Decentralized Control of Swarm Robots in Unknown Environment

Aman Sharma
B. Pranay
K. Naga Ganesh
Tejkaran Charan

Under the Supervision of
Mr. M. Manjunath
Department of Electrical and Electronics Engineering,
National Institute of Technology Goa

February 3, 2014

Overview

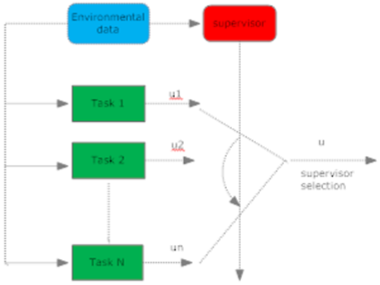
- Modelling of Differential drive Robot and Disk abstraction model for sensor network
- Go to Goal , Obstacle avoidance , Follow wall behavior
- Arbitration Mechanisms
- Sliding Mode control
- Solution to Navigation problem by Hybrid Automata model

What is Centralized and Decentralized approach ???

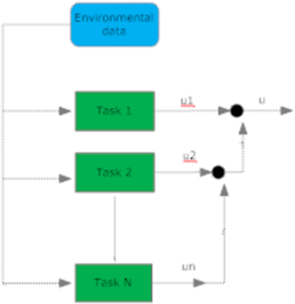
- Centralized approach- An approach where the decision making mechanism is handled by single robot which transmits and receives the order to the group of robots.
  Ex: Internet Hub.
- In the case of Decentralized approach every robot in the group acts independently.Further Each task is subdivided into subtasks.
  Ex: swarm of bees

Why Behavioral Control???
For point-to-point $\rightarrow$ Cost function is function of distance from the target and Control is a constant value.
If it meets Obstacle$\rightarrow$ Function must be modified to avoid the Obstacle
What the Problem $\rightarrow$ Driving function tries at the same time to reach towards the target And to avoid the Obstacle.
Solution ?? $\rightarrow$ We deal with the conflicting tasks

Competitive behaviour coordination

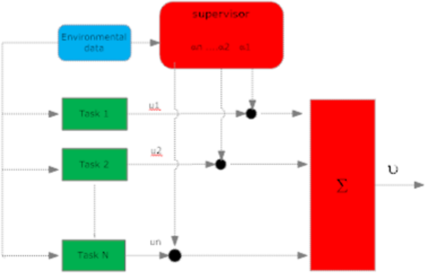


Supervisor can switch task-all the tasks have to be computed in parallel



Assigned priority level-As long as primary task is satisfied the secondary task is deactivated.

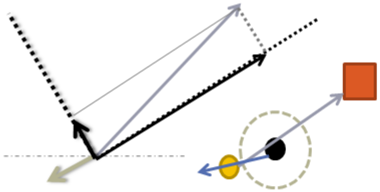
Cooperative behaviour coordination (motor schema)



Weights can be instantaneously changed by the supervisor, depending on the importance that each task. The motor schema has an output given by the linear combination of the weighted contribution provided by each task. High computational power is requested.

Drawbacks of cooperative and competitive approach

- Conflicting behaviors result in cooperative approach.
- Computation complexity is large in competitive approach



Introduction to swarm robotics

Figure: Swarm in Action

Swarm robotics is a field that focuses on controlling large-scale homogeneous multi-robot systems.It takes its inspiration from societies of insects that can perform tasks that are beyond the capabilities of the individuals. These systems are used to develop useful macro-level behaviors while being made of modules that are very simple in design and compact in size.

- The robots of the swarm must be autonomous robots, able to sense and actuate in a real environment.
- The number of robots in the swarm must be large or at least the control rules allow it.
- Robots must be homogeneous. There can exist different types of robots in the swarm, but these groups must not be too many.
- A single robot is generally incapable or inefficient respect to the main task it has to solve, therefore, they need to collaborate in order to succeed or to improve the performance.
- Robots have only local communication and sensing capabilities. It ensures the coordination is distributed, so scalability becomes one of the properties of the system.

