ACES Autopoietic Cognitive Edge-cloud Services

# Agent-based Modeling in the Edge Continuum using Swarm Intelligence

Melanie Schranz, Ksenia Harshiina, Peter Forgacs, Fred Buining

ADAPTIVE, Venice, IT

Funded by the European Union

# Processing at the edge.

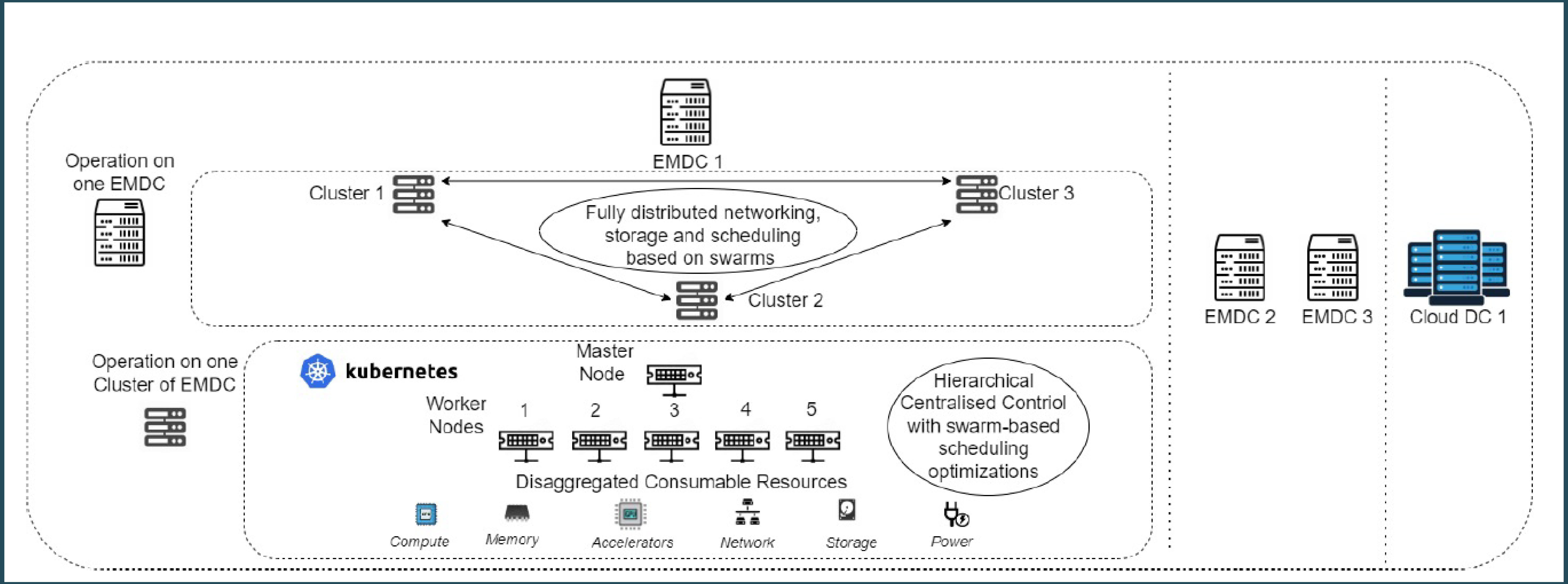
## Emergence of Local Processing Capacity at the Edge

- Advantages: Security, Reliability, Latency, Energy
- Drivers: Upcoming processing tasks of real-time applications (smart grids, mobility, etc.)

## Management of the Edge Continuum

- Dynamic computing landscape
- Edge Continuum: Network of Edge Micro Data Centers (EMDCs)
  - Stringent latency and autonomy requirements
  - Distribution across multiple sites
  - Local limited size
  - Multitenancy and multi-operators
  - Local management
  - Components being concurrent and asynchronous

