Universitat Pompeu Fabra Barcelona

Robust Service Orchestration for Computing Continuum Systems

Boris Sedlak

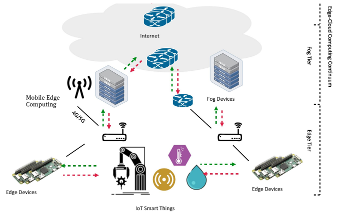
Co-funded by the European Union
Funded by the European Union NextGenerationEU
Gobierno de España, Ministerio de Industria y Turismo
Plan de Recuperación, Transformación y Resiliencia
Universidad de Extremadura

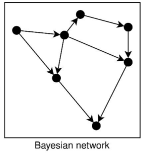
Abstract

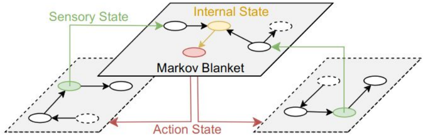
Hidden in main presentation

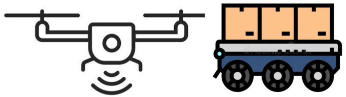
While the emergence of Edge computing promises major improvements in IoT processing, the heterogeneity and volatility of Edge infrastructures make service orchestration increasingly complex. Yet, to maintain robust system operation, we must be certain—or at least certain enough—about the expected outcome of our actions, such as shifting resources or workloads. This talk highlights Active Inference (AIF), an agent-based framework inspired by neuroscience, which supports operators in developing a causal understanding of the underlying generative processes. By continuously exploring and interacting with their environment, AIF agents develop models of the systems they govern and their interdependencies. This, in turn, enables them to predict the outcomes of actions when composing systems into larger architectures. The talk first outlines how AIF, as a concept, is inherently designed for acting under uncertainty, and then presents examples of how AIF can establish robust understanding of the environment to ensure continuous and adaptive system operation.

Structure of the Talk









Motivation & Common Problems — Methodologies & Latest Results — Future Work

I – Who am I ??

Boris Sedlak

Postdoc @ UPF Barcelona
Distributed Intelligence & Systems-Engineering Lab

PhD @ TU Wien, Vienna
Distributed Systems Group

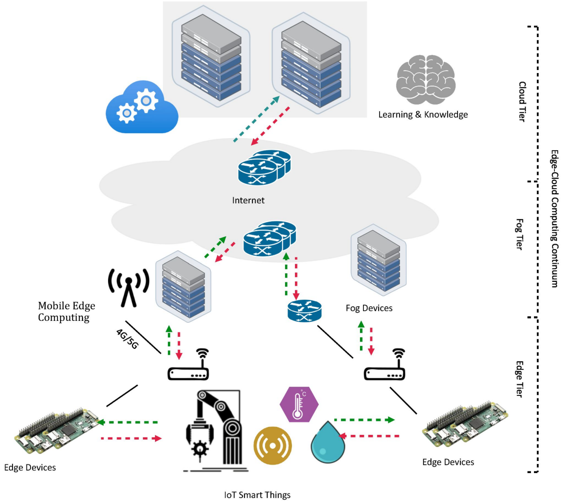
Software Engineer @ Lotterien
Agile Development and Testing

I – Common Concepts

Computing Continuum (CC) as a composition of multiple processing tiers that stretch from IoT and edge computing, over Fog resources, to distant Cloud centers

Combines the benefits of all its tiers, i.e., low-latency and privacy-protecting computation from Edge, high availability and virtually unlimited processing resources from Cloud

Smart Cities are a common instance of distributed systems, where interconnected services (e.g., traffic surveillance or road surveillance) collaborate based on collected sensor data



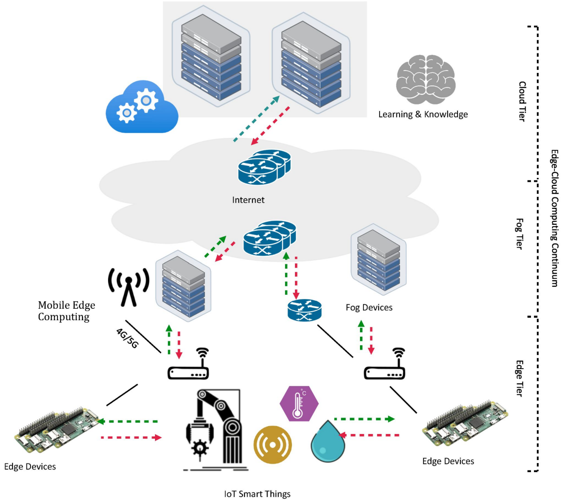
Example of a Computing Continuum architecture [1]

[1] P. Donta, I. Murturi, V. Casamayor, B. Sedlak, and S. Dustdar; **Exploring the Potential of Distributed Computing Continuum Systems** (2023)

Computing Continuum (CC) as a composition of multiple processing tiers that stretch from IoT and edge computing, over Fog resources, to distant Cloud centers

Combines the benefits of all its tiers, i.e., low-latency and privacy-protecting computation from Edge, high availability and virtually unlimited processing resources from Cloud

Smart Cities are a common instance of distributed systems, where interconnected services (e.g., traffic surveillance or road surveillance) collaborate based on collected sensor data

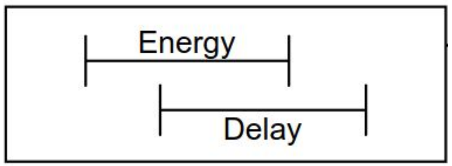


Example of a Computing Continuum architecture [1]

[1] P. Donta, I. Murturi, V. Casamayor, B. Sedlak, and S. Dustdar; **Exploring the Potential of Distributed Computing Continuum Systems** (2023)

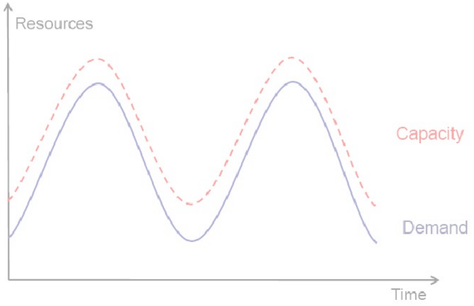
I – Common Concepts (cont.)

Service Level Objectives (SLOs) specify requirements that must be ensured throughout operation (e.g., latency < *t*).



Elasticity Strategies scale a system according to current demand; e.g., if performance is insufficient, allocate more resources. However, what if this does not fulfill SLOs?

Service Level Agreements (SLAs) as binding agreement between service provider and consumer. However, very limited support in resource-restricted environments

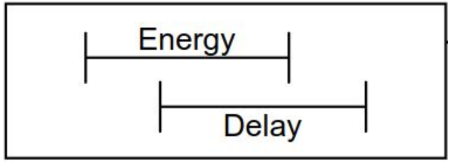


Elasticity allocates the right amount of resources [2]

[2] Ricciardi et al., **Saving Energy in Data Center Infrastructures** (2011)

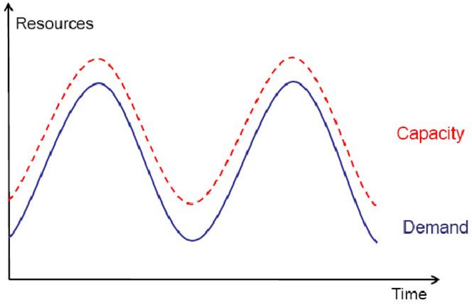
I – Common Concepts (cont.)

Service Level Objectives (SLOs) specify requirements that must be ensured throughout operation (e.g., latency < *t*).



Elasticity Strategies scale a system according to current demand; e.g., if performance is insufficient, allocate more resources. However, what if this does not fulfill SLOs?

Service Level Agreements (SLAs) as binding agreement between service provider and consumer. However, very limited support in resource-restricted environments

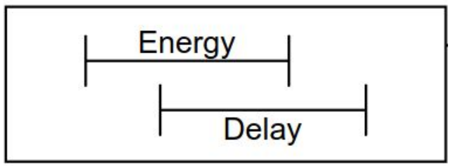


Elasticity allocates the right amount of resources [2]

[2] Ricciardi et al., **Saving Energy in Data Center Infrastructures** (2011)

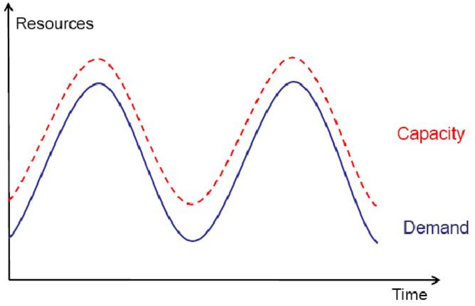
I – Common Concepts (cont.)

Service Level Objectives (SLOs) specify requirements that must be ensured throughout operation (e.g., latency < *t*).



Elasticity Strategies scale a system according to current demand; e.g., if performance is insufficient, allocate more resources. However, what if this does not fulfill SLOs?