Origin of NSB

- To eliminate the component of secondary task along the primary task.
- Directional filtering.
- A task, which is composed of sub-tasks, which may be conflicting at some point of time can be handled well by NSB.
- Not only restricted to Robotics, can be employed in any area, where prioritizing may be required to accomplish numerous conflicting subtasks

Indeterminate Systems

Consider a Linear System $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{R}^{M \times N}$, $\mathbf{x} \in \mathbb{R}^N$ and $\mathbf{b} \in \mathbb{R}^M$

A system is said to be indeterminate if number of variables are more than the equations, i.e. $M < N$.

Let us consider two different cases here:

1. $\text{rank}(\mathbf{A}) = M < N \longleftrightarrow$ Full Row Rank
2. $\text{rank}(\mathbf{A}) < M < N$

- Full Row rank leads to $N - M$ free variables or free *columns*, which inturn lead to a $N - M$ dimensional Null Space of $\mathbf{A}$, denoted by $\mathcal{N}(\mathbf{A})$ and the range space is entire $\mathbb{R}^M$.
- $\text{rank}(\mathbf{A}) = r < M < N$, lead to a $N - r$ dimensional Null space, but the range space shrinks to a r dimensional subspace in $\mathbb{R}^M$. Thus the feasible space reduces.

General Solution of Indeterminate System

In general, the solution to a Linear system $\mathbf{Ax} = \mathbf{b}$ is given by:

$$\mathbf{x} = \mathbf{x}_P + \mathbf{x}_N,$$

where $\mathbf{x}_P$ is called the particular solution and
$\mathbf{x}_N \in \mathcal{N}(\mathbf{A}) \longleftrightarrow \mathbf{Ax}_N = \mathbf{0}$
For a full row rank matrix, infinite number of solutions exist and the minimum norm solution, which is the particular solution, is given by:

$$\mathbf{x}_P = \mathbf{A}^+ \cdot \mathbf{b}$$

and $\mathbf{A}^+ = \mathbf{A}^T(\mathbf{AA}^T)^{-1}$
Minimum norm means: Min $||\mathbf{x}||_2$
s.t. $\mathbf{Ax} = \mathbf{b}$ and
$\mathbf{x}_N = \mathbf{P}\left(\mathcal{N}(\mathbf{A})\right) \cdot \mathbf{w}$, where $\mathbf{w} \in \mathbf{R}^N$ and $\mathbf{P}$ is Null Space Projector Matrix, given by $\mathbf{P} = (\mathbf{I} - \mathbf{A}^+\mathbf{A})$.

Mathematics of NSB

- Let $\mathbf{p}_i = [x_i \ y_i]^T$ and $\mathbf{v}_i = [v_{x_i} \ v_{y_i}]^T$ be the position and velocity of the i^{th} robot in 2-D space with a total of N robots and $\mathbf{p} = [\mathbf{p}_1^T \ \mathbf{p}_2^T \cdots \mathbf{p}_N^T]^T$, $\mathbf{v} = [\mathbf{v}_1^T \ \mathbf{v}_2^T \cdots \mathbf{v}_N^T]^T$ and $\mathbf{p} \in \mathbb{R}^{2Nx1}$, $\mathbf{v} \in \mathbb{R}^{2Nx1}$
- Let $\sigma(\mathbf{p}) \in \mathbb{R}^m$ be the task function to be controlled. This forward mapping let's us to compute the value of the task function in each iteration and compare it with the reference value σ_d.

$$\sigma = \begin{pmatrix} f_1(\mathbf{p}) \\ f_2(\mathbf{p}) \\ \vdots \\ f_m(\mathbf{p}) \end{pmatrix}$$

- Velocity $\mathbf{v}$ is chosen as the control signal and Position is computed using an inverse mapping.

