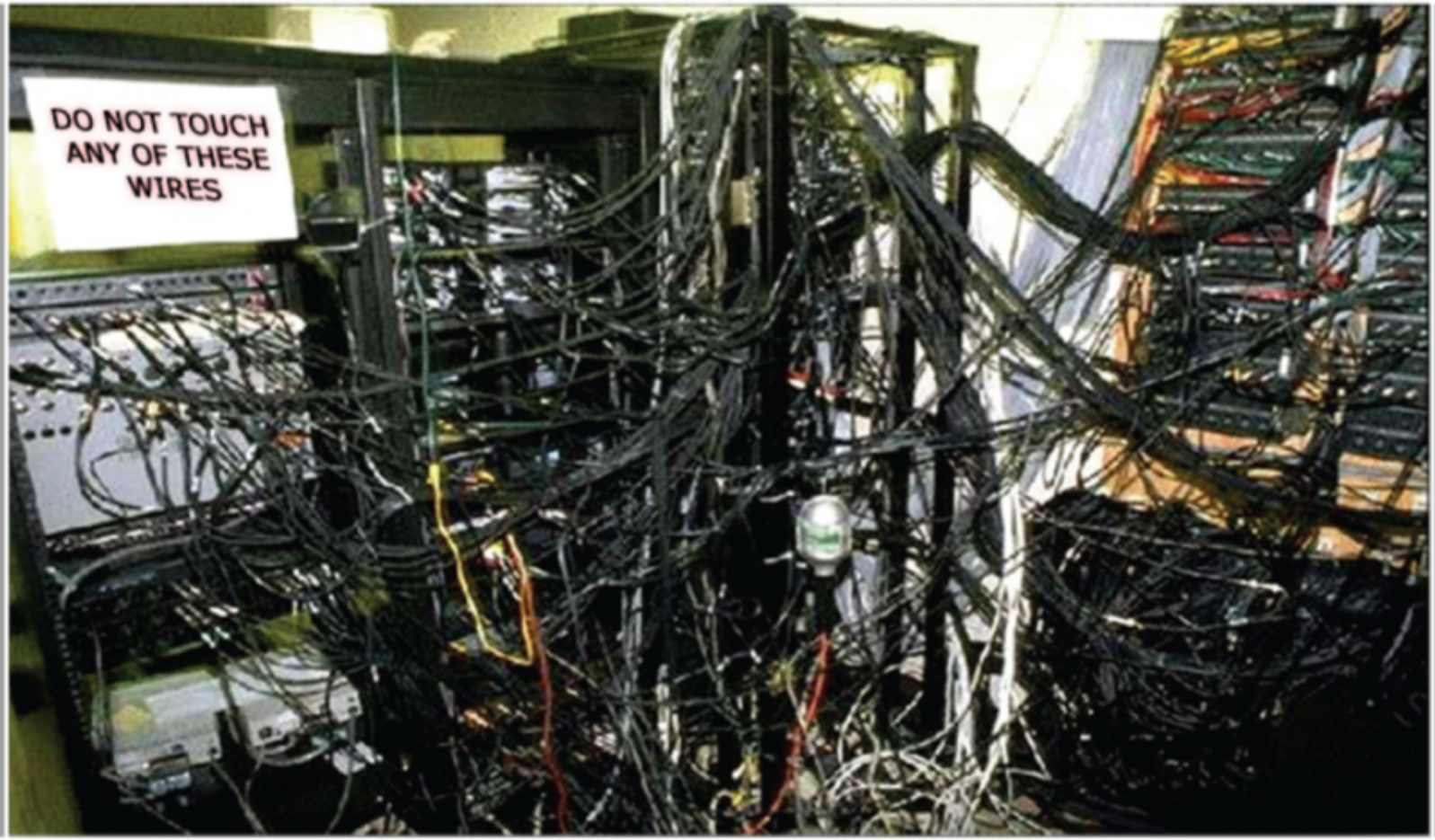
Department of information engineering,
computer science and mathematics
Center Of Excellence DEWS
University of L'Aquila, Italy

Modeling and Co-design of Control Tasks over Wireless Networking Protocols

Alessandro D'Innocenzo

October 27, 2016
Co^4 Workshop, Toulouse

Paradigm shift towards wireless control architectures



"Removing cables undoubtedly saves cost, but often the ***real cost gains lie in the radically different design approach that wireless solutions permit****. […] In order to fully benefit from wireless technologies, a rethink of existing automation concepts and the complete design and functionality of an application is required."* Jan-Erik Frey, R&D Manager ABB

Wireless Control Network

A collection of cooperating algorithms (controllers) designed to achieve a set of common goals, aided by interactions with the environment through distributed measurements (sensors) and actions (actuators) exchanged via a wireless communication network

WirelessHART

ISA100Wireless

Opportunities vs scientific challenges with Wireless Control Networks

Lower costs, easier installation

- Suitable for emerging markets

Broadens scope of sensing and control

- Easier to sense/monitor/actuate: opens new application domains

Compositionality

- Enables system evolution via composable control loops

Runtime adaptation and reconfiguration

- Control can be maintained in response to failures and malicious attacks

---

Complexity

- Systems designers and programmers need **suitable abstract models** to hide the complexity from wireless channels and communication protocols

Reliability

- Need for **robust and predictable behavior** despite wireless non-idealities

Security

- Wireless technology is vulnerable: **security mechanisms** for control loops

Take into account communication protocol dynamics

Opportunities vs scientific challenges with Wireless Control Networks

Lower costs, easier installation

- Suitable for emerging markets

Broadens scope of sensing and control

- Easier to sense/monitor/actuate: opens new application domains

Compositionality

- Enables system evolution via composable control loops

Runtime adaptation and reconfiguration

- Control can be maintained in response to failures and malicious attacks

---

Complexity

- Systems designers and programmers need **suitable abstract models** to hide the complexity from wireless channels and communication protocols

Reliability

- Need for **robust and predictable behavior** despite wireless non-idealities

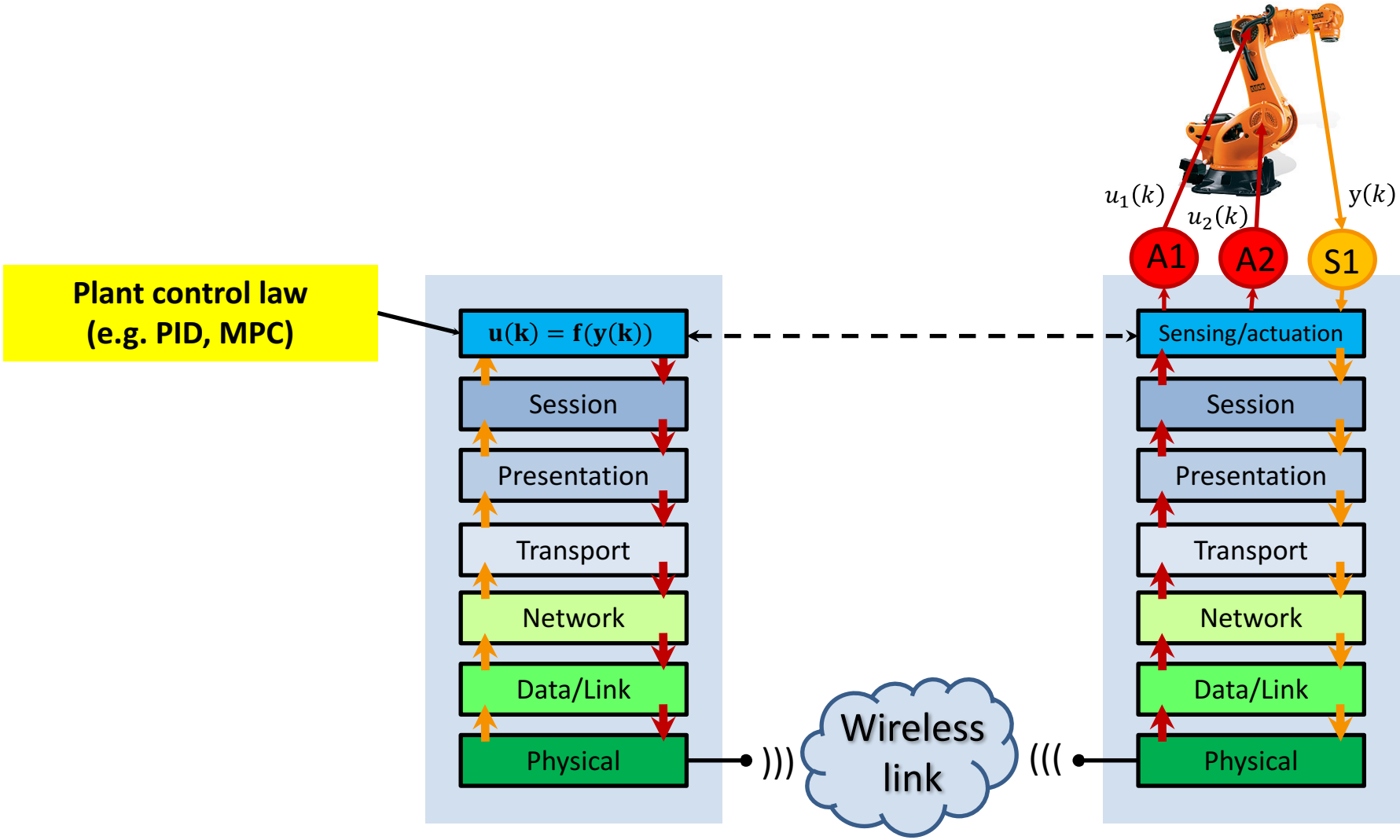
Security

- Wireless technology is vulnerable: **security mechanisms** for control loops

Take into account communication protocol dynamics

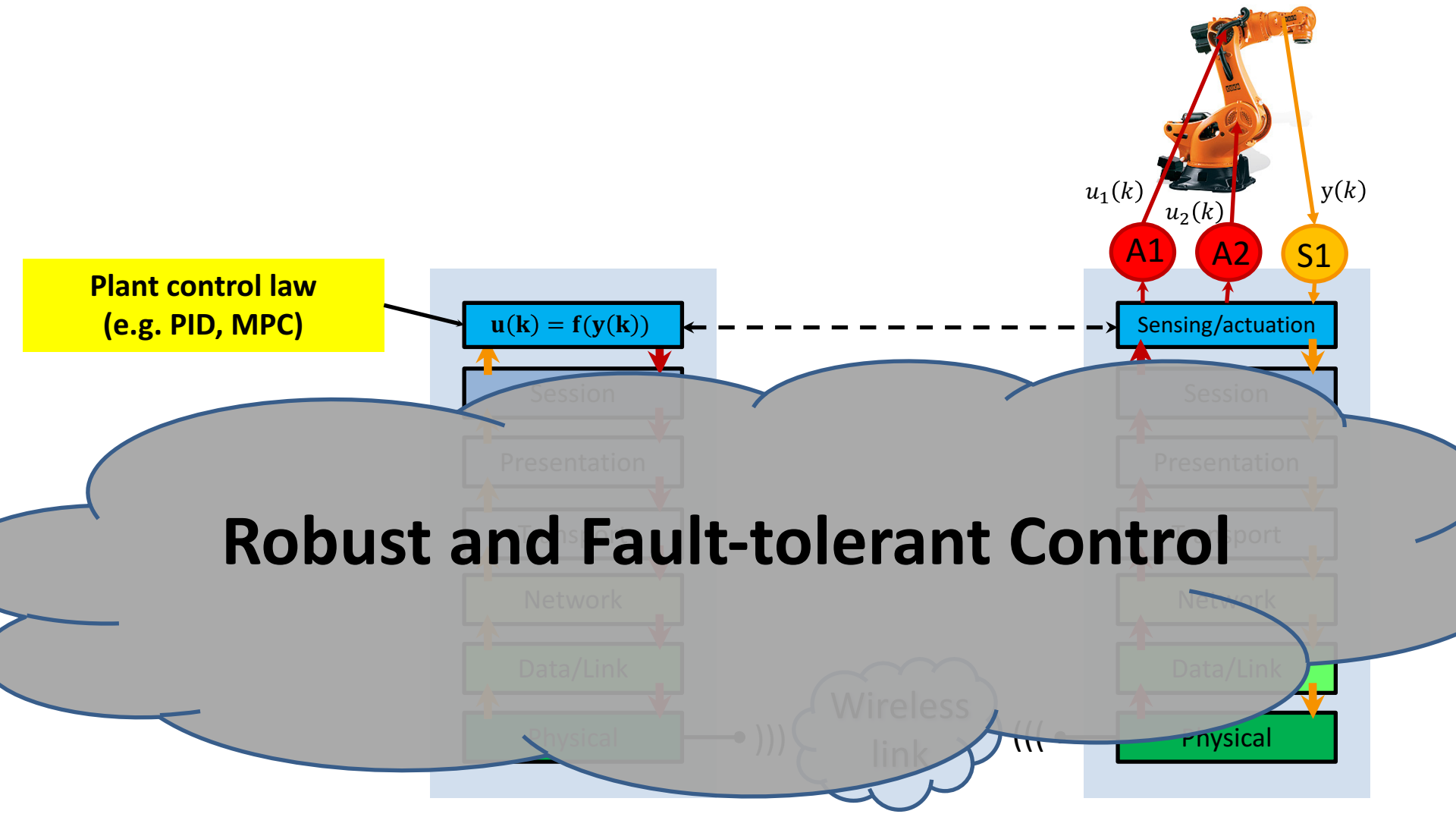
Control loop over a P2P wireless network

- Sensing and actuation data are relayed via the protocol stack layers



Control loop over a P2P wireless network

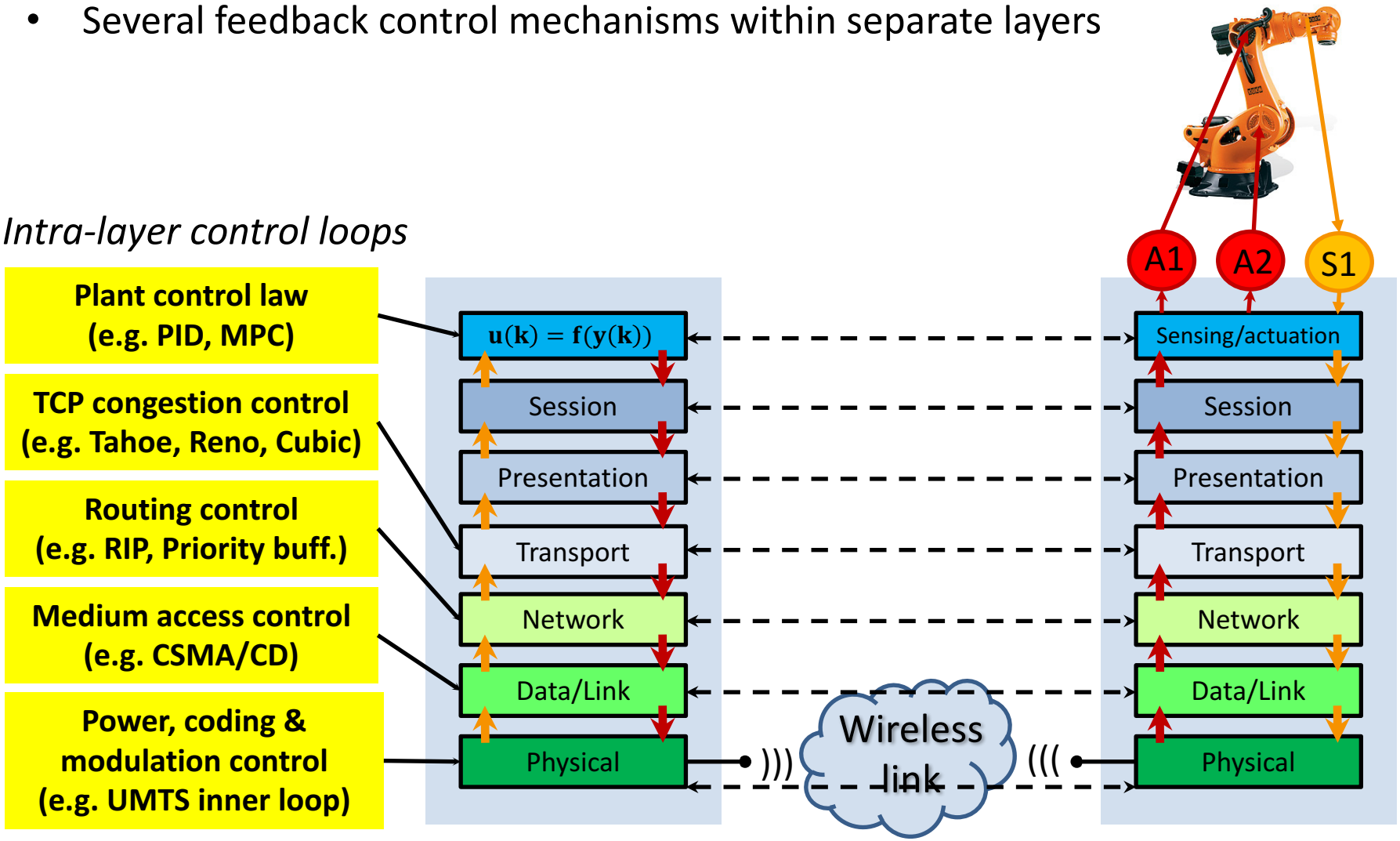
- Sensing and actuation data are relayed via the protocol stack layers
- In classical control theory communication stack and medium are considered as generic disturbances in the controller design



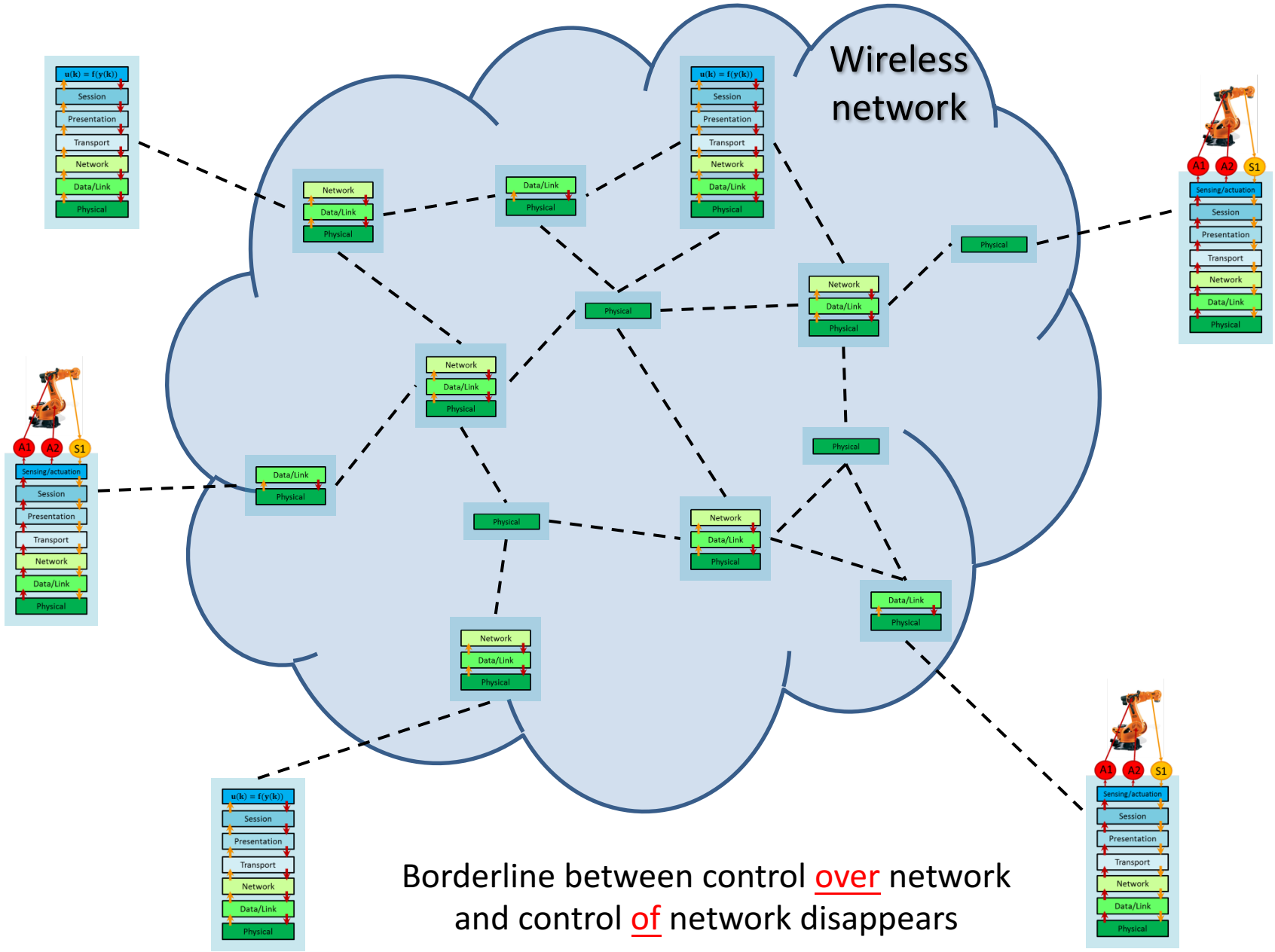
Control loops over a P2P wireless network

- Sensing and actuation data are relayed via the protocol stack layers
- In classical control theory communication stack and medium are considered as generic disturbances in the controller design
- Several feedback control mechanisms within separate layers

Intra-layer control loops

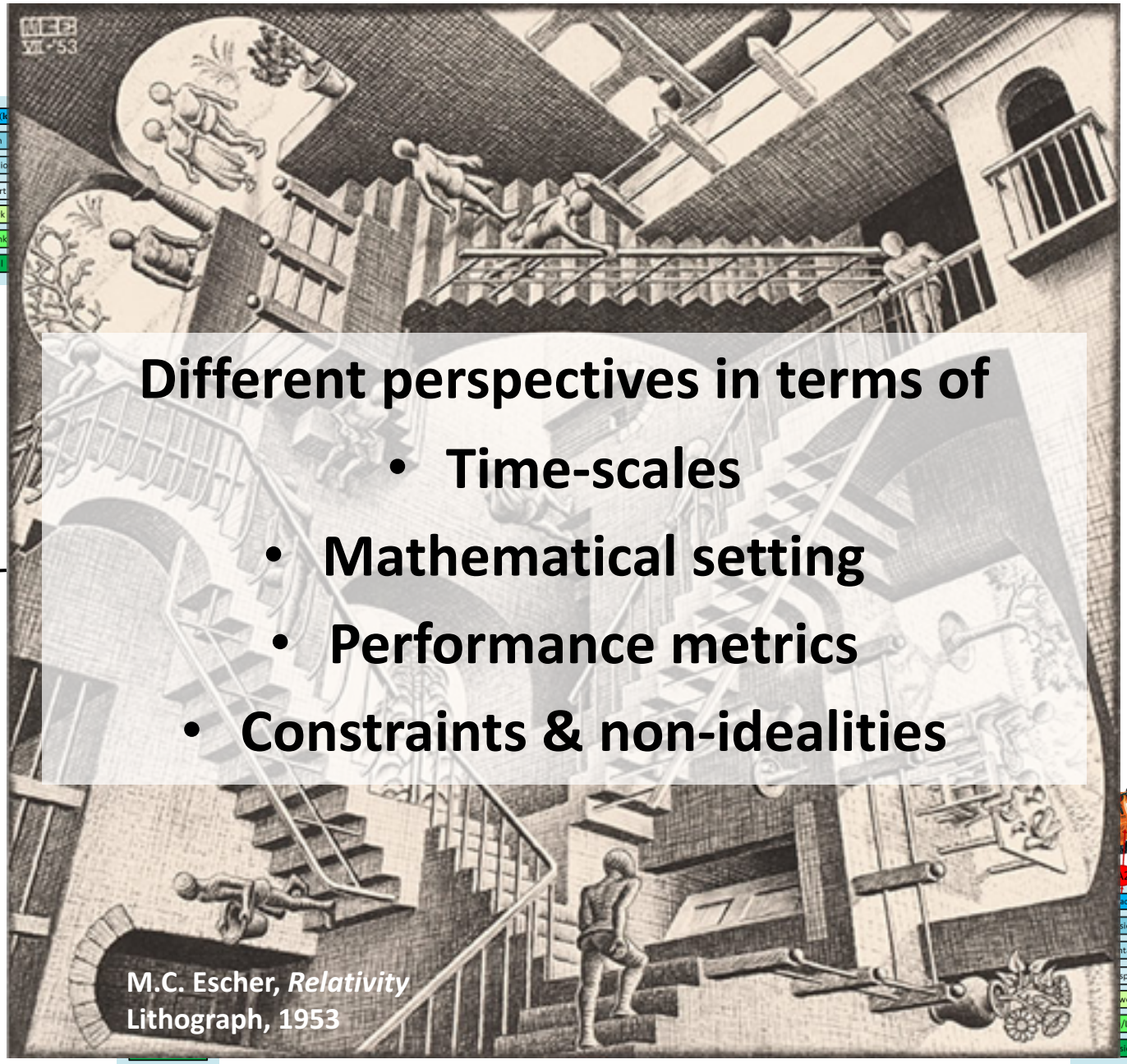


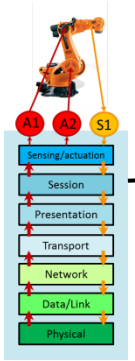
Control loops over a mesh wireless network

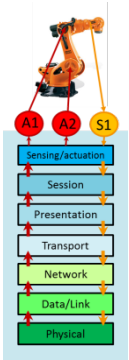


Borderline between control over network and control of network disappears

Control loops over a mesh wireless network





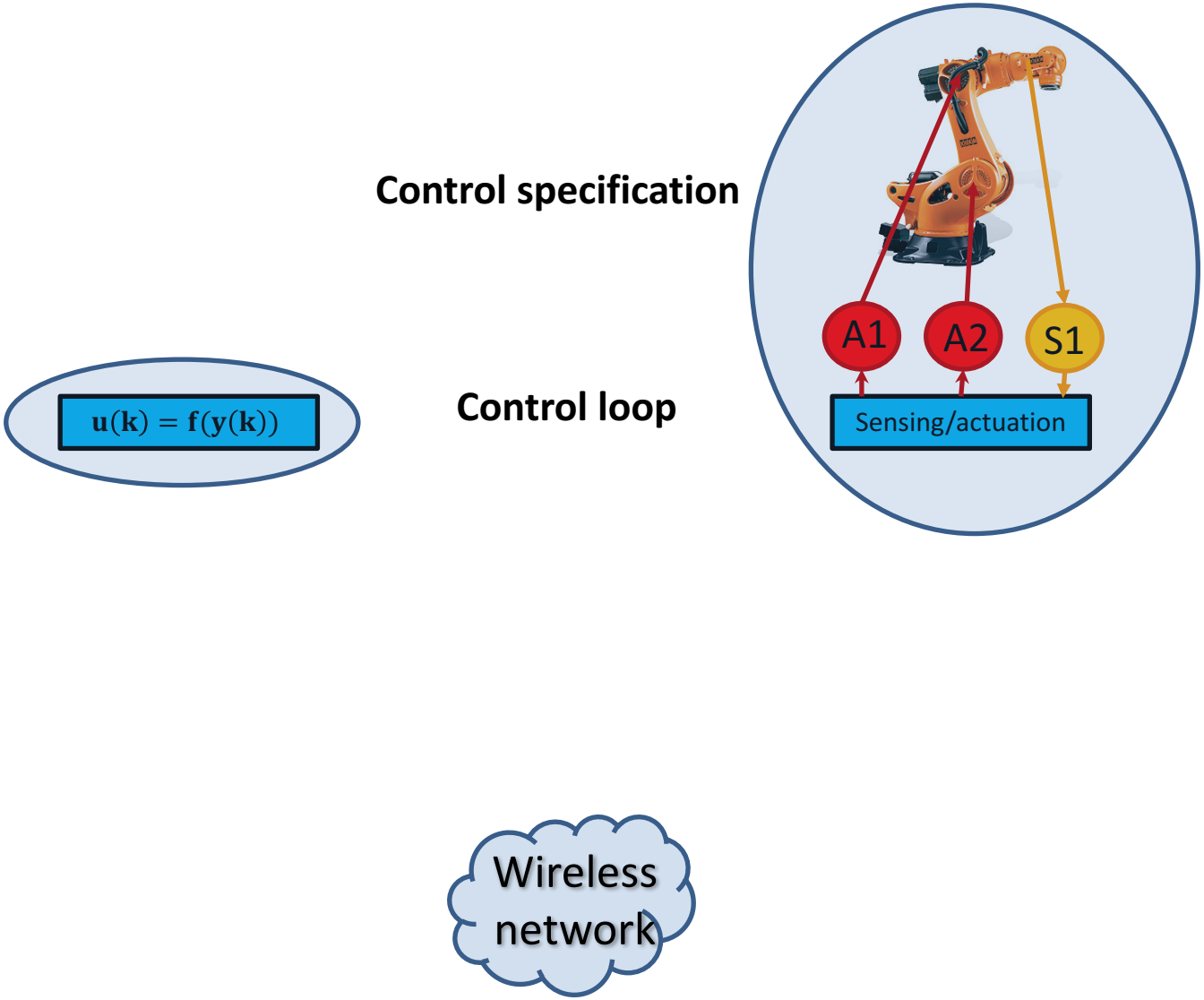


Different perspectives in terms of

- **Time-scales**
- **Mathematical setting**
- **Performance metrics**
- **Constraints & non-idealities**

Control-aware networking and communication

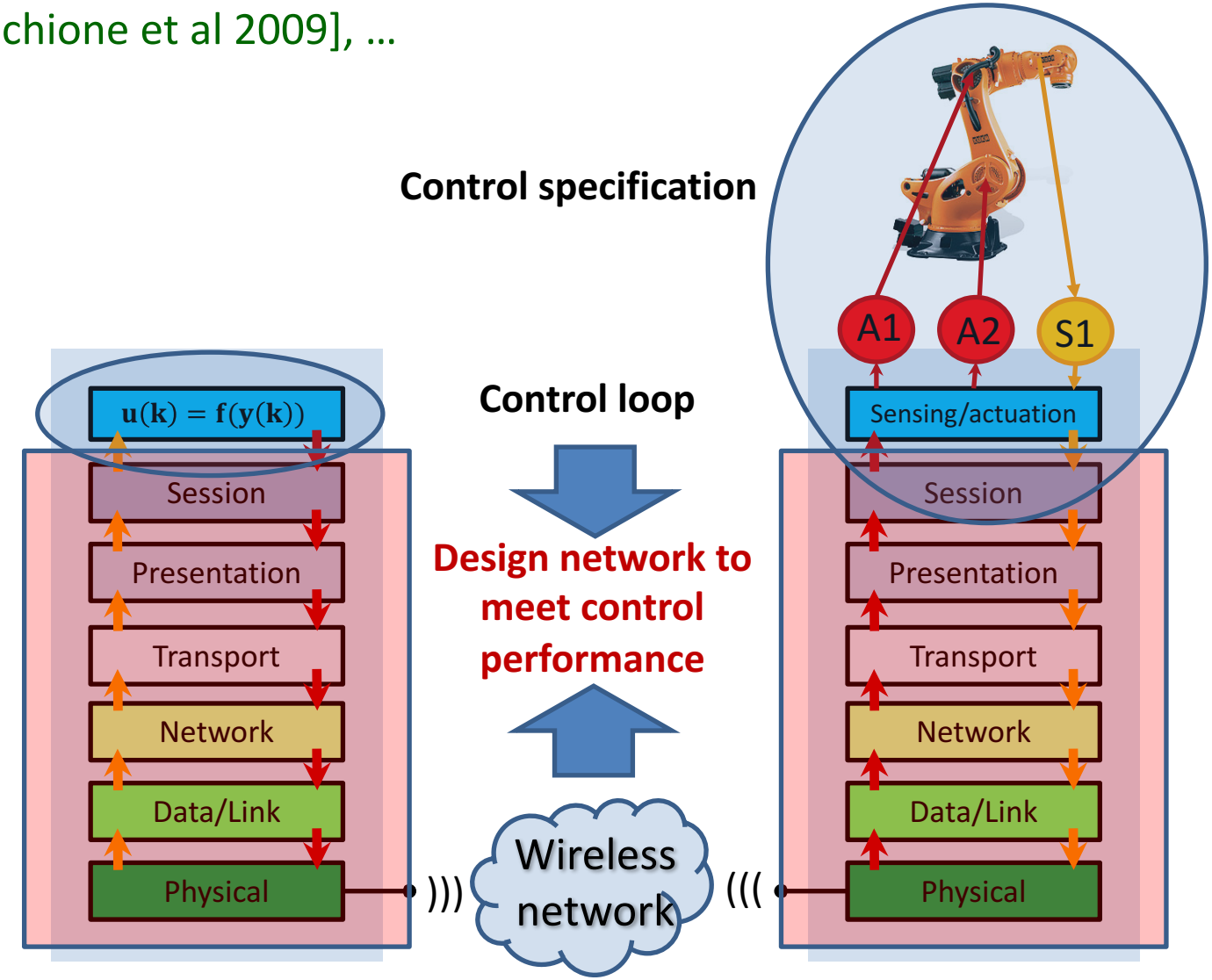
Modify network protocols and radio links for better real-time control performance



Control-aware networking and communication

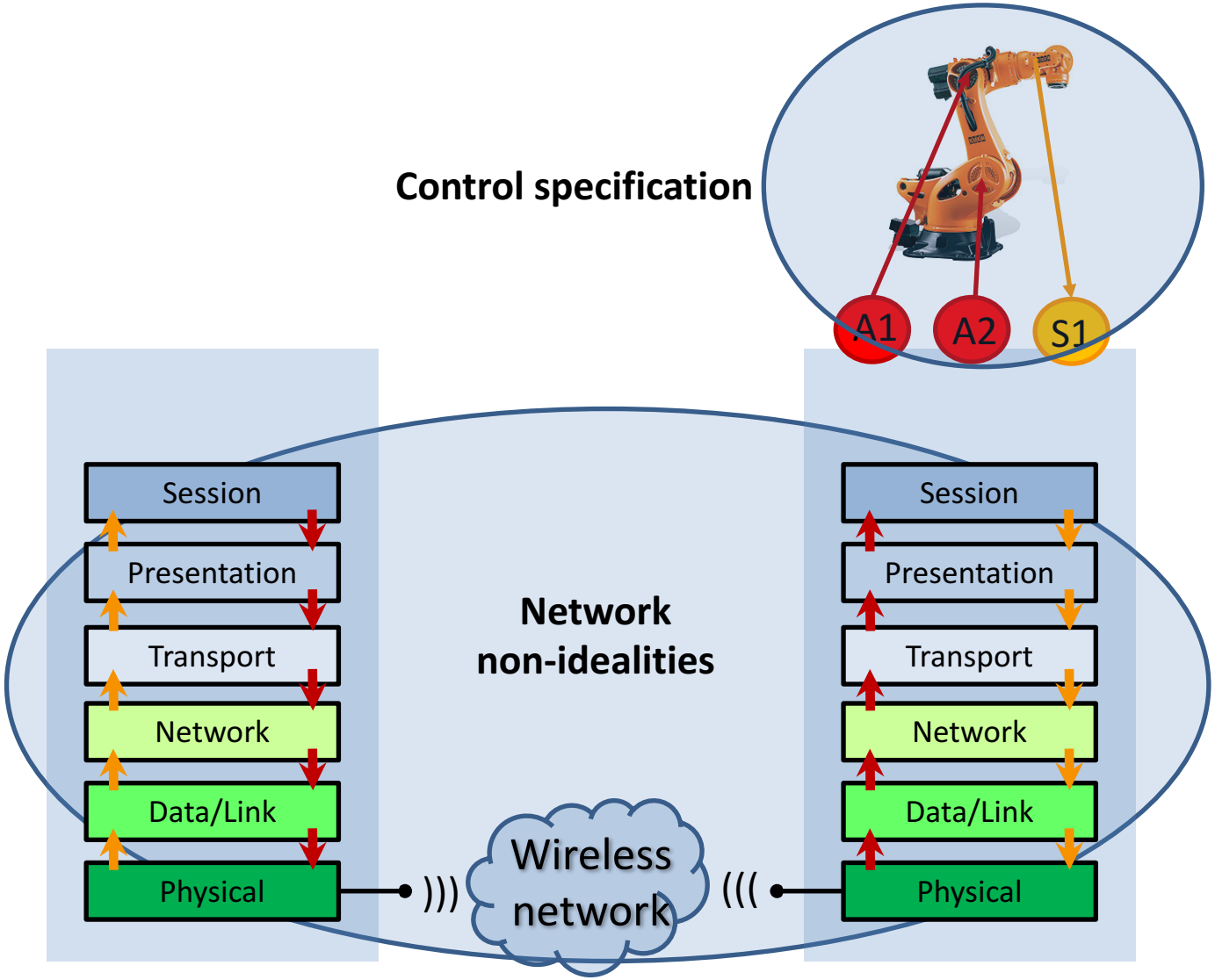
Modify network protocols and radio links for better real-time control performance

