Lightweight Distributed Adaptive Algorithm for Voting Procedures by using Network Average Consensus

Clement Duhart[1 2], Michel Cotsaftis[1], and Cyrille Bertelle[2]

[1] LACSC, ECE Paris School of Engineering, FRANCE,
<duhart,mcot>@ece.fr
[2] LITIS, Havre University, FRANCE
cyrille.bertelle@univ-lehavre.fr

Abstract. Consensus Seeking (CS) is an important research area in which this work contributes by proposing a new distributed algorithm to solve Voting Procedures (VP) problem in mobile network with link failures under strong constraints on communication capacity.

Keywords: Multi Agent System (MAS), Network Average Consensus (NAC), Wireless Sensor Network (WSN), Voting Procedures (VP)

Introduction

Synopsis. In Multi Agent System (MAS) community, Consensus Seeking (CS) has been an attractive research for a long time with recent emphasis by using new approaches through the Network Average Consensus (NAC) framework. Initially applied in Dynamic System (DS) and particularly in nonholonomic systems, it is a very powerful framework to ensure coordination of autonomous mobile networks performing distributed tasks like in Multi-Vehicle Systems. From good properties in terms of robustness according to switching communication topologies and transmission time delay issues, it is interesting to explore NAC approach in data fusion applied on distributed statistical Decisions Theory (DT). Moreover, this framework do not require complex network protocol with extra semantic concepts which introduce protocol overhead. Therefore application to MAS running on networks with restricted communication capacity is investigated with an interest on network load costs for Wireless Sensor Network (WSN) implementation. This paper focuses on preference aggregation to allow a set of agents to select a common, consistent and unique decision in dynamic networks with constrained communication capacity in a fully distributed way. As NAC is an asymptotic approximation of consensus equilibrium, finite time convergence issue is addressed by using multi-scale adaptive algorithm which ensures likewise discrete result values. The paper is organized as follows. Section 1 presents theoretical background for the proposed algorithm. Section 2 defines implementation requirements. Section 3 presents an application example with associated results and some discussions. Conclusion and future works appear in Section 4.

Literature Review. NAC has been studied initially in DS and Control Theory (CT) e.g. Synchronization of Coupled Oscillators, Flocking Theory, Fast Consensus, Rendez-vous in Space or Distributed Formation Control around issues concerning continuous versus discrete time consensus and undirected or directed graphs [5] and references herein. Several contributions have demonstrated the insensitivity of NAC convergence according to given conditions for Time Delay, Switching Topologies [6,8] and Network costs in real applications with localized in time communication failures [4,7]. After robustness study, convergence time has been studied [3] according to network topology (regular lattive, random network, smallworld [5] and free-scale network [10]) to define a performance indicator λ_2 called algebraic connectivity. Finally, other update schemes are currently studied for finite time convergence [1] and the definition of general function operators framework [2]. This brief outline is not comprehensive regarding the amount of work on NAC, but illustrates its possibility for new kind of distributed algorithms in MAS, especially for Voting Procedures (VP).

1 Theoretical Background

This paper proposes a way to perform Voting Procedures (VP) in Multi Agent System (MAS) by a distributed approach. Each node has its own preference order which must be aggregated with those of the other nodes to represent the whole network preference order. Preferential model uses utility function to model numerically the preference order for each node. Each of these utility functions are aggregated by executing Multi-Network Average Consensus (NAC) on them to build up an unique aggregated utility function on which decision rules are applied. Present work proposes a distributed algorithm which guarantees convergence to an unique preference order produced by the consensus. This allows nodes to take the same decision at the end of the algorithm without using extra mechanisms to ensure consistency and uniformity of node decisions. Network agents are distributed and constrained by their communication capacity such as in Wireless Sensor Network (WSN) applications. Their ability to communicate with agent community is limited to their neighbours which excludes full broadcast exchanges producing network overload.