- On differentiating: $\dot{\sigma} = \mathbf{J}(\mathbf{p}) \cdot \mathbf{v}$, where $\mathbf{J} \in \mathbb{R}^{mx2N}$

$$J = \begin{pmatrix} \nabla f_1^T \\ \nabla f_2^T \\ \vdots \\ \nabla f_m^T \end{pmatrix}$$

- On inverting the mapping: $\mathbf{v}_d = \mathbf{J}^+ \dot{\sigma_\mathbf{d}}$, where $\mathbf{J}^+ \in \mathbb{R}^{2Nxm}$
- Discrete Time implementation follows the Closed Loop Inverse Kinematic (CLIK) Dynamics, given by:

Control Signal

$$\mathbf{v}_d(t_k) = \mathbf{J}^+(\mathbf{p}_d(t_k - 1)) \left[\dot{\sigma}_d + \mathbf{K}\left[\sigma_d(t_k - 1) - \sigma_d(\mathbf{p}(t_k - 1))\right]\right]$$

Position Estimation

$\mathbf{p}_d(t_k) = \mathbf{p}_d(t_k - 1) + \mathbf{v}_d(t_k) \cdot \Delta t$, where t_k signifies the k^{th} sample and Δt denotes the sample time.

Resolving Confliction

1. To Accomplish a global objective $\longrightarrow$ Local Subtasks must be accomplished $\longrightarrow$ Formulation of various task function σ_i, $i \in 1, 2, \cdots, k$ and $i = 1$ be the highest priority task.
2. The control output command to the actuator of the system due to each subtask is given by:

$$\mathbf{v}_{d_i} = \mathbf{J}_i^{+}\left[\dot{\sigma}_{d_i} + \mathbf{K}_i\left(\sigma_{d_i} - \sigma_i\right)\right]$$

3. The control signal, at some time instant, may be conflicting in nature $\longrightarrow$ None of the task objective is individually fulfilled $\longrightarrow$ Need to take priority into consideration such that all subtasks are individually satisfied.

1. The component of the lower priority subtask, which conflicts with the higher priority task needs to be filtered out to yield the final control signal, concretely:

$$\mathbf{v}_d = \mathbf{v}_{d_1} + (\mathbf{I} - \mathbf{J}_1^{+}\mathbf{J}_1)\mathbf{v}_{d_2} \text{ and for three subtasks:}$$

$$\mathbf{v}_d = \mathbf{v}_{d_1} + (\mathbf{I} - \mathbf{J}_1^{+}\mathbf{J}_1) \cdot \left(\mathbf{v}_{d_2} + \left(\mathbf{I} - \mathbf{J}_2^{+}\mathbf{J}_2\right) \cdot \mathbf{v}_{d_3}\right)$$

2. This can be extended to q tasks.

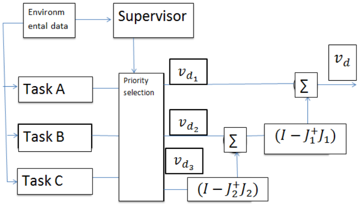
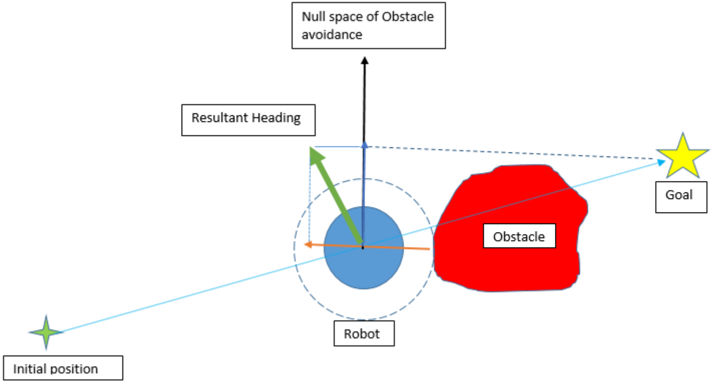


Figure: Block Diagram of NSB for three conflicting task


Null space of Obstacle avoidance
Resultant Heading
Goal
Obstacle
Robot
Initial position

