An ILP-based Real-Time Scheduler for Distributed and Heterogeneous Computing Environments

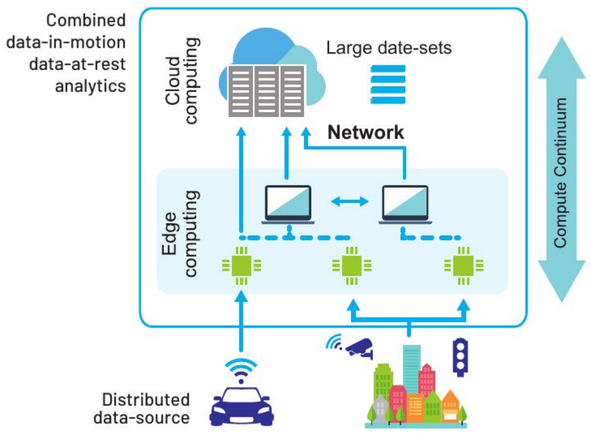
Eudald Sabaté[1,2], Maria A. Serrano[1], Eduardo Quiñones[1]

[1]Barcelona Supercomputing Center (BSC); [1,2]Universitat Politècnica de Catalunya (UPC)
{eudald.sabate, maria.serranogracia, eduardo.quinones}@bsc.es

MOTIVATION

New Distributed Computational Environments

- The computational challenges of smart cities and connected cars can be addressed by **distributing computation across the compute continuum**, from edge to cloud



A Distributed Computing Framework

- **COMPSs** is a task-based development framework for distributed and heterogeneous computing environments
- COMPSs workflows are represented by a DAG composed of **tasks** (nodes) and **data dependencies** (edges)

Challenge

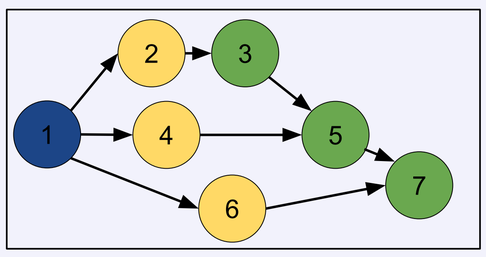
- Develop **new scheduling solutions** capable of efficiently distributing computation across the compute continuum while **providing real-time guarantees**

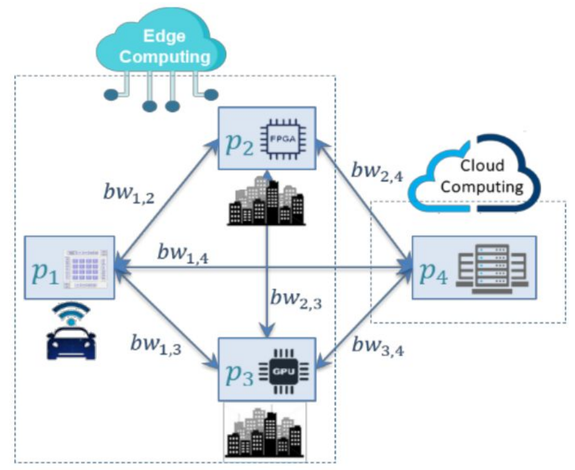
SYSTEM MODEL DESCRIPTION

The DAG-based Tasking Model

- Represents the **timing behavior of a COMPSs workflow**, based on a DAG-based scheduling model extension used in critical real-time systems [1]
  1. Workflows are represented as a DAG in which a node i executed on the compute resource s is represented with an upper bound execution time $C_{i,s}$
  2. An edge between nodes i and j is represented with the data transfer size $z_{i,j}$

```
@task(returns = Configuration)
def init_conf():
   ...
@task(returns = dict, step=IN, conf = IN)
def compute_step(step, conf):
   ...
@task(dict1 = INOUT, dict2 = IN)
def merge(dict1, dict2):
   ...
num_steps = 3
conf = init_conf()
for i in range(num_steps):
   step_res = compute_step(i, conf)
   merge(result, step_res)
```





The Compute Continuum Model

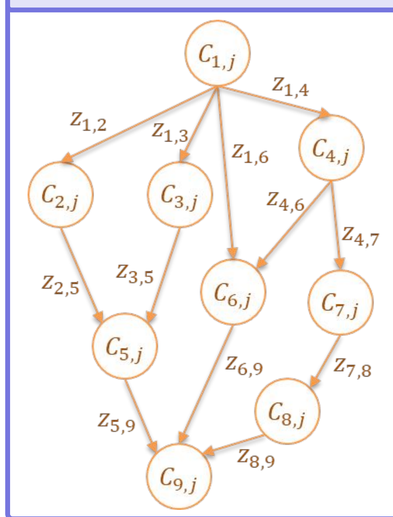
- Represents the set of **computing resources** and the **communication network links** upon which a COMPSs workflow executes
  - A computing resource is characterized by the type (GPU, CPU, FPGA) and the computation capabilities
  - A communication link is characterized by the bandwidth

