A feedback-control approach for resource management in public clouds

Domenico Grimaldi, Valerio Persico, Antonio Pescapè, Alessandro Salvi, Stefania Santini

University of Napoli "Federico II" (Italy)

IEEE GLOBECOM 2015

San Diego, CA, (USA) December 7 2015

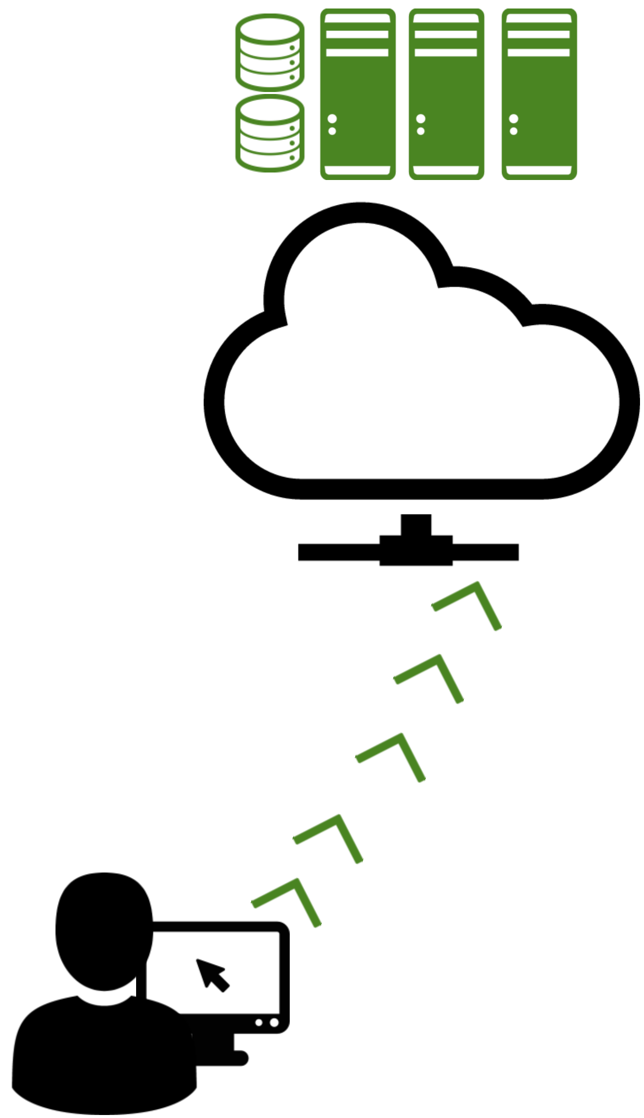
(Public) Cloud Paradigm: XaaS

Industry and market increasingly depend on cloud-based infrastructures

- No upfront investments
- Real-time provisioning
- *Pay-as-you-go* resources

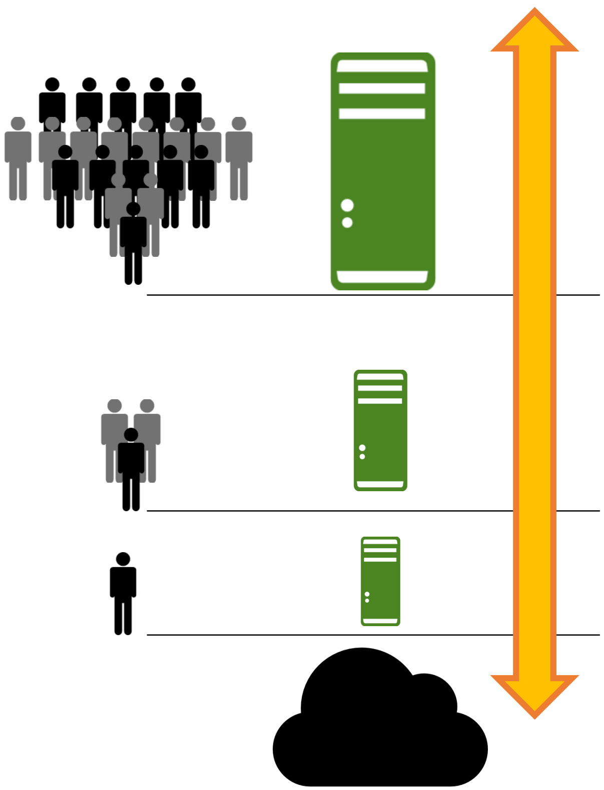
Control and management of the resources are of the utmost importance

- Elasticity allows cloud customers to acquire and release resources dynamically
- Applications may face large fluctuating loads
- Deciding the right amount of resources is not an easy task

