# Robotic Communication

A white paper prepared for NASA by

Prof. Vincenzo Liberatore
CASE
CASE SCHOOL OF ENGINEERING

August 21, 2004

## *Executive Summary*

Lunar and Mars exploration will critically depend on heterogeneous and reconfigurable robotic teams to perform complex real-time tasks. Robotic teams necessitate innovative networks technology to enable robotic communication and cooperation. Robotic networks must support real-time operations and must guarantee safety, reliability, stability, and effectiveness. As a consequence, robotic networks will depend on distributed software layers that are tolerant to signal losses and adaptive to communication delays and jitter.

This white paper summarizes our current capabilities in the area of robotic networks. Our algorithms render robotic networks tolerant and adaptive to communication vagaries and support an unprecedented degree of control of networked physical environments in spite of fast physical dynamics, exogenous disturbances, and network disruptions. Furthermore, algorithms can be implemented in a modular platform that is re-usable across disparate applications, and consequently appropriate within a system-of-systems approach, sustainable, and affordable. Our approach constitutes the fundamental end-to-end building block to robotics networks much like TCP is the backdrop of terrestrial bulk transfers. Network modules support distributed robotic software that is developed according to a sophisticated software engineering methodology involving the extensive use of middleware and of component mobility. The innovative component of this project is that, although much research has been devoted to tolerance and adaptability in real-time applications such as Voice-over-IP or streaming video, little work has been dedicated to achieving and maintaining desirable systems properties in robotic networks. Our capabilities are currently at TRL 2-3 and have been developed primarily with support from NSF and also through interaction with NASA GRC and industry.

## *An Example*

Human exploration of Moon's surface will be supported by a Wireless Surface Local Area Network (WSLAN) consisting of small solar-powered communication towers with simple switched patch antennas and access point transceivers, antennas, and access points on roving and fixed habitats. Each WSLAN subsystem routes data, voice, video, sensor streams, and control signals to and from space-suited humans, robots, sensors, rovers, and habitats. A WSLAN enables robots to coordinate their activities (e.g., assemble a power system) using video, sensor streams, control signals, and the exchange of software components. The WSLAN is the outpost of a space communication infrastructure that makes it possible to tele-supervise autonomous robots from Earth, Moon orbit, or a Crew Exploration Vehicle (CEV).

## *TRL Assessment*

The ultimate objective of communication networks is to enable the sustainable, affordable, and flexible exploration of the solar system. However, current space communication systems are monolithic, vertically integrated, and mission-specific.

Several projects are currently striving for a more flexible, affordable, and sustainable technology, including:

- NASA centers and Cisco have supported the development of space-ground Internet Protocol that has evolved to mid-TRLs and that is flexible, affordable, and sustainable (e.g., [7, 8, 9, 10, 16]), but not targeted toward the reliable real-time communication between sensors, controllers, and actuation units.
- Full-fledged HTTP/TCP/IP network stacks have been embedded in devices as small as a 5×3mm with a code footprint of around 1KB or less (e.g., [13, 14]) and require as little as 20μW (e.g., [15]) in a laboratory environment.
- Sensor networks have been recently developed and are currently at TRL's of 3-4 for terrestrial applications (e.g., [11]). Sensor networks support telemetry over packet networks, but do not address the need of real-time delivery of control signals from controllers to actuators.
- Networks of sensors and actuators are deployed in commercial products for manufacturing automation and vehicle control systems (e.g., [12]), but do not scale to the computational, radiation-tolerance, communication, and power requirements of space exploration.
- We have developed distributed layers for that are tolerant to signal losses and adaptive to communication delays and jitter [5, 6]. We estimate that this technology is at TRL 2.
- We have used mobile software agents and middleware for developing complex distributed real-time robotic software [1, 2, 3]. We estimate that this technology is at TRL 3.

## *Summary of Approach*

**Objectives.** The communication architecture should be generic and flexible so as to become affordable and sustainable. The technical approach should integrate communications and robotic systems for an end-to-end evaluation of the design trade-offs and to achieve interoperability. The resulting system is complex, demands advanced levels of collaboration and interactivity, and should lead to the establishments of standards and practices. The technology should be continuously validated on a test bed that demonstrates the feasibility of the approach.

