A Multi-Agent Control Algorithm for Failure Resilience and Energy Reduction in Multi-Wheeled Payload Transport Platforms

Rahul Tallamraju[1], Venkatesh Sripada[2], Kamalakar Karlapalem[3]
[1,2,3]International Institute of Information Technology, Hyderabad.

Abstract

In this extended abstract we propose a novel distributed control algorithm for a multi-wheel robotic platform. The robotic platform considered consists of a number of multiple wheels with independent low power motors attached to each wheel. Each wheel with its associated motor are modeled as agents with limited sensing, battery and communication. The robot chassis has a provision to transport a payload. The control algorithm uses the concepts of multi-agent coordination, task allocation and scheduling to transport the payload efficiently. The goal of the control algorithm is to reduce energy usage and improve system failure resilience. The proposed end result is a real-time velocity profile and scheduled activation/deactivation cycles for each agent. The research aims to promote decentralization of multi-robot payload transport systems to achieve reduction in energy consumption and failure resilience behaviors.

1 Introduction

Multi-Wheel Robotic Platforms [1, 2] are popular and have several advantages over three or four wheeled robot systems, especially while traversing on a rough terrain because of the flexibility incorporated in their design. Such platforms have not been studied from the perspective of energy minimization, failure resilience and scalability of control algorithm to a larger number/swarm of wheels. This paper defines and makes use of two key properties of such multi-wheeled robotic systems:

1. *All the motors do not have to be powered all the time. For a given instant of time only a subset of total motors can be active so as to ensure the payload transport system is in motion.*

2. *Failure in functionality of a few motors should not affect the high level control algorithm.*

In order to reduce energy, each agent must identify a velocity profile and a schedule which defines the period of activation and deactivation of its motor. The concept of identifying an optimal velocity profile for a single agent robotic system with three or four wheels has been researched previously from an analytic and empirical perspective [3, 4]. These profiles define the optimal temporal changes in velocity for a straight trajectory to minimize total energy consumed by a wheeled robot. In this work, a multi-wheel chassis is used as an experimental robot platform. Each wheel in the platform is powered by an independent motor. Also, each of these wheels is modeled as an agent with limited battery power and processing capabilities. Each agent is also equipped with a communication module to communicate with other agents, thus making it a distributed multi-agent system. A control scheme is defined to transport the robotic platform from start to goal along a straight trajectory while maintaining a pre-defined velocity. This control scheme ensures the right amount of power is provided to the payload transport system to travel at the user-defined reference velocity. At its core, this control algorithm performs task allocation and scheduling to identify which set of motors to activate and deactivate to reach the reference velocity of robot.

Although the idea of using control algorithms to maintain a pre-defined velocity is not new [5], to the best of our knowledge, the idea of identifying and scheduling sets of active/inactive agents or motors in order to reduce energy and wear of motors has not been researched. Such a system is also resilient to failures in motors i.e. failure in a few of the motors would still not affect the high level task of transportation of payload. Scalability of control is another important aspect under consideration. The designed control algorithm must be scalable to a larger swarm of robots. Although each agent in this system is a single wheel/unicycle robot, this system must be scalable to differential drive robots as well. There are a few mechanical and electrical modeling challenges in such a system which would be highlighted in subsequent sections. The paper is organized as follows. Section 2 formally defines the problem and highlights the key ideas and assumptions in this work. Section 3 describes the mechanical structure of the robotic chassis and various design related challenges. Section 4 describes in brief the challenges w.r.t control of such a system, the agent models, the control hierarchy and a brief discussion of the control algorithm. Section 5 provides an overview of the goals to be achieved as part of the proposal and discusses further work.

