Distributed Cooperative Control for Stepper Motor Synchronization

Sauro J Yague[1,a], Guillermo Reyes Carmenaty[1], Alejandro Rolán Blanco[1] and Aurelio García Cerrada[2]

[1] *IQS School of Engineering. Ramon LLull University. Via Augusta 390. 08017 Barcelona (Spain).*
[2] *ICAI School of Engineering. Comillas Pontifical University. Alberto Aguilera 23. 28015 Madrid (Spain).*

Abstract. This paper describes the design and simulation of a distributed cooperative control algorithm based on multi-agents to synchronize a group of stepper motors. Modeling of the two-phase hybrid stepper motor in closed loop is derived in $\{d - q\}$ rotary reference frame, based on field-oriented control techniques to provide torque control. The simulation obtained by MATLAB-Simulink shows that the distributed cooperative control effectiveness depends on the network topology defined by the graph.

1 Introduction

Distributed and decentralized synchronization of large groups of dynamic systems is an area of intensive research [1]. In this article, a cooperative control algorithm will be applied to synchronize five stepper motors in a topology network defined by its graph.

A cooperative control law has been implemented based on a multi-agent dynamical system where agents are interconnected by a "fixed" communication network, and each agent or node is mathematically modeled by a dynamical linear time-invariant system, with communication delays being neglected.

Multi-agent systems consist of autonomous agents which are able to interact with each other and/or with their environment. During the last 20 years, the cooperative control of multi-agent system has been attracting great attention in biology to understand animal group behavior, in situations such as the flocking of birds, the schooling of fishes, and the swarming of bacteria. Results have been translated into engineering disciplines, like formation control of aircraft and robots.

The stepper motor model implemented in the simulation is firstly discussed in Section 2, and a detailed review on field-oriented control to regulate flux and torque separately is given in Section 3. The closed-loop position control system based on $\{d - q\}$ rotary reference frame is justified in Section 4. A review on algebraic graph theory is presented in Section 5, together with the cooperative control law designed to run the simulations. The cooperative control law will be evaluated under different network topologies, with simulation results shown in Section 6. A closing chapter in Section 7 will address final conclusions of the simulation runs.

2 Stepper Motor Model

---

[a] Corresponding author : sauro.yague@iqs.url.edu

The first stepper motor model available was manufactured with a six-salient-pole stator, a four magnetless salient-pole rotor, it had three phases, and rotated in thirty-degree steps. Since then, stepper motors have evolved significantly over the years, and modern hybrid two-phase stepper motors incorporate toothed stator and rotor, as well as a permanent magnet in the rotor. The two-phase windings in the stator are physically oriented on axes that are 90º apart from each other. The current flowing in each stator winding produces a magnetic field vector, and by controlling the currents in the two stator windings, a magnetic field of arbitrary direction and magnitude can be produced by the stator.

Hybrid steppper motors have smaller size, higher stepping rate and higher torque than variable reluctance stepper motors. The dynamic model for a two-phase hybrid stepper motor modeled by [2], consists of (1), (2), (3), and (4).

$$v_\alpha = Ri_\alpha + L\frac{di_\alpha}{dt} - K_m w \sin(N_r\theta) \tag{1}$$

$$v_\beta = Ri_\beta + L\frac{di_\beta}{dt} + K_m w \cos(N_r\theta) \tag{2}$$

$$T_L + Bw = J\frac{dw}{dt} - K_m i_\alpha \sin(N_r\theta) - K_m i_\beta \cos(N_r\theta) \tag{3}$$

$$\frac{d\theta}{dt} = w \tag{4}$$

where v_α and v_β are the phase voltages $[V]$, i_α and i_β are the currents $[A]$, w is the rotor speed $[rad/s]$, θ is the rotor angle $[rad]$, and T_L is the load torque $[Nm]$. The parameters of the motor are phase resistance $R = 10\Omega$, phase inductance $L = 6mH$, torque constant $K_m = 2Nm/A$, number of rotor teeth per phase $N_r = 25$, inertia of motor $J = 1.9 \times 10^{-2} kg/m^2$, and coefficient of viscous friction $B = 1 \times 10^{-3} Nms/rad$.

