Towards Sentient Chips:
Self-Awareness through On-Chip Sensemaking

Nikil D. Dutt

Center for Embedded and Cyber-Physical Systems (CECS)
University of California, Irvine
dutt@uci.edu
http://www.ics.uci.edu/~dutt
https://duttgroup.ics.uci.edu

Joint work with Santanu Sarma and Dutt Research Group
Research Partially Supported by the NSF Variability Expedition Project

sentient

\[ˈsenCH(ə)nt/ \]

adjective

adjective: sentient

able to perceive or feel things.
"she had been instructed from birth in the equality of all sentient life forms"
synonyme: (capable of) feeling, living, live; More

Origin

LATIN

sentire \rightarrow \text{sentient-} \rightarrow \text{sentient}
feeling \rightarrow \text{early 17th century}

early 17th century: from Latin sentient- 'feeling,' from the verb sentire.
Towards Sentient Chips: **Self-Awareness** through On-Chip

Self-awareness

From Wikipedia, the free encyclopedia

Not to be confused with Self-concept, Self-consciousness, Self-perception, or Self image.

This article has multiple issues. Please help improve [hide]

* This article may require cleanup to meet Wikipedia's quality standards. (March 2005)
* This article needs attention from an expert on the subject. (May 2005)

Self-awareness is the capacity for introspection and the ability to recognize oneself as an individual separate from the environment and other individuals.

Contents [hide]

1. In philosophy
2. In biology
   2.1 Animals
   2.2 Evolution
   2.3 Neurological basis

The mirror test is a simple measure of self-awareness.

Self-Awareness & Adaptation in Biology

[David Gallo: Underwater astonishments, TED]
Self-Awareness vs Context-Awareness

• **Self-Awareness** [Hinchey2006]: System is aware of its *self states and behaviors*

• **Context-Awareness** [Parashar 2005]: System is aware of *context – i.e., its operational environment*

  • *Self-configuring* -> capability of reconfiguring automatically
  • *Self-healing* [Robertson2005] -> *self-diagnosing and self-repairing*
  • *Self-optimizing* -> capability of self-tuning or *Self-adjusting*
  • *Self-protecting* -> capability of detecting dangerous outcomes (e.g. security breaches) and recovering from their effects

---

**A Hierarchical View**

- **General Level**
  - Self-Adaptiveness
  
- **Major Level**
  - Self-Configuring
  - Self-Optimizing
  - Self-Healing
  - Self-Protecting

- **Primitive Level**
  - Self-Awareness
  - Context-Awareness

Fig. 1. Hierarchy of the self-* properties.

[SALEHIE 2009, TAAS]
Self-Reflection

- **Self-Reflection**: 
  - Ability to create a *self-model*
  - Ability to model their own body/structure (usually known as *self-modeling*)
  - Ability to model their own *behavior*
  - **Metacognition capacity**: "models one's own thinking", 'think about thinking'
  - System with two/multiple minds: one being modeled and other doing *modeling*
  - Control Systems Theory: also called Dynamical System identification

Towards Sentient Chips: Self-Awareness through On-Chip Sensemaking

**Sensemaking**

From Wikipedia, the free encyclopedia

*Sensemaking* is the process by which people give meaning to experience. While this process has been studied by other disciplines under other names for centuries, the term "sensemaking" has primarily marked three distinct but related research areas since the 1970s: Sensemaking was introduced to Human–computer interaction by PARC researchers Russell, Steffl, Pirolli and Card in 1993, to information science by Brenda Dervin, and organizational studies by Karl Weick.

In information science the term is most often written as "sense-making." In both cases, the concept has been used to bring together insights drawn from philosophy, sociology, and cognitive science (especially social psychology). Sensemaking research is therefore often presented as an interdisciplinary research.
Self-Assembling Robots (Sensemaking)

Outline

- Self-Awareness, Sentience, Sensemaking
- Cyber-Physical Systems-on-Chip (CPSoC)
- CPSoC Exemplars and Prototype
- Wrap-up
Outline

• Self-Awareness, Sentience, Sensemaking

• Cyber-Physical Systems-on-Chip (CPSoC)

• CPSoC Exemplars and Prototype

• Wrap-up

What are Sentient Chips?

