UNIVERSITÄT
KLAGENFURT

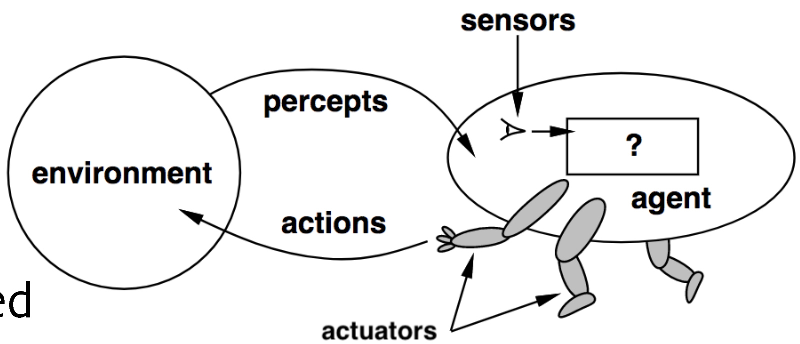
Self-Awareness for Autonomous Systems

Bernhard Rinner

AIDA, January 11, 2022

Autonomous Systems (AS)

- Intelligent autonomous agents
  - Embodied sense-process-act cycle fusing data from multiple sensors
  - Reasoning based on models, provided a-priori or learned from experience
  - Autonomy strongly influenced by models and capabilities (resources + reasoning)



[Norvig, Russel. *Artificial Intelligence*. PrenticeHall, 2003]

- Key application: robotics (real or virtual robots)
- Networked autonomous systems
  - Single, independent to multiple, collaborative systems
  - Closed-loop sense-process-communicate-act cycle
  - Applications: CPS, IoAT, swarm robotics [1]

[1] David Cearley, Brian Burke. Top 10 Strategic Technology Trends for 2019. *Gartner report*

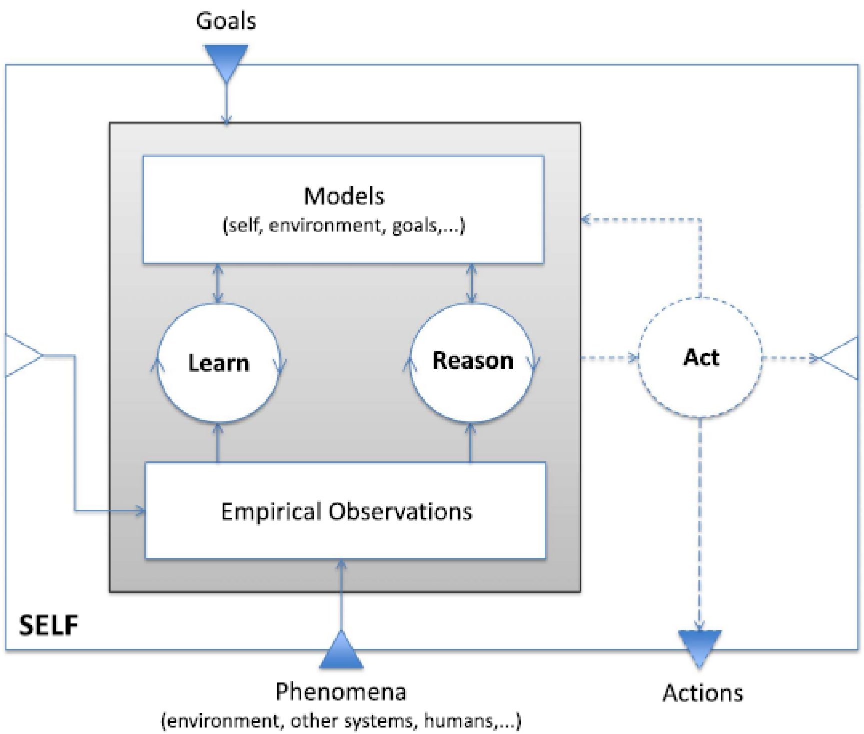
Self-Awareness (SA)

- Concept borrowed from psychology and cognitive science
  - Described as capabilities of biological or artificial agents
  - Has some knowledge about “itself” based on its own senses and internal models
  - Knowledge may take on different forms resulting from perceptions of internal and external phenomena
  - Agent exploit this knowledge to react appropriately (behavior)
- SA means different things in different contexts
- Focus on SA in computational context
  - Able to learn and exploit models of itself and its environment
  - Act in accordance with high-level goals
  - Adopted methods and algorithms from different disciplines

[U. Neisser. The Roots of Self- Knowledge: Perceiving Self, It, and Thou. *Annals of the New York Academy of Sciences*, 1997.]

Self-Awareness Architecture

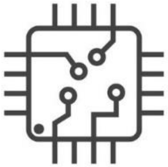
- (Computational) framework
  - Analyzing observations
  - Modelling
  - Reasoning and decision making
- Various approaches
  - Abstraction level
  - Innate knowledge and capabilities
  - Application domain
- Representation of “self”
  - Internal (proprioceptive) vs. external perception and actuation
  - Reasoning about own decision making



[Kounev et al. The notion of self-aware computing. In *Self-Aware Computing Systems*, Springer, 2017]

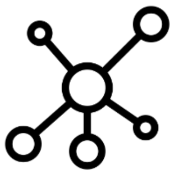
Examples where we find SA Concepts

- Hardware components
  - Monitor state of a system-on-chip and perform adaptations based on changes in environment

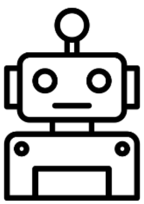


- Software systems
  - Design self-aware and self-adaptive software system considering architectural patterns

- Networks
  - Optimize of quality-of-service, energy and security in networks



- Robots
  - Explore tradeoff between collaborative localization and navigation



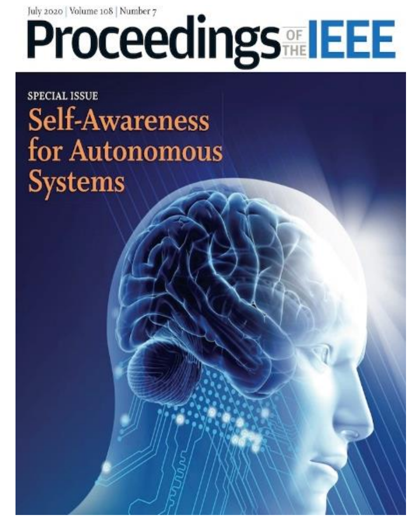
- Acoustic perception and sound generation
  - Perform active sensing for ego noise and acoustic scene analysis

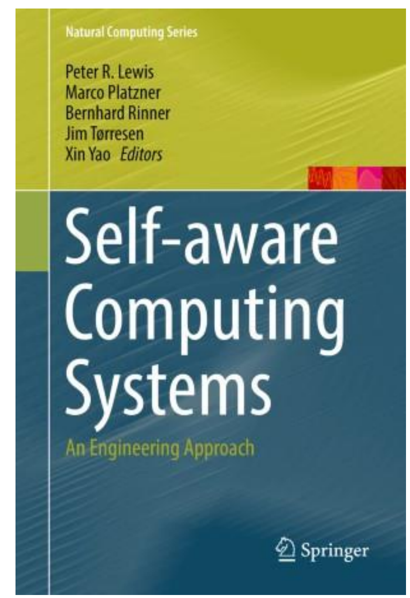
Agenda

1. SA Concepts
   - Architectures, models and capabilities

2. SA Techniques
   - Modelling techniques and reasoning

3. Challenges
   - Discuss limitations and open challenges





[Dutt, Regazzoni, Rinner, Yao. Self-Awareness for Autonomous Systems (special issue). *Proceedings of the IEEE*, 108(7), 2020]