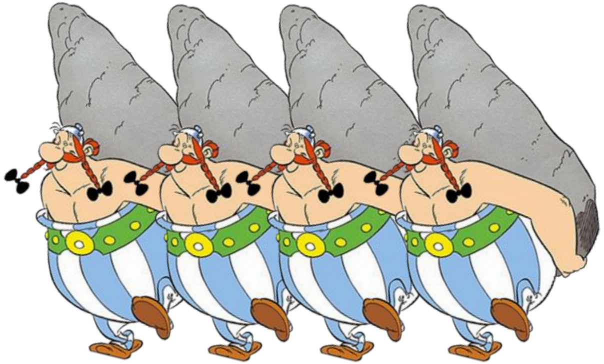
Service Level Agreements (SLAs) as binding agreement between service provider and consumer. However, very limited support in resource-restricted environments



Elasticity allocates the right amount of resources [2]

[2] Ricciardi et al., **Saving Energy in Data Center Infrastructures** (2011)

I – Fundamental Problems



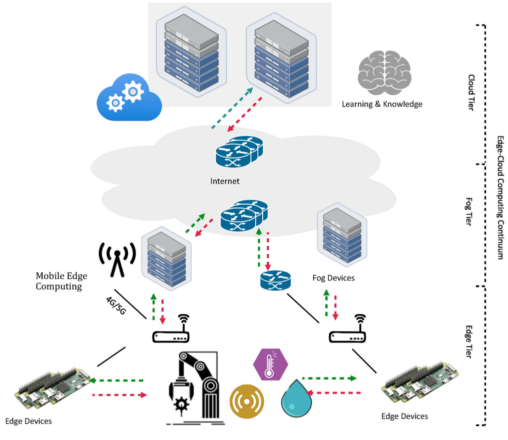
Homogeneous Resources (Cloud)

Heterogeneous Resources (CC)

Free images from https://www.nicepng.com/ourpic/u2e6r5y3e6y3t4i1_dogmatix-obelix-asterix-obelix-asterix-a-obelix-png/

I – Fundamental Problems

Homogeneous Resources (Cloud)

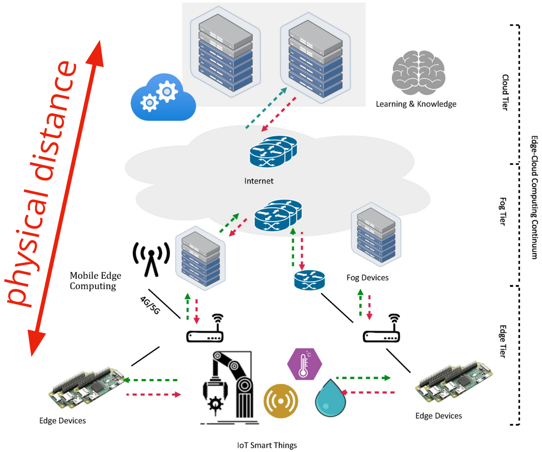


Heterogeneous Resources (CC)

Free images from https://www.freepik.com/free-photos-vectors/server-rack

I – Fundamental Problems

Homogeneous Resources (Cloud)

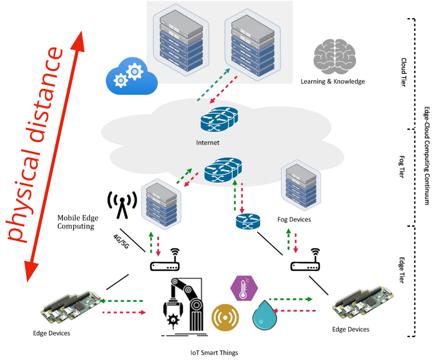


Heterogeneous Resources (CC)

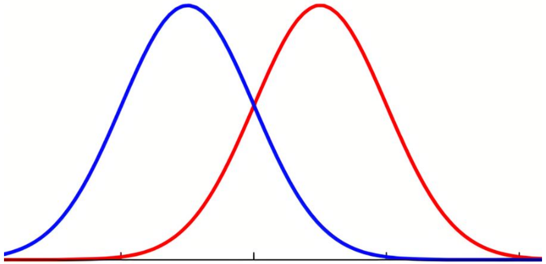
I – Fundamental Problems

Heterogeneity prevents simple task distribution between available devices. Don't know a priori what will be the *behavior* of service X on device Y

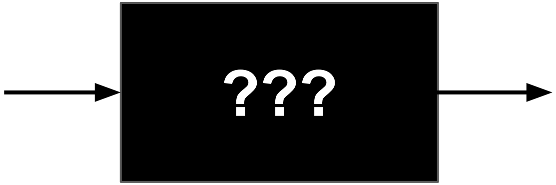
Physical distribution creates latency between different distributed devices; transferring state information (e.g., CPU load) creates lots of traffic on the network.



I – Fundamental Problems (cont.)



System dynamics change over time

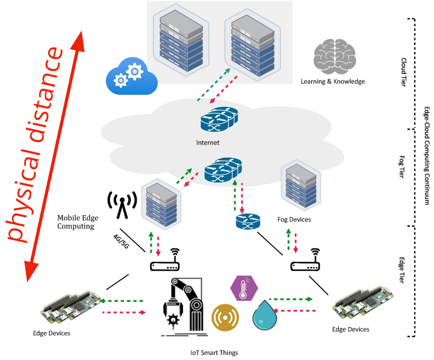


Low trust into black-box models

I – Fundamental Problems (cont.)

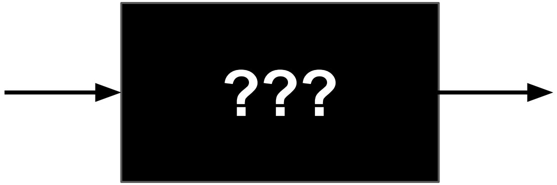
Heterogeneity prevents simple task distribution between available devices. Don't know a-priori what will be the *behavior* of service X on device Y

Physical distribution creates latency between different distributed devices; transferring state information (e.g., CPU load) creates lots of traffic on the network.



Variable drifts cause any ML model to lose accuracy over time. Cannot train models in one-shot operations but must incorporate changes continuously.

Low-trust into black-box ML models that cannot give human-verifiable chains of thought. E.g., debugging



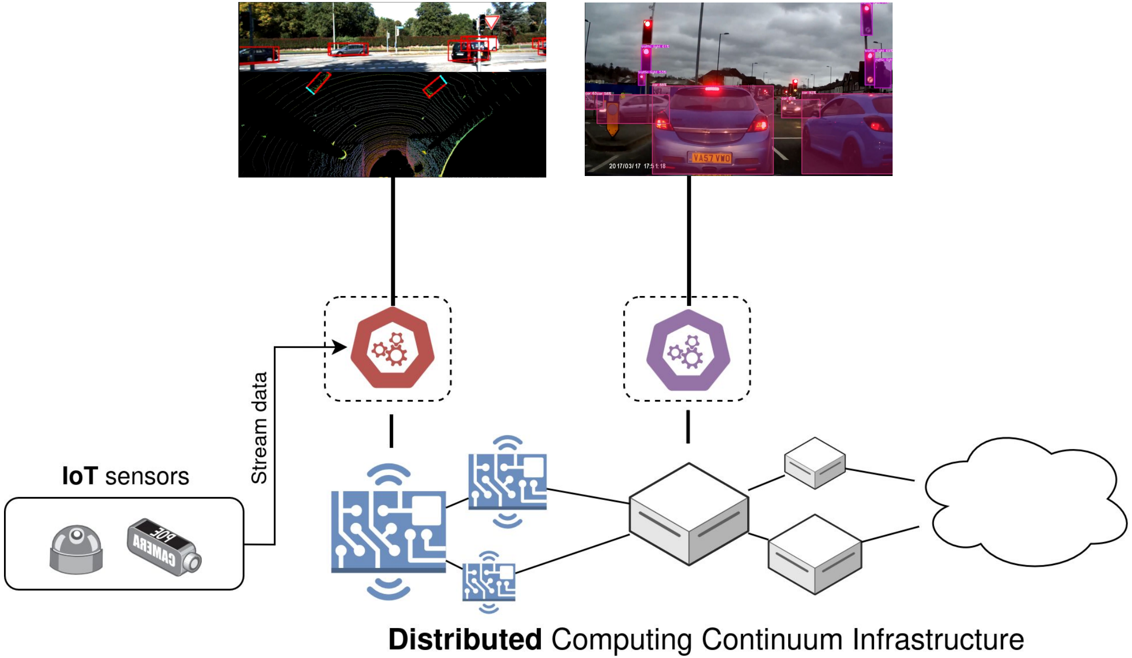
I – Overview of our Approach

Hidden in main presentation

Distributed processing

– Distributed CC infrastructure composed of various **device types** at different **locations**

– Sensor data continuously streamed from IoT devices to different processing services