• Sentient chips
  – *Construct model* of behaviors and environment using sensor data
  – Achieve *self-awareness* through on-chip sensors and monitors
    • Experience phenomena
    • Aware of state and behavior
  – The ability to *introspect*
  – *Adapt behavior* based on model of external and internal environment
Why On-Chip Self-Awareness?

- Tremendous variation in applications, environment, platforms
- Chips must adapt to Dynamic Performance, Power, Resilience, Security,…
  - See Radar chart (Kiviat graph) examples
- Provide Guarantees
- Exploit trade-offs in several dimensions

Ideal Radar Chart?
Performance Driven

Energy/Power Driven

Reliability Driven

Security Driven

QoS Combination

Reality
What we want: QoS Combination

Need Adaptability and Autonomy
A Perspective on the Future: Cyber-Physical Systems

Cyber-physical systems are physical, biological, and engineered systems whose operations are integrated, monitored, and/or controlled by a computational core. Components are networked at every scale. Computing is "deeply embedded" into every physical component, possibly even into materials. The computational core is an embedded system, usually demands real-time response, and is most often distributed. The behavior of a cyber-physical system is a fully-integrated hybridization of computational (logical) and physical action.

Examples of cyber-physical systems include micro- and nano-scale cyber and physical materials, controlled components, cooperating medical devices and systems, next-generation power grid, future defense systems, next-generation automobiles and intelligent highways, flexible robotic manufacturing, next-generation air vehicles and airspace management, ....

Courtesy: Dr. Helen Gill, NSF
What’s a Cyber-Physical System-on-Chip (CPSoC)?

◆ Cyber-Physical System
  - Cyber = Embedded Systems and IT
    - HW, SW, Networking/Communication
  - Physical = Environment
    - Sensing, Actuation, Control

◆ System-on-Chip (SoC):
  - Many-core computing platform: processors, memories, interconnects

◆ Cyber-Physical System-on-Chip (CPSoC):
  - Sensor-actuator rich adaptive SoC platform
    - Self-awareness
    - Predictive modeling and learning

CPSoc Vision: Across Design Features

CPSoc provides opportunity to improve multiple design dimensions (in addition to performance)
CPSSoC Vision: Across Applications

CPSoC aims to adapt across a wide range of dynamic applications to yield acceptable QoS
Cyber-Physical System-on-Chip (CPSoC)

- **Cross-Layer Virtual and Physical Sensing & Actuation**
  - Sensor fusion and Actuation
    - Combine hardware and software sensors

- **Self-Awareness and Adaptation**
  - Combines *Simple* and *Self-Aware* adaptations
  - A reflexive (Observe- Decide-Adapt) architecture to achieve closed loop system control

- **Predictive Modeling & Learning**
  - Dynamic characterization of platform variability across multiple levels of the system stack.
Cross-Layer Physical/Virtual Sensing & Actuation

Examples of Virtual Sensors and Actuators Across Layers of CPSoC

<table>
<thead>
<tr>
<th>Layers</th>
<th>Virtual/Physical Sensors</th>
<th>Virtual/Physical Actuators</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application</td>
<td>Execution Time, Workload Power, Energy,</td>
<td>Loop perforation Algorithmic Choice</td>
</tr>
<tr>
<td>Operating System</td>
<td>System Utilization Peripheral States</td>
<td>Task Allocation, Scheduling, Migration, Duty Cycling</td>
</tr>
<tr>
<td>Network/Bus Communication</td>
<td>Bandwidth; Packet/Flit status; Channel Status, Congestion, Latency</td>
<td>Adaptive Routing Dynamic Bandwidth Allocation Ch. no and direction</td>
</tr>
<tr>
<td>Hardware Architecture</td>
<td>Cache misses, Miss rate; access rate; IPC, Throughput, ILP/MLP, Core asymmetry</td>
<td>Cache Sizing; Reconfiguration, Resource Provision Static/Dynamic Redundancy</td>
</tr>
<tr>
<td>Circuit/Device</td>
<td>Circuit Delay, Aging, leakage Temperature, oxide breakdown</td>
<td>DVFS, DFS, DVS ABB, Clock and Power-gating</td>
</tr>
</tbody>
</table>
CPSoC Basic Computational Block