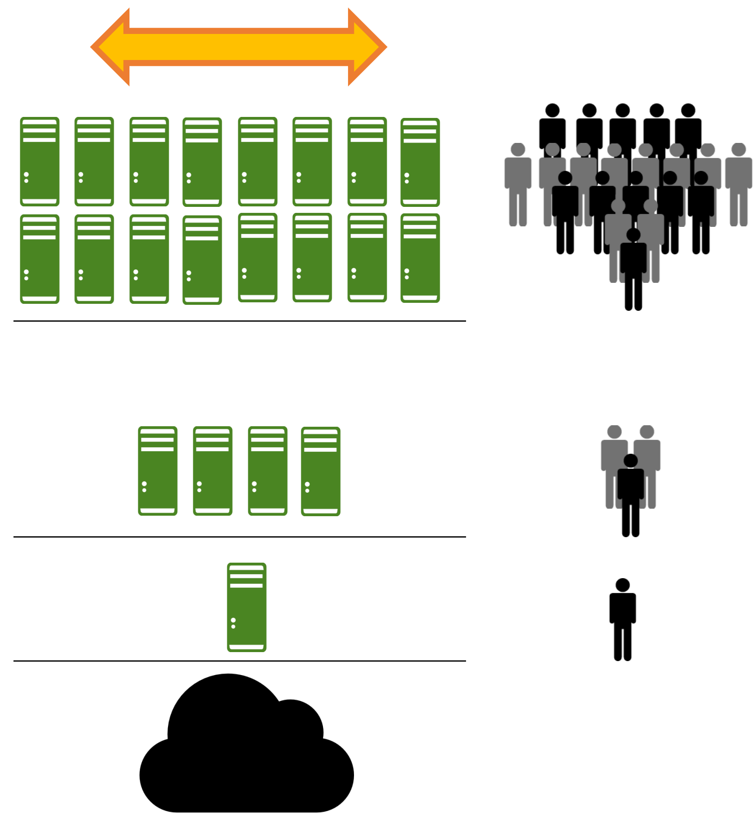


Resource Elasticity and Scaling

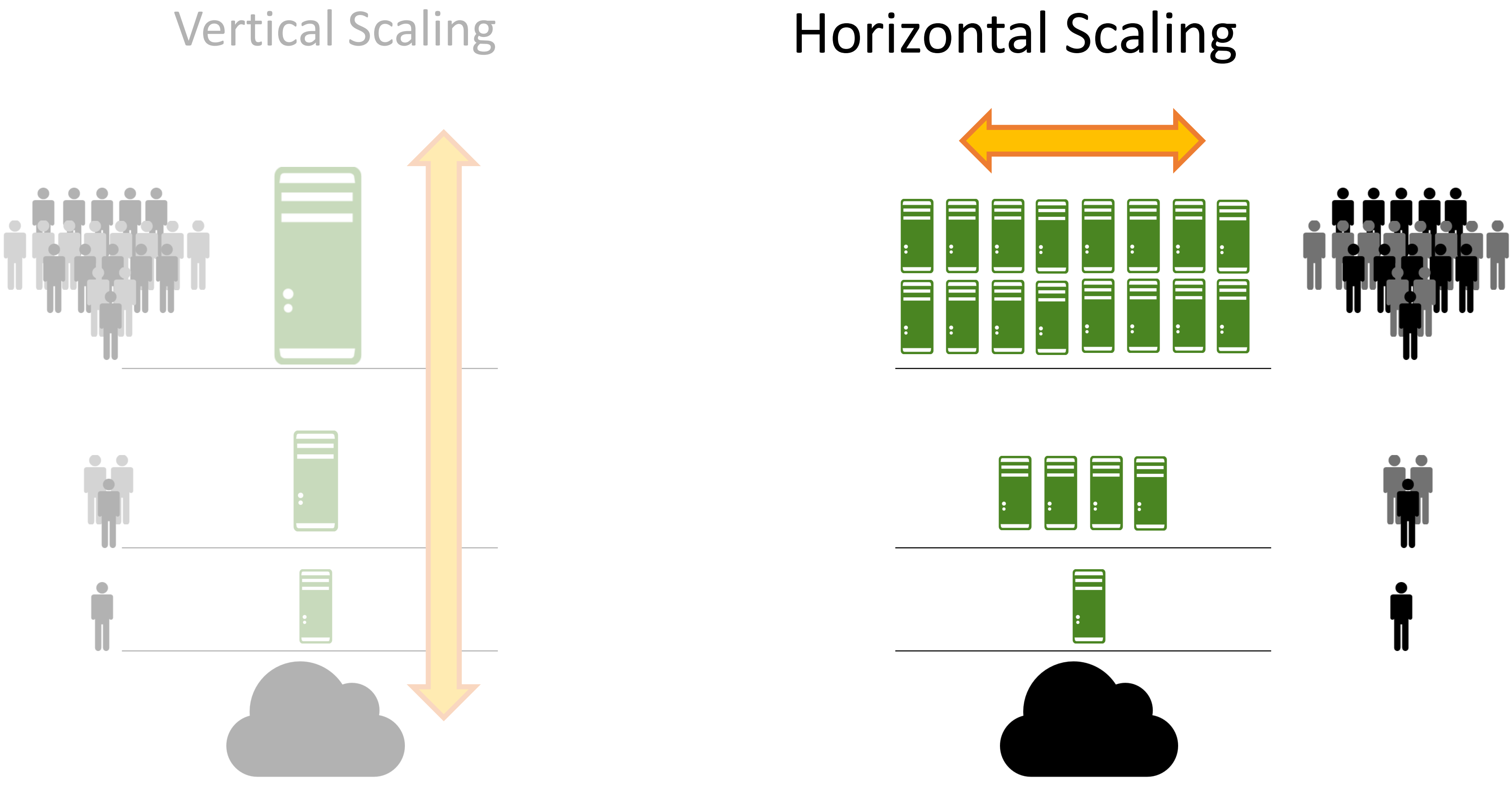
Vertical Scaling



Horizontal Scaling

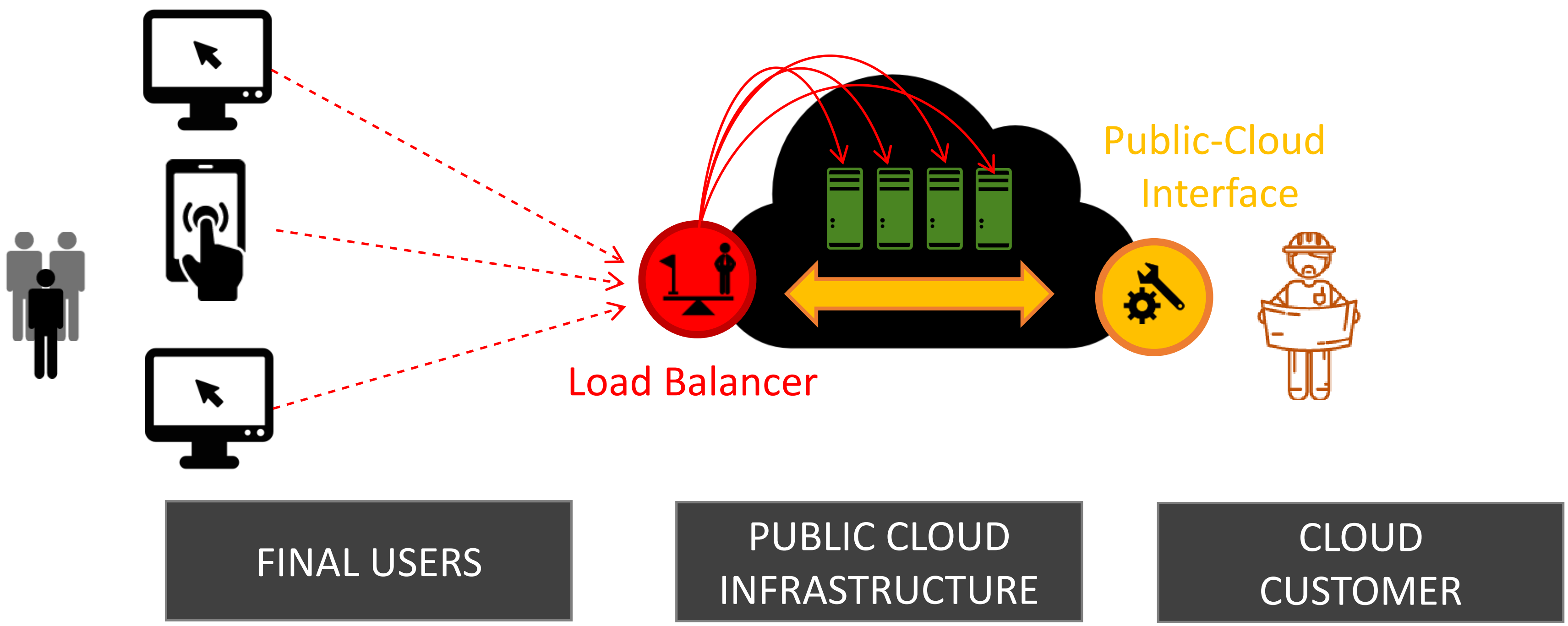


Resource Elasticity and Scaling



Problem Statement

Properly dimensioning a set of resources allocated to an application to guarantee a desired performance