3 Field Oriented Control

For the last two decades, servomotors have evolved from largely brush types to brushless, providing higher reliability and lower maintenance. Most high performance servo systems employ an inner control loop to regulate torque, which is enclosed by outer control loops to control rotor angular velocity and rotor angle position. While the design of the outer loop is considerably independent of the motor type, the design of the torque loop is inherently specific to the motor being controlled.

Torque control in a brush motor is rather simple, since the motor commutates itself. A Proportional-Integral (PI) controller which adjusts the voltage applied to the stator windings is used, in the effort to minimize the error between the requested and measured motor current. However, brushless motors, like stepper motors, are not self-commutating, resulting in the necessity to control currents/voltages applied to the stator windings, to produce a useful torque.

The two stator windings in the stepper motor are physically oriented on axes that are 90º apart from each other, while brushless synchronous DC (BLDC) motors includes three stator windings physically separated 120º instead. In the stepper motor, the current flowing in one of the stator winding produces a magnetic field vector, which sums with the field from the second winding. By controlling currents in the two windings, a magnetic field of arbitrary direction and magnitude can be produced by the stator. Consequently, torque is produced by the attraction or repulsion between the net stator field and the magnetic field in the rotor. For any rotor position, when the stator field is orthogonal to the field produced by the rotor, the torque will be maximum, and when the stator field is in line with the rotor field, they both superpose to each other, and no torque will be produced.

A generic stator field can be decomposed into two components: one parallel to the rotor field (direct component), and the other orthogonal to the rotor field (quadrature component), where only the quadrature-component produces torque.

By means of modeling the field produced by the current stator windings, a conceptual current space vector is used. The current space vector for a given winding has the direction of the field it produces, and a magnitude proportional to the current going through it. This makes it possible to represent the total stator field as a current space vector which is the vector sum of the current phasor components, one for each of the stator windings. The stator current space vector is a fictitious current that would flow in a single winding rotating so as to produce the same stator field direction and magnitude as the combination of the real currents through the real stator windings. The current space vector for two stator windings in a stepper motor and three stator windings in a brushless synchronous DC motor are defined in (5) and (6), respectively.

$$\bar{i}_{sv} = \frac{2}{3}\left[i_\alpha(t) + i_\beta(t) \cdot e^{+j\frac{\pi}{2}}\right] \tag{5}$$

$$\bar{i}_{sv} = \frac{2}{3}\left[i_a(t) + i_b(t) \cdot e^{+j\frac{4\pi}{3}} + i_c(t) \cdot e^{-j\frac{4\pi}{3}}\right] \tag{6}$$

In order to efficiently produce constant smooth torque, the stator current space vector should, ideally, be constant in magnitude and should turn with the rotor so as to always be in quadrature with it, regardless of the rotor position or speed. While the stator current space vector may be constant in magnitude and direction if viewed from the rotating reference frame of the rotor, from a fixed frame viewpoint the current space vector describes a circle as the rotor turns. Since the current space vector is produced by the vector sum of components from each of the motor windings, and due to the fact that the two windings in the stepper motor are physically orientated on axes that are 90º apart from each other, the motor currents should ideally be two sinusoidal waveforms, with a 90º phase shift.

In the effort to generate a smooth sinusoidal modulation of the motor currents as the stepper motor turns, an accurate measurement of rotor position is required (e.g. using high resolution encoders). These two sinusoidal current command signals are provided as inputs to a pair of PI controllers that regulate the current in the two stator windings. Tracking a sinusoidal waveform by a PI controller has some limitations at high rotor speeds. High rotor speed results in large errors which perturb the direction of the current space vector relative to the rotor, causing it to shift away from the quadrature direction, reaching a situation where no torque is delivered.

The limited bandwidth of PI controllers is tackled by using field oriented control [3]. The current space vector is controlled directly in the $\{d-q\}$ rotary reference frame of the rotor. In the ideal case, the current space vector is fixed in magnitude and direction (in quadrature) with respect to the rotor. Since the current space vector in the $\{d-q\}$ rotary reference frame is static, the PI controllers operate on DC values, rather than in time varying sinusoidal signals. This isolates the controller from the time varying winding currents and voltages, and therefore eliminates the limitation of the controller frequency response, proving an efficient torque control. For this reason, the measured stator currents in the stepper motor must be mathematically transformed from the two phase $\{\alpha-\beta\}$ static reference frame, to a $\{d-q\}$ rotary reference frame, before being processed by the PI controllers.

