Constraint-based Data Streaming Algorithm for Human-Assisting Robotic Teams

Milan Rollo
Center of Applied Cybernetics
Czech Technical University in Prague
Karlovo nám. 13, Prague 2, 121 35, Czech Republic
Email: rollo@labe.felk.cvut.cz

Pavel Jisl, Petr Benda, Miloš Krejník
Gerstner Laboratory
Czech Technical University in Prague
Technická 2, Prague 6, 166 27, Czech Republic
Email: {jisl, bendap1}@labe.felk.cvut.cz,
krejnm1@fel.cvut.cz

***Abstract*—Teams of autonomous robots are becoming a part of the field operations like disaster relief or battlefield. Although the situation drives towards the use of completely autonomous robotic teams, humans still remain inevitable part of those teams. Number of communication protocols for data streaming in mobile ad-hoc networks was developed, but they are usually suitable for communication among autonomous robots rather than for communication with human team members. In this paper a constraint-based data streaming algorithm that provides the reliable communication among autonomous robots and humans is presented. This algorithm is based on the use of social knowledge about other team members and ensures delivery of the on-demand data streams to human team members while allowing the robots to fulfill their tasks at the same time.**

I. Introduction

Autonomous robots are becoming an important part of the field operations like a disaster relief operations, battlefield operations, or surveillance. In the past robots were usually teleoperated or only a single autonomous robot was operating in the environment. Human operator was usually required to control the robot or to assist it with task selection.

At present, as the robots became more cost-effective, easier to maintain and the communication hardware has improved, situation drives towards wider use of robotic teams and mixed human-machine teams. Although the situation drives towards the use of completely autonomic robotic teams, humans will still remain essential part of the team in some types of field operations. Those teams may consist of a high number of autonomous robots, unmanned aerial vehicles, unattended ground sensors and humans. Instead of being assisted by the humans robots proactively select their tasks accordingly to the overall goal of the mission to provide human team members with useful information or to help them to avoid risky situations. Those human-assisting robotic teams can enhance the capability of the people involved in the mission.

Use of the autonomous robots and unattended ground sensors brings new research problems: (i) missions usually take place in environments without pre-existing communication infrastructure and units are thus connected by ad-hoc communication networks and (ii) there is no central component where all the information about robots, like their positions, abilities and plans can be stored and accessed by other team members. Information among the team members thus has to be exchanged over the ad-hoc network, which has to serve a high amount of on-demand data stream requests.

Due to the nature of the environment the missions take place in (highly dynamic topology changes, radio jam, changing quality of the links and limited range and bandwidth) units may become inaccessible for communication.

Due to the limited communication range information usually has to be transmitted through several intermediate nodes. Number of data streaming protocols for mobile ad-hoc networks has been developed. These algorithms are designed to proactively or reactively look for the transmission routes in the network. However they do not ensure that the data streams will be delivered to the consumer, because the underlying communication network may become disconnected.

There are several approaches how to deal with the communication inaccessibility in a team of autonomous robots like the negotiation, commitments or stand-in agents [10]. Those approaches, however, are not suitable for interaction with human team members.

In general we can distinguish two ways the humans interact with the team of autonomous robots: (i) remote operator, which controls the activities of the autonomous robots and assigns a high level tasks to the team, and (ii) mixed human-robot teams where humans are operating in the environment simultaneously with the robots and task selection and assignment process is done automatically.

In both cases human team members need to be aware of the situation in their surrounding. This can be achieved by transmitting the information from the sensors of the autonomous robots. In the first case this information allows the operator to select best goals the robotic team has to achieve. In second case this information helps human team members to improve their own situational awareness, avoid risky areas, plan their own movement and select the goals.

To ensure an effective and safe cooperation between human team members and autonomous robots, communication protocol must ensure, that the data stream delivered from the sensors of the autonomous robots to the human user will not be interrupted.

In this paper we describe coordination mechanism for planning and controlling the communication flows in mixed human-machine teams. Proposed mechanism affects task selection process of the autonomous robots and their movement during the task execution to respect the ongoing data streams.