An automatic feedback-control strategy to scale cloud resources

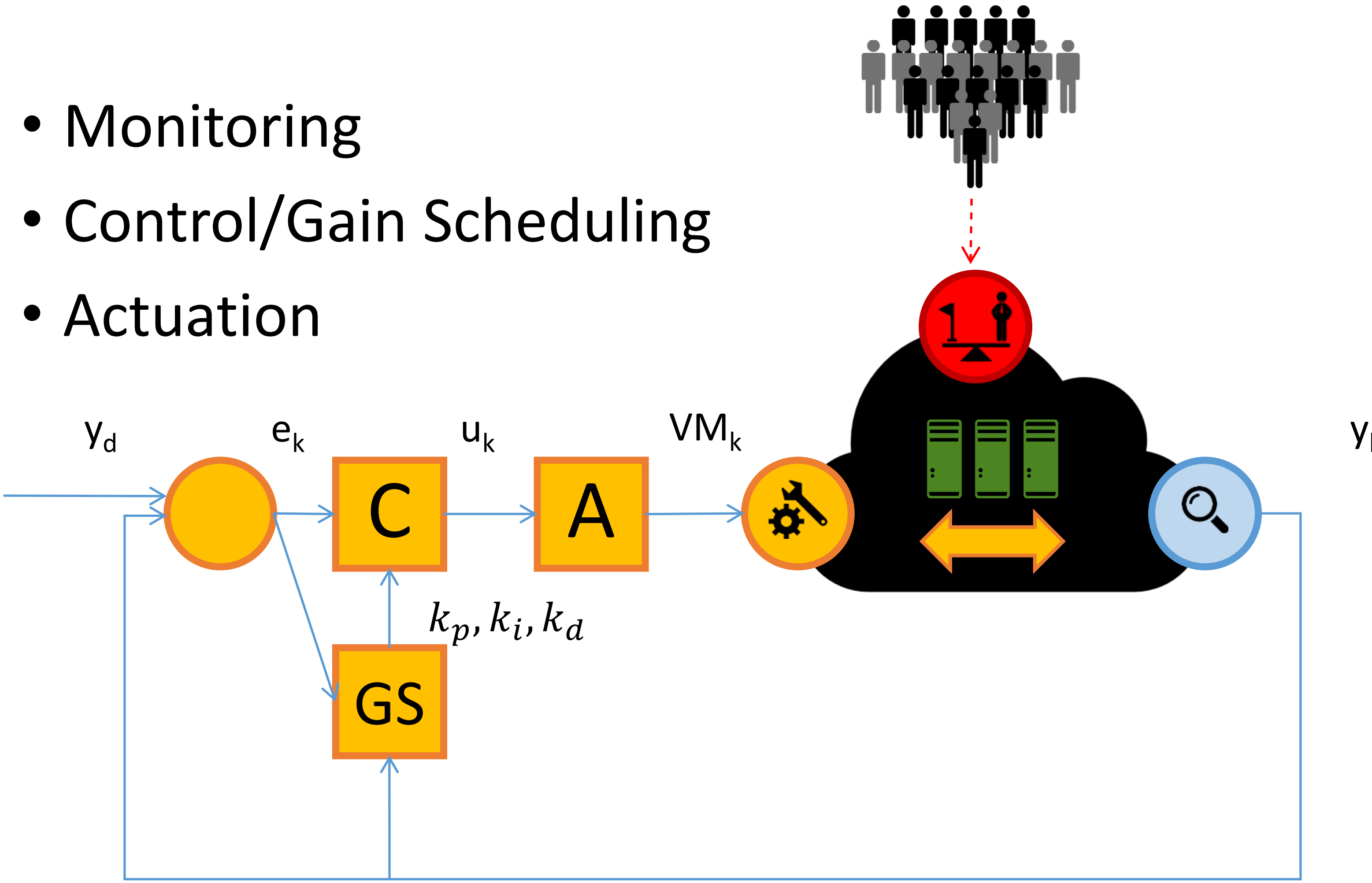
- **Goal**
  - Guaranteeing a pre-specified Service Level (SL)

- **Resources**
  - Virtual Machines (VMs)---*IaaS* model

- **Control Strategy**
  - No previsional model of the system behavior needed
  - Tailored for public clouds