For brushless DC synchronous motors (BLDC), with three-phase stator windings, with phases physically separated 120º, the reference frame transformation can be performed in a single step: from three phase $\{a,b,c\}$ stationary frame to $\{d-q\}$ rotary frame, using Park's Transformation [4]. However, that transformation is best described as two step process, where the motor currents are first translated from the 120º physical frame of the motor stator windings to a fixed orthogonal reference frame. This transformation is known as Clark Transformation. Two PI controllers will be required to control the $\{d-q\}$ currents, and since only the quadrature current iq produces useful torque, the iq controller will track the requested torque, and the id controller will be configured with a zero input reference.

Figures 1 and 2 summarize stator winding current waveforms in a BLDC motor, as seen in three different reference frames.
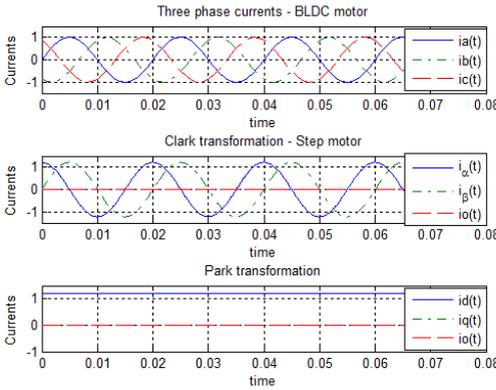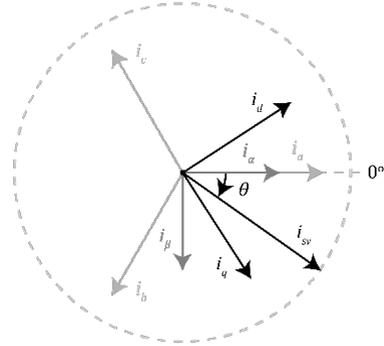

Figure 1. Clark and Park Transformations.

$\{i_a, i_b, i_c\}$ and $\{i_\alpha, i_\beta\}$ are stationary reference frames, while $\{i_d, i_q\}$ is a rotary reference frame

Figure 2. Stationary and rotary reference frame.

4 Position Control for a Stepper Motor

Two-phase hybrid stepper motors require a dual H bridge driver, and work reasonably well in open loop by half-stepping, full-stepping or micro-stepping.

However, most high performance servo systems employ an inner control loop to regulate torque, which it is enclosed by an outer control loop to handle rotor angular velocity, and followed by a second outer control loop for rotor angle positioning. While the design of the outer loops are considerably independent of motor type, the design of the torque loop is inherently specific to the motor being controlled. The block diagram in Figure 3, shows the three control loops employed at each stepper motor included in the group.
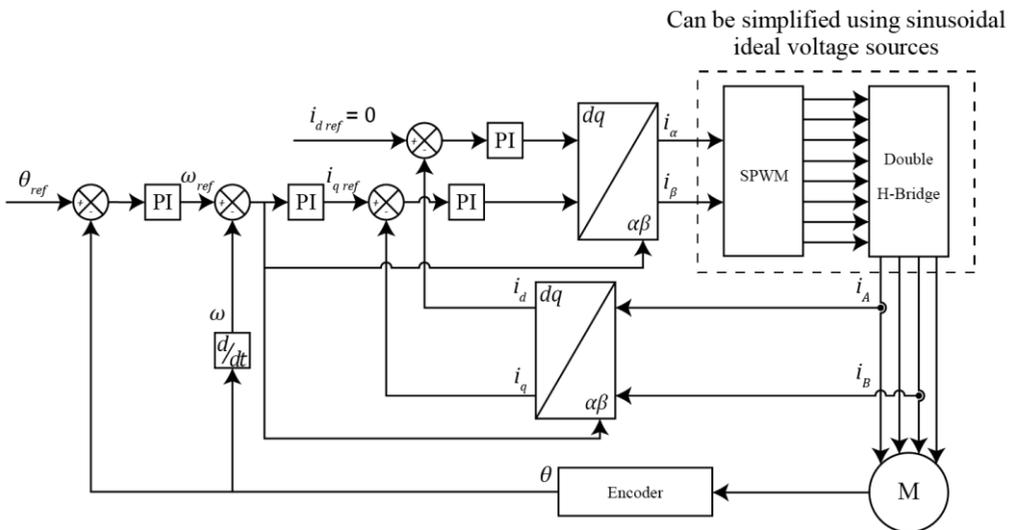

Figure 3. Block diagram for stepper motor position control

5 Distributed Cooperative Control

Cooperative control studies the dynamics of multi-agent dynamical systems linked to each other by a communication graph. The graph represents the allowed information flow between the agents, where each agent is confined to depend only on information about that agent and its neighbors in the graph [5].