The paper is organized as follows: in section II related work is introduced. Section III describes the distributed mobility control mechanism and stream planning algorithm. In sections IV and V we present simulation scenario and results of the experiments. Further improvements of the algorithm are proposed in section VI. Section VII contains conclusions.

II. State of the Art

We can distinguish two basic types of the routing protocols - (i) ad-hoc routing protocols and (ii) location aware routing protocols. Ad-hoc routing protocols are based on a graph model, where the mobile host only knows the connectivity relation with its neighbors, but not the relative location of neighbors in the environment. The location aware routing protocols are based on geographical model, because the data packets are actually routed in a physical area. These protocols try to exploit location information about node itself or neighbor nodes to the routing.

A. Ad-hoc Routing Protocols

The ad-hoc routing protocols may be generally categorized as (i) *proactive* (table-driven) and (ii) *reactive* (source-initiated or demand driven) [11].

The *proactive* protocols maintain consistent, up-to-date routing information from each node to every other node in the network. Each node maintains one or more tables to store routing information and they respond to changes in a network topology by propagating updates through the network in order to maintain a consistent network view. One of the protocols, which use this approach is, Destination-Sequenced Distance Vector (DSDV) [9] protocol.

The *reactive* protocols in contrast to the proactive protocols create routes only when there is a need for sending data from a source to a destination node. The source node initiates a processes of discovering the network which is finished, when the desired route is found or all possible route permutations have been examined. Once the route is established, it is maintained by a maintenance procedure until either the destination node is inaccessible along every path from the source or the route is no longer needed. The typical protocols for this category are Ad-hoc On-demand Distance Vector (AODV) [8] - see sec. II-A1 and Dynamic Source Routing (DSR) [5].

The proactive routing protocols, which require a knowledge of topology of the entire network, are not suitable in highly dynamic environments, since the topological updates need to be broadcasted frequently throughout the network. These update packets consume a large portion of the network bandwidth, even when the network traffic is low.

The reactive, on-demand based, protocols use a route discovery procedure that generates also a large volume of a flooding control traffic. The actual data transmission is also delayed until a route is discovered. The example is DSR routing protocol, that might perform the route discovery procedure too often during a real-time communication and consumes the bandwidth, which is intended for data, to transmit route-request or route-reply packets.

1) Ad-hoc On-demand Distance Vector: The Ad-hoc On-demand Distance Vector (AODV) [8] protocol was designed as a successor of the DSDV protocol [9]. The difference is, that only nodes on active paths maintain routing information. The other nodes do not participate in any periodic routing table exchange and do not discover and maintain routes to another nodes, until they need to communicate. However, they offer its services as an intermediate forwarding station to maintain connectivity between two other nodes. Authors call it *pure on-demand route acquisition* system.

B. Location-aware routing protocols

1) Location-Aided Routing (LAR): LAR protocol [6] follows the reactive routing approach, the routes are searched on-demand, using same algorithm as AODV (Sec. II-A1). This algorithm is modified to use the localization information for reducing route discovery overhead while broadcasting the route request packets.

The algorithm utilizes two zones - expected zone and request zone. Consider a node S, that needs to find a route to node D. Node S knows, that node D was at location L at time t_0, than the *expected zone* of node D, from viewpoint of node S at time t_1, is the region that node S expects to contain node D at time t_1. Node S can determine this expected zone based on the knowledge, that node D was at location L at time t_0.

If node S does not know a previous location of node D, than node S is not able to determine the expected zone and algorithm is reduced to the modified basic flooding algorithm (see II-A1). This modification consists from the specification of the *request zone*. Node S defines a request zone for the route request. A node forwards a route request only if it belongs to the request zone. To increase the probability that the route request will reach node D, the request zone should include the expected zone. Additionally, the request zone may also include other regions around the request zone.

