Cyber Physical Systems

a perspective

Johan Lukkien
System Architecture and Networking
Eindhoven University
Agenda

• CPS definition
• What’s new and what’s special, with examples
• Research directions from literature
CPS: Definitions @ April 2013

- UC Berkeley (April 2013). “Cyber-Physical Systems (CPS) are integrations of computation, networking, and physical processes.”

- “Cyber-physical systems (CPS) are next-generation embedded systems featuring a tight integration of computational and physical elements.”

CPSNA 2013
The 1st International Conference on Cyber-Physical Systems, Networks, and Applications
Taipei, Taiwan August 19 - 20, 2013

Annotated concept map from http://cyberphysicalsystems.org/
“Cyber-physical systems (CPS) are physical and engineered systems whose operations are monitored, coordinated, controlled and integrated by a computing and communication core.

This intimate coupling between the cyber and physical will be manifested from the nano-world to large-scale wide-area systems of systems.

And at multiple time-scales.”
Two more definitions

• “A cyber-physical application is a computer system that processes and reacts to data from external stimuli from the physical world and make decisions that also impact the physical world”
  (cited from White et.al.; R&D challenges and solutions for mobile cyber-physical applications and supporting Internet services, Journal of Internet Services and Applications, May 2010, Volume 1, Issue 1, pp 45-56)

• “A cyber-physical system (CPS) is a system featuring a tight combination of, and coordination between, the system’s computational and physical elements.”
  – Wikipedia, May 2013
So, CPS is…

• something that’s more than just control through embedded electronics

• a CPS impacts on the world

• we don’t have them yet

• it will come and be very important

• hm
So, what is CPS?

- **Eq. 1:** CPS = Computing + Control
  - (“Embedded 2.0”)
- **Eq. 2:** CPS = Embedded Systems + Internet of Things
  - (“Embedded meets Ubiquitous Sensing and Communication”)
- **Eq. 3:** CPS = Embedded Systems + Internet Services
  - (“Embedded meets Big Data”, “Virtual embedding”)
- **Eq. 4:** CPS = Physical world + Sense/Act. + Control + Computation + Communication

- **The novelty is to consider them all together**
  - a theory of CPS, in which 5 aspects influence each other
  - (perhaps 6, if you include data)
Two directions

1. High-performance control systems

2. Deep penetration of sensing and actuation into the physical world
   - Internet of Things
     - unified protocol and addressing scheme between any pair of (electronically enriched) objects
     - service discovery, (resource) management
     - large-scale heterogeneous data processing

(CPS) challenges of both:
- robustness under uncertainty and complexity
- much tighter integration

From: *Network QoS Management in Cyber-Physical Systems*, Feng Xia et. al
Examples (from STW)

- professional printing
- electron microscopy
- subcutaneous sensing
- server farms
- platooning
- medical imaging
- chip fabrication
Example (from the ACCUS project)

- Integration of independent urban subsystems
- New applications…
  - …operating across different subsystems
  - …using services of different subsystems
- Examples:
  - emergency response:
    - use traffic lights, street lighting and vehicle guidance
  - energy optimization
Example (from the SENSAFETY project)

Twitter and facebook are sensors…
Smartphones are sensors..
Fences have sensors…
Camera’s everywhere…

In-network processing to reduce traffic
Latency vs bandwidth vs storage
Privacy vs safety

Backend combines information from different networks (ad-hoc, dedicated, internet)
So, CPS is newish (since ~2008)

- What makes it interesting to lift the strict separation of domains (separation of concerns!)?

1. the business of exponential growth
2. higher efficiency, lower cost
3. integration of systems rather than components
4. networked control, in multiple layers
5. ubiquitous systems and wireless technology
The business of exponential growth

- Moore’s law
  - # transistors / area doubles every two years
  - (… at the same price)
The business of exponential growth

- Kryder’s law
  - storage capacity in \# bits / surface doubles every 18 months
  - 14TB @ $40 in 2020
The business of exponential growth

- Nielsen’s law
- a high-end-user’s connection speed grows by 50% annually
The business of exponential growth

• Gilder’s ‘law’: the total bandwidth of the Internet triples every year

• Metcalfe’s ‘law’: the value of a (telecommunication) network is proportional to the square of connected users
TED talk Peter Diamandis: Abundance is our future

- Average human lifespan: ↑ 2x
- Avg. per-capita Income: ↑ 3x
- Childhood Mortality: ↓ 10x

Cost of Food: ↓ 10x
Cost of Electricity: ↓ 20x
Cost of Transportation: ↓ 100x
Cost of Communications: ↓ 1000x

Technologies riding Moore’s Law

1. Infinite Computing
2. Sensor & Networks
3. Robotics
4. 3D Printing
5. Synthetic Biology
6. Digital Medicine
7. Nanomaterials
8. Artificial Intelligence

ABUNDANCE:
HEALTH & EDUCATION

TU/e
Technische Universiteit Eindhoven
University of Technology

Eindhoven Institute for Research on ICT
Consequences to the embedded domain

- Enormous processing capacity in control loops
  - good! we want higher sampling rates and more control loops
- Enormous amounts of data to act upon
  - coming from extensive sensing, high-resolution images, higher rates
- Latency requirements and storage restrictions lead to complex challenges
  - distributed processing, reducing data volume and response latency
    - using knowledge of physical processes and application requirements
    - e.g. to determine order, and quality of processing
  - network as part of the control system
Higher efficiency, lower cost