# The ACES architecture



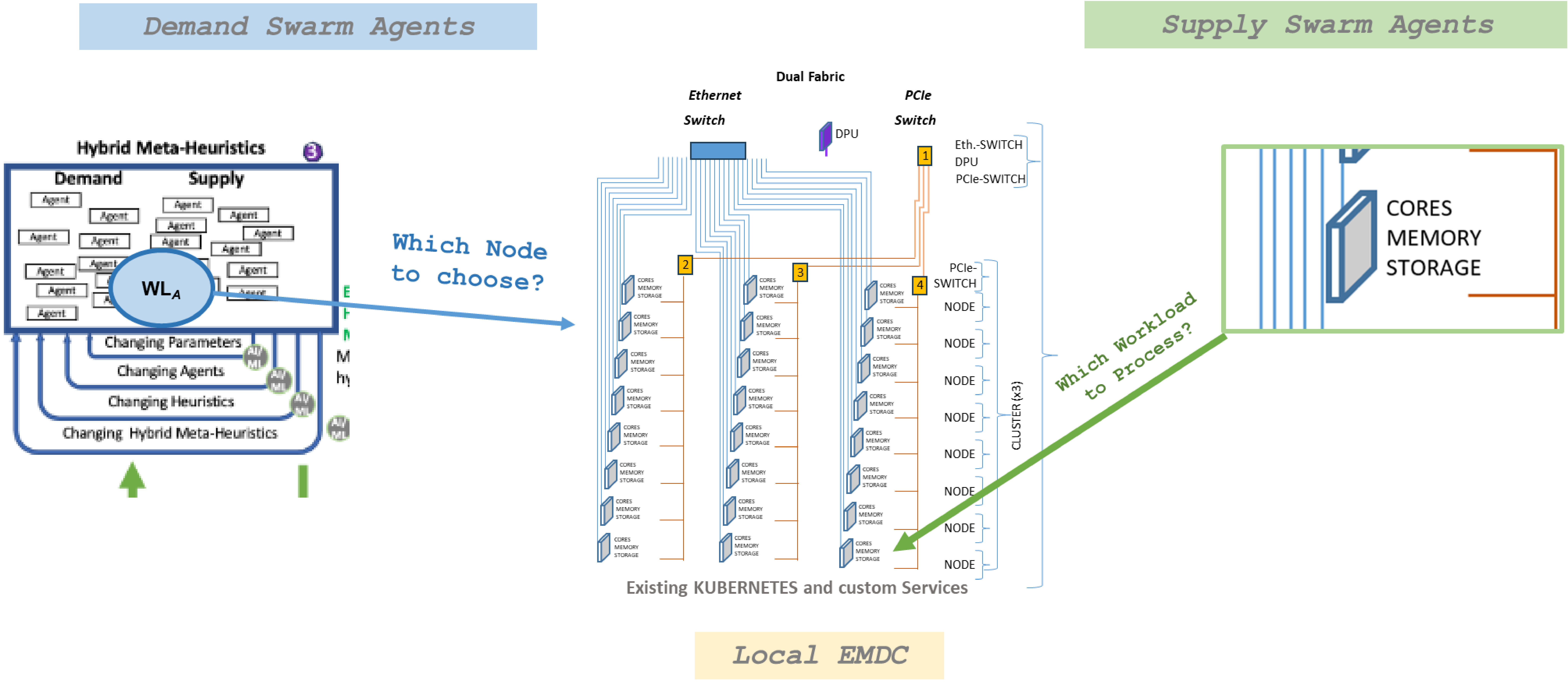

ACES vision is to is to research an evolution of cloud computing, an edgeservices cloud with hierarchical intelligence, specifically Autopoiesis and cognitive behaviours, to manage and automate a compute platform, network fabric, storage resources, virtualization, and analytics to increase resilience while managing simultaneous service constraints.

# Challenges.

- Number of connected devices and their data-producing and data-consuming capabilities,
- Intelligence embedded in edge devices,
- Atomization of monolithic applications,
- Scale, speed, and complexity of edge device interactivity in a zero-trust environment.

Resource allocation, workload scheduling, and data management are challenges that increase in the complexity of the edge orchestration and edge-cloud interaction

# Position in the current architecture.



# Central to our approach: Swarm Intelligence.

- Many similar members
- No central control
- Local rules & knowledge
- Local direct/indirect communication
- Advantages: Adaptability – Robustness - Scalability