2 Problem Statement

Given the multi-motor robotic system has accelerated to a fixed velocity v from rest along a trajectory with no curvature($\kappa = 0$). Assuming N agents/motors in the system, and all N agents were active during the period of accleration to velocity v, each agent now has utilized the same amount of power. Also, if the agents are equipped with a good low-level control algorithm each agent is moving approximately at the same angular velocity ω. To maintain velocity v, a subset of agents M decide to power their motors off. We now have less power supplied to the system with only $K = N - M$ agents active for the time step δt. Therefore, the set of inactive agents M represent the power advantage for that time step. At the next time step the agents collectively identify new sets of active and inactive agents using coordinated self-allocation of task.

2.1 Key Ideas and Assumptions:

- A robot travelling at a constant velocity v needs little additional power to keep it in motion at the same velocity i.e. maintain momentum.

- Although there are many control algorithms[ref] which target velocity maintenance, the proposed control algorithm differs from the classical control algorithms, by providing constant power to each agent but identifying two subsets of agents i.e. a) Active Agents/Motors b) Inactive Agents to ensure velocity maintenance.

- The shafts of the set of non-powered/inactive motors are free moving.

- Each agent is equipped with limited battery and processing capabilities.

- Localization and Velocity Measurement for initial experimentation happens through Visual SLAM or through a overhead camera based trajectory tracking.

- There are no heterogeneous agents which differ from the swarm of unicycle agents.

2.2 Centralized vs Distributed Control

Scalability of the control algorithm is an important aspect under consideration in this work. Current control algorithms for multi-wheel payload transport system use a centralized controller. Considering a centralized controller works well for a small number of wheels (eg: 8-10 wheels). The focus of our research is to design a controller for a swarm of small low-power motors. Designing a controller to plan velocity profiles for such a large number of robots centrally can be computationally very expensive and can suffer from a single point of failure. We therefore consider a distributed/decentralized control algorithm [7, 8, 9] where the swarm of agents make a collective

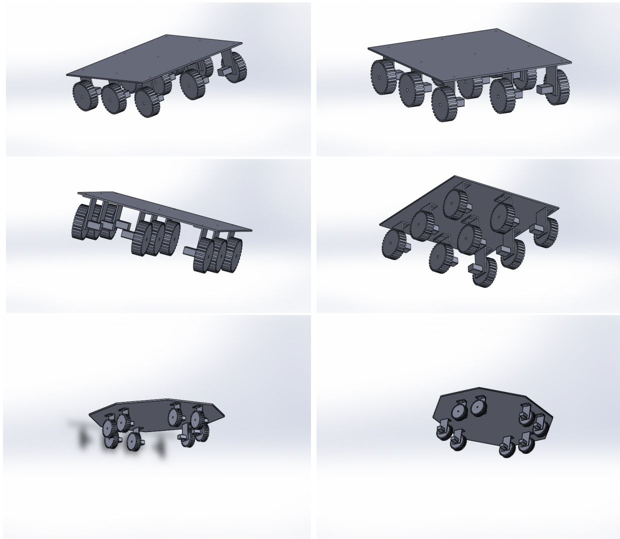

Figure 1: Representative figures of two different small-scale models of the proposed multi-agent/motor experimental setup. Differing chassis designs would result in different agent formations

decision to identify the set of active and inactive agents. This problem is planned to be solved as a distributed constraint optimization.

2.3 Failure Resilience

Motors are prone to failures and these failures result in incorrect trajectory tracking. Single controller, multi-wheeled robotic systems suffer from single point of failure w.r.t controller failures. In the multi-agent decentralized motor control algorithm, failure of a subset of motors does not affect the high level algorithm and hence a robust controller can be designed. The distributed nature of the algorithm ensures that the failure in any aspect of the agent does not lead to failure of the entire payload transport system. The division of the N agents into active/inactive agents ensures that each motor is not overused i.e for the entire duration of the path and is instead used only intermittently based on the battery power left in the associated agent battery. The motor wear would decrease considerably with increase in number of low-power motors(agents) in the system.