**Capabilities.** Adaptable algorithms are aware of both network dynamics and the underlying physics, and affordable solutions can be implemented through middleware support. A facet of adaptability is a novel rate or congestion control algorithm that determines the rate and timing of data injection into a network. Rate control depends on the target physical dynamics but is also generic and flexible. The approach uses the Internet protocol as an integrated communication substratum that supports flexible, sustainable, and affordable human-robotic missions. In turn, the approach enables complex control and robotic communication that exploit extensively middleware and software mobility.

## *Approach Steps*

### A. Robotic Communication Networks

A flexible communication suite should link sensing, actuation, and control units but it should also be independent of specific tasks or missions. For example, a communication protocols could be used to enable the communication among nearby robots, and the same protocol should be used to connect sensors and actuators within one spacecraft. A common platform would be flexible, affordable, and sustainable. An instantiation of such architecture is the Internet Protocol (IP) suite, which is a programmable and manageable communication substrate to which new applications and software can be seamlessly added. The Internet forms a common architecture that leads to shared interfaces and re-usable systems. However, the Internet was originally designed to support bulk data transfer and remote log-in applications. Although its applicability has since extended to new domains and applications, there is relatively less work that addresses the Internet-enabled communication among sensors and actuators. For example, a networked robot should exhibit real-time properties, such as stability and tracking, in spite of communication vagaries (see [1, 2, 3, 4, 5, 6] for examples in this direction). Robotic networks poses special research, integration, and development challenges, which will described throughout this white paper.

> **Observation 1.** Internet protocols form an integrated communication substratum that supports flexible, sustainable, and affordable human-robotic missions, but they must be integrated with the communication requirements of sensing, actuation, and control units, as described below.

### B. Systems Complexity

Robotic control software is complex and constantly evolving. A networked control system is the intricate composition of subsystems that collectively address the needs of sensing, actuation, communication, and computing. Complex control and robotic communication requires:

- Flexibility and interoperability to support different applications, protocols, and communication needs,
- The ability to coordinate multiple units and to aggregate robot teams into controllable units,
- Control evolvability, in terms of
  - Rapid re-programmability (addition of new functionality after hardware deployment),
  - Dynamic reconfiguration (creation of new collections of sensors, actuators, computers, robots, vehicles, and instruments into coordinated, task-oriented teams), and
  - Extensibility (growth through modular incorporation of additional assets),
- Adaptation to computing needs and resources,
- Survivability and fault-tolerance (automatic reallocation of communications software in response to component failures).

A correct architecture will also ease the maturation of innovative technical contributions.

We have supported the requirements of complex applications with advanced middleware (e.g., for resource discovery that enables modular growth). Furthermore, mobile software can support survivability and rapid re-programmability by allowing software component to stop their execution on one host and resume seamlessly on a different host [2].

> **Step 1.** Complex control and robotic communication is supported by the extensive use of middleware and software mobility.

## C. Communication Network: Middleware for Adaptable Control

Networked control poses the unique challenge that a feedback loop is closed through a communication infrastructure, whose real-time characteristics affect the performance and stability of the connected physical systems. As a result, the communication infrastructure must adapt to the network vagaries inherent in space-communication. Communication adaptability must account for network behaviors as well as for the physical dynamics of the embedded units, and in particular it must exploit continuous or hybrid descriptions of the target physics. At the same time, communication adaptability must be flexible and sustainable, and in particular it must be applicable across a range of robotics or actuation units, and across missions. We have developed flexible adaptability strategies [5], which can be incorporated in a lightweight middleware layer that would be accessible to distributed real-time control applications.

> **Step 2.** Control and robotic communication must be adaptable to long and unpredictable delays. Adaptable algorithms exploit their awareness of network and physical dynamics.

## D. Communication Network: Rate Control

A fundamental issue in networked systems is the rate at which networked units inject data into the network. Rate control is a broad issue that affects the traffic, performance, quality of service, and reliability of a network. Rate control has been extensively investigated in terrestrial networks, recently under the name of congestion control, for example. However, the state-of-the-art has focused mostly on bulk data transfers and streaming media, but little work has been done on networked control, where injection rates should be a function of the underlying physics and should strive for appropriate task or control-theoretical objectives. At the same time, rate control must be as flexible and generic as those implemented in current transport layers, and interoperable with them. We have devised distributed and asynchronous algorithms for congestion control based on fundamental optimization principles.

> **Step 4.** Control and robotic communication necessitates rate or congestion control algorithms to determine the rate and timing of data injection into a network. Our rate control algorithms depend on the target physical dynamics but are also generic, flexible, and distributed.