- Integrate different applications on the same hardware
- must still guarantee spatial and temporal protection
- mixed-criticality systems
Sharing by multiple critical applications

Tasks, located in arbitrary applications, may share resources
Higher efficiency, lower cost

• “Control on demand” saves resources and energy
  • corrective action based on state or output of system
    – event-based control, stability-dependent deadlines:
      – determine interval in which action must be taken
      – determine consequences of delayed action
    – self-triggered control
Higher efficiency, lower cost

- Technology is not perfect
  - ‘perfect’ can be extremely costly, or impossible
    - e.g. wireless communication, with inherent limitations, is nevertheless used in control systems
- analyze impact of relaxed assumptions on physical system
  - packet loss, delay
- *Trade quality of service versus quality of control*

picture by Maurice Heemels
Higher efficiency, lower cost

- The highest quality outcome of subsystems is not always required
  - (response-)time versus quality trade-off

- An “any-time” algorithm
  - provides a valid solution any time when it is interrupted
  - increases quality over time

- Examples:
  - Newton-Raphson
  - Trajectory estimation
  - Video decoders
Integrate systems instead of components

• Feedback control loops spanning multiple layers, networks and time scales
  • integration of *systems* rather than of (dedicated) *components*
    – “Systems of Systems”
  • leading to *resource sharing* within systems by applications with distinct importance
    – “mixed criticality” (see before)

• ‘System perimeter’ much less clear
  • systems must work with *incomplete knowledge*, and uncertainty about their context
  • “virtual embedding”: use Internet services in embedded systems
1. Brake pedal pushed

2. Pressure passed to the brake fluid

3. Wheel disc brakes squeezed

4. If the brake pedal is pushed too hard, the wheel will lock → a sensor detects this and notifies the controller

5. Controller releases the pressure on the discs by releasing some brake fluid in a container

6. The fluid is pumped back to repeat the pressure on the discs

7. Entire process is repeated about 15 times/sec

(by courtesy of Damir Isovíc Mälardalen University)
ABS is just one subsystem
Car is part of traffic system: V2V, V2I
Car is part of traffic system: V2V, V2I
Pattern: local-global control cycle

- Collect CAN signals and location data from individual cars

- Interpret CAN signals and location:
  - road condition
  - traffic situation
  - weather
  - accidents

- Global feedback
  - traffic routing
  - planning of maintenance

- Local feedback
  - individual guidance
  - perhaps: adaptation of in-vehicle systems, or of V2V
• Can we avoid sending (all) data to a central location?

• levels of aggregation in the data?

• can we keep local what is required locally?

• intrusion protection, trustworthiness of data and control
Some literature

- Cyber-Physical Systems: A New Frontier
  - Lui Sha, Sathish Gopalakrishnan, Xue Liu, and Qixin Wang
    2008 IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing

- Cyber Physical Systems: Design Challenges
  - Edward Lee, 2008 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC)
Advances: *CPS Systems Science*

- A *robust* design flow
  - … combining computational and physical aspects
  - … with multi-scale dynamics
  - … and integrated wired and wireless networking

- Management of systems’ resources, including e.g. mass, energy and information
Advances: Ubiquitous, trustworthy computing

• Adequate abstractions:
  • Distributed, real-time computing
  • Real-time group communication
  • Dynamic topology management in mobile systems

• Programming models
  • that include timing and deal with events in time and space
  • have real-time and concurrency abstractions
    – e.g. bands of simultaneity
  • configurable / tunable software components
Advances: Robustness, safety and security

- Coherent set of metrics that capture uncertainty, faults / error / failures and security
  - ...to be included in the software / system design process
  - ...and in algorithms (e.g. anytime algorithms, event-based control)

- A philosophy of a small, correct kernel
  - formally specified and verified against a sound and complete environment model
  - against which guarantees for (safety-)critical services are given

- Self monitoring and feedback control within software systems
Advances: composition

- Composition (of software components) includes service quality and resource use
  - Resource specification at interfaces
  - Resource aware behavior

- Derive correctness and QoS properties from:
  - Architecture (logical, physical)
  - Protocols, mappings
  - Component properties

- Theory of composition
  - Language support, model-driven (DSL)
  - Integrating time-triggered and event-triggered
Advances: *trust, trustworthiness*

- (tools to) Visualize and analyze a CPS within its broader context (social, technical)
- Transparent privacy protection
  - concepts, analytical and engineering framework
- Cohesive, conceptual, predictable, transparent from user’s perspective
Conclusion: techniques / domains

- Systems of systems integration
  - integration of independent subsystems with *architectural diversity, uncorrelated requirements, competition of control*
  - concerns of *interoperability and emergent properties*

- Mixed Criticality
  - *a mixed-critical system is an integrated suite of hardware, operating system and middleware services and application software that supports the execution of safety-critical, mission-critical, and non-critical software within a single, secure compute platform* (*from [http://www.cse.wustl.edu](http://www.cse.wustl.edu)*, research agenda for Mixed-Criticality Systems)

- Real-time techniques
  - event-based control, sharing

- Anytime algorithms
Conclusion

• CPS
  • removes the distinction between specializations
  • calls for an integrated approach, which may lead to new methods

• In order to retain separation of concerns, new approaches to interfaces between systems and components need to be investigated
  • particularly, specification of extra-functional properties
  • and policies for data sharing, control

• .... and these must be supported by proper theory