1.1 Preference Model

The agent community is composed of N nodes which must select a common profile among a set $\rho_0, \rho_1 ... \rho_M$ of size M without centralization of all node preferences. Preference notation is defined such as A is preferred to B by the node i : $A \underset{i}{\succ} B$. It is assumed that each node can define a partial order of their profile preferences which are modelled by a strict monotonic discrete utility function denoted $u^i(t)$ where $u^i_j(t)$ is the utility value of profile ρ_j for the node i at the time t and $\dot{u}^i_j = \lim_{t \mapsto +\infty} u^i_j(t)$. A profile preference partial order is representative of all node i if, it is established for all of them that in Eq. 1.

$$\exists j / \ \forall k, \rho_j \succ \rho_k \Longleftrightarrow \exists j / \ \forall i, k, \rho_j \underset{i}{\succ} \rho_k \Longleftrightarrow \exists j / \forall i, k, \dot{u}^i_j > \dot{u}^i_k \tag{1}$$

1.2 Distributed Aggregation Process

Based on the discrete utility function $u^i(0)$ of each node i, aggregation process must build up an aggregated utility function $\dot{u}$ representative of the whole network. The utility function is discrete, so aggregation process can be executed on each utility value independently. So, global aggregation process is composed of a set of elementary aggregation by using NAC algorithm [9]. In Eq. 2, $(\dot{u}_0, ..., \dot{u}_M)$ represents the result values of consensus on each profile j which has its final value $\dot{u}_j$ equal to the means of the utilities $u^i_j(0)$ weighted by their relevant node importance w^i. So, as NAC algorithm is a decentralized algorithm, the global aggregation process is also decentralized because it is composed of M NAC executed simultaneously. NAC algorithm has an asymptotic convergence to the initial value of node's utility by using gradient-descent algorithm with the error approximation ε.

$$\dot{u} = (\dot{u}_0, ..., \dot{u}_M) = \frac{1}{N} \sum w^i u^i_j(0) \Longrightarrow \forall i, j |\dot{u}_j - \dot{u}^i_j| < \varepsilon \tag{2}$$

The analytic form of iterative algorithm NAC is remembered in Eq. 3 under the necessary condition of convergence given in [5] : $\forall i, w^i < \frac{1}{\Delta}$; $\Delta = max(\#\Upsilon_i)$ with $\#\Upsilon_i$ is the degree of vertex i and Δ the graph's degree.

$$u^i_j(t+1) = u^i_j(t) - w^i \sum_{k=0}^{N} (u^i_j(t) - u^k_j(t)) \tag{3}$$

By applying NAC on each profile, each node obtains the same profile utility function according to error interval $]-\varepsilon + \dot{u}; \dot{u} + \varepsilon[$ produced by asymptotic convergence. Unfortunately, this error interval is unknown and cannot be computed analytically without full knowledge of the problem data. This theory limitation is bypassed, by using an upper bound function as discussed in Section 2.1.

1.3 Consistent Decision Rules

After execution of the aggregation process, nodes must choose the best profile. According to the convergence error, some case cannot be discriminated enough. Consequently, Th. 1 gives conditions under which interval between utility values is large enough to choose without ambiguity. Else, Def. 1 proposes decision-making which ensures its unicity for each node.

Theorem 1. *If and only if there exists a profile utility function u^i_j greater than any other profile utility function u^i_k over 4ε for any decision makers i estimated by network average consensus, then this profile is preferred to any other profile by all decision makers.*

$$\forall i, k, \exists j / u^i_j > u^i_k \text{ and } |u^i_j - u^i_k| > 4\varepsilon \Longleftrightarrow \rho = \rho^i = j \tag{4}$$

Proof. Network average consensus converge to average value $\dot{u}$ of node initial value $u(0)$ for any network topology if convergence rate of gradient-descent algorithm is limited to $\frac{1}{\Delta}$ with $\Delta = \max(\#\Upsilon_i)$ [5]. Average value $\dot{u}$ is unique by definition, so the set of utility order $u_0^i(0) > ... > u_M^i(0)$ of each nodes i will evolve to an unique average utility order $\dot{u}_k > ... > \dot{u}_j,\ k,j \in [0,\ M]$ according to convergence error ε which defines an error interval $]-\varepsilon+\dot{u};\dot{u}+\varepsilon[$. Assume that there exists a node which prefers a profile j different from the other node preferences. This can happen if and only if an interval error of this preference utility is juxtaposed to another interval error of another near preference utility $:]-\varepsilon+\dot{u}_j,\dot{u}_j+\varepsilon[\ \bigcap\ ]-\varepsilon+\dot{u}_k,\dot{u}_k+\varepsilon[\neq\emptyset$ with $\dot{u}_j^i\in]-\varepsilon+\dot{u}_j,\dot{u}_j+\varepsilon[$ and $\dot{u}_k^i\in]-\varepsilon+\dot{u}_k,\dot{u}_k+\varepsilon[$. Now, if $|\dot{u}_j^i-\dot{u}_k^i|>4\varepsilon$, it is obvious by using triangle inequality that $]-\varepsilon+\dot{u}_j,\dot{u}_j+\varepsilon[\ \bigcap\ ]-\varepsilon+\dot{u}_k,\dot{u}_k+\varepsilon[=\emptyset$. Thus, if a node has other preference than the other nodes, its preference utility cannot be spaced from its other preference utility by more than 4ε.