[Park et al 2011], [Fischione et al 2009], …



Network-aware control

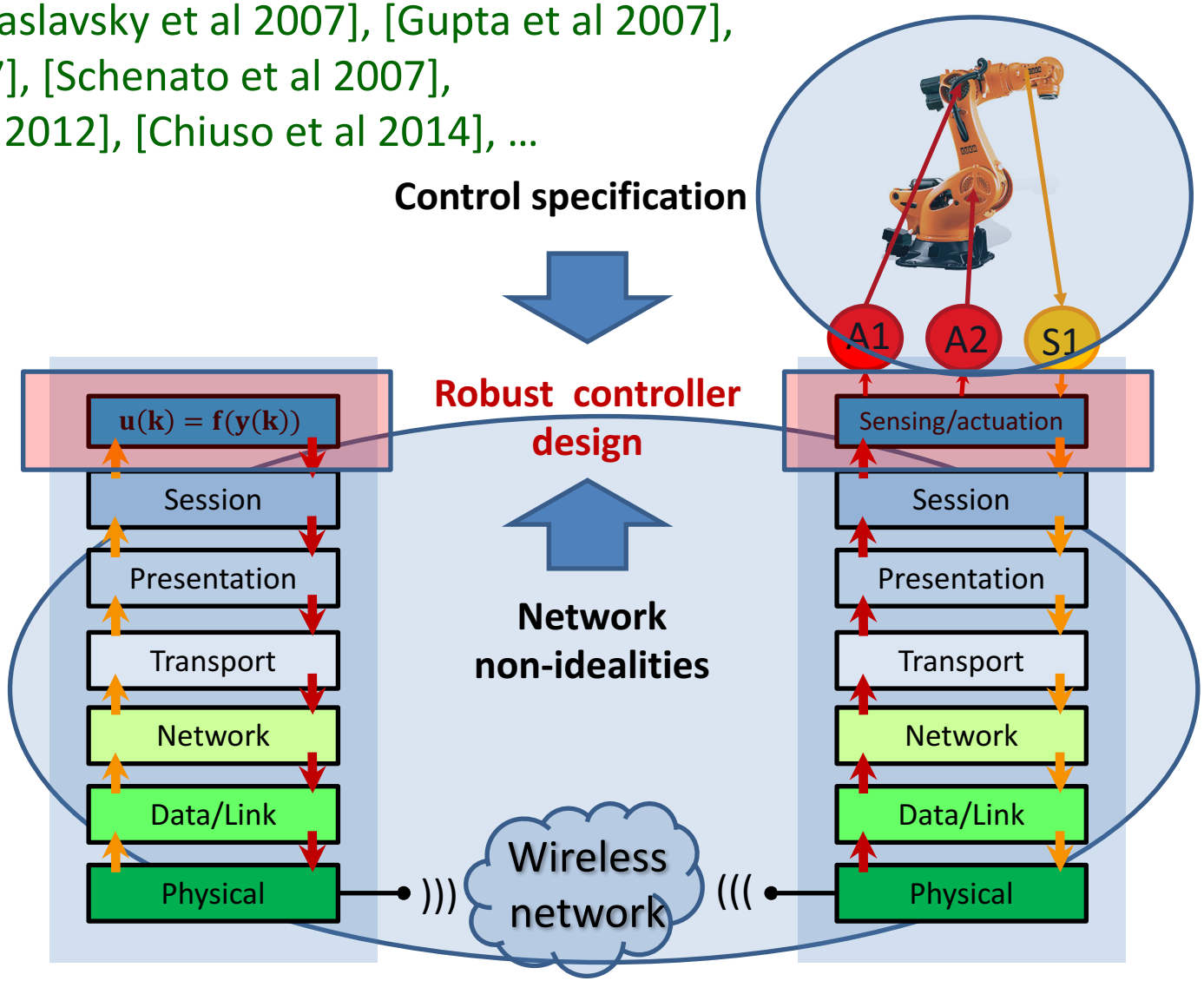
Modify control algorithms to cope with communication imperfections



Network-aware control

Modify control algorithms to cope with communication imperfections

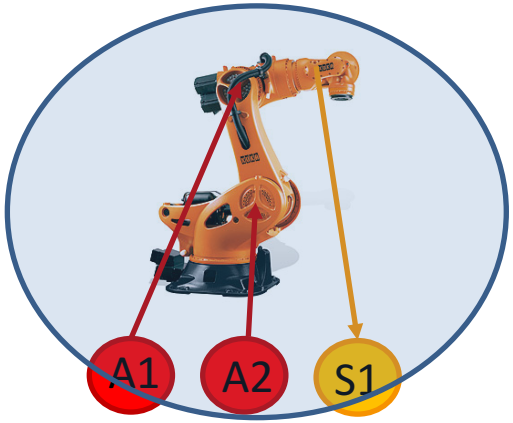
[Seiler&Sengupta 2001], [Jacobsson et al. 2004], [Sinopoli et al 2004], [Elia 2005], [Imer et al 2006], [Braslavsky et al 2007], [Gupta et al 2007], [Hespanha et al 2007], [Schenato et al 2007], [Heemels et al 2011, 2012], [Chiuso et al 2014], …



Co-design

Joint design of the control algorithm and the network protocol configuration

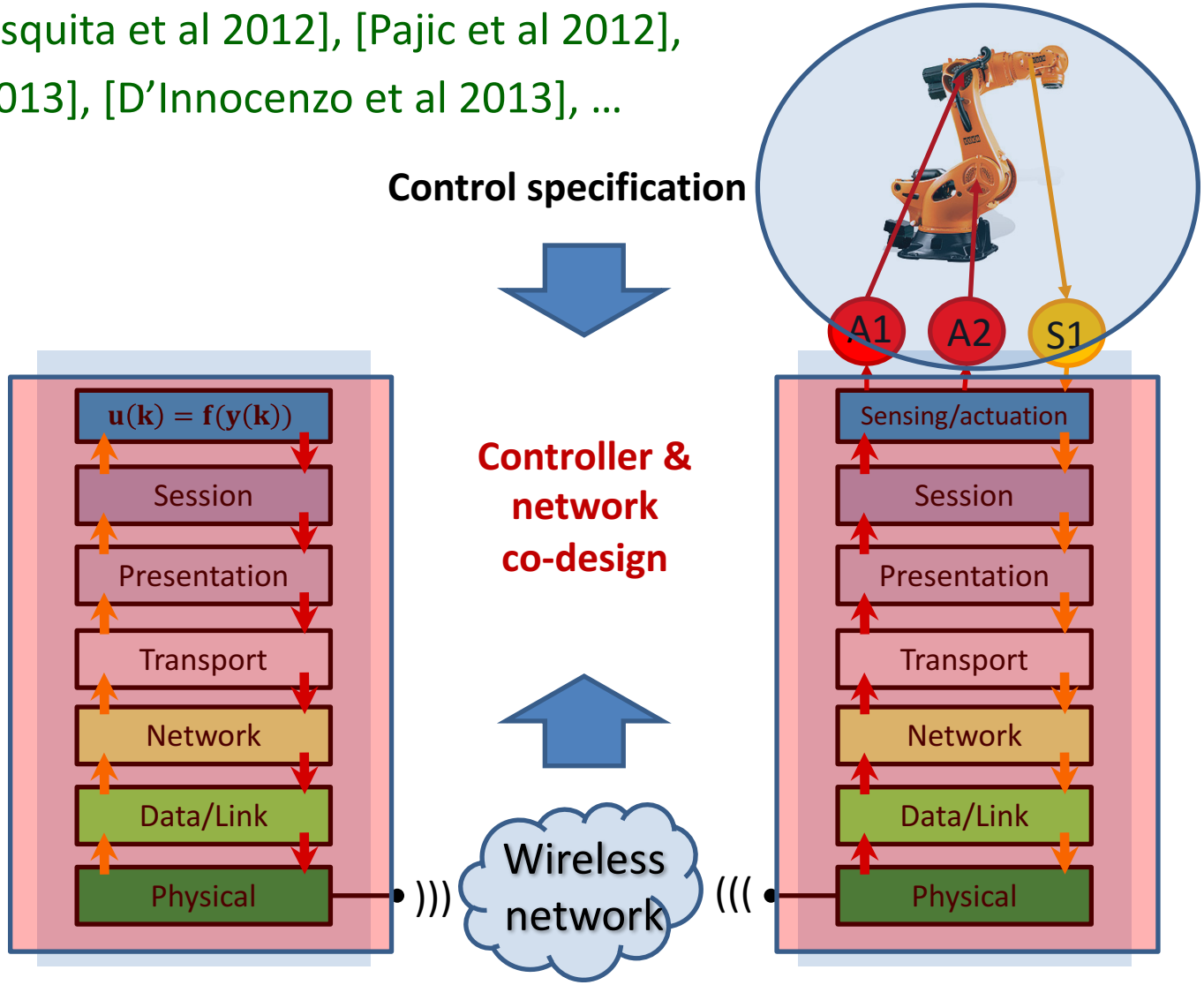
Control specification



Co-design

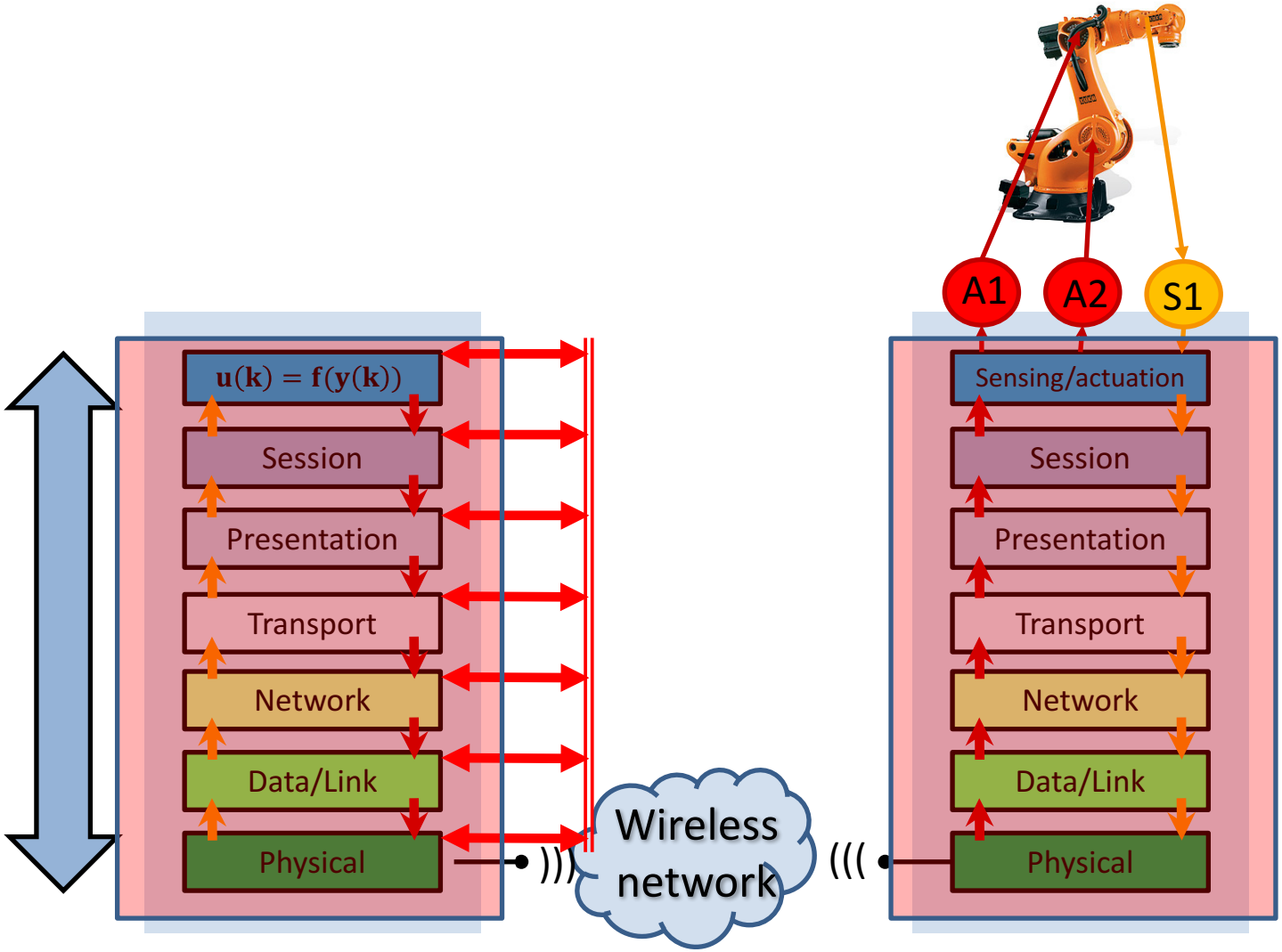
Joint design of the control algorithm and the network protocol configuration

[Park et al 2011], [Mesquita et al 2012], [Pajic et al 2012], [Antunes&Heemels 2013], [D'Innocenzo et al 2013], …



Cross-layer adaptation & optimization

Desirable signalling between communication layers to improve overall performance

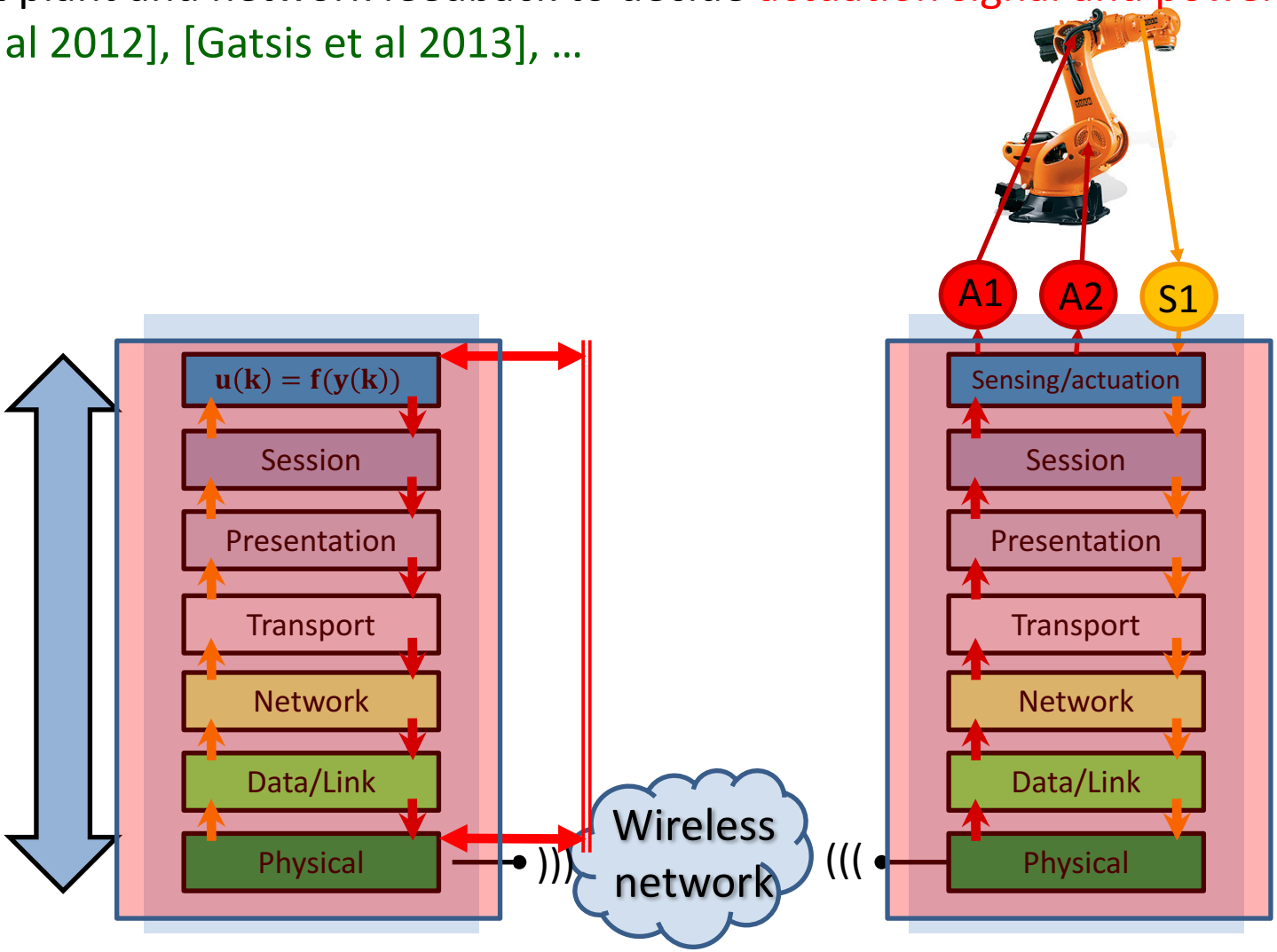


Cross-layer adaptation & optimization

Desirable signalling between communication layers to improve overall performance

Example: exploit plant and network feedback to decide actuation signal and power

[D'Innocenzo et al 2012], [Gatsis et al 2013], …

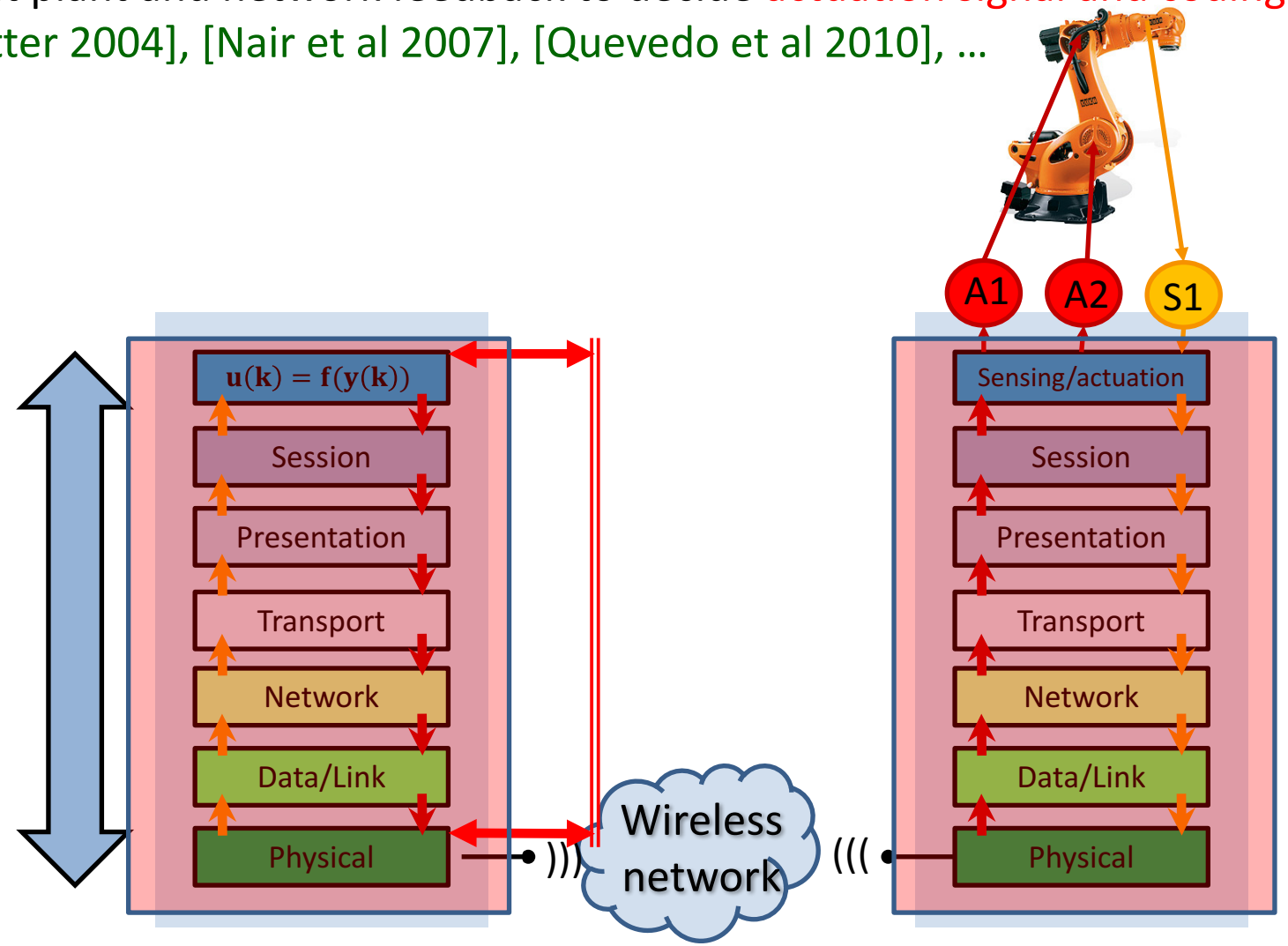


Cross-layer adaptation & optimization

Desirable signalling between communication layers to improve overall performance

Example: exploit plant and network feedback to decide actuation signal and coding

[Tatikonda&Mitter 2004], [Nair et al 2007], [Quevedo et al 2010], …

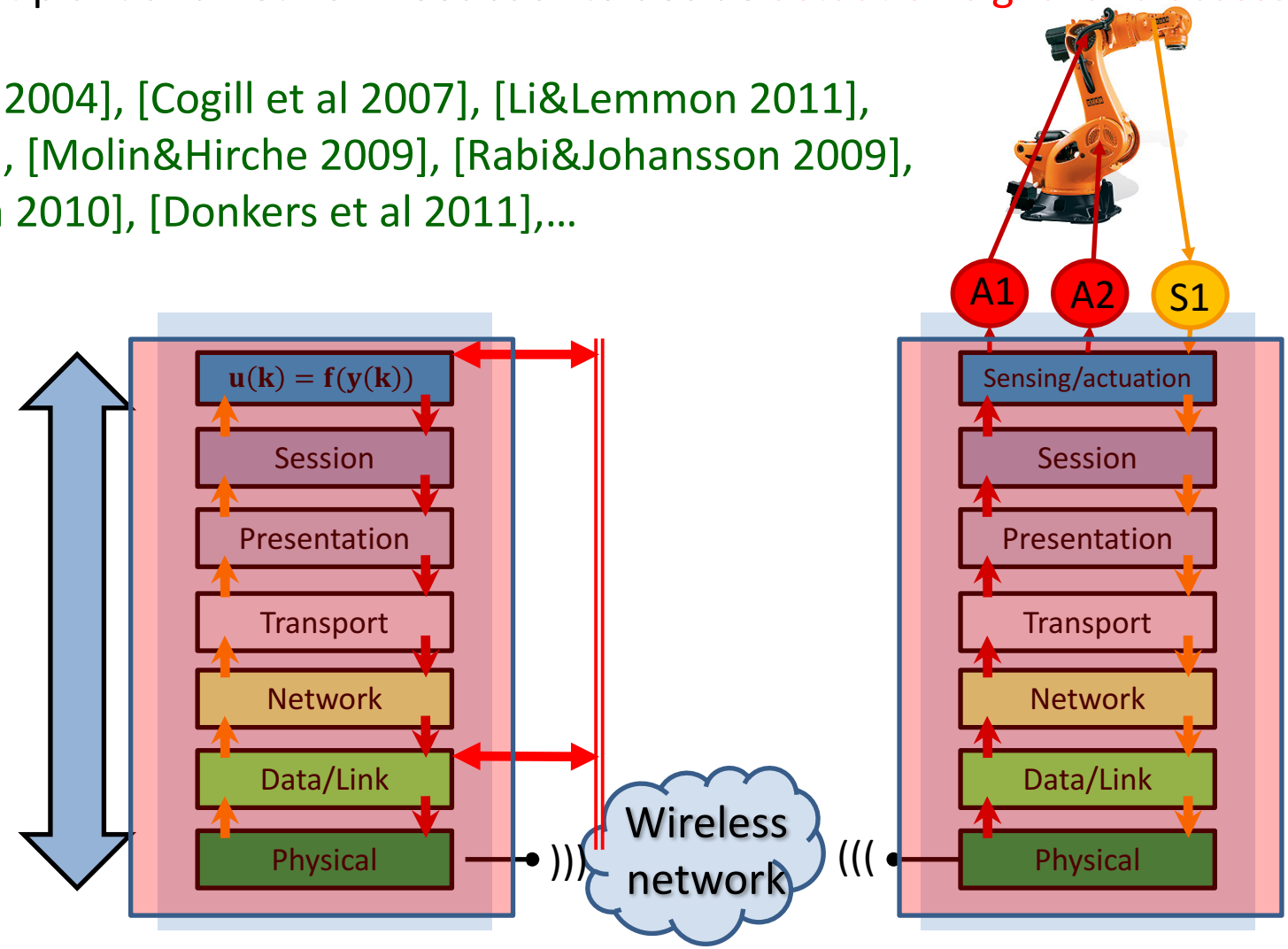


Cross-layer adaptation & optimization

Desirable signalling between communication layers to improve overall performance

Example: exploit plant and network feedback to decide actuation signal and access to channel

[Xu&Hespanha 2004], [Cogill et al 2007], [Li&Lemmon 2011], [Tabuada 2007], [Molin&Hirche 2009], [Rabi&Johansson 2009], [Anta&Tabuada 2010], [Donkers et al 2011],…

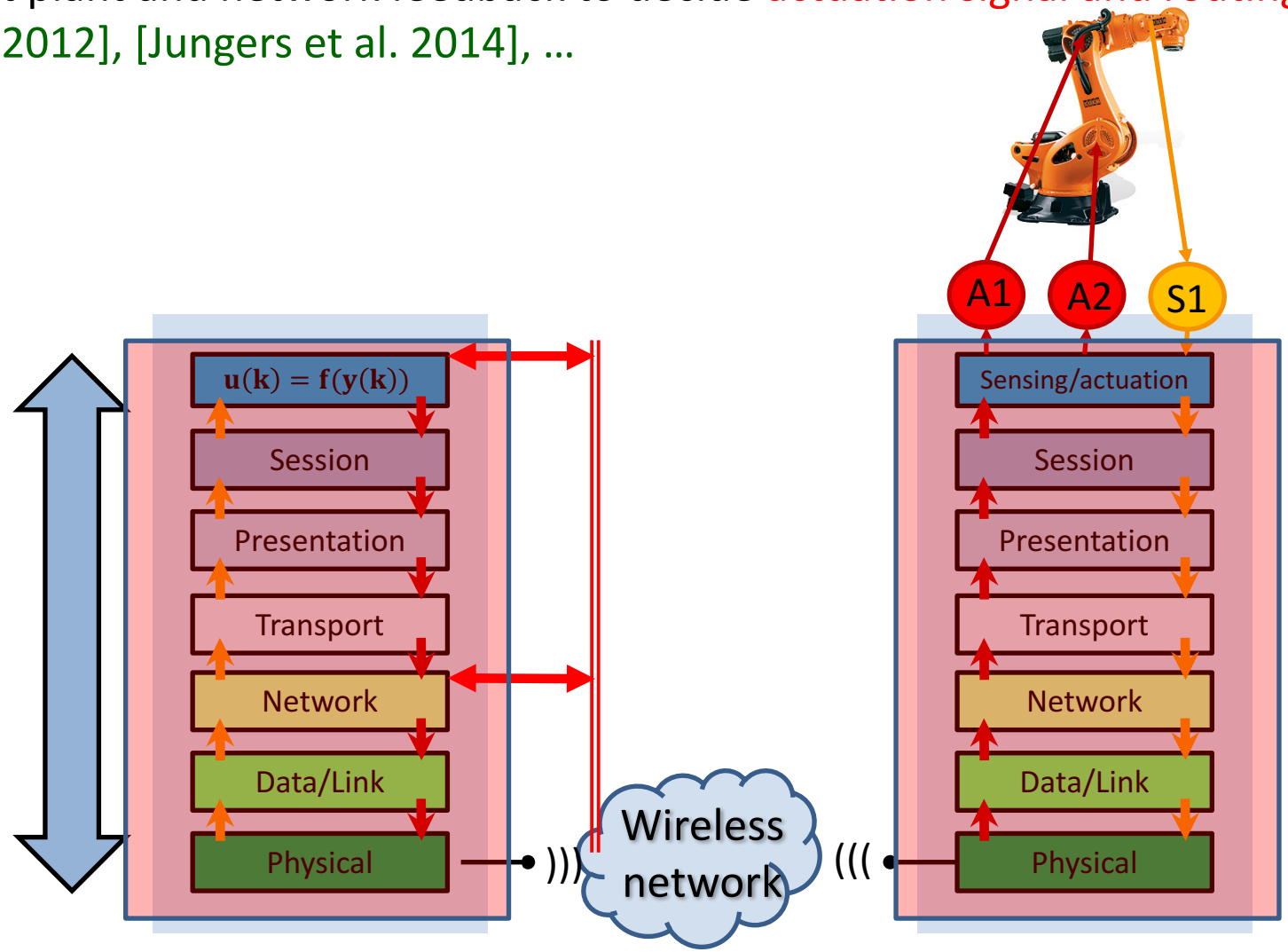


Cross-layer adaptation & optimization

Desirable signalling between communication layers to improve overall performance

Example: exploit plant and network feedback to decide actuation signal and routing

[Mesquita et al 2012], [Jungers et al. 2014], …



Opportunities vs scientific challenges with Wireless Control Networks

Lower costs, easier installation

- Suitable for emerging markets

Broadens scope of sensing and control

- Easier to sense/monitor/actuate: opens new application domains

Compositionality

- Enables system evolution via composable control loops

Runtime adaptation and reconfiguration

- Control can be maintained in response to failures and malicious attacks

---

Complexity

- Systems designers and programmers need **suitable abstract models** to hide the complexity from wireless channels and communication protocols

Reliability

- Need for **robust and predictable behavior** despite wireless non-idealities

Security

- Wireless technology is vulnerable: **security mechanisms** for control loops

Take into account communication protocol dynamics

Role of communication in cyber-physical security research?

Y. Zacchia Lun , A. D'Innocenzo , I. Malavolta and M.D. Di Benedetto. Cyber-Physical Systems Security: a Systematic Mapping Study. Submitted for publication, preprint on arXiv.

Experimental Security Analysis of a Modern Automobile

Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, and Tadayoshi Kohno
Department of Computer Science and Engineering
University of Washington
Seattle, Washington 98195–2350
Email: {supersat,aczeskis,franzi,shwetak,yoshi}@cs.washington.edu

Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and Stefan Savage
Department of Computer Science and Engineering
University of California San Diego
La Jolla, California 92093–0404
Email: {s,dlmccoy,brian,d8anders,hovav,savage}@cs.ucsd.edu

***Abstract*—Modern automobiles are no longer mere mechanical devices; they are pervasively monitored and controlled by dozens of digital computers coordinated via internal vehicular networks. While this transformation has driven major advancements in efficiency and safety, it has also introduced a range of new potential risks. In this paper we experimentally evaluate these issues on a modern automobile and demonstrate the fragility of the underlying system structure. We demonstrate that an attacker who is able to infiltrate virtually any Electronic Control Unit (ECU) can leverage this ability to completely circumvent a broad array of safety-critical systems. Over a range of experiments, both in the lab and in road tests, we demonstrate the ability to adversarially control a wide range of automotive functions and completely ignore driver input — including disabling the brakes, selectively braking individual wheels on demand, stopping the engine, and so on. We find that it is possible to bypass rudimentary network security protections within the car, such as maliciously bridging between our car's two internal subnets. We also present composite attacks that leverage individual weaknesses, including an attack that embeds malicious code in a car's telematics unit and that will completely erase any evidence of its presence after a crash. Looking forward, we discuss the complex challenges in addressing these vulnerabilities while considering the existing automotive ecosystem.**

***Keywords*—Automobiles, communication standards, communication system security, computer security, data buses.**

I. Introduction

Through 80 years of mass-production, the passenger automobile has remained superficially static: a single gasoline-powered internal combustion engine; four wheels; and the familiar user interface of steering wheel, throttle, gearshift, and brake. However, in the past two decades the underlying control systems have changed dramatically. Today's automobile is no mere mechanical device, but contains a myriad of computers. These computers coordinate and monitor sensors, components, the driver, and the passengers. Indeed, one recent estimate suggests that the typical luxury sedan now contains over 100 MB of binary code spread across 50–70 independent computers — *Electronic Control Units* (*ECUs*) in automotive vernacular — in turn communicating over one or more shared *internal network buses* [8], [13].

While the automotive industry has always considered *safety* a critical engineering concern (indeed, much of this new software has been introduced specifically to *increase* safety, e.g., *Anti-lock Brake Systems*) it is not clear whether vehicle manufacturers have anticipated in their designs the possibility of an adversary. Indeed, it seems likely that this increasing degree of computerized control also brings with it a corresponding array of potential threats.

Compounding this issue, the attack surface for modern automobiles is growing swiftly as more sophisticated services and communications features are incorporated into vehicles. In the United States, the federally-mandated *On-Board Diagnostics* (*OBD-II*) port, under the dash in virtually all modern vehicles, provides direct and standard access to internal automotive networks. User-upgradable subsystems such as audio players are routinely attached to these same internal networks, as are a variety of short-range wireless devices (Bluetooth, wireless tire pressure sensors, etc.). Telematics systems, exemplified by General Motors' (GM's) OnStar, provide value-added features such as automatic crash response, remote diagnostics, and stolen vehicle recovery over a long-range wireless link. To do so, these telematics systems integrate internal automotive subsystems with a remote command center via a wide-area cellular connection. Some have taken this concept even further — proposing a "car as a platform" model for third-party development. Hughes Telematics has described plans for developing an "App Store" for automotive applications [22] while Ford recently announced that it will open its Sync telematics system as a platform for third-party applications [14]. Finally, proposed future vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2X) communications systems [5], [6], [7], [25] will only broaden the attack surface further.

Appears in *2010 IEEE Symposium on Security and Privacy*. See http://www.autosec.org/ for more information.

2008 IEEE Symposium on Security and Privacy

Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses

Daniel Halperin[†] University of Washington
Thomas S. Heydt-Benjamin[†] University of Massachusetts Amherst
Benjamin Ransford[†] University of Massachusetts Amherst
Shane S. Clark University of Massachusetts Amherst
Benessa Defend University of Massachusetts Amherst
Will Morgan University of Massachusetts Amherst
Kevin Fu, PhD* University of Massachusetts Amherst
Tadayoshi Kohno, PhD* University of Washington
William H. Maisel, MD, MPH* BIDMC and Harvard Medical School

***Abstract*—Our study analyzes the security and privacy properties of an implantable cardioverter defibrillator (ICD). Introduced to the U.S. market in 2003, this model of ICD includes pacemaker technology and is designed to communicate wirelessly with a nearby external programmer in the 175 kHz frequency range. After partially reverse-engineering the ICD's communications protocol with an oscilloscope and a software radio, we implemented several software radio-based attacks that could compromise patient safety and patient privacy. Motivated by our desire to improve patient safety, and mindful of conventional trade-offs between security and power consumption for resource-constrained devices, we introduce three new zero-power defenses based on RF power harvesting. Two of these defenses are human-centric, bringing patients into the loop with respect to the security and privacy of their implantable medical devices (IMDs). Our contributions provide a scientific baseline for understanding the potential security and privacy risks of current and future IMDs, and introduce human-perceptible and zero-power mitigation techniques that address those risks. To the best of our knowledge, this paper is the first in our community to use general-purpose software radios to analyze and attack previously unknown radio communications protocols.**

I. Introduction

Wirelessly programmable *implantable medical devices* (*IMDs*) such as *pacemakers*, *implantable cardioverter defibrillators (ICDs)*, *neurostimulators*, and implantable *drug pumps* use embedded computers and radios to monitor chronic disorders and treat patients with automatic therapies. For instance, an ICD that senses a rapid heartbeat can administer an electrical shock to restore a normal heart rhythm, then later report this event to a health care practitioner who uses a *commercial device programmer*[1] with wireless capabilities to extract data from the ICD or modify its settings without surgery. Between 1990 and 2002, over 2.6 million pacemakers and ICDs were implanted in patients in the United States [19]; clinical trials have shown that these devices significantly improve survival rates in certain populations [18]. Other research has discussed potential security and privacy risks of IMDs [1], [10], but we are unaware of any rigorous public investigation into the observable characteristics of a real commercial device. Without such a study, it is impossible for the research community to assess or address the security and privacy properties of past, current, and future devices. We address that gap in this paper and, based on our findings, propose and implement several prototype attack-mitigation techniques.