2) GRID: Fully location-aware routing protocol: The routing protocol GRID [7] tries to exploit location information in all the route discovery, packet relay and route maintenance. This protocol uses algorithm based on a number of logical grids, each as a square. For each grid, one node is elected as the leader of the grid. Routing is then performed in a grid-by-grid manner through the grid leaders.

The grid leader responsibilities includes: (i) forwarding route discovery requests to neighboring grids, (ii) propagation of data packets to neighboring grids and (iii) maintaining routes which pass the grid. All other nodes in the grid are not responsible for these, unless they reach destinations of (i) and (ii) and sources/destinations of (ii). For maintaining quality of routes, there is suggestion that the leader of the grid should be the one nearest to the physical center of the grid.

For the route maintenance the feature of ”handoff” procedure is used. The route is considered alive as long as there exists a leader in each grid that constitutes the route. If a leader leaves its original grid, this leader pass its routing table to the next leader (through broadcast or negotiation).

The route discovery in this protocol also reduces the route discovery overhead. Because only grid leaders are responsible for route search, the number of packets related to route search is insensitive to the density of mobile hosts in the searched area.

III. Constraint-based Data Streaming Algorithm

Existing data streaming algorithms usually take into account only the information about the current accessibility of the neighborhood nodes, because the motion of nodes is generally unpredictable in the field of ad-hoc networking.

In scenarios our algorithm is designed for we can assume, that the team members are cooperating and trying to achieve a common goal. In such a case a team members can exchange additional information, like a current position, future plans, free resources or ongoing streams, which can be used to improve efficiency and reliability of the data streaming algorithm. Such an information about other team members is usually referred to as a *social knowledge*.

In the proposed algorithm the data streaming phase is tightly coupled with the task selection and execution process - robot has to fulfill its primary tasks, but at the same time it must transmit the data streams. Algorithm takes into account not only the robot's primary tasks, but also the social knowledge and data stream requests.

Algorithm works in the following phases:

A. Task Assignment Phase

During the task assignment process robot selects set of tasks it is able to achieve and which will lead to the fulfilment of the mission goal. Those tasks can be assigned by the operator, negotiated with the other robots, planned by central or distributed planning mechanism, etc. Each of those tasks has defined following parameters:

- set of the required resources,
- location where the task must be accomplished (in some cases like a data processing the task is not tied to any location),
- expected duration of the task,
- utility which the fulfilment of the task will bring to the team.

Task assignment phase can be invoked anytime during the mission execution and the set of tasks thus can be dynamically modified.

B. Task Selection Phase

From the set of the tasks one (or subset) with the highest utility is selected in respect to the following constraint - execution of the task must not affect the current data streams. Task selection process checks parameters of each task (as defined above) against the information stored in local knowledge base. This information acts as a constraint on tasks which can be executed at the moment. Beside the set of robot's current tasks knowledge base contains following information:

- knowledge about surrounding environment (geographical map, sources of radio signal which can interfere with the transmission, dangerous areas, etc.) - robots can have this knowledge available in advance or build/update it during the mission,
- social knowledge - models of other robots/entities, their primary tasks, plans of their movement, available resources, etc.,
- knowledge about ongoing streams - bandwidth required, data source, data consumer, robots involved in data stream transmission.

Selected tasks must not violate the constraints - i.e. two tasks cannot be executed at different locations at the same time, robot has to take into account environment restrictions when planning path to the location where the task will be executed (its movement will affect the topology of the network) and ongoing data streams must not be interrupted. When the robot estimates whether the execution of task can cause stream interruption, it checks its plan of movement against the information about plan of the predecessor and successor in a data stream. During the task execution robot must stay within a transmission reach of those robots. In this way ongoing data streams and movement of the robots involved in this stream constraint my movement and thus the task selection process.

Following events can invoke the task selection phase:

- executed task was successfully accomplished or failed,
- task assignment phase modified the set of assigned tasks,
- data stream finished,
- predefined time from the last task selection elapsed (periodical invocation of the task selection phase).