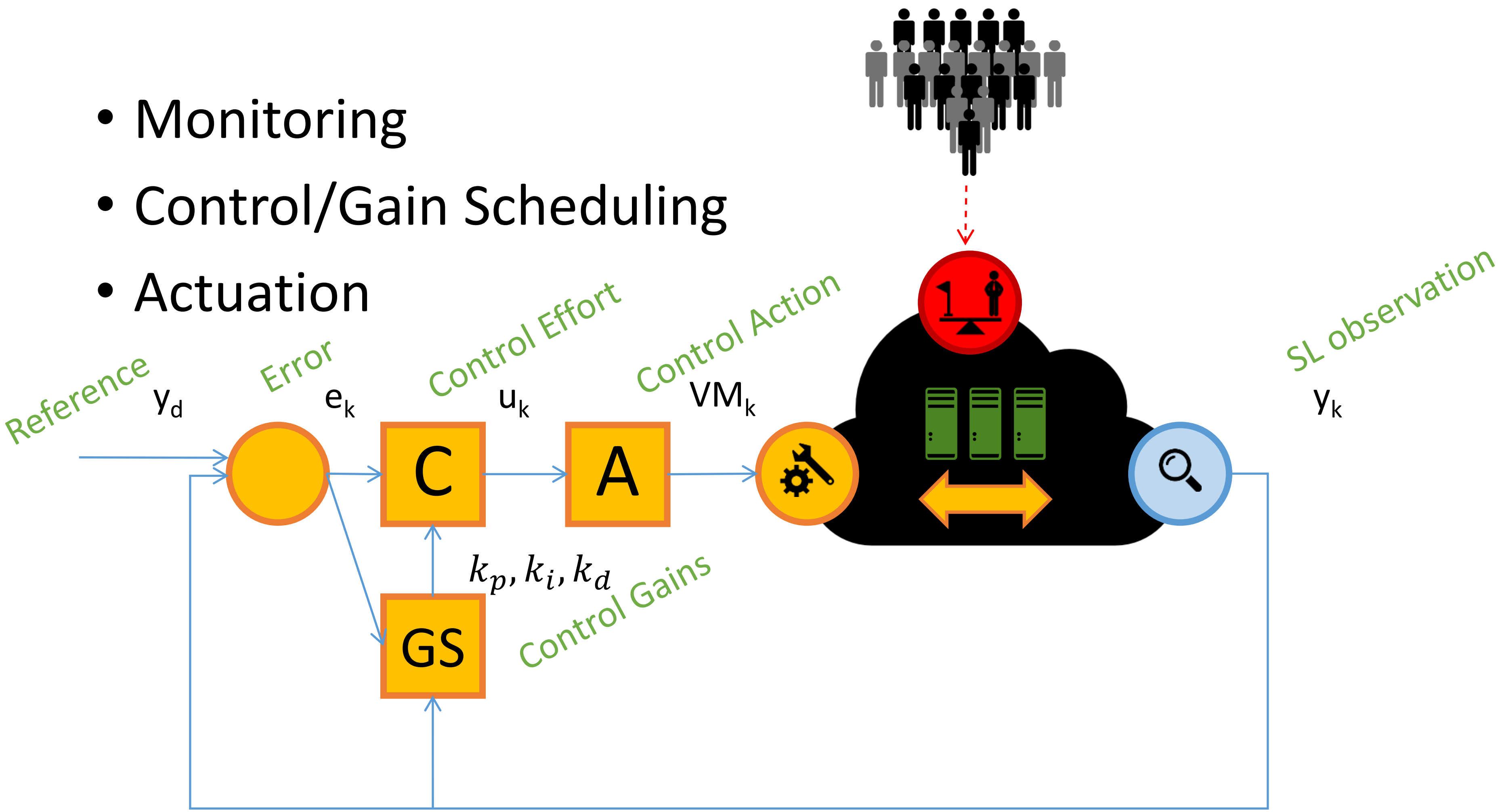
Control Loop

- Monitoring
- Control/Gain Scheduling
- Actuation



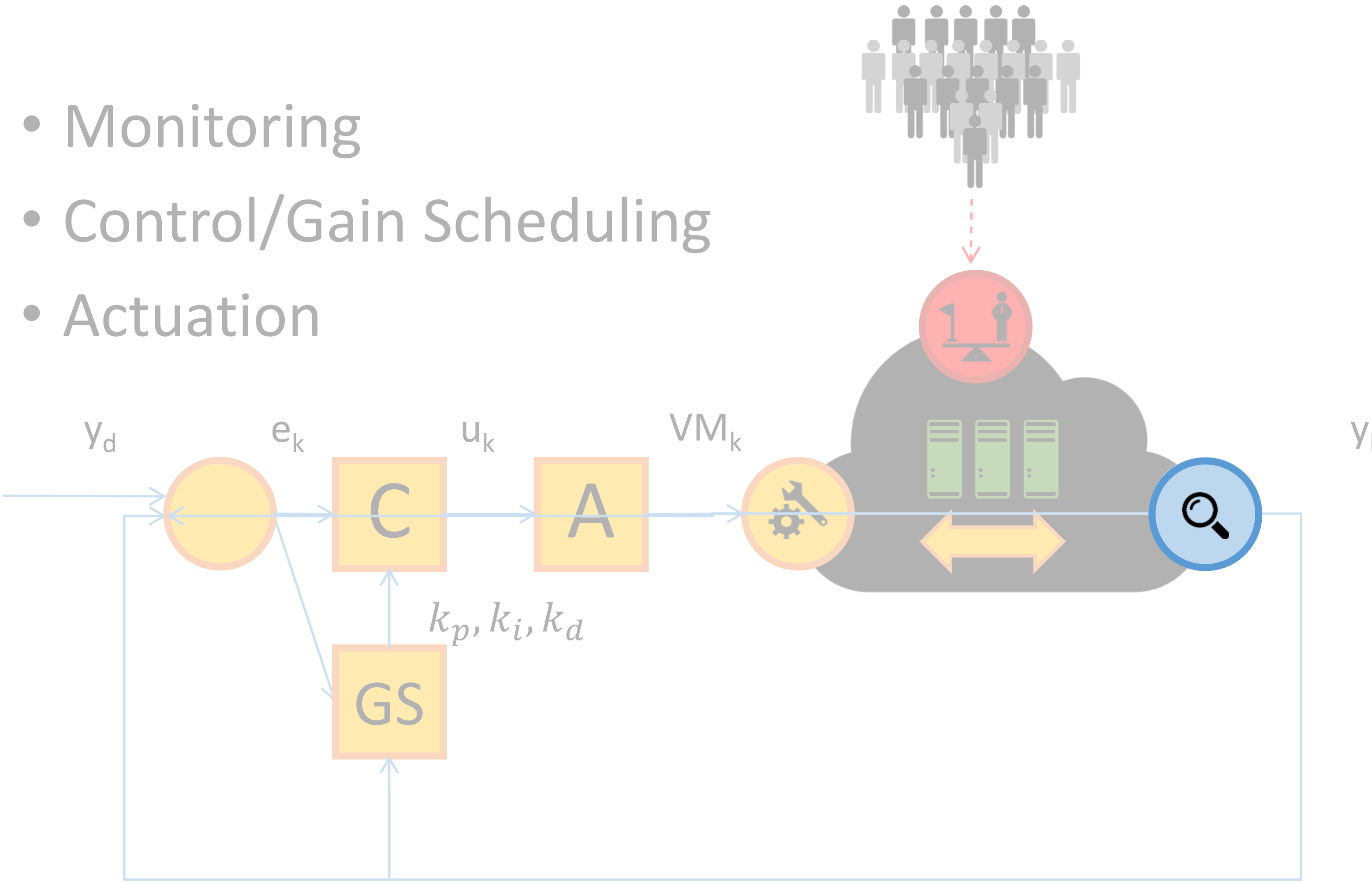
Control Loop

- Monitoring
- Control/Gain Scheduling
- Actuation

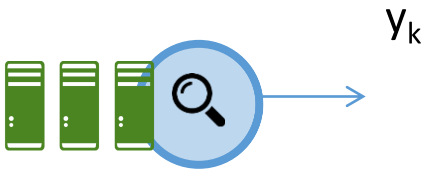


Monitoring Block

- Monitoring
- Control/Gain Scheduling
- Actuation



Monitoring Block



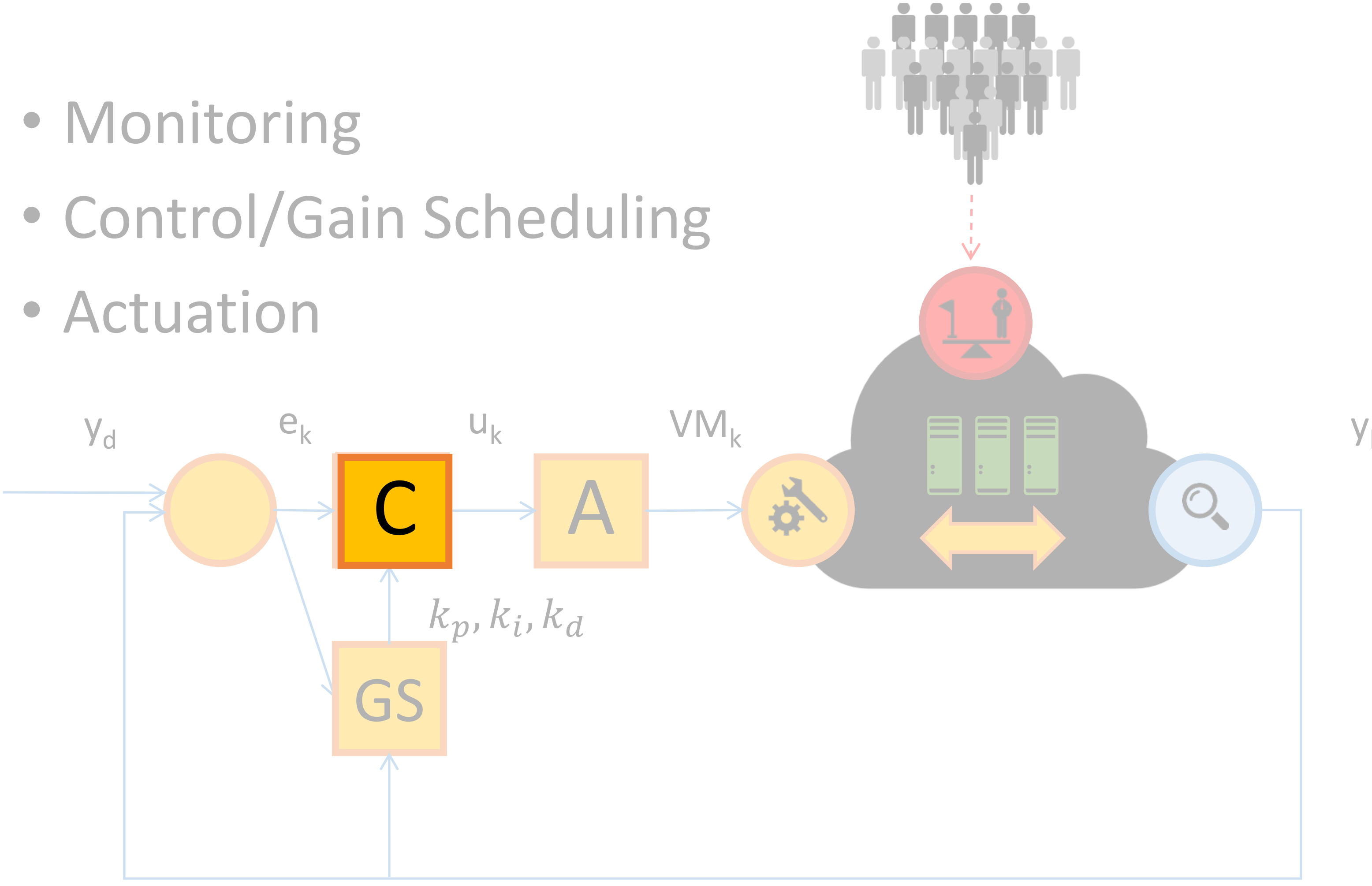
- Metric observed: CPU load
  - Impacts task-completion time and latency

