Automated Argumentation Among Internet of Things: A Case Study for Driverless Vehicles

K. Rodriguez[1] **and H. Hexmoor**[1]
[1]Computer Science Department, Southern Illinois University, Carbondale, IL, United States of America

Abstract - *Internet of Things must rely on their agent counterparts (IoA) in order to achieve shared and competing objectives. We have considered the production of sets of arguments that support different objectives at the same time. This will lead to myriad forms of contradictions. We present an account of automated argumentation system that facilitate contradictions using a system of social voting. We have implemented a case study using driverless vehicles and how changing lanes can produce conflicts with contradictions among corresponding arguments.*

Keywords: Social Argumentation, Internet of Things

1. Introduction

Cyber-physical systems (CPS) are algorithmically controlled mechanisms involving networked devices [1]. Whereas physical components of CPS (e.g., robots sand devices) are tangible, embodied, and occupy physical space, cyber components are largely disembodied, intangible, and location-independent. As such, internet of things (IoT) are a subset of CPS. In sharp contrast to the passive view for entities of things as objects, agent inhabitants of IoT are active and may take action proactively. In this perspective, things are enlivened with agent overlays that take advantage of smart sensors and provide intended decision making capacities for things.

Numerous suggestions posit that things in physical proximity form social ties creating collaboration networks. Minimally, things provide profiles that include goods and services relevant to other things. We are focusing on interactions among people and things that lead to creation of trust, delegation, role arbitration, and thus collaboration. Nodes may perceive a level of social capital as experience prior and expected future beneficial interactions. Much needs to be developed to exploit the spectrum of sociality. We have used crowd evacuation as an illustrating case study and an exemplar for other scenarios.

In the burgeoning era of cyber physical systems, it is *essential* that embedded IoT nodes work together on matters of common interest and using *automated* argumentation reach agreements or at least commonly identify the strongest position on consequential topics. For instance, a driverless car must determine the best course of action when confronted with unavoidable collision [2]. Legal considerations as well as ethical resolutions will remain outside current proposed work. However, future explorations may bring them into our focus. We define machine to machine social argumentation *as negotiation that* include but goes beyond argumentation when individuals are in a socially connected network as in [3]. Settings where humans and things form collaborative teams are fascinating but remain outside our current scope. Instead, we target machine to machine argumentation as in the case of vehicle to vehicle *networks*. There have been attempts to form autonomous robotic ad hoc coalitions; e.g., [4]. Similarly, IoT nodes that monitor health status of occupants in a building must agree on the safest building location for people to congregate. This is crucial for all types of *disaster* from weather concerns to the active shooter incidents.

Argumentation is the process in which agents exchange and evaluate interacting and inevitably conflicting arguments. It is a form of dialog during which beliefs, understanding and opinions are presented, explained, compared, and defended. The arguments are the basis for inferences, negotiations, conflict resolution, and conclusions drawn by logical reasoning. Argumentation is one of the oldest research foci and one of the most enduring ones in Artificial Intelligence [5] [6] and in parallel in Philosophy, first reported in [7] and most recently in [8]. Automated Argumentation has been adapted to many domains including computational law and multi-agent negotiations [9]. The most vigorous and prolific argumentation research has been conducted with Argugrid (www.argugrid.eu), which is a grid based research consortium funded by the European Union and directed by Dr. Francesca Toni of Imperial College in London, United Kingdom. Whereas social abstract argumentation [3] facilitates online argumentation among human social media participants, commonly found on Facebook, we aim to facilitate social argumentation chiefly among machines. Numerous suggestions posit that things in physical proximity form social links creating social networks. Minimally, things provide profiles that include goods and services relevant to other things [10]. For effective interaction with human peers and other animals, things need to be equipped with biological sensors (i.e., biosensors) so that their corresponding agents would ascertain conditions of their bio-organism cohabitants. For example, graphene nano-sensors are available for passive sensing of bacteria. Other typical passive biosensor exemplars are motion and vibration sensors, thermometers, audio and visual sensors, touch and tensile sensors, barometric pressure sensors, and a variety of chemical sensors. By fusing sensory information, a thing may determine bio-organism presence including humans at given radii from it. Agents controlling things can use biosensors as proximity sensors and behave in socially meaningful ways. Once agents inhabiting things perceive bio-presence, they may perceive and initiate as well as expect reciprocal sociality. Reciprocally, humans may perceive electro-mechanical things by sensing energy and wireless networking measures.

In the context of smart IoT devices, the first task is identification of arguments generated by their corresponding agent. Each agent is designed to receive sensory data and perform problem solving that produces an output, which might be a mere perception or an action to perform.

