GREEN-VANETS: Improving transportation using Car-2-X communication and multi agent systems

Adrian Groza, Bogdan Iancu, Kinga Marton, Vlad Muresan and Anca Marginean

Intelligent Systems Group
Department of Computer Science
Technical University of Cluj-Napoca, Romania
Adrian.Groza@cs.utcluj.ro

GREEN-VANETS project

Adrian Groza — Agent programming, Ontologies

Bogdan Iancu — Ad-hoc Networks

Kinga Marton — Network Security

Vlad Muresan — Real time systems

Anca Marginean — Geospatial Reasoning

Outline

1. Vehicular Ad-Hoc Networks
2. Current results
   - Emergency transportation
   - Multi-agent cooperation
   - VANET ontology

Outline

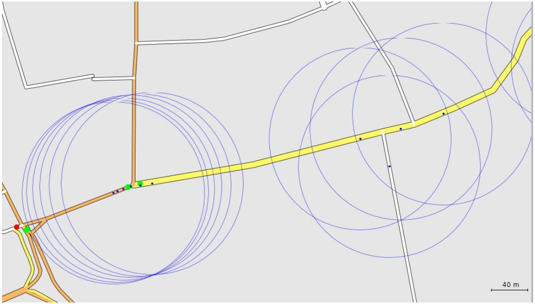
1. Vehicular Ad-Hoc Networks
2. Current results
   - Emergency transportation
   - Multi-agent cooperation
   - VANET ontology

Ambulance-2-X communication

- Aim: improve situation awareness during emergency transportation
- Technology: Ambulance-2-X communication and semantic stream reasoning
- Context: ”eCall” system by October 2015
- Paper: Towards improving situation awareness during emergency transportation through Ambulance-2-X communication and semantic stream reasoning. A. Groza , A. Marginean and B. Iancu, MEDITECH 2014, 5-7 June, Cluj-Napoca, Romania, IFMBE Proceedings Series, Springer (to appear).

Running scenario

Simulating an eCall on Ion Creanga street with ambulance approaching from the Republicii street.



Our system makes use of:

- VANET simulator
- AllegroGraph system for geospatial reasoning
- RacerPro server for semantic and temporal reasoning.

Emergency messages

When the ambulance is approaching an accident, it sends GeoBroadcast signals enhanced with intended path/route information. The messages sent by the emergency vehicle are selected accordingly to the current situation and running events:

Stop	*Ambulance vehicle crossing!*
Drive right	*Ambulance vehicle in oncoming traffic!*
Drive right	*Ambulance vehicle overtaking!*
Form corridor	*Ambulance vehicle approaching!*

Domain Knowledge

1. Vehicular ontologies
   (in-tbox emergency-vehicles)
   (define-primitive-role on-same-street
   :inverse on-same-street :domain vehicle :range vehicle)
   (implies bus vehicle)
   (implies emergency-vehicle vehicle)
   (implies ambulance emergency-vehicle)
   (disjoint bus emergency-vehicle)
   (in-abox emergency-vehicles-cluj)
   (instance a1 ambulance)
   (instance b1 bus)
   (related b1 a1 on-same-street)
2. Street topology - imported from Open Street Map

Temporal predicates in vehicular streams

Temporal predicate	Informal semantics
((move ?o) t_{start} t_{end}*)*	object ?o is known to be moving between time t_{start} and time t_{end}
((approach ?o1 ?o2) t_{start} t_{end}*)*	?$o1$ is approaching object ?$o2$ during the time interval $[t_{start}, t_{end}]$
((behind ?o1 ?o2) t_{start} t_{end}*)*	?$o1$ is behind object ?$o2$ during the time interval $[t_{start}, t_{end}]$
((beside ?o1 ?o2) t_{start} t_{end}*)*	?$o1$ is beside object ?$o2$ during the time interval $[t_{start}, t_{end}]$
((in-front-of ?o1 ?o2) t_{start} t_{end}*)*	?$o1$ is beside object ?$o2$ during the time interval $[t_{start}, t_{end}]$

11. (define-event-assertion ((move a1) 5 60))
12. (define-event-assertion ((move b1) 1 50))
13. (define-event-assertion ((approach a1 b1) 10 20))
14. (define-event-assertion ((behind a1 b1) 10 20))
15. (define-event-assertion ((beside a1 b1) 20 30))
16. (define-event-assertion ((in-front-of a1 b1) 30 60))
17. (define-event-assertion ((recede a1 b1) 30 40))