[Lewis, Platzner, Rinner, Torresen, Yao (Eds.). *Self-aware Computing Systems: An Engineering Approach*. Springer, 2016]

SA Concepts

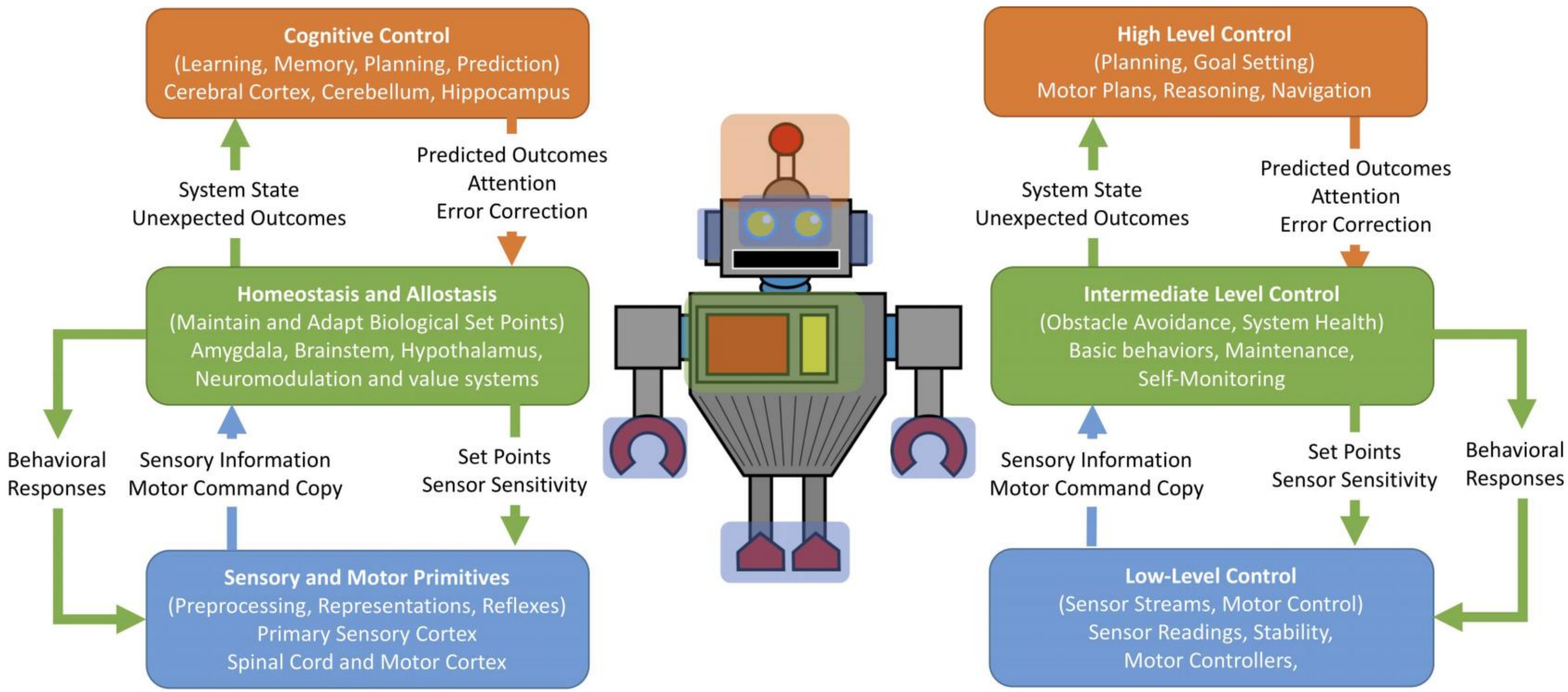
Architectures, models and capabilities

Key Terminology

- Private and public self-awareness to obtain knowledge on
  - Internal phenomena via internal sensors (i.e., internal state)
  - External phenomena via external sensors (i.e., environment)
- Levels of self-awareness
  - Different capabilities for obtaining and exploiting knowledge
  - Stimulus-, interaction-, time-, goal- and meta-self-awareness
- Collective self-awareness
  - SA capabilities of group of agents where knowledge may be distributed
- Self-expression
  - Capability concerned about the behavior (decision making)
  - To emphasize difference to knowledge obtainment ("self-awareness")

[Lewis, Platzner, Rinner, Torresen, Yao. *Self-aware Computing Systems: An Engineering Approach*. Springer, 2016]

Neurobiological inspired SA System



Terms derived from neuroscience mapped to autonomous robots

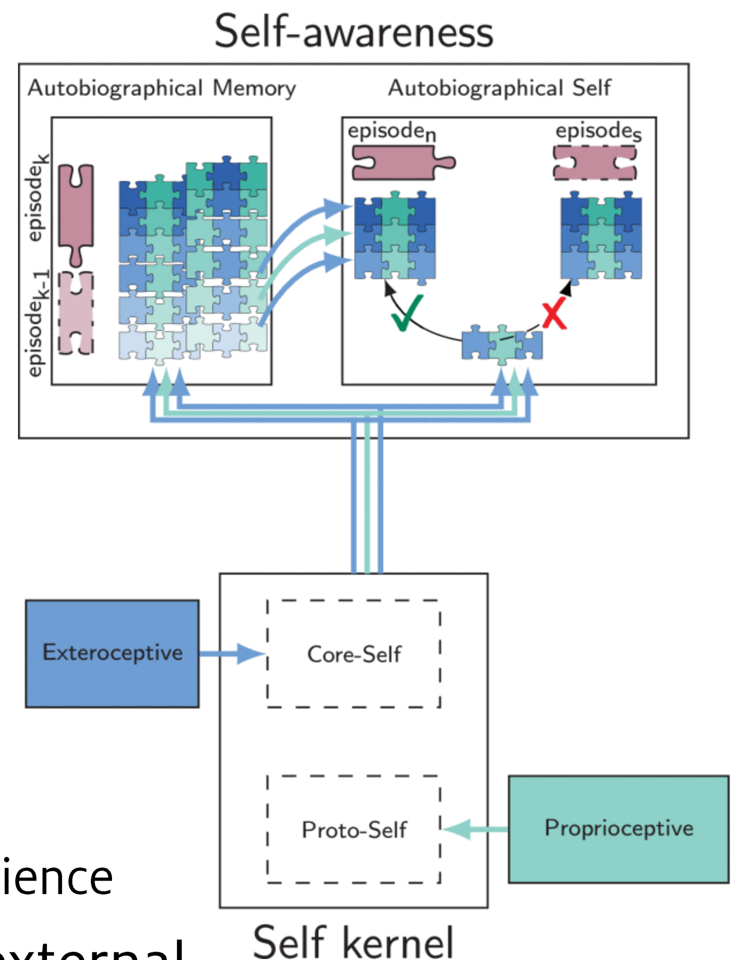
- Innate preferences and behaviors/reflexes basis for exploration
- Homeostasis and allostasis as permanent complementary control processes
- Cognitive control plans and executes long-term strategies