The problem solving module shown in Figure 1 is an expert system that encapsulates agent problem solving. Agents will fuse one or more sensory data for determining an input for reasoning. The expert system will include design and a model current applicable conditions. A periodically generated argument is a pair of sensed data and an output encapsulated as an atomic abstract argument that will be cast to compete with other arguments in the system argument pool.

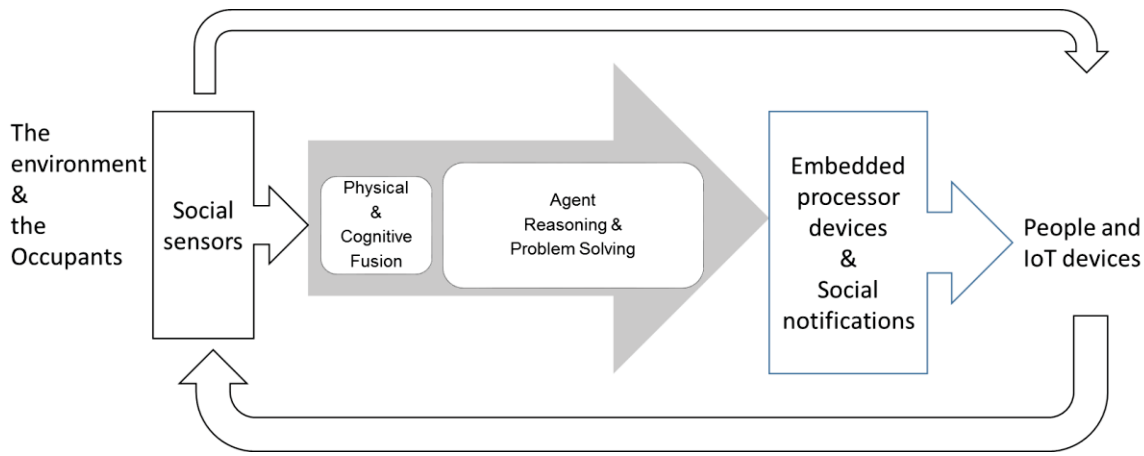

Figure 1. An agent corresponding to a smart IoT device

The bulk remainder of this paper is devoted to describing our case study for driverless vehicle lane change decisions and arguments for such that are conflicting and are resolved via social voting. We end the paper with concluding remarks.

2. An expert system for vehicular lane selection

Using a Toulmin style form of argument formation, we have developed an expert system (ES) for lane selection among smart vehicles that embody two main components: (a) an inference engine, and (b) an inter-agent argumentation resolution component.

As shown in Figure 2, the inference engines use the context taken from the environment, as well as from argument losses, and formulates all possible feasible actions. The process of determining all feasible actions within this testbed is illustrated in Algorithm 1. A vehicle may not enter a global expressway position that is out of bounds of the expressway, occupied by another vehicle, and claimed by another vehicle. In this context, a lane position being claimed by another vehicle is the result of an agent "winning" its argument for a lane position. This "win" ultimately forces the losers to update the taken position as claimed and, therefore, marking the option of occupying said position as not feasible.

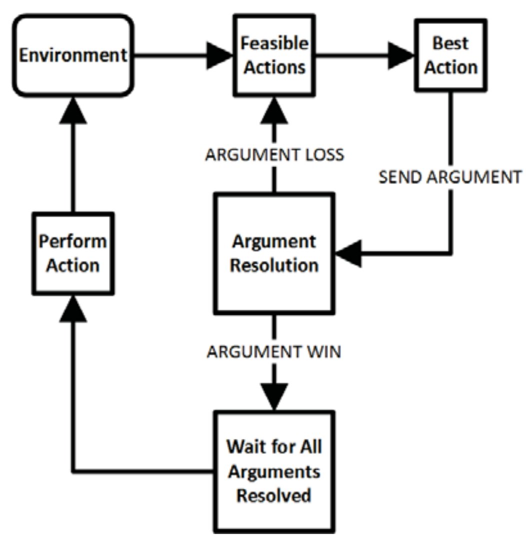

Figure 2. The Life Cycle of Argument Formation, Conflict Resolution, and Action Enactment

if *Lane Above in Bounds && Not Claimed && Not Occupied* **then**
  Can Move Up = true;
end
if *Lane Below in Bounds && Not Claimed && Not Occupied* **then**
  Can Move Down = true;
end
if *Space Ahead Not Claimed or Occupied* **then**
  Can Maintain Speed = true;
end
if *Current Speed Not Below Lane Min* **then**
  Can Decelerate = true;
end
if *Current Speed Below Lane Max && Space Ahead Not Claimed or Occupied* **then**
  Can Accelerate = true;
end

