



**POLITECNICO  
MILANO 1863**

# Dynamic Application Autotuning for Approximate Computing

Cristina Silvano

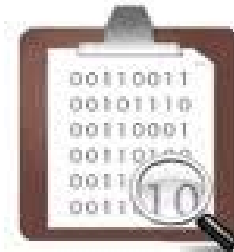
# Approximate Computing Applications



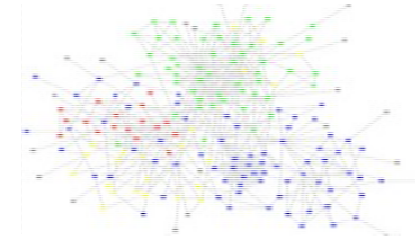
**Image Processing**



**Machine Learning**



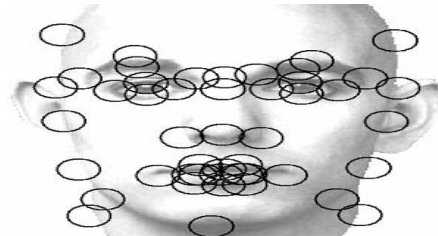
**Big Data Analytics**



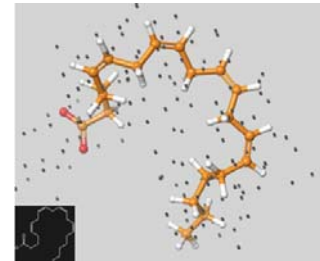
**Graph Analytics**



**Multimedia Applications**



**Computer Vision**



**Drug Discovery**

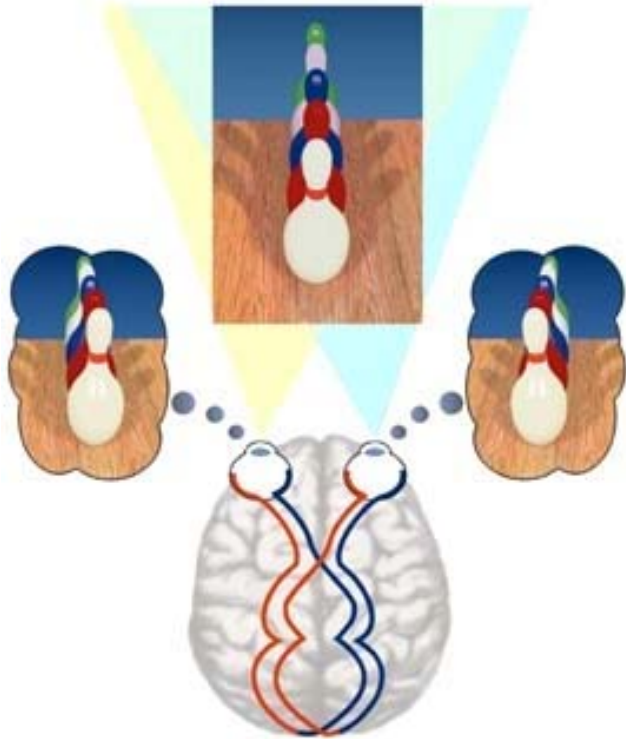


**Traffic Prediction**

**100% computation accuracy is not always required...**

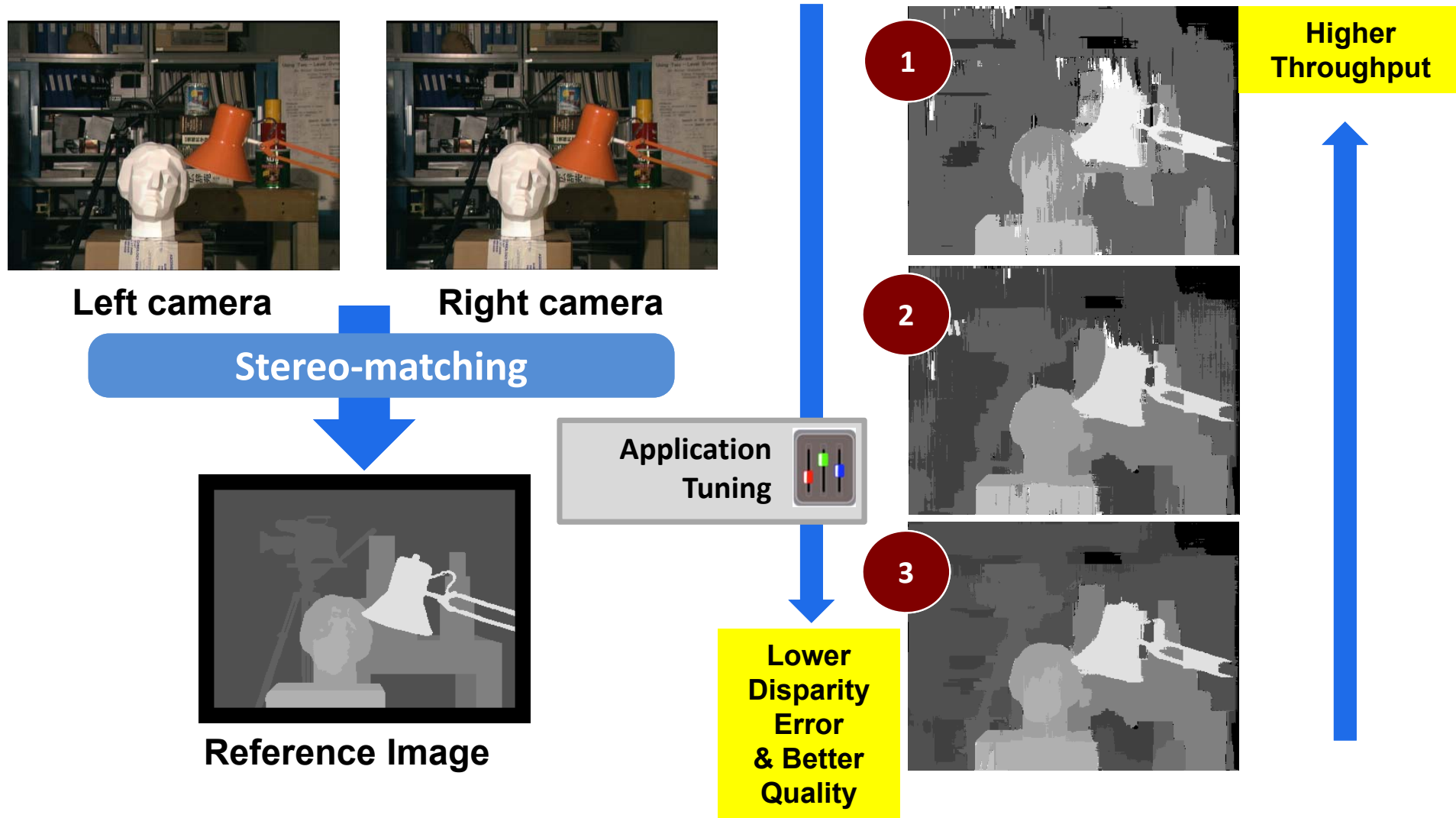
**Approximation offers opportunities for trading off Accuracy vs. Performance vs. Energy**