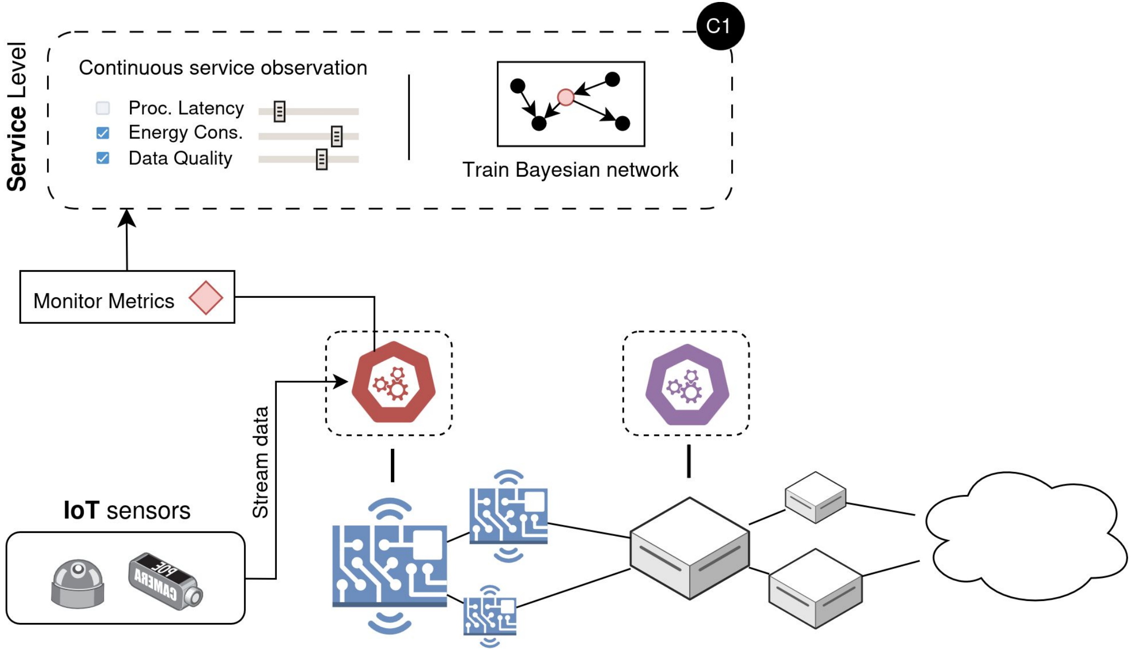
I – Overview of our Approach (C1) Hidden in main presentation

Service interpretation

– Monitor processing across CC by collecting metrics; eval. real-time SLO fulfillment

– Train a model (i.e., BN) for predicting SLO fulfillment of **individual** services in different deployment configurations [7]

– Retrain model according to SLO prediction accuracy; slows down as model improves [9]



[7] Sedlak et al., **Designing Reconfigurable Intelligent Systems with Markov Blankets**, at ICSOC 2023
[9] Sedlak et al., **SLO-Aware Task Offloading Within Collaborative Vehicle Platoons**, at ICSOC 2024

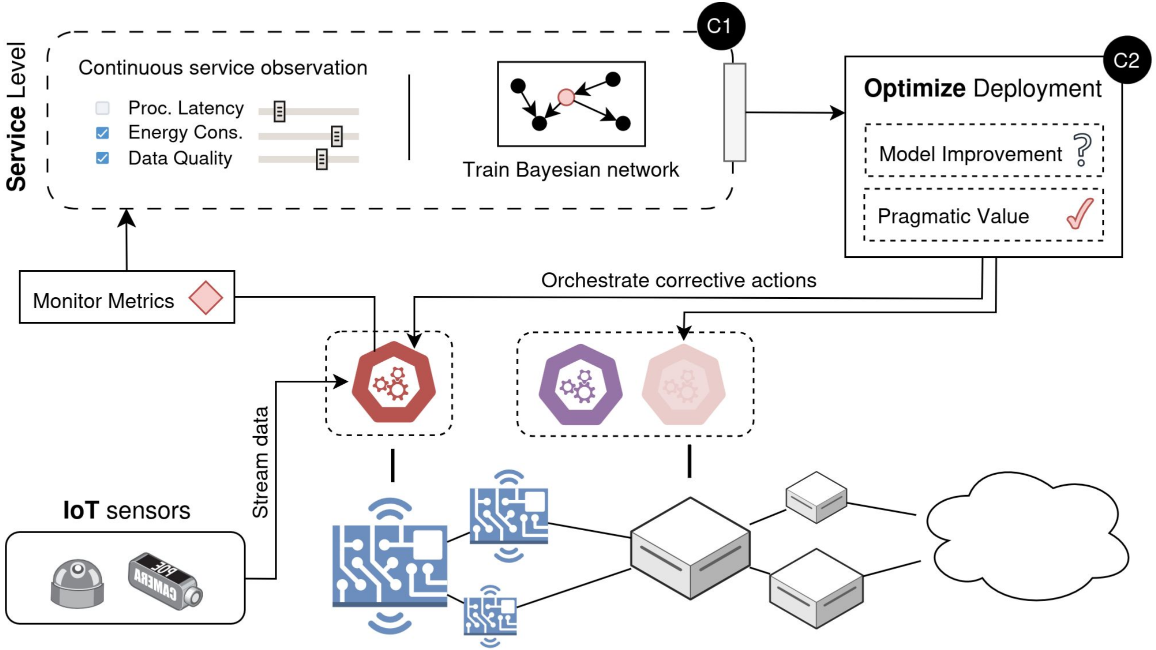
I – Overview of our Approach (C2) Hidden in main presentation

Service adaptation

– Optimize **local SLO fulfillment** by reconfiguring processing services; choose between avail. elasticity strategies [5]

– Escape local optima through continuous exploration; check poss. **model improvement** [12]

– Decisions can be empirically verified and interpreted; useful for non-technical explanation



[5] Sedlak et al., **From Metrics to Multidimensional Elasticity Strategies**, at IEEE Services EDGE 2023
[12] Sedlak et al., **Active Inference on the Edge: A Design Study**, at PerconAI 2024

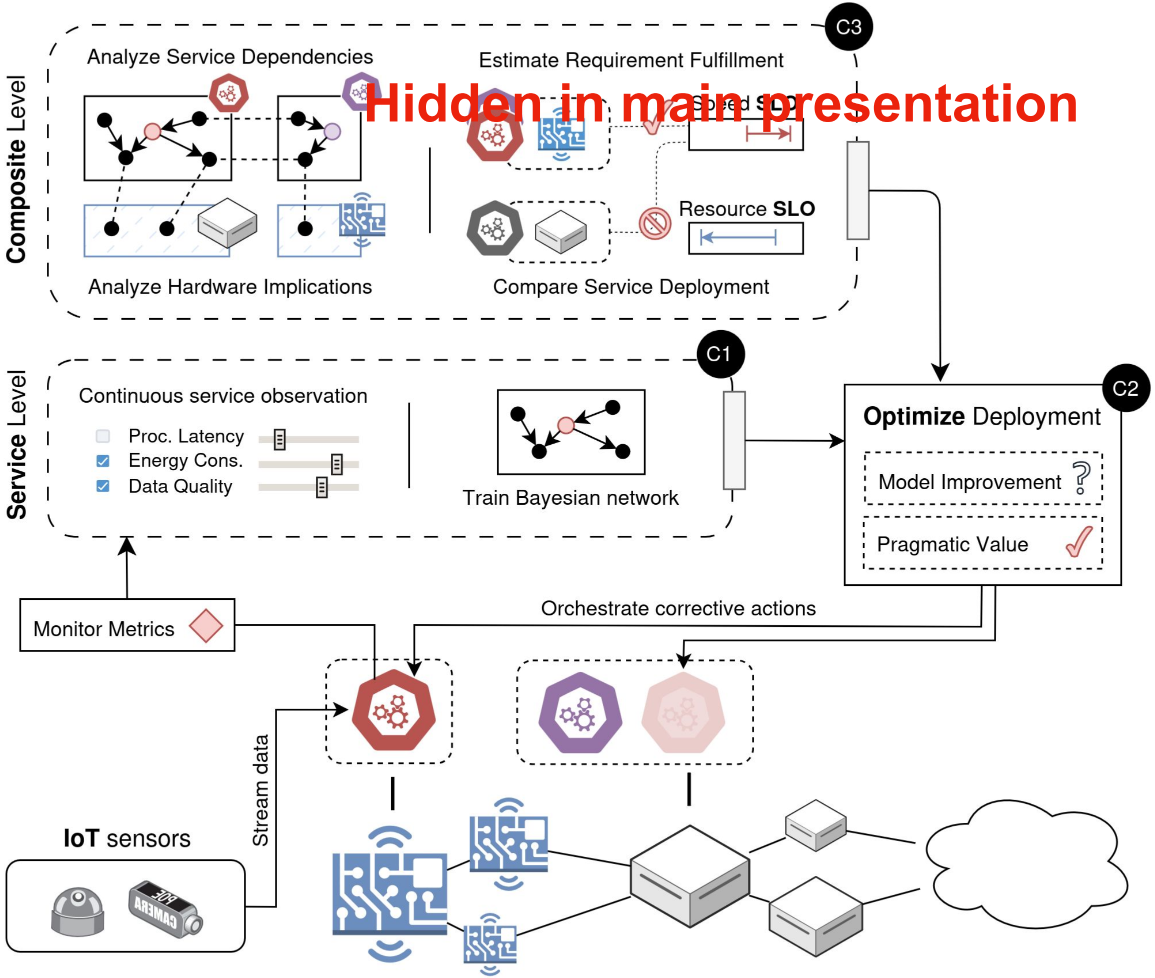
I – Overview of our Approach (C3)

Service collaboration

– Compose individual models to overarching representation; quantify service dependencies and **impact on hardware** [4]

– Collaborative orchestration considers service relations to optimize global SLO fulfillment

– Exchanging and merging BNs between services to speed up the **onboarding** of new service types or devices types [13]



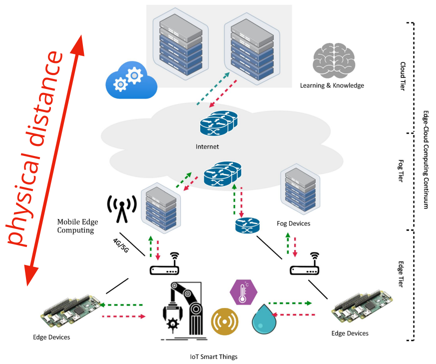
[4] Sedlak et al., **Markov Blanket Composition of SLOs**, at IEEE Service EDGE 2024
[13] Sedlak et al., **Equilibrium in the Computing Continuum through Active Inference**, at Elsevier FGCS, 2024

I – Fundamental Problems (cont.)