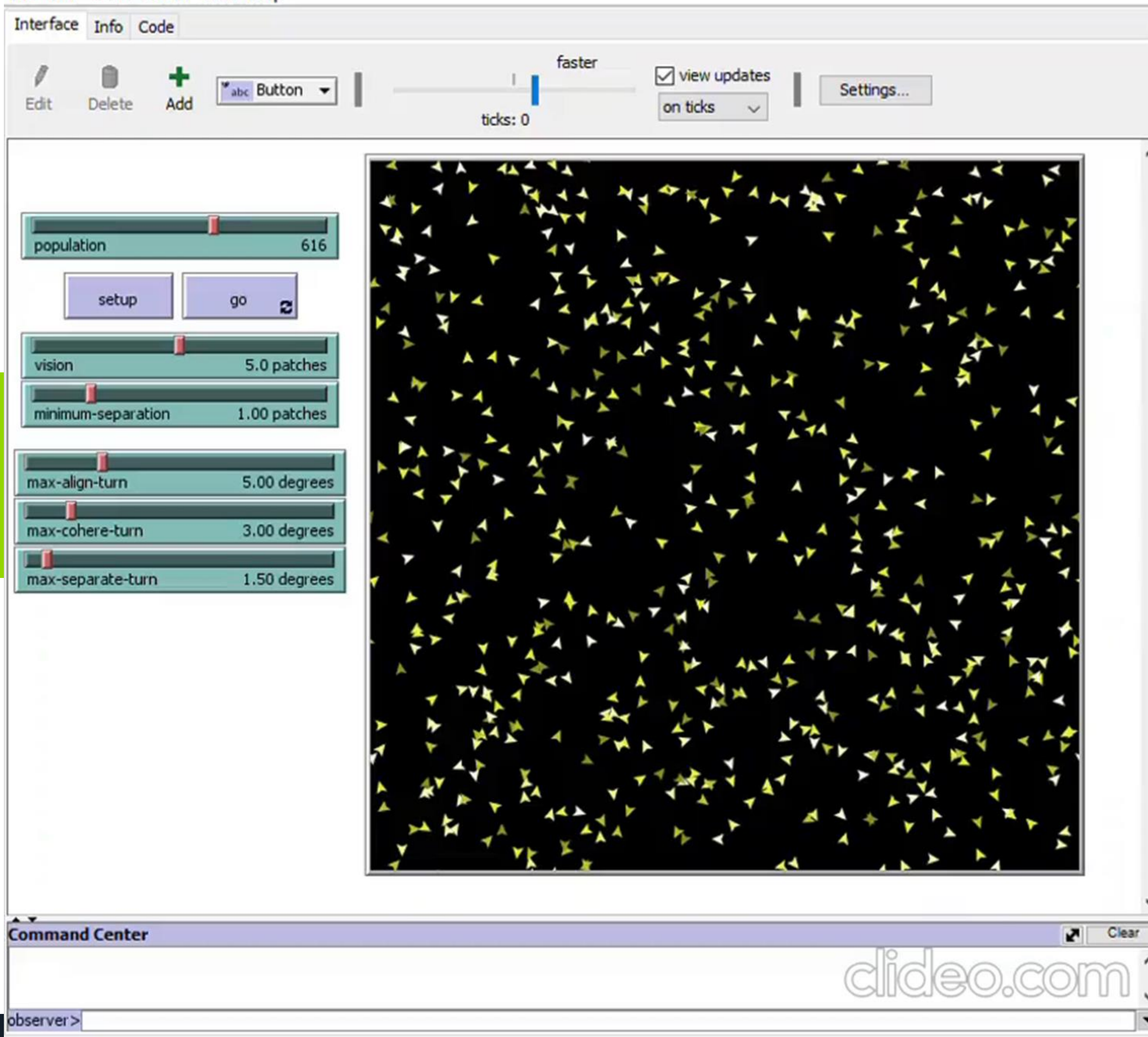
Bird flocking, © pixabay

**System function arises automatically from local rules**

**emergence**

# What is swarm intelligence?

https://www.youtube.com/watch?v=4cPZ2HC2QHg



# Modeling Agents in the Edge-Cloud Continuum.

## Demand Swarm Agents

### Pod Level

- Intended (sequence of) resources
- Response Time
- (Priority?)

Each application is presented as a service $\mathfrak{S}$

Split into a set of related pods $P^s = \{p^s_1, p^s_2, \ldots\}$

The pod $p^s_j$ an choose which of the suitable nodes tc $N^n_i$ for each necessary process step $P^r$

## Supply Swarm Agents

### Node Level

### Pool of Resources

- Resource/s type
- Resource/s capacities

The EMDC $\mathfrak{E}$ ntains a set of nodes $N^r = \{N^r_1, N^r_2, \ldots\}$

with r as the resource type(s).

Each resource $N^r_i$ s a queue . $Q^r_i$

# Agent Collaboration and Self-Organization.

- **Autonomy and Emergence** in the Workload Placement

**Interaction** between demand swarm **agents** and supply swarm agents is **orchestrated through swarm intelligence algorithms:**

1. Demand swarm agents autonomously seek out the most suitable node for workload placement
2. Supply swarm agents determine the optimal workload to process based on available resources and capacity.

- **Collaborative Decision Making**

✓Efficiently allocate workloads to nodes,
✓Adapt to changing workloads and resource abilities,
✓Optimize processing,
✓Optimize latency, and resource utilization.