Decentralized control systems refer to the case in which the agents are not following a centralized master/slave configuration. Instead, they are distributed within the system in a way that each agent is connected into a network of agents. Some of the advantages of distributed control systems are that a multi-agent system can perform tasks more efficiently than a single agent, increasing tolerance to possible agent failure. To coordinate with other agents in a network, every agent needs to share information with its neighbor peers, so that all can agree on a common goal of interest, like for instance, keeping the heading of aircraft in formation, or a target trajectory for a team of robots.

It was Reynolds [6] the first to propose a computer animation model to simulate collective behavior of multiple-agents, and after him several researches have provided some new in-sites of systematic framework of consensus problems. While most literature concentrates on studying consensus under fixed communication topology, in some applications the communication topology might change due to communication range limitations, mainly when agents are moving in space, like in the case of flight formation or robots synchronization.

5.1 Algebraic Graph Theory

The communication network is modeled by a graph and establishes the links and interconnections between the agents. The graph consists of nodes representing the agents, and directed edges corresponding to the allowed flow of information between the agents. Some basic graph theory concepts [7] are essential in the study of multi-agent dynamical systems.

A weighted order n graph is defined as a pair $G = (V, E)$, with a finite nonempty set of n nodes $V = \{v_1, \ldots, v_n\}$, a set of edges or arcs $E \subseteq V \times V$, and a weighted adjacency matrix $A = (a_{ij})_{n \times n}$. Each node v_i corresponds to an agent i, and each edge denoted as $(v_i, v_j) \in E$, is represented as an arrow from v_i (the tail) to v_j (the head). We assume the graph is simple, i.e., $(v_i, v_i) \notin E$, meaning that at any agent i there will be no "self-loops", and there will be no multiple edges between the same pairs of nodes.

Each edge $(v_i, v_j) \in E$ in a weighted direct graph (or digraph) represents the allowed flow of information from agent i to agent j, which means that agent j can received information from agent i. In contrast, the pairs of nodes in undirected graphs (also known as bidirectional graphs) are unordered, where an edge $(v_i, v_j) \in E$ denotes that agents i and j can receive information from each other.

The weighted adjacency matrix $A \in R^{n \times n}$ of a digraph G is defined by: a_{ji} weights of the link (v_i, v_j) and $a_{jj} = 0$ for any $v_j \in V$, $a_{ji} > 0$ if $(v_i, v_j) \in E$, and $a_{ji} = 0$ otherwise.

The weighted adjacency matrix A of a weighted undirected graph is defined analogously except that $a_{ij} = a_{ji}$, $\forall i \neq j$, since $(v_i, v_j) \in E$ implies $(v_j, v_i) \in E$, resulting in a symmetric adjacency matrix A.

The weighted in-degree of a node v_i is the number of edges having v_i as a head, so it will be equal to the i-th row sum of A:

$$d_{in} = \sum_{j=1}^{n} a_{ij} \tag{7}$$

and the weighted out-degree of node v_i is the number of edges having v_i as a tail, so it will be equal to the i-th column sum of A:

$$d_{out} = \sum_{i=1}^{n} a_{ij} \tag{8}$$

The in-degree and out-degree are local properties of the graph. A (directed) tree is a connected digraph where every node except one, called the root, has in-degree equal to one. A spanning tree of a digraph is a directed tree formed by graph edges that connects all the nodes of the graph. A graph is said to have a spanning tree, if a subset of the edges forms a directed tree meaning that there is a node (called root node) with a direct path from that node to every other nodes in the graph with no cycles (a cycle is a simple path that starts and ends at the same node). The root set or leader set of a graph is defined as the set of nodes that are the roots of all spanning trees. A graph may have multiple trees; however, if it contains at least one spanning tree, the graph is declared as strongly connected.