C. Task Execution Phase

Information stored in the robot's knowledge base is sufficient during the task selection process, but the task execution in real-world environment can bring problems, which cannot be assumed during selection phase, like the signal reflection, obstacles preventing from the signal transmission, new sources of the signal/noise, etc. During the task execution thus robot has to continuously check that all the constraints given by ongoing streams are satisfied - especially that the predecessor and successor in the data stream are within its communication range.

Robots have to synchronize their movement to keep connectivity of the network. As the geographical position is not sufficient to determine the communication accessibility, robots have to measure the strength of the signal and signal-to-noise ratio (SNR). When robots will start getting out of the communication range, they have to pause the execution of current tasks.

If new data stream request arrives during the task execution, robot checks that the bandwidth required for this stream is available and eventually adds the stream as a new constraint.

D. Constraint Relaxation Phase

Due to the constraints situation can happen when robot has a set of tasks assigned, but none of them can be achieved. In such a case a constraint relaxation process is invoked - robot starts to negotiate with other robots in its surrounding about the relaxation of constraints that prevents it from selection of task to be executed. Robot tries to find alternative route for one or more data streams it is routing to reduce the set of constraints.

For the negotiation process a monotonic concession protocol (MCP) is used [3]. Negotiation process is fairly complex and number of interactions among robots is required to find a solution. Robots do not negotiate in peer-to-peer manner, but a number of robots can be involved (multi-party negotiation).

First the robot tries to find an alternative route for the stream which prevents the task with the higher utility to be executed. If the search for alternative route fails robot continues the negotiation with subsequent data stream. Robot sends information about the data stream to robots in its neighborhood and those robots evaluate, whether they can transmit the stream instead of the original robot (they try to add the stream as a constraint into their knowledge base and check it against the set of selected tasks). Use of the MCP protocol allows to carry out a nested negotiation - if constraints given by the data stream prevent robot from executing its tasks only partially (predecessor or successor in the data stream will get out of the communication range during the execution of the current tasks), it can start a nested negotiation with other robots to find an alternative predecessor or successor. In this way the MCP protocol allows to relax the constraints using more then one robot (completely alternative route can be found).

To cut the number of interactions during the negotiation robot first asks for help set of robots with higher probability that they will be able to route the data stream. This probability is estimated from the social model, based on the knowledge of current data streams transmitted by those robots and their tasks and plan of movement. If the alternative solution cannot be found, robot extends the set of robots in the next round of negotiation.

If the negotiation process fails, robot will not execute any task until some of the ongoing data streams is finished. During that period robot will only move in dependance on movement of neighboring robots and will serve as a communication bridge.

Negotiation process can be interrupted and restarted anytime as the robots involved in the negotiation can get out of the communication range or the conditions given on the start of the negotiation process change (data stream finishes, new data stream request arrives, etc.).

E. Additional Algorithm Features

When robot has no tasks assigned, it can serve as a mobile communication bridge to improve the connectivity of the network (as described in section VI-A). In such a way it removes constraints from other robots.

Two types of messages are distinguished in the system - (i) data messages carrying the sensor data from the data stream source to the consumer and (ii) system messages used by the constraint-based data streaming algorithm to coordinate activities of the robots. System messages have higher priority than the data messages.

IV. Simulation Scenario

To evaluate the features of the proposed algorithm and its comparison with other routing protocols, the simulation scenario similar to disaster relief operations was created.

This scenario simulates the search of unknown environment by a team of autonomous robots and transmission of video stream to a human operator. In a real-world this video stream will be transmitted in cases when a man or a hazardous object is found to allow the operator to carry out a detailed inspection of these objects (communication stream also allows to send commands back to the robot). After the object inspection operator can decide to send a human rescuers to the given location. During the video transmission it is necessary to keep the data source and operator connected. Existing routing algorithms are not suitable for this purpose which was shown during the experiments.

Area in which the robots operate is flat 2D square without any obstacles. In this area operate 10 robots, each of them has 15 target points randomly assigned at the beginning of the simulation run. Those target points represent points of interest during the search - in the real-world scenario will those points be assigned dynamically during the search process (see e.g. [1]).