CPSoC Computational Platform

- Sensor/Actuator-rich SoC fabric
  - OCSA: On-Chip Sensing and Actuation unit
- NoC overlay (or separate network)
- SW enabled sensors & actuators
- Adaptive control of platform resources
**CPSoC Hardware Fabric**

**Homogeneous**

**Heterogeneous**

*Distributed resources supporting homogeneous or heterogeneous or mixed fabric*

---

**CPSoC HW/SW Stack**
CPSoC HW/SW Stack

Cross-Layer Physical/Virtual Sensing & Actuation
Cross-Layer Physical/Virtual Sensing

- Many restrictions in physical deployment of sensors and test structures in MPSoCs:
  - Resource constraints
    - e.g., Area, Power
  - Limited number, resolution, accuracy, range
  - Placement Restrictions
  - Complexity of sensing and observation structures
  - Inaccessibility or inability of direct measurement
  - Prohibitive cost

*Virtual Sensing is a Indirect Computational Approach to overcome several sensing limitations*

Example Virtual Power Sensing with few Thermal Sensors
Example Virtual Power Sensing with few Thermal Sensors

Average Power of Each Blocks

- **Power, Watts**
- **Actual**
- **Estimated**

Cyber-Physical System-on-Chip (CPSoC)

- **Cross-Layer Virtual and Physical Sensing & Actuation**
  - Sensor fusion and Actuation
    - Combine hardware and software sensors

- **Self-Awareness and Adaptation**
  - Combines *Simple* and *Self-Aware* adaptations
  - A reflexive *Observe- Decide-Adapt* architecture to achieve closed loop system control

- **Predictive Modeling & Learning**
  - Dynamic characterization of platform variability across multiple levels of the system stack.
MPSoC with Simple Adaptation

Sentient Chips (Self-Awareness): CPSoC

[Sarma14, CODES+ISSS14]
CPSoC Context

CPSoC: Application-layer input
CPSoC: Two Adaptation Loops

1. CPSoC Simple Adaptation
2. CPSoC Self-Aware Adaptation

Adaptive, Reflexive CyPhy Middleware

- Self-Aware Linux with CyPhy Middleware
- Middleware Layer incorporates adaptation policies
Outline

• Self-Awareness, Sentience, Sensemaking

• Cyber-Physical Systems-on-Chip (CPSoC)

• CPSoC Exemplars and Prototype

• Wrap-up

Sample Application and Use cases

• Energy Efficiency (Throughput/Power)
  – Dynamic Workloads
  – Opportunistic Load balancing
  – Adaptive Scheduler
  – Evolutionary Approach

• Thermal-Aware Performance
  – Dynamic/Adaptive Parallelization
  – Heterogeneous Architecture
  – Adaptive Scheduling

• Aging and Resilience
  – Opportunistic Allocation
  – Duty cycling of Active and Resting periods
Energy Efficiency Improvement

Goal:
- Energy Efficiency

51% Average Improvement for Quad-core
Thermal-Aware Performance

**Goal:**
- Improve throughput under max temp & power constraint

For same power consumption and peak temperature, throughput improved
- 4 core
- 20 core, equal area

Throughput improved by 70% -300%
For same power & peak temp
Thermal-Aware Performance

Throughput improved for same power and peak temperature!

Thermal-Aware Performance

For same power consumption and peak temperature, throughput improved
CPSoC FPGA Prototype

- Goal: validate simulation studies for task migration, temperature, wear-out, etc.