Algorithm 1. Feasible Action Derivation

Upon feasible action derivation, the best action in accordance with the ES objective is determined. The argument for this respective action encapsulates the projected position that the vehicle expects to occupy because of their chosen action and encapsulates the vehicle itself. The argument by each agent is consolidated into an argument pool where the overseeing system will identify conflicting arguments that are projecting two vehicles to enter the same global expressway position. The overseeing system will gather votes from each vehicle corresponding to their acceptance or rejection of each argument within the pool. After the social support for each argument is known, conflicts between arguments are resolved, with each argument with the most support within a conflicted arguments subset of the entire argument pool being given permission to perform their next action. Upon approval and modification of all argument actions such that there are no longer any conflicts between arguments, each agent will then perform their chosen action.

Once the inference engine has determined all feasible actions, rule sets pertaining to each of the objectives then determine the best action for the said objective. In our current testbed, these ES objectives may be in one of three priority modes: (1) global emission, (2) local lane congestion, or (3) personal travel time. The congestion-based objective is concerned with reducing the congestion of the vehicles current lane, the travel time objective is concerned with attaining and exceeding the vehicles preferred speed, and the emission objective is inclined to reduce the global emission levels caused by all vehicles on the highway. The objective that the system is currently prioritizing directly affects the formulated argument. The following three arguments exhibit possible conclusions and actions pertaining to each objective.

- Travel Time Objective: a1 = Since my lane has a max speed limit below my adjusted preferred speed and the lane above is feasible; then I want a faster lane and can move up; therefore, I should move up one lane.
- Emission Objective: a2 = Since my lane has a minimum speed limit above my emission adjusted preferred speed and the lane below is feasible; then I am comfortable with moving to a slower lane and I can move down; therefore, I should move down one lane.
- Congestion Objective: a3 = Since my lane has a high relative congestion and the lane above me is feasible, within my adjusted preferred speed range, and has a low relative congestion; then moving up a lane will benefit my local lane congestion and satisfy my adjusted preferred speed requirement; therefore, I should move up one lane.

The process that the travel time objective perscribes in order to select a best action is described in Algorithm 2. The travel time objective uses an attribute, denoted in Algorithm 2 as *speed factor*, that affects how willing the vehicle is to travel exceeding its preferred speed. If the conditions of the current lane that the vehicle is in are not in accordance with their adjusted preferred speed, then the vehicle will either choose actions that lead to moving up to a faster lane or moving down to a slower lane. The other objectives of congestion and emission both operate in a similar manner. They have factors that directly affect how willing they are to ignore their preferred speed and to prioritize their objective. To maintain brevity in this paper they are not explicitly articulated.

if Current Lane Max < Preferred-Speed + Speed Factor **then**
  if Can Accelerate **then** Best Action = Accelerate **end if**
  if Can Move Up **then** Best Action = Move Up **end if**
end if
if Current Lane Min > Preferred-Speed + Speed Factor **then**
  if Can Move Down **then** Best Action = Move Down **end if**
  if Can Decelerate **then** Best Action = Decelerate **end if**
end if
if Current Speed == Preferred-Speed + Speed Factor **then**
  if Can Maintain Speed **then** Best Action = Maintain Speed **end if**
end if

Algorithm 2. Conclusion Derivation for Travel Time Objective

Vehicles who determine a need for lane change generate corresponding arguments. Naturally, multiple vehicles attempting to change lanes into the same lane at the same time will be physically conflicting, resulting in attacks between their respective arguments. We are working on several conflict resolution strategies including a social voting mechanism known as the social abstract argumentation approach (SA), all vehicles within the section of highway under consideration "vote" over what arguments they view most align with their specific objectives. Once the best actions are determined from the inference systems of all agents, arguments are formed for each agent that correspond to the projected global configuration that their chosen action will result in the vehicle occupying.

To realize this process of argumentation among agents, a social abstract argumentation framework is developed. This framework, shown in Equation 1, consists of a tuple, F, that consists of a set of arguments, A, a set of relations between arguments, R, and a set of votes for the arguments, V_a.

Equation 1: $F = < A, R, V_a >$

For every action taken by a vehicle upon the expressway, the vehicle must state their intention to all other vehicles in the vicinity. If there is a conflict of interest between any two vehicles, this conflict is labeled as an argument and voted up by the group of surrounding agents. In order to evaluate the framework that embodies these arguments, relations, and votes, a semantic framework, S, is given in Equation 2. The semantic framework is a tuple that consists of a vote evaluation function, Ψ, a conflict resolution function, Γ, and a negation operator, ¬.