Definition 1. *Two profiles ρ_i and ρ_j are defined as equivalent if the distance between their estimated aggregated utility value is inferior to 4ε.*

$$\rho_j \sim \rho_k \Longleftrightarrow |u_j^i - u_k^i| < 4\varepsilon \tag{5}$$

Based on Th. 1 and Def. 1, a uniqueness profile decision rule can be defined such as in Eq. 6. Th. 1 guarantees that it is possible to build a unique aggregated order of preferences which is representative of all node's profile preference where it exists an enough discriminant interval. Def. 1 defined equivalence state if it is a partial order. Finally, space U^i is the set of equivalent best aggregated profile preference utility on which is applied an arbitrary rule to select one unique and common profile ρ

$$\rho = \begin{cases} U^i \in \mathbb{N}\ /\ j,k \in U^i \text{ if } \forall l,i,\ \rho_j \underset{i}{\sim} \rho_k \underset{i}{\succ} \rho_l \\ \rho_i = \min(U^i) \end{cases} \tag{6}$$

2 Algorithm Implementation

2.1 Time Convergence Estimator

Previous theoretical background presents convergence process under infinite time. Also, it is assumed that it is possible to determine a cone distance ε around ideal average value $\dot{u}_j$ where each estimated value u_j^i is inside it. But, as each node cannot know the utility values of other nodes, it is impossible for them to know when consensus is reached. Banach fixed point theorem can give a bounded time to reach it because NAC uses gradient-descent algorithm which has a q-linear convergence and so it is a k-lipschitzien function such as defined in Eq. 7.

$$|u_j^i(t+1) - \dot{u}_j| \leq k\ |u_j^i(t) - \dot{u}_j|\ ,\ k \in [0,1] \Longrightarrow \varepsilon \leq \frac{k^n}{(1-k)}[\max(u_j^i) - \min(u_j^i)] \tag{7}$$

Unfortunately and to the best of our knowledge, there is no analytical method to determine value k without knowing about the network topology (like algebraic connectivity λ_2) and initial value of nodes. Indeed, based on initial value, the graph laplacian matrix and the algebraic value λ_2, it is possible to determine an upper bounded value of required iterations [1] : $\frac{||Lu_j(0)||_2}{\lambda_2}$, when L is graph Laplacian.

2.2 Multi-Scale Adaptive Accuracy

Implementation is limited by hardware accuracy and finite time requirement. In this case, proposed algorithm must refine its equivalence Def. 1 between two utility values and its stop condition for the processing. As ε determines maximum error interval for a given utility value for all nodes, after enough iterations their equivalence according to ε and encoding base q is defined by Eq. 8.

$$\forall i, \exists j, k \ / \ \lceil \frac{(u_j^i - u_k^i)}{(q+1)\ \varepsilon} \rfloor = 0 \Longrightarrow \rho_j \sim \rho_k \tag{8}$$

The equivalence in Eq. 8 is not true, indeed two profiles can be equivalent without equality of their utility values according to ε because of a lack of significant digits. To increase accuracy of estimated utility value, the refining process is executed NAC with a decreasing error interval $\varepsilon = \frac{\varepsilon}{q}$. Refining process must continue until it is possible to guarantee that utility values are discriminant enough for allowing each node to extract the same U^i space defined in Eq. 6. As node decisions must be consistent by defining the common U space, process of estimation refining must stop if each node has its set U distant enough from the other utility values according to ε error interval and encoding limitations as defined in Eq. 9.

$$\forall i, \exists j, k \ / \ u_j^i \in U^i, u_k^i \notin U^i, |u_j^i - u_k^i| > q\varepsilon + \varepsilon \Longleftrightarrow \forall i, U = U^i \tag{9}$$

2.3 Voting Procedure Algorithm

Algorithm is composed of two main steps: the aggregation of utility values and the selection of the best profile. During aggregation step, nodes communicate together by broadcasting their current utility value $u^i(t)$ to their neighbours. This step is called with a decreasing ε until aggregated utility values are spaced enough according to convergence error criterion to discriminate utility values.