- Platforms: Virtex 5 and Virtex 6
- Processor Core: SPARC/Leon 3, Leon 2, S1
- No Of Processor: 2-8
- NoC: Mesh Connected, 200-800 MBPS Bandwidth
- On-chip Memory: 16-64kB per core
- External Memory: 4 GB of DRAM
- Sensors:
  - Ring Oscillators: 20-50
  - Thermal: 20-40
  - Aging: 10-20
  - Razor/EDS: 20-50
- OS: Linux 2.x/3.x

Xilinx Vertex 6 Board

[Sarma14, RSP14]

FPGA Library for CPSoC

Develop an FPGA library to construct CPSoC

[Sarma14, RSP14]
CPSoC Multi-FPGA Distributed Platform

[Sarma13, CECS TR]

CPSoC Simulation Framework

[Sarma14, RSP14]
Self-* Computing: Related Efforts

- Autonomic Computing (IBM)
- SEEC (MIT & Milano)
  - Software centric/focused adaptation with homogeneous arch
  - Uses ODA loop for feedback control
- Invasive Computing (Erlangen & KIT)
  - Adaptive use of computing platform resources
  - Distributed management
  - No Self-modeling and system behavior identification

Self-Awareness in Software Systems (IBM’s Autonomic Computing)

[Autonomic Computing, IBM]

[Kephart2003]
Invasive Computing (Erlangen/KIT)

- **Definition: Invasive Programming** denotes the capability of a program running on a parallel computer to request and temporarily claim processor, communication and memory resources and to be capable to subsequently free these resources again.
Self-* Computing: Related Efforts

- Autonomic Computing (IBM)
- SEEC (MIT & Milano)
  - Software centric/focused adaptation with homogeneous arch
  - Uses ODA loop for feedback control
- Invasive Computing (Erlangen & KIT)
  - Adaptive use of computing platform resources
  - Distributed management
  - No Self-modeling and system behavior identification

Outline

- Self-Awareness, Sentience, Sensemaking
- Cyber-Physical Systems-on-Chip (CPSoC)
- CPSoC Exemplars and Prototype
- Wrap-up
Key Take-Aways

**CPSoC**: First step towards **Sentient Chips**

Key CPSoC features:

- *Cross-Layer Virtual and Physical Sensing & Actuation*
  - Combine hardware and software sensors across multiple layers
Key Take-Aways

**CPSoC:** First step towards **Sentient Chips**

Key CPSoC features:

- **Self-Awareness and Adaptation**
  - *Simple* and *Self-Aware* adaptions
  - Adaptive, reflexive architecture (*Observe- Decide-Adapt*)

- **Predictive Modeling & Learning**
  - Dynamic platform characterization across multiple levels

---

Key Take-Aways

**CPSoC:** First step towards **Sentient Chips**

Key CPSoC features:

- **Cross-Layer Virtual and Physical Sensing & Actuation**
  - Combine hardware and software sensors

- **Self-Awareness and Adaptation**
  - *Simple* and *Self-Aware* adaptions
  - Adaptive, reflexive architecture (*Observe- Decide-Adapt*)

- **Predictive Modeling & Learning**
  - Dynamic platform characterization across multiple abstraction levels
Quo Vadis, Sentient Chips?

Good? or Evil?

Professor’s Worst Nightmare:
MOOCs + Sentient Robots
Acknowledgements

- Santanu Sarma
  - Lead PhD student on CPSoC theme

- Other members of Dutt Research Group
  - Tiago Mück, Majid Shoushtari, Bryan Donyanavard, Hossein Tajik, Abbas BanaiyanMofrad, Roger Hsieh, Jurngyu Park

- Collaborating Faculty:
  - UCI: Profs. Alex Nicolau, Nalini Venkatasubramanian
  - UCLA: Prof. Puneet Gupta

- NSF Variability Expedition Project Team
  - www.variability.org
Key References

- [RoboBee] http://robobees.seas.harvard.edu/