Equation 2: $S = < \Psi, \Gamma, \neg >$

The vote evaluation function, Ψ, accepts a set of votes for and against a particular argument and returns a scalar value that signifies either approval or disapproval of the voted upon argument. As seen in Equation 3, the conflict resolution function, Γ, takes as input the set of all relations between arguments and the set of votes that accompany each argument. This function reduces the set of relations to a conflict-free set where no two arguments conflict or attack one another. The conflict resolution function achieves this reduction by utilizing the social support given for each agent.

Equation 3: $R_r = \Gamma (R, V_a)$

If the argument of one agent conflicts with the argument of another agent, the agent with the highest social support will ultimately "win", meaning, that the agent with the highest social support of the two arguments involved in an attack relation will be allowed to perform the action encapsulated within the argument, whereas the "loser" will be forced to give up their current course of action and either choose a different action or choose the universally acceptable decision of decelerating.

3. An illustrative example

In order to illustrate the salient features of our argumentation procedure, we evaluate the example scenario shown in Figure 3. The scenario in Figure 3 describes a 3-lane highway with six vehicles. Of these vehicles: three prioritize local lane congestion (W, X, R), two emphasize their personal travel time (Y, Z), and one has a global emission objective (Q). The inference engines of both Vehicle X and Vehicle Z chose lane change actions to the lane position directly ahead of Vehicle Y.

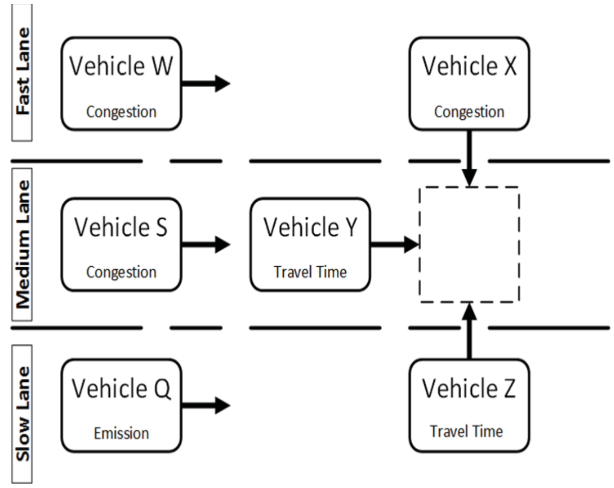

Figure 3. Social Abstract Argumentation Example Scenario

The set of all arguments, A, generated by the vehicles within the scenario, where the arguments themselves are named for their corresponding vehicle id, is found to be:

A = { W, X, Y, Z, S, Q }

These arguments in A are representative of the agent's identity and the configuration projected for these vehicles. Vehicle X currently gains the most utility from arguing for a configuration that both vehicle X and vehicle Y are at odds with. These arguments lead to the set of binary relations, R, between conflicting arguments within A.

$R = \{ (X \rightarrow Y), (X \rightarrow Z), (Z \rightarrow X), (Z \rightarrow Y), (Y \rightarrow X), (Y \rightarrow Z) \}$

All of the agents within the system determine the corresponding social support for each of the arguments. Each agent confers with their respective inference engines and determines if the argument will negatively affect the agent's next move. Congestion agents will disapprove of another vehicle entering their lane, emission agents will only approve of other agents staying or moving to the slowest lane, and travel time agents will approve any argument except those that impede their ability to accelerate, maintain speed, or move to a different lane. The voting process by these agents for a specific argument, X, is given in Table 1. There is a corresponding attack graph representing the arguments, A, their relations, R, and the votes for each. The set of votes for and against each argument, V_a, has been verified in a NetLogo implementation of this model, and was calculated to be:

$V_a = \{< W, 5, 1 >, < X, 3, 3 >, < Y, 2, 4 >, < Z, 2, 4 >, < S, 4, 2 >, < Q, 6, 0 >\}$

Table 1. Determination of Votes for Argument X

Vehicle	Vehicle Objective	Generated Vote for X	Vehicle Vote Reasoning
W	Congestion	V_+	X is moving out of my lane
X	Congestion	V_+	I am X
S	Congestion	V_-	X is entering my lane
Y	Travel Time	V_-	X will hinder my speed
Q	Emission	V_+	X is moving down a lane
Z	Travel Time	V_-	X is taking my desired lane position

The process of reducing the set of conflicting argument relations, R, into a conflict-free set, R_r, takes the form, $R_r = \Gamma$ (R, V_a). This reduced set of conflicting argument relations is seen in Figure 4.