# The human eye stereo-matching

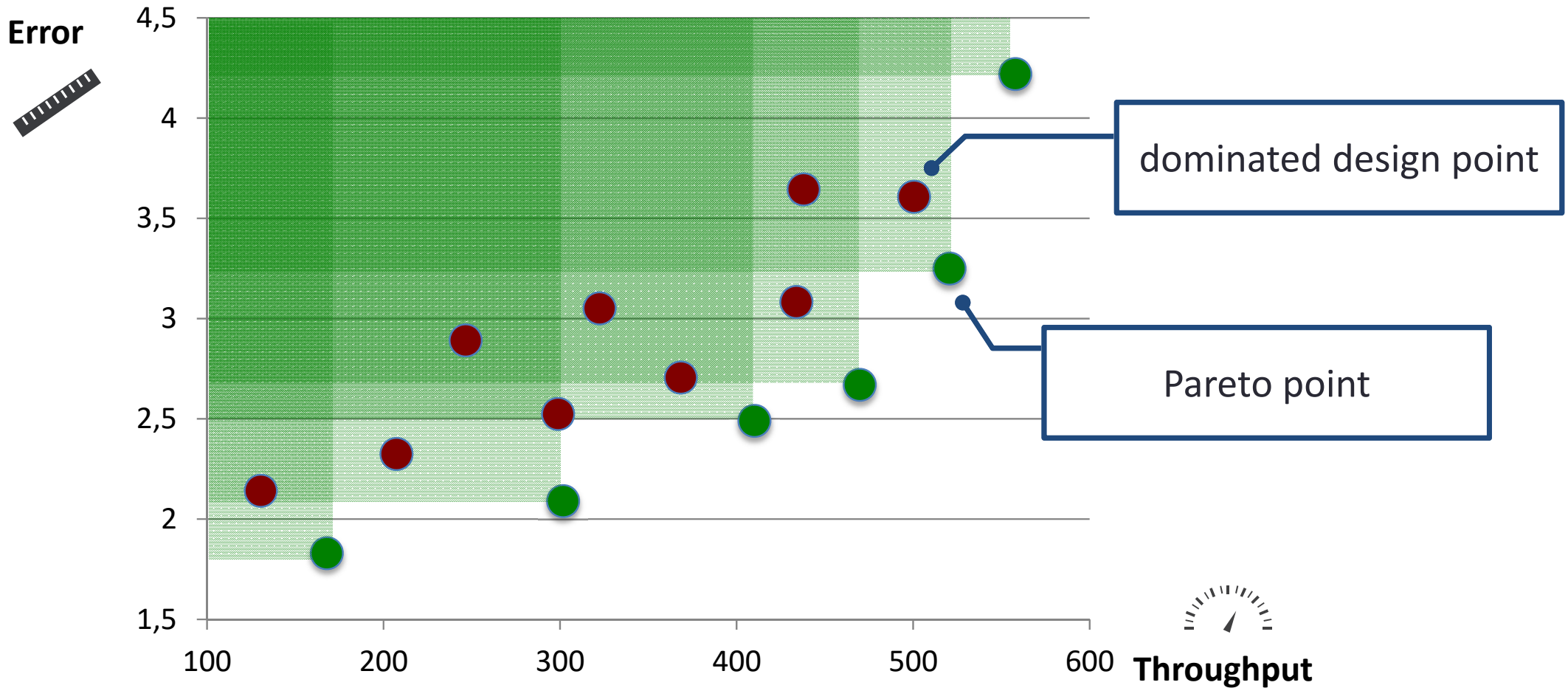


**2 eyes → third dimension**

# Stereo-matching: Pixel Disparity Error vs Throughput



# Approximate Computing: Pareto Points



# Dynamic Autotuning

The diagram consists of three horizontal bars: a red bar on the left, a green bar in the middle, and a blue bar on the right. Three blue arrows point downwards from each bar to its corresponding text block below.

At **runtime** according to the computation evolution

**Automatic**

**Tuning:**

**1:** to adjust in musical pitch or cause to be in tune: *tune a guitar*

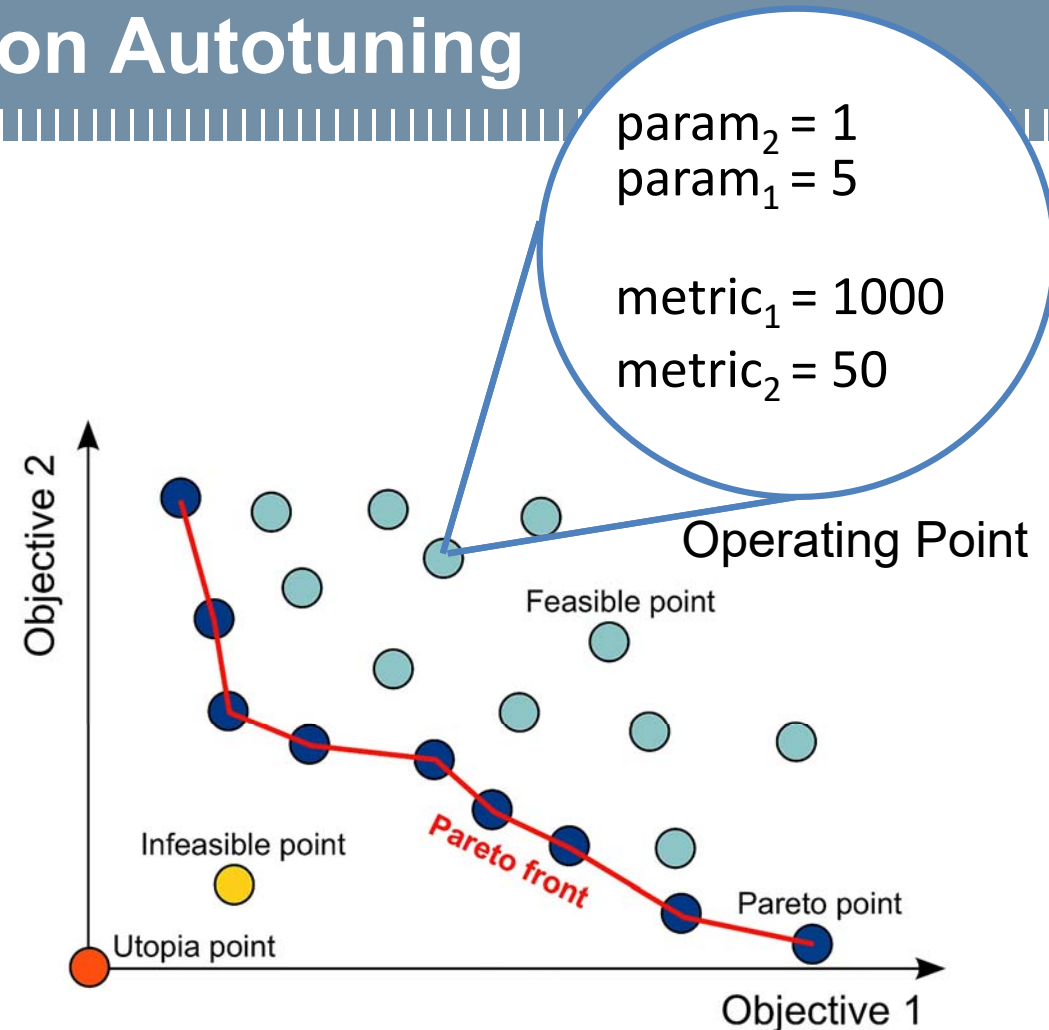
**2 :** to adjust for precise functioning: *tune up an engine*

**In our context:**

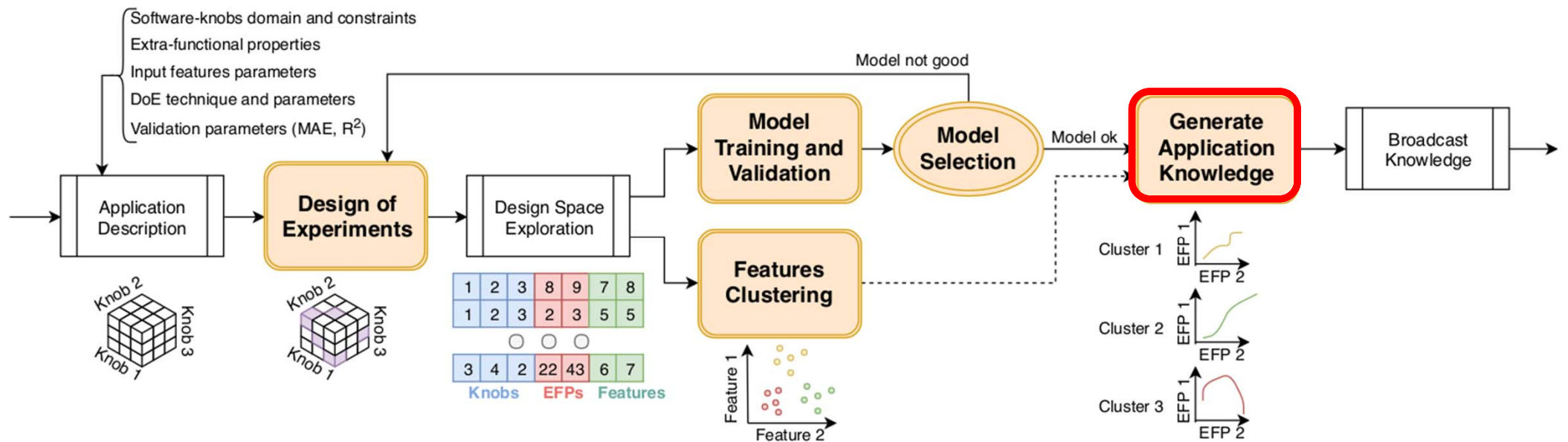
**3: To adjust the values of application parameters to optimize the application metrics**