3 Experimental Setup, Design and Modelling Challenges

The figures in 1 depict some preliminary/representative mechanical constructions of the multi-wheel/agent payload transport system. Initial experiments would try to identify energy efficient formations for the unicycle robotic agents. Experiments using different symmetrically shaped chassis are performed to identify the formation which consumes least energy along a trajectory with zero curvature. Scheduling and task allocation is then experimented with the agents placed in accordance with the selected chassis design so as to identify the right set/group of active agents which consume least amount of energy. This system is then automated as a distributed constraint optimization problem to make it failure resilient and scalable.

3.1 Design Challenges

- Utilization of a large number of wheels would result in surface contacts and therefore high amount of friction. Hence, the affect of friction on the performance of the multi-agent system needs to be modeled.

- In an intermixed group of active and inactive agents, the active agents should be distributed such that there is no shift in the center of gravity and no couple effect is created about the center of gravity as we are considering straight line trajectories.

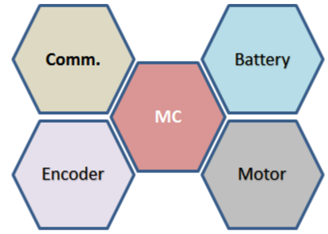

Figure 2: Agent Model

- Effect of in-rush motor currents to system performance due to repeated activation/deactivation cycles. Such currents could affect system energy consumption but failure resilient characteristics of the system remain unaffected.

3.2 Dynamic Validations

The need for dynamic validation is summaraized in the next few slides:

- Failure Resilient behavior of systems modeled from the perspective of dynamics of the multi-agent system.

- Concept of additivity of torque would mean that a certain minimum number of motors are required to keep the payload transportation system in motion. Handling situations where fewer motors than the defined minimum number of motors are available is important.

- Effect of adding a non-uniformly shaped payload or a payload with non-uniorm mass density.

- Analyzing the addition of unicycle-agents towards the center of mass of the chassis. To study if the system performance is adversely affected or aided by such an addition.

4 Control, Task Allocation and Scheduling

It is important to define the model/components of an agent before defining the control architecture. Figure 2 summarizes the agent model by depicting its various components. Each agent is equipped with limited battery power, a low power micro-controller/processor, a communication module to communicate with other agents, a motor which is the most power consuming component of the agent and limited sensing capabilities in wheel encoders. The control architecture of such a distributed multi-agent system can be defined in two broad layers [11] as shown in Figure 3 and Figure 4:

1. Low Level: Maintaining equal angular velocity for all wheels such that there is no differential velocity developed that would cause the robotic system to deviate from a straight trajectory.

2. High Level: Identifying the set of agents to activate and deactivate as a function of the energy consumed by each agent until the present time instant and velocity of the entire payload transport system.