Our investigation was motivated by an interdisciplinary study of medical device safety and security, and relied on a diverse team of area specialists. Team members from the security and privacy community have formal training in computer science, computer engineering, and electrical engineering. One team member from the medical community is a practicing cardiologist with hundreds of pacemaker and implantable defibrillator patients and was past chairperson of the FDA's Circulatory System Medical Device Advisory Panel. Our technical contributions toward understanding and improving the security, privacy, and safety of these devices include: analyses; software radio-based methodologies; and human-perceptible and zero-power (battery-free) defenses.

Overview of contributions. We assess the security and privacy properties of a common ICD and present attacks on privacy, integrity, and availability. We show that the ICD discloses sensitive information in the clear (unencrypted); we demonstrate a *reprogramming* attack that changes the operation of (and the information contained in) the ICD; and

*Corresponding faculty authors:
- Kevin Fu, Medical Device Security Center, Department of Computer Science, University of Massachusetts Amherst, 140 Governors Drive, Amherst, Massachusetts 01003 (kevinfu@cs.umass.edu);
- Tadayoshi Kohno, Medical Device Security Center, Department of Computer Science and Engineering, University of Washington, Box 352350, Seattle, Washington 98195 (yoshi@cs.washington.edu);
- William H. Maisel, Medical Device Safety Institute, Beth Israel Deaconess Medical Center, Harvard Medical School, 185 Pilgrim Road, Baker 4, Boston, MA 02215 (wmaisel@bidmc.harvard.edu).

Additional information online at http://www.secure-medicine.org.
[†]Co-student leads listed in alphabetical order; each participated equally.

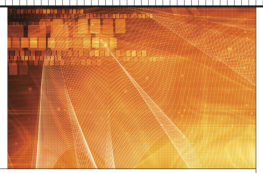
[1]The reader should not confuse the term "device programmer" with a person who programs computers. The former is an external device that communicates with and adjusts the settings on an IMD.

978-0-7695-3168-7/08 $25.00 © 2008 IEEE
DOI 10.1109/SP.2008.31

IEEE computer society

Lessons from Stuxnet

Thomas M. Chen, ***Swansea University***
Saeed Abu-Nimeh, ***Damballa Inc.***



Malware such as Stuxnet can affect critical physical infrastructures that are controlled by software, which implies that threats might extend to real lives.

Thousands of new malware appear in the wild daily. Most are evolutionary variants of existing families and don't have a widespread impact. However, occasionally a noteworthy new piece of malware will change the security landscape. For example, the 1988 Morris attack showed that an aggressive worm could bring down a substantial part of the Arpanet, and the 2003 SQL Slammer attack demonstrated that a simple user datagram protocol (UDP)-based worm could create devastating network congestion.

Stuxnet is teaching the security community new lessons. Since VirusBlokAda discovered the Windows worm in Belarus in July 2010, researchers have studied it intensely. They believe Stuxnet spread for several months before discovery and that it has already compromised its intended target.

As Table 1 shows, Stuxnet differs from past malware in several ways. First, most malware tries to infect as many computers as possible, whereas Stuxnet appears to target industrial control systems and delivers its payload under very specific conditions. Second, Stuxnet is larger and more complex than other malware. It contains exploits for four unpatched vulnerabilities—an unusually high number. The code is approximately 500 Kbytes and written in multiple languages. As a reference, the SQL Slammer worm was 376 bytes; the Code Red worm was approximately 4 Kbytes; the Nimda worm was 60 Kbytes; and variants of the Zeus banking Trojan ranged between 40 and 150 Kbytes. Virtually all malware is less than 1 Mbyte.

Based on Stuxnet's code, experts have speculated on its creators and intention. Its sophistication suggests that the creators had detailed knowledge of its target and access to immense resources, perhaps with government backing. Its choice of targets also suggests a political motive.

TARGET SELECTION

Unlike most malware, Stuxnet targets industrial control systems, which are used widely in factories, assembly lines, refineries, and power plants. It attacks Windows PCs that program specific Siemens programmable logic controllers—specialized computers that control automated physical processes, such as robot arms, in common industrial control systems. PLCs can have elaborate input/output arrangements for various applications in different physical environments. They often have sensors on the inputs (for example, for temperature), and the outputs typically operate equipment such as motors, switches, and relays.

Stuxnet targets vulnerable PCs running WinCC/Step 7 control software, which is normally used to program PLCs. When an infected PC connects to a Siemens Simatic PLC, Stuxnet installs a malicious .dll file, replacing the PLC's original .dll file. The malicious .dll file lets Stuxnet monitor and intercept all communication

Table 1. Stuxnet's novel characteristics.

Aspect	Stuxnet	Common malware
Targeting	Extremely selective	Indiscriminate
Type of target	Industrial control systems	Computers
Size	500 Kbytes	Less than 1 Mbyte
Probable initial infection vector	Removable flash drive	Internet and other networks
Exploits	Four zero-days	Possibly one zero-day

0018-9162/11/$26.00 © 2011 IEEE
Published by the IEEE Computer Society
APRIL 2011

Role of communication in cyber-physical security research?

Y. Zacchia Lun , A. D'Innocenzo , I. Malavolta and M.D. Di Benedetto. Cyber-Physical Systems Security: a Systematic Mapping Study. Submitted for publication, preprint on arXiv.

A systematic mapping study is a research methodology intended to provide an ***unbiased, objective and systematic*** *instrument* to ***identify, classify, and analyze*** *existing research on a specific research area: cyber-physical systems security* in our case.

K. Petersen, S. Vakkalanka, and L. Kuzniarz, "Guidelines for conducting systematic mapping studies in software engineering: An update," Information and Software Technology, vol. 64, pp. 1–18, 2015

Experimental Security Analysis of a Modern Automobile

Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, and Tadayoshi Kohno
Department of Computer Science and Engineering
University of Washington
Seattle, Washington 98195–2350
Email: {supersat,aczeskis,franzi,shwetak,yoshi}@cs.washington.edu

Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and Stefan Savage
Department of Computer Science and Engineering
University of California San Diego
La Jolla, California 92093–0404
Email: {s,dlmccoy,brian,d8anders,hovav,savage}@cs.ucsd.edu

***Abstract*—Modern automobiles are no longer mere mechanical devices; they are pervasively monitored and controlled by dozens of digital computers coordinated via internal vehicular networks. While this transformation has driven major advancements in efficiency and safety, it has also introduced a range of new potential risks. In this paper we experimentally evaluate these issues on a modern automobile and demonstrate the fragility of the underlying system structure. We demonstrate that an attacker who is able to infiltrate virtually any Electronic Control Unit (ECU) can leverage this ability to completely circumvent a broad array of safety-critical systems. Over a range of experiments, both in the lab and in road tests, we demonstrate the ability to adversarially control a wide range of automotive functions and completely ignore driver input—including disabling the brakes, selectively braking individual wheels on demand, stopping the engine, and so on. We find that it is possible to bypass rudimentary network security protections within the car, such as maliciously bridging between our car's two internal subnets. We also present composite attacks that leverage individual weaknesses, including an attack that embeds malicious code in a car's telematics unit and that will completely erase any evidence of its presence after a crash. Looking forward, we discuss the complex challenges in addressing these vulnerabilities while considering the existing automotive ecosystem.**

***Keywords*—Automobiles, communication standards, communication system security, computer security, data buses.**

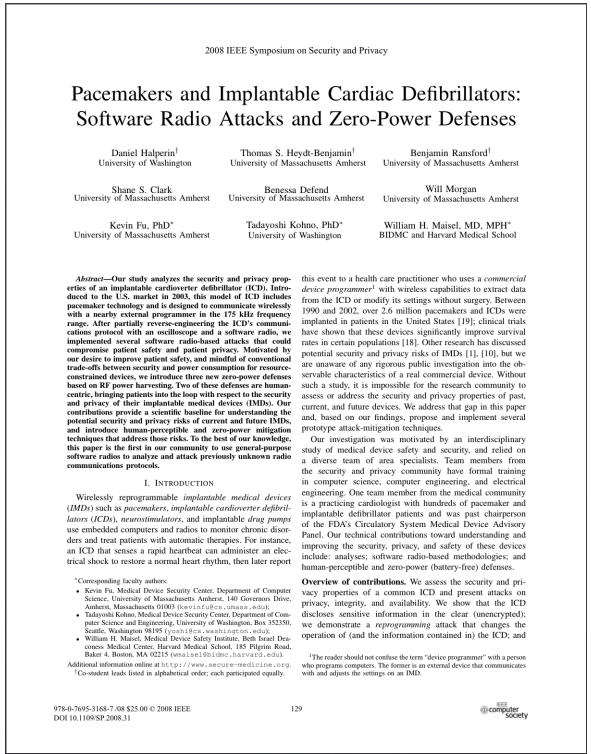
I. Introduction

Through 80 years of mass-production, the passenger automobile has remained superficially static: a single gasoline-powered internal combustion engine; four wheels; and the familiar user interface of steering wheel, throttle, gearshift, and brake. However, in the past two decades the underlying control systems have changed dramatically. Today's automobile is no mere mechanical device, but contains a myriad of computers. These computers coordinate and monitor sensors, components, the driver, and the passengers. Indeed, one recent estimate suggests that the typical luxury sedan now contains over 100 MB of binary code spread across 50–70 independent computers—*Electronic Control Units* (*ECUs*) in automotive vernacular—in turn communicating over one or more shared *internal network buses* [8], [13].

While the automotive industry has always considered *safety* a critical engineering concern (indeed, much of this new software has been introduced specifically to *increase* safety, e.g., *Anti-lock Brake Systems*) it is not clear whether vehicle manufacturers have anticipated in their designs the possibility of an adversary. Indeed, it seems likely that this increasing degree of computerized control also brings with it a corresponding array of potential threats.

Compounding this issue, the attack surface for modern automobiles is growing swiftly as more sophisticated services and communications features are incorporated into vehicles. In the United States, the federally-mandated *On-Board Diagnostics* (*OBD-II*) port, under the dash in virtually all modern vehicles, provides direct and standard access to internal automotive networks. User-upgradable subsystems such as audio players are routinely attached to these same internal networks, as are a variety of short-range wireless devices (Bluetooth, wireless tire pressure sensors, etc.). Telematics systems, exemplified by General Motors' (GM's) OnStar, provide value-added features such as automatic crash response, remote diagnostics, and stolen vehicle recovery over a long-range wireless link. To do so, these telematics systems integrate internal automotive subsystems with a remote command center via a wide-area cellular connection. Some have taken this concept even further—proposing a "car as a platform" model for third-party development. Hughes Telematics has described plans for developing an "App Store" for automotive applications [22] while Ford recently announced that it will open its Sync telematics system as a platform for third-party applications [14]. Finally, proposed future vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2X) communications systems [5], [6], [7], [25] will only broaden the attack surface further.

Appears in *2010 IEEE Symposium on Security and Privacy*. See http://www.autosec.org/ for more information. 1



2008 IEEE Symposium on Security and Privacy

Pacemakers and Implantable Cardiac Defibrillators: Software Radio Attacks and Zero-Power Defenses

Daniel Halperin[†] University of Washington
Thomas S. Heydt-Benjamin[†] University of Massachusetts Amherst
Benjamin Ransford[†] University of Massachusetts Amherst
Shane S. Clark University of Massachusetts Amherst
Benessa Defend University of Massachusetts Amherst
Will Morgan University of Massachusetts Amherst
Kevin Fu, PhD* University of Massachusetts Amherst
Tadayoshi Kohno, PhD* University of Washington
William H. Maisel, MD, MPH* BIDMC and Harvard Medical School

***Abstract*—Our study analyzes the security and privacy properties of an implantable cardioverter defibrillator (ICD). Introduced to the U.S. market in 2003, this model of ICD includes pacemaker technology and is designed to communicate wirelessly with a nearby external programmer in the 175 kHz frequency range. After partially reverse-engineering the ICD's communications protocol with an oscilloscope and a software radio, we implemented several software radio-based attacks that could compromise patient safety and patient privacy. Motivated by our desire to improve patient safety, and mindful of conventional trade-offs between security and power consumption for resource-constrained devices, we introduce three new zero-power defenses based on RF power harvesting. Two of these defenses are human-centric, bringing patients into the loop with respect to the security and privacy of their implantable medical devices (IMDs). Our contributions provide a scientific baseline for understanding the potential security and privacy risks of current and future IMDs, and introduce human-perceptible and zero-power mitigation techniques that address those risks. To the best of our knowledge, this paper is the first in our community to use general-purpose software radios to analyze and attack previously unknown radio communications protocols.**

I. Introduction

Wirelessly reprogrammable *implantable medical devices* (*IMDs*) such as *pacemakers*, *implantable cardioverter defibrillators* (*ICDs*), *neurostimulators*, and implantable *drug pumps* use embedded computers and radios to monitor chronic disorders and treat patients with automatic therapies. For instance, an ICD that senses a rapid heartbeat can administer an electrical shock to restore a normal heart rhythm, then later report this event to a health care practitioner who uses a *commercial device programmer*[1] with wireless capabilities to extract data from the ICD or modify its settings without surgery. Between 1990 and 2002, over 2.6 million pacemakers and ICDs were implanted in patients in the United States [19]; clinical trials have shown that these devices significantly improve survival rates in certain populations [18]. Other research has discussed potential security and privacy risks of IMDs [1], [10], but we are unaware of any rigorous public investigation into the observable characteristics of a real commercial device. Without such a study, it is impossible for the research community to assess or address the security and privacy properties of past, current, and future devices. We address that gap in this paper and, based on our findings, propose and implement several prototype attack-mitigation techniques.

Our investigation was motivated by an interdisciplinary study of medical device safety and security, and relied on a diverse team of area specialists. Team members from the security and privacy community have formal training in computer science, computer engineering, and electrical engineering. One team member from the medical community is a practicing cardiologist with hundreds of pacemaker and implantable defibrillator patients and was past chairperson of the FDA's Circulatory System Medical Device Advisory Panel. Our technical contributions toward understanding and improving the security, privacy, and safety of these devices include: analyses; software radio-based methodologies; and human-perceptible and zero-power (battery-free) defenses.

Overview of contributions. We assess the security and privacy properties of a common ICD and present attacks on privacy, integrity, and availability. We show that the ICD discloses sensitive information in the clear (unencrypted); we demonstrate a *reprogramming* attack that changes the operation of (and the information contained in) the ICD; and

*Corresponding faculty authors:
- Kevin Fu, Medical Device Security Center, Department of Computer Science, University of Massachusetts Amherst, 140 Governors Drive, Amherst, Massachusetts 01003 (kevinfu@cs.umass.edu);
- Tadayoshi Kohno, Medical Device Security Center, Department of Computer Science and Engineering, University of Washington, Box 352350, Seattle, Washington 98195 (yoshi@cs.washington.edu);
- William H. Maisel, Medical Device Safety Institute, Beth Israel Deaconess Medical Center, Harvard Medical School, 185 Pilgrim Road, Baker 4, Boston, MA 02215 (wmaisel@bidmc.harvard.edu).

Additional information online at http://www.secure-medicine.org.
[†]Co-student leads listed in alphabetical order; each participated equally.

[1]The reader should not confuse the term "device programmer" with a person who programs computers. The former is an external device that communicates with and adjusts the settings on an IMD.

978-0-7695-3168-7/08 $25.00 © 2008 IEEE 129
DOI 10.1109/SP.2008.31

SECURITY

Lessons from Stuxnet

Thomas M. Chen, *Swansea University*
Saeed Abu-Nimeh, *Damballa Inc.*

Malware such as Stuxnet can affect critical physical infrastructures that are controlled by software, which implies that threats might extend to real lives.

Thousands of new malware appear in the wild daily. Most are evolutionary variants of existing families and don't have a widespread impact. However, occasionally a noteworthy new piece of malware will change the security landscape. For example, the 1988 Morris attack showed that an aggressive worm could bring down a substantial part of the Arpanet, and the 2003 SQL Slammer attack demonstrated that a simple user datagram protocol (UDP)-based worm could create devastating network congestion.

Stuxnet is teaching the security community new lessons. Since VirusBlokAda discovered the Windows worm in Belarus in July 2010, researchers have studied it intensely. They believe Stuxnet spread for several months before discovery and that it has already compromised its intended target.

As Table 1 shows, Stuxnet differs from past malware in several ways. First, most malware tries to infect as many computers as possible, whereas Stuxnet appears to target industrial control systems and delivers its payload under very specific conditions. Second, Stuxnet is larger and more complex than other malware. It contains exploits for four unpatched vulnerabilities—an unusually high number. The code is approximately 500 Kbytes and written in multiple languages. As a reference, the SQL Slammer worm was 376 bytes; the Code Red worm was approximately 4 Kbytes; the Nimda worm was 60 Kbytes; and variants of the Zeus banking Trojan ranged between 40 and 150 Kbytes. Virtually all malware is less than 1 Mbyte.

Based on Stuxnet's code, experts have speculated on its creators and intention. Its sophistication suggests that the creators had detailed knowledge of its target and access to immense resources, perhaps with government backing. Its choice of targets also suggests a political motive.

TARGET SELECTION

Unlike most malware, Stuxnet targets industrial control systems, which are used widely in factories, assembly lines, refineries, and power plants. It attacks Windows PCs that program specific Siemens programmable logic controllers—specialized computers that control automated physical processes, such as robot arms, in common industrial control systems. PLCs can have elaborate input/output arrangements for various applications in different physical environments. They often have sensors on the inputs (for example, for temperature), and the outputs typically operate equipment such as motors, switches, and relays.

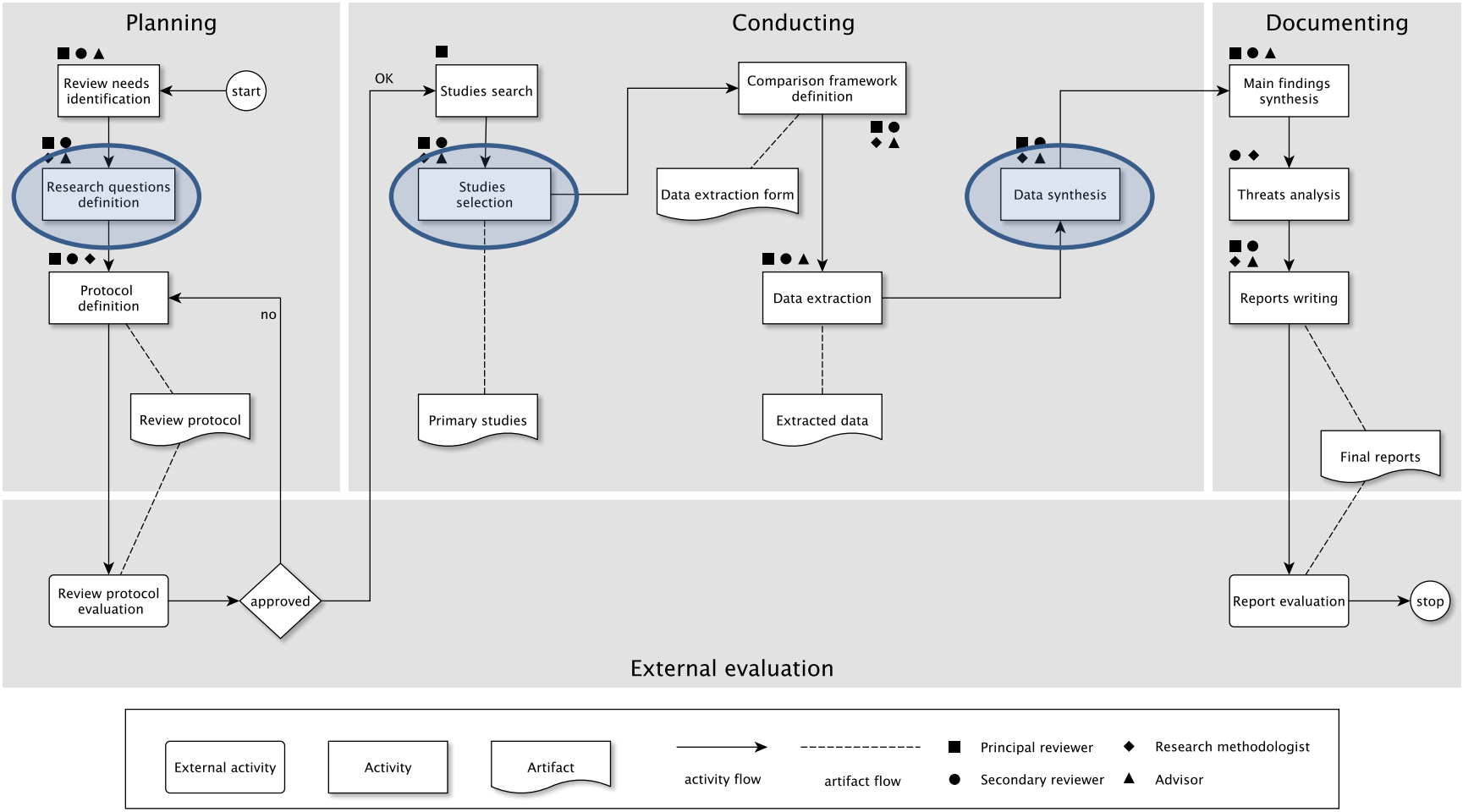
Stuxnet targets vulnerable PCs running WinCCiStep 7 control software, which is normally used to program PLCs. When an infected PC connects to a Siemens Simatic PLC, Stuxnet installs a malicious .dll file, replacing the PLC's original .dll file. The malicious .dll file lets Stuxnet monitor and intercept all communication

Table 1. Stuxnet's novel characteristics.

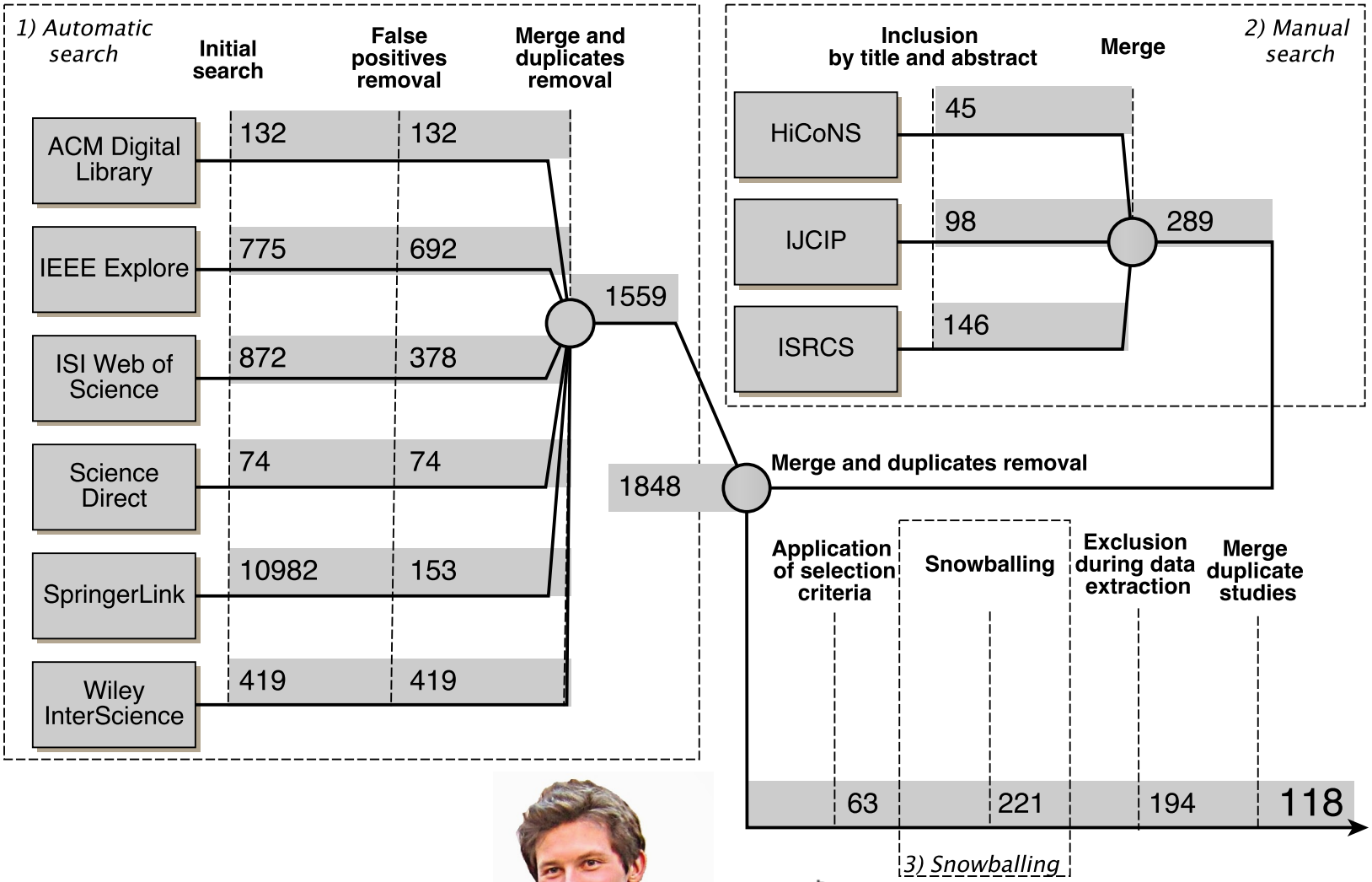
Aspect	Stuxnet	Common malware
Targeting	Extremely selective	Indiscriminate
Type of target	Industrial control systems	Computers
Size	500 Kbytes	Less than 1 Mbyte
Probable initial infection vector	Removable flash drive	Internet and other networks
Exploits	Four zero-days	Possibly one zero-day

0018-9162/11/$26.00 © 2011 IEEE Published by the IEEE Computer Society APRIL 2011

Overview of the whole review process

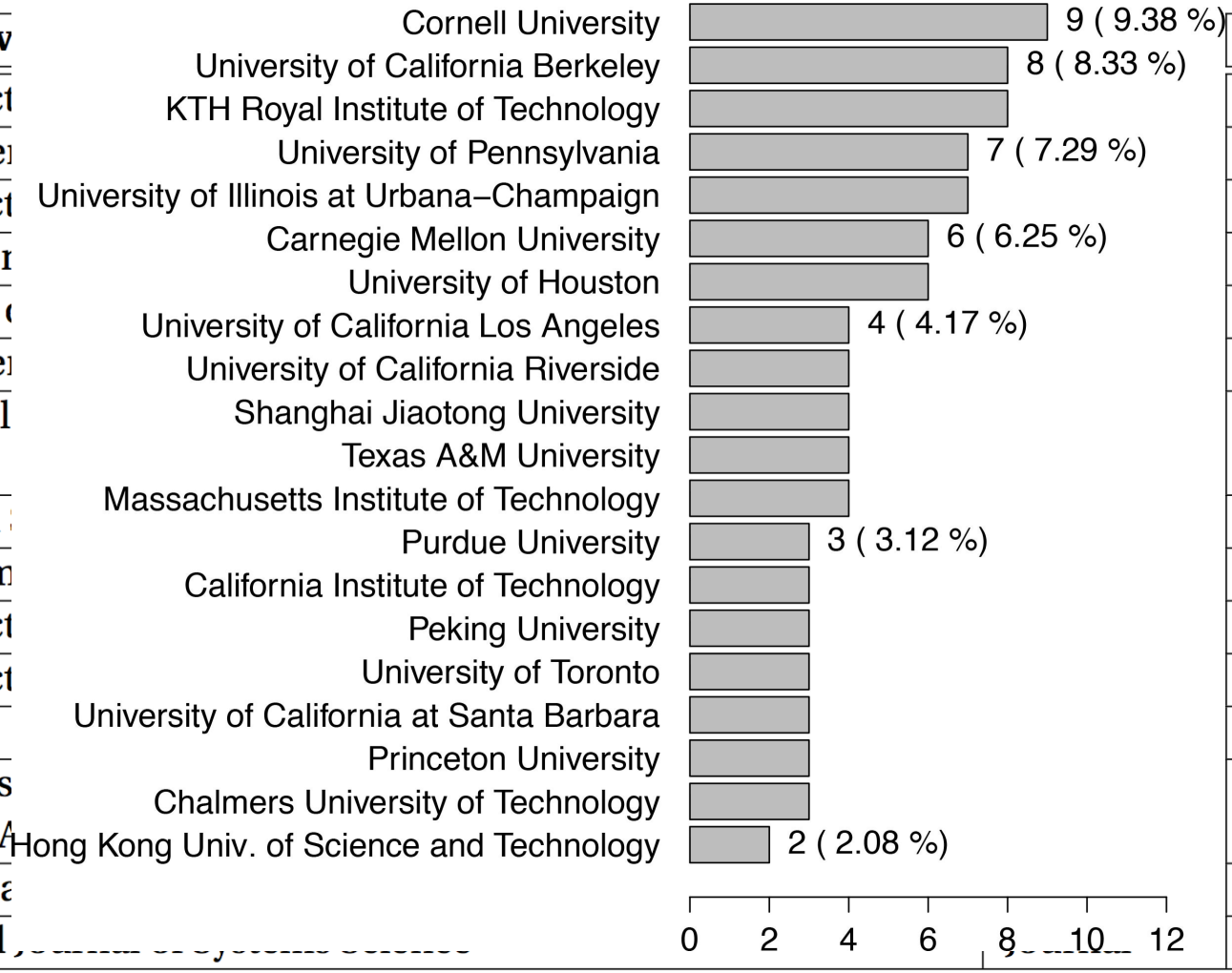


Search strategy



Publication trends

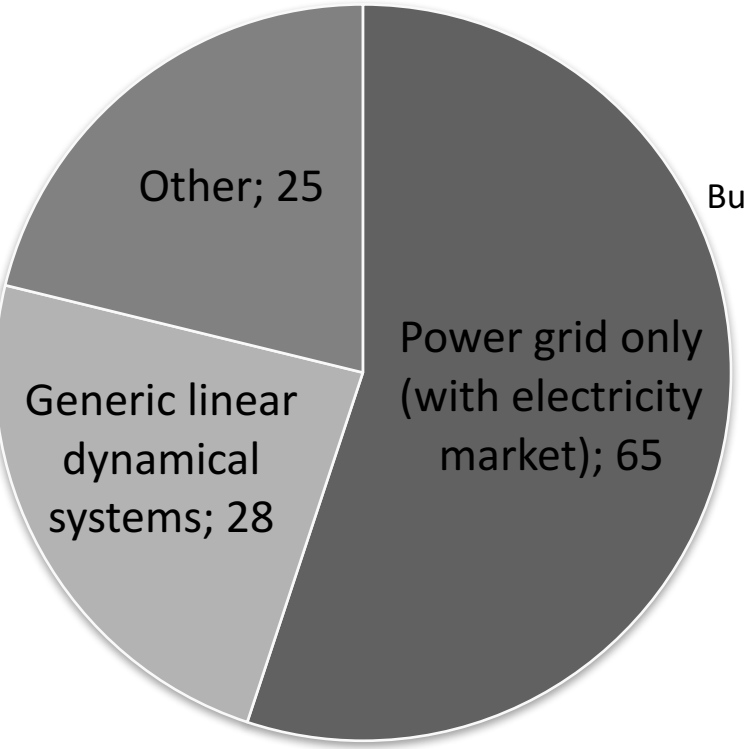
Distribution of primary studies over the years*

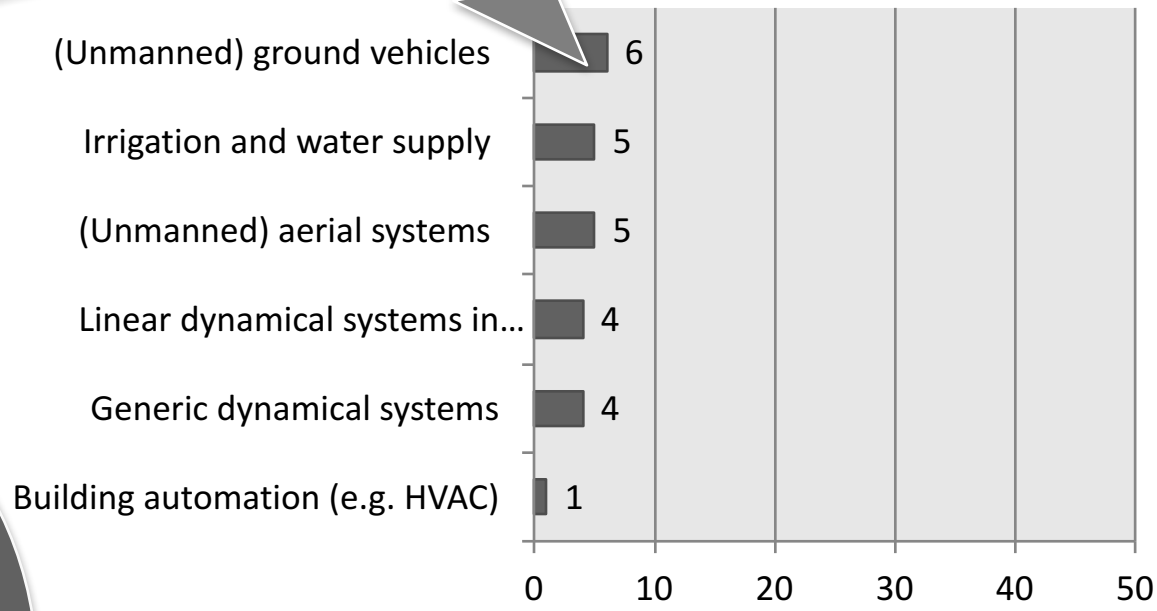


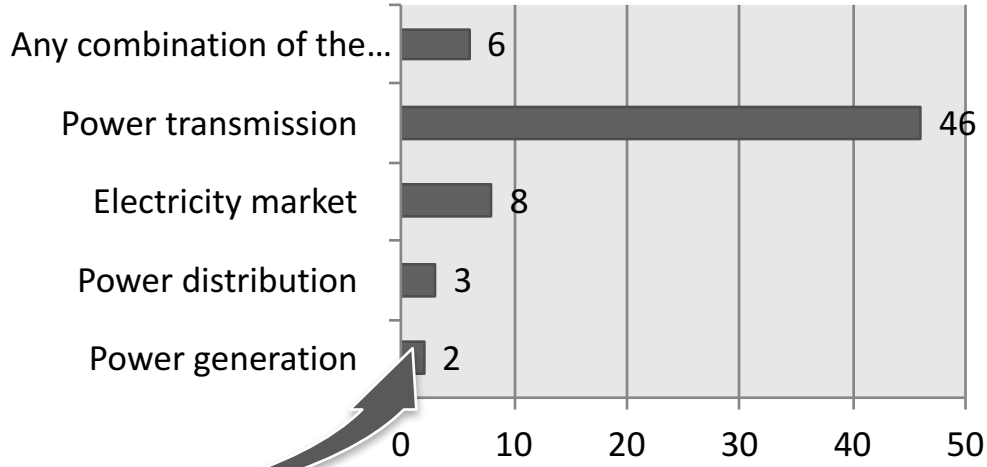
Publication v		#Studies
IEEE Transact		19 (16.10%)
IEEE Confere		11 (9.32%)
IEEE Transact		9 (7.62%)
American Co		6 (5.08%)
IEEE Journal		6 (5.08%)
IEEE Confere		6 (5.08%)
International (HiCoNS)		4(3.38%)
IEEE Control		3 (2.54%)
Global Comm		3 (2.54%)
IEEE Transact		3 (2.54%)
IEEE Transact		3 (2.54%)
Automatica		2 (1.69%)
ACM Sympos Security (ASIA		2 (1.69%)
Cyber Physica		2 (1.69%)
International		2 (1.69%)
TOTAL	-	**81 (68.64%)**