$$y_k = \frac{CPU_{VM1}(k) + \ldots + CPU_{VMN}(k)}{\#VM(k)}$$

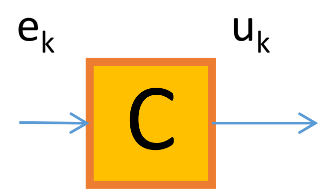
$\#VM(k)$: active virtual machines at time interval k
$CPU_{VMi}(k)$: CPU load of the ith VM

Control Block

- Monitoring
- Control/Gain Scheduling
- Actuation



Control Block



- Control strategy: PID
  Proportional-Integrative-Derivative

$$
\begin{aligned}
u_k = \quad & u_{k-1} + \frac{T_d(e_k,\Delta e_k)}{\Delta t} e_{k-2} \\
& + k_p(e_k,\Delta e_k)\left[-1 - \frac{2T_d(e_k,\Delta e_k)}{\Delta t}\right] e_{k-1} \\
& + k_p(e_k,\Delta e_k)\left[\left(1 + \frac{\Delta t}{T_i(e_k,\Delta e_k)} + \frac{T_d(e_k,\Delta e_k)}{\Delta t}\right) e_k\right]
\end{aligned}
$$

with

$$T_i(e_k,\Delta e_k) = \frac{k_p(e_k,\Delta e_k)}{k_i(e_k,\Delta e_k)},$$

$$T_d(e_k,\Delta e_k) = \frac{k_d(e_k,\Delta e_k)}{k_p(e_k,\Delta e_k)},$$

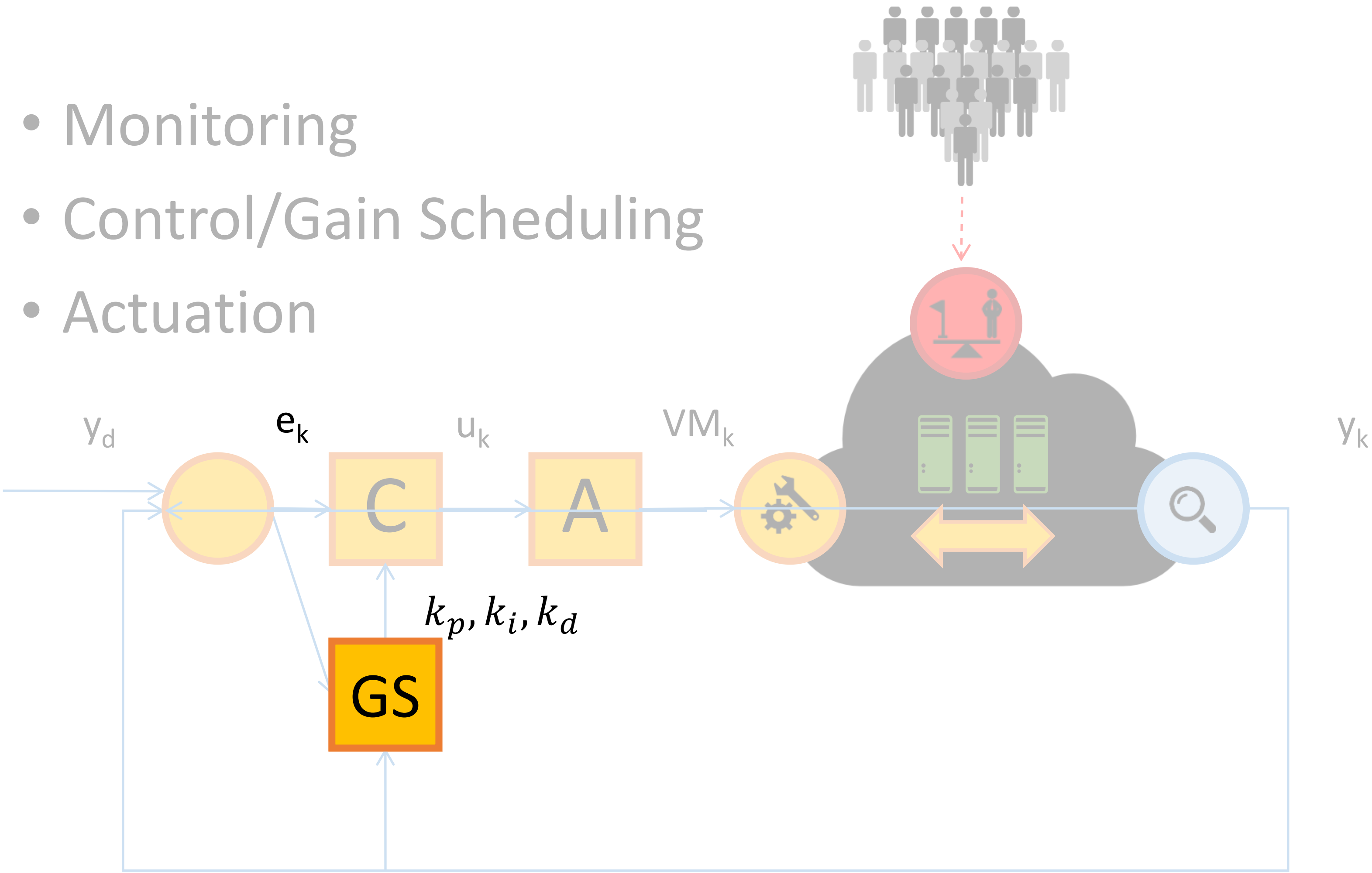
Evolution of e_k

- Present
- Past
- “Future”