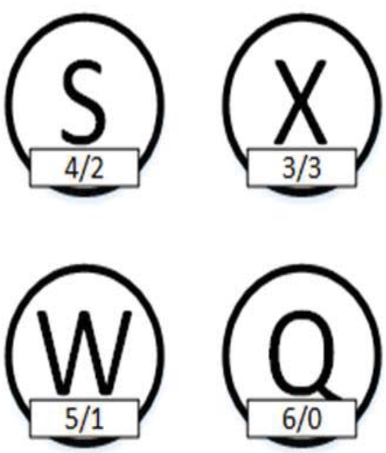

Figure 4. Social Abstract Argumentation Example Attack Graph After Reduction

$R_r = \{ (X \rightarrow Y), (X \rightarrow Z) \}$

The argument relations found within Rr represent the conflicting actions that have been approved by the agents within the group. Any agents that are the aggressor of any relation contained within the resulting set of the operation, R \ R_r, are forced to amend their choice of best action to the automatically accepted state of "decelerate." The remaining agents that did not choose to decelerate either had no conflicting argument relations or were elected as the top choice among the conflicting relations. Now that all conflicts are resolved, each agent will take their respective best action and the next round of argument formation, conflict rectification, and action enactment, will start again.

4. Conclusion

We have been addressing the need for codifying interaction among agents that represent nodes of a group of cyber-enabled systems with automated argumentation that extends abstract, Dung style argumentation. We illustrated the model of abstract arguments to components of a Toulmin style argument and explained how expert systems can be used to produce arguments. Arguments are grouped for competition by the overarching objectives used by peer agents that generate them. We have considered smart vehicles who may consider specific objectives that lead them to prefer specific lanes. This process creates conflicts between vehicles attempting lane change. Arguments that represent intentions for lane change are pitted against one another in argument conflicts.

For simplicity, we considered conflicting vehicles to share their underlying objective that govern their pattern of driving. A natural extension for further research is to consider argument conflicts that arise from agents possessing heterogeneous objectives. Our methodology heralds a step toward automated negotiation beyond automated argumentation that has been the focus of present work. Many other cyber physical environments embody stages for opposing positions that may benefit from automated argumentation as a tool for collaboration.

References

[1] G. Fortino, A. Guerrieri, W. Russo and C. Savaglio, "Integration of agent-based and Cloud Computing for the smart objects-oriented IoT," *Proceedings of the 2014 IEEE 18th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, Hsinchu, 2014, pp. 493-498.

[2] N. J. Goodall, "Can you program ethics into a self-driving car?," in *IEEE Spectrum*, vol. 53, no. 6, pp. 28-58, June 2016.

[3] J. Leite and J. Martins, 2011, "Social Abstract Argumentation," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence*, 2011.

[4] H. Yu, Z. Shen and C. Leung, "From Internet of Things to Internet of Agents," *2013 IEEE International Conference on Green Computing and Communications and IEEE Internet of Things and IEEE Cyber, Physical and Social Computing*, Beijing, 2013, pp. 1054-1057.

[5] T.J.M. Bench-Capon and P.E. Dunne, "Argumentation in artificial intelligence," *Artificial Intelligence*, vol. 171(10-15), 2007, pp. 619–641.

[6] I. Rahwan and G.R. Simari, "Argumentation in Artificial Intelligence," *Springer*, eds. 2009.

[7] S. Toulman, "Uses of Arguments," *Cambridge University Press*, 1958.

[8] J. Pollock, "Defeasible reasoning and degrees of justification, in Argument & Computation," *Taylor and Francis*, 2010.

[9] S. Fatima, M. J. Wooldridge and N. R. Jennings, "An agenda based framework for multi-issues negotiation," *Artificial Intelligence Journal*, 152(1), 2004, pp. 1–45.

[10] P. Hinds, T. Roberts, H. Jones, "Whose Job Is It Anyway? A Study of Human–Robot Interaction in a Collaborative Task," *Journal of Human-Computer Interaction*, Volume 19, pp. 151–181, Lawrence Erlbaum Associates pub. 2004.

[11] E. Lovellette and H. Hexmoor, "Abstract Argumentation Using Voronoi Diagrams," *IEEE Collaborative Systems and Technologies*.

[12] J.L. Cohon and D.H. Marks, "A Review and Evaluation of Multiobjective Programing Techniques," in *Water Resources Research*, Vol. 11, No. 2, pp. 208-220, Wiley Press. 1975.