Computer Simulation-Stability analysis of Swarm

Parameters	Value
Simulation Platform	MATLAB 2012a
No. of Robots (N)	3
Position of Robot 1 ($\mathbf{X}_1$)	$[20\ 10]^T$
Position of Robot 2 ($\mathbf{X}_2$)	$[12\ 19]^T$
Position of Robot 3 ($\mathbf{X}_3$)	$[13\ 04]^T$

Table: Simulation parameters

- Let the state space vector be $\mathbf{x} = [\mathbf{X}_1^T\ \ \mathbf{X}_2^T\ \ \mathbf{X}_3^T]^T \in \mathbb{R}^6$
- Let Intial state $\mathbf{x}_0 = [20\ 10\ 12\ 19\ 13\ 4]^T \in \mathbb{R}^6$
- Centroid $(\mathbf{G}) = [15\ 11]^T \in \mathbb{R}^2$
- Let the control signal be $\mathbf{u}_i = \dot{\mathbf{X}}_i = \mathbf{G} - \mathbf{X}_i \in \mathbb{R}^2$

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

$$\dot{\mathbf{x}} = A\mathbf{x}$$

$$\mathbf{A} = \frac{1}{3}\begin{pmatrix} -2 & 0 & 1 & 0 & 1 & 0 \\ 0 & -2 & 0 & 1 & 0 & 1 \\ 1 & 0 & -2 & 0 & 1 & 0 \\ 0 & 1 & 0 & -2 & 0 & 1 \\ 1 & 0 & 1 & 0 & -2 & 0 \\ 0 & 1 & 0 & 1 & 0 & -2 \end{pmatrix} \text{ Also by Diagonalization}$$

$\mathbf{A} = \mathbf{V}\mathbf{D}\mathbf{V}^{-1}$, where

$$V = \begin{pmatrix} 0.4082 & -0.6272 & 0.2732 & -0.1789 & -0.1673 & 0.5526 \\ -0.5774 & -0.2408 & 0.4669 & 0.2396 & 0.5526 & 0.1673 \\ -0.4082 & 0.4283 & 0.0047 & -0.5626 & -0.1673 & 0.5526 \\ 0.5774 & 0.5055 & 0.2770 & -0.0318 & 0.5526 & 0.1673 \\ 0 & 0.1989 & -0.2779 & 0.7415 & -0.1673 & 0.5526 \\ 0 & -0.2648 & -0.7439 & -0.2078 & 0.5526 & 0.1673 \end{pmatrix} = [\mathbf{V}_1 \cdots \mathbf{V}_6]$$

$$D = \begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$\mathbf{V}\mathbf{c} = \mathbf{x}_0 \longrightarrow \mathbf{c} =$
$[8.4621 \;\; 1.3200 \;\; 8.8643 \;\; 0.2707 \;\; 10.7062 \;\; 30.3871]^T = [c_1 \cdots c_6]$

$$\mathbf{x}(t) = \sum_{i=1}^{6} c_i \cdot \mathbf{V}_i \cdot e^{\lambda_i t}, \text{ where } \lambda_i \in eig(\mathbf{A})$$
$$\lim_{t\to\infty} \mathbf{x}(t) = c_5 \cdot \mathbf{V}_5 + c_6 \cdot \mathbf{V}_6 = [15 \;\; 11 \;\; 15 \;\; 11 \;\; 15 \;\; 11]^T$$

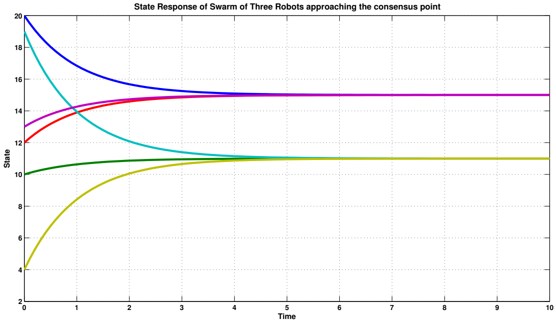


Figure: state response of the system

System is Stable

Rendezvous Problem

Converge to the centroid of the N-edge Polygon.