(*) partial data for 2015

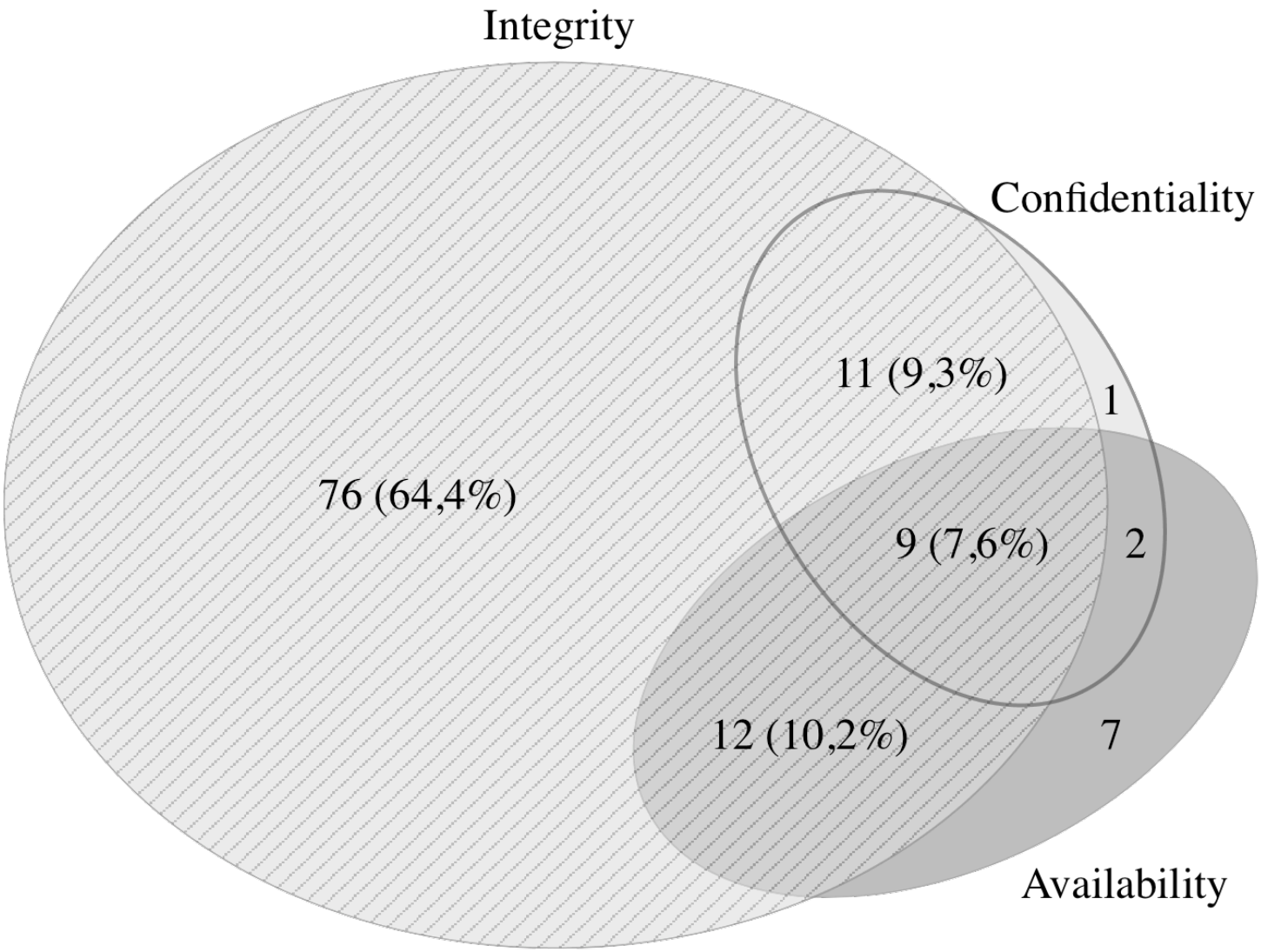
Characteristics and focus of research: application fields







Characteristics and focus of research: security attributes

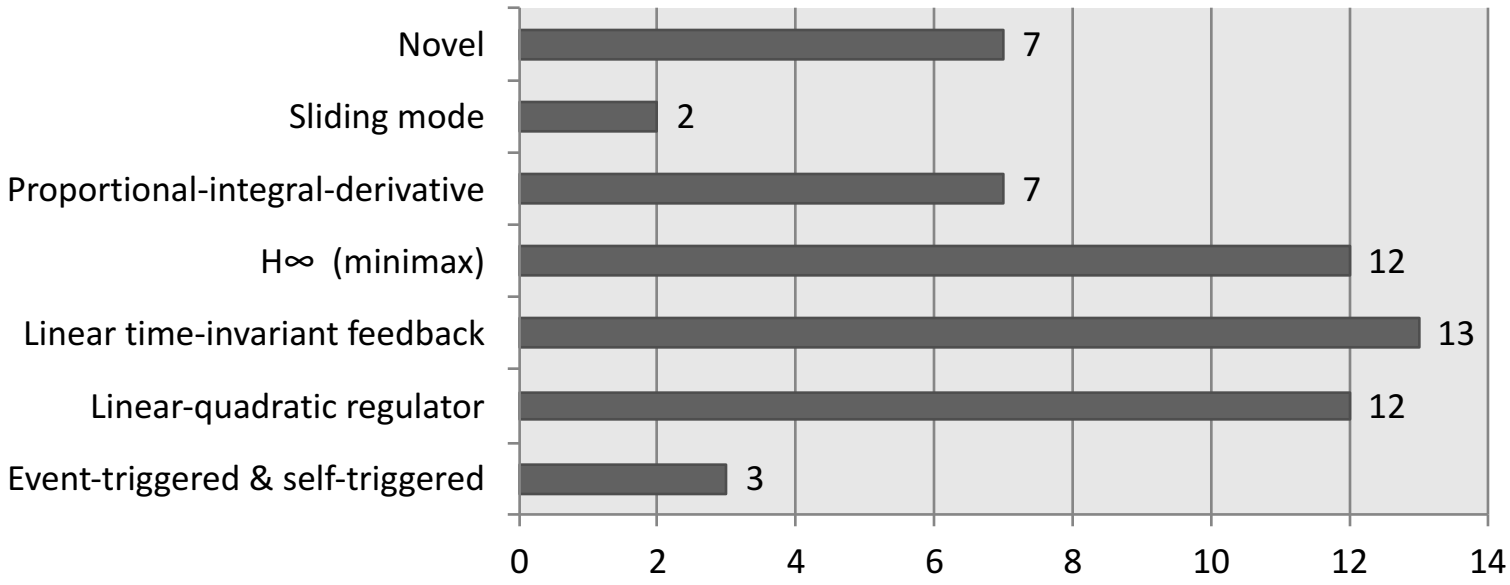


Characteristics and focus of research: controllers

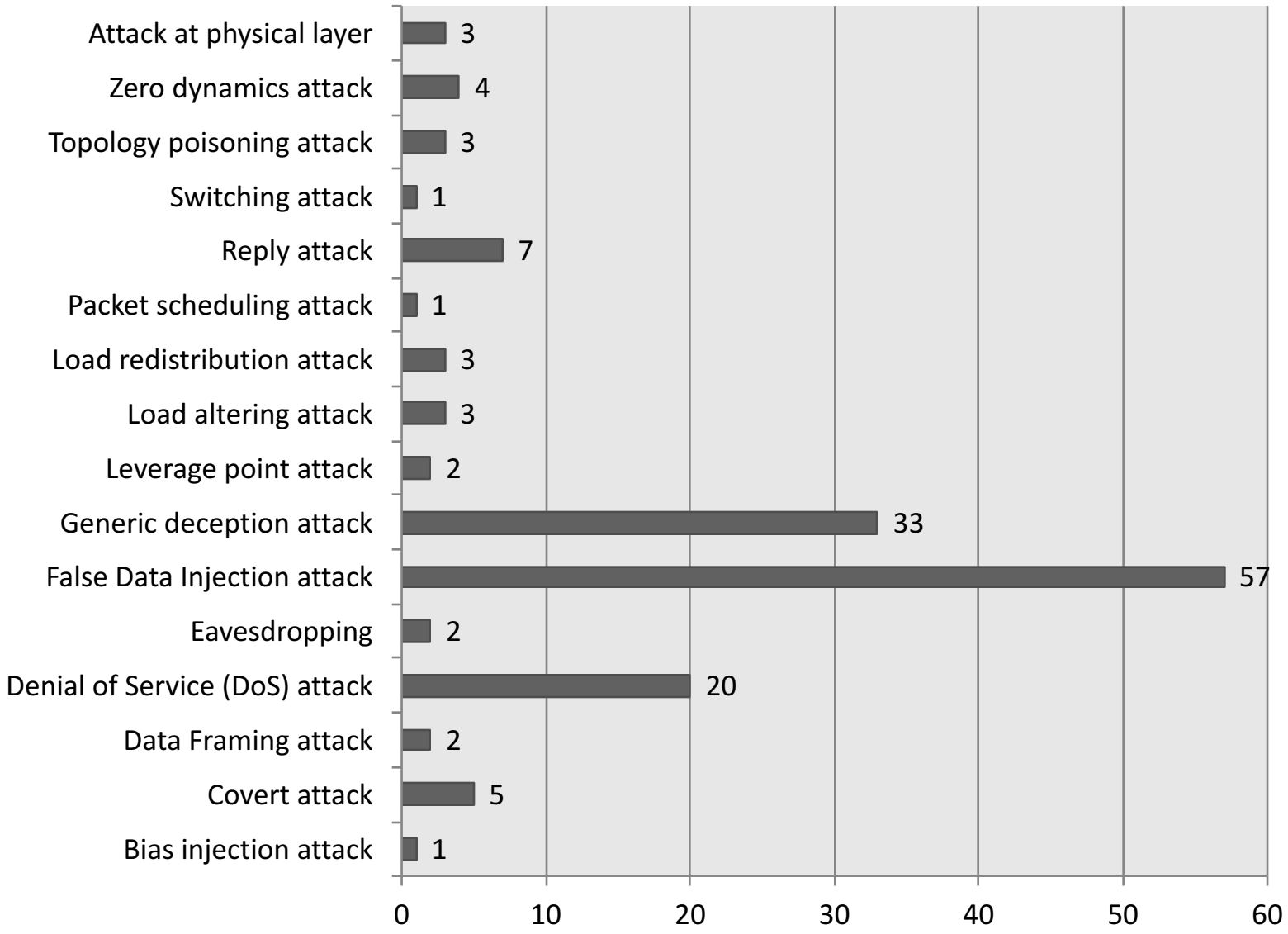
Studies focusing on state estimation usually do not examine at all the controller.

In fact, in 82 (69,49% of 118 selected) studies the controller is not available.

Some of the remaining 36 studies consider more than one controller at once.

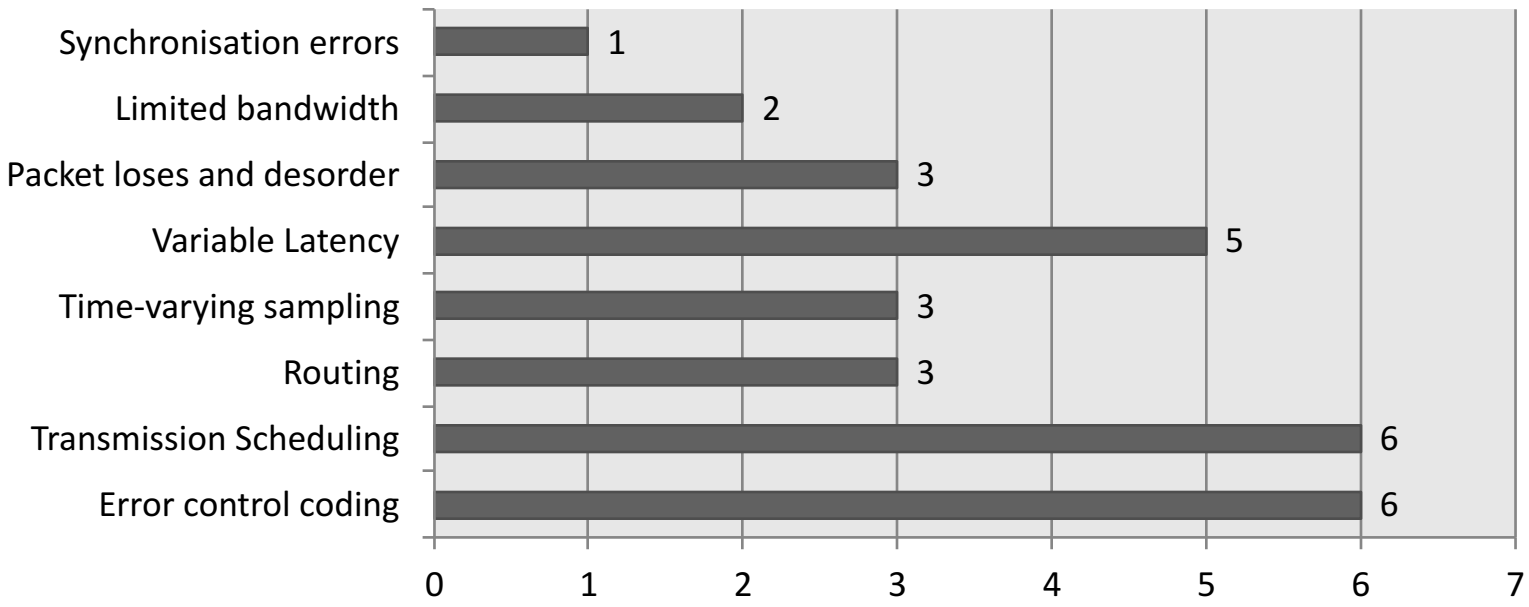


Attacks and their characteristics



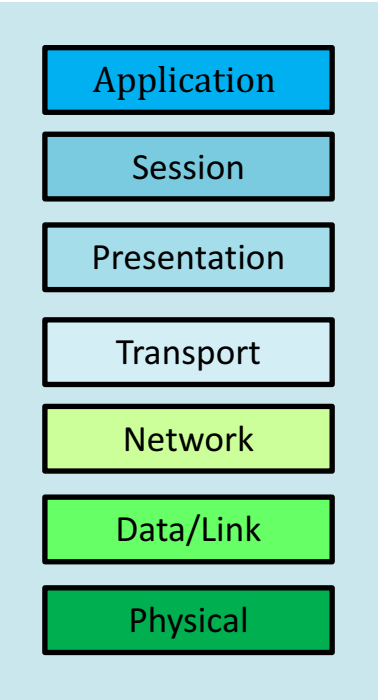
Communication aspects and network-induced imperfections

Surprisingly, 100 out of 118 studies (i.e., 84,75%) do not explicitly consider any communication aspect or imperfection, while only 6 studies (i.e. 5,08%) address more than one aspect.



Surprisingly, very few papers (attempt to) provide non-trivial mathematical models of the communication protocol, which indeed is a fundamental actor of almost any CPS. In particular, only in 2 works a specific standard for communication is explicitly considered in the CPS mathematical model.

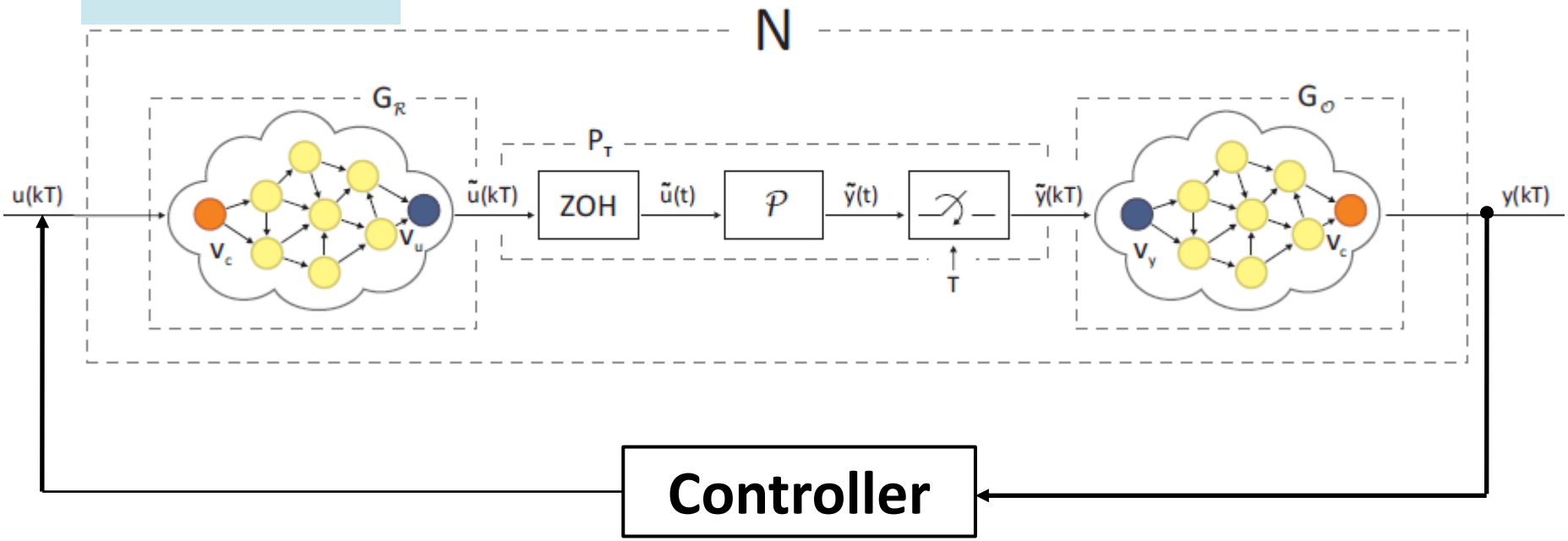
Co-design over time-triggered communication protocols



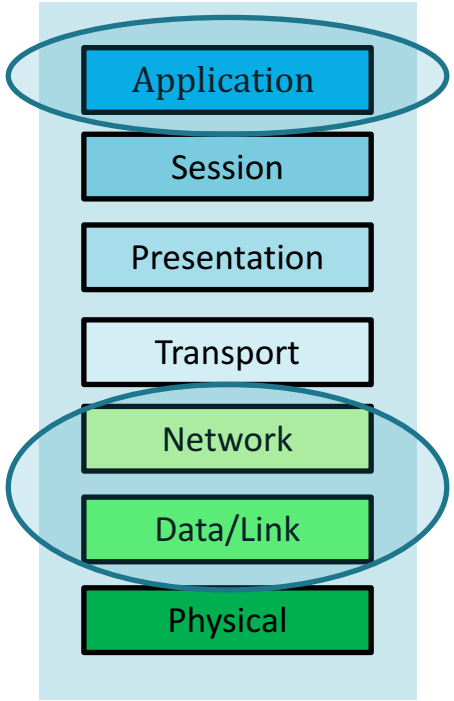
Challenge: Co-design the control algorithm and the communication protocol

WirelessHART

ISA100Wireless



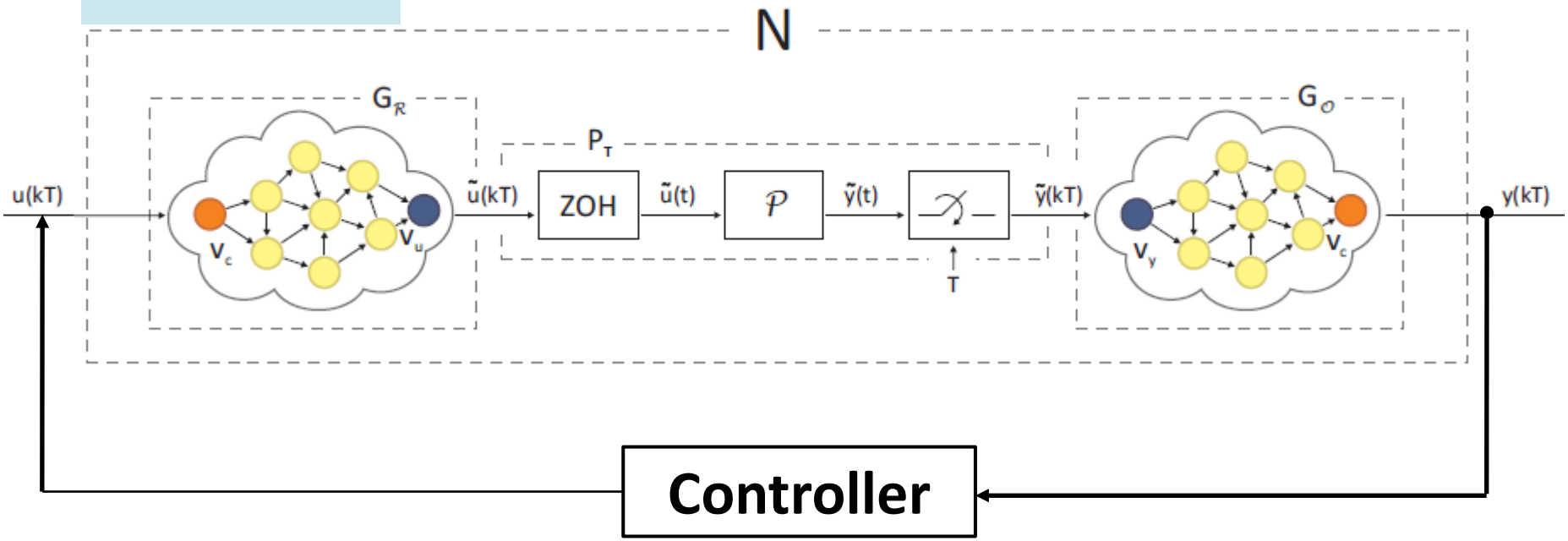
Co-design over time-triggered communication protocols



Challenge: Co-design the control algorithm and the communication protocol (scheduling, routing and control)

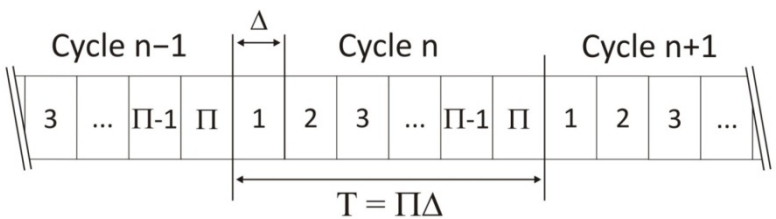
WirelessHART

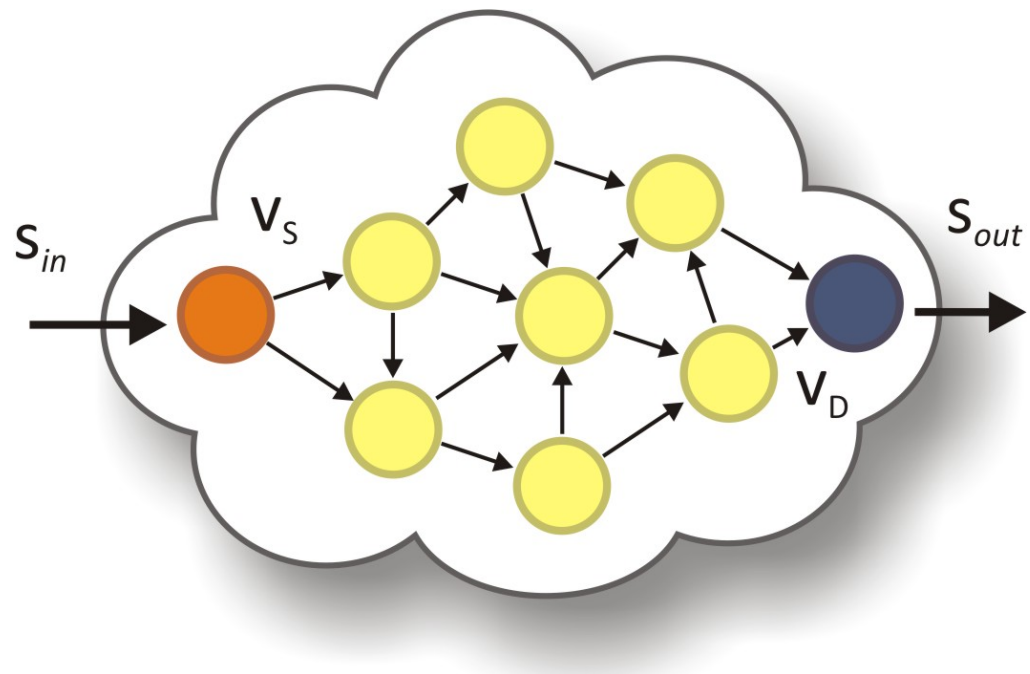
ISA100Wireless



WirelessHART MAC (*scheduling*) and Network (*routing*) layers

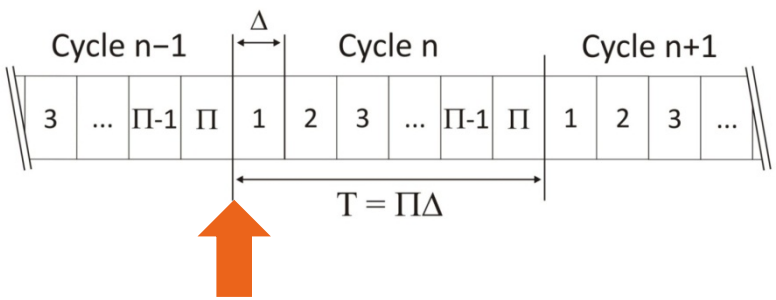
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ

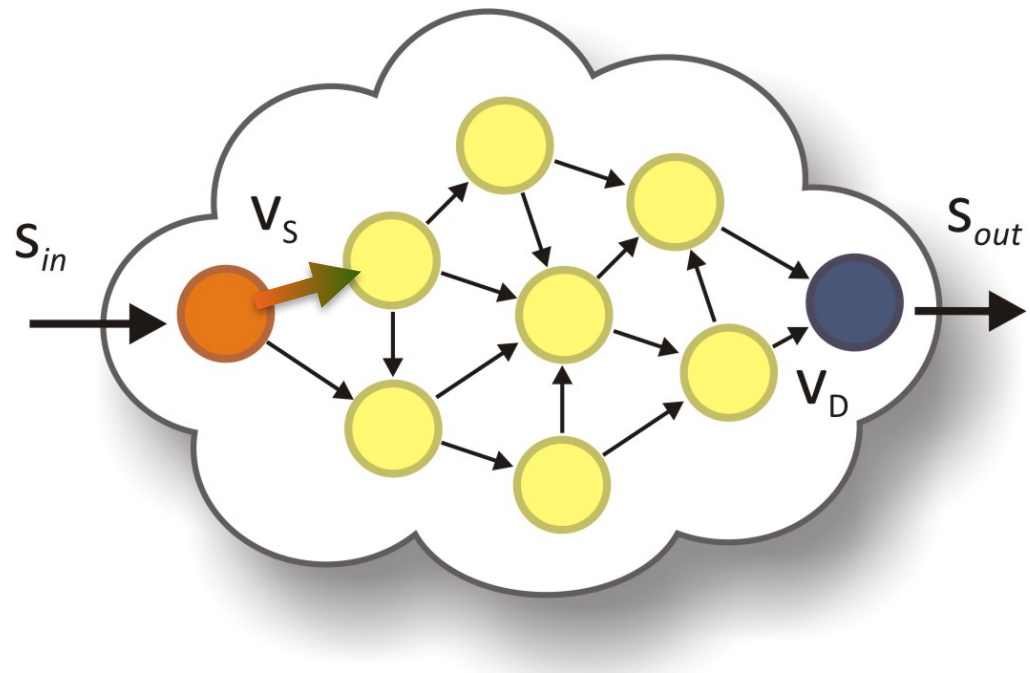




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

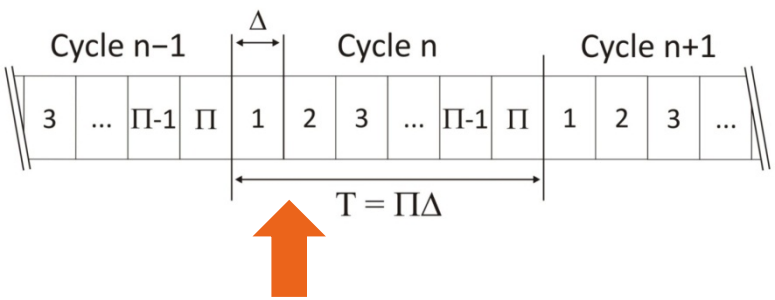
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ

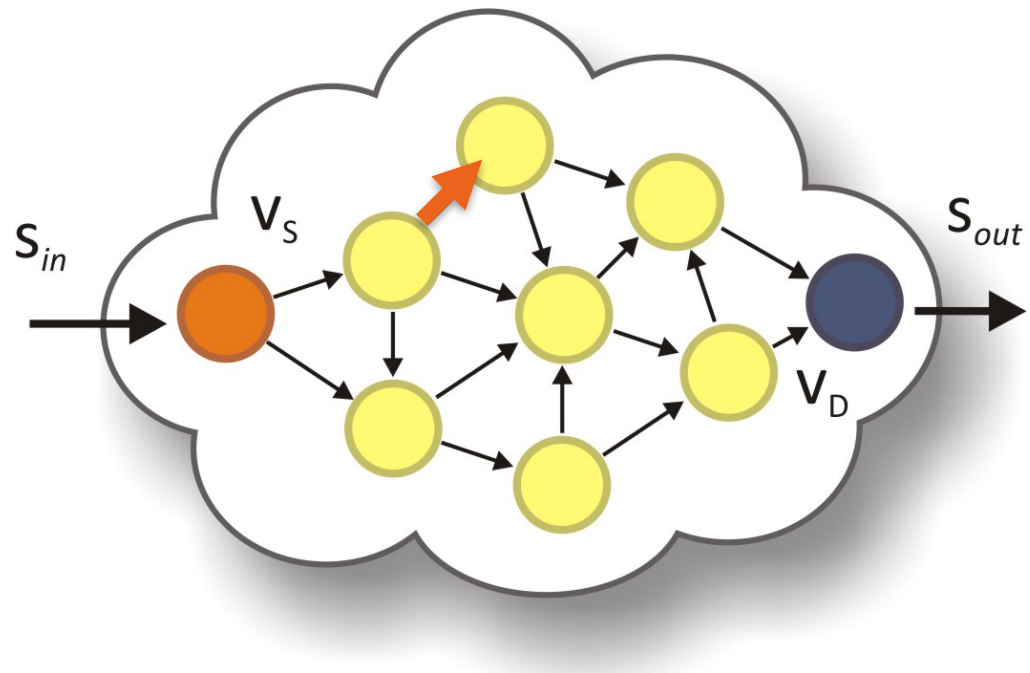




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

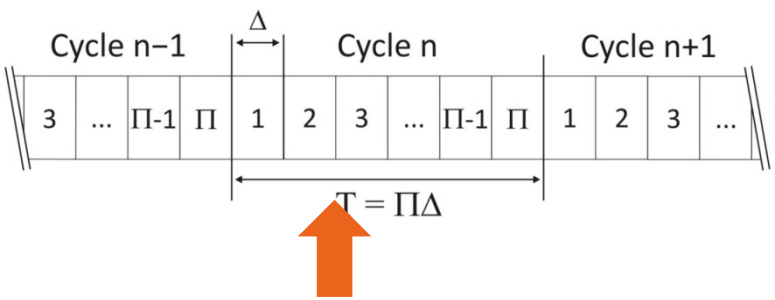
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ
- Enables redundancy in data routing

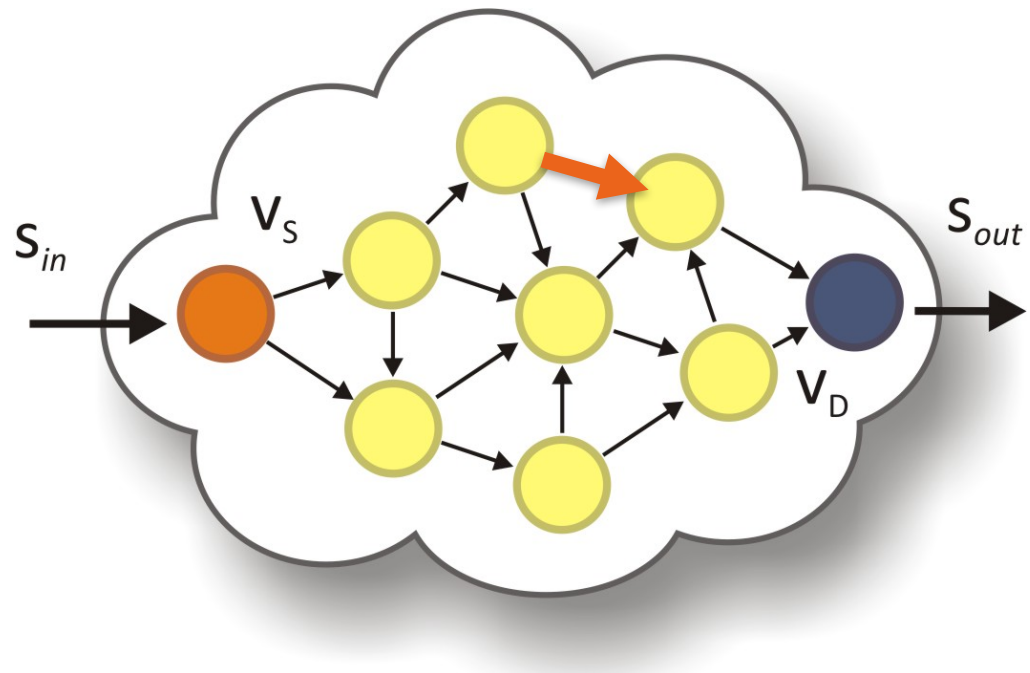




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

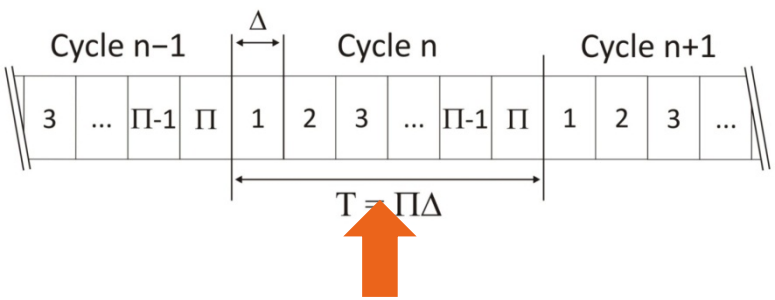
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ
- Enables redundancy in data routing

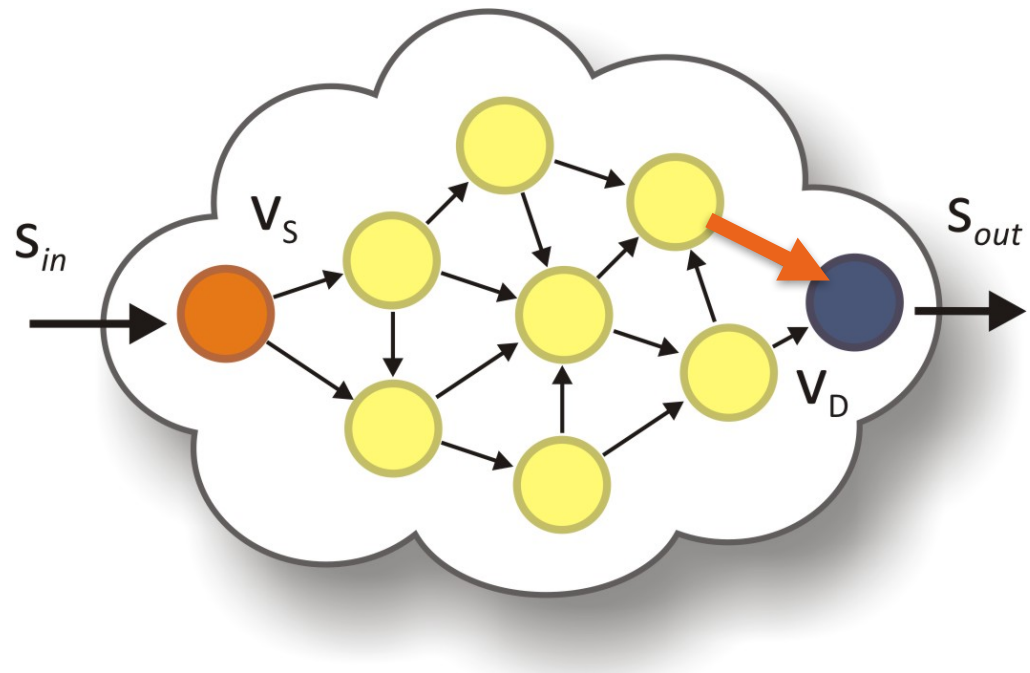




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

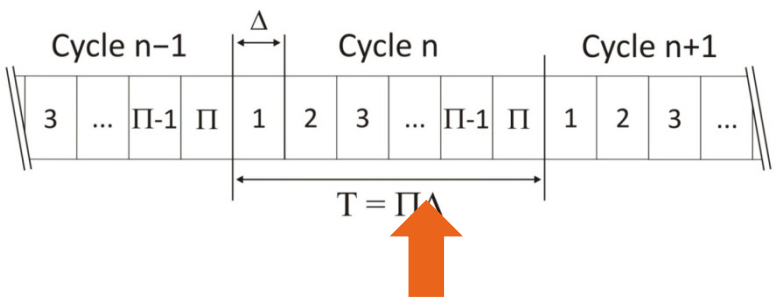
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ
- Enables redundancy in data routing