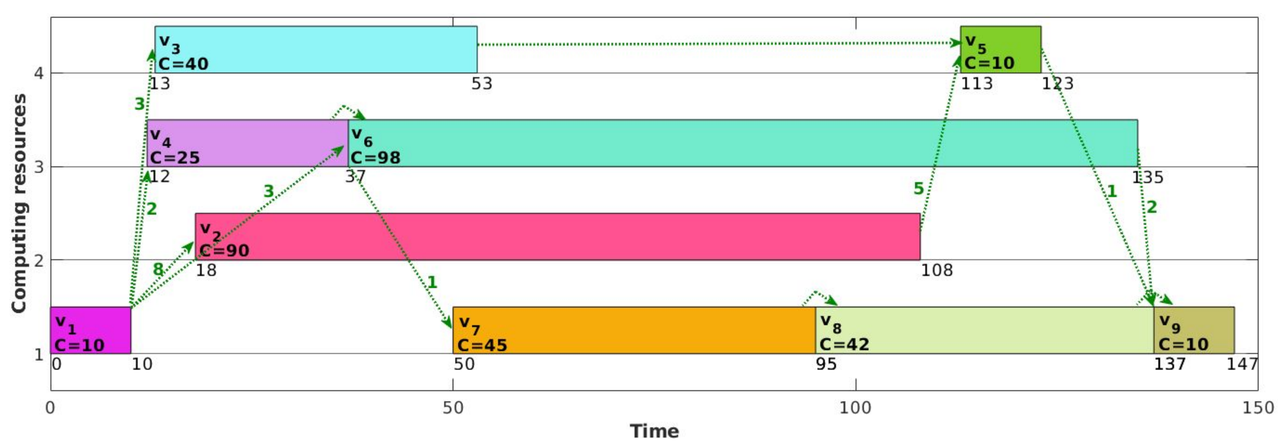
ILP-based SCHEDULING STRATEGY

ILP Formulation

1. **Optimally distributes a COMPSs workflow** (represented by the DAG-based model) on a distributed computing environment (represented by the compute continuum model) considering:
   - The set of $C_{i,s}$ of each COMPSs task
   - The communication cost of data transfers among COMPSs tasks
2. Provides a **minimum end-to-end upper bound response time** of the COMPSs workflow

Example





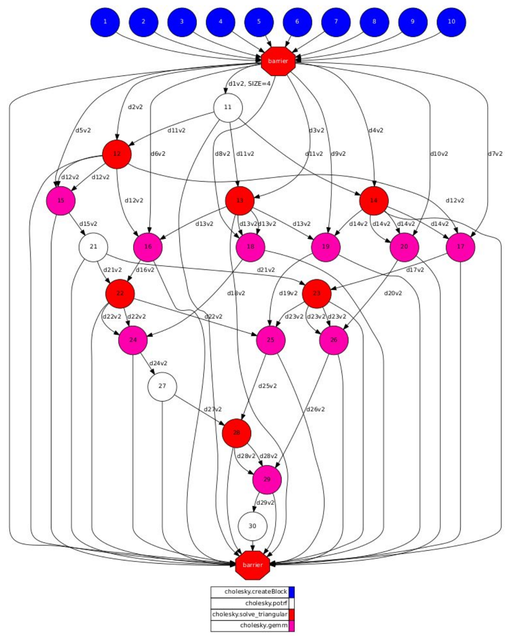
PRELIMINARY EVALUATION

Experimental Setup

- **DAG-based model**
  - COMPSs workflow composed of 30 COMPSs tasks implementing a Cholesky factorization of a 2048×2048 real floating-point matrix
- **Compute continuum model**
  - Computing resources: a four-core Intel(R) Core(TM), and a four-core ARMv8
  - Communication link: IEEE 802.11g at 54 Mbps

Preliminary Results

Response time (in seconds) of the Cholesky factorization				
ILP Scheduler	COMPSs baseline schedulers			
	FIFO	FIFO + Data Locality	LIFO	Load Balancing
8.16	16.36	12.53	14.68	11.92



[1] M. A. Serrano, A. Melani, M. Bertogna, E. Quiñones, "Response-time analysis of DAG tasks under fixed priority scheduling with limited preemptions", in DATE 2016

CLASS Edge and Cloud Computation: A Highly Distributed Software for Big Data Analytics

This work was supported by EU project CLASS from the Horizon 2020 research and innovation programme under the agreement No 780622