# Design-time Phase: Application Autotuning

- Best practice is to write parametric code with software parameters:
  - Number of iterations
  - Application-specific parameters
- At design-time we extract the application knowledge:
  - Instrument the application
  - Design Space Exploration
  - Machine-learning Models
  - Store the Pareto front to get the best tradeoffs



# Design-time: Learning the Application Knowledge

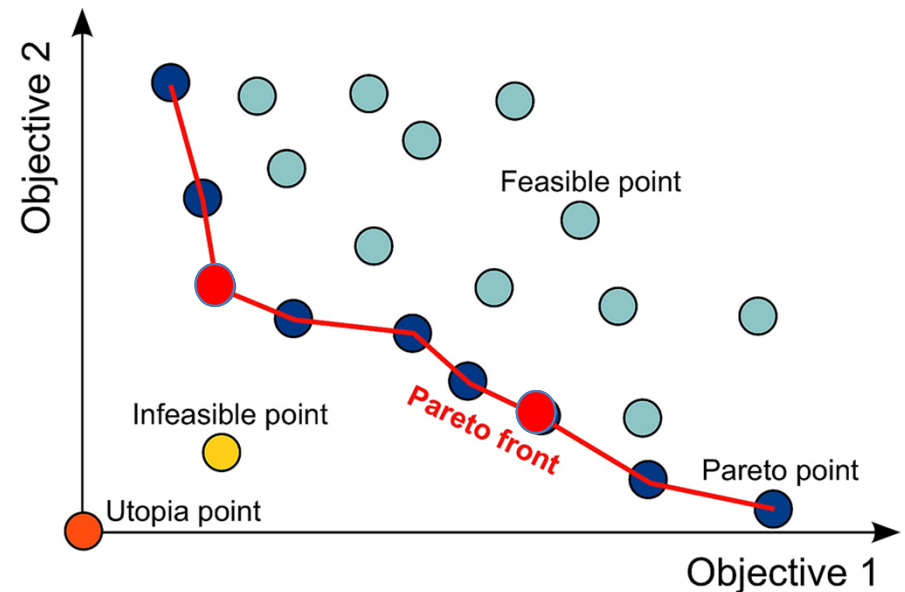
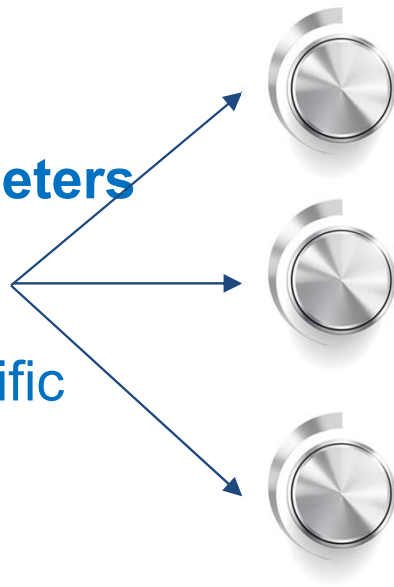


# Dynamic Autotuning

*It is a way to constantly improve performance/energy tradeoffs with low developer effort over a wide range of run-time situations*

## Dynamic SW parameters

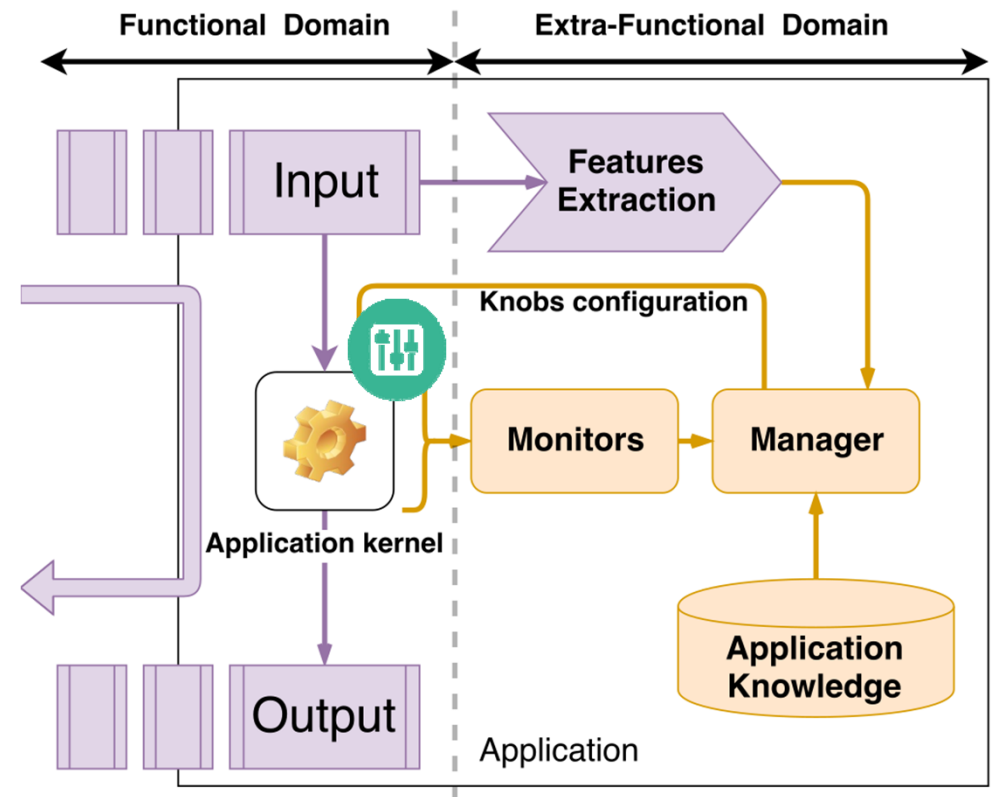
- num sw-thread
- loop perforation
- application-specific



# mARGOT Dynamic Autotuning Framework

It enhances a target application with an **adaptation layer**

- It is a C++ library to be linked to the target application
- Separation of concerns between functional and extra-functional domains.