[Chiba and Krichmar. Neurobiologically Inspired Self-Monitoring Systems. *Proceedings of the IEEE*, 108(7):976-986, 2020]

Damasio's Human Conscience Model

UNIVERSITÄT KLAGENFURT

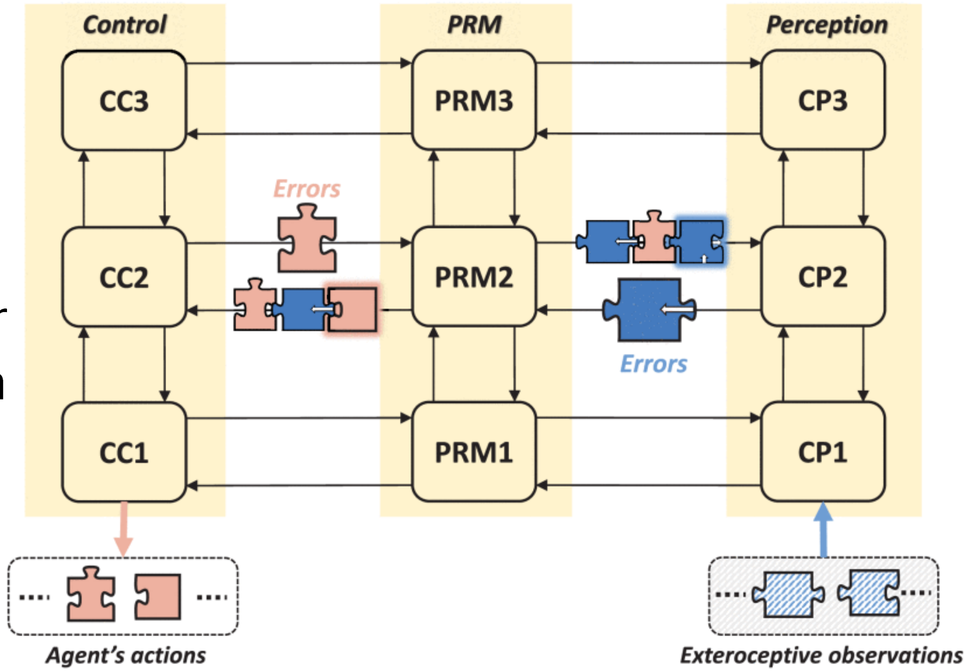
- Autobiographical memory (AM)
  - Memorizing models of learned episodes
  - Episodes encoded by neural vocabulary (proprio- and exteroceptive information)
- "Dispositional units" as information
  - Exteroceptive data contextualized to proprioceptive data, and vice versa
  - Represent state changes, can be hierarchically structured
- Autobiographical self
  - Comparing stored AMs with current experience
- Core- and proto-self process internal/external data



[Damasio. *The Feeling of What Happens: Body and Emotion in the Making of Consciousness*. Harcourt Incorporated, 1999]

Haykin's Cognitive Dynamic Systems

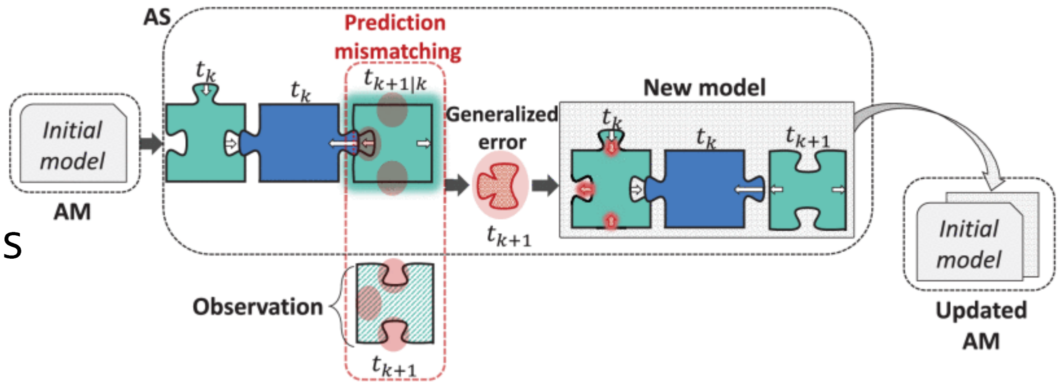
- Probabilistic reasoning machine (PRM) hierarchically organizes probabilistic information from control (CC) and perception (CP)
  - Maintain meta-level perception-action cycle by switching CC and CP behavior
  - Proposes Bayesian framework to model uncertainty&causality
- Cognitive control (CC)
  - Generates output towards lower levels, control actions at bottom
- Cognitive preceptor (CP)
  - No proprioceptive data, but internal strategies



[Haykin, Fuster. On cognitive dynamic systems: Cognitive neuroscience and engineering learning from each other, *Proceedings of the IEEE*, 102(4):608-628, 2014]

Friston's Bayesian Dynamical Systems

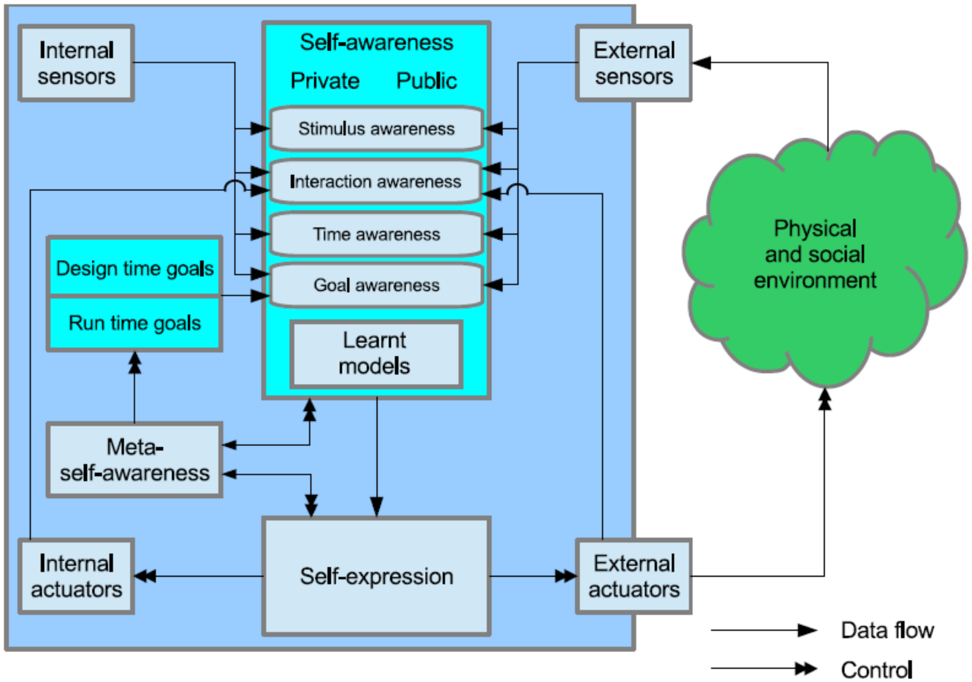
- Cognitive framework for SA linking neuroscientific observations with computational models (BDS)
  - Relates statistical mechanics' optimization with Bayes filtering
  - Proposes "generalized states" as basis for hierarchical filters
- Free energy principle as optimization criterion for discovering SA models from given experiences
- Variational computational representation & inference techniques
  - Discriminating models from different experiences
  - Prediction at different hierarchical&temporal levels