## *E. Test Bed*

The communication architecture should be generic and flexible so as to become affordable and sustainable. However, it should also be grounded in a representative test bed. The evaluation process should be introduced as early as possible in the design of the communication infrastructure and pursued continuously throughout a project. The evaluation involves methods ranging from analytical characterization, simulated linear physical dynamics on network simulators, and simulated physics on wide-area terrestrial networks. The next steps will involve simulated or emulated space networks and simulated force-controlled robot during representative tasks (e.g., manipulation for assembly). The test bed elements should include devices whose computational, communication, and power constraints are comparable to those of future radiation-tolerant platforms.

> **Planned Step 1.** Network protocols and algorithms should be tested as early as possible on representative test beds of networked sensing and actuation units.

## *Conclusions*

This white paper has summarized our approach and our low-TRL capabilities for the communication of sensing, actuation, and control units. When our approach matures, it will enable the development of sustainable, flexible, high performance robotic communication networks, which are a critical mission element for human-robotic space exploration.

## *Acknowledgments*

We would like to acknowledge the help of the following individuals for useful suggestions with the preparation of this white paper: Mark Allman (ICSI Center for Internet Research, University of California, Berekeley), Kul B. Bhasin (NASA GRC), Irene S. Bibyk (NASA GRC), Wyatt S. Newman (Case Western Reserve University), Stephen M. Phillips (Arizona State University), Brian P. Robinson (ABB), and Shivakumar Sastry (University of Akron).

## *References*

[1] V. Liberatore, W. S. Newman, and K. Bhasin. IP communication and Distributed Agents for Unmanned Autonomous Vehicles. *AIAA-UAV*, 2003.

[2] A. Al-Hammouri, A. Covitch, M. Kose, D. Rosas, W. S. Newman, and V. Liberatore. Compliant Control and Software Agents for Internet Robotics. *Eighth IEEE International Workshop on Object-oriented Real-time Dependable Systems (WORDS 2003)*, 280--287.

[3] M. L. Ngai, V. Liberatore, and W. S. Newman. An Experiment in Remote Robotics. *2002 IEEE International Conference on Robotics and Automation (ICRA 2002)*, 2190--2195.

[4] http://home.cwru.edu/~vxl11/NetBots/

[5] M. S. Branicky, V. Liberatore, and S. Phillips. Networked Control System Co-Simulation for Co-Design. *2003 American Control Conference*.

[6] W. Zhang, M.S. Branicky, and S.M. Phillips. Stability of networked control systems. *IEEE Control Systems Magazine*, **21**(1):84-99, February 2001.

[7] S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss. Delay-Tolerant Networking: An Approach to Interplanetary Internet. *IEEE Communications Magazine*, June 2003.
[8] J. Ishac and M. Allman. On the Performance of TCP Spoofing in Satellite Networks. *IEEE Milcom*, October 2001.
[9] R. Krishnan, J. Sterbenz, W. Eddy, C. Partridge, and M. Allman. Explicit Transport Error Notification (ETEN) for Error-Prone Wireless and Satellite Networks. To appear in *Computer Networks*.
[10] http://www.scps.org/
[11] D. Culler, D. Estrin, and M. Srivastava. Overview of Sensor Networks. *IEEE Computer*, **37**(8):41-49, August 2004.
[12] B. P. Robinson and V. Liberatore. On the Impact of Bursty Cross-Traffic on Distributed Real-Time Process Control. *WFCS 2004*.
[13] http://www-ccs.cs.umass.edu/~shri/iPic.html
[14] http://d116.com/ace/
[15] http://d116.com/spud/index.html
[16] http://ipinspace.gsfc.nasa.gov/

## Author

Prof. *Vincenzo Liberatore* is the Schroeder Assistant Professor in Computer Engineering and Networking in the Electrical Engineering and Computer Science Department at Case Western Reserve University. He holds a Laurea degree in Electrical Engineering from the University of Rome “La Sapienza” and a Ph.D. in Computer Science from Rutgers University. His previous appointments include a visiting position at Bell Labs and a research associate position at the University of Maryland, College Park. He has extensive publications in networking, theoretical computer science, compilers, and security. He has been a principal investigator for projects in the area of networked control (http://home.cwru.edu/~vxl11/NetBots) and has been supported by the National Science Foundation and by NASA. He can be reached at `vincenzo.liberatore@case.edu`.