- Closed loop dynamics depend on the choice of control gains

Gain Scheduling

- Monitoring
- Control/Gain Scheduling
- Actuation



Gain Scheduling

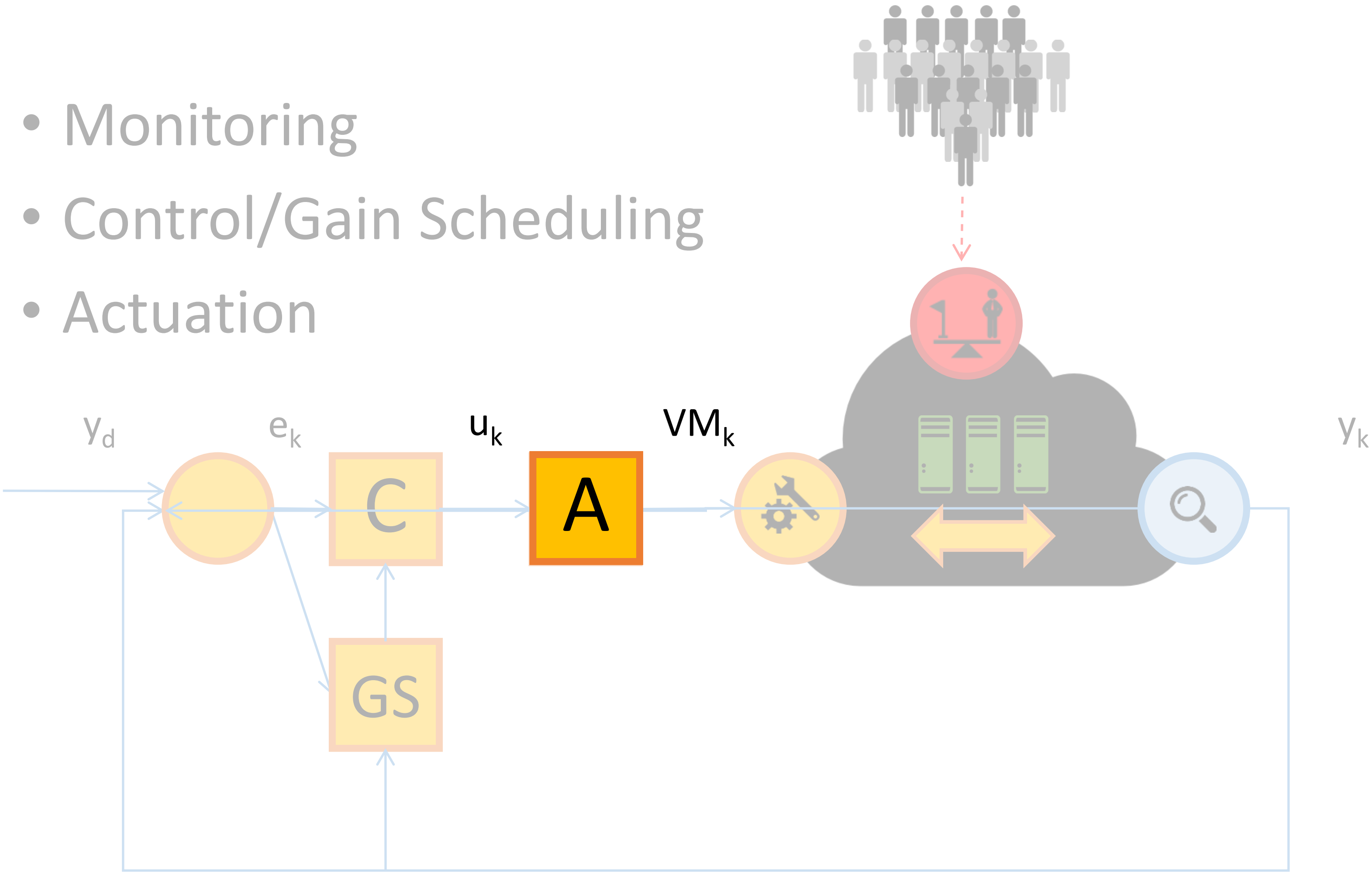
$$e_k \rightarrow \boxed{\text{GS}} \rightarrow k_p, k_i, k_d$$

- k_p, k_i, k_d are initialized according to known optimization procedures
- Experimental observations disclosed large variability that may depend on the actual conditions of the system
- To achieve further robustness, control parameters are adjusted on-line and in real time, according to the actual error dynamics

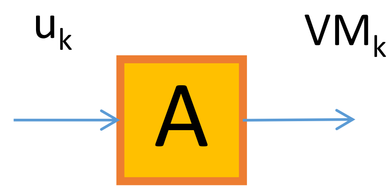
$$\boxed{\begin{array}{c} k_p(e_k, \Delta e_k) \\ k_i(e_k, \Delta e_k) \\ k_d(e_k, \Delta e_k) \end{array}}$$

Actuation Block

- Monitoring
- Control/Gain Scheduling
- Actuation



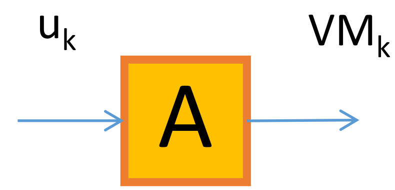
Actuation Block



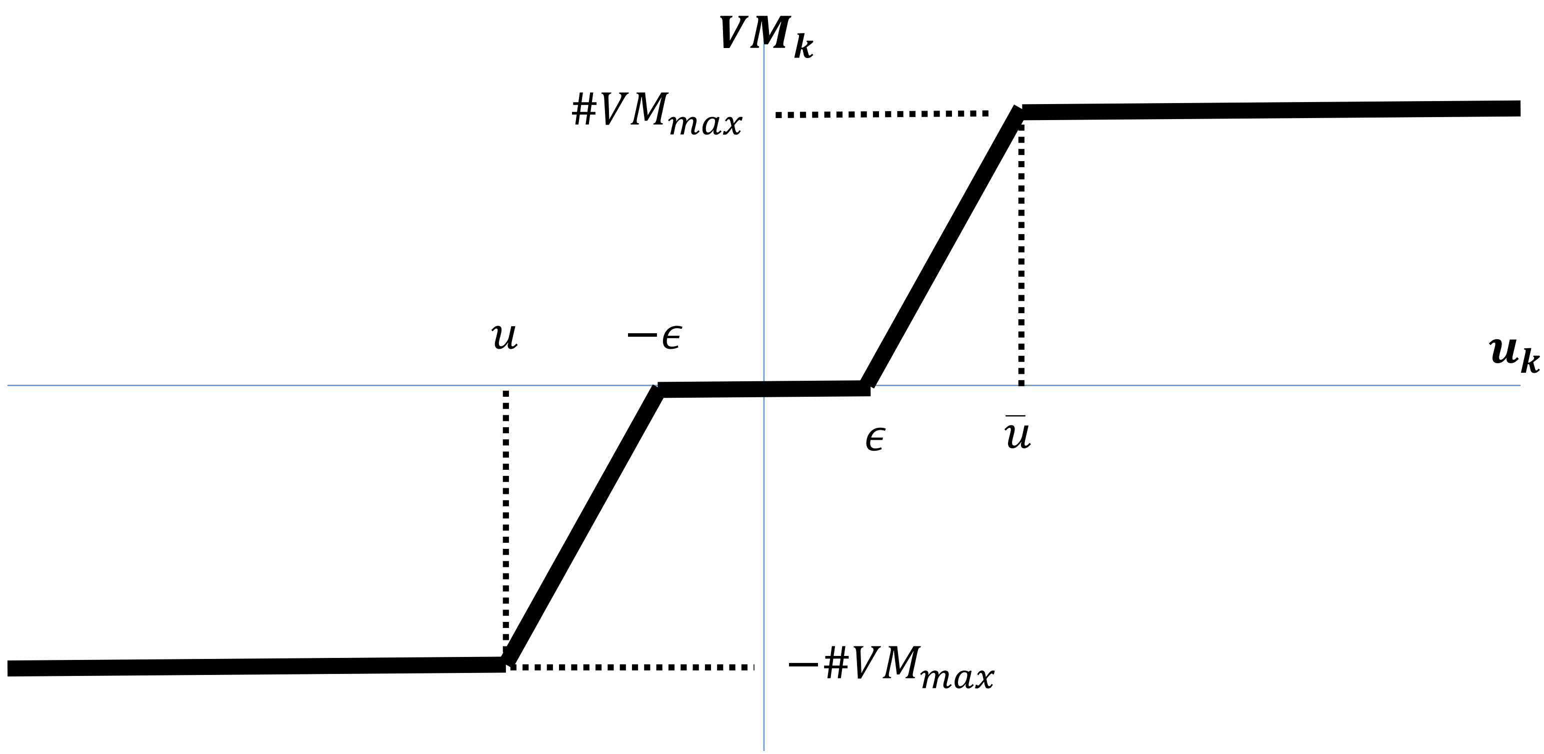
- The number of VMs activated or terminated at time interval k (VM_k) depends on the actual value of the control signal u_k