The adjacency matrix A of an undirected graph is symmetric, $A = A^T$. A graph is said to be weight balaced if the weighted in-degree equals the out-degree for all i. If all the nonzero edge weights are equal to 1, this is the same as the definition of balanced graph. An undirected graph is weight balanced, since if $A = A^T$, then the i-th row sum equals the ith column sum.

The Graph Laplacian matrix is defined as follows:

$$L = D - A \qquad (9)$$

where $D = diag\{d_{in}\}$ is the in-degree matrix, and A the adjacency matrix. Note that L has all row sums equal to zero.

Many properties of a graph may be studied in terms of its graph Laplacian matrix, providing a key role in the analysis of dynamical multi-agent systems on graphs. The eigenvalues of the Laplacian matrix explains properties of the underlying graph topology.

Any undirected graph has $L = L^T$, so all its eigenvalues are real and can be ordered as $\lambda_1 \le \lambda_2 \le ... \le \lambda_n$.

A weighted undirected graph is connected, if and only if, its Laplacian matrix L has an eigenvalue zero with multiplicity one, and all other eigenvalues have positive real parts [8].

A weighted directed graph has a directed spanning tree, if and only if, its Laplacian matrix L has an eigenvalue zero with multiplicity one, and all other eigenvalues have positive real parts [9].

5.2 Cooperative Control Algorithm

When multiple agents agree on the value of a variable of interest, they are said to have reached consensus. Consensus algorithms are designed to be distributed, assuming only neighbor-to-neighbor interaction between agents, so agents will update the value of their information based on the information of their neighbors. Hence, the goal will be to design an updated law, so that the information of all agents in the network converge to a common value.

Consensus algorithms have been studied extensively in the context of cooperative control of multi-agents. The most general continuous time consensus algorithm was formulated by [10], and it is given by the first order single integrator dynamics:

$$\dot{x}_i = \sum_{j=1}^{n} a_{ij}[x_j - x_i] \qquad (10)$$

where a_{ij} is the (i, j) entry of the adjacency matrix $A \in R^{n \times n}$ associated with the graph G, and x_i is the information of the ith agent. Setting $a_{ij} = 0$ denotes the fact that agent i cannot receive information from agent j. A consequence of equation (10) is that the information of agent i is driven toward the information of its neighbors.

Equation (10) can be written in matrix form, considering the global dynamics of the state vector $x = [x_1 ... x_n]^T \in R^n$ as:

$$\dot{x} = -(D - A)x = -Lx \qquad (11)$$

According to [10], the cooperative control law in (11) guarantees consensus, if and only if, the graph has a spanning tree, meaning that the dynamics given by (11) has a system matrix $-L$ with eigenvalues in the left-hand complex plane, except for only one of the eigenvalues, which will be located at the origin (stability condition).

In this research paper, distributed cooperative control of multi-agent systems has been applied for rotor synchronization of five stepper motors, with a communication network topology defined by their graphs. The cooperative control law applied is the following:

$$\partial\theta_i = -k\left[\sum_{j=1}^{n} a_{ij}\left[\theta_i - \theta_j\right] + g_i\left[\theta_i - \theta_{ref,i}\right]\right] \qquad (12)$$

where k is the control gain, a_{ij} is the adjacency matrix defined by the graph network topology, g_i is the pinning gain, with a non-zero value for only one of the stepper motors that has the reference rotor position angle (leader agent), and $\partial\theta_i$ is the rotor angle tracker error.

The cooperative control law in (12), can be written in matrix form as shown in (13), and its implementation in block diagram will be as shown in the Figure 4.

$$\begin{bmatrix} \partial\theta_1 \\ \partial\theta_2 \\ \partial\theta_3 \\ \partial\theta_4 \\ \partial\theta_5 \end{bmatrix} = -K\left[(L+G)\begin{bmatrix} \theta_1 - \theta_{ref,1} \\ \theta_2 - \theta_{ref,2} \\ \theta_3 - \theta_{ref,3} \\ \theta_4 - \theta_{ref,4} \\ \theta_5 - \theta_{ref,5} \end{bmatrix}\right] \qquad (13)$$

where $K = diag(k)$ is the control gain matrix, $G = diag(g_i)$ is the pinning matrix, $L = D - A$ is the Laplacian matrix, A is the adjacency matrix, and D is the in-degree matrix of A.
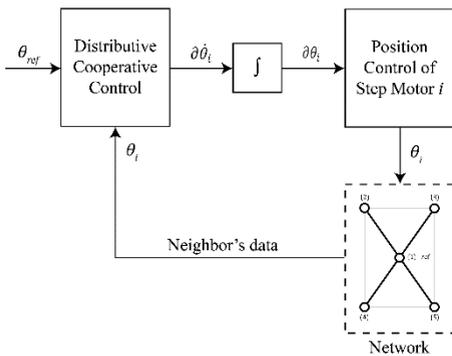