[Friston, Sengupta, Auletta. Cognitive dynamics: From attractors to active inference. *Proceedings of the IEEE*, 102(4):427-445, 2014]

Reference Architecture

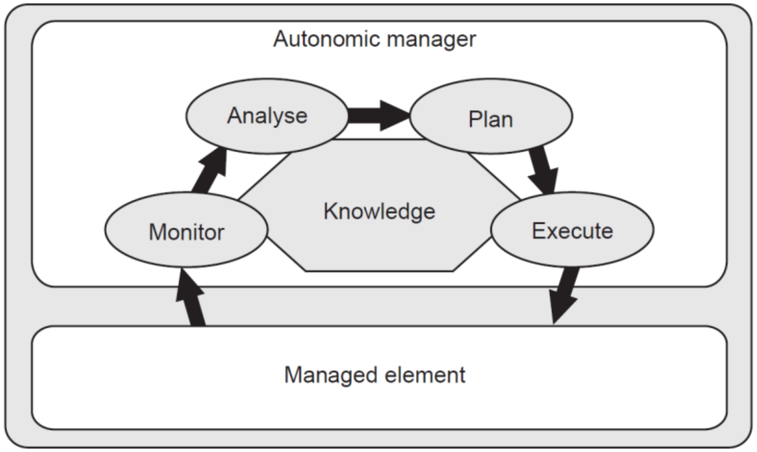
- Self-Awareness
  - How to learn models
  - Different levels
  - Private and public scope
- Self-Expression
  - How to make decisions
  - Exploiting learnt models
- Meta-Self-Awareness
  - Reasoning about own decision making
- Internal (proprioceptive) vs. external perception and actuation



[Lewis et al. Architectural Aspects of Self-Aware and Self-Expressive Computing Systems: From Psychology to Engineering. *IEEE Computer*, 48(8):62-70, 2015]

Autonomic Software Systems

- Effective self-management of complex IT systems at runtime, according to high-level objectives
  - Decomposed into four activities: Self-configuration, Self-optimization Self-healing, Self-protection
- Each component is managed by autonomic manager
  - Monitoring the managed element and its environment
  - Constructing and executing plans
  - Utilizing a knowledge repository
- MAPE-K architecture
  - Implements a control loop
  - Adopts ideas from reinforcement learning



[Kephart, Chess. The vision of autonomic computing. *IEEE Computer*, 36(1):41-50, 2003]

SA Architectures: Some Observations

- Architectures are quite diverse
  - From modelling cognitive processes to controlling software systems
  - Available initial knowledge and capabilities
  - Interpretation of sensor data and actuation of system components
- Differences in the representation of the "self"
  - Modelling proprioception and reasoning based on internal state
- Learning capabilities are fundamental
  - All architectures rely on some learning, some remain vague on techniques
  - Model creation and explainability important
- Hierarchical representation required for complex reasoning

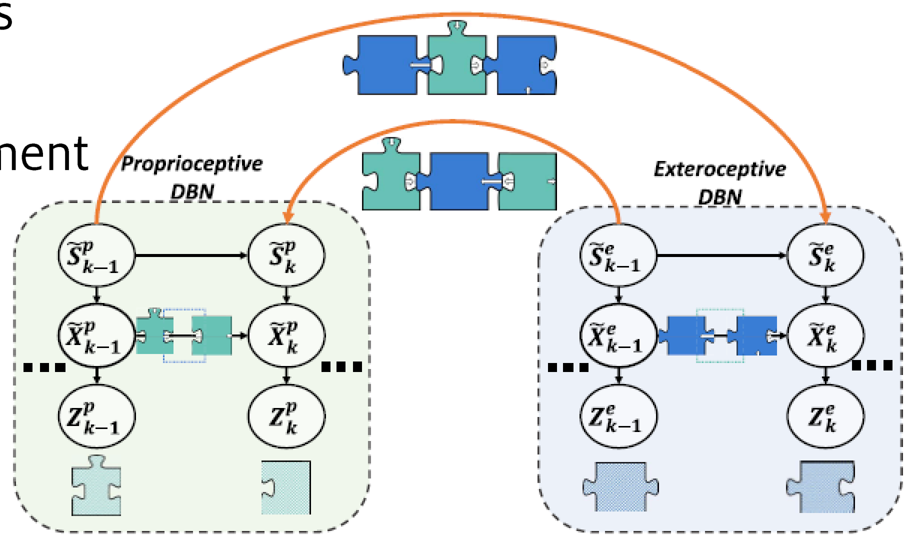
Self-Awareness Capabilities

- Initialization
  - Initial knowledge/cababilities of agent starts building own memories
- Memorization
  - Storing and retaining information (experience)
- Inference
  - Making predictions about future states
- Anomaly detection
  - Recognize observations that cannot be explained (by its memory)
- Model creation
  - Generate models than encode previous experiences (and predictions are supported by observations)
- Decision-making influence
  - Generate actuator signals such that actions can be self-monitored

SA Techniques

SA Modeling Using A Generalized Markov Jump Particle Filter

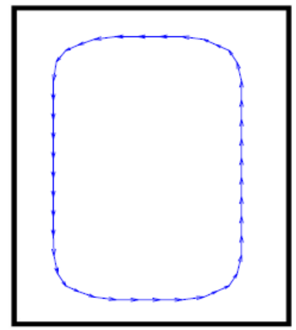
- Realize SA capabilities with hierarchical and multi-sensorial filtering
- Build proprioceptive (PM) and exteroceptive models (EM) with Dynamic Bayesian Networks (DBN)
  - PM: allow inferences of sensory information wrt its actions and internal states
  - EM: allow inferences of sensory information wrt sensed environment
- Multi-level models w. discrete and continuous states
- Coupling PM and EM allows context-based inferences

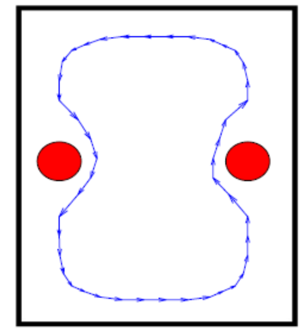


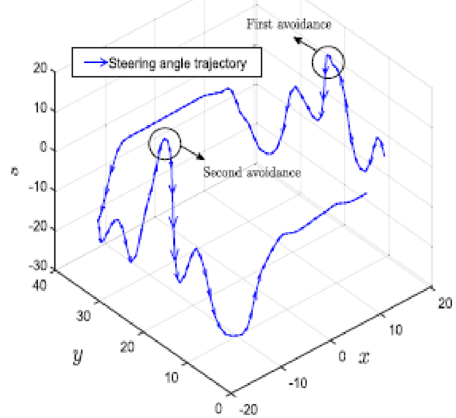
[Regazzoni et al. Multisensorial Generative and Descriptive Self-Awareness Models for Autonomous Systems. *Proceedings of the IEEE*, 108(7):987-1010, 2020]