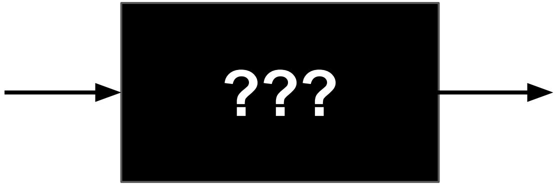
Heterogeneity prevents simple task distribution
betwe
will be

Physic
distrib
CPU load) creates lots of traffic on the network.

Decentralized decision-making based on local observations and understanding



Variable drifts cause any ML model to lose accuracy over time. Cannot train models in one-shot operations but must incorporate changes continuously.



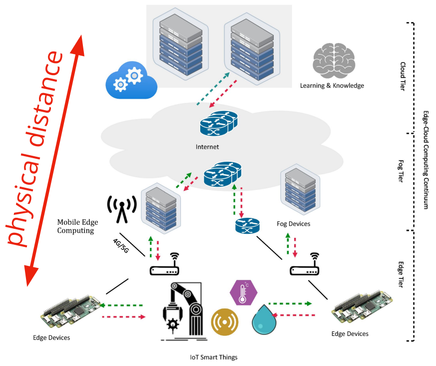
Low-trust into black-box ML models that cannot give human-verifiable chains of thought. E.g., debugging

I – Fundamental Problems (cont.)

Heterogeneity prevents simple task distribution
betw[...] available devices. Don't know a priori what
will be

Physic
distrib
CPU load) creates lots of traffic on the network.

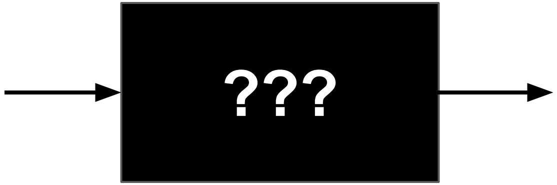
Decentralized decision-making based on local observations and understanding



Varia
over t
but m

Low-t
human-verifiable chains of thought. E.g., debugging

Continuous and lifelong learning of environment, verifiable through structural causal models

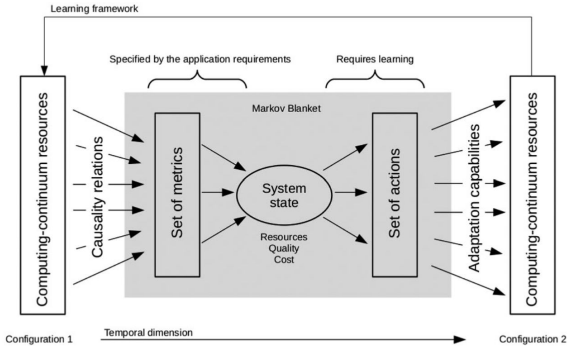


II – Markov Blanket (MB)

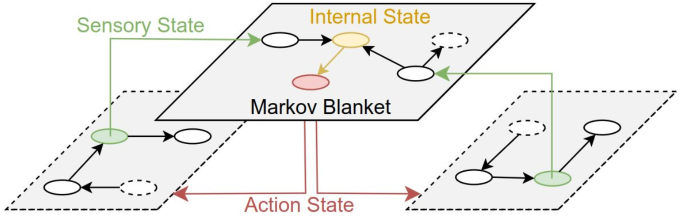
Interactions between **systems** (e.g., human in world) can be expressed through MBs; fulfill Markov property – so decisions can be taken based on system's current state

Creates **formal boundary** between a system and external states; within MB, discard all variables that show no impact on internal states and requirements (i.e., SLO fulfillment)

Provides clear interfaces for **sensory** and action **states**; policy (e.g., scaling) as a mapping between these states



Behavioral Markov blanket of a system [3]



Action-perception cycle between multiple entities [4]

[3] Dustdar et al., **On Distributed Computing Continuum Systems** (2023)
[4] Sedlak et al., **Markov Blanket Composition of SLOs**, at IEEE Services Edge 2024

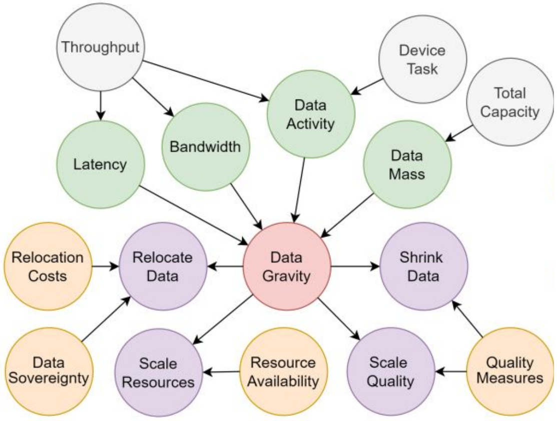
II – SLOs and Behavioral Models

Hidden in main presentation

MB: Expresses how to evaluate a composite SLO and how to react according to the current device context

Behavioral model

Internal state (●) evaluates objectives and how these relate to external sensory inputs (●); can interact with the world through action, i.e., elasticity strategies (●), which are influenced by contextual factors (●)



Example of a behavioral model for data gravity [5]

[5] Sedlak et al., **Controlling Data Gravity and Data Friction: From Metrics to Multidimensional Elasticity Strategies**, at IEEE Service EDGE 2023

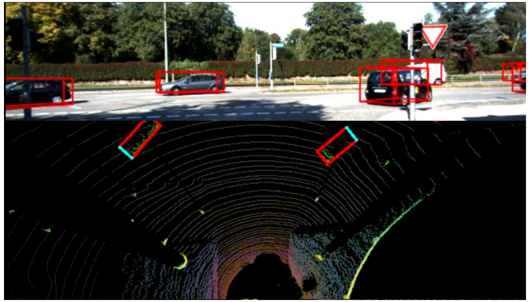
II – Stream Processing Scenarios

Commonly addressed use cases revolve around continuous **stream processing**; in case **time-critical** adaptations are required, this poses a higher need for sophisticated adaptation mechanisms.

Video Processing (Yolo V8)

Object detection in a video stream using Yolo [6]

Mobile Mapping (Lidar)



Creating a mobile map from binaries using Lidar [6]

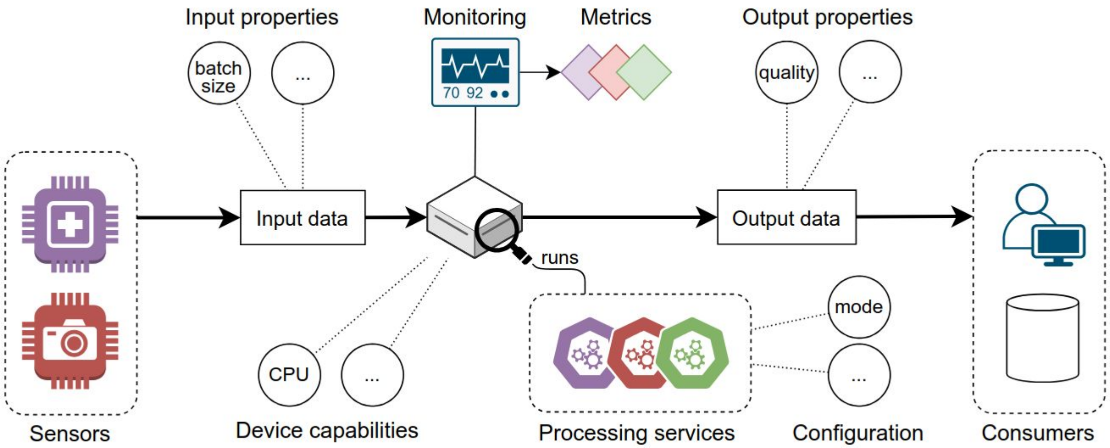
QR Scanner (OpenCV)

QR code scanning in a video using OpenCV [6]

[6] Sedlak et al., **Adaptive Stream Processing on Edge Devices through Active Inference** (Scheduled for 2025 at Springer ES)

II – Stream Processing Scenarios

Commonly addressed use cases revolve around continuous **stream processing**; in case **time-critical** adaptations are required, this poses a higher need for sophisticated adaptation mechanisms.

