-talk in mobile robot obstacle avoidance, *IEEE International Conference on Robotics and Automation*, Seoul, Korea, pp 2879–2884.

[10]   The official website of the Player/Stage/Gazebo project [Accessed 28/09/2009] http://playerstage.sourceforge.net

[11]   **Ulrich, I. & Borenstein, J.** 1998. VFH+: Reliable obstacle avoidance for fast mobile robots, In *Proceedings of the 1998 International Conference on Robotics and Automation*, pp 1572–1577.

[12]   **Brockett, R.W.** 1983. Asymptotic stability and feedback stabilization, In R. Brockett, R. Millman & H. Sussmann, *Differential geometric control theory*, Birkhauzer, pp 181-191.

[13]   **Vendittelli, M., De Luca, A. & Oriolo, G.** 2000. Control of wheeled mobile robots: An experimental overview, in *RAMSETE* (Articulated and Mobile Robots for Services and Technology), 270.

[14]   **Tseng, M.M. & Wany, Y.** 2008. Incorporating probabalistic model of customers' preferences in concurrent engineering, in *Annuals of the CIRP - Manufacturing Technology*, 57, pp 137–140.