Implementation of SA Capabilities

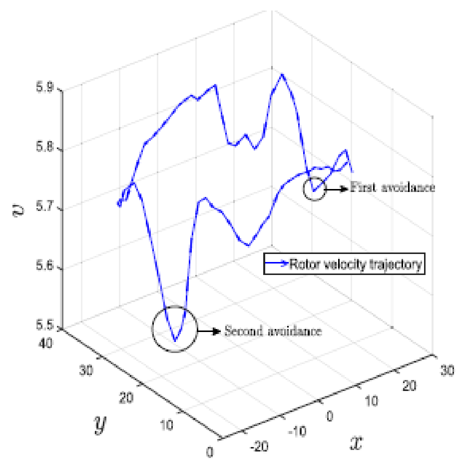
- Autonomous vehicle executing reference task (w/ disturbance)
  - Steering angle, power and rotor velocity as proprioceptive sensors
  - Vision as exteroceptive sensors
- Detect abnormality wrt task's model
- Sensory input



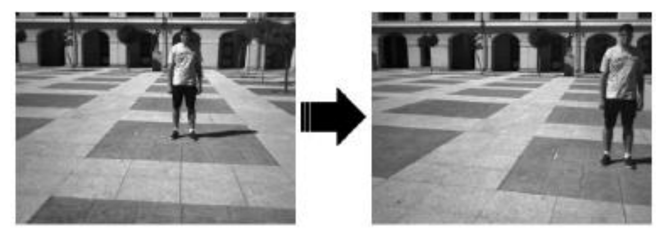




Steering angle



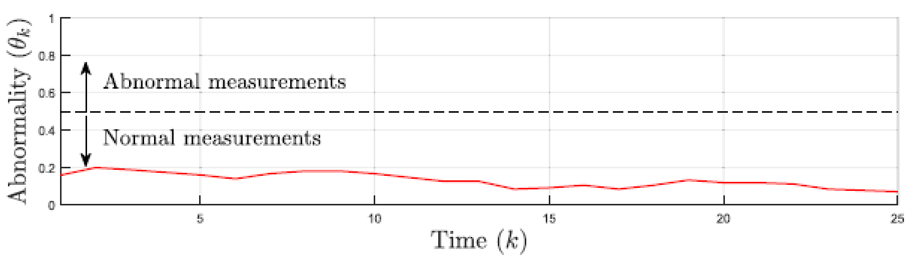
Rotor velocity



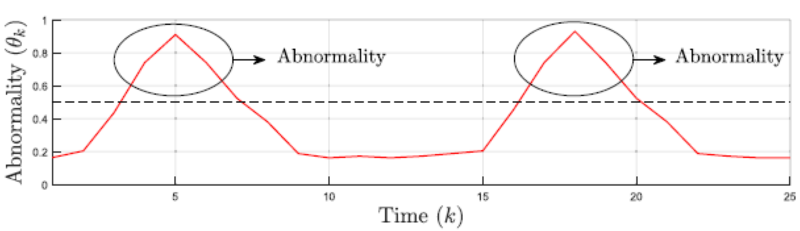
Vision

Implementation of SA Capabilities

- Initialization based on sensory input of reference behavior
  - PM: random walk dynamics (based on fixed internal states)
  - EM: GANs codifying image & optical flow on linear motion
- Inference at continuous and discrete levels
  - PM: continuous state changes; regions with linear movement
  - EM: set of GANS with generated errors
- Abnormality detection based distance metrics



Reference task w/o disturbance



Reference task w disturbance

- Model creation based on identified errors of previous model

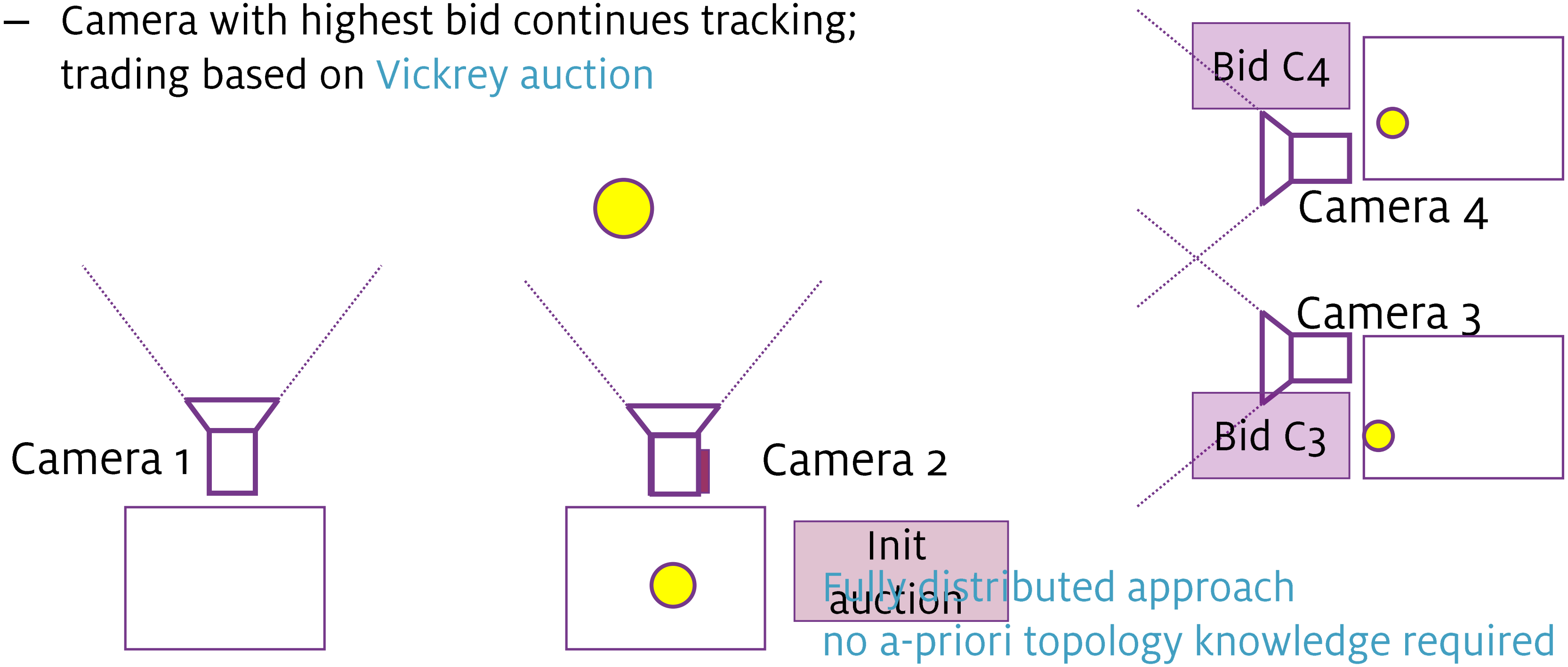
Self-aware Camera Network

- Perform autonomous, decentralized and resource-aware network-wide analysis

- Demonstrate autonomous multi-object tracking in camera network
  - Exploit single camera object detector & tracker
  - Perform camera handover
  - Learn camera topology
- Key decisions for each camera
  - When to track an object within its FOV
  - When to initiate a handover
  - Whom to handover