Parameters	**Value**
Simulation Platform	MATLAB 2012a
No. of Robots (N)	3,10,100
Robot Position	Randomly Initialized
Grid Size	100 X 100

Table: Simulation parameters

Pseducode for Rendezvous Problem

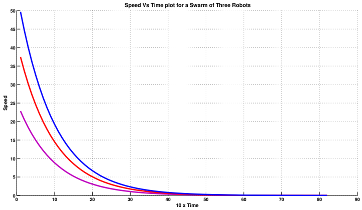
- N $\rightarrow$ Number of robots by user.
- X $\rightarrow$ Empty Matrix
- t $\rightarrow$ time
- for $\langle$ all N $\rangle$
  $\langle$ initialise (x,y) $\rightarrow$ X =[X Pos] $\rangle$
  end loop
- Calculation of Centroid

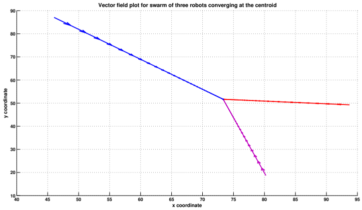
$$\langle C = \left(\frac{1}{N}\right) \sum [X_i] \rangle$$

- Solution to Rendezvous Problem
- Function $\rightarrow$ C - X
- Initialise Sum = 0
- for $\langle$ all N $\rangle$
  $\langle$ sum = sum + Function(C-X) $\rangle$
  end loop

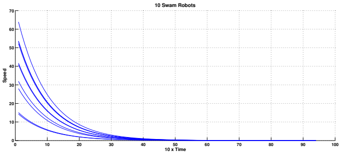
- Velocity $\rightarrow$ [Velocity MATRIX]
- Coordinate $\rightarrow$ [Position MATRIX]
- while (sum $>$ threshold)
  for $\langle$ all N $\rangle$
  $\langle$ New Position $\rightarrow$ [Position Matrix] $\rangle$
  $\langle$ New Velocity $\rightarrow$ [Velocity Matrix] $\rangle$
  $\langle$ Update sum $\rangle$
- RESULTS
- plot (Velocity Vector Field)
- plot (speed , time)
- end

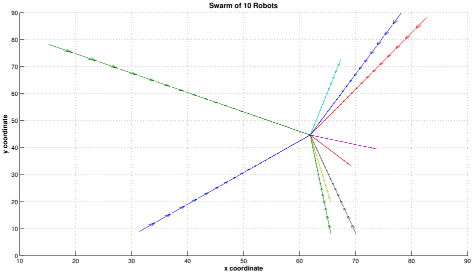
Results


Speed Vs Time plot for a Swarm of Three Robots
Speed
10 x Time
0
5
10
15
20
25
30
35
40
45
50
0
10
20
30
40
50
60
70
80
90

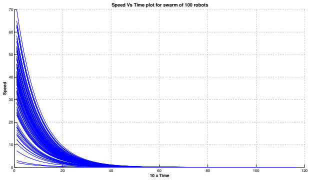

Vector field plot for swarm of three robots converging at the centroid
y coordinate
x coordinate
10
20
30
40
50
60
70
80
90
40
45
50
55
60
65
70
75
80
85
90
95

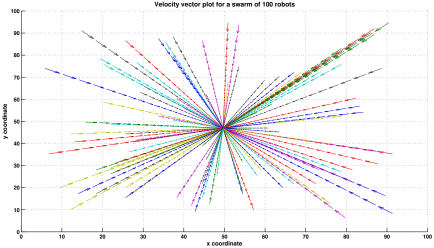
Results


10 Swam Robots
Speed
10 x Time
0
10
20
30
40
50
60
70
80
90
100


Swarm of 10 Robots
y coordinate
x coordinate
0
10
20
30
40
50
60
70
80
90

Results


Speed Vs Time plot for swarm of 100 robots
Speed
10 x Time


Velocity vector plot for a swarm of 100 robots
y coordinate
x coordinate

Solution to Rendezvous Problem in presence of obstacle

Parameters	**Value**
Simulation Platform	MATLAB 2012a
No. of Robots (N)	2
Position of Robot 1 ($\mathbf{X}_1$)	$[5\ 4]^T$
Position of Robot 2 ($\mathbf{X}_2$)	$[7\ 4]^T$
Position of Obstacle ($\mathbf{X}_0$)	$[6\ 6]^T$
Clearing Distance from Obstacle (r)	1