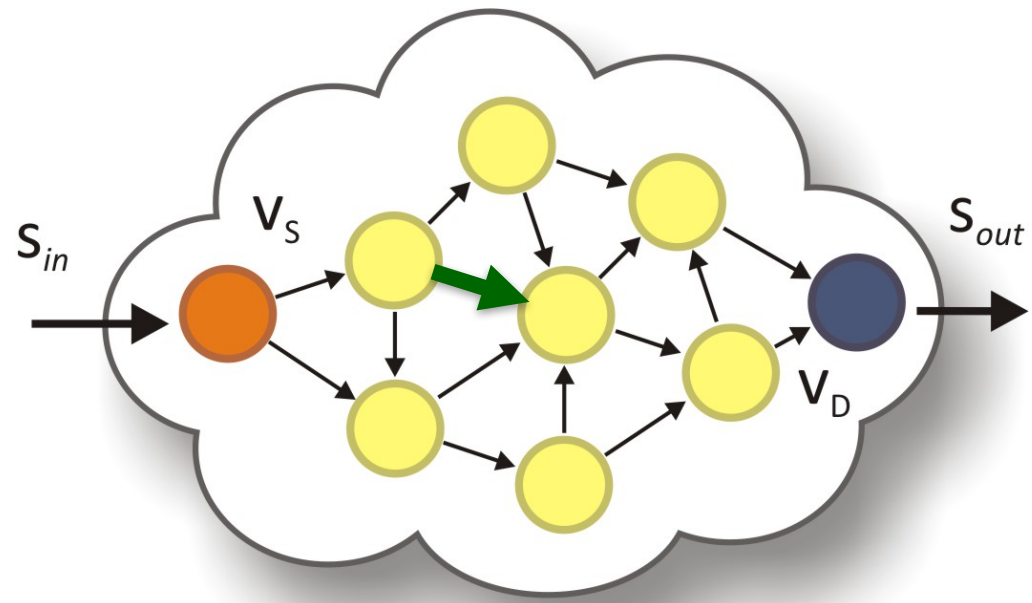




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

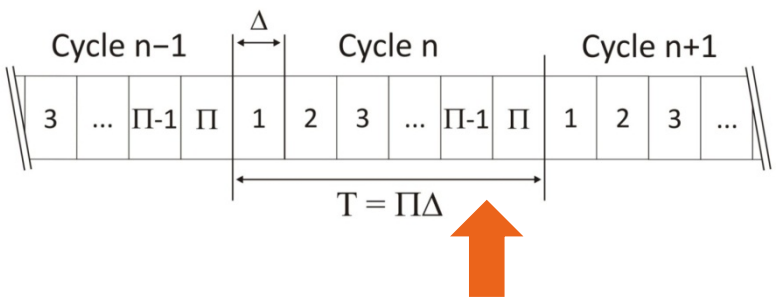
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ
- Enables redundancy in data routing
- Scheduling must guarantee relay via multiple paths

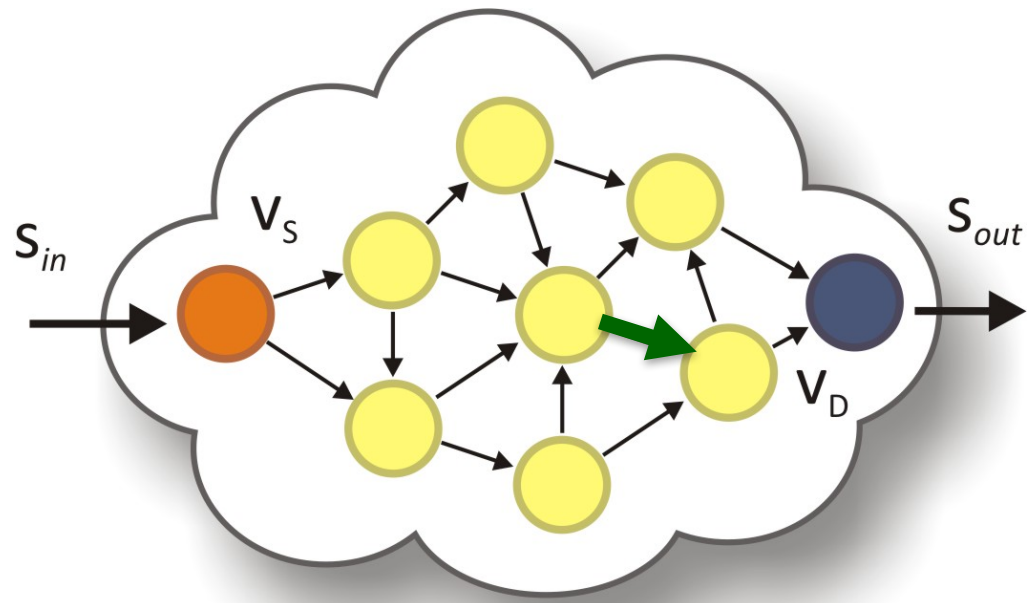




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

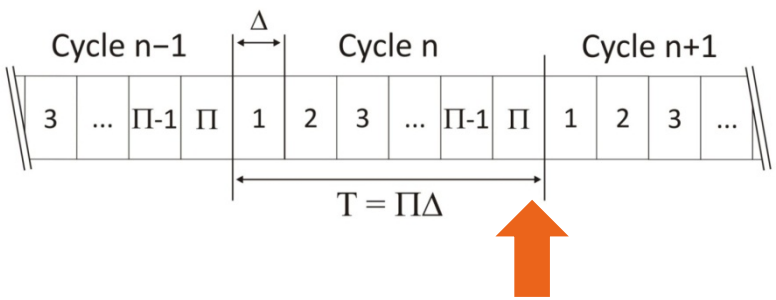
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ
- Enables redundancy in data routing
- Scheduling must guarantee relay via multiple paths

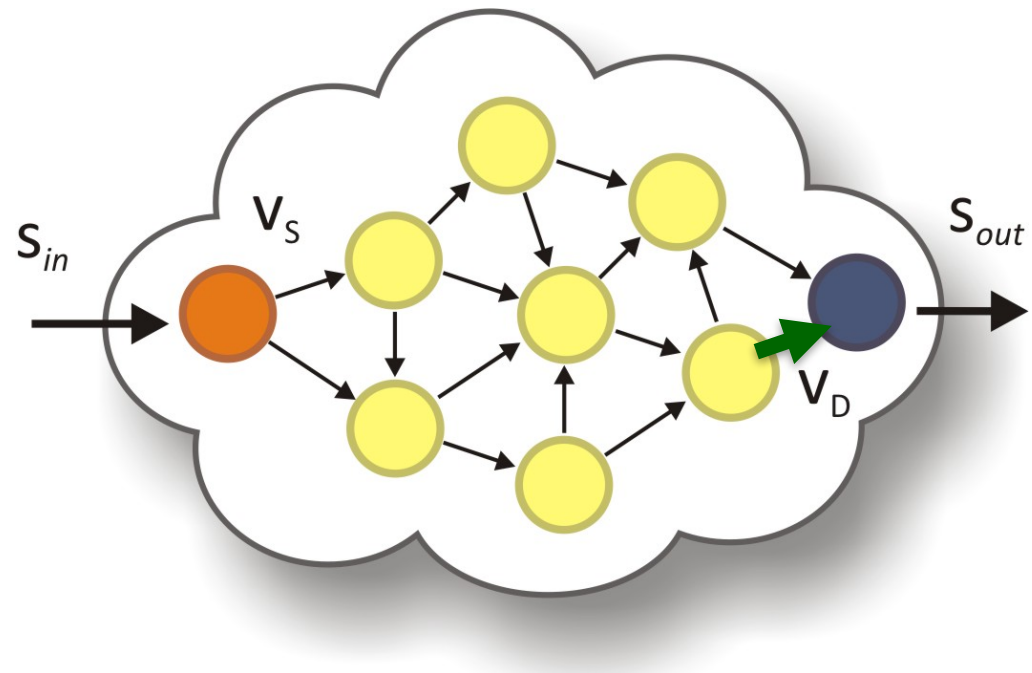




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

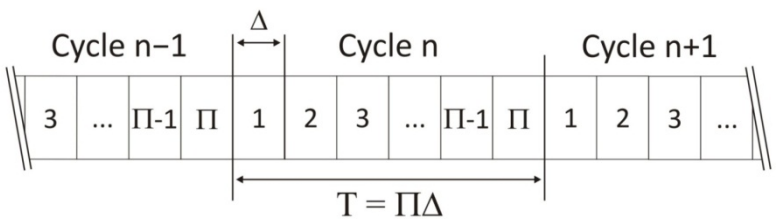
- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ
- Enables redundancy in data routing
- Scheduling must guarantee relay via multiple paths

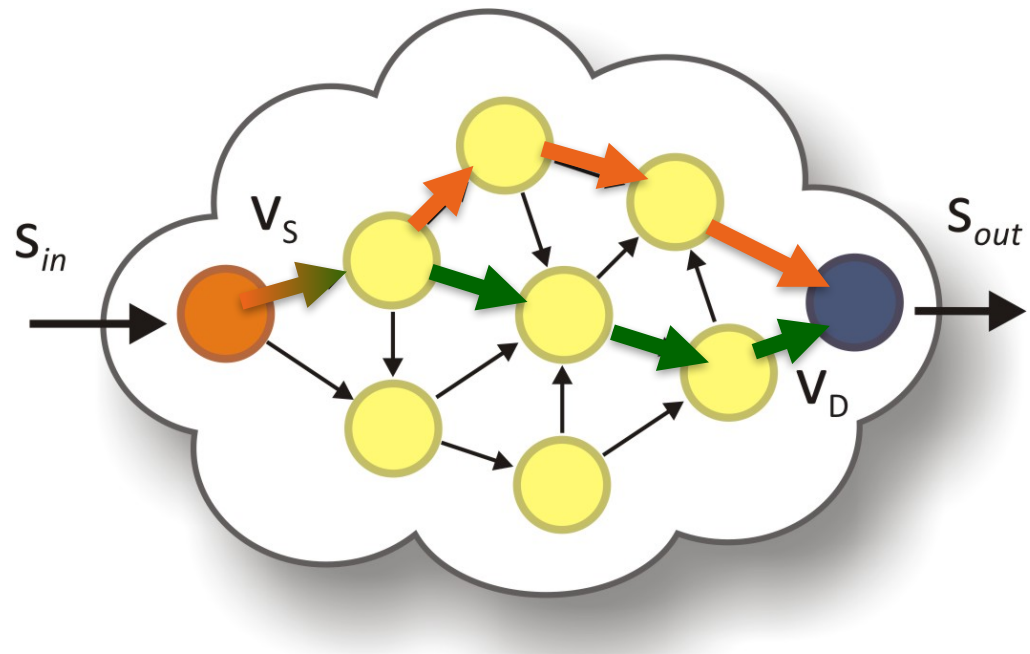




WirelessHART MAC (*scheduling*) and Network (*routing*) layers

- Time-triggered access to the channel
- Time divided in periodic frames
- Each frame divided in Π time slots of duration Δ
- Enables redundancy in data routing
- Scheduling must guarantee relay via multiple paths

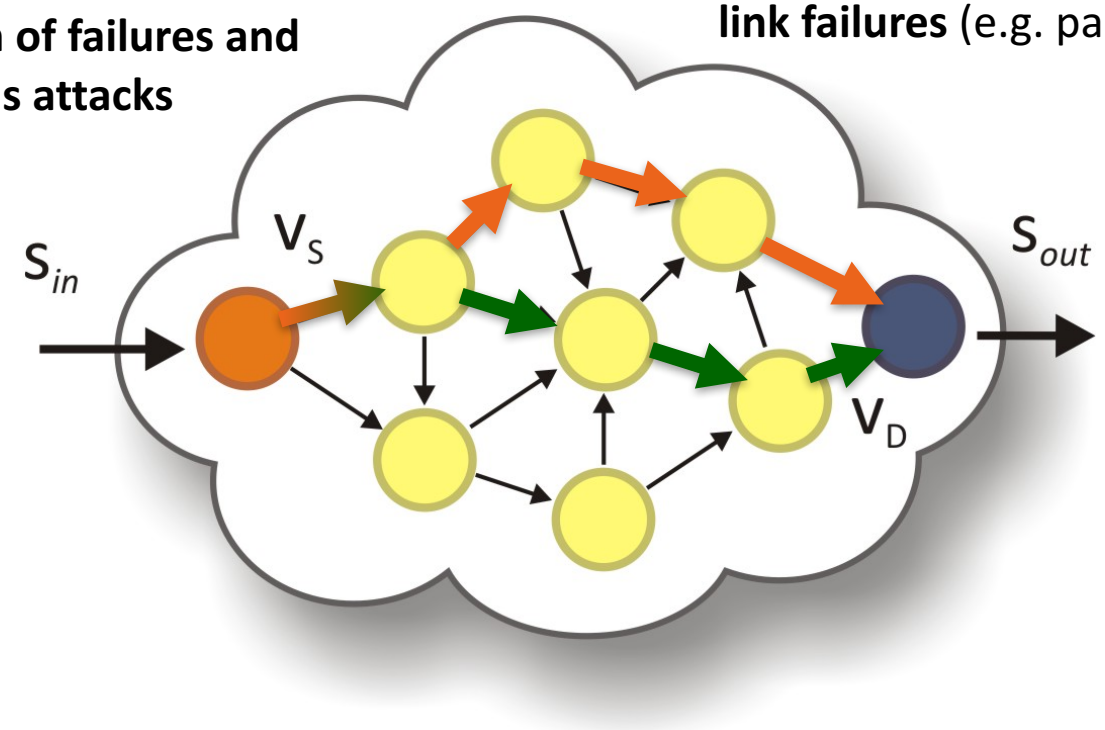




How to exploit redundancy optimally w.r.t. control performance metrics?

Redundancy in data routing…

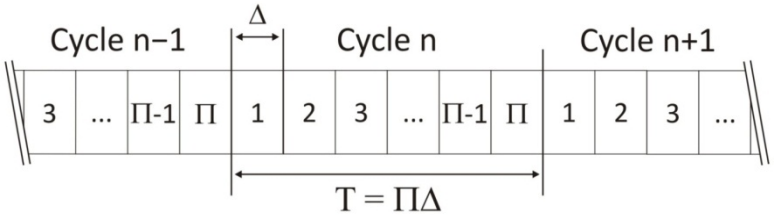
- …enables **detection and isolation of failures and malicious attacks**
- …makes system **robust to short-term link failures** (e.g. packet losses)

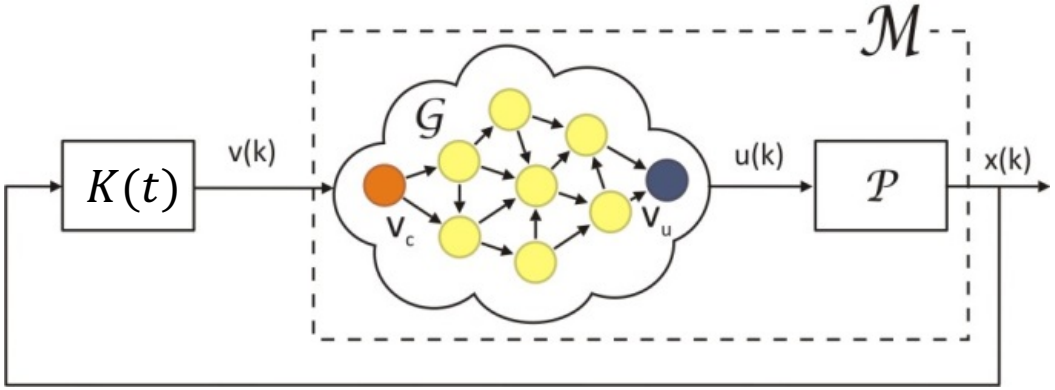


- …makes system **tolerant to long-term link failures**

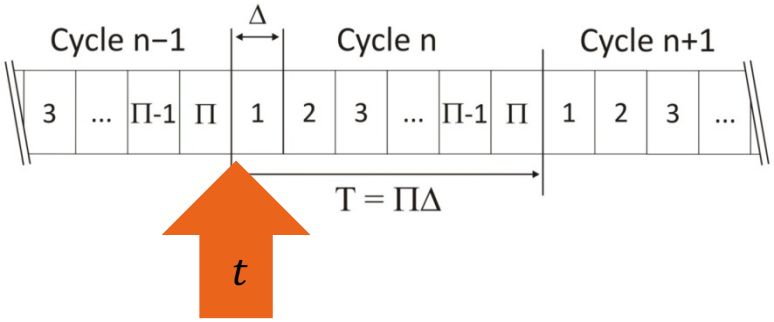
1. Separation of concerns
2. Co-design

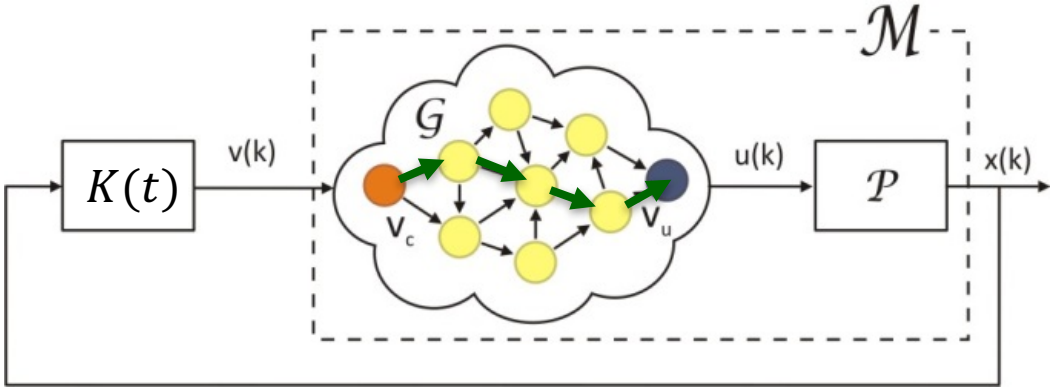
Wireless control networks as switching systems



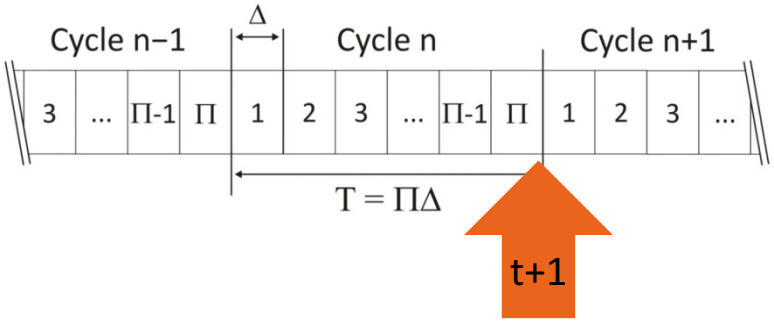


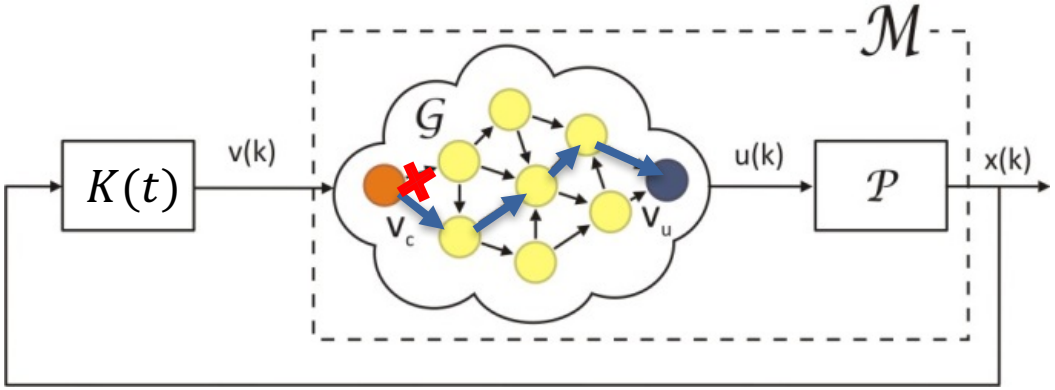
Wireless control networks as switching systems



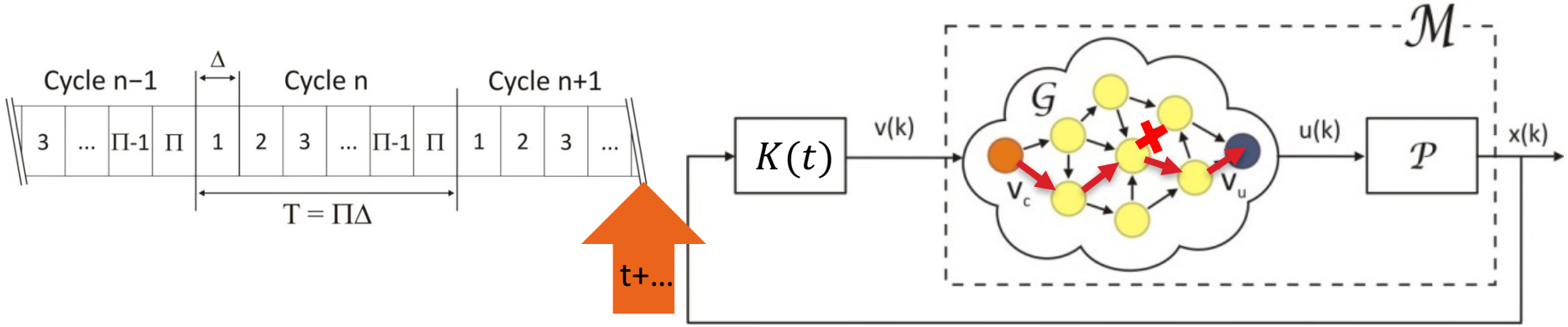


Wireless control networks as switching systems



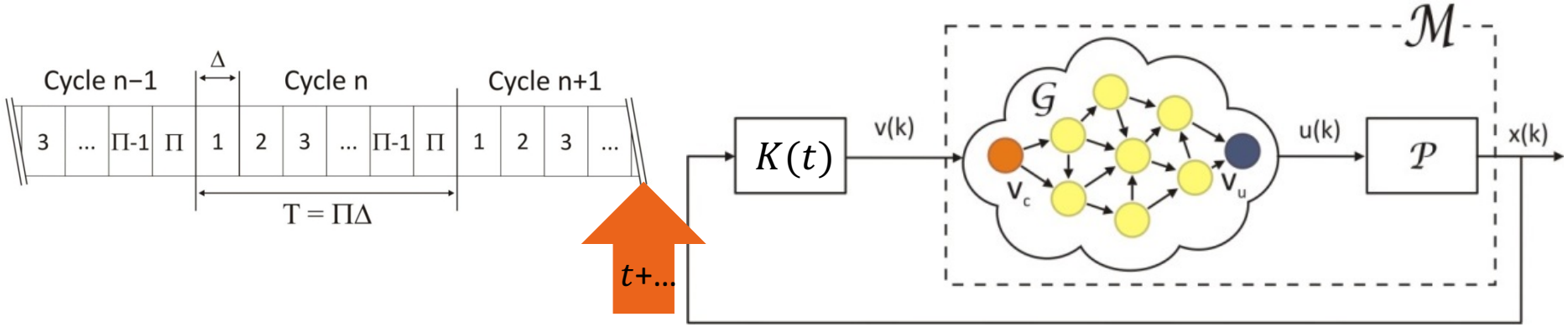


Wireless control networks as switching systems



Different paths are associated with different delays.

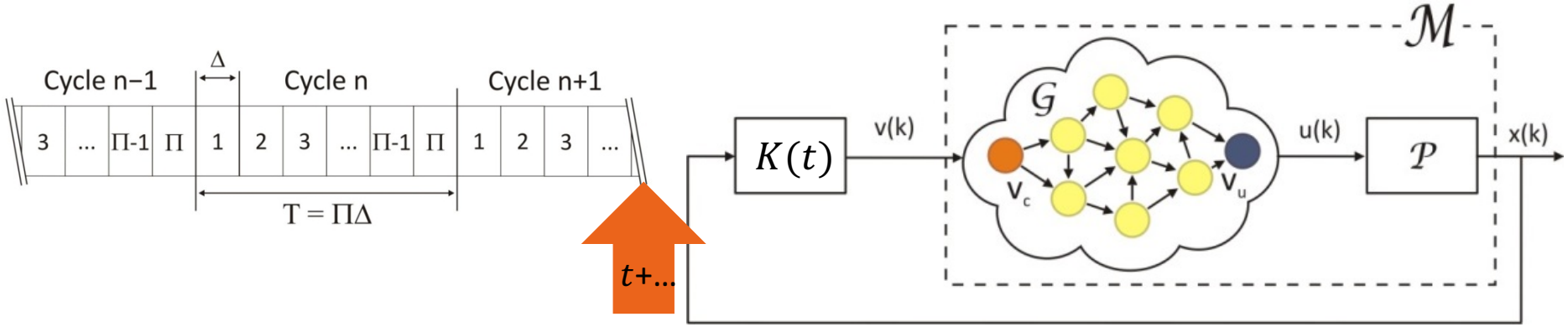
Wireless control networks as switching systems



Different paths are associated with different delays.

Mathematical model: $x(t+1) = Ax(t) + B\big(\sigma(t)\big)v(t), t \in \mathbb{N}$, where $x(t)$ is the plant and network state, $\sigma(t) \in \Sigma$ depends on routing/scheduling. The switching signal is considered as a disturbance.

Wireless control networks as switching systems

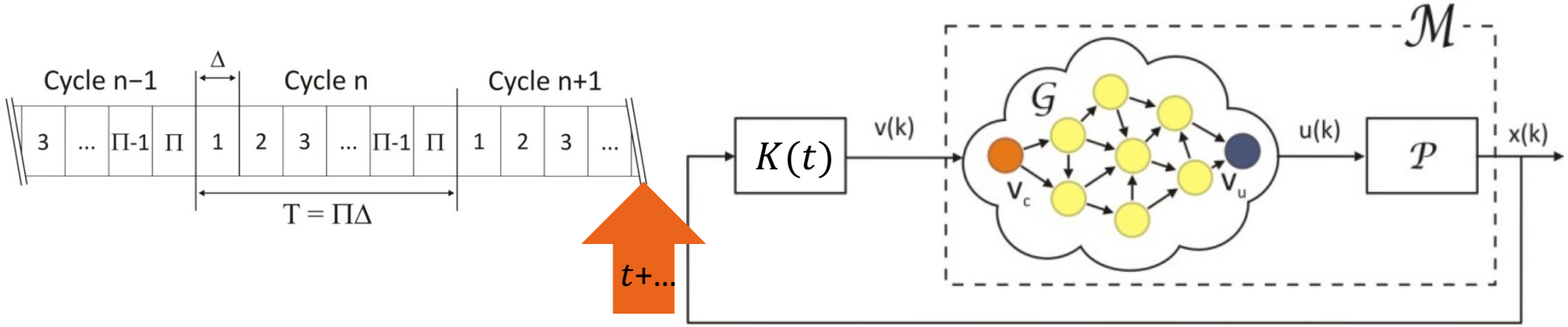


Different paths are associated with different delays.

Mathematical model: $x(t+1) = Ax(t) + B\big(\sigma(t)\big)v(t), t \in \mathbb{N}$, where $x(t)$ is the plant and network state, $\sigma(t) \in \Sigma$ depends on routing/scheduling. The switching signal is considered as a disturbance.

$$A = \begin{pmatrix} A_P & B_P & 0 & \cdots & 0 & 0 \\ 0 & 0 & I & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & I & 0 \\ 0 & 0 & 0 & \cdots & 0 & I \\ 0 & 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \qquad B\big(\sigma(t)\big) = \begin{pmatrix} B\delta_{\sigma(t),0} \\ I\delta_{\sigma(t),1} \\ \vdots \\ I\delta_{\sigma(t),D-2} \\ I\delta_{\sigma(t),D-1} \\ I\delta_{\sigma(t),D} \end{pmatrix}$$

Wireless control networks as switching systems



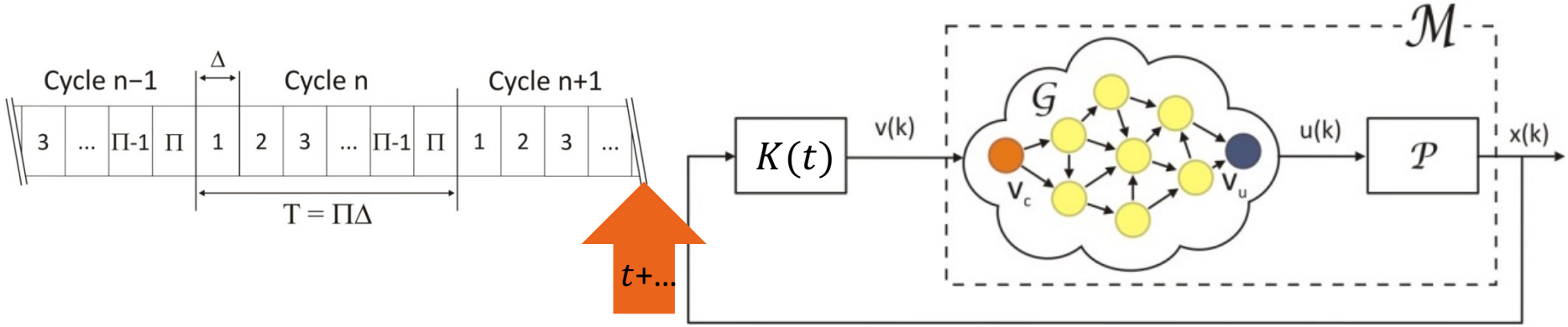
Different paths are associated with different delays.

Mathematical model: $x(t+1) = Ax(t) + B\big(\sigma(t)\big)v(t), t \in \mathbb{N}$, where $x(t)$ is the plant and network state, $\sigma(t) \in \Sigma$ depends on routing/scheduling. The switching signal is considered as a disturbance.

Problem: Design a controller $K(t)$ s.t. the closed loop system is asymptotically stable.

Given a state-feedback controller $K(t)$**,** the closed loop systems is asymptotically stable iff the **Joint Spectral Radius** of $\{A + B\big(\sigma(t)\big)K(t)\}_{\sigma(t)\in\Sigma}$ is smaller than 1.

Wireless control networks as switching systems



Different paths are associated with different delays.

Mathematical model: $x(t+1) = Ax(t) + B\big(\sigma(t)\big)v(t), t \in \mathbb{N}$, where $x(t)$ is the plant and network state, $\sigma(t) \in \Sigma$ depends on routing/scheduling. The switching signal is considered as a disturbance.

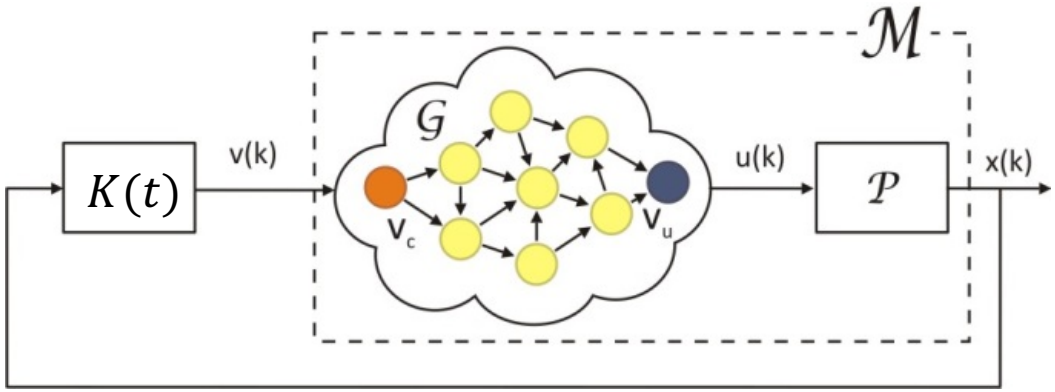
Problem: Design a controller $K(t)$ s.t. the closed loop system is asymptotically stable.

Given a state-feedback controller $K(t)$**,** the closed loop systems is asymptotically stable iff the **Joint Spectral Radius** of $\{A + B\big(\sigma(t)\big)K(t)\}_{\sigma(t)\in\Sigma}$ is smaller than 1.

Insights: Switching systems analysis and design is a crowded research area:

- Leverage special structure of matrices A and $B\big(\sigma(t)\big)$ to provide tailored results that outperform classical results on general switching systems
- Decidability results on controllability and stabilizability depend on knowledge of $\sigma(t)$

References



R. M. Jungers, A. D'Innocenzo, M. D. Di Benedetto. Controllability of Linear Systems With Switching Delays. IEEE Transactions on Automatic Control, 61(4):1117-1122, 2016.

A. Cicone, A. D'Innocenzo, N. Guglielmi, L. Laglia. A sub-optimal solution for optimal control of linear systems with unmeasurable switching delays. 54th IEEE Conference on Decision and Control, Osaka, Japan, December 15-18, 2015.

R. M. Jungers, A. D'Innocenzo, M. D. Di Benedetto. Further results on controllability of linear systems with switching delays. 9th IFAC World Congress, Cape Town, South Africa, August 24-29, 2014.

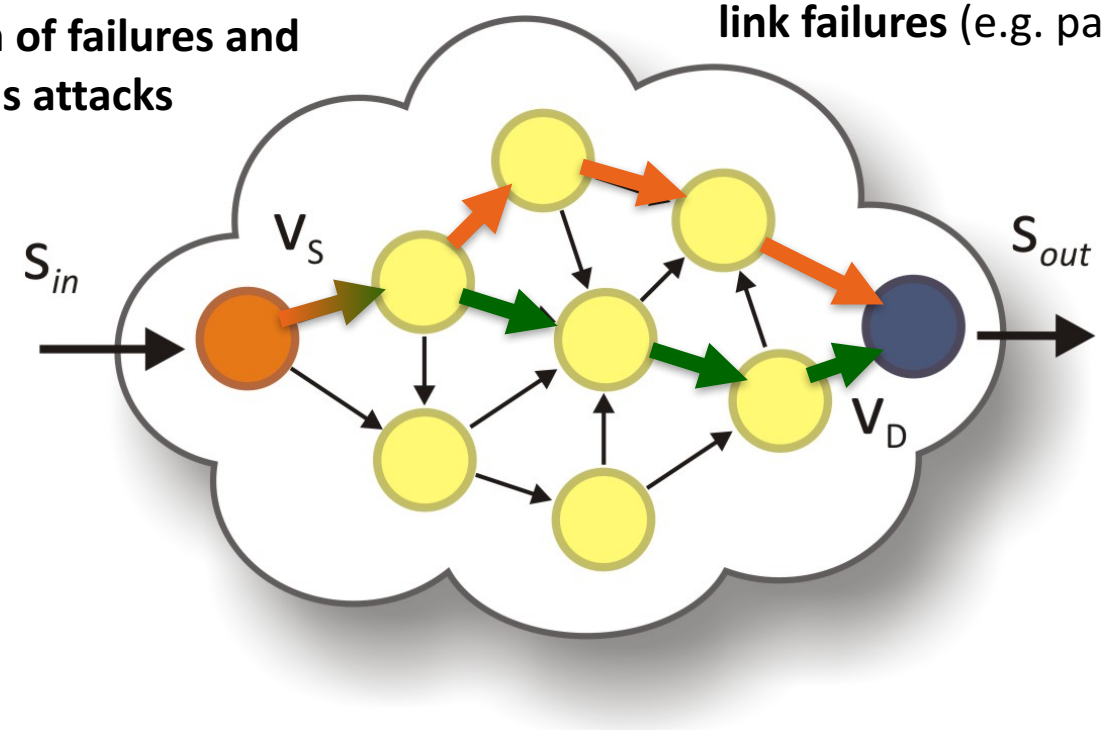
R. M. Jungers, A. D'Innocenzo, M. D. Di Benedetto. How to control Linear Systems with switching delays. 13th European Control Conference (ECC14), Strasbourg, France, June 24-27, 2014.