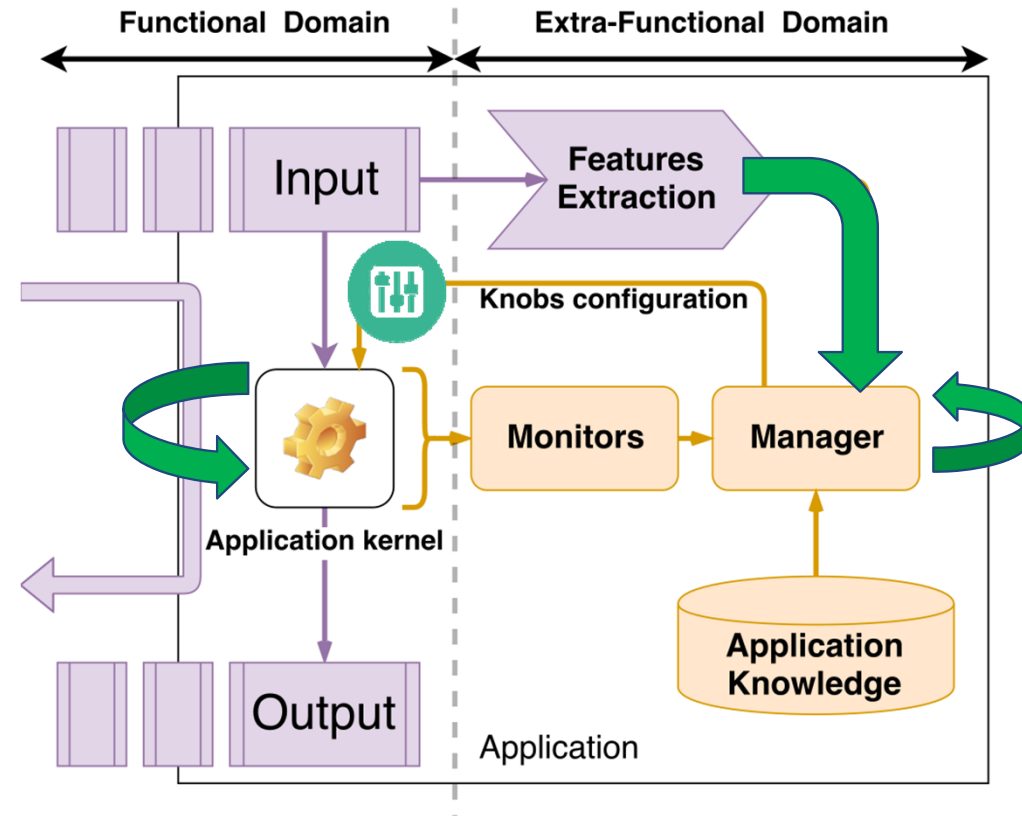
**Public repository:** [https://gitlab.com/margot\\_project/core](https://gitlab.com/margot_project/core)

D. Gadioli, E. Vitali, G. Palermo, C. Silvano, "mARGOT: a Dynamic Autotuning Framework for Self-aware Approximate Computing", IEEE Trans. on Computers, Nov. 2018.

# mARGOT Dynamic Autotuning Framework

mARGOT provides an adaptation mechanism to react to changes in:

- Application requirements
- Application-knowledge due to online learning
- System monitoring values
- Data-features extracted from input data (such as image resolution)





# *What sort of society challenges could be addressed by exploiting the ANTAREX technologies?*



**POLITECNICO**  
MILANO 1863



Universidade do Porto  
Faculdade de Engenharia  
**FEUP**

**ETH**

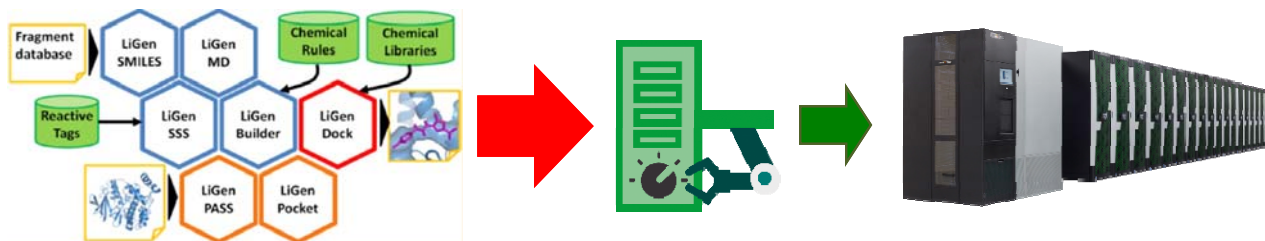
Eidgenössische Technische Hochschule Zürich  
Swiss Federal Institute of Technology Zurich

*inria*  
informatics mathematics



IT4Innovations  
national  
supercomputing  
center





**ANTAR**  <sup>10<sup>18</sup></sup>



European  
Commission

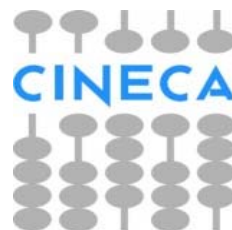
Horizon 2020  
European Union funding  
for Research & Innovation



**Dompé**



**POLITECNICO  
MILANO 1863**

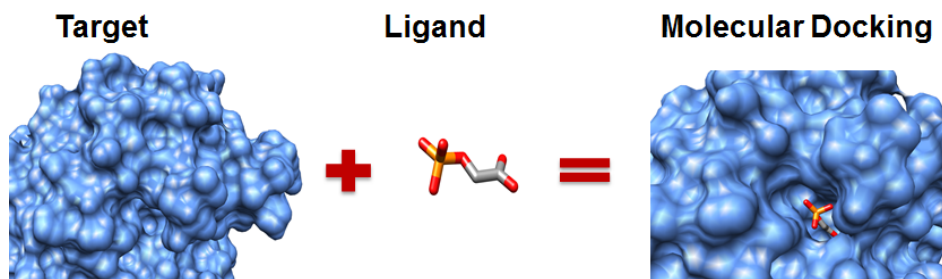


# Autotuning Geometric Docking for HPC Accelerated Drug Discovery

# HPC Accelerated Drug Discovery



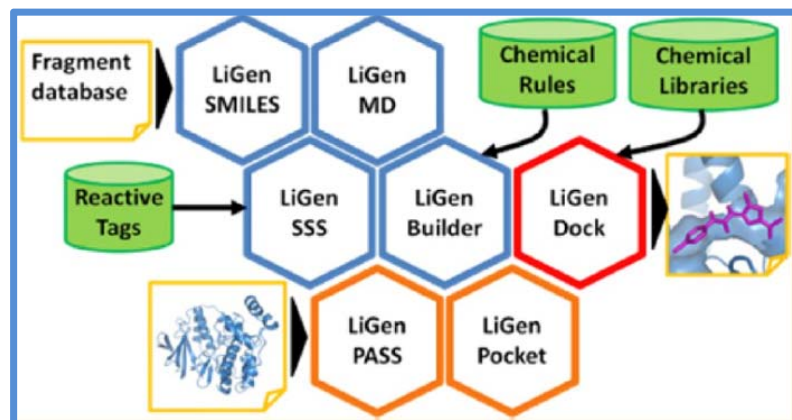
## Need of HPC in Drug Discovery: HPC Molecular Simulations



POLITECNICO  
MILANO 1863

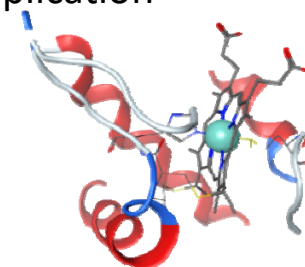


Dompé



Developing energy and resource efficient algorithms  
Using self-functionalities to adapt and scale-out the application

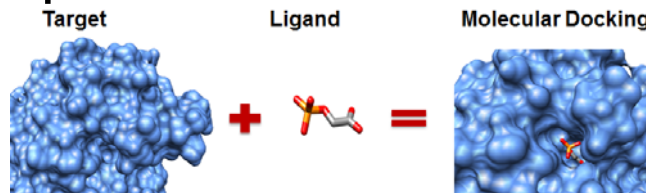
**Exascale-ready HPC Virtual Screening**



LiGen HPC application for drug discovery

**Molecular docking** is a method to estimate the preferred 3D position and shape of a candidate drug (ligand) in the target protein pocket when bound to each other

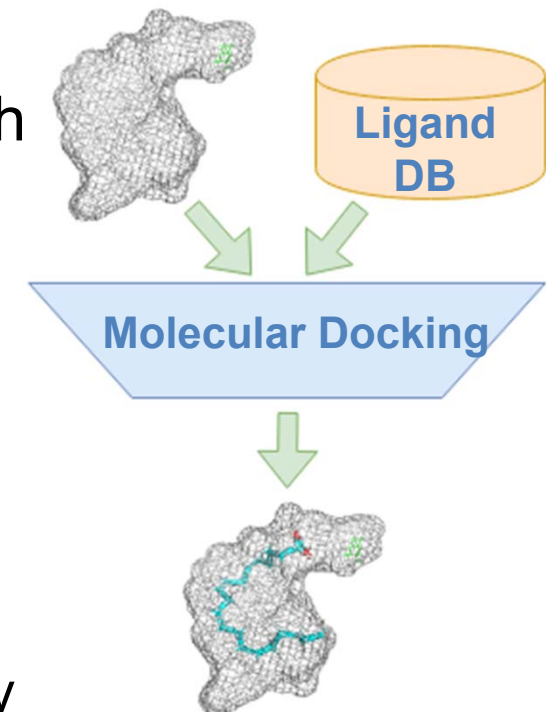
- **Geometric Docking**



- **Shape Complementarity: 3D geometric matching search to find out compatible pairs and most suitable poses**