Algorithm 1: Algorithm executed on each node i.

```
Data: u^i(0), w^i
Result: ρ
begin
    ε ← 1;
    repeat
        ε ← ε * 0.1;
        repeat
            foreach j=1...M do
                u_j^i(t+1) ← u_j^i(t) − w^i Σ_{k=0}^{N} [u_j^i(t) − u_j^k(t)]
        until (k^n/(1−k)) [max(u_j^i) − min(u_j^i)] < ε/2;
        [value, k] ← max(u)
        DONE ← true;
        foreach j=1...M do
            if |value − u_j| < (q+1) ε and ⌈(value−u_j)/((q+1) ε)⌋ ≠ 0 then
                DONE ← false;
    until ! DONE;
    [value, ρ] = max(u);
    foreach j=1...M do
        if ⌈(value−u_j)/((q+1) ε)⌋ = 0 then
            ρ ← min(ρ, j)
```

3 Experimental Application

Application considered here is a set of WSN agents which have to select a common profile ρ together. Random network topology of degree 5 is considered in the proposed experimentation in which 25 nodes should select the best profile among a list of 10 profiles with their utility value defined such as $\rho_0 \succ_i \rho_1 \iff u_0^i(0) = u_1^i(0) + 1$. But, this proposal is absolutely not a requirement such as it can be modelled by other functions which must be monotonic, discrete and bounded function. Resulting utility functions are reported on Figure 1 for each node at time 0. As illustrated on Figure 2, convergence time of the proposed algorithm is 50 iterations composed of three Multi Network Average Consensus (MNAC) loop. Figure 3 shows the adaptive error criterion such as common space U is not distant enough from other utility values at first loops to ensure consistency of node decisions. Finally, utility functions of each node on Figure 4 are equal according to last ε value.

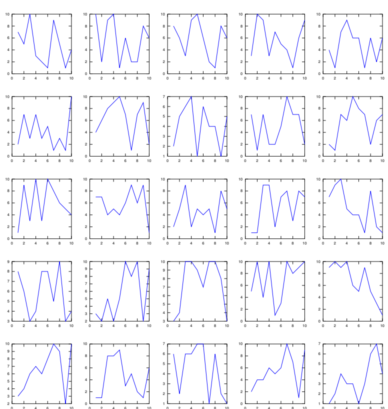

Fig. 1. Node's utility functions $u^i(0)$.

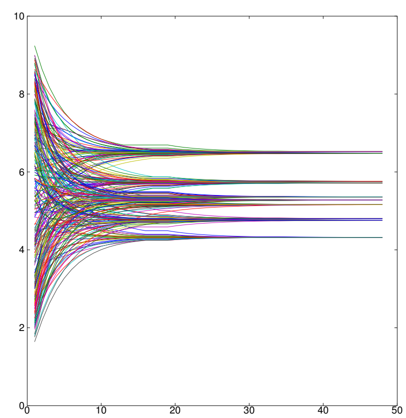

Fig. 2. Convergence to utility $\dot{u}$.

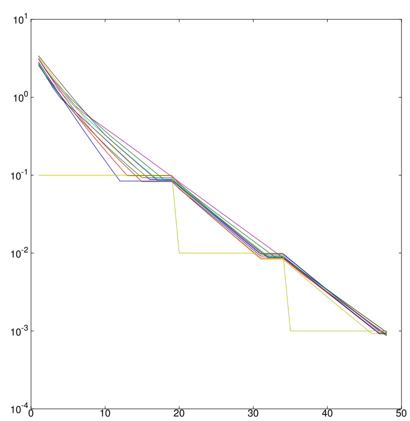

Fig. 3. Mean estimated error ε.

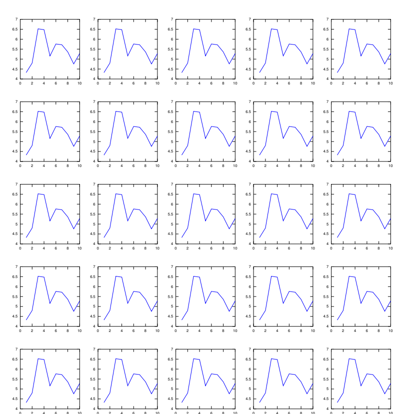

Fig. 4. Final utility function $\dot{u}$.