Event recognition

```
(define-event-rule ((overtake ?o1 ?o2) ?t1 ?t2)
        ((?o1 ambulance) ?t0 ?tn)
        ((?o1 ?o2 on-same-street) ?t0 ?tn)
        ((move ?o1) ?t0 ?t2)
        ((move ?o2) ?t1 ?t2)
        ((approach ?o1 ?o2) ?t1 ?t3)
        ((behind ?o1 ?o2) ?t1 ?t3)
        ((beside ?o1 ?o2) ?t3 ?t4)
        ((in-front-of ?o1 ?o2) ?t4 ?t2))
```

SPARQL query for retrieving the vehicles on the street

```
select * {GEO SUBTYPE
        <prefix:/spherical/degrees/-180.0/180.0/-90.0/90.0/5.0>
        POLYGON (RESOURCE x:w7934417 ){?a ex:location ?b.}
        Where{}}
ORDER BY <prefix:/fn/haversineKilometers>(?o,POINT(23.58,46.76))
```

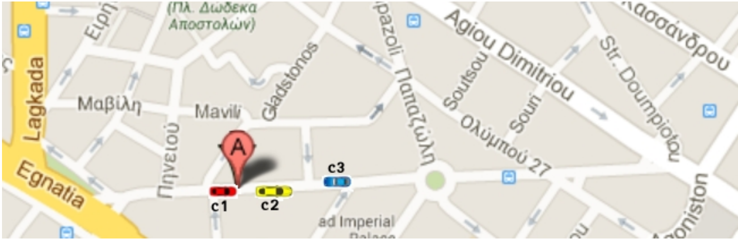
Conclusion

- We proposed a method to increase situation awareness during emergency transportation of patients.
- Our approach combines semantic reasoning with the emerging Car-2-X technology.
- We employed reasoning in description logic on top of data collected continuously from vehicular communication.
- The developed system performs temporal reasoning on real topological maps imported from OpenStreetMap.
- A step towards minimizing hazards during medical emergency services.

Multi-agent and Car-2-X

- Aim: integration of agent technology in the emerging field of vehicular networks.
- Technology: Car-2-X communication, multi-agent systems, reasoning on ontologies
- Paper: A multi-agent approach towards cooperative overtaking in vehicular networks, Adrian Groza, Bogdan Iancu, Anca Marginean, 4th International Workshop on Applications of Software Agents, Salonic, Greece, 2-4 June 2014, ACM

Vehicle overtaking scenario



Our system makes use of:

- the AllegroGraph system for geospatial reasoning
- RacerPro server for semantic and temporal reasoning.

Assertions for the overtaking scenario

```
(define-event-assertion ((hasLocation c1 l1) 5 6))
(define-event-assertion ((hasLocation c1 l2) 6 7))
(define-event-assertion ((hasLocation c1 l3) 7 8))
(define-event-assertion ((hasLocation c1 l3) 8 9))
(define-event-assertion ((hasLocation c2 l2) 5 6))
(define-event-assertion ((hasLocation c3 l4) 5 6))
(instance l1 (and (= hasLat 40.63935) (= hasLong 22.9446606)))
(instance l2 (and (= hasLat 40.63936) (= hasLong 22.9446606)))
(instance l3 (and (= hasLat 40.63937) (= hasLong 22.9446606)))
(instance l4 (and (= hasLat 40.63938) (= hasLong 22.9446607)))
(instance l5 (and (= hasLat 40.63938) (= hasLong 22.9446607)))
(equiv Lane1 (and (< hasLat 40.63939) (> hasLat 40.63930)
              (= hasLong 22.9446606)))
(equiv Lane2 (and (< hasLat 40.639390) (> hasLat 40.63930)
              (= hasLong 22.9446607)))
(instance c1 (and PassiveCooperative NormalAgent Polite))
(instance c2 (and ActiveCooperative Impolite))
(instance c3 (and ActiveCooperative Polite))
```

Types of agents

```
91. (implies PassiveAg (and Agent (some sendMsg PeriodicMsg)))
92. (implies ActiveAg (and Agent (some sendMsg EventDrivenMsg)))
93. (implies NormalAg (and Agent (some hasEvent NormalOvertaking)))
94. (implies FlyingAg (and Agent (some hasEvent FlyingOvertaking)))
95. (implies PiggyAg (and Agent (some hasEvent PiggyOvertaking)))
97. (implies Two+Ag (and Agent (some hasEvent TwoPlusOvertaking)))
98. (implies PoliteAg (and Agent
99.           (or (some hasEvent DecreasingSpeedDuringOvertaking))
100.            (some hasEvent SignalsRightBeforeOvertaking)
101.             (some hasEvent ThankMsgAfterLaneChanging))))
```