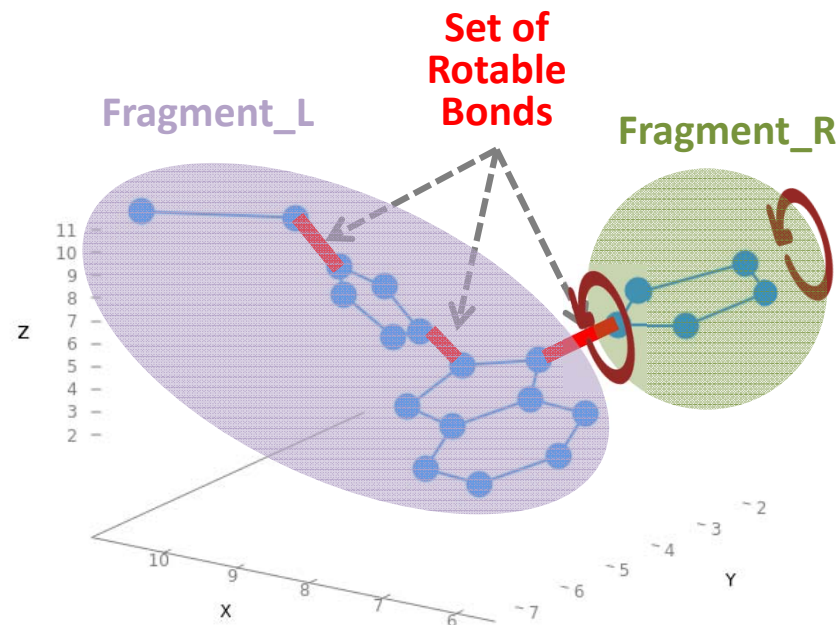
- **Pharmacophoric Docking**

- **Molecular Simulation:** exploration of a large energy landscape given by chemical & physical interactions



# GeoDock Tuning Parameters

Expose software-parameters from the geometric docking application



Low Precision  
Rotation Step



High Precision  
Rotation Step



#Iterations

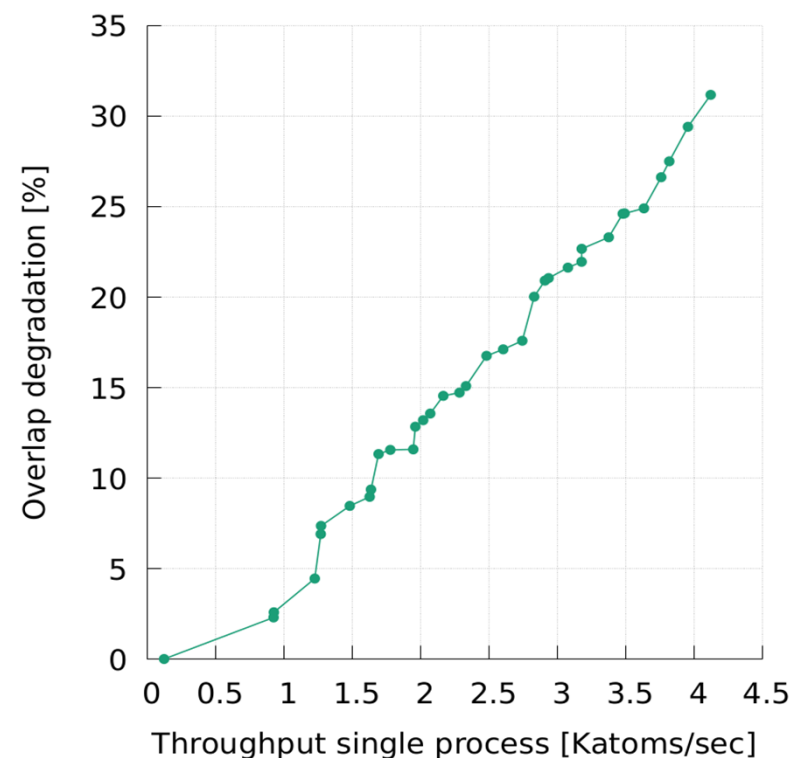
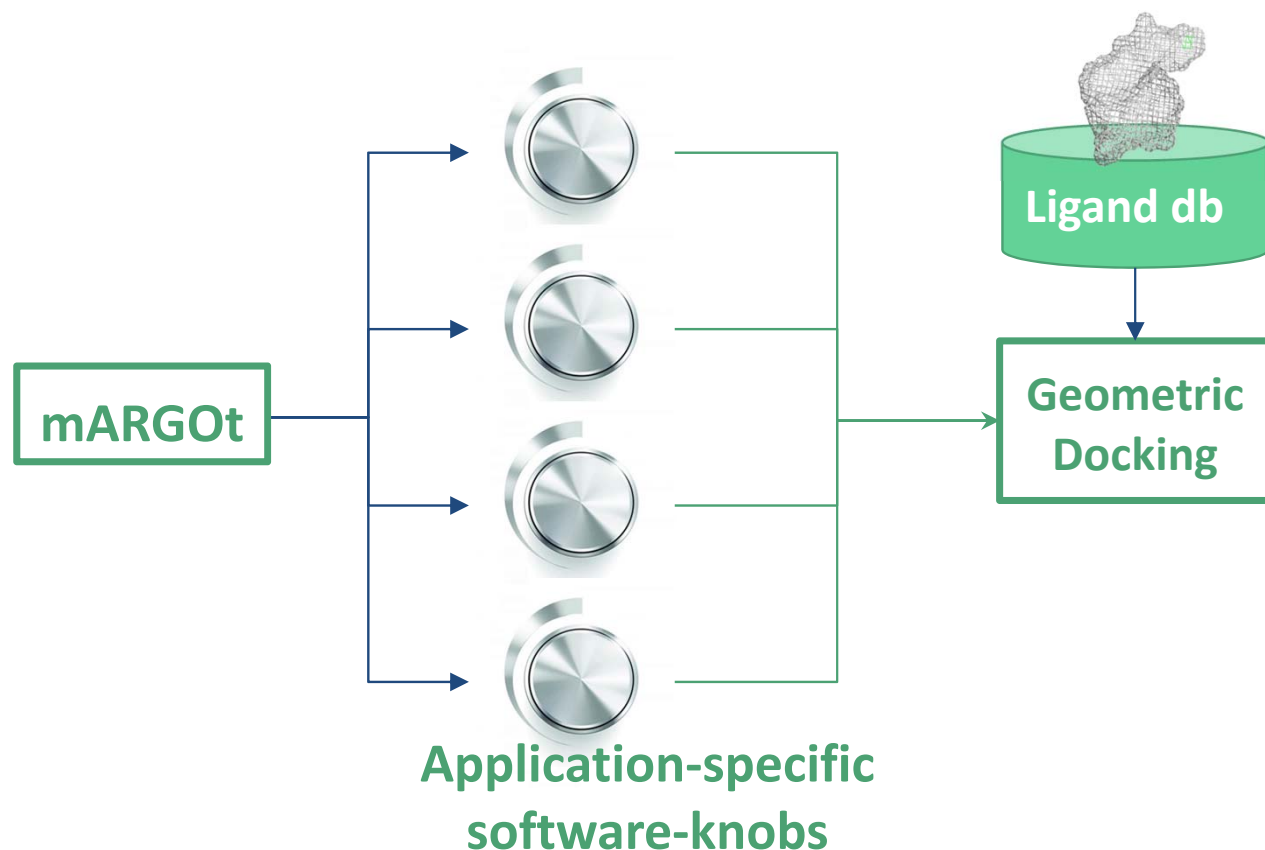