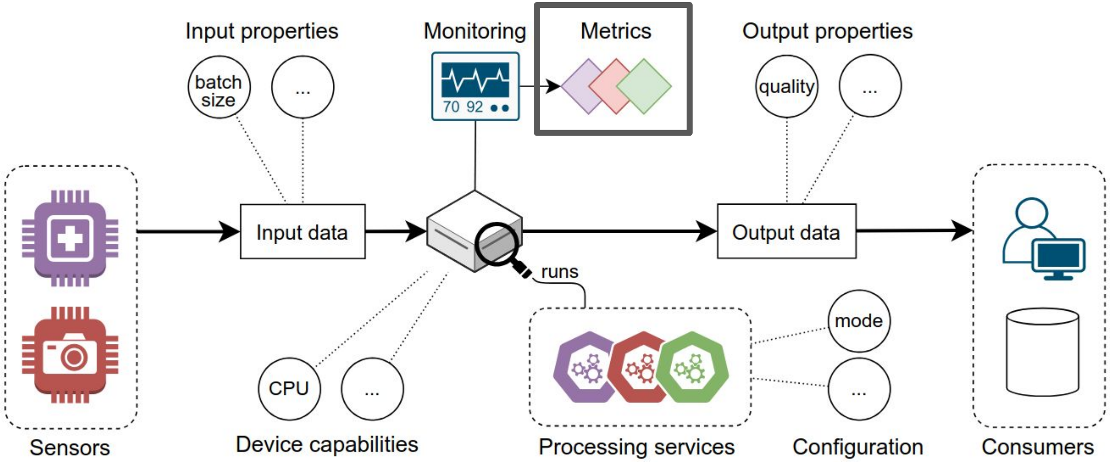


Abstract representation of a monitored stream processing service [6]

[6] Sedlak et al., **Adaptive Stream Processing on Edge Devices through Active Inference** (Scheduled for 2025 at Springer ES)

II – Stream Processing Scenarios

Commonly addressed use cases revolve around continuous **stream processing**; in case **time-critical** adaptations are required, this poses a higher need for sophisticated adaptation mechanisms.



Abstract representation of a monitored stream processing service [6]

fps	pixel	cores	change_flag	timestamp
31	800	3	False	2024-11-30
32	800	3	False	2024-11-30
32	800	3	False	2024-11-30
32	800	3	False	2024-11-30
32	800	3	False	2024-11-30
30	800	3	False	2024-11-30
32	800	3	False	2024-11-30
32	800	3	False	2024-11-30
32	800	3	False	2024-11-30

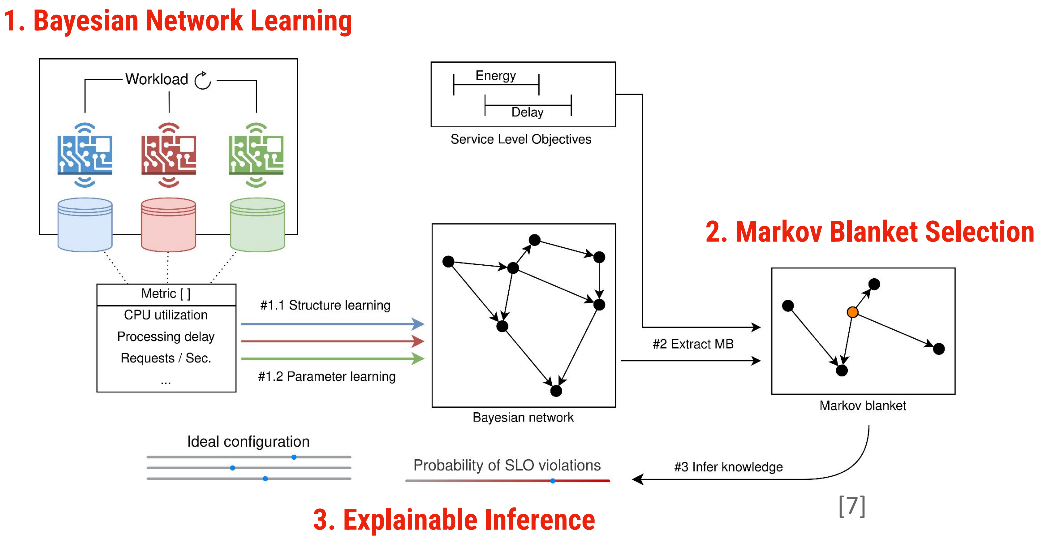
Example of processing metrics as tabular data [6]

[6] Sedlak et al., **Adaptive Stream Processing on Edge Devices through Active Inference** (Scheduled for 2025 at Springer ES)

II – Basic Methodology

Optimize the system (i.e., its SLO fulfillment) according to model

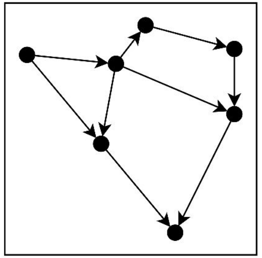
Model: use processing metrics and properties to train a *generative model* that describes the variable relations within the environment.



[7] Sedlak et al., **Designing Reconfigurable Intelligent Systems with Markov Blankets**, at ICSOC 2023

II – Basic Methodology

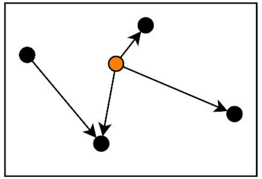
Bayesian Network Learning



Bayesian network

- **Structure Learning**
  Various algorithms (e.g., HCS)
  Directed Acyclic Graph (DAG)
- **Parameter Learning**
  Various algorithms (e.g., MLE)
  Conditional Prob. Table (CPT)

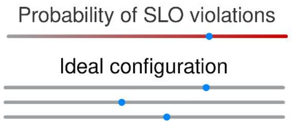
Markov Blanket Selection



Markov blanket

- **Causality filter**
  Extract subset of variables
  that impact SLO fulfillment
- **Behavioral MB**
  MB now contains contextual
  factors & elasticity strategies

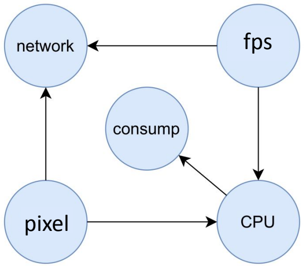
Knowledge Extraction



- **Conditional Inference**
  Estimate impact of different
  deployment configuration
- **Optimize SLOs**
  Adjust processing services
  according to inferred policy

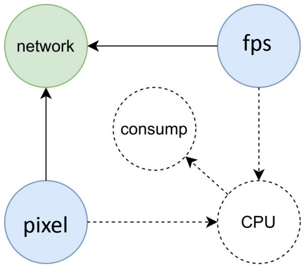
II – Basic Methodology

Bayesian Network Learning



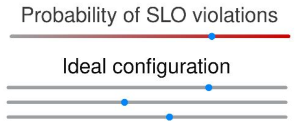
- **Structure Learning**
  Various algorithms (e.g., HCS)
  Directed Acyclic Graph (DAG)
- **Parameter Learning**
  Various algorithms (e.g., MLE)
  Conditional Prob. Table (CPT)

Markov Blanket Selection



- **Causality filter**
  Extract subset of variables
  that impact SLO fulfillment
- **Behavioral MB**
  MB now contains contextual
  factors & elasticity strategies

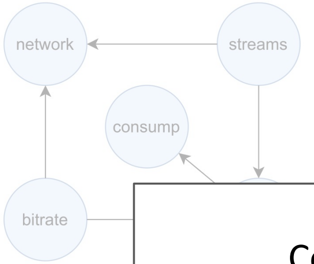
Knowledge Extraction



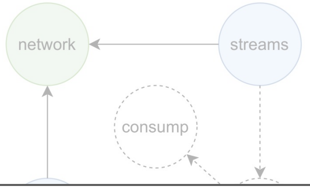
- **Conditional Inference**
  Estimate impact of different
  deployment configuration
- **Optimize SLOs**
  Adjust processing services
  according to inferred policy

II – Basic Methodology

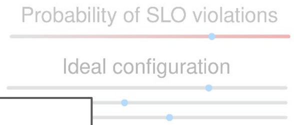
Bayesian Network Learning



Markov Blanket Selection



Knowledge Extraction



Core methodology applied in multiple application.