Types of data -Tbox

```
(implies Latitude PrimitiveDataElement)
(implies Longitude PrimitiveDataElement)
(implies Velocity PrimitiveDataElement)
(implies VehicleLength PrimitiveDataElement)
(implies Latitude (and (some hasValue Real)
          (all measures UnitOfMeasure)
          (some hasAcc Real)))
(implies DataFrame (and (some hasID ID)
          (some hasDescription String)
          (some hasContent PrimitiveDataElement)))
(implies PositionDataFrame (and DataFrame (equal hasID 21)
          (some hasLat Latitude) (some hasLong Longitude)))
(implies SenderDataFrame (and DataFrame (equal hasID 15)
          (some hasLength Real) (some hasWidth Real)
          (some hasModel Vehicle)))
(parent-role hasLatitude hasData)
(parent-role hasLongitude hasData)
(parent-role hasLength hasData)
(parent-role hasWidth hasData)
```

Types of data -Abox

```
(instance lat1 (and Latitude (= hasValue 40.6393) (= hasAcc .2)))
(instance long1 (and Longitude (= hasValue 22.9446)(= hasAcc .2)))
(instance p1 (and PositionDataFrame
           (= hasLatitude lat1) (= hasLongitude long1)))
(instance daciaLogan Vehicle)
(instance s1 (and SenderDataFrame
           (= hasLength 4.288) (= hasWidth 1.989)))
           (equals hasModel daciaLogan)))
```

Types of messages

```
(implies Message (and (some hasComm CommunicationType)
            (some hasTransmission TransmissionMode)
            (some hasContent Data)
            (some hasRange Integer)))
(implies Data (or (some hasContent DataFrame)
            (some hasContent PrimitiveDataElement)))
(equiv CommunicationType (or V2V V2I))
(disjoint V2V V2I)
(equiv TransmissionMode (or Periodic EventDriven))
(disjoint Periodic EventDriven)
(implies PeriodicMessage (and Message
            (some hasTransmission Periodic)
            (some hasfrequency Time)))
(implies EventDrivenMessage (and Message
            (some hasTransmission Event-Driven)
            (some isTriggeredby Event)))
(implies Accident Event)
(implies TrafficJam Event)
(implies Overtaking Event)
(ShortRangeMessage (and Message (< hasRange 1000)))
```

Lane Changing Warning Message

Definition

A vehicular message is a tuple $\langle CT, TM, Data, Range \rangle$, where CT is a shortcut for the concept *CommunicationType*, TM transmission mode, Data represents the content of the message, and Range is maximum communication range.

```
LaneChangeWarningMessage = ⟨ V2V, EventDriven
 Data[
  DataFrames [SenderDataFrame, PositionDataFrame],
  PrimitiveDataElements[Velocity, Acceleration,
                        TurnSignalStatus]
    ], 400⟩
```

VANET ontology

- Aim: engineering a Vanet ontology.
- Paper: An Ontology-Based Model for Vehicular Ad-hoc Networks, Adrian Groza, Anca Marginean, INES 2014, Tihani, Hungary, 5-7 July, IEEE (under review)

No	*Competency question*
CQ_1	Which are the vehicles on the same lane within a specific area?
CQ_2	Which data is available about the closest vehicle in front/behind?
CQ_3	Which is the closest vehicle approaching from opposite direction?
CQ_4	Which is the average speed for the next 5km?
CQ_5	Is it safe to change lane?
CQ_6	Is it safe to overtake the vehicle in front?
CQ_7	Which vehicles in the current VANET can perform multi-hop routing?
CQ_8	Are there any emergency vehicles in the nearby?

Conclusions

Aim: Integrating semantic reasoning and multi-agent technology for Car-2-X comunication

Contact:

- Project Webpage: http://cs-gw.utcluj.ro/∼adrian/projects/vanets/
- Intelligent Systems Group: http://cs-gw.utcluj.ro/∼isgroup
- Adrian Groza: http://cs-gw.utcluj.ro/∼adrian/

Thank you!