R.M. Jungers, A. D'Innocenzo, M.D. Di Benedetto. Feedback stabilization of dynamical systems with switched delays. 51st IEEE Conference on Decision and Control, Maui, Hawaii, December 10-13 2012.

Jungers, Heemels et Al.: extension to *deterministic* packet losses

Redundancy in data routing…

- …enables **detection and isolation of failures and malicious attacks**
- …makes system **robust to short-term link failures** (e.g. packet losses)



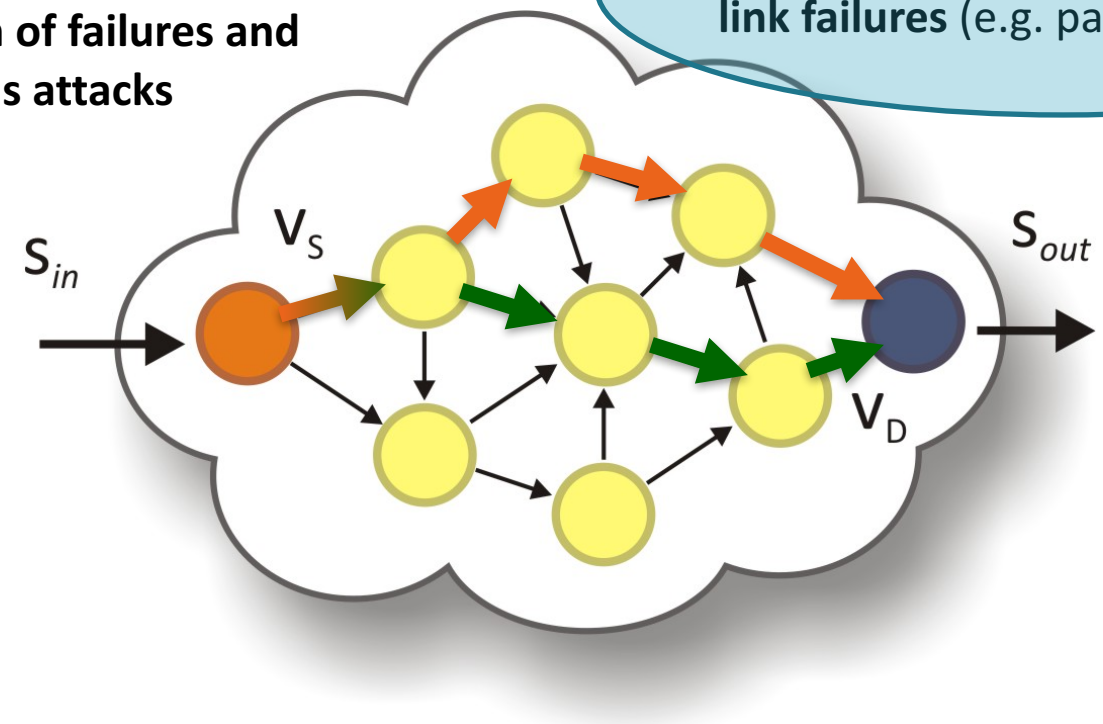
- …makes system **tolerant to long-term link failures**

1. Separation of concerns
2. Co-design

Redundancy in data routing…

- …enables **detection and isolation of failures and malicious attacks**

- …makes system **robust to short-term link failures** (e.g. packet losses)



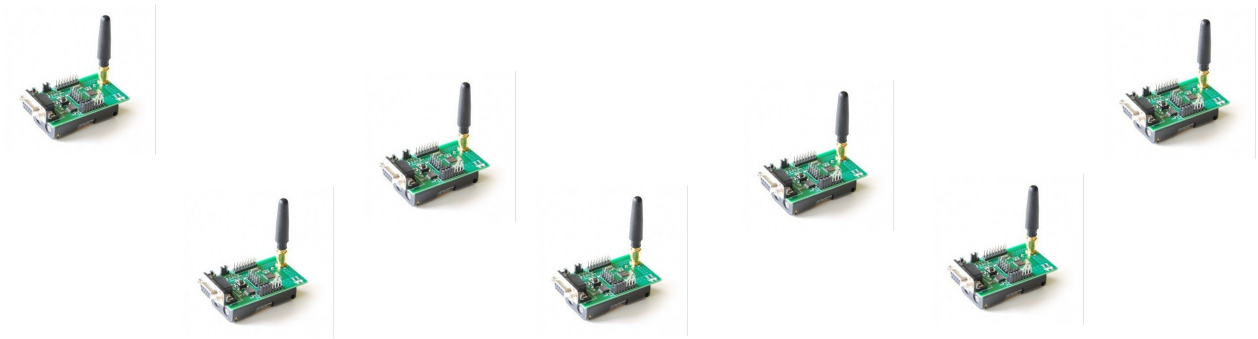
- …makes system **tolerant to long-term link failures**

Investigate algorithms to merge redundant data:

- Objective: stabilize the closed-loop system or maximise control performance
- Best strategy: keep most recent packet vs. compute combination?
- Different paths are associated with different delays
- Not a trivial question, best strategy from the point of view of stability strongly depends on plant and network: need for a control-theoretic approach

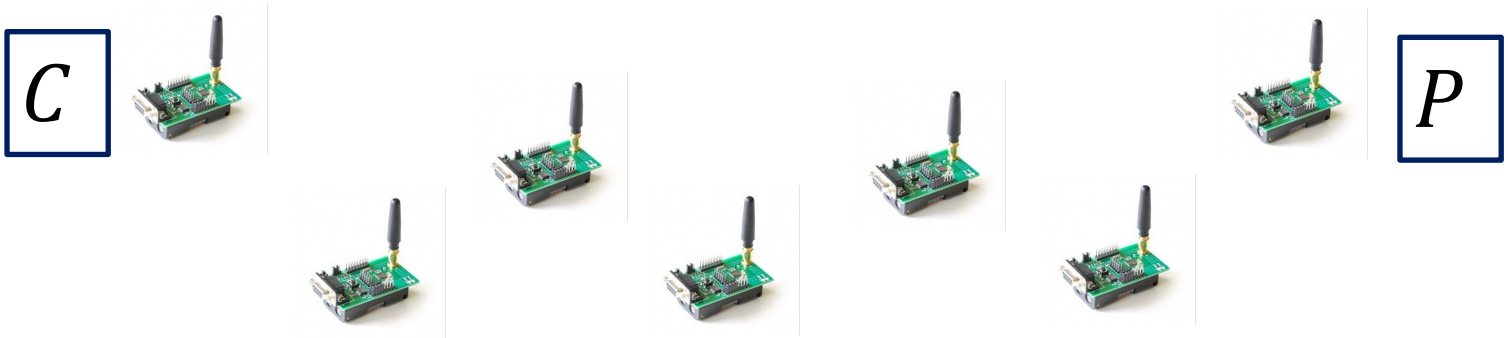
Motivational example

Network paths characteristics are often “at odds”



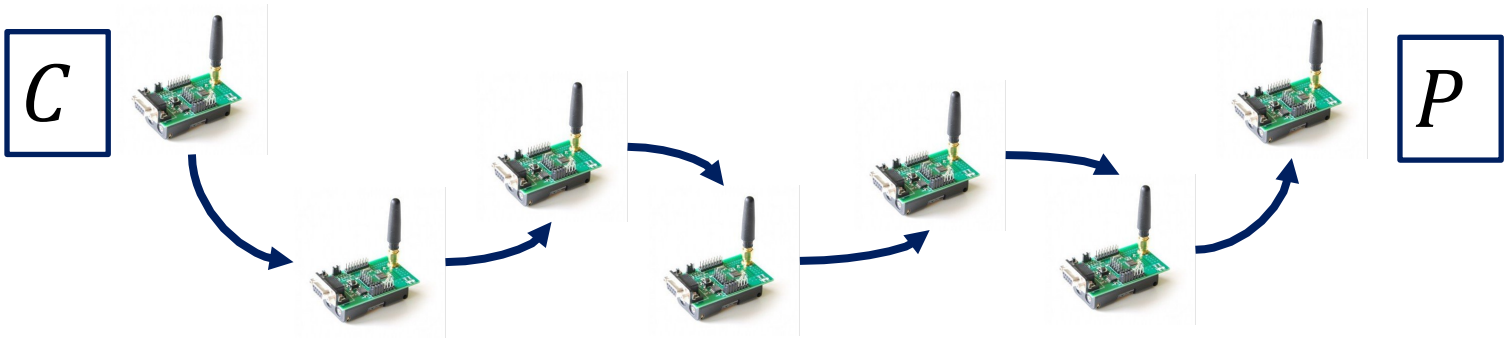
Motivational example

Network paths characteristics are often “at odds”



Motivational example

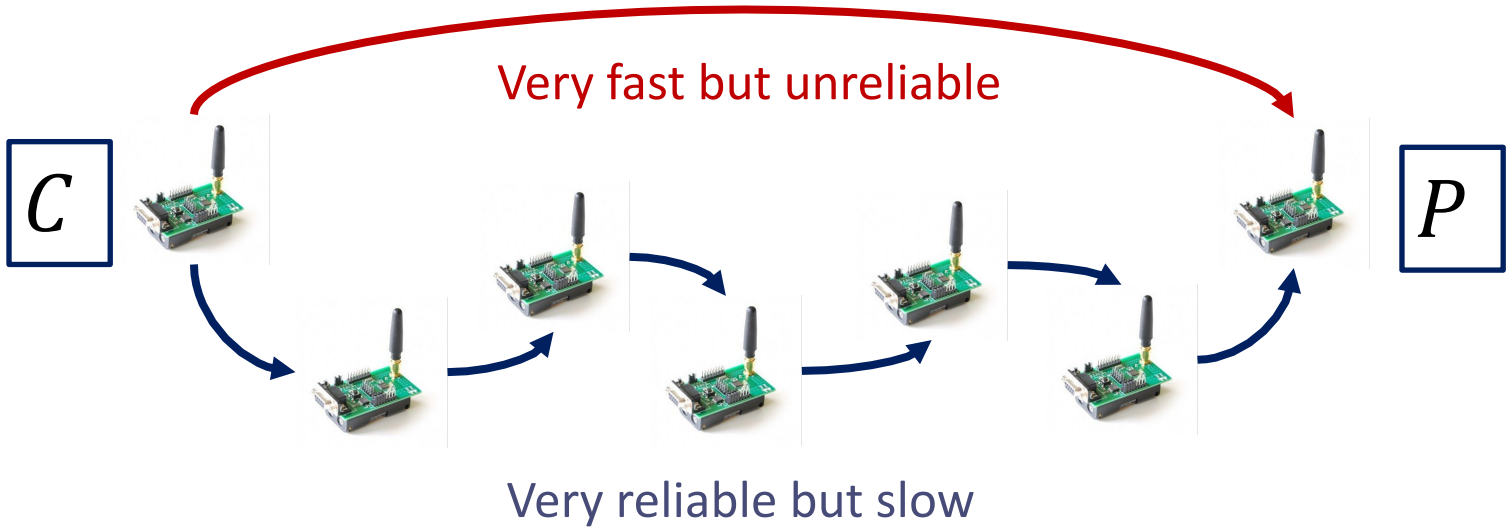
Network paths characteristics are often “at odds”



Very reliable but slow

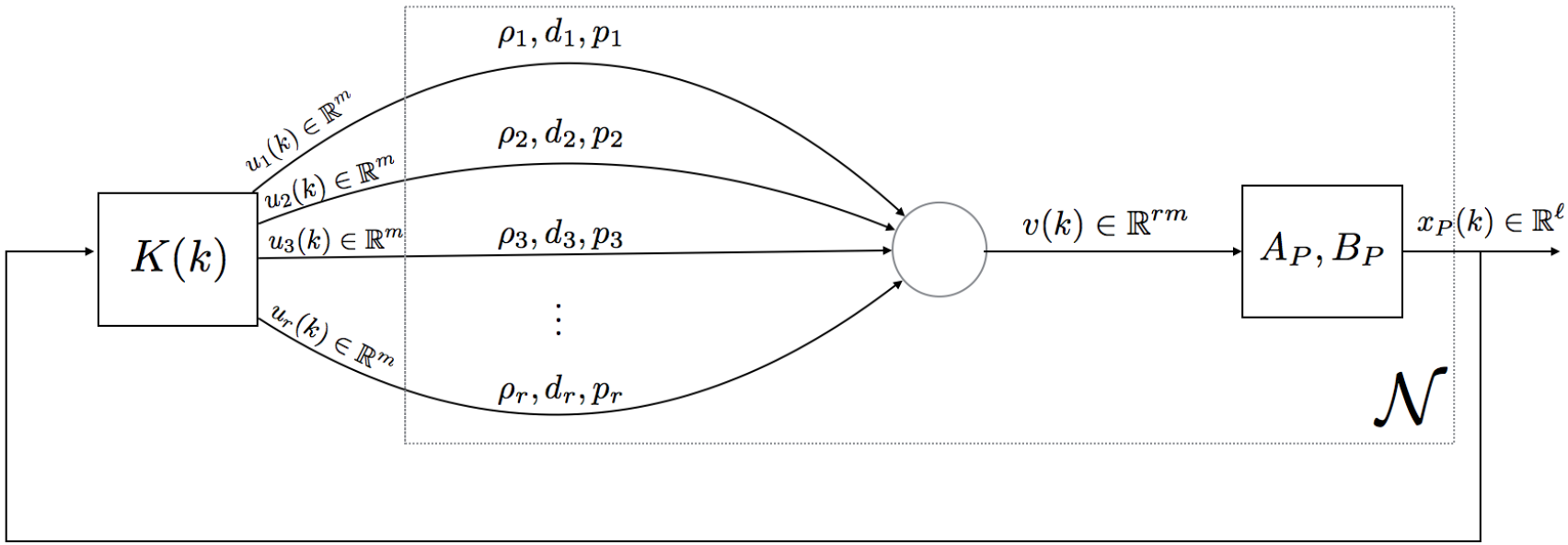
Motivational example

Network paths characteristics are often “at odds”

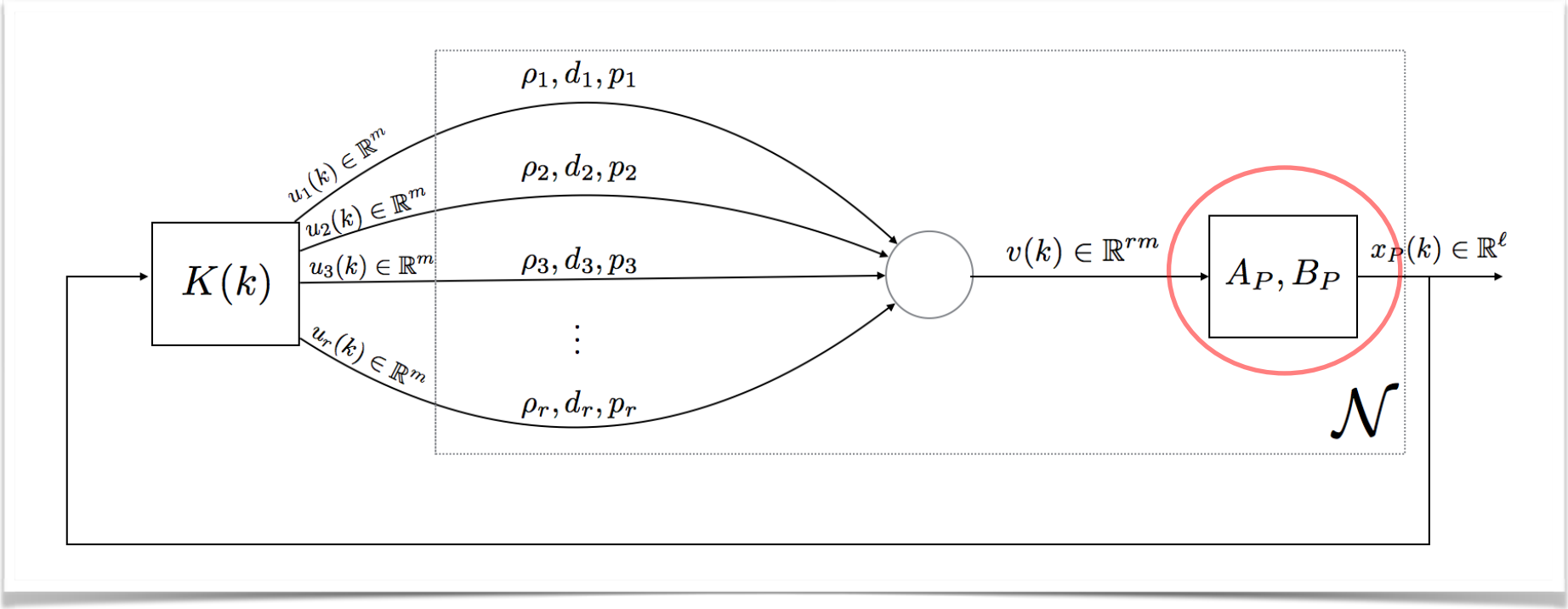


Optimally exploit the best of each path by co designing controller and routing

Network model

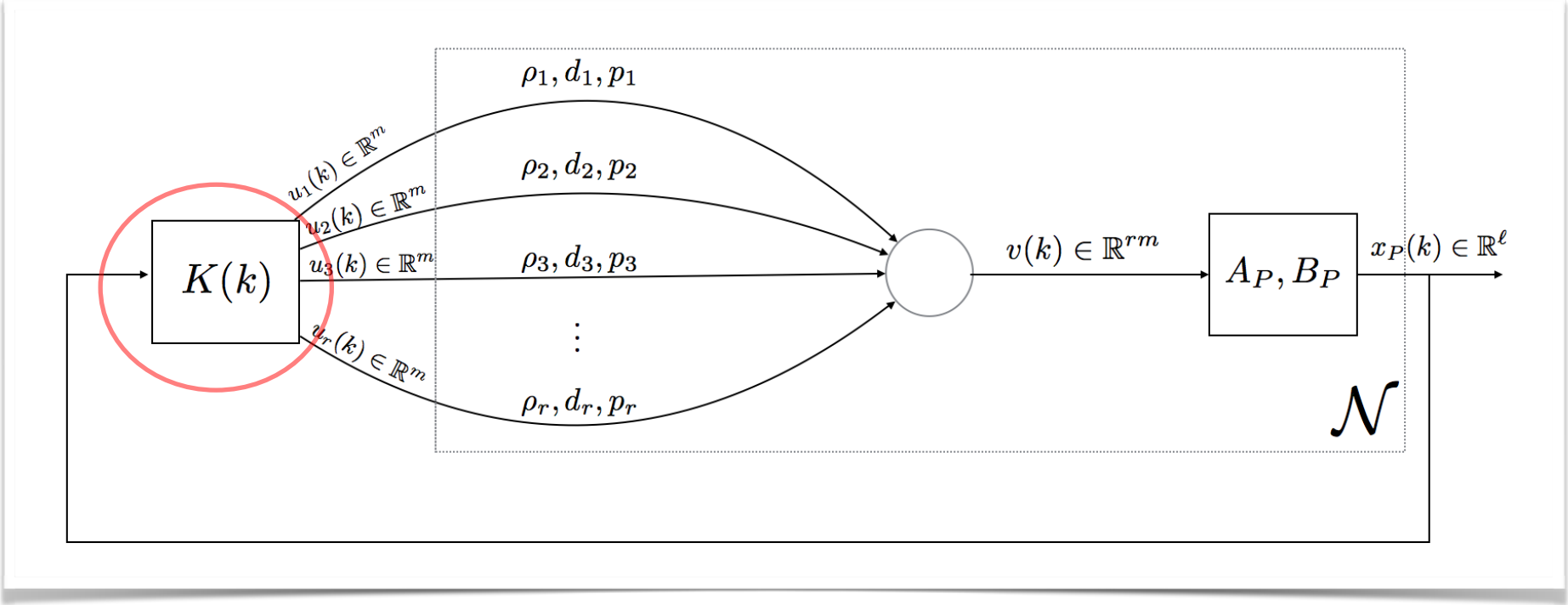


Network model



- Plant dynamics: $x_P(k+1) = A_P x_P(k) + B_P v(k)$

Network model

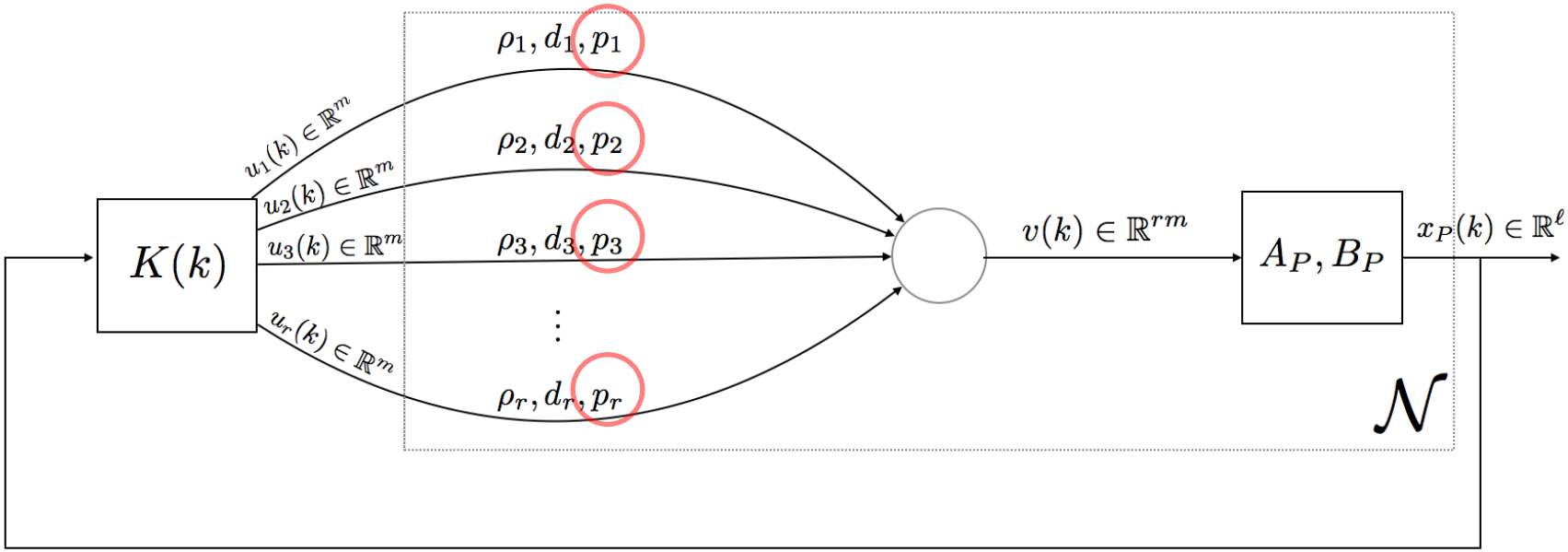


- Send different actuation data trough the paths of the network

$$u_i(k) = K_i\ (k)x_P(k), \qquad K(k) = \begin{bmatrix} K_1(k) \\ K_2(k) \\ \vdots \\ K_r(k) \end{bmatrix}$$

- Controller can measure via sensors the plant state (state feedback architecure)
- Controller is aware of the current and past actuation signals (memory)

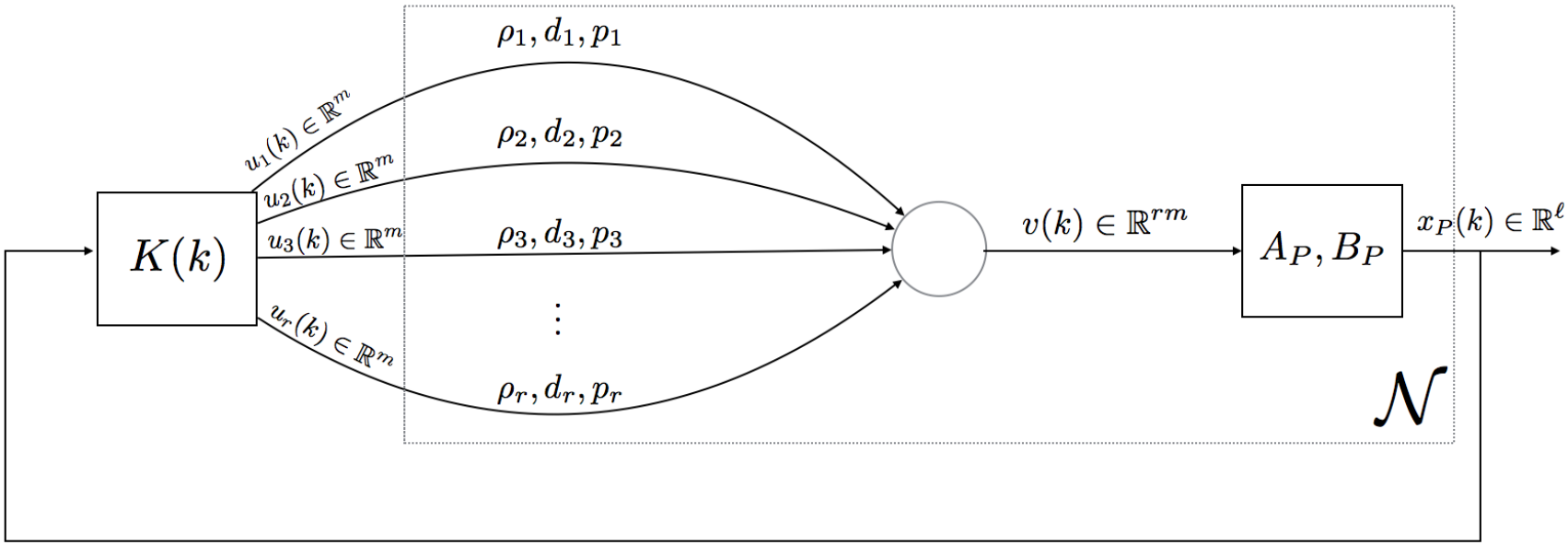
Packet losses model



- $\sigma_i(k)=\begin{cases} 0 & LOST \\ 1 & OK \end{cases}$ is a Markov chain modeling packet losses at time k on path i

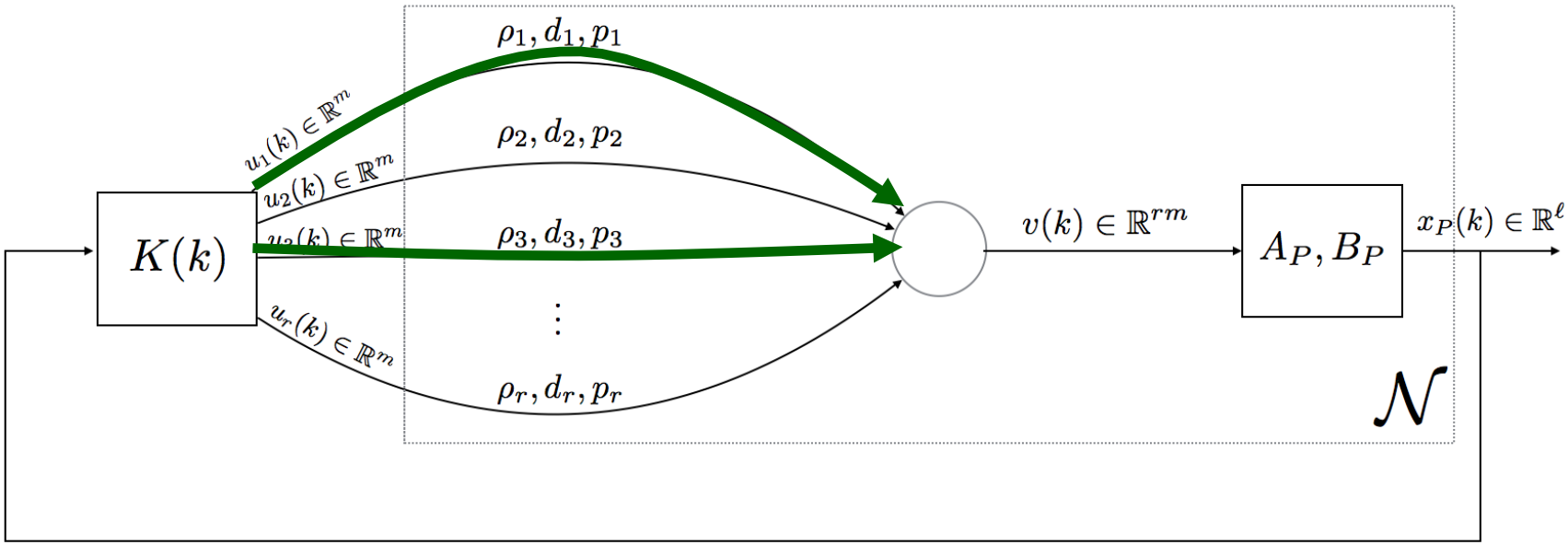
- $\sigma(k) = [\sigma_1(k) \quad \dots \quad \sigma_r(k)]$ occurrence of packet losses in the network - not necessarily measurable

Routing redundancy control



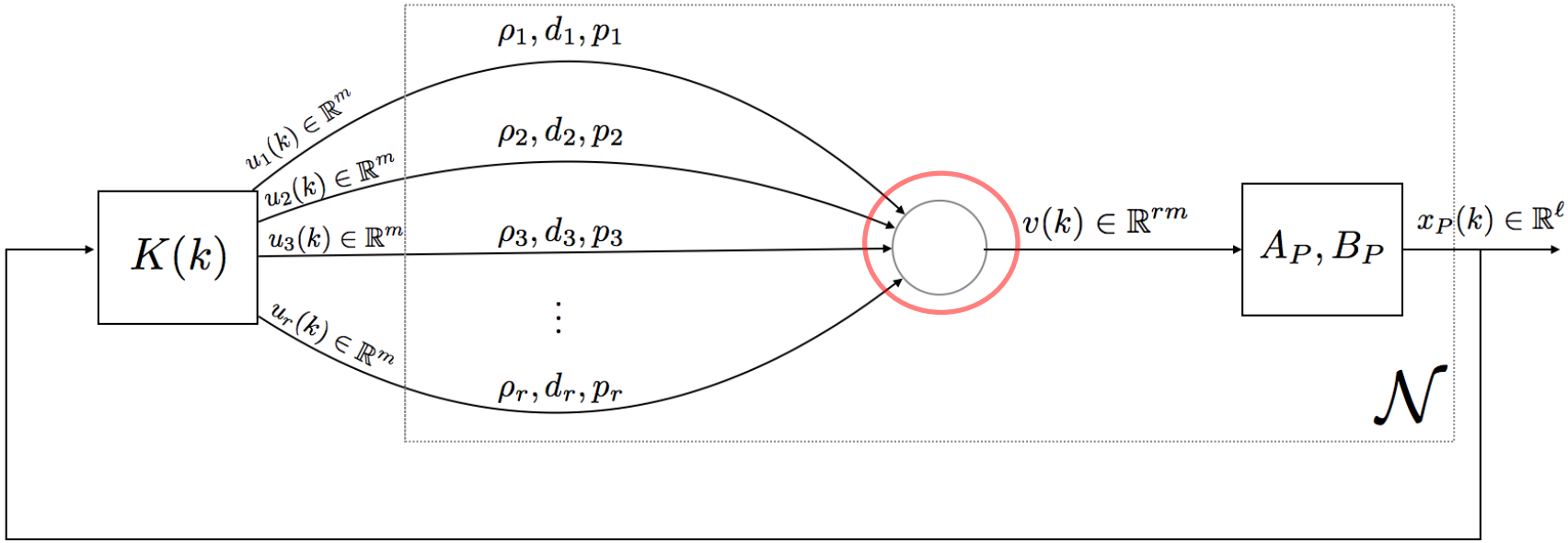
- $a_i(k)$ models the choice at time k of sending a packet via path i
- $a_i(k)=\begin{cases} 0 & \neg SEND \\ 1 & SEND \end{cases}, a(k) = [a_1(k) \quad \dots \quad a_r(k)]$

Routing redundancy control



- $a_i(k)$ models the choice at time k of sending a packet via path i
- $a_i(k)=\begin{cases} 0 & \neg SEND \\ 1 & SEND \end{cases}, a(k) = [a_1(k) \quad \dots \quad a_r(k)]$
- $a(k)$ is a control variable
- Example: $a(k) = [1, 0, 1, 0, \dots, 0]$ means that at time k we send actuation data only on paths 1 and 3.