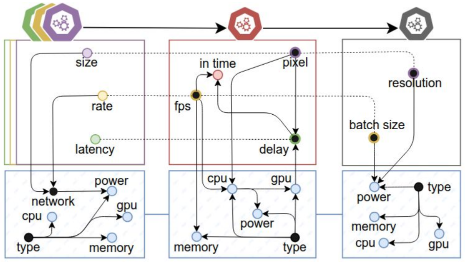
- **Structure Learning**
  Various algorithms (e.g., HCS)
  Directed Acyclic Graph (DAG)
- **Parameter Learning**
  Various algorithms (e.g., MLE)
  Conditional Prob. Table (CPT)

- **Causality filter**
  Extract subset of variables
  that impact SLO fulfillment
- **Behavioral MB**
  MB now contains contextual
  factors & elasticity strategies

- **Conditional Inference**
  Estimate impact of different
  deployment configuration
- **Optimize SLOs**
  Adjust processing services
  according to inferred policy

II – Multiple Applications

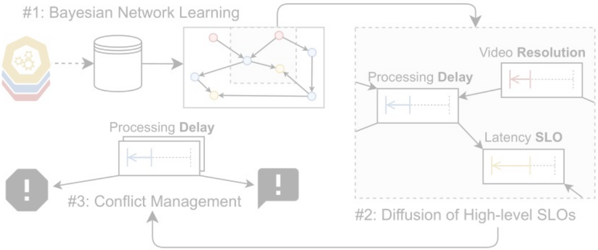
Transitive Requirements [4]



Optimize SLO fulfillment by deploying microservices over **heterogeneous hardware**;

use BNs to analyze & optimize service/device dependencies

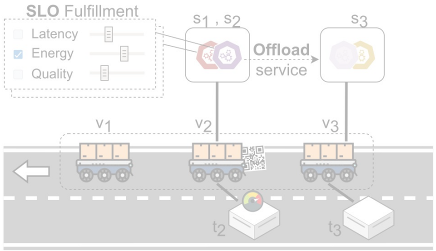
Diffusing High-Level SLOs [8]



Find configurations for subsystems accordion to **high-level SLO** values;

create hierarchy of dependencies and infer lower-level configurations

SLO-Aware Offloading [9]



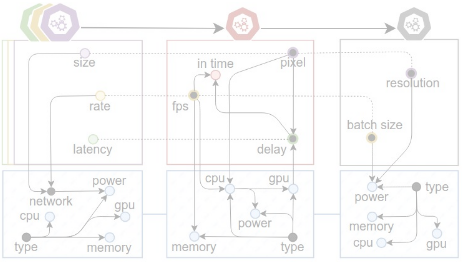
Offload microservices over **heterogeneous hardware**;

estimate effects of service swapping on SLO fulfillment

[4] Sedlak et al., **Markov Blanket Composition of SLOs**, at IEEE Services EDGE 2024
[8] Sedlak et al., **Diffusing High-level SLO in Microservice Pipelines**, at IEEE SOSE 2024
[9] Sedlak et al., **SLO-Aware Task Offloading Within Collaborative Vehicle Platoons**, at ICSOC 2024

II – Multiple Applications

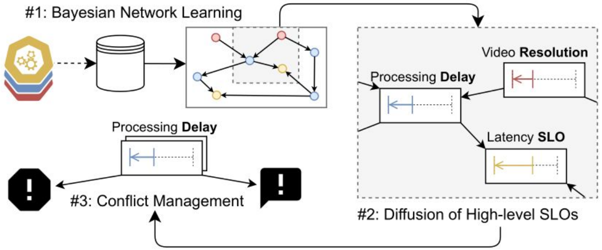
Transitive Requirements [4]



Optimize SLO fulfillment by deploying microservices over **heterogeneous hardware**;

use BNs to analyze & optimize service/device dependencies

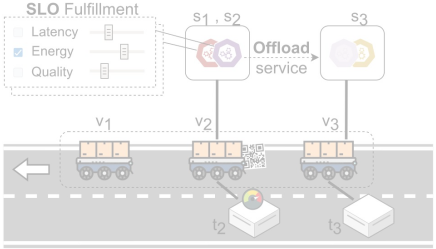
Diffusing High-Level SLOs [8]



Find configurations for subsystems accordion to **high-level SLO** values;

create hierarchy of dependencies and infer lower-level configurations

SLO-Aware Offloading [9]



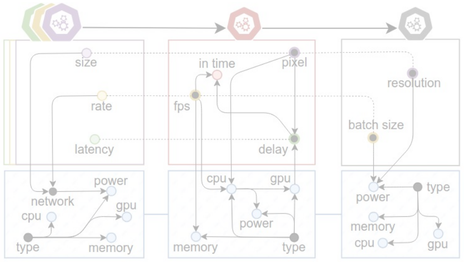
Offload microservices over **heterogeneous hardware**;

estimate effects of service swapping on SLO fulfillment

[4] Sedlak et al., **Markov Blanket Composition of SLOs**, at IEEE Services EDGE 2024
[8] Sedlak et al., **Diffusing High-level SLO in Microservice Pipelines**, at IEEE SOSE 2024
[9] Sedlak et al., **SLO-Aware Task Offloading Within Collaborative Vehicle Platoons**, at ICSOC 2024

II – Multiple Applications

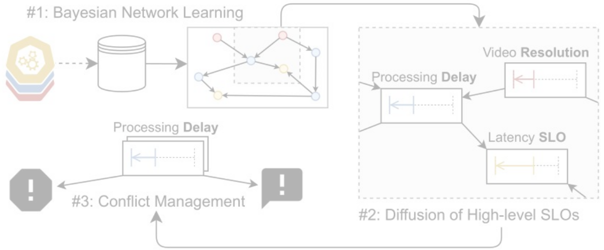
Transitive Requirements [4]



Optimize SLO fulfillment by deploying microservices over **heterogeneous hardware**;

use BNs to analyze & optimize service/device dependencies

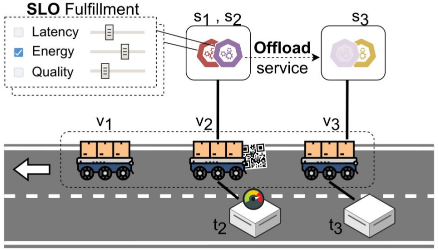
Diffusing High-Level SLOs [8]



Find configurations for subsystems accordion to **high-level SLO** values;

create hierarchy of dependencies and infer lower-level configurations

SLO-Aware Offloading [9]



Offload microservices over **heterogeneous hardware**;

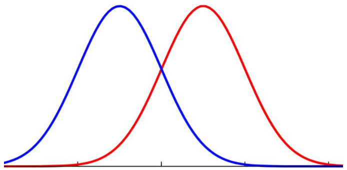
estimate effects of service swapping on SLO fulfillment

[4] Sedlak et al., **Markov Blanket Composition of SLOs**, at IEEE Services EDGE 2024
[8] Sedlak et al., **Diffusing High-level SLO in Microservice Pipelines**, at IEEE SOSE 2024
[9] Sedlak et al., **SLO-Aware Task Offloading Within Collaborative Vehicle Platoons**, at ICSOC 2024

II – Refining our Approach

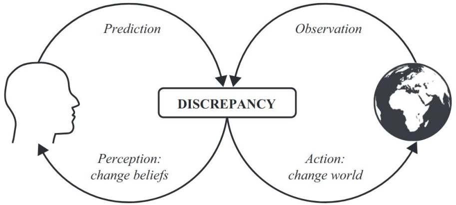
Known Shortcomings

(**1**) BNL requires large amounts of training data upfront
(**2**) can't visit all possible states, so where to start exploring?
(**3**) over time, models get distorted due to variable drifts



Active Inference

Concept from **neuroscience** developed by Friston et al. [10,11]; allows agents to interact with their environment by learning the underlying **generative models** to persist over time



Action-perception cycle in Active Inference [11]

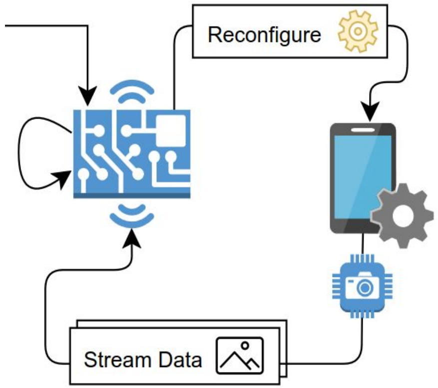
[10] Parr et al., **Active Inference: The Free Energy Principle in Mind, Brain, and Behavior** (2022)
[11] Friston et al., **Designing ecosystems of intelligence from first principles** (2024)

II – Active Inference in CC Systems

Mapping between neuroscience and distributed computing systems [6,12,13]; understanding processing requirements (i.e., SLOs) as a form of **homeostasis,** e.g., cell temperature

Create autonomous components that identify how to ensure requirements and resolve them independently, clear modelling between higher-level and low-level components

Simplify service orchestration in large-scale distributed systems; decentralized decision-making of individual components **avoids** transferring service states to the Cloud



Ensure internal requirements [13]

[6] Sedlak et al., **Adaptive Stream Processing on Edge Devices through Active Inference** (Scheduled for 2025 at Springer ES)
[12] Sedlak et al., **Active Inference on the Edge: A Design Study**, at PerconAI 2024
[13] Sedlak et al., **Equilibrium in the Computing Continuum through Active Inference**, Elsevier FGCS (2024)