Virtual Market-based Handover

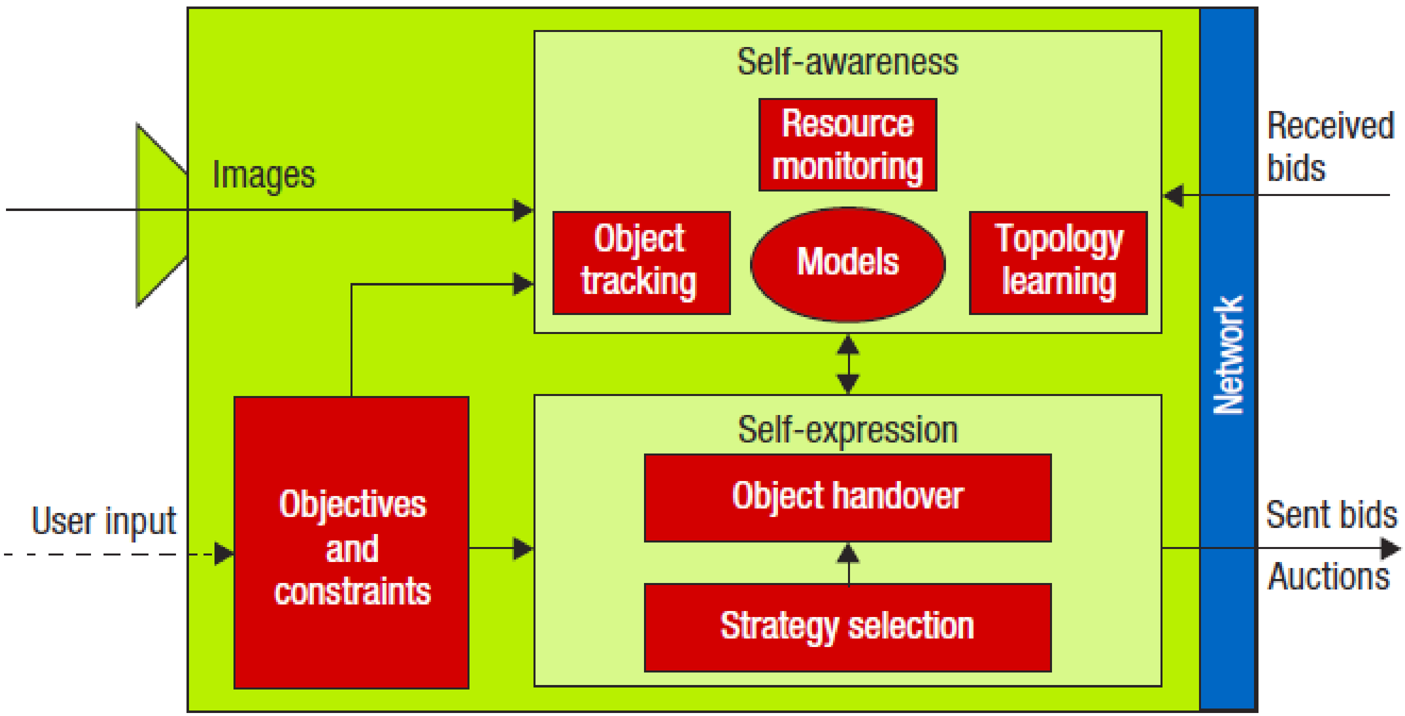
- Initialize auctions for exchanging tracking responsibilities
  - Cameras act as self-interested agents, i.e., maximize their own utility
  - Selling camera (where object is leaving FOV) opens the auction
  - Other cameras return bids with price corresponding to “tracking” confidence
  - Camera with highest bid continues tracking;
    trading based on Vickrey auction



Fully distributed approach
no a-priori topology knowledge required

Self-aware Camera Node

- Design following computational self-awareness



[Rinner et al. Self-aware and Self-expressive Camera Networks. *IEEE Computer*, 48(7):21-28, 2015]

Camera Control

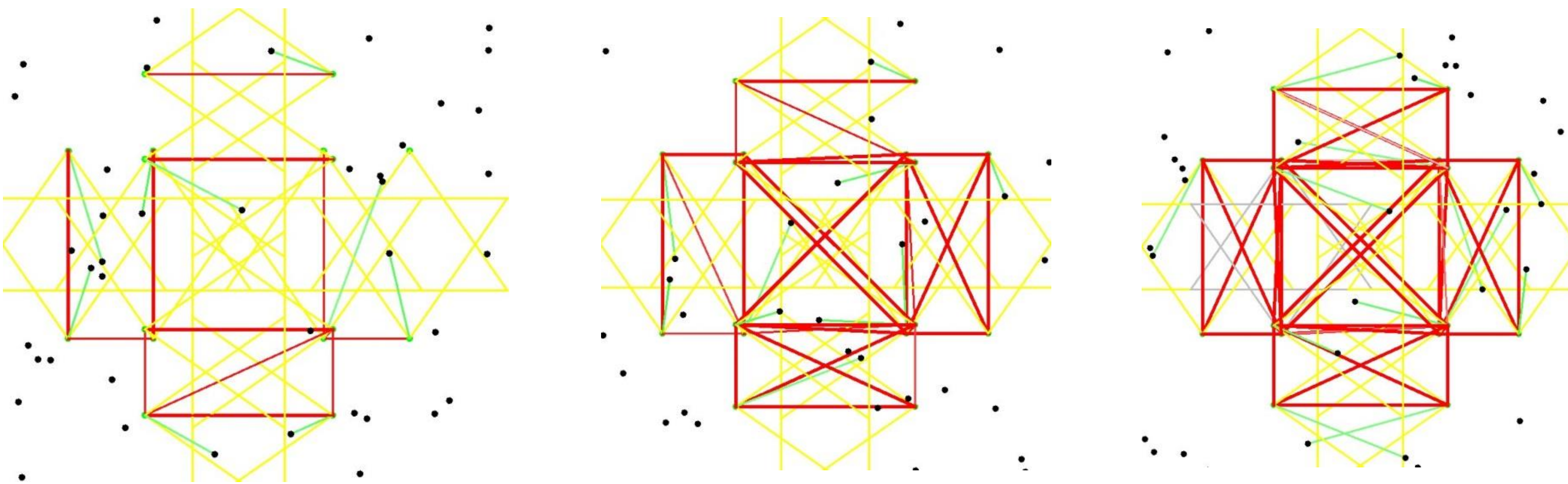
- Each camera acts as agent maximizing its utility function

$$U_i(O_i) = \sum_{j \in O_i} [c_j \cdot v_j \cdot \Phi_i(j)] - p + r$$

- Local decisions
  - When to initiate an auction (at regular intervals or specific events)
  - Whom to invite (all vs. neighboring cameras)
  - When to trade (depends on valuation of objects in FOV)
- Learn neighborhood relations with trading behavior ("pheromones")
  - Strengthen links to buying cameras
  - Weaken links over time