Table: Simulation parameters

Formulation of task function for Go-to-Goal and Avoid-Obstacle Behavior.

Priority

1. Obstacle Avoidance
2. Go-To-Goal

$$\sigma_{1i} = ||\mathbf{X}_i - \mathbf{X}_0||$$

$$J_{1_i} = \left[\frac{\mathbf{X}_i - \mathbf{X}_0}{||\mathbf{X}_i - \mathbf{X}_0||}\right]^T = \hat{z}_i^T$$

$$J_{1_i}^{+} = \hat{z}_i \cdot \hat{z}_i^T$$

$$\sigma_{1_{d_i}} = r$$

$$\mathbf{v}_{1_{d_i}} = J_{1_i}^{+}\mathbf{K}(r - ||\mathbf{X}_i - \mathbf{X}_0||)$$

$$\mathcal{N}(\mathbf{J}_{1_i}) = \mathbf{I} - \hat{z}_i \cdot \hat{z}_i^T$$

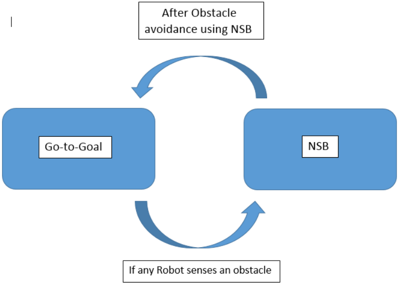
$$\sigma_{2_i} = \mathbf{X}_i$$

$$\sigma_{2_{d_i}} = \mathbf{X}_g$$

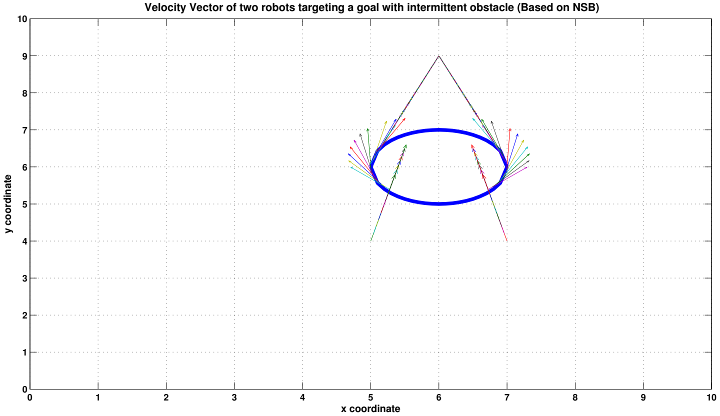
$$J_{2_i} = \mathbf{I} = J_{2_i}^{+}$$

$$\mathbf{v}_{2_{d_i}} = K_2(\mathbf{X}_g - \mathbf{X}_i)$$

$$\mathbf{v}_{d_i} = \mathbf{v}_{1_{d_i}} + \mathcal{N}(\mathbf{J}_{1_i}) \cdot \mathbf{v}_{2_{d_i}}$$



Results


Velocity Vector of two robots targeting a goal with intermittent obstacle (Based on NSB)
y coordinate
x coordinate
0
1
2
3
4
5
6
7
8
9
10

References

- S. Carpin and L. E. Parker. Cooperative motion coordination amidst dynamic obstacles.
- Maja J Mataric. Issues and approaches in the design of collective autonomous agents. Robotics and Autonomous Systems, 16:321331, December 1995.
- Linear Algebra by Gilbert Strang, Dept of mathematics. MIT