II – Active Inference Architecture

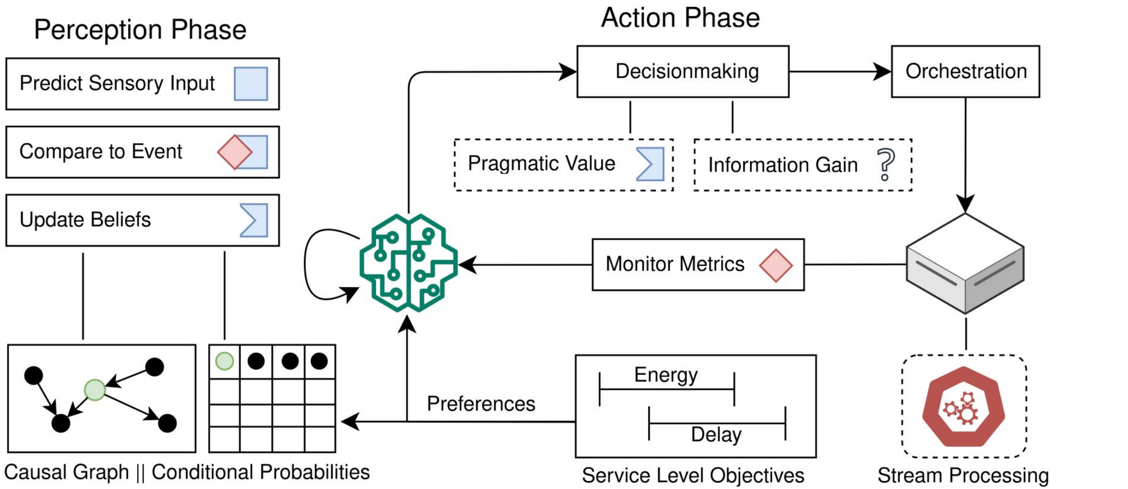
Approach

(1) **Specify** ideal runtime behavior through SLOs

(2) **AIF agents** monitor their environment & collect metrics

(3) **Perception phase** predicts expected SLO fulfillment and adjusts the generative model

(4) **Action phase** orchestrates the processing environment to optimize both SLOs and model



Action and perception cycles performed by the AIF agent to create an accurate model and shape the world [6]

[6] Sedlak et al., **Adaptive Stream Processing on Edge Devices through Active Inference** (Scheduled for 2025 at Springer ES)

II – Active Inference Architecture (cont.)

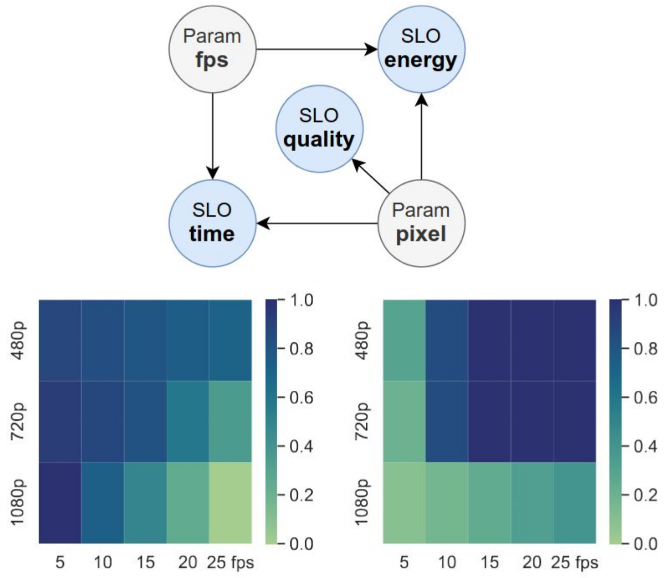
Interpretable behavior [6]

– Empirically verify variable relations in the BNs, e.g., increasing quality (pixel) leads to high energy usage; adjust **parameters** (i.e., pixel & fps) according to SLOs

– Quantified preferences of the agent: (1) expected SLO fulfillment or (2) potential model improvement; determine the behavior of the scaling agent

Continuous composition [4]

– Gradually create increasingly accurate models for individual processing services; continuously compose to estimate the **impact** they have on each other



(b) Pragmatic value (c) Information gain

[4] Sedlak et al., **Markov Blanket Composition of SLOs**, at IEEE Services EDGE 2024
[6] Sedlak et al., **Adaptive Stream Processing on Edge Devices through Active Inference** (Scheduled for 2025 at Springer ES)

![upf logo]

II – Active Inference Architecture (cont.)

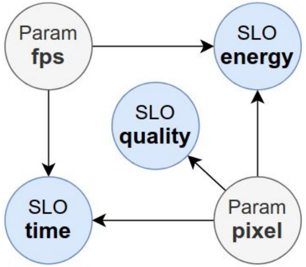
Interpretable behavior [6]

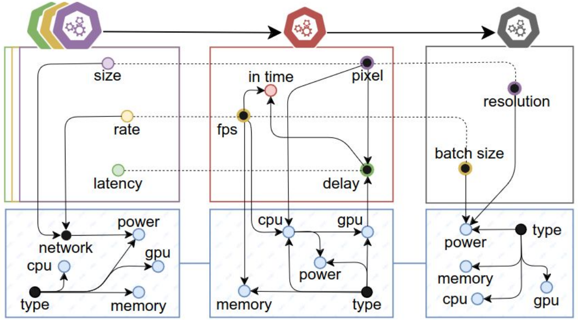
– Empirically verify variable relations in the BNs, e.g., increasing quality (pixel) leads to high energy usage; adjust **parameters** (i.e., pixel & fps) according to SLOs

– Quantified preferences of the agent: (1) expected SLO fulfillment or (2) potential model improvement; determine the behavior of the scaling agent

Continuous composition [4]

– Gradually create increasingly accurate models for individual processing services; continuously compose to estimate the **impact** they have on each other

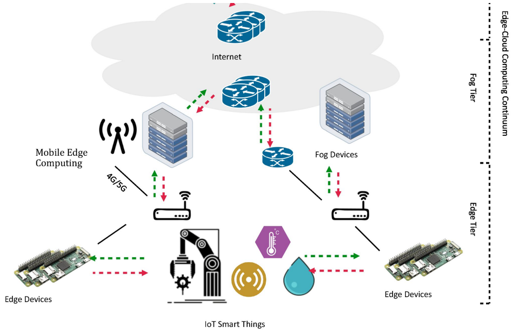




[4] Sedlak et al., **Markov Blanket Composition of SLOs**, at IEEE Services EDGE 2024
[6] Sedlak et al., **Adaptive Stream Processing on Edge Devices through Active Inference** (Scheduled for 2025 at Springer ES)

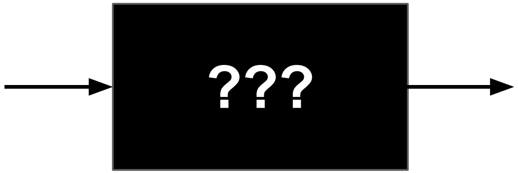
III – Summary
Contemporary Challenges

IoT & Edge enable large-scale distributed services that optimize our daily routines; CC as underlying infrastructure for supporting these services



Resource limitations and device heterogeneity complicate service orchestration; device & service behavior not guaranteed, leads to violated SLOs

Missing explainability for black box ML models; leads to low trust and non-interpretable behavior

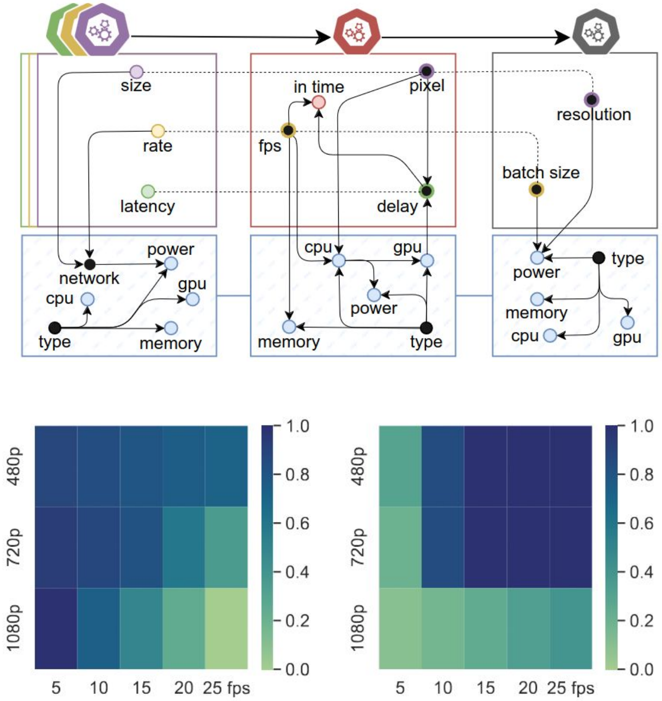


III – Summary
Contributions & Results

Monitor processing services closely, analyze their behavior, and infer optimal elasticity strategy; train a MB that combines all these factors in one model

Dynamically optimize SLOs fulfillment for a network of processing services and heterogeneous devices; orchestrate services (i.e., placement, configuration, replication, etc) according to current context

Active Inference as a natural fit to train behavioral MBs and keep them accurate; balance continuously between ensuring SLOs and improving the model



III – Summary
Limitations & Challenges (1/2)

High cost of running empirical experiments; slow training progress, needs training environment.