3.1 Byzantine Threats and Voting Veto Discussions

Proposed algorithm is interesting by its properties inherited from Network Average Consensus (NAC) for Voting Procedures (VP). Indeed, VP can require veto possibilities concerning some profile alternatives for application reasons. NAC algorithm solves alignment problem when a single agent (called the leader) keeps its value unchanged. In this case, all other agents will asymptotically agree with it [5]. This alignment property allows agent to use their veto on a given profile by keeping its value unchanged. If an agent cannot allow a profile to be selected by the community, this agent should bypass update scheme of the concerned NAC and use instead the minimal utility value. Unfortunately, this property of alignment can be undesirable in case of Byzantine threats. Byzantine threats should be detectable by comparing utility value evolution from neighbour agents which should have a q-linear convergence as previously presented. In this case, nodes have just to exclude this node from their NAC update scheme to eliminate the threat.

4 Conclusion

This paper proposes a new distributed algorithm for Voting Procedures (CP) in Multi Agent System (MAS). It is composed of a distributed preferences aggregation step with adaptive refining according to requirements of selection step. This latter is realized by each agent individually selecting the profile preferred by the agent community according to a common decision function. As consistency of the result is ensured at the aggregation step, the selection step do not require any mechanism to check if all agents have taken the same decision.

Alignment property of Network Average Consensus (NAC) allows the chosen preference model to be extended by adding Veto possibility. A veto is a node's behavior which allows a profile to be excluded of the Voting Procedures (VP). If any node keeps the utility value of a profile unchanged and low, the community will converge asymptotically to an agreement of exclusion. But, Byzantine agent can hit back by using this property to change the profile preferred by the community. In all case, nodes decisions stay consistent with or without Byzantine agent in the community. Moreover, the proposed Voting Procedures (VP) algorithm is interesting by its good properties inherited from Network Average Consensus (NAC) research concerning mobile network constraints such as robustness in switching topology and large scale network but also time delay. This algorithm is simply based on an utility function of preferences for each node and do not require any extra information about network topology than the list of its neighbours. As this algorithm converges asymptotically and exponentially as observed on Figure 2, it requires several iterations and exchanges over the network. But these exchanges remain localized in local neighbours area, which allows network load balancing. As these exchanges have a very small payload with a null overhead (it just contains utility vector of the node), this algorithm is very interesting in Wireless Sensor Network (WSN) applications which are extremely limited by their communication capacity.

However, aggregation operator studied in this paper is the arithmetic mean for min or max operator decision function which can limit application case. Future works will be interested in studying this algorithm on general functions for reaching consensus during aggregation step. J.Cortes [2] proposes several new NAC update schemes which will be explored such as the minimum and maximum operator, the harmonic mean, the geometric mean, the arithmetic mean and the root mean square. A study of Ordered Weighted Average operator as a generic decision function will also extend algorithm genericity.

Acknowledgements. The authors thanks M. Zitouni and M. Courbin for discussions during the development of this works.

References

1. Cortés, J.: Finite-time convergent gradient flows with applications to network consensus. Automatica **42**(11) (2006) 19932000
2. Cortés, J.: Distributed algorithms for reaching consensus on general functions. Automatica **44**(3) (2008) 726–737
3. Hatano, Y., Mesbahi, M.: Agreement over random networks. Automatic Control, IEEE Transactions on **50**(11) (2005) 1867–1872
4. Kar, S., Moura, J.: Distributed consensus algorithms in sensor networks with imperfect communication: Link failures and channel noise. IEEE Transactions on Signal Processing **57**(1) (Jan 2009) 355–369
5. Olfati-Saber, R., Fax, J.A., Murray, R.M.: Consensus and cooperation in networked multi-agent systems. Proceedings of the IEEE **95**(1) (January 2007) 215233
6. Olfati-Saber, R., Murray, R.M.: Consensus problems in networks of agents with switching topology and time-delays. Automatic Control, IEEE Transactions on **49**(9) (2004) 1520–1533
7. Patterson, S., Bamieh, B., El Abbadi, A.: Distributed average consensus with stochastic communication failures. In: Decision and Control, 2007 46th IEEE Conference on, New Orleans, Louisiana, USA (Dec 2007) 42154220
8. Ren, W., Beard, R.W.: Consensus of information under dynamically changing interaction topologies. In: American Control Conference, 2004. Proceedings of the 2004. Volume 6., Boston, USA, IEEE (June 2004) 4939–4944
9. Ren, W., Beard, R.W., Atkins, E.M.: A survey of consensus problems in multi-agent coordination. In: American Control Conference, 2005. Proceedings of the 2005, Portland Oregon, USA (June 2005) 18591864
10. Wang, H., Guo, Y.: Consensus on scale-free network. In: American Control Conference, 2008, Seattle, Washington, USA, IEEE (June 2008) 748–752