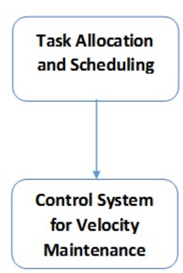

Figure 3: 2-Layer Control Architecture

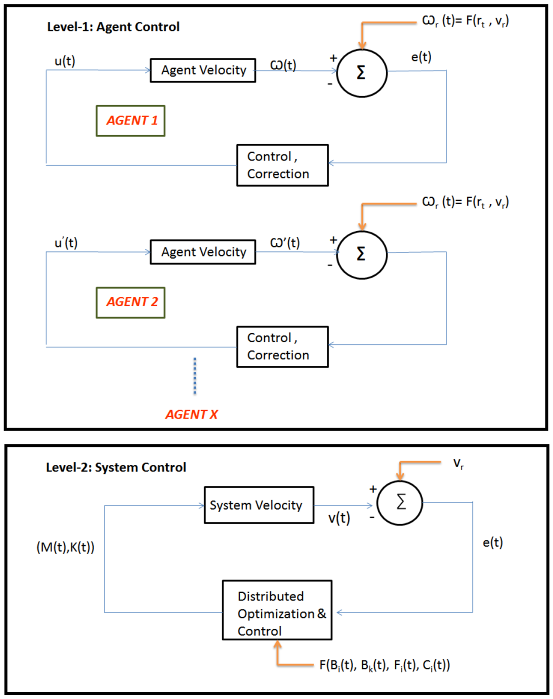

Figure 4: Internal Feedback Control Loops of the 2-Layer Control Architecture

Table 1: Example of Control and Scheduling Algorithm

Time	Active Agents (K_i)	Inactive Agents (M_i)	Velocity	$M_i > K_i$
T	N	0	v	False
$T+\delta t$	K_1	M_1	$v-\varepsilon_1$	True
$T+2\delta t$	K_2	M_2	$v+\varepsilon_2$	False
$T+3\delta t$	K_3	M_3	$v-\varepsilon_3$	True

.........so on........

Given that the robots have to move in a straight path, angular velocities of all agents must be approximately equal. To ensure this, a distributed control algorithm can be considered where, agents achieve the same motor angular velocity using a form of consensus [6]. Referring to Figure 4 for Level 1:Agent Control, the reference signal ω_r to be applied to each agent is a function of angular velocity of neighboring agents and the user-defined reference velocity v_r of the system. A correction control signal is then applied to the agent to get close to reference angular velocity. For Level 2:System Control, agents decide by themselves if they have to be active or inactive for the next time step as a function of power left in battery of itself($B_i(t)$), power left in battery of k other neighboring agents in the system ($B_{ik}(t)$), known failed agents ($F_i(t)$) and pairs of agents which form a couple about center of gravity of the payload transport system ($C_i(t)$) which are avoided to ensure straight movement of system.

4.1 Task Allocation and Scheduling

Given T is the time taken to accelerate from velocity 0 ms^{-1} to v ms^{-1}, then for time steps of δt after T, Table 1 provides a brief overview of how the control algorithm functions at each iteration by switching the number of active and inactive agents. After each iteration the error ε_i fluctuates over velocity v ultimately tending to *zero* after few iterations. The task allocation algorithm for agents would re-assign active and inactive agents at each iteration as agents must consume power uniformly. The number of agents in each set is defined by the error ε w.r.t reference velocity. The problem is looked at from a scope of minimizing overall energy utilized or ensuring that such a control algorithm performs just as good as existing control algorithms in terms of overall energy utilized for a unit battery life.
Figure 5 provides a representative schedule for an example similar to Table 1 where different sets of agents are active/inactive for different time steps.

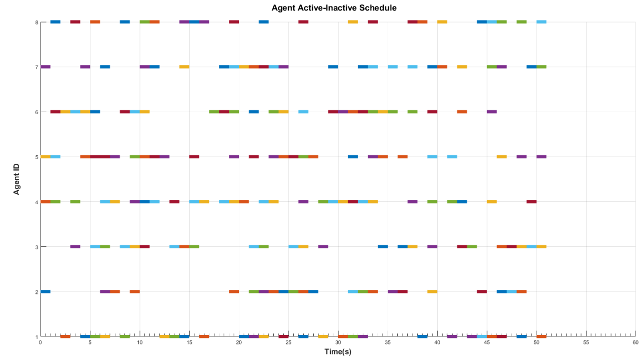

Figure 5: A Representative Figure of Schedule of Active/Inactive Agents for a time step of 1s. The colored lines represent that the agents are active for that time instant. A blank space for a time step represents that the agent is inactive for that time step.