→ Train agents in simulated environments; creating accurate environment, like for **digital twinning**

Overhead of extracting baselines from publication; low rigor from generic algorithms, e.g., SB3

→ Support baseline comparison and standardized problem instances in evaluation environment

Governance limited to individual vendor; SLAs only supported by one provider and its subset of nodes

→ Create overarching models for orchestration and client compensation; technical & economical part

Active Inference for Digital Twins: Predicting and Optimising IoT Processing Service Performance

Elena Pretel
LoUISE Research Group,
I3A, University of Castilla-La Mancha
Albacete, Spain
mariaelena.pretel@uclm.es

Boris Sedlak
Vienna University of Technology
Vienna, Austria
b.sedlak@dsg.tuwien.ac.at

Víctor Casamayor-Pujol
Universitat Pompeu Fabra
Barcelona, Spain
victor.casamayor@upf.edu

Elena Navarro
LoUISE Research Group,
I3A, University of Castilla-La Mancha
Albacete, Spain
elena.navarro@uclm.es

Víctor López-Jaquero
LoUISE Research Group,
I3A, University of Castilla-La Mancha
Albacete, Spain
victormanuel.lopez@uclm.es

Pascual González
LoUISE Research Group,
I3A, University of Castilla-La Mancha
Albacete, Spain
pascual.gonzalez@uclm.es

Schahram Dustdar
Vienna University of Technology
and Universitat Pompeu Fabra
Wien and Barcelona, Austria and Spain
dustdar@dsg.tuwien.ac.at

Abstract

Digital Twins (DTs) are emerging as enablers for real-time monitoring and control in IoT systems. In this work, we propose a DT enabled with Active Inference (AIF), a framework rooted in the Free Energy Principle, to endow DTs with predictive and decision-making capabilities. Our approach is evaluated on a realistic edge computing scenario where two co-located video processing services—a computer vision pipeline using YOLOv8 and a QR code reader—compete for limited CPU resources. The DT continuously infers physical twin state, predicts future performance, and selects actions to meet Service Level Objectives (SLOs). In a series of experiments, we validate the predictive accuracy and control effectiveness of the AIF-enabled DT. Notably, the DT achieves a cumulative SLO fulfillment above 80% for both services, and predicted throughput trajectories show high alignment with real observations, confirmed through statistical testing.

CCS Concepts

• **Computer systems organization → Real-time system architecture**.

Keywords

Digital Twins, Predictability, Active Inference, Internet of Things, IoT

IOT 2025, Vienna, Austria

ACM ISBN 979-8-4007-1595-2/25/11
https://doi.org/10.1145/3770501.3770527

ACM Reference Format:
Elena Pretel, Boris Sedlak, Víctor Casamayor-Pujol, Elena Navarro, Víctor López-Jaquero, Pascual González, and Schahram Dustdar. 2025. Active Inference for Digital Twins: Predicting and Optimising IoT Processing Service Performance. In *The 15th International Conference on the Internet of Things (IOT 2025), November 18–21, 2025, Vienna, Austria.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3770501.3770527

1 Introduction

The fast advancement of technology over the last few decades has transformed the way systems are monitored, controlled, and optimized in diverse domains: from manufacturing and logistics to smart cities and healthcare. The rise of smart devices, Internet of Things (IoT), Artificial Intelligence (AI), and Cloud Computing have enabled the emergence of Digital Twins (DTs).

In large-scale and distributed IoT environments, such as smart manufacturing lines, intelligent transportation systems, or e-health care infrastructures, managing the complexity of the system becomes a fundamental challenge [13]. DTs provide a suitable way to monitor, model and manage these IoT systems by creating virtual replicas of physical entities operating under real-time data flows enabling proactive and adaptive decision making [7].

The DT concept was originally coined by Grieves and Vicker in 2003 defined as the coexistence of three elements: (1) a physical space, (2) a virtual space, and (3) a bidirectional connection between both spaces allowing their convergence [4]. Since then, the concept has evolved, currently, DTs are understood as dynamic virtual representations of physical entities that allow real-time monitoring, simulation, prediction, and control of the modelled entities' behaviour [8]. One of the main properties of a DT is the *predictability* [12], which refers to the ability of the DT to anticipate and improve the future behaviour of its physical twin [8]. This property is crucial to enable intelligent decision-making within the DT, in particular when leveraging AI-based techniques.

Upcoming paper at ACM IoT [14]

[14] Pretel E., Sedlak B. et al., **Active Inference for Digital Twins: Predicting and Optimising IoT Processing Service Performance**, at ACM IoT 2025

III – Summary
Limitations & Challenges (1/2)

High cost of running empirical experiments; slow training progress, needs training environment.

→ Train agents in simulated environments; creating accurate environment, like for **digital twinning**

Overhead of extracting baselines from publication; low rigor from generic algorithms, e.g., SB3

→ Support baseline comparison and standardized problem instances in evaluation environment

Governance limited to individual vendor; SLAs only supported by one provider and its subset of nodes

→ Create overarching models for orchestration and client compensation; technical & economical part


Active Inference for Digital Twins: Predicting and Optimising IoT Processing Service Performance

Elena Pretel, LoUISE Research Group, I3A, University of Castilla-La Mancha, Albacete, Spain, mariaelena.pretel@uclm.es
Boris Sedlak, Vienna University of Technology, Vienna, Austria, b.sedlak@dsg.tuwien.ac.at
Victor Casamayor-Pujol, Universitat Pompeu Fabra, Barcelona, Spain, victor.casamayor@upf.edu
Elena Navarro, LoUISE Research Group, I3A, University of Castilla-La Mancha, Albacete, Spain, elena.navarro@uclm.es
Victor López-Jaquero, LoUISE Research Group, I3A, University of Castilla-La Mancha, Albacete, Spain, victormanuel.lopez@uclm.es
Pascual González, LoUISE Research Group, I3A, University of Castilla-La Mancha, Albacete, Spain, pascual.gonzalez@uclm.es
Schahram Dustdar, Vienna University of Technology and Universitat Pompeu Fabra, Wien and Barcelona, Austria and Spain, dustdar@dsg.tuwien.ac.at

Upcoming paper at ACM IoT [14]

[14] Pretel E., Sedlak B. et al., **Active Inference for Digital Twins: Predicting and Optimising IoT Processing Service Performance**, at ACM IoT 2025

III – Summary
Limitations & Challenges (1/2)

High cost of running empirical experiments; slow training progress, needs training environment.

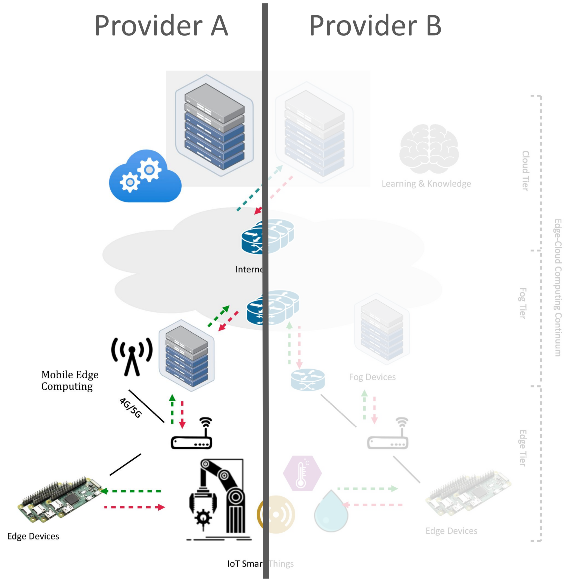
→ Train agents in simulated environments; creating accurate environment, like for **digital twinning**

Overhead of extracting baselines from publication; low rigor from generic algorithms, e.g., SB3

→ Support baseline comparison and standardized problem instances in evaluation environment

Governance limited to individual vendor; SLAs only supported by one provider and its subset of nodes

→ Create overarching models for orchestration and client compensation; technical & economical part

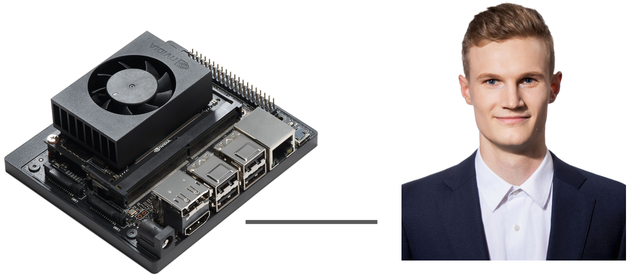


CC infrastructure split between providers

III – Summary
Limitations & Challenges (2/2)

Missing sovereignty over data processing; devices rarely in possession of service consumers

→ Integrate personal devices into infrastructure; combine with personal wallets and trusted env.



Rigid inference quality for recommending actions; prone to violate operational boundaries, e.g., time

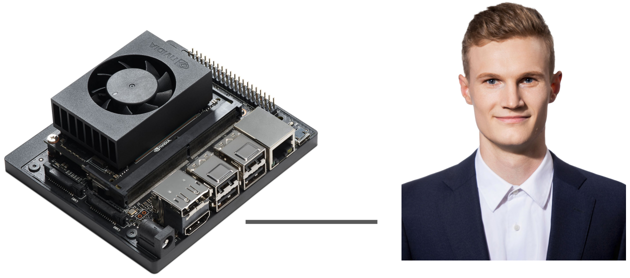
→ Consider resources and context when inferring actions, e.g., smaller graph; create **elastic certainty**

Bayesian network

III – Summary
Limitations & Challenges (2/2)

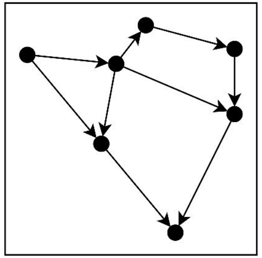
Missing sovereignty over data processing; devices rarely in possession of service consumers

⟶ Integrate personal devices into infrastructure; combine with personal wallets and trusted env.



Rigid inference quality for recommending actions; prone to violate operational boundaries, e.g., time

⟶ Consider resources and context when inferring actions, e.g., smaller graph; create **elastic certainty**



Bayesian network