$$VM_k = \begin{cases} \#VM_{\max} & \textit{if} \quad \bar{u} \leq u_k \\ \alpha u_k - \beta & \textit{if} \quad \epsilon \leq u_k < \bar{u} \\ 0 & \textit{if} \quad -\epsilon < u_k < \epsilon \\ \alpha u_k + \beta & \textit{if} \quad -\underline{u} < u_k \leq -\epsilon \\ -\#VM_{\max} & \textit{if} \quad u_k \leq -\underline{u} \end{cases}$$

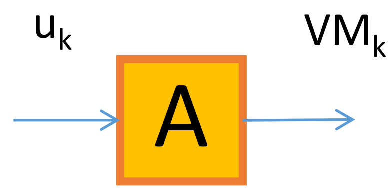
Actuation Block



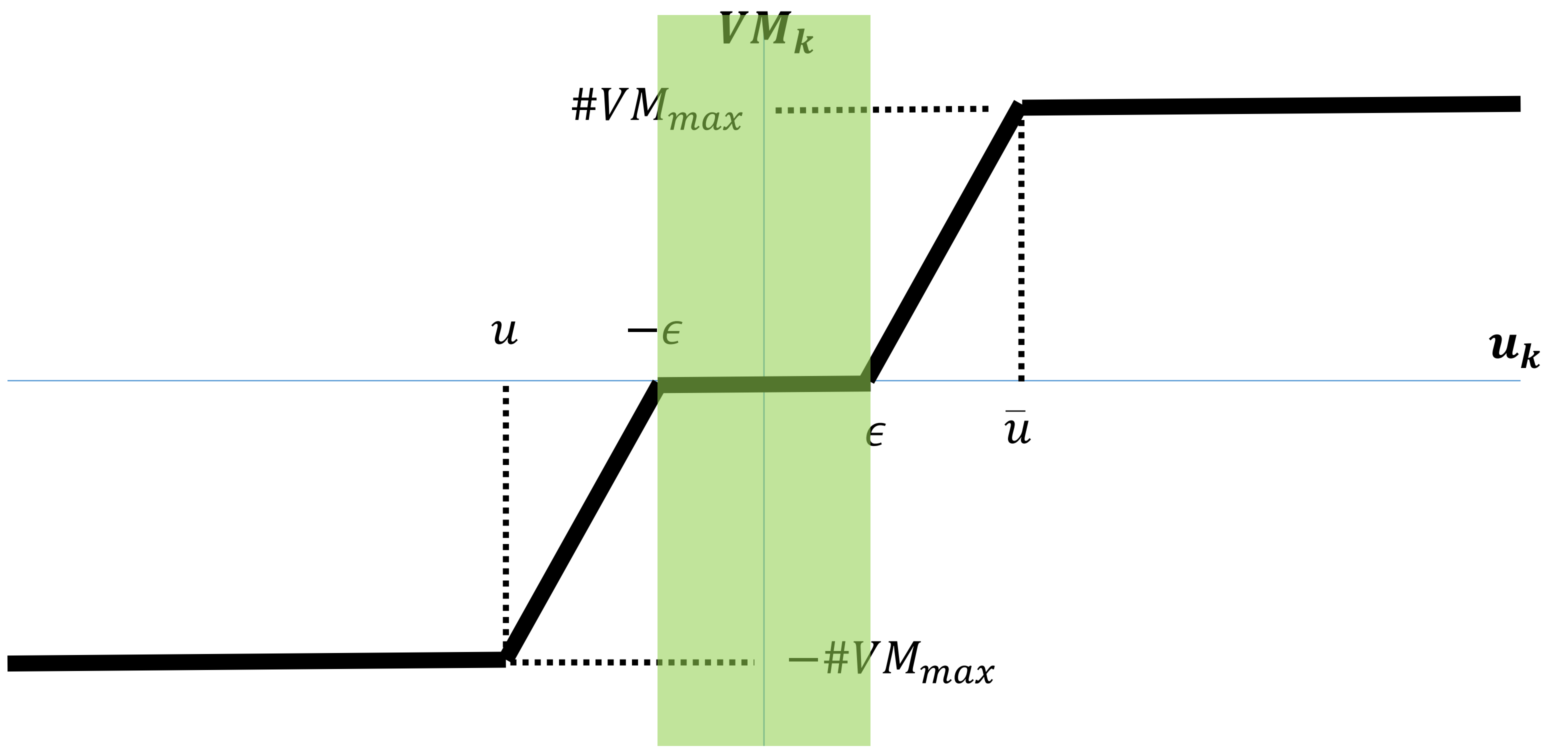
- The number of VMs activated or terminated at time interval k (VM_k) depends on the actual value of the control signal u_k