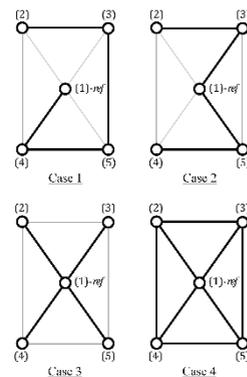**Figure 4.** Cooperative control algorithm.

Figure 5. Network topology-graphs.

6 Simulation Results

A distributed cooperative control law based on multi-agents has been used to synchronize five stepper motors, where each agent will communicate with its neighbors according to a predefined network topology given by the graphs shown in Figure 5.

Leadership is appointed to one of the stepper motors by the pinning matrix G, to which reference signal is applied, for the other agents to follow according to the cooperative control law defined at (12). Simulations are analyzed for a square, and triangular reference signal, with results shown in Figures 6-to-10.

Even though redundant communication links were assigned in the network of case 4, same results as in case 3 are obtained. The topology defined in case 1 is the most unfavorable, because each agent delays the information to the next agent, as confirmed by the simulation response.

7 Conclusions

The cooperative control algorithm implemented to synchronize rotor position of five stepper motors, with the internal closed-loop rotor position control derived based on field-oriented control techniques to provide torque control, gives different results depending on the topology of the selected network.

From the experiments carried out, it could be concluded that the network topology plays an important role when cooperative control based in multi-agents is considered. Hence, an optimization tool to find the optimal communication graph, attending to minimize a predefined cost function, could be an interesting future research topic.
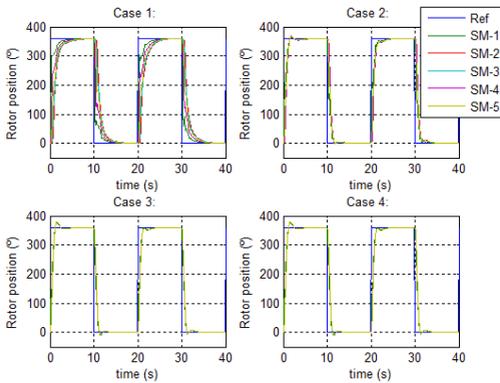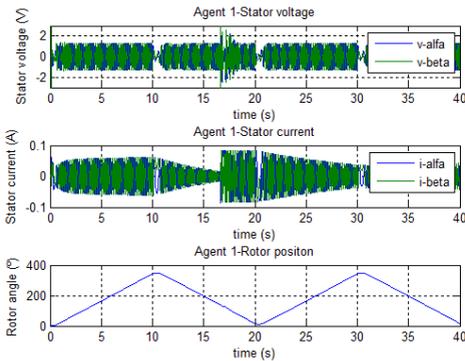

Figure 6. Square reference signal.

Figure 7. Triangle reference signal.
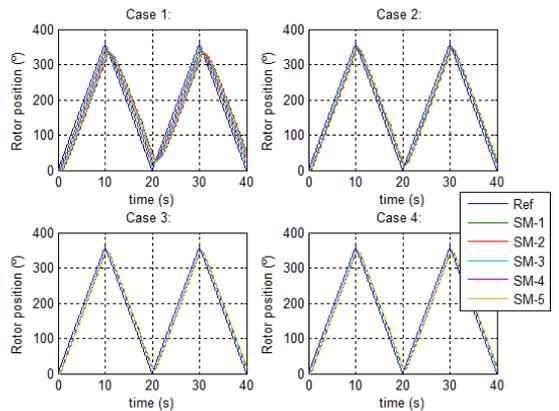

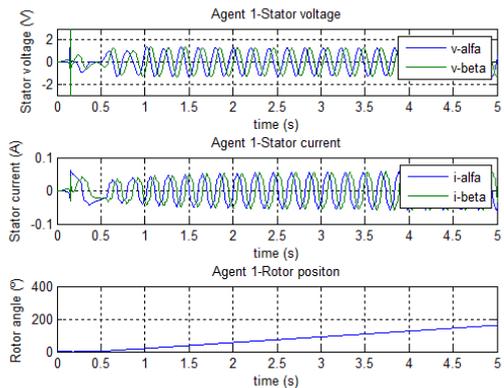**Figure 8.** Stepper motor voltage and current.