Size Threshold



# Dynamic Autotuning of GeoDock

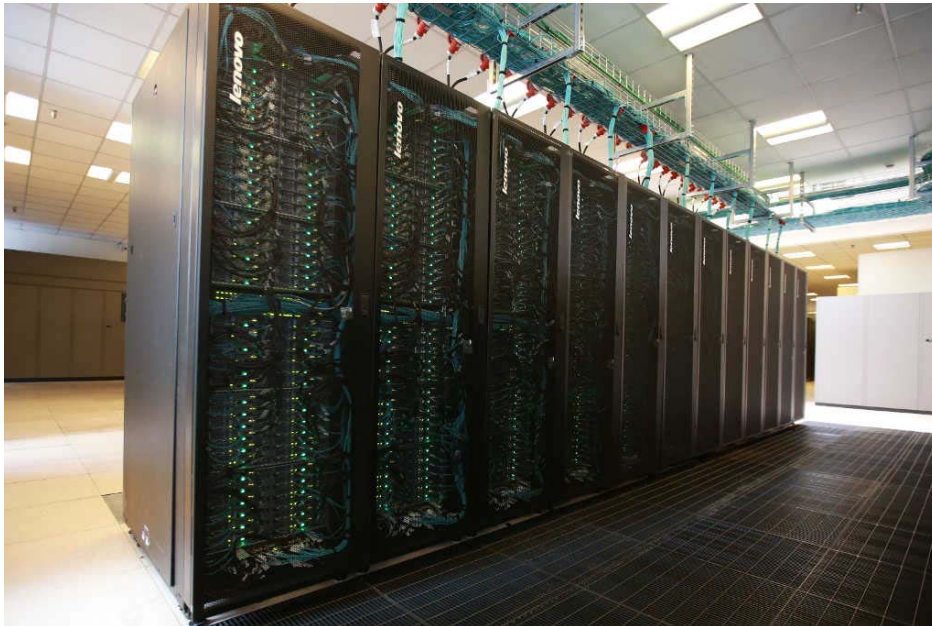
Expose software-knobs to get trade-offs between accuracy and throughput



## Marconi: the most powerful public supercomputer in Italy



- **No. 19 in Top500 and No.4 in Europe: Marconi Intel Xeon Phi:** 10.38 PetaFlops (Linpack performance) 18.8 PetaFlops (peak performance) with 348,000 cores. Site: Casalecchio di Reno, Bologna (Italy)

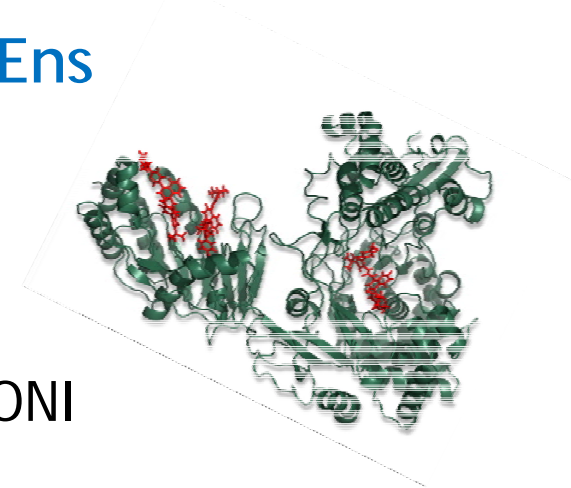


- **Marconi is the Cineca's Tier-0 system, co-designed by Cineca and Lenovo based on the Lenovo NeXtScale platform and Intel® Xeon Phi™ product family alongside with Intel® Xeon® processor and Intel Omni-Path**

# LiGen GeoDock: The Zika virus experiment

## EXSCALATE: ExaScale smArt pLatform Against pathogEns

- 1.2 Billion ligands dataset (candidate drugs)
- 26 Zika binding sites (pharmacological targets)
- 8 Trillion poses scored (by GeoDock)
- 260 TeraByte of stored data
- About 900K Threads on 300k cores on 10 petaFLOPs MARCONI
- 1 MW measured power consumption
- Run Time to Solution: 3.2 h for 1 out of 26 sites (run in Jan 2019)
- Total Time to Solution: 3.5 days (84 h) for 26 sites
- Energy to Solution: 84 MWh



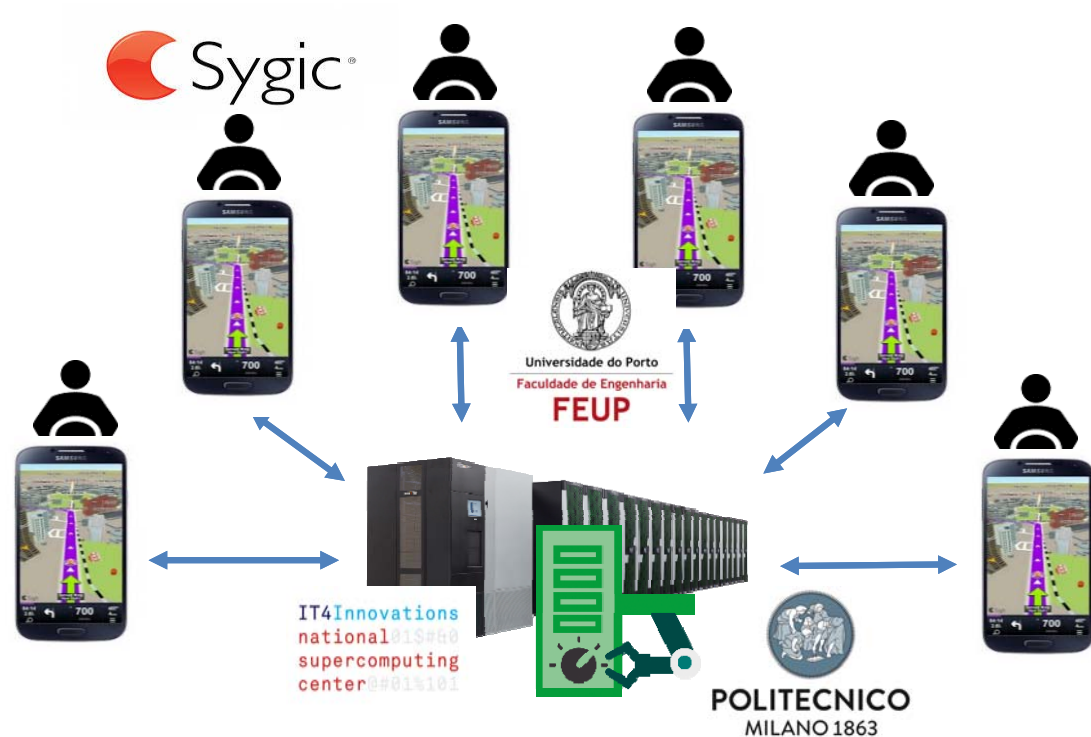
***Estimated Exascale Run in 2021: from 84 h to less than 1 h***

**Experiment website:**

<https://www.antarex4zika.eu>

# ANTAREX4ZIKA Video

<https://www.exscalate.eu/en/projects.html#Antarex4Zika>



**ANTAR**EX<sup>10<sup>18</sup></sup>



European  
Commission

Horizon 2020  
European Union funding  
for Research & Innovation

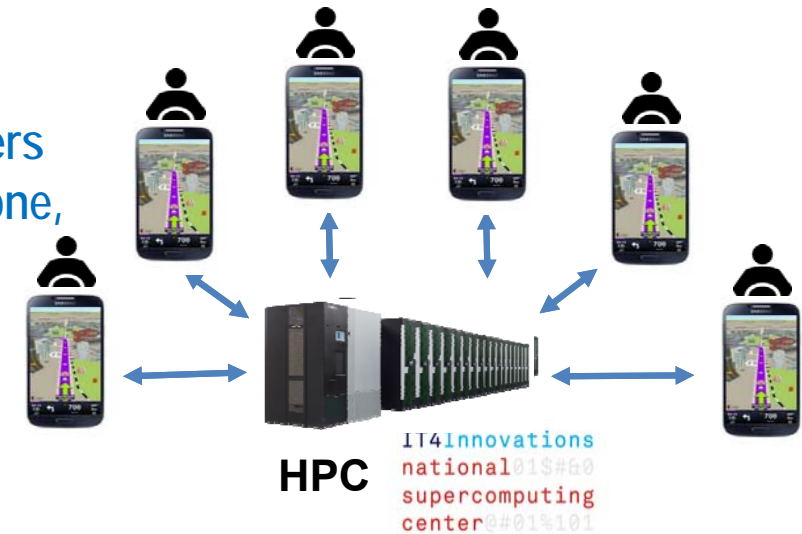
# Autotuning an HPC-based Navigation System for Smart Cities