Learn Neighborhood Relationships

- Gaining knowledge about the network topology (vision graph) by exploiting the trading activities
- Temporal evolution of the vision graph



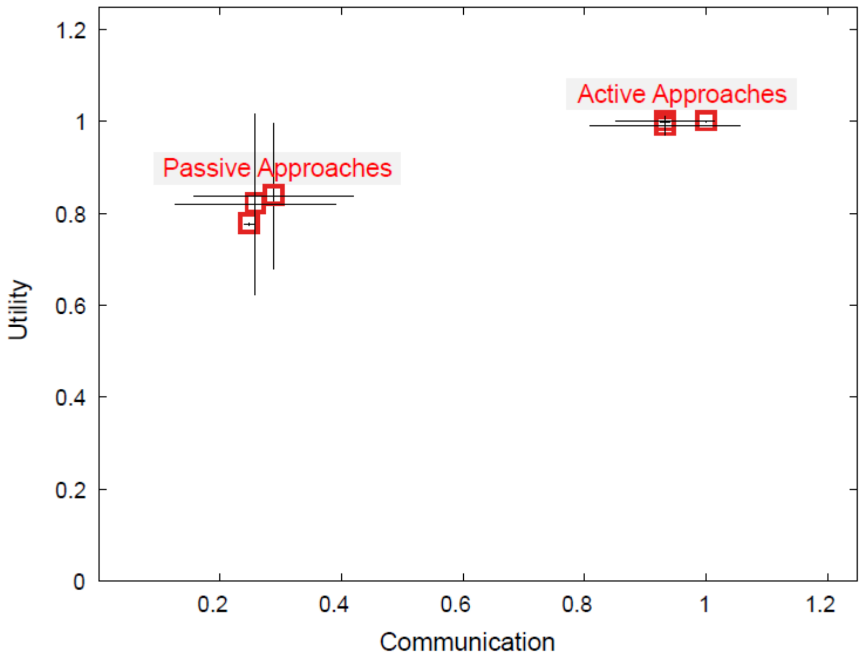
Six Camera Strategies

- Auction initiation
  - "Active": at regular intervals (at each frame)
  - "Passive": only when object is about to leave the FOV
- Auction invitation
  - "Broadcast": to all cameras
  - "Smooth": probabilistic proportional to link strength
  - "Step": to cameras with link strengths above threshold (and rest with low probability)

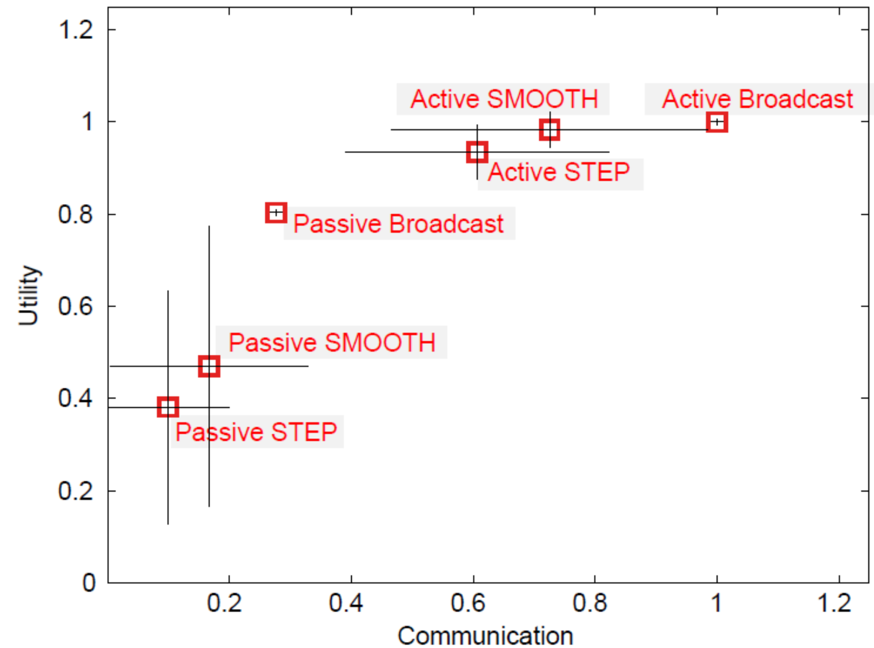
- Selected strategy influences network performance (utility) and communication effort

Tracking Performance

- Tradeoff between utility and communication effort



Scenario 1 (5 cameras, few objects)



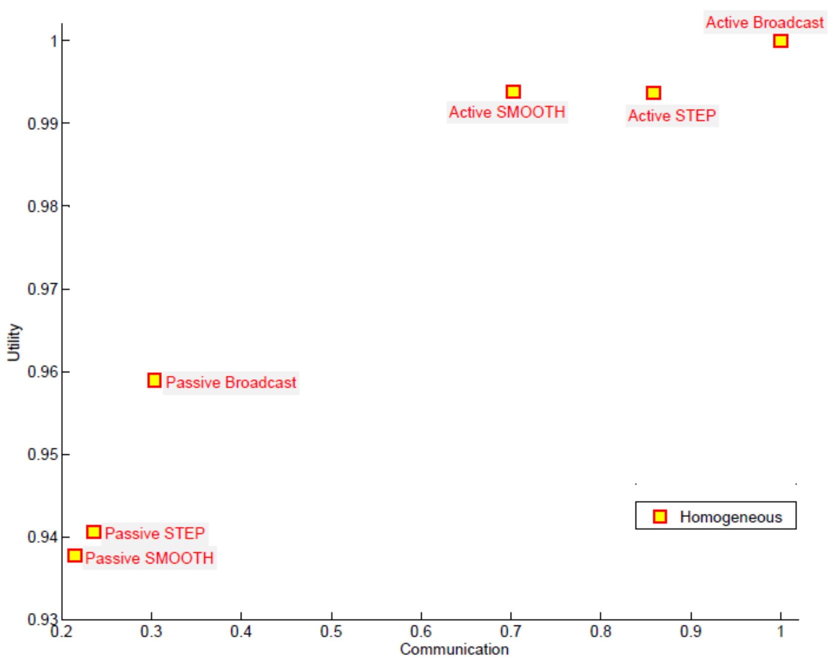
Scenario 2 (15 cameras, many objects)