Beside those 10 robots two other scenario actors are present: (i) video data source and (ii) human operator. Those actors do not move during the simulation and their position is fixed in all simulation runs (to ensure that several nodes serving as communication bridges will be required to establish a connection between them). All entities in the simulation scenario have limited communication range, which depends on their equipment and signal transmission power. We simulate the communication using the wireless network specified by the IEEE 802.11 standards.

A set of measurements was carried out as a function of the data packets transmitted from the video source to the operator. In each video stream 50 packets (representing images) at a rate of 10 frames per second were transmitted. Delay between the transmissions was changing to simulate different load of the network (from 10% to 90%). Overall time to complete the assigned tasks (visit all points of interest) and number of packets dropped was measured.

Simulation scenario is built on the top of the $\mathcal{A}$-**globe** agent platform [12]. Beside the features common to other agent platforms (like communication support, agent life-cycle management, directory services, etc.), $\mathcal{A}$-**globe** provides user with environment simulation, communication inaccessibility simulation and visualization support.

Environment simulation runs on the server side of the framework and provides clients with information about the actual state of the environment. There is a number of simulation providers and control agents running on the server container. Client containers represent the real-world entities (sensors, robots, humans).

V. Experiments

During the experiments following data streaming approaches were compared:

- AODV protocol,
- Fixed positions of robots during the transmission,
- Constraint-based Data Streaming Algorithm.

The results obtained from the first experiment illustrate the dependency of the overall mission time on the data streaming rate (ratio of the time the data stream was transmitted to the duration of the experiment), as shown in Fig. 1. Duration of the experiments is interval from the start of the experiment to the moment when all the robots complete their tasks.

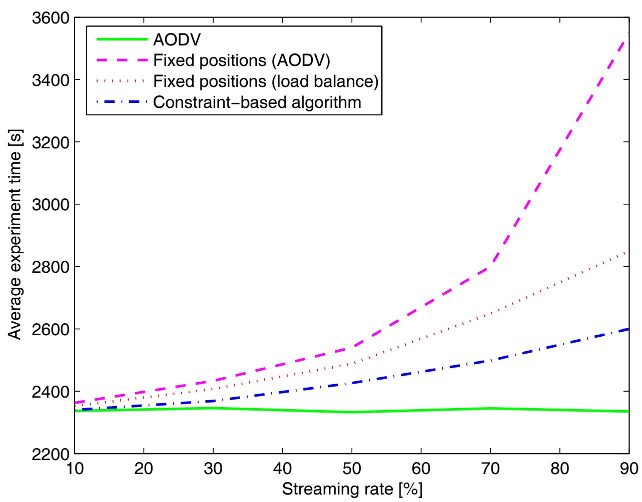

Fig. 1. Dependency of average experiment time on streaming rate

When an AODV protocol is used for data streaming, the overall duration of the mission is independent on the streaming rate. Movement of the robots is not affected by the ongoing data streams. Robots fulfill their tasks all the time and when a data stream is interrupted, protocol tries to find an alternative routing path.

In the second approach, in opposite to the previous case, movement of the robots is fully dependent on the ongoing data streams. When a robot starts to receive a data stream, it interrupts execution of all its tasks until the data stream is finished. In this case the overall mission time depends on the streaming rate and takes much longer time. Mission time is strongly affected by the way the robots participating in the route are selected. Two approaches were compared during the experiments: (i) routing tables like in the AODV are used - in this case the same robots are usually selected which prevents them from movement and extends the duration of the mission, (ii) participants are selected to balance their load, i.e. robots are sorted by the time of their last participation on data stream and the set of the least recent is selected.

When the Constraint-Based Data Streaming Algorithm duration of the mission is shorter then in previous case, but not as good as when the AODV protocol is used, because the constraints given by the ongoing streams may prevent robots from movement.

In the second experiment ratio of the packets dropped due to the interrupted communication stream was measured (ratio of undelivered data packets to all packets send by the data source). Results of the experiment are shown in Fig. 2.