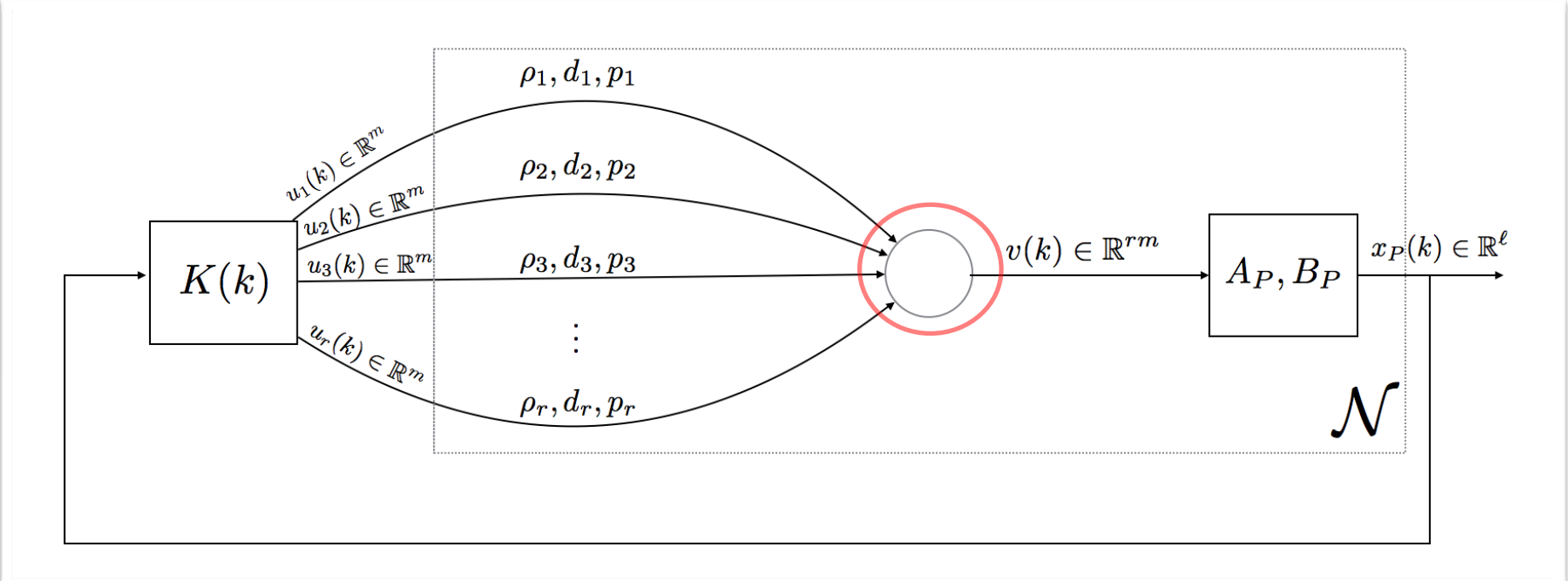
Network model



- Actuate sum of received packets…

$$v(k) = \sum_{i=1}^{r} u_i(k - d_i)$$

Network model

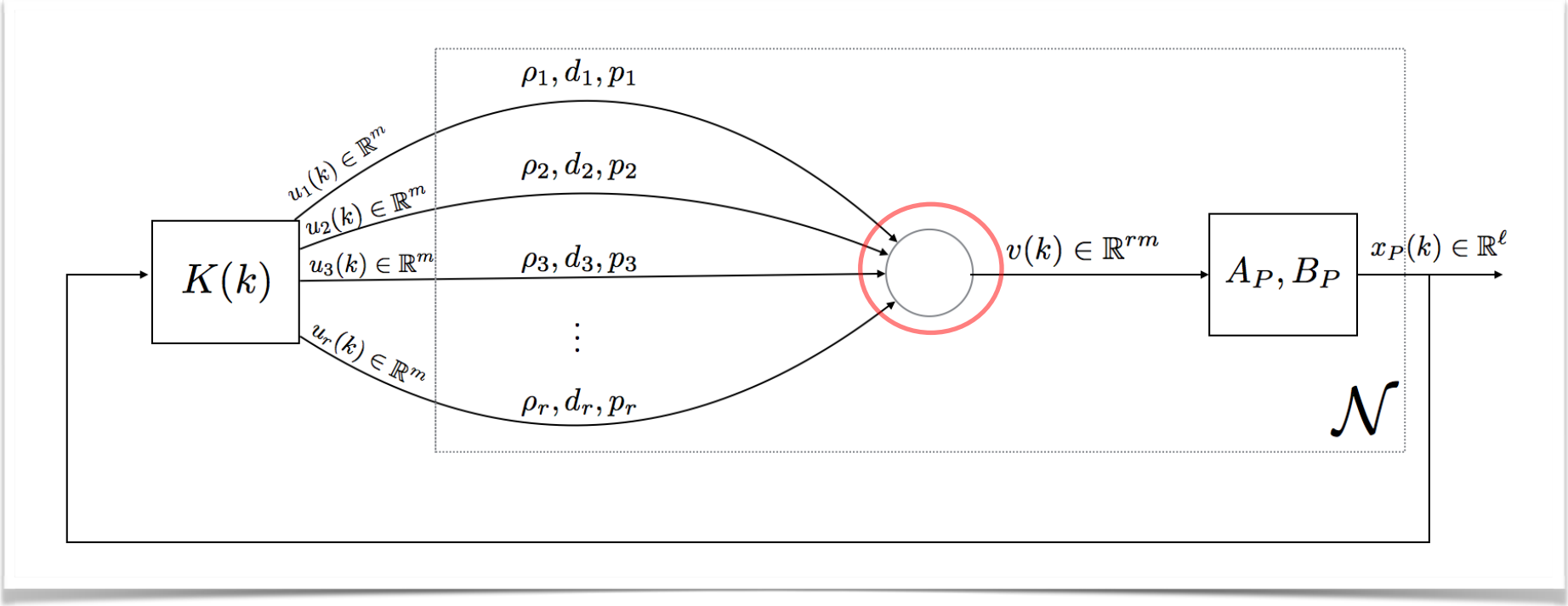


- Actuate sum of received packets…or actuate most recent received packet

$$v(k) = \sum_{i=1}^{r} u_i(k - d_i)$$

$v(k) = u_{i^*}(k - d_{i^*})$, with i^* the index of the path associated to the smallest delay among received packets

Network model



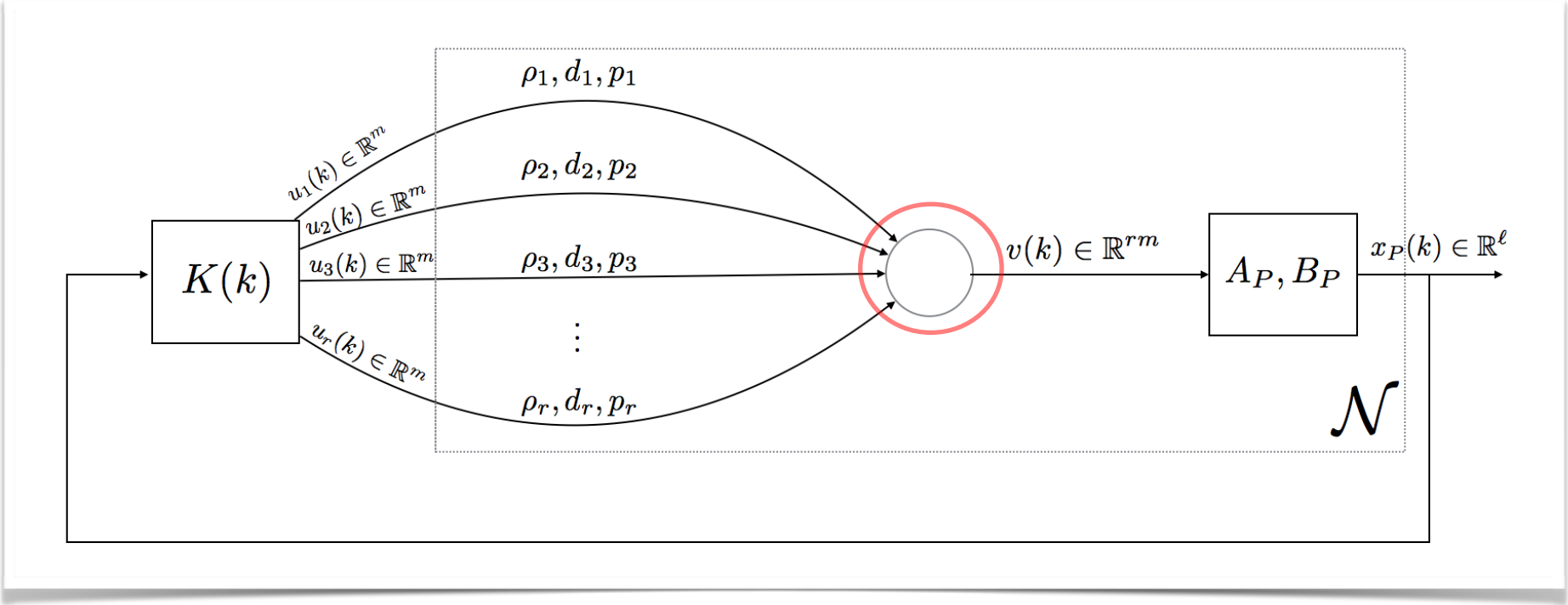
- Actuate sum of received packets…or actuate most recent received packet

$$v(k) = \sum_{i=1}^{r} u_i(k - d_i)$$

$v(k) = u_{i^*}(k - d_{i^*})$, with i^* the index of the path associated to the smallest delay among received packets

- When the actuator receives no packets, actuate zero…

Network model



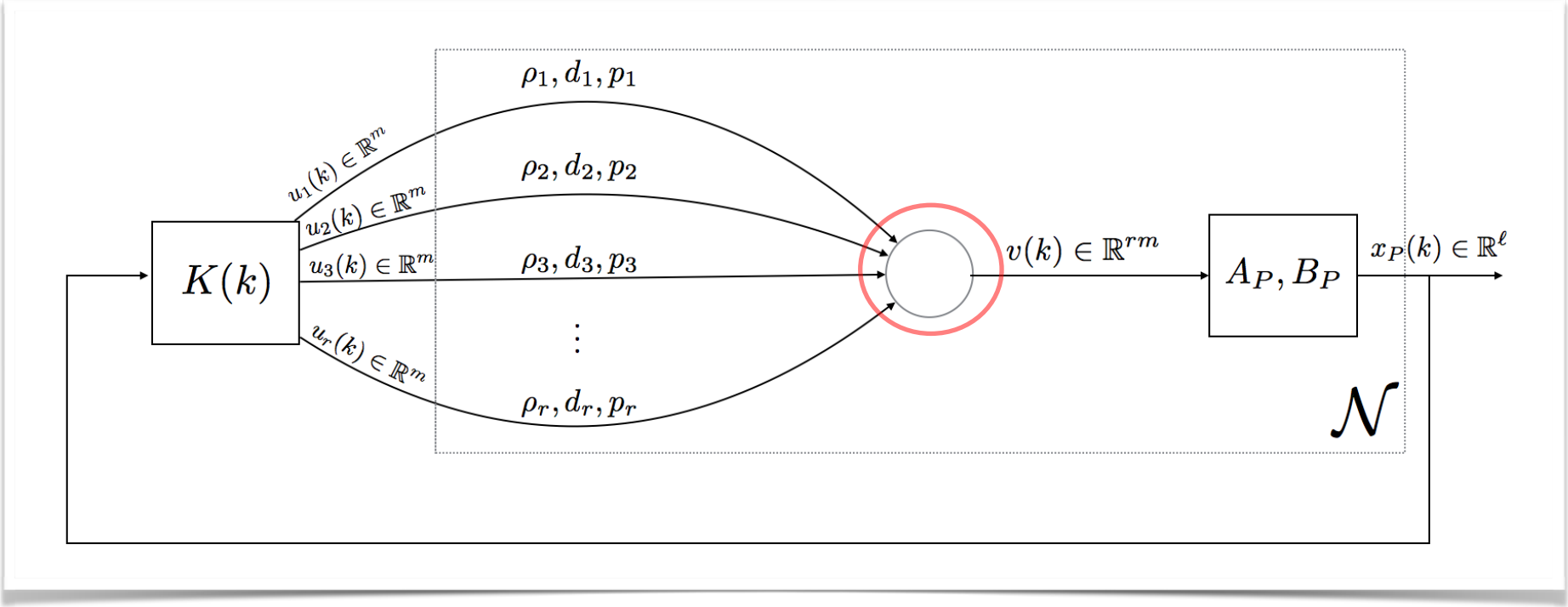
- Actuate sum of received packets…or actuate most recent received packet

$$v(k) = \sum_{i=1}^{r} u_i(k - d_i)$$

$v(k) = u_{i^*}(k - d_{i^*})$, with i^* the index of the path associated to the smallest delay among received packets

- When the actuator receives no packets, actuate zero…or hold the previous actuation

Network model



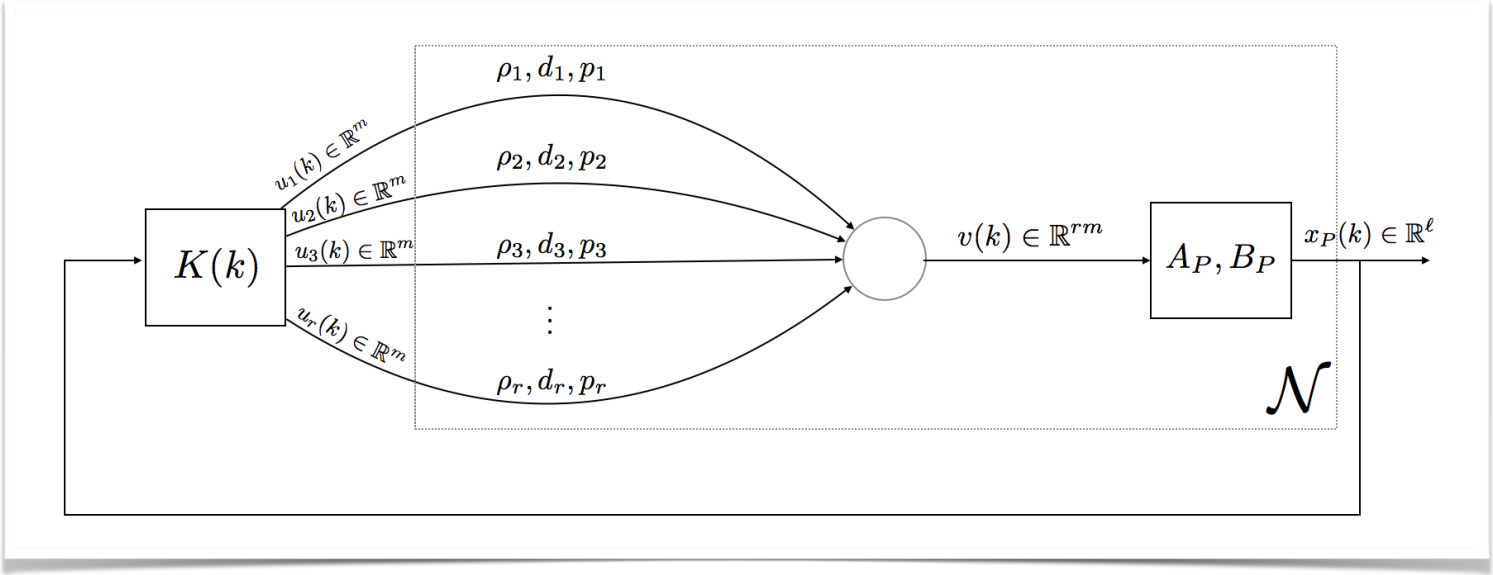
- Actuate sum of received packets…or actuate most recent received packet

$$v(k) = \sum_{i=1}^{r} u_i(k - d_i)$$

$v(k) = u_{i^*}(k - d_{i^*})$, with i^* the index of the path associated to the smallest delay among received packets

- When the actuator receives no packets, actuate zero…or hold the previous actuation
- Optimal design of $K(k), a(k)$ allows determining what is the best protocol to apply at the actuator

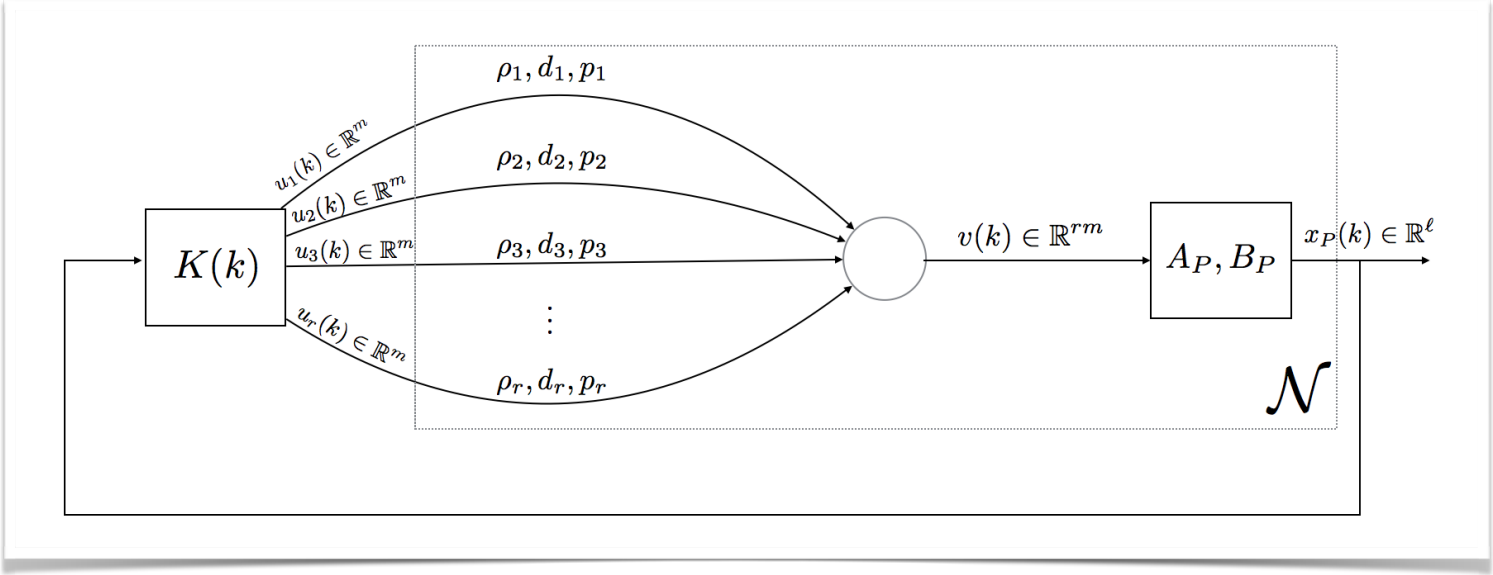
Mathematical model



$$\begin{cases} x(k+1) = A_{\sigma(k)}x(k) + B_{a(k)}u(k) \\ y(k) = x(k) \end{cases}$$

$$x(k) = [x_p(k) \;\; x_N(k)]'$$

Mathematical model



$$\begin{cases} x(k+1) = A_{\sigma(k)}x(k) + B_{a(k)}u(k) \\ y(k) = x(k) \end{cases}$$

$$x(k) = [x_p(k) \;\; x_N(k)]'$$

$$A_{\sigma(k)} = \begin{bmatrix} A_P & \Lambda_1(\sigma(k)) & \Lambda_2(\sigma(k)) & \cdots & \Lambda_r(\sigma(k)) \\ 0 & \Gamma_1 & 0 & \cdots & 0 \\ 0 & 0 & \Gamma_2 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \Gamma_r \end{bmatrix} \in \mathbb{R}^{\ell+\nu(r)\times\ell+\nu(r)}$$

$$\Lambda_i(\sigma(k)) \doteq \sigma_i(k) \begin{bmatrix} B_P & 0 & \cdots & 0 \end{bmatrix} \in \mathbb{R}^{\ell\times d_i},$$

$$\Gamma_i \doteq \begin{bmatrix} 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \\ 0 & 0 & \cdots & 0 & 1 \\ 0 & 0 & \cdots & 0 & 0 \end{bmatrix} \in \mathbb{R}^{d_i\times d_i},$$

$$B_{a(k)} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ a_1(k)\mathbf{e}_{d_1} & 0 & \cdots & 0 \\ 0 & a_2(k)\mathbf{e}_{d_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & a_r(k)\mathbf{e}_{d_r} \end{bmatrix} \in \mathbb{R}^{\ell+\nu(r)}$$

MJLS mathematical framework

We leverage **discrete-time Markov-jump linear systems (MJLS)**. A MJLS is a **switching** linear system where the switching signal is a **Markov chain**. The transition probability matrix (TPM) $P(k)$ of the Markov chain $\sigma(k)$ can be used to model the stochastic process that rules packet losses due to wireless communication.

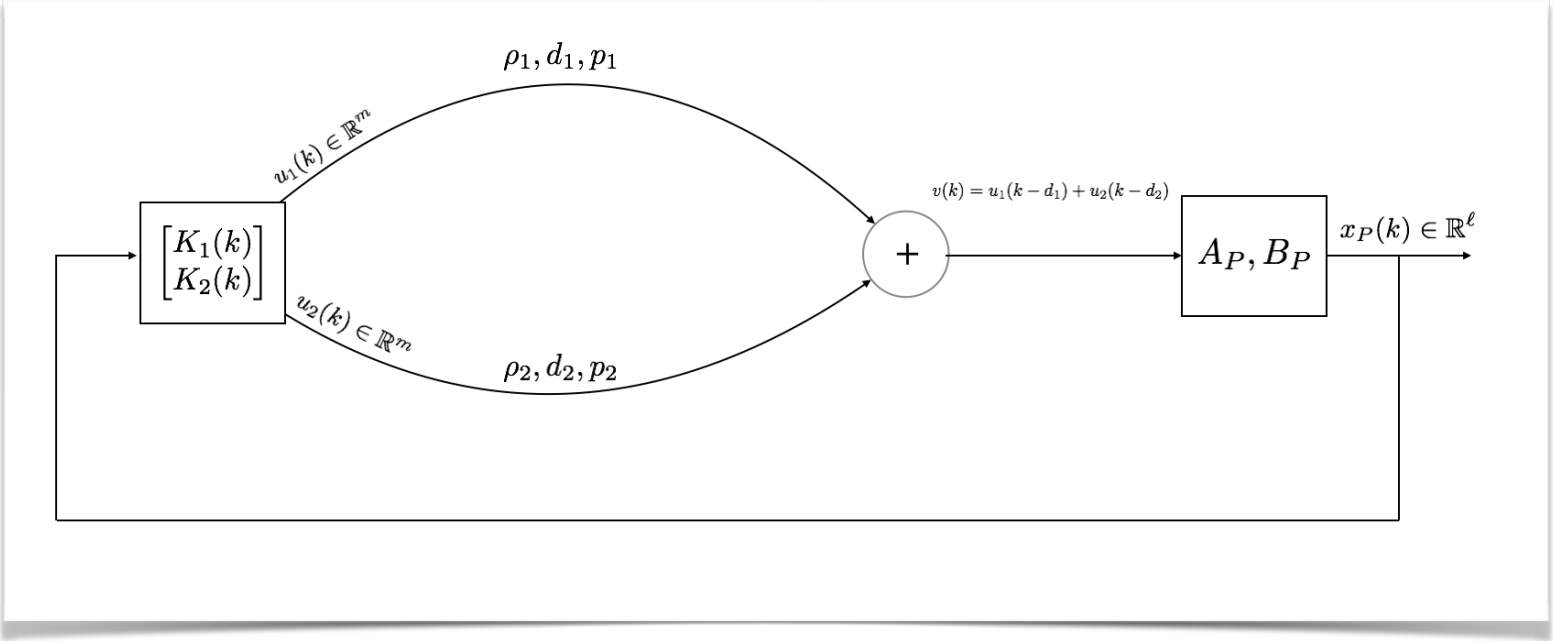
$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,

$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,

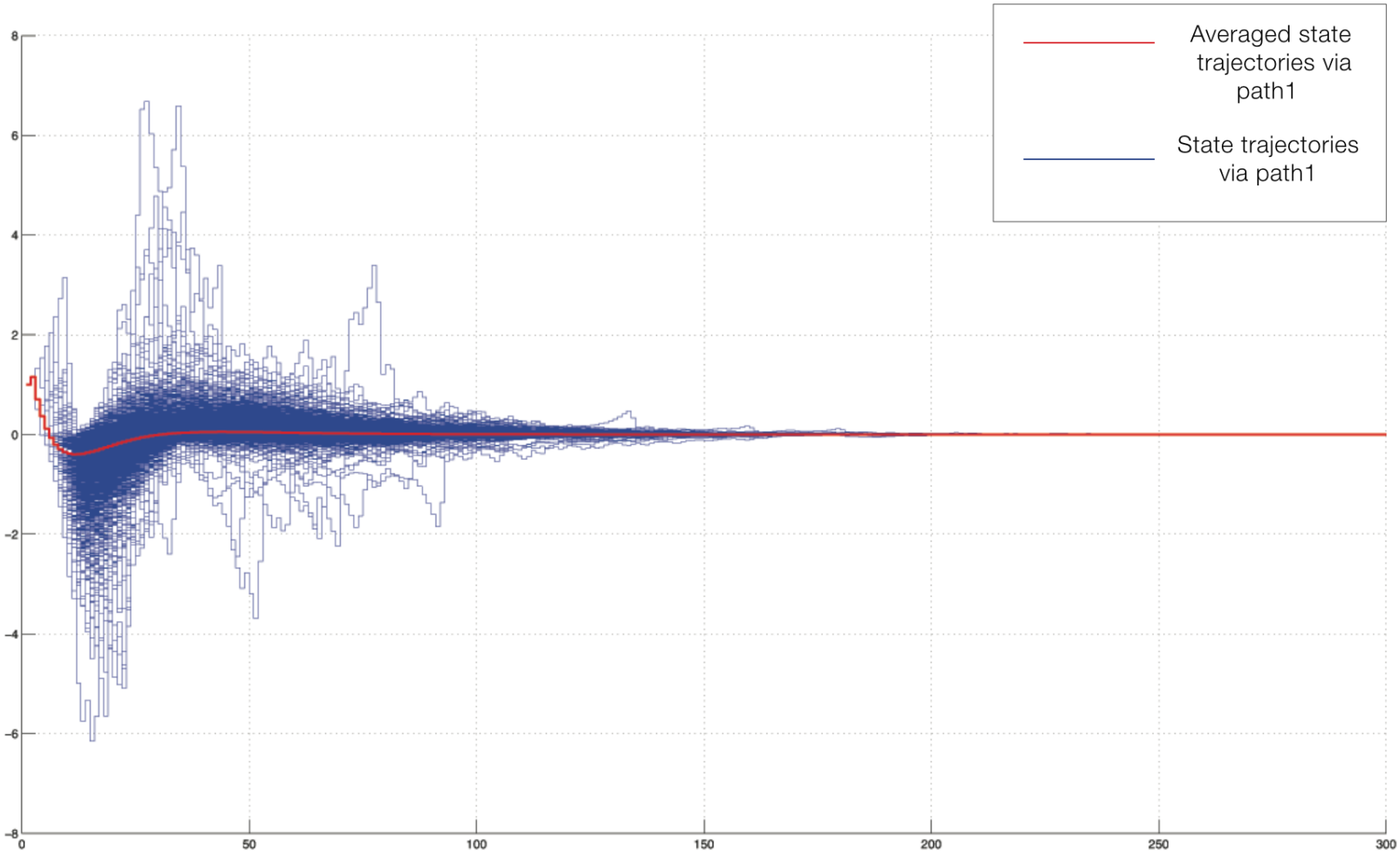
$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

Example

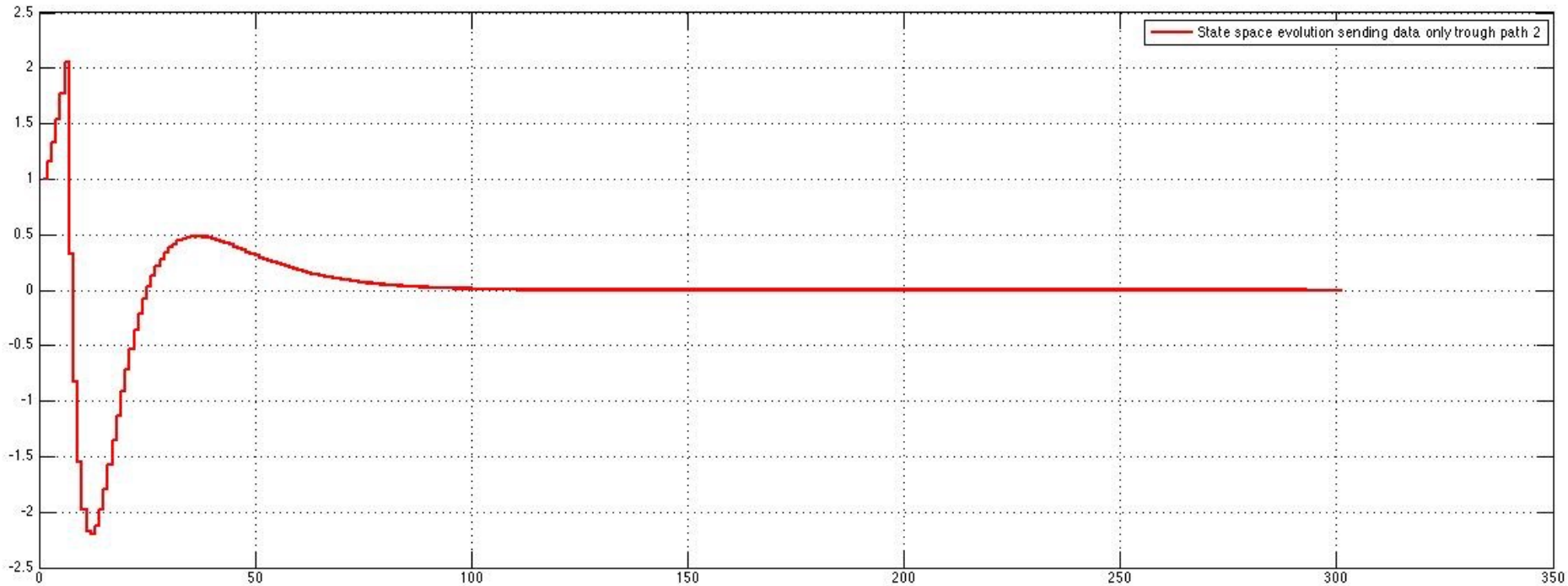


- 4-dimensional unstable plant
- 2 routing paths: $p_1 = {}^1/_4, d_1 = 1; p_2 = 0, d_2 = 5$
- 5k Monte Carlo simulations
- Compare 3 routing policies:
  - LQR using only path 1 for any k
  - LQR using only path 2 for any k
  - LQR using both paths for any k

Use only path 1 for any k

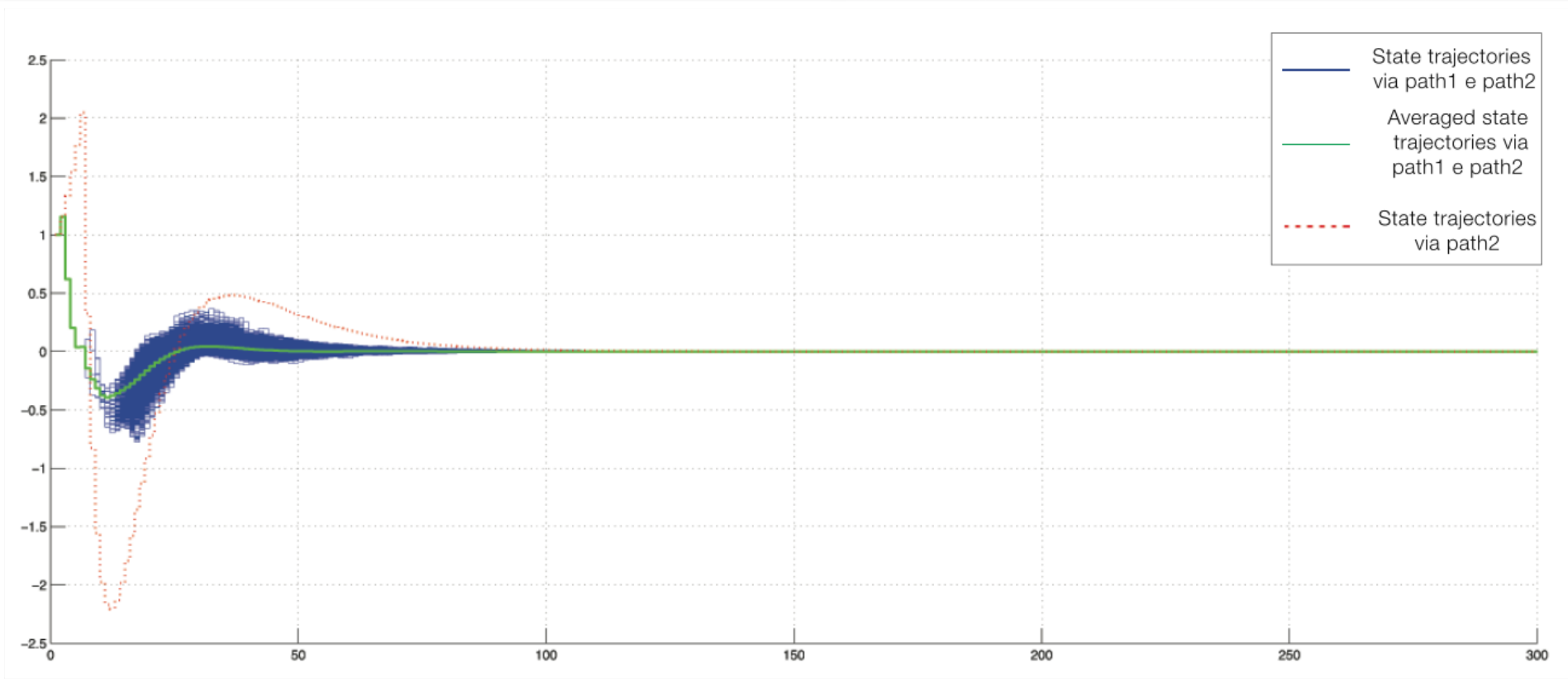


Use only path 2 for any k



Use both paths for any k

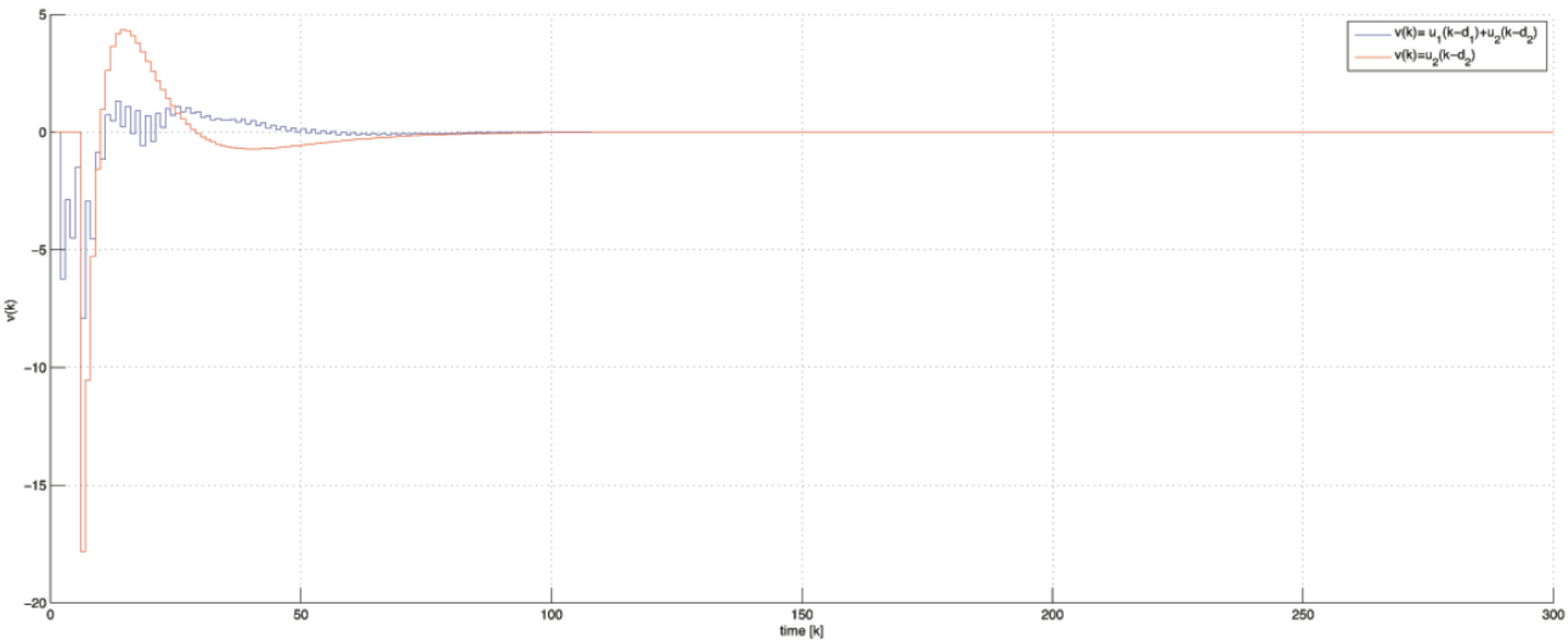
Plant state plot



	Averaged cost
Via path 1	~ 900
Via path 2	~ 250
Via paths 1 and 2	~ 100

Use both paths for any k

Actuation signals plot



MJLS mathematical framework

We leverage **discrete-time Markov-jump linear systems (MJLS)**. A MJLS is a **switching** linear system where the switching signal is a **Markov chain**. The transition probability matrix (TPM) $P(k)$ of the Markov chain $\sigma(k)$ can be used to model the stochastic process that rules packet losses due to wireless communication.

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,
$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,
$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

<u>Extension 1</u>: In general wireless channels are time-varying and packet loss probability not easy to compute/measure, thus $P(k)$ is unmeasurable and time-varying

<u>Extension 2</u>: We want to control the routing policy: MJLS where $\theta(k)$ is a Markov-decision process

MJLS mathematical framework

We leverage **discrete-time Markov-jump linear systems (MJLS)**. A MJLS is a **switching** linear system where the switching signal is a **Markov chain**. The transition probability matrix (TPM) $P(k)$ of the Markov chain $\sigma(k)$ can be used to model the stochastic process that rules packet losses due to wireless communication.

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,
$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,
$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

<u>Extension 1</u>: In general wireless channels are time-varying and packet loss probability not easy to compute/measure, thus $P(k)$ is unmeasurable and time-varying

<u>Extension 2</u>: We want to control the routing policy: MJLS where $\theta(k)$ is a Markov-decision process

time-inhomogeneous MJLS

MJLS is **time-inhomogeneous**, i.e. a Markov chain has its **TPM varying over time**, with variations that are **arbitrary** within a **polytopic** set of **stochastic matrices**.

Formally, TPM $P(k) = \left[p_{ij}(k)\right]_{N \times N}$, $k \in \mathbb{N}_0$, $N, L \in \mathbb{N}$,

time-inhomogeneous MJLS

MJLS is **time-inhomogeneous**, i.e. a Markov chain has its **TPM varying over time**, with variations that are **arbitrary** within a **polytopic** set of **stochastic matrices**.

Formally, TPM $P(k) = \left[p_{ij}(k)\right]_{N\times N}$, $k \in \mathbb{N}_0$, $N, L \in \mathbb{N}$,

$$P(k) = \sum_{\ell=1}^{L} \lambda_l(k) P_\ell \,,\, \lambda_l(k) \geq 0 \,,\, \sum_{\ell=1}^{L} \lambda_l(k) = 1 \,,$$

$\{P_\ell\}_{\ell\in\mathcal{L}\triangleq\{1,\ldots,L\}} = \mathcal{P}_L$ is a given set of vertices

time-inhomogeneous MJLS

MJLS is **time-inhomogeneous**, i.e. a Markov chain has its **TPM varying over time**, with variations that are **arbitrary** within a **polytopic** set of **stochastic matrices**.