Figure 9. Stepper motor voltage and current.
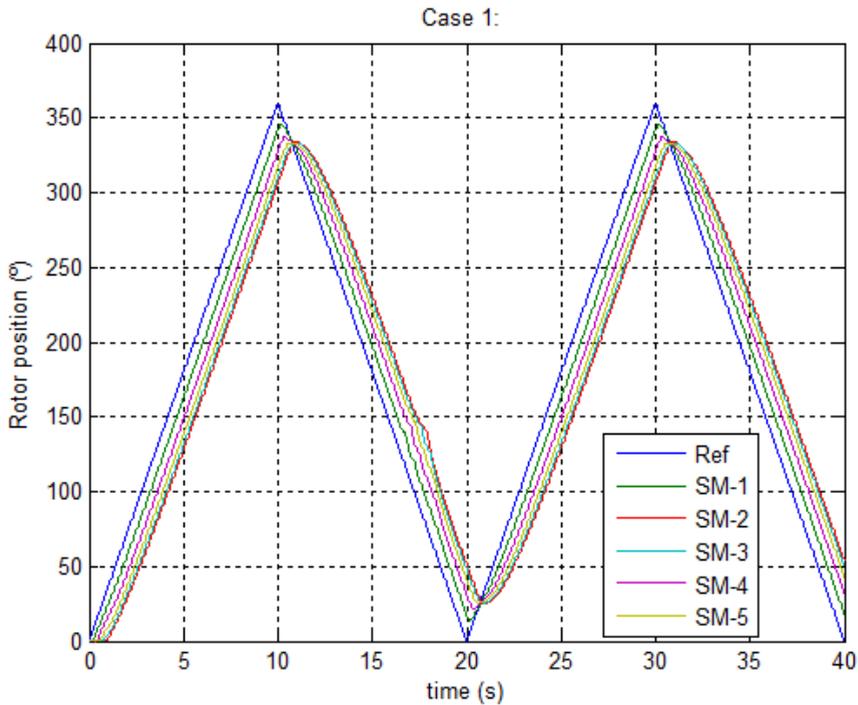
Figure 10. Delay due to the communication among agents.

Acknowledgement

References

1. Richard M Murray. Recent research in cooperative control of multi-vehicle systems. *International Conference on Advances in Control and Optimization of Dynamical Systems*, pages 1–27, 2007.
2. Takashi Kenjo and Akira Sugawara. *Stepping Motors and Their Microprocessor Controls*. Oxford University Press, 1995.
3. G. Brandenburg J. Heinzl H. Kellermann, P. Hildinger. Field oriented position control of a hybrid stepper motor. *In Proceedings of the EPE Conference*, pages 908–913, 1995.
4. R.H. Park. Two-Reaction Theory of Synchronous Machines, Generalized Method of Analysis, Part I. *Winter Convention of the A.I.E.E., New York*, 1929.
5. Kristian Hengster-Movric Frank L. Lewis, Hongwei Zhang and Abhijit Das. *Cooperative Control of Multi-Agent Systems*. Springer, 2014.
6. Craig W. Reynolds. Flocks, herds and schools: A distributed behavioral model. *ACM SIGGRAPH Computer Graphics*, (4):25–34, 1987.
7. Jorge Cort. *Distributed Control of Robotic Networks A Mathematical Approach to Motion Coordination Algorithms*. 2009.
8. F. R. K Chung. *Spectral Graph Theory*. American Mathematical Society, 1997.
9. Lixin Gao, Daizhan Cheng, and Jinhuan Wang. On consensus of multiagent systems under dynamically changing interaction topologies. *2007 IEEE International Conference on Control and Automation, ICCA*, pages 7–11, 2008.

[10] Ali Bidram, Frank L. Lewis, Ali Davoudi, and Zhihua Qu. Frequency control of electric power microgrids using distributed cooperative control of multi-agent systems. *2013 IEEE International Conference on Cyber Technology in Automation, Control and Intelligent Systems*, pages 223–228, 2013.