In case of AODV protocol the ratio of packets dropped is constant. In case of random movements of the robots the probability that the packet will be delivered does not depend on the number of data streams (streaming rate). Note that the value depends on the communication radius (number of robots required to form the communication bridge) - the higher the range is, the lower will be the ratio of packets dropped.

When robots stay on fixed positions during the transmission, the ratio of packet dropped decreases with the streaming rate. This is because once the route for the particular stream is found, the stream will not be interrupted. Exponential character of the curve is caused by the fact, that with higher streaming rate the robots participating on the route tend to stay closer to each other - the short time that elapses between the data streams prevents move far and thus break the connections. This behavior is much more obvious when selection of routing robots is based on the AODV routing tables then on the load-balance basis.

In case of Constraint-Based Data Streaming Algorithm the robots that are less affected by the data stream stream constraints are selected. The ratio of packets dropped is slightly worse than in the case of robots on fixed positions selected by AODV routing tables, but as can be seen in the Fig. 1, the mission time is much shorter.

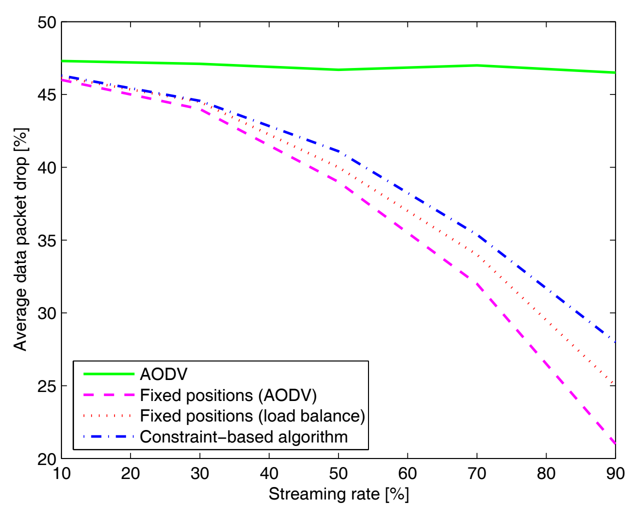

Fig. 2. Dependency of packet drop on streaming rate

VI. Further Work

Although the developed algorithm proved to improve the features of the existing data streaming algorithms, it has several insufficiencies and can be further improved.

A. Network Connectivity Improvements

Main insufficiency of the current algorithm is that it works in a reactive mode. Robots start to look for the route from the data source to the data consumer when new request from the human user arrives. It can happen, that the route doesn't exist and communication stream thus cannot be delivered to the consumer.

Such a situation can be handled in several ways:

- Robots will break their current tasks and will start search for the collaborators to build the communication bridge.
- Robots operate in proactive mode, where the task selection process is influenced by the topology of the communication network. Beside fulfilling the tasks robots try to maintain the connectivity of the network.
- Additional robots (or other unmanned vehicles like UAVs) are dedicated to act as communication bridges and will try to keep the nodes of the network connected together.

Each of those approaches has advantages and disadvantages. Main disadvantage of the first approach is that searching for other robots to create a communication bridge can take a long time, which is inadmissible in time-critical scenarios, like a battlefield operations. On the other hand the robots are not limited during the task selection process when there are no ongoing streams. When operating in proactive mode the response time for the user data stream requests is minimal. However, the requirement to keep the network connected can be very limiting, especially in cases when small number of robots operate in a large area. The third approach doesn't affect the task selection process. Disadvantage of that approach is, that additional robots must operate in the environment, thus increasing the overall cost of the mission. On the other hand those robots can only be equipped with simple sensors and not a mission-specific hardware. Instead of the robots a number of the static beacons/singal repeaters can be placed in the environment. This solution is not so effective and adaptable, but it is cheaper. The use of the UAVs to maintain the network connectivity proved to be very effective [13], but the environment in which the mission takes place can be a limiting factor. In such a case a number of the packets lost due to failed connections will decrease.