# Self-adaptive Navigation System



- ✓ Sygic Top #2 App in navigation category worldwide with 200 M users
- ✓ Sygic world's 1st for iPhone, 2nd for Android

Sygic Company develops world's most popular navigation application & provides professional navigation software for business solutions



Exploit synergies between client-side and server-side:

- Many drivers – many routing requests to HPC system
- Traffic status data sources
- Continuous update of traffic flow calculation
- **Smart City Challenge**



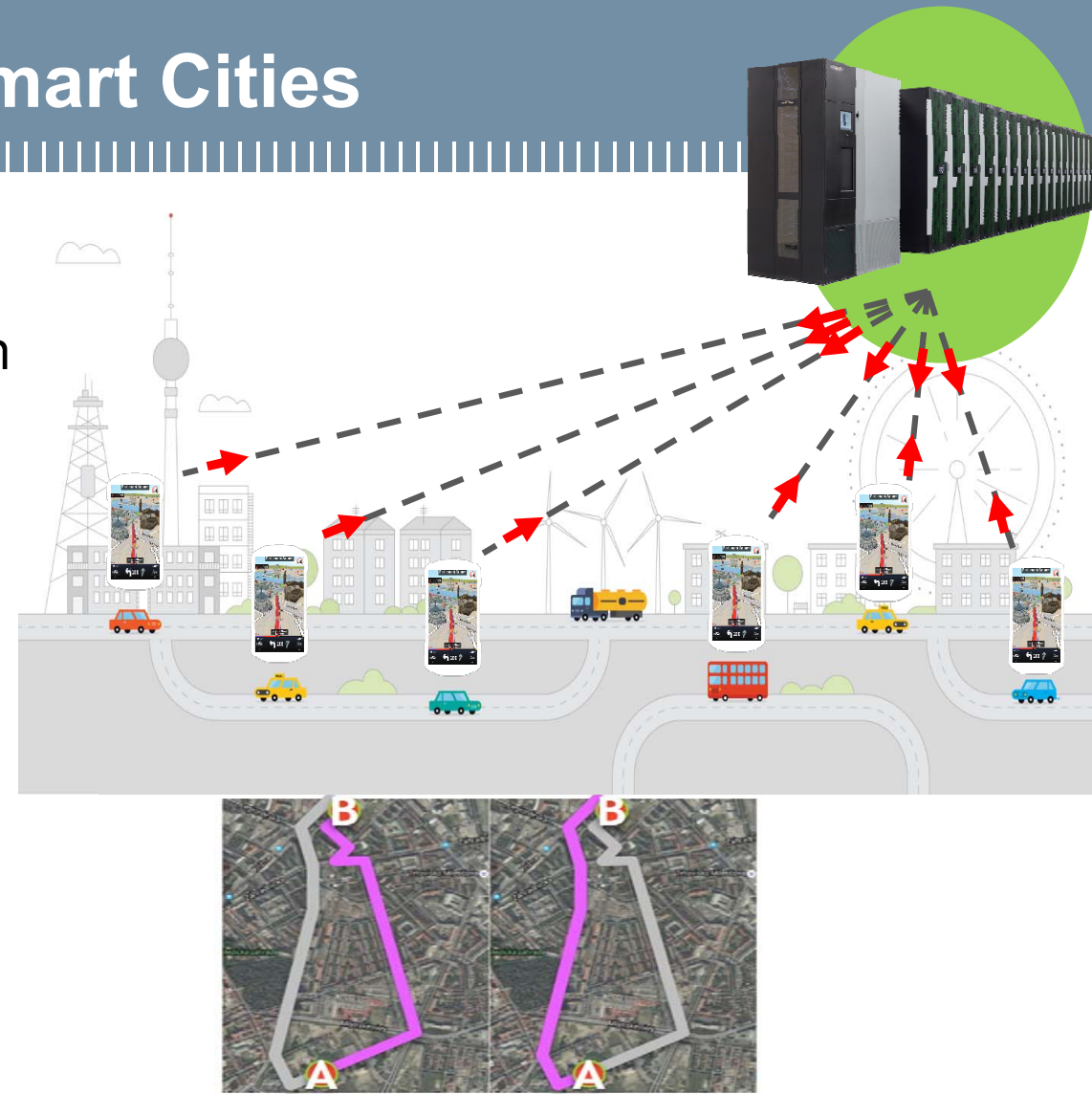
# Intelligent Navigation for Smart Cities

## Motivations:

- Provide optimal routes to hundred thousands of drivers/cars operating in the city area
- Serve all drivers' requests with global best to reduce total driving time
- Avoid traffic jams

## Requires:

- Intelligent routing based on accurate calculation of traffic view state
- Balance routes for a city global optimum
- Minimize data transfer

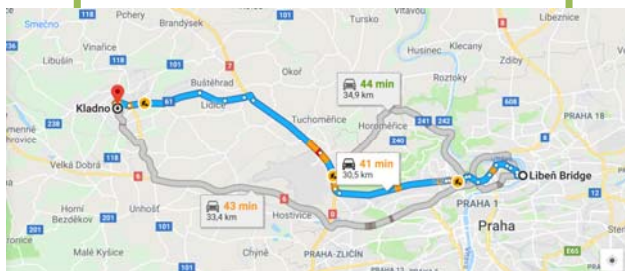


# Server-side Routing Pipeline

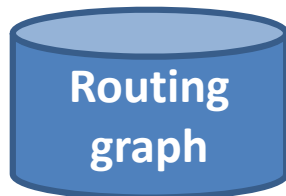
## INPUT



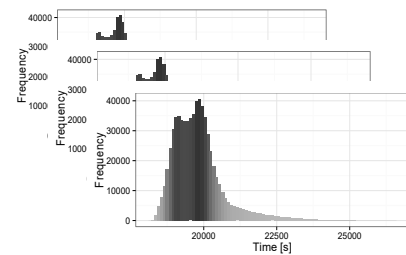
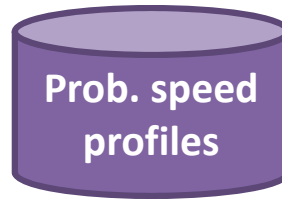
- ✓ GPS coordinates
- ✓ Origin – Destination
- ✓ Time of departure



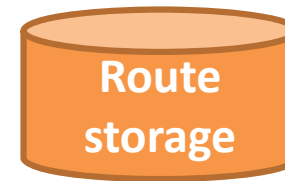
## First stage



## Second stage



## Third stage



Weight	Routes	Order
$w_1$	Route 1	2
$w_2$	Route 2	4
$w_3$	Route 3	1
$w_4$	Route 4	3