# Challenges in Modelling Agents.

## Pool of Resources

- Single Resource Requests for Pod Processing
- Stability and Performance Enhancement

## Application Types

- Long-Running Applications (LRAs)
- Batch Processing
- Stream Processing

## Relationship among Pods

- Interpod Relationships
- Parallel Processing and Dependencies
- Dynamic Pod Creation for Response Time Optimization

# Candidate Algorithms.

## Hormone Algorithm

Inspiration from the biological endocrine system, which regulates various metabolic processes within our bodies:

1. Production: Hormone Synthesis by Nodes
- Controlled by Number of Demand Swarm Agents

$$H^m = \frac{1}{|Q_i^m| + \beta}$$

2. Evaporation: Gradual Hormone Reduction
- Controlled by Evaporation Rate (α)

$$H_{i,t+1}^m = H_{i,t}^m \cdot (1 - \alpha)$$

3. Diffusion: Hormone Propagation Between Nodes
Controlled by Number of Demand Swarm Agents

$$\Delta H = H_i^m \cdot \gamma \qquad H_i^m - = \Delta H$$

4. Diffusion Through Pod Movement: Influence of Pod Movement on Hormones
- Controlled by Diffusion Factor (δ)

$$\Delta H = H_i^m \cdot \delta \qquad H_i^m - = \delta H$$

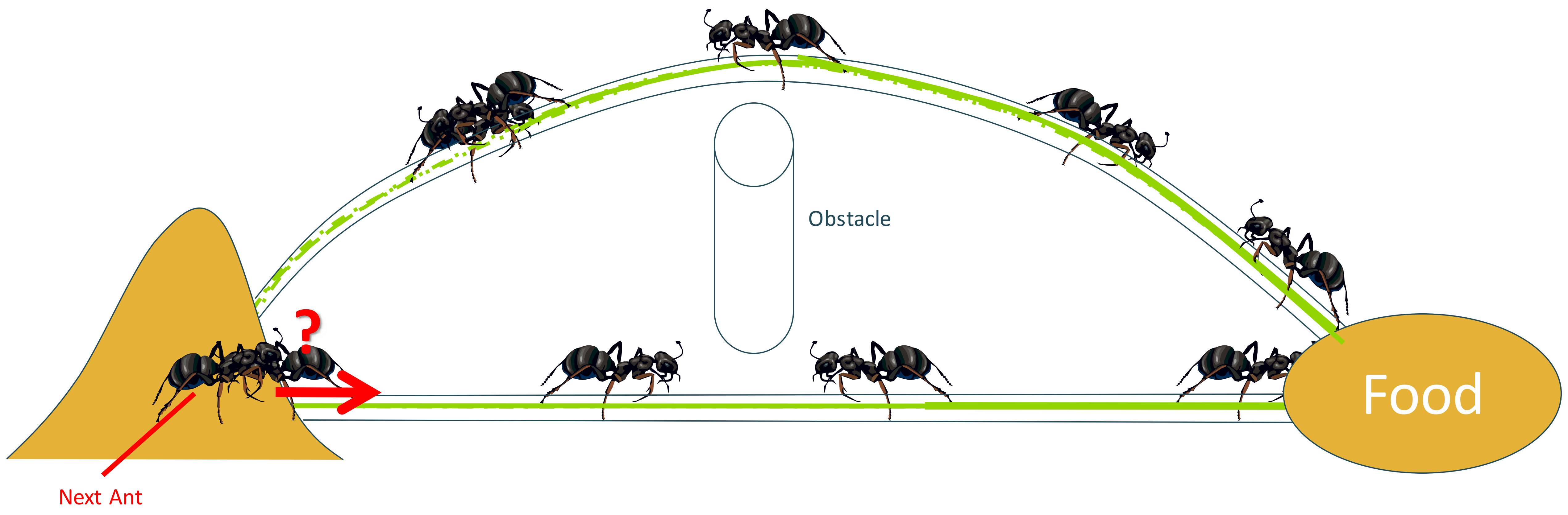
5. Attraction: Hormone-Induced Pod Attraction to Nodes
- Attraction Factor (ε)

$$attraction = \sum_{i,m} H_i^m \cdot \varepsilon^n$$

# Candidate Algorithms.

A single ant is unlikely to be “intelligent”,

but in the system they are unbeatable!



# Candidate Algorithms.

## Ant Algorithm

Ant algorithms draw inspiration from the decentralized foraging behavior of ants, a natural phenomenon where ants can efficiently find near optimal paths to food sources without relying on global knowledge.

1. Trail Following: Probabilistic Node Selection Influenced by Local Pheromones and Pod Heuristic

$$P_{i,j} = \frac{\tau_{i,j,d} + \alpha\eta_{i,j}}{1 + \alpha(N_i - 1)}$$

2. Trail Laying: Pheromone Update After Pod Processing Memory of Processing Time and Resource Utilization

$$\tau_{x,d} \leftarrow \tau_{x,d} + r(1 - \tau_{x,d})$$

3. Evaporation: Periodic Pheromone Fade Controlled by Evaporation Rate (p)

$$\tau(t+1) = \tau(t)(1-p)$$

ACES Autopoietic Cognitive Edge-cloud Services

# Thanks for your attention!

Questions?

schranz@lakeside-labs.com

Funded by the European Union