B. Data Stream Reuse and Transformation

Proposed algorithm can be improved with the capabilities of stream reusing and in-stream data transformation. It can be also used to change the stream quality when streaming to robot connected with low bandwidth connection, duplicate the streams or to set the stream priority level. In military operations the protocol must also control the access to the information by applying the policy management.

VII. Conclusions

A distributed algorithm for efficient data streaming in mixed human-robot teams is presented in this paper. The algorithm is based on the use of social knowledge about the other team members. This knowledge affects the task assignment process and forms a constraints laid on the robots during the task execution phase. Robots may relax those constraints by negotiation with other robots using the monotonic concession protocol.

It the experiments it has been shown, that the algorithm significantly reduces the number of packets dropped, while the duration mission performed the team of robots is only slightly longer than in the case of use of AODV protocol.

This algorithm helps the human team members to improve their situational awareness which leads to closer cooperation between the humans and robotics components. Further work will be driven towards the analysis of the impact of communication failures on the stability of the algorithm and coordination of the movement and activities of the robots towards the improvement of the network connectivity, which will reduce the number of packets dropped yet more.

VIII. Acknowledgement

We gratefully acknowledge the support of the presented research by the Office for Naval Research project N00014-06-1-0232 and research programme No. 1M0567 funded by the Ministry of Education of the Czech Republic.

References

[1] Wolfram Burgard, Mark Moors, Dieter Fox, Reid Simmons, and Sebastian Thrun. Collaborative multi-robot exploration. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2000.

[2] M. Scott Corson and Anthony Ephremides. A distributed routing algorithm for mobile wireless networks. *Wirel. Netw.*, 1(1):61–81, 1995.

[3] Ulle Endriss. Monotonic concession protocols for multilateral negotiation. In *AAMAS '06: Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems*, pages 392–399, New York, NY, USA, 2006. ACM Press.

[4] D. B. Johnson and D. A. Maltz. Dynamic source routing in ad hoc wireless networks. *Mobile Computing*, 1996.

[5] David B Johnson and David A Maltz. Dynamic source routing in ad hoc wireless networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer Academic Publishers, 1996.

[6] Young-Bae Ko and Nitin H. Vaidya. Location-aided routing (lar) in mobile ad hoc networks. *Wireless Networks*, 6:307–321, 2000.

[7] Wen-Hwa Liao, Jang-Ping Sheu, and Yu-Chee Tseng. GRID: A fully location-aware routing protocol for mobile ad hoc networks. *Telecommunication Systems*, 18(1-3):37–60, 2001.

[8] C. Perkins. Ad-hoc on-demand distance vector routing, 1997.

[9] Charles Perkins and Pravin Bhagwat. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, pages 234–244, 1994.

[10] M. Rehák, M. Pěchouček, J. Tožička, and D. Šišlák. Using stand-in agents in partially accessible multi-agent environment. In *Proceedings of Engineering Societies in the Agents World V, Toulouse, October 2004*, number 3451 in LNAI, pages 277–291. Springer-Verlag, Heidelberg, 2005.

[11] E. Royer and C. Toh. A review of current routing protocols for ad-hoc mobile wireless networks. *Mobile Wireless Networks. IEEE Personal Communications*, 1999.

[12] David Šišlák, Martin Rehák, Michal Pěchouček, Milan Rollo, and Dušan Pavlíček. $\mathcal{A}$-**globe**: Agent development platform with inaccessibility and mobility support. In Rainer Unland, Matthias Klusch, and Monique Calisti, editors, *Software Agent-Based Applications, Platforms and Development Kits*, pages 21–46, Berlin, 2005. Birkhauser Verlag.

[13] T.X.Brown, B. Argrow, S. Doshi, R.-G. Thekkekunnel, and D. Henkel. Ad hoc uav ground network (augnet). In *Proceedings of the AIAA 3rd "Unmanned Unlimited" Technical Conference*, Chicago, IL, USA, 2004.