## OUTPUT

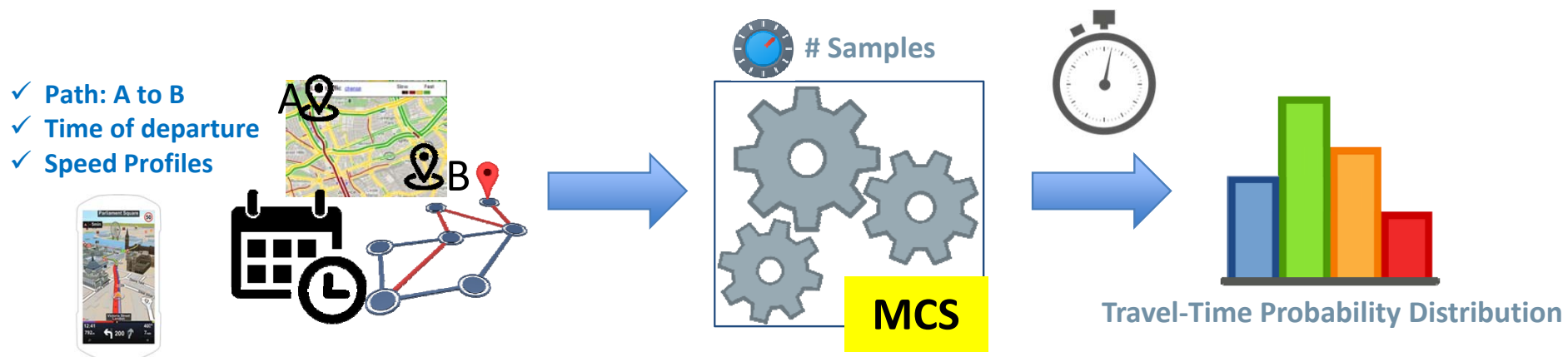


Route

# Probabilistic Time-dependent Routing

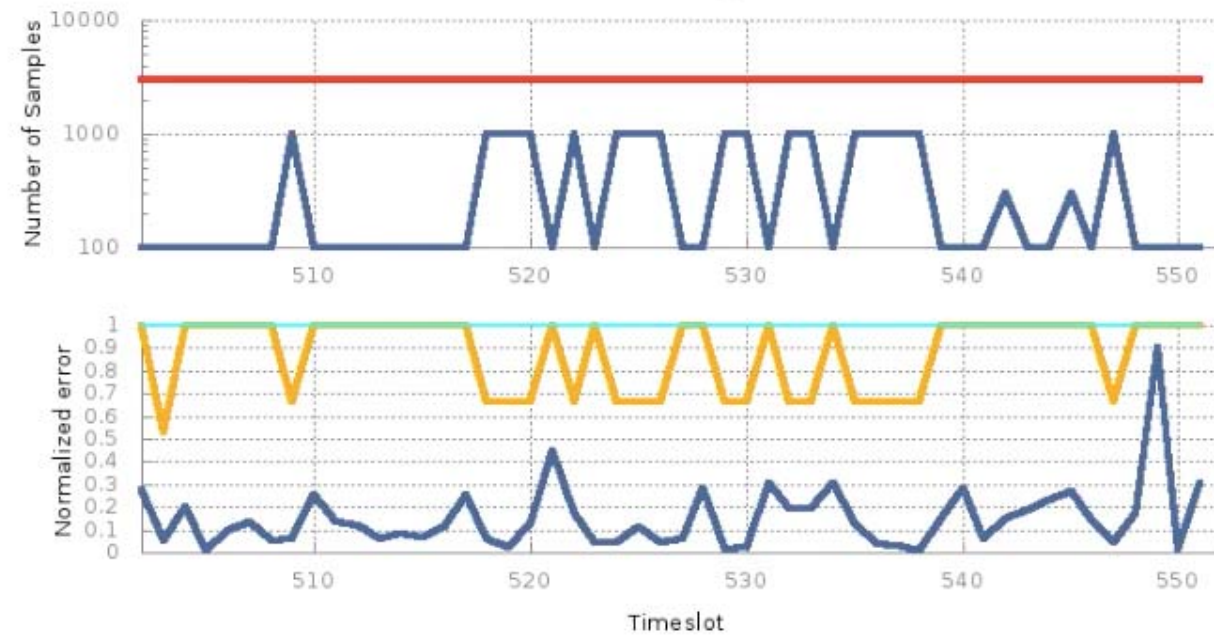
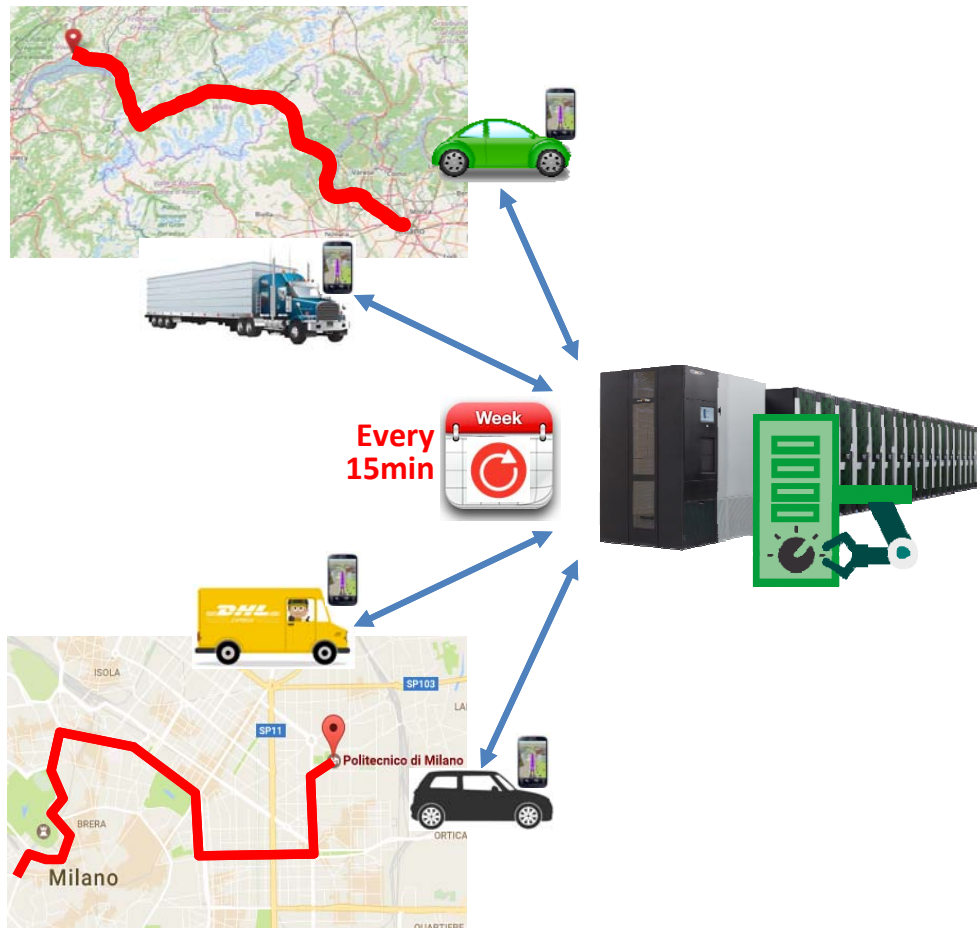
*What is the Probabilistic Time-dependent Routing for a navigation system?*

- ✓ Module to evaluate the *expected travel time*
- ✓ In a client-server navigation infrastructure, the *server-side* must evaluate accurate expected travel time with updated traffic information
- ✓ Implemented by a *MonteCarlo Simulation (MCS)* to evaluate *the probabilistic speed profile* for each hop
- ✓ *Dynamic autotuning* of the number of samples for the MCS



E. Vitali, D. Gadioli, G. Palermo, M. Golasowski, J. Bispo, P. Pinto, J. Martinovic, K. Slaninova, J. Cardoso, C. Silvano, "An Efficient Monte Carlo-based Probabilistic Time-Dependent Routing Calculation Targeting a Server-Side Car Navigation System", Minor revision in IEEE Trans. on Emerging Topics in Computing; Open Access: <http://arxiv.org/abs/1901.06210>

# Example: Same user request during the week



**Dynamic approach uses the 30% of the number samples with respect to static approach**



Best Early Stage Innovation

## ✓ **ICT 2018 Vienna Award**

### ✓ *Top 20 Innovations in 2018*

*“Saves money to drivers and cities.*

*Contributes to reduction of CO2 emissions.*

*Improves quality of life in urban areas.*

*Reduces time spent in daily travel traffic by more than 20 percent”*

Im Bereich Innovation Radar-Preis (2018) ernannt. Der Fachausschuss auf der ICT anerkannte die positive Auswirkung auf Städten.



Projektleiter Radim Cmar von Sygic bei der Präsentation der Navigationslösung. (c) Sygic

Die inter...  
die Lei...  
Superc...  
optima...  
in der...  
der täg...  
ist aucl...  
andere...  
das Co...  
den Da...  
wie bei

ANTAREX<sup>10<sup>th</sup></sup>



<http://www.antarex-project.eu/>