Formally, TPM $P(k) = \left[p_{ij}(k)\right]_{N\times N}$, $k \in \mathbb{N}_0$, $N, L \in \mathbb{N}$,

$$P(k) = \sum_{\ell=1}^{L} \lambda_l(k) P_\ell \,,\; \lambda_l(k) \geq 0 \,,\; \sum_{\ell=1}^{L} \lambda_l(k) = 1 \,,$$

$\{P_\ell\}_{\ell\in\mathcal{L}\triangleq\{1,\ldots,L\}} = \mathcal{P}_L$ is a given set of vertices

So, the TPM $P(k) = \left[p_{ij}(k)\right]_{N\times N}$ is a stochastic $N\times N$ matrix with entries $p_{ij}(k)$.

Stability of autonomous time-homogeneous MJLS

The noiseless autonomous discrete-time time-homogeneous **MJLS** $(\boldsymbol{S})$ is

$$\begin{cases} x(k+1) = A_{\theta(k)} x(k) \,, \\ x(0) = x_0 \,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector, $k \in \mathbb{N}_0$,
$A_{\theta(k)} \in \mathbb{C}^{n \times n}$ the state matrix,
associated with the operational mode $\theta(k)$

Stability of autonomous time-homogeneous MJLS

The noiseless autonomous discrete-time time-homogeneous **MJLS** $(\boldsymbol{S})$ is

$$\begin{cases} x(k+1) = A_{\theta(k)} x(k)\,, \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector, $k \in \mathbb{N}_0$,
$A_{\theta(k)} \in \mathbb{C}^{n \times n}$ the state matrix,
associated with the operational mode $\theta(k)$

For a **time-homogeneous** MJLS, the system is **Mean Square Stable (MSS)** if and only if a spectral radius of the augmented matrix $\mathcal{A}$, associated to the second moment of the state vector $x(k)$, is < 1

Stability of autonomous time-homogeneous MJLS

The noiseless autonomous discrete-time time-homogeneous **MJLS** $(\boldsymbol{S})$ is

$$\begin{cases} x(k+1) = A_{\theta(k)} x(k)\,, \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector, $k \in \mathbb{N}_0$,
$A_{\theta(k)} \in \mathbb{C}^{n\times n}$ the state matrix,
associated with the operational mode $\theta(k)$

For a **time-homogeneous** MJLS, the system is **Mean Square Stable (MSS)** if and only if a spectral radius of the augmented matrix $\mathcal{A}$, associated to the second moment of the state vector $x(k)$, is < 1, where

$$\mathcal{A} \triangleq (P^T \otimes \mathbb{I}_{n^2})\,\mathrm{diag}\left[\overline{\mathrm{A}}_i \otimes \mathrm{A}_i\right], \qquad \mathrm{diag}\left[\overline{\mathrm{A}}_i \otimes \mathrm{A}_i\right] \triangleq \begin{bmatrix} \overline{\mathrm{A}}_1 \otimes \mathrm{A}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \overline{\mathrm{A}}_N \otimes \mathrm{A}_N \end{bmatrix},$$

$\overline{\mathrm{A}}_i$ is the conjugate of A_i , and P^T is transpose of P.

Stability of autonomous time-homogeneous MJLS

The noiseless autonomous discrete-time time-homogeneous **MJLS** $(\boldsymbol{S})$ is

$$\begin{cases} x(k+1) = A_{\theta(k)} x(k)\,, \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector, $k \in \mathbb{N}_0$,
$A_{\theta(k)} \in \mathbb{C}^{n\times n}$ the state matrix,
associated with the operational mode $\theta(k)$

For a **time-homogeneous** MJLS, the system is **Mean Square Stable (MSS)** if and only if a spectral radius of the augmented matrix $\mathcal{A}$, associated to the second moment of the state vector $x(k)$, is < 1, where

$$\mathcal{A} \triangleq (P^T \otimes \mathbb{I}_{n^2})\,\mathrm{diag}\left[\overline{\mathrm{A}}_i \otimes \mathrm{A}_i\right], \qquad \mathrm{diag}\left[\overline{\mathrm{A}}_i \otimes \mathrm{A}_i\right] \triangleq \begin{bmatrix} \overline{\mathrm{A}}_1 \otimes \mathrm{A}_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \overline{\mathrm{A}}_N \otimes \mathrm{A}_N \end{bmatrix},$$

$\overline{\mathrm{A}}_i$ is the conjugate of A_i , and P^T is transpose of P.

Does a similar condition hold also for time-inhomogeneous MJLS?

Illustrative example

Let us consider the MJLS (S) with $N = 3$ operational modes, where the state matrices associated with the operational modes are

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1.2 \end{bmatrix},\ A_2 = \begin{bmatrix} 1.13 & 0 \\ 0.16 & 0.48 \end{bmatrix},\ A_3 = \begin{bmatrix} 0.3 & 0.13 \\ 0.16 & 1.14 \end{bmatrix}$$

Illustrative example

Let us consider the MJLS (S) with $N = 3$ operational modes, where the state matrices associated with the operational modes are

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1.2 \end{bmatrix},\ A_2 = \begin{bmatrix} 1.13 & 0 \\ 0.16 & 0.48 \end{bmatrix},\ A_3 = \begin{bmatrix} 0.3 & 0.13 \\ 0.16 & 1.14 \end{bmatrix}$$

The time-varying probability matrix $P(k)$ is uncertain and belongs to a polytope with $L = 2$ vertices

$$P_1 = \begin{bmatrix} 0 & 0.35 & 0.65 \\ 0.6 & 0.4 & 0 \\ 0.4 & 0.6 & 0 \end{bmatrix},\quad P_2 = \begin{bmatrix} 0.25 & 0.75 & 0 \\ 0 & 0.6 & 0.4 \\ 0 & 0.4 & 0.6 \end{bmatrix},$$

Any probability matrix within a polytope is represented by

$$P(k) = \lambda(k)P_1 + \big(1 - \lambda(k)\big)P_2$$

Illustrative example

Let us consider the MJLS (S) with $N = 3$ operational modes, where the state matrices associated with the operational modes are

$$A_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1.2 \end{bmatrix},\ A_2 = \begin{bmatrix} 1.13 & 0 \\ 0.16 & 0.48 \end{bmatrix},\ A_3 = \begin{bmatrix} 0.3 & 0.13 \\ 0.16 & 1.14 \end{bmatrix}$$

The time-varying probability matrix $P(k)$ is uncertain and belongs to a polytope with $L = 2$ vertices

$$P_1 = \begin{bmatrix} 0 & 0.35 & 0.65 \\ 0.6 & 0.4 & 0 \\ 0.4 & 0.6 & 0 \end{bmatrix},\quad P_2 = \begin{bmatrix} 0.25 & 0.75 & 0 \\ 0 & 0.6 & 0.4 \\ 0 & 0.4 & 0.6 \end{bmatrix},$$

Any probability matrix within a polytope is represented by

$$P(k) = \lambda(k)P_1 + \big(1 - \lambda(k)\big)P_2$$

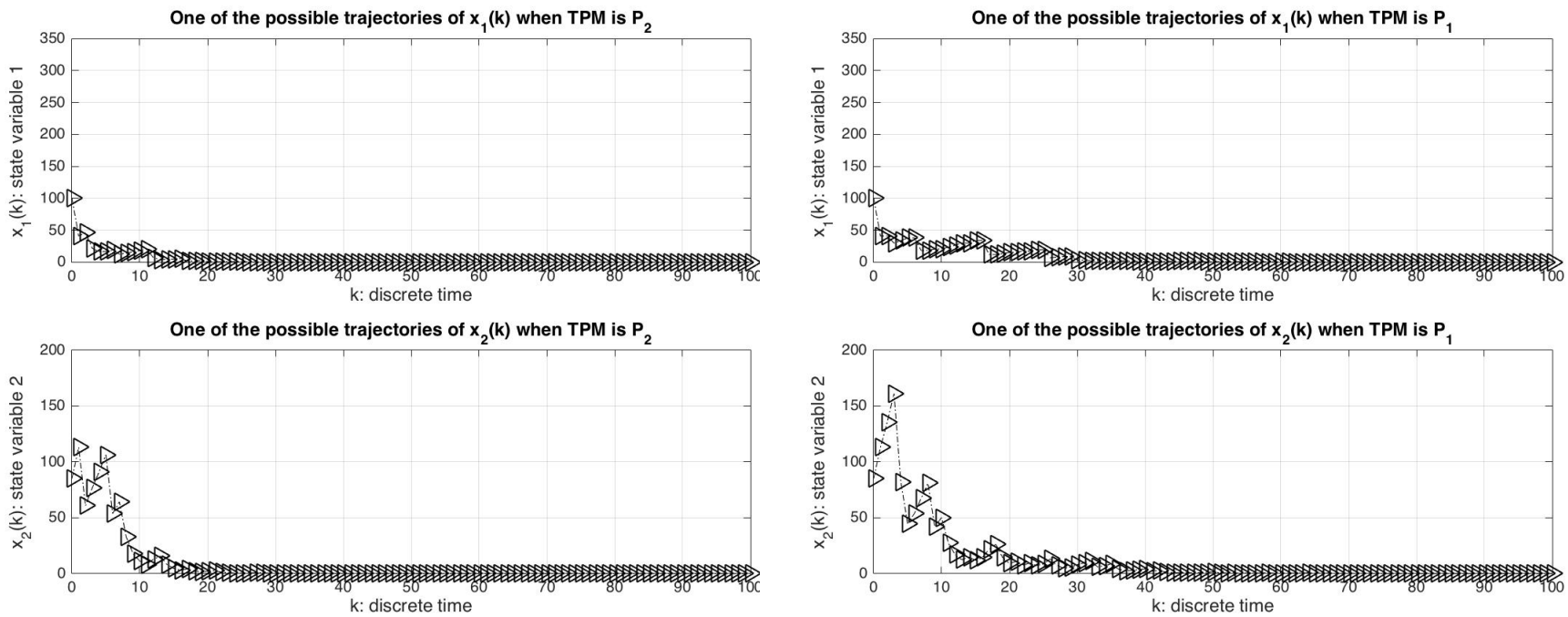
The spectral radii ρ of augmented matrices from the previous slide, here denoted by Λ_i, are

$$\rho(\mathcal{A}_1) = 0.901601 < 1\,, \qquad \rho(\mathcal{A}_2) = 0.905686 < 1$$

Thus, the time-homogeneous MJLS with TPM P_1, P_2 are mean square stable.

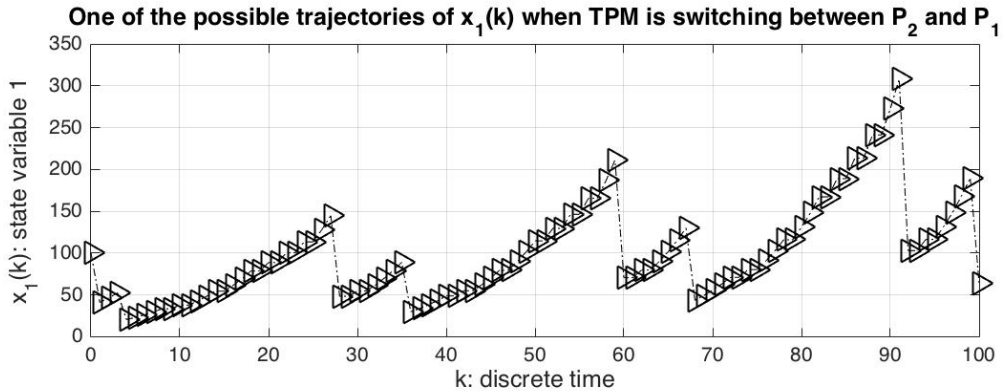
Illustrative example

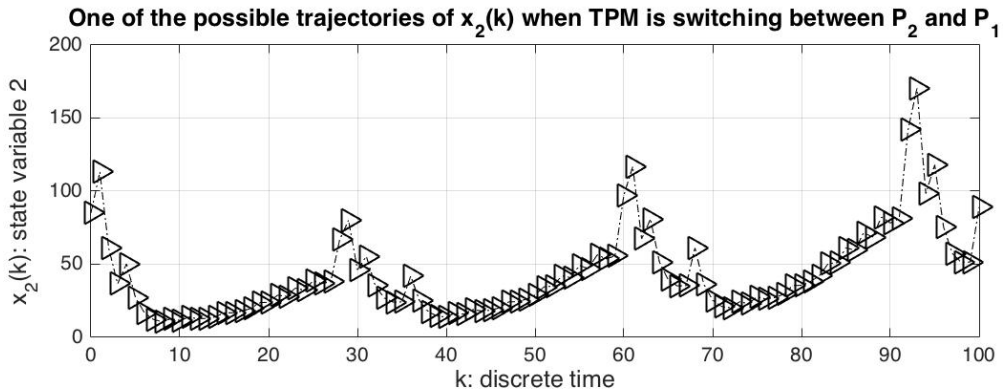
Sample trajectories of the system state vector having for all time instants either TPM P_1(left) or P_2(right), for $x_0 = \begin{bmatrix} 100 \\ 85 \end{bmatrix}$ and initial probability distribution $p_0 = [0.33 \quad 0.34 \quad 0.33\,]$



Illustrative example

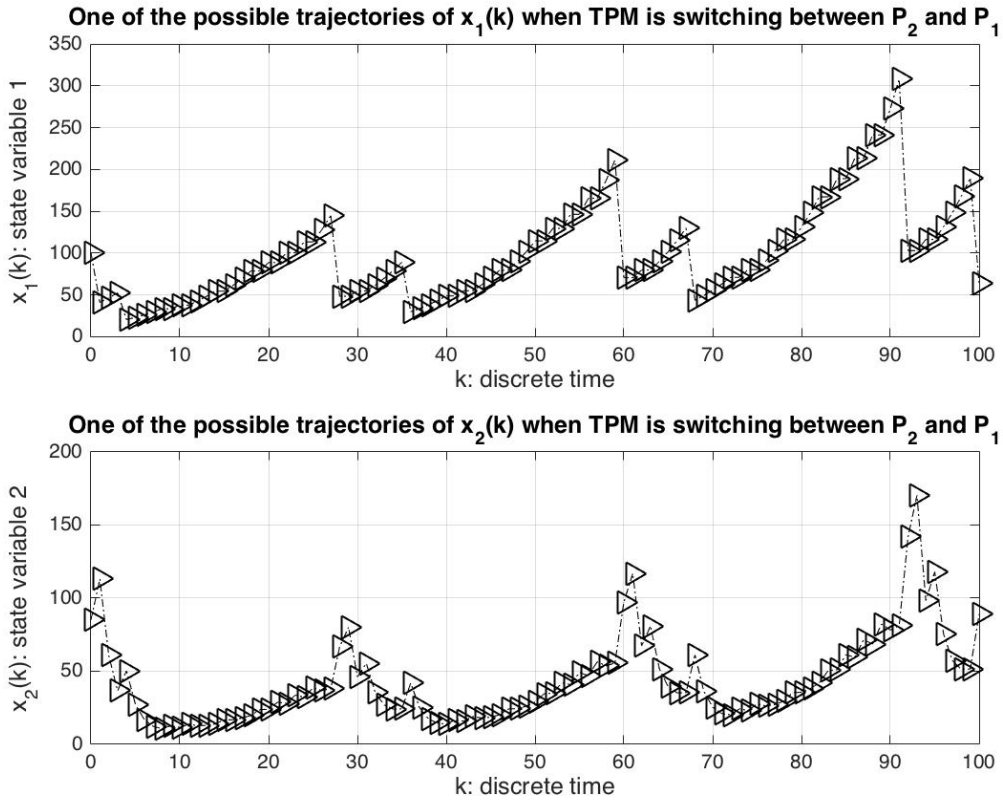
However, when the transition probability matrix is time-inhomogeneous and is switching between P_1 and P_2, a trajectory for the same conditions may become





Illustrative example

However, when the transition probability matrix is time-inhomogeneous and is switching between P_1 and P_2, a trajectory for the same conditions may become



The system is clearly unstable. Hence, some perturbations on transition probability matrix P can make a stable MJLS system unstable.

In the following we prove N&S conditions, based on the notion of Joint Spectral Radius, that will show that the considered system is indeed unstable.

Related works on stability of time-inhomogeneous MJLS

Only sufficient stability **conditions** have been derived for MJLS with time-inhomogeneous Markov chains having TPM arbitrarily varying within a polytopic set of stochastic matrices:

Related works on stability of time-inhomogeneous MJLS

Only sufficient stability **conditions** have been derived for MJLS with time-inhomogeneous Markov chains having TPM arbitrarily varying within a polytopic set of stochastic matrices:

- a **sufficient** condition for **stochastic stability** (SS) in terms of **linear matrix inequality** feasibility problem is provided in
  S. Aberkane, "*Stochastic stabilization of a class of nonhomogeneous Markovian jump linear systems*," *Syst. & Control Lett.*, vol. 60, no. 3, pp. 156–160, 2011.

Related works on stability of time-inhomogeneous MJLS

Only sufficient stability **conditions** have been derived for MJLS with time-inhomogeneous Markov chains having TPM arbitrarily varying within a polytopic set of stochastic matrices:

- a **sufficient** condition for **stochastic stability** (SS) in terms of **linear matrix inequality** feasibility problem is provided in
  S. Aberkane, "*Stochastic stabilization of a class of nonhomogeneous Markovian jump linear systems*," *Syst. & Control Lett.*, vol. 60, no. 3, pp. 156–160, 2011.
- a **sufficient** condition for MSS of MJLS with **interval TPM**, which in turn can be represented as a convex polytope, is presented in relation to **spectral radius** in
  S. Chitraganti, S. Aberkane, and C. Aubrun, "*Mean square stability of non-homogeneous Markov jump linear systems using interval analysis*," in Proc. of the 2013 European Control Conf. (ECC), 2013, pp. 3724–3729.

Stability of time-inhomogeneous MJLS

Our contribution:

- We provide a **necessary and sufficient** condition for MSS of discrete-time MJLS with time-inhomogeneous polytopic TPM, which require to decide whether the joint spectral radius of a finite family of matrices is smaller than 1

  Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. On stability of time-inhomogeneous Markov jump linear systems. 55th IEEE Conference on Decision and Control, Las Vegas, US, December 12-14, 2016.

Stability of time-inhomogeneous MJLS

Our contribution:

- We provide a **necessary and sufficient** condition for MSS of discrete-time MJLS with time-inhomogeneous polytopic TPM, which require to decide whether the joint spectral radius of a finite family of matrices is smaller than 1
  Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. On stability of time-inhomogeneous Markov jump linear systems. 55th IEEE Conference on Decision and Control, Las Vegas, US, December 12-14, 2016.
- We provide a **necessary and sufficient** condition for “practical” MSS with bounded disturbances
  Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. On robust stability of time-inhomogeneous Markov jump linear systems. In preparation for submission.

Joint Spectral Radius (JSR)

Introduced by Rota and Strang in **1960**, Joint Spectral Radius (JSR)

- characterizes the **maximal asymptotic growth rate** of the norms of **long products** of matrices taken in a set;
- is subject of intense research due to its role in the study of wavelets, switching systems, approximation algorithms, curve design, and many other topics.

Joint Spectral Radius (JSR)

Introduced by Rota and Strang in **1960**, Joint Spectral Radius (JSR)

- characterizes the **maximal asymptotic growth rate** of the norms of **long products** of matrices taken in a set;
- is subject of intense research due to its role in the study of wavelets, switching systems, approximation algorithms, curve design, and many other topics.

Formally, let $\mathcal{M}$ be a family of complex square matrices, i.e. $\mathcal{M} = \{M_\ell\}_{\ell \in \mathcal{L} \triangleq \{1,\dots,L\}}$, where $M_\ell \in \mathbb{C}^{n \times n}$. Consider the **set $\boldsymbol{\Pi_k(\mathcal{M})}$**, $k \in \mathbb{N}$, be a set of **all possible products** of length k whose factors are elements of $\mathcal{M}$, that is

$$\Pi_k(\mathcal{M}) = \left\{ M_{\ell_k} M_{\ell_{k-1}} \cdots M_{\ell_1} \middle| \ell_1, \dots, \ell_k \in \mathcal{L} \right\}.$$

Joint Spectral Radius (JSR)

Introduced by Rota and Strang in **1960**, Joint Spectral Radius (JSR)

- characterizes the **maximal asymptotic growth rate** of the norms of **long products** of matrices taken in a set;
- is subject of intense research due to its role in the study of wavelets, switching systems, approximation algorithms, curve design, and many other topics.

Formally, let $\mathcal{M}$ be a family of complex square matrices, i.e. $\mathcal{M} = \{M_\ell\}_{\ell \in \mathcal{L} \triangleq \{1,\dots,L\}}$, where $M_\ell \in \mathbb{C}^{n\times n}$. Consider the **set $\boldsymbol{\Pi_k(\mathcal{M})}$**, $k \in \mathbb{N}$, be a set of **all possible products** of length k whose factors are elements of $\mathcal{M}$, that is

$$\Pi_k(\mathcal{M}) = \left\{ M_{\ell_k} M_{\ell_{k-1}} \cdots M_{\ell_1} \middle| \ell_1, \dots, \ell_k \in \mathcal{L} \right\}.$$

Definition. For any matrix norm $\|\cdot\|$ on $\mathbb{C}^{n\times n}$ consider the supremum among the normalized norms of all products in $\Pi_k(\mathcal{M})$, i.e.

$$\hat{\rho}_k(\mathcal{M}) \triangleq \sup_{\Pi \in \Pi_k(\mathcal{M})} \|\Pi\|^{\frac{1}{k}}, \qquad k \in \mathbb{N}$$

The joint spectral radius of $\mathcal{M}$ is defined as

$$\hat{\rho}(\mathcal{M}) \triangleq \lim_{k\to\infty} \hat{\rho}_k(\mathcal{M})$$

Stability of time-inhomogeneous MJLS

Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. On stability of time-inhomogeneous Markov jump linear systems. 55th IEEE Conference on Decision and Control, Las Vegas, US, December 12-14, 2016

Theorem 1. Given a discrete-time MJLS (S) with polytopic TPM, it is **MSS** if and only if **JSR of $\mathcal{A}_L$ is < 1**, where $\mathcal{A}_L$ is a set of all vertices of the polytope associated to the second moment of $x(k)$.

Formally, $\mathcal{A}_L = \{\Lambda_\ell\}_{\ell \in \mathcal{L} \triangleq \{1,\ldots,L\}}$, $\Lambda_\ell \triangleq \left(P_\ell^T \otimes \mathbb{I}_{n^2}\right) \operatorname{diag}\left[\overline{\mathrm{A}}_i \otimes \mathrm{A}_i\right]$

(The JSR of the set of TPMs of the system provided in the illustrative example is $\hat{\rho}(\mathcal{A}_L) > 1$)

Stability of time-inhomogeneous MJLS

Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. On stability of time-inhomogeneous Markov jump linear systems. 55th IEEE Conference on Decision and Control, Las Vegas, US, December 12-14, 2016

Theorem 1. Given a discrete-time MJLS (S) with polytopic TPM, it is **MSS** if and only if **JSR of $\mathcal{A}_L$ is < 1**, where $\mathcal{A}_L$ is a set of all vertices of the polytope associated to the second moment of $x(k)$.

Formally, $\mathcal{A}_L = \{\Lambda_\ell\}_{\ell \in \mathcal{L} \triangleq \{1,\dots,L\}}$, $\Lambda_\ell \triangleq \left(P_\ell^T \otimes \mathbb{I}_{n^2}\right) \operatorname{diag}\left[\overline{\mathrm{A}}_i \otimes \mathrm{A}_i\right]$

(The JSR of the set of TPMs of the system provided in the illustrative example is $\hat{\rho}(\mathcal{A}_L) > 1$)

Theorem 2. Given a discrete-time MJLS (S) with polytopic TPM, unless P = NP, there is **no polynomial-time algorithm** that decides whether it is MSS.

(by reduction from the matrix semigroup stability)

Stability of time-inhomogeneous MJLS

Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. On stability of time-inhomogeneous Markov jump linear systems. 55th IEEE Conference on Decision and Control, Las Vegas, US, December 12-14, 2016

Theorem 1. Given a discrete-time MJLS (S) with polytopic TPM, it is **MSS** if and only if **JSR of $\mathcal{A}_L$ is < 1**, where $\mathcal{A}_L$ is a set of all vertices of the polytope associated to the second moment of $x(k)$.

Formally, $\mathcal{A}_L = \{\Lambda_\ell\}_{\ell \in \mathcal{L} \triangleq \{1,\dots,L\}}$, $\Lambda_\ell \triangleq \left(P_\ell^T \otimes \mathbb{I}_{n^2}\right) \operatorname{diag}\left[\overline{\mathrm{A}}_i \otimes \mathrm{A}_i\right]$

(The JSR of the set of TPMs of the system provided in the illustrative example is $\hat{\rho}(\mathcal{A}_L) > 1$)

Theorem 2. Given a discrete-time MJLS (S) with polytopic TPM, unless P = NP, there is **no polynomial-time algorithm** that decides whether it is MSS.

(by reduction from the matrix semigroup stability)

Theorem 3. The following assertions are equivalent for a system (S):

i. (S) is **mean square stable** (MSS).

ii. (S) is **exponentially mean square stable** (EMSS), i.e.

$$\mathbf{E}[\|x(k)\|^2] \leq \beta \varsigma^k \|x_0\|_2^2 , \qquad \forall x_0 \in \mathbb{C}^n, k \in \mathbb{N}_0 , \beta \geq 1 , 0 < \varsigma < 1 , P_0 , \theta_0 .$$

iii. (S) is **stochastically stable** (SS), i.e.

$$\sum_{k=0}^{\infty} \mathbf{E}[\|x(k)\|^2] < \infty \qquad \forall x_0 \in \mathbb{C}^n, k \in \mathbb{N}_0 , P_0 , \theta_0 .$$

MJLS mathematical framework

We leverage **discrete-time Markov-jump linear systems (MJLS)**. A MJLS is a **switching** linear system where the switching signal is a **Markov chain**. The transition probability matrix (TPM) $P(k)$ of the Markov chain $\sigma(k)$ can be used to model the stochastic process that rules packet losses due to wireless communication.

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,
$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,
$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

<u>Extension 1</u>: In general wireless channels are time-varying and packet loss probability not easy to compute/measure, thus $P(k)$ is unmeasurable and time-varying

<u>Extension 2</u>: We want to control the routing policy: MJLS where $\theta(k)$ is a Markov-decision process

Robust LQR on MJLS with continous and discrete inputs

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,

$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,

$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

where $\theta(k)$ is a Markov-decision process with discrete input $a(k)$

Robust LQR on MJLS with continous and discrete inputs

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,

$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,

$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

where $\theta(k)$ is a Markov-decision process with discrete input $a(k)$

Problem: design an optimal state-feedback control policy $a^*(x(k)) \in A$, $u^*(x(k))$, with $u^*(x(k)) = K^*(k)x(k)$ and $a^*(x(k))$: $\mathbb{R}^n \to A$, minimizing the cost function:

$$J(x(0), u(0), a(0)) = E\left\{\sum_{k=0}^{N-1}(x'(k)Mx(k) + u'(k)Ru(k)) + x'(N)Qx(N)\middle|\aleph_0\right\}$$

Only suboptimal solutions in the literature [Vargas et al., CDC'10]. <u>Our contribution</u>:

Robust LQR on MJLS with continous and discrete inputs

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,

$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,

$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

where $\theta(k)$ is a Markov-decision process with discrete input $a(k)$

Problem: design an optimal state-feedback control policy $a^*(x(k)) \in A$, $u^*(x(k))$, with $u^*(x(k)) = K^*(k)x(k)$ and $a^*(x(k))$: $\mathbb{R}^n \to A$, minimizing the cost function:

$$J(x(0), u(0), a(0)) = E\left\{\sum_{k=0}^{N-1}(x'(k)Mx(k) + u'(k)Ru(k)) + x'(N)Qx(N)\middle|\aleph_0\right\}$$

Only suboptimal solutions in the literature [Vargas et al., CDC'10]. <u>Our contribution</u>:

- Suboptimal LQR, optimal for a specified subset of initial conditions – homogeneous MDP [Di Girolamo et Al., NecSys'15]

Robust LQR on MJLS with continous and discrete inputs

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,

$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,

$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

where $\theta(k)$ is a Markov-decision process with discrete input $a(k)$

Problem: design an optimal state-feedback control policy $a^*(x(k)) \in A$, $u^*(x(k))$, with $u^*(x(k)) = K^*(k)x(k)$ and $a^*(x(k))$: $\mathbb{R}^n \to A$, minimizing the cost function:

$$J(x(0), u(0), a(0)) = E\left\{\sum_{k=0}^{N-1}(x'(k)Mx(k) + u'(k)Ru(k)) + x'(N)Qx(N)\middle|\aleph_0\right\}$$

Only suboptimal solutions in the literature [Vargas et al., CDC'10]. <u>Our contribution</u>:

- Suboptimal LQR, optimal for a specified subset of initial conditions – homogeneous MDP [Di Girolamo et Al., NecSys'15]
- Optimal robust LQR – inhomogeneous MC [Zacchia Lun et Al., submitted]

Robust LQR on MJLS with continous and discrete inputs

$$\begin{cases} x(k+1) = A_{\theta(k)}x(k) + B_{\theta(k)}u(k), \\ x(0) = x_0\,, \\ \theta(0) = \theta_0 \end{cases}$$

$x(k) \in \mathbb{C}^n$ is the state vector,

$\mathrm{u}(k) \in \mathbb{C}^m$ is the input vector, $k \in \mathbb{N}_0$,

$A_{\theta(k)} \in \mathbb{C}^{n\times n}$, $B_{\theta(k)} \in \mathbb{C}^{n\times m}$ are the state and input matrices associated with mode $\theta(k)$

where $\theta(k)$ is a Markov-decision process with discrete input $a(k)$

Problem: design an optimal state-feedback control policy $a^*(x(k)) \in A$, $u^*(x(k))$, with $u^*(x(k)) = K^*(k)x(k)$ and $a^*(x(k))$: $\mathbb{R}^n \to A$, minimizing the cost function:

$$J(x(0), u(0), a(0)) = E\left\{\sum_{k=0}^{N-1}(x'(k)Mx(k) + u'(k)Ru(k)) + x'(N)Qx(N)\middle|\aleph_0\right\}$$

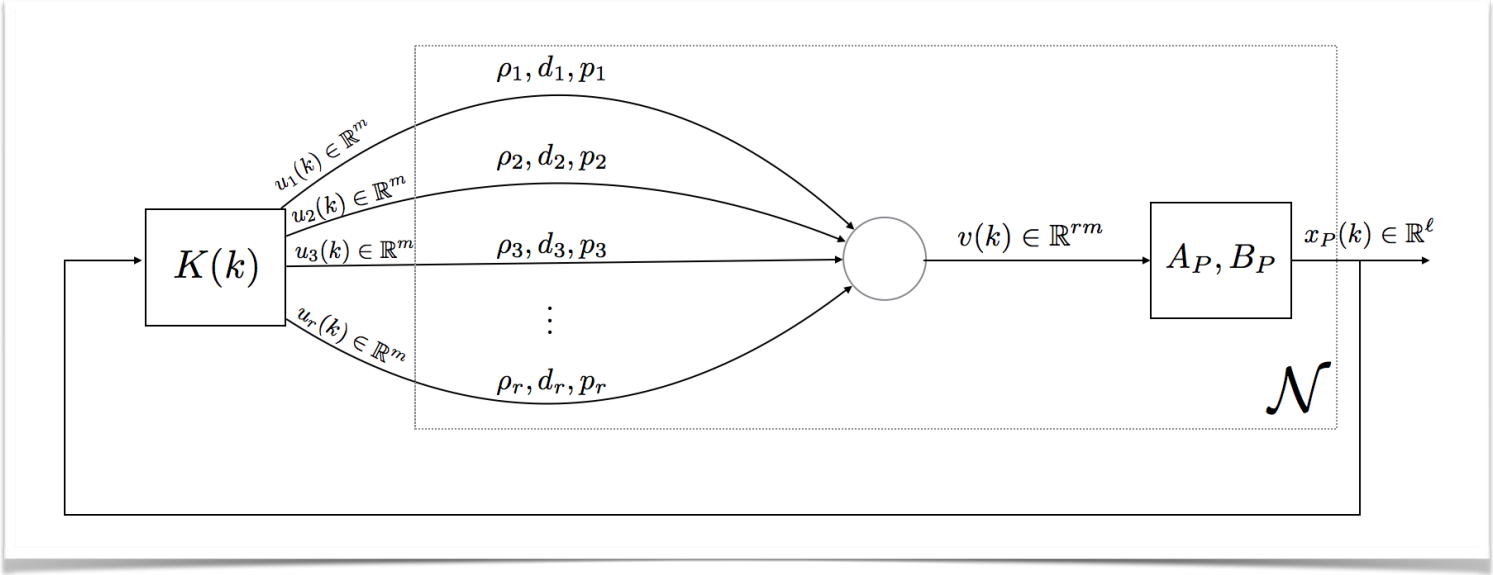
Only suboptimal solutions in the literature [Vargas et al., CDC'10]. <u>Our contribution</u>:

- Suboptimal LQR, optimal for a specified subset of initial conditions – homogeneous MDP [Di Girolamo et Al., NecSys'15]
- Optimal robust LQR – inhomogeneous MC [Zacchia Lun et Al., submitted]
- Optimal robust LQR – inhomogeneous MDP [Zacchia Lun et Al., submitted]

Future work

- Take into account infinite horizon LQR & Stability
- Co-design control and transmission power
- Co-design control and network coding
- Co-design control and tx time (static PETC)
- Develop FDI techniques on our model (CPS security)

References



Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. Robust LQR of time-inhomogeneous Markov jump switched linear systems. In preparation for submission.

Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. Robust LQR of time-inhomogeneous Markov jump linear systems. Submitted.

Y. Zacchia Lun, A. D'Innocenzo, M.D. Di Benedetto. On stability of time-inhomogeneous Markov jump linear systems. 55th IEEE Conference on Decision and Control, Las Vegas, US, December 12-14, 2016.

G. D. Di Girolamo, A. D'Innocenzo, M. D. Di Benedetto. Co-design of controller and routing redundancy over a wireless network. 5th IFAC Workshop on Estimation and Control of Networked Systems (NecSys 2015), Philadelphia PA, September 10-11, 2015

F. Smarra, A. D'Innocenzo, M. D. Di Benedetto. Approximation methods for optimal network coding in a multi-hop control network with packet losses. 14th European Control Conference (ECC 2015), Linz, Austria, July 15-17, 2015

European Embedded Control Institute

M14 – L'AQUILA (ITALY) 18/04/2017 – 21/04/2017	*Modeling, analysis and design of wireless sensor and actuator networks*

Alessandro D'Innocenzo
Department of Information Engineering,
Computer Science and Mathematics
Center of Excellence DEWS
University of L'Aquila, Italy
http://people.disim.univaq.it/~alessadin/
alessandro.dinnocenzo@univaq.it

Carlo Fischione
Department of Automatic Control
School of Electrical Engineering
KTH Royal Institute of Technology, Sweden
https://people.kth.se/~carlofi/
carlofi@kth.se

Abstract of the course:

The massive deployment of smart and wirelessly interconnected devices in cyber-physical systems such as smart grids, networks, and intelligent transport systems, is providing extensive information from the physical world through distributed sensing mechanisms. With the emergence of low-cost controllers/actuators that can be potentially embedded in everything (e.g., vehicles, robots, buildings, human body), the sensed information can be utilized to act and perform control, estimation and monitoring at an unprecedented scale. This is demanding the development of fundamental design principles for Wireless Sensor Network and Actuators Networks (WSAN) so to reliably and certifiably observe the physical world, processing the data, making decisions based on the sensor observations and performing appropriate control actions. In addition, due to the vulnerability of wireless networks to cyber-physical malicious attacks, security is a primary concern for WSAN.

This course is devoted to the study of WSANs: a co-design approach of Control Systems and Networks supporting control applications over WSAN will be presented. In particular, we will discuss the following scenarios:

Control-aware network design, when the protocol parameters can be adapted by an optimization problem whose objective is a network utility, and the constraints are the reliability and latency of the messages as requested by a control application;

Network-aware controller design, when the controller can be adapted to guarantee some control specifications robustly with respect to networking non-idealities and malicious attacks;

Cross-layer optimization, where desirable signaling between communication layers enable simultaneous computation of control actions and networking configuration to improve the overall system performance;

Distributed optimization, including privacy constrains, where the nodes of a WSAN jointly cooperate to solve network global optimization problems that may possibly include privacy constraints.

Topics:

- Mathematical modeling of networks for WSANs
- Mathematical modeling of closed-loop systems over WSANs
- Robust, resilient and secure control co-design over WSANs
- Distributed optimization over WSANs
- Discussion of open problems and opportunities for research