- Emerging Pareto front

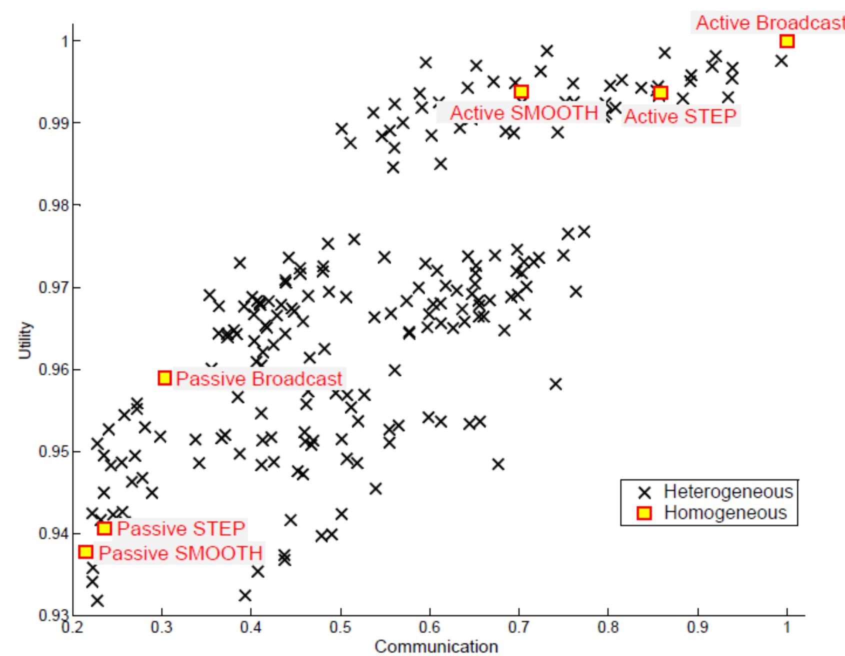
[Esterle et al. Socio-Economic Vision Graph Generation and Handover in Distributed Smart Camera Networks. *ACM Trans. Sensor Networks*, 10(2):1-24, 2014]

Assigning Strategies to Cameras

- Identical strategy for all cameras may not achieve best result



Homogeneous strategies (3 cameras)



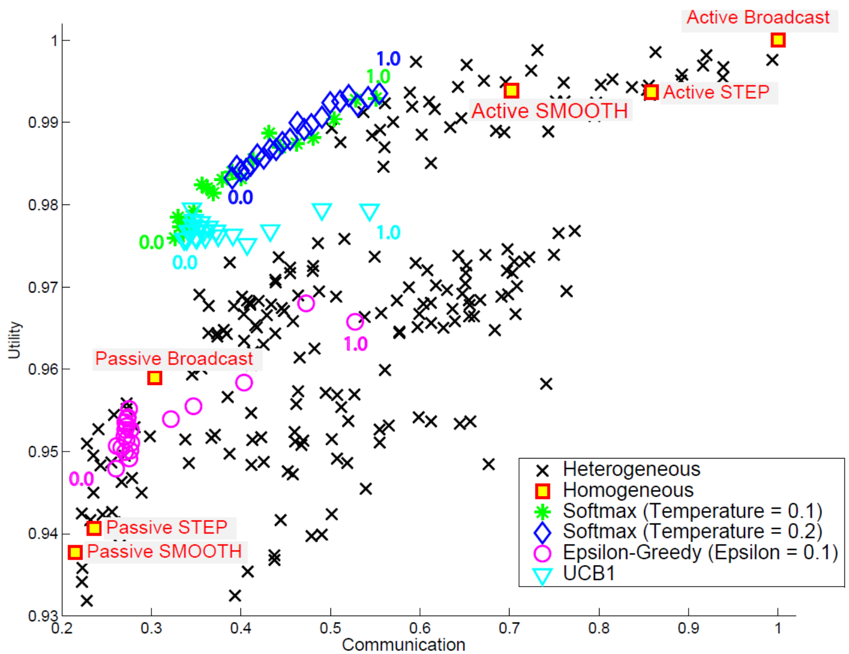
Heterogeneous strategies (3 cameras)

- Strategy depends on various parameters (FOV, neighbors, scene …)
  - Let cameras learn their best strategy

UNIVERSITÄT KLAGENFURT

Decentralized Multi-Agent Learning

- Exploit bandit solver framework to maximize global performance
  - Co-dependency among agents' performance
  - Complex relationship between local reward global performance



[Lewis et al. Static, Dynamic and Adaptive Heterogeneity in Socio-Economic Distributed Smart Camera Networks. *ACM Trans. Autonom. Adapt. Syst.* 10(2):1-30, 2015]

Challenges

How to Design SA Systems

- Complexity of SA models may vary
  - Type and dimensionality of sensed data
  - Actions and goal of the agent
- Mostly case-by-case design approaches are currently used
  - Application or extension of existing techniques
  - Based on engineers experience adopting trail-and-error
- Towards a "Theory" for SA systems
  - Collecting best practice examples
  - Analyzing benefits and limitations
  - Proposing design guidelines, patterns

[Lewis, Platzner, Rinner, Torresen, Yao. *Self-aware Computing Systems: An Engineering Approach*. Springer, 2016]]

How to Achieve Collective SA

- Often a collection of autonomous agents is working on mission
  - Individual sensors and decision making
  - Lack of global knowledge
- Build joint models and generate decisions at team level
- Decompose mission and let individual agents solve them
  - But system-level knowledge is distributed (or not available)
- Communication essential for knowledge sharing and coordination
  - Learning models of other agents
  - Interaction between human and SA agent

[Birke et al. *Self-aware Computing Systems: Open Challenges and Future Research Directions*. Chapter 25, Springer, 2017]

How to Guarantee/Avoid Behaviors

- Exploiting models derived from sensors may result in emergent behaviors
- Designing an agent such that particular behavior eventually emerges is challenging
  - Guarantees about behaviors are important for safety and resilience
  - Some specifications of models and behaviors are required, but such formal verification "contradicts" the concept of self-awareness
- Inherent (systematic) uncertainty particularly challenging for analysis and verification
  - Pursuing new goals due to (unknown) changes

[Dennis, Fisher. Verifiable Self-Aware Agent-Based Autonomous Systems. *Proceedings of the IEEE*, 108(7):1011-1026, 2020]

What is the Cost of SA

- For engineering SA systems we must be able to estimate the system's performance and properties such as
  - Resource usage and cost
  - Reliability and safety
- Self-awareness incurs overheads and tradeoffs between overheads and capabilities need to be explored and quantified
  - Cost and benefit of SA capabilities (actual and worst case)
- Basis for (online) decision what SA capability should be used when
  - Modeling type and complexity
  - Inference techniques

[Birke et al. *Self-aware Computing Systems: Open Challenges and Future Research Directions*. Chapter 25, Springer, 2017]

How to Earn Trust in SA

- Autonomous systems will participate in society in many ways.
- Trust is essential to cooperation. AS must be designed to
  - understand and follow social norms, including morality, ethics, and convention
  - earn the trust of others in society
- To be accepted, and to strengthen our society, AS must show they are worthy of trust according to the social norms.

[Kuipers. How can we trust a robot? *Communications of the ACM*, 61(3):86-95, Feb. 2018]

Conclusion

- Computing systems with various levels of autonomy have been proposed to manage ever increasing levels of complexity and uncertainty
- Self-awareness in a computational context is founded on advanced methods and algorithms from different disciplines
- Research on self-awareness in AS is fragmented over several fields and lacks a common terminology
- Self-awareness may be a fundamental principle with potential of becoming an enabling technology for AS applications
- Various research challenges and technical issues needs to be solved

Further Information

- Pervasive Computing group

  http://nes.aau.at

  http://www.bernhardrinner.com

- Special issue on Self-Awareness for Autonomous Systems, 2020

- Book on Self-aware Computing Systems
  Lewis, Platzner, Rinner, Torresen, Yao (Eds)
  Springer, 2016