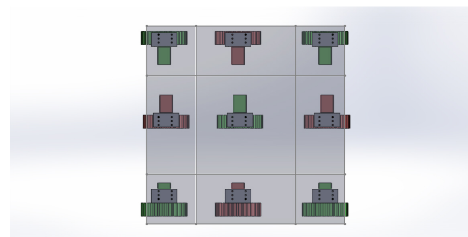

Figure 6: Top View: Red-Inactive Agent, Green-Active Agent

The schedule is plotted for 8 Agents for a duration of 50 seconds. The colors of active agents are randomly assigned for each time step. Figure 6 provides a top view of the payload transport system and depicts the active(green) and inactive agents(red) for an arbitrary time step.
The problem of distributed task allocation and schedule could be posed as a distributed constrained optimization problem(DCOP) [10] to reduce a cost function representing energy utilized. Consider the tuple:

$$\{A, V, D, f, \alpha\} \tag{1}$$

where A represents the set of all agents in the payload transport system. V represents the set of all variables associated with all agents and D defines the domain of the variables in V. f is the fitness or cost function of the payload transport system. α represents the function mapping variables to their associated agent.
The value function f depends on parameters like battery power left in agent A_i, battery power left in other agents in its neighborhood $A_j \forall j \neq i$ within a radius r of agent A_i, existing known failures in agents and the system velocity $v \pm \varepsilon$.
The output of the optimization function is a binary vector of length N defining active/inactive agents as *zero* or *one* for the next time step.

4.2 Challenges in Control

Since scalability of control is a challenge in itself, various considerations with respect to scalability can be summarized as follows.

- Angular velocity of each wheel should be approximately equal, to ensure the payload transport system does not drift.
- Distributed computation must ensure good timing synchronization between agents as there could be a large communication overhead or incorrect synchronization between start and stop of agents leading to drift in the system from straight-line trajectory.
- Failure detection and recovery must be performed quickly to avoid system drift.

5 Conclusions and Further Work

This extended abstract proposes a novel distributed control system for a swarm of agents/robots transporting a payload. Although we consider unicycle robots carrying the payload this algorithm should scale to a swarm of any type of differential drive robots and ensure failure resilience and reduced energy consumption. Research in relation to this proposal is still at a preliminary level and most of the challenges mentioned in this proposal are to be addressed in terms of their feasibility. To the best of our knowledge such an algorithm for payload transport has not been explored and hence research in this direction may explore new frontiers in future intelligent transport systems.

6 References

[1] Wilcox, Brian H., et al. "ATHLETE: A cargo handling and manipulation robot for the moon." Journal of Field Robotics 24.5 (2007): 421-434.

[2] Iagnemma, Karl, and Steven Dubowsky. "Mobile robot rough-terrain control (RTC) for planetary exploration." Proceedings of the 26th ASME Biennial Mechanisms and Robotics Conference, DETC. Vol. 2000. 2000.

[3] Tokekar, Pratap, Nikhil Karnad, and Volkan Isler. "Energy-optimal velocity profiles for car-like robots." Robotics and Automation (ICRA), 2011 IEEE International Conference on. IEEE, 2011.

[4] Velenis, Efstathios, and Panagiotis Tsiotras. "Optimal velocity profile generation for given acceleration limits: theoretical analysis." system 2 (2005): 5.

[5] Dorf, Richard C., and Robert H. Bishop. "Modern control systems." (1998).

[6] Ren, Wei. "Distributed Consensus in Multivehicle Cooperative Control: Theory and Applications."

[7] Michael, Nathan, et al. "Distributed multi-robot task assignment and formation control." Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on. IEEE, 2008.

[8] Arai, Tamio, Enrico Pagello, and Lynne E. Parker. "Editorial: Advances in multi-robot systems." IEEE Transactions on robotics and automation 18.5 (2002): 655-661.

[9] Brooks, Rodney A., and Jonathan H. Connell. "Asynchronous distributed control system for a mobile robot." Cambridge Symposium Intelligent Robotics Systems. International Society for Optics and Photonics, 1987.

[10] Modi, Pragnesh Jay, et al. "ADOPT: Asynchronous distributed constraint optimization with quality guarantees." Artificial Intelligence 161.1 (2005): 149-180.

[11] Brooks, Rodney A. "A robust layered control system for a mobile robot." Robotics and Automation, IEEE Journal of 2.1 (1986): 14-23.