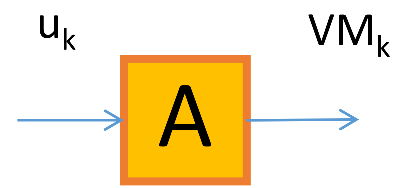
Actuation Block



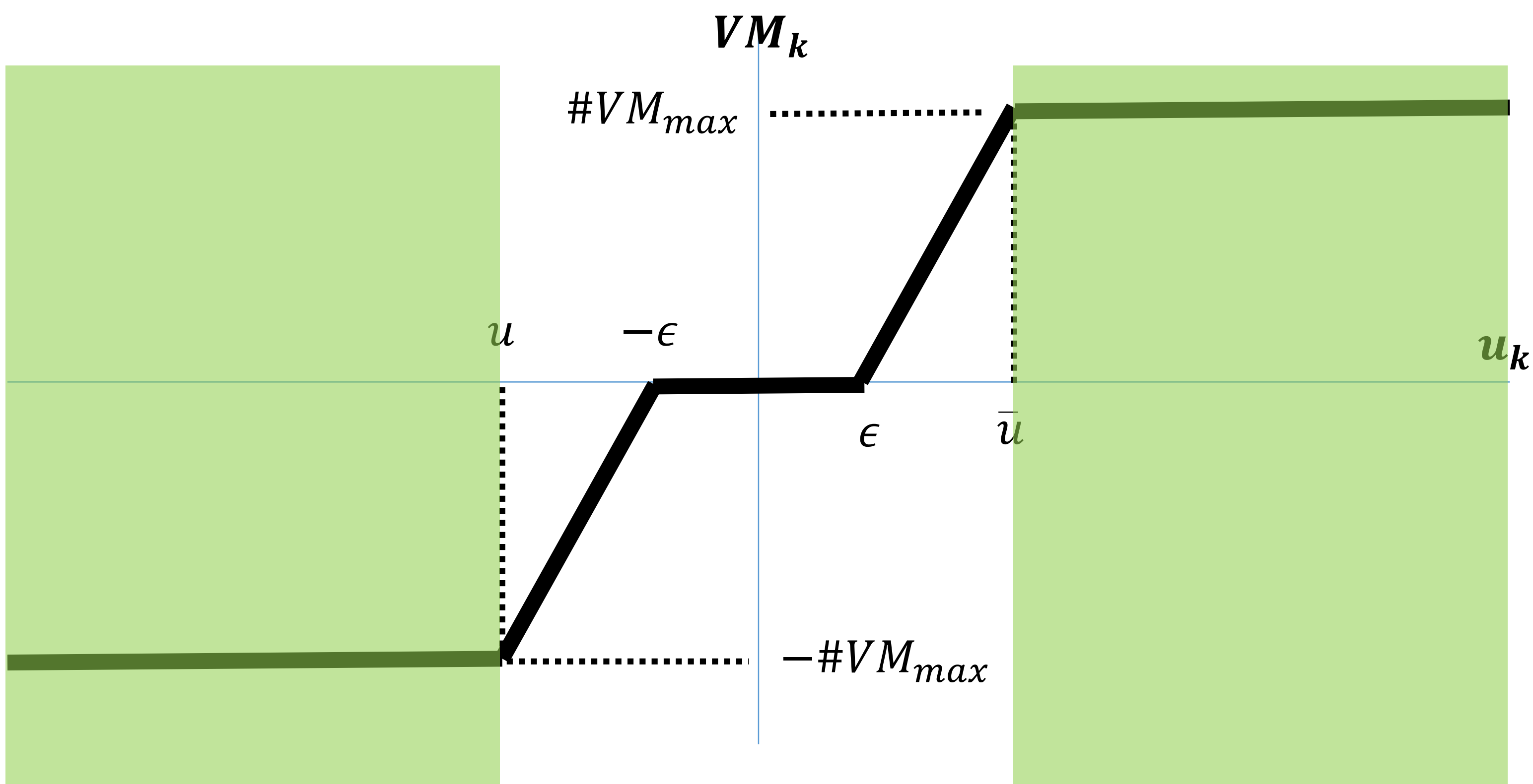
- The number of VMs activated or terminated at time interval k (VM_k) depends on the actual value of the control signal u_k



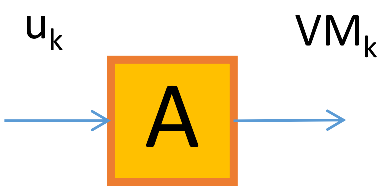
Actuation Block



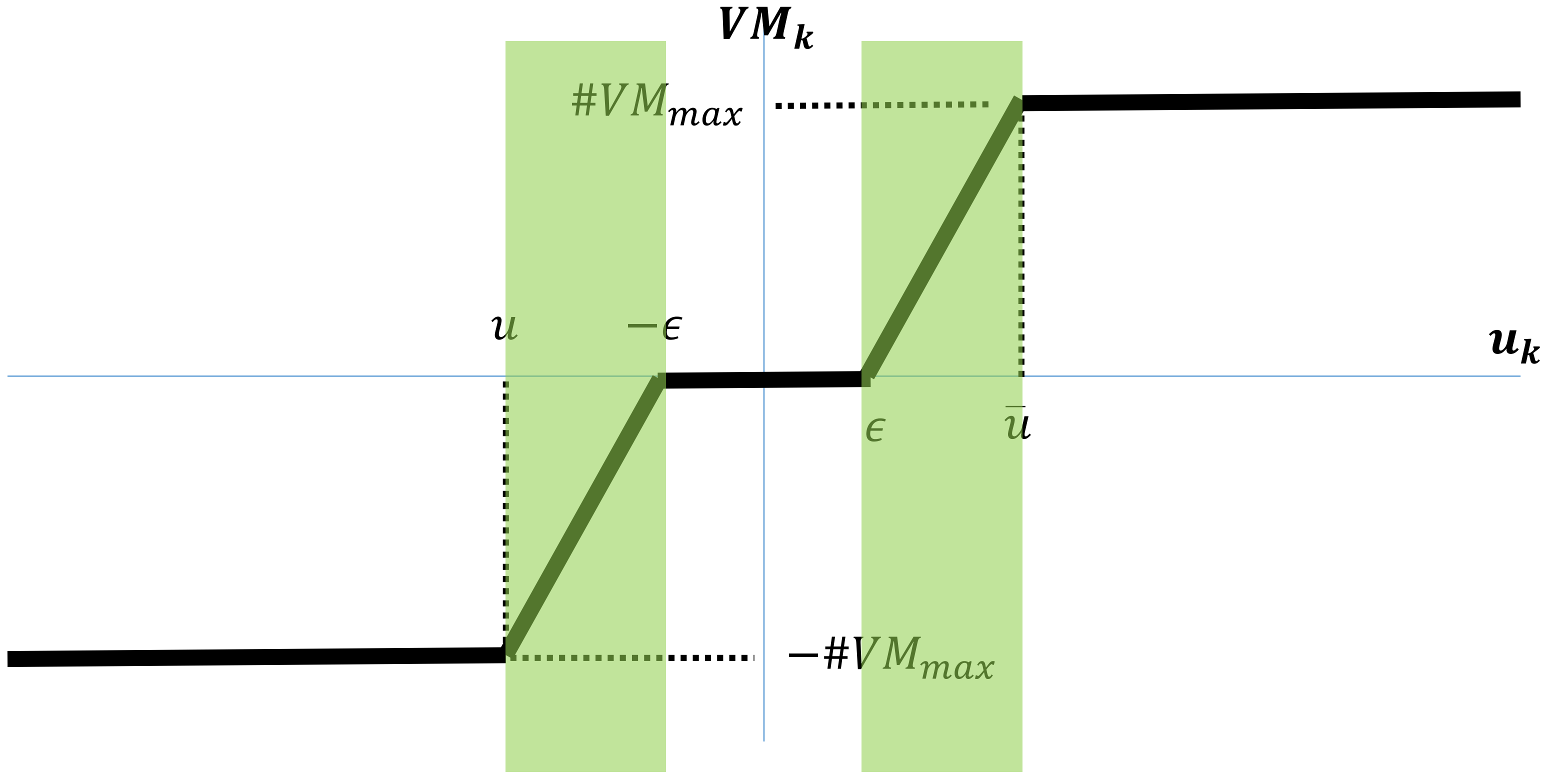
- The number of VMs activated or terminated at time interval k (VM_k) depends on the actual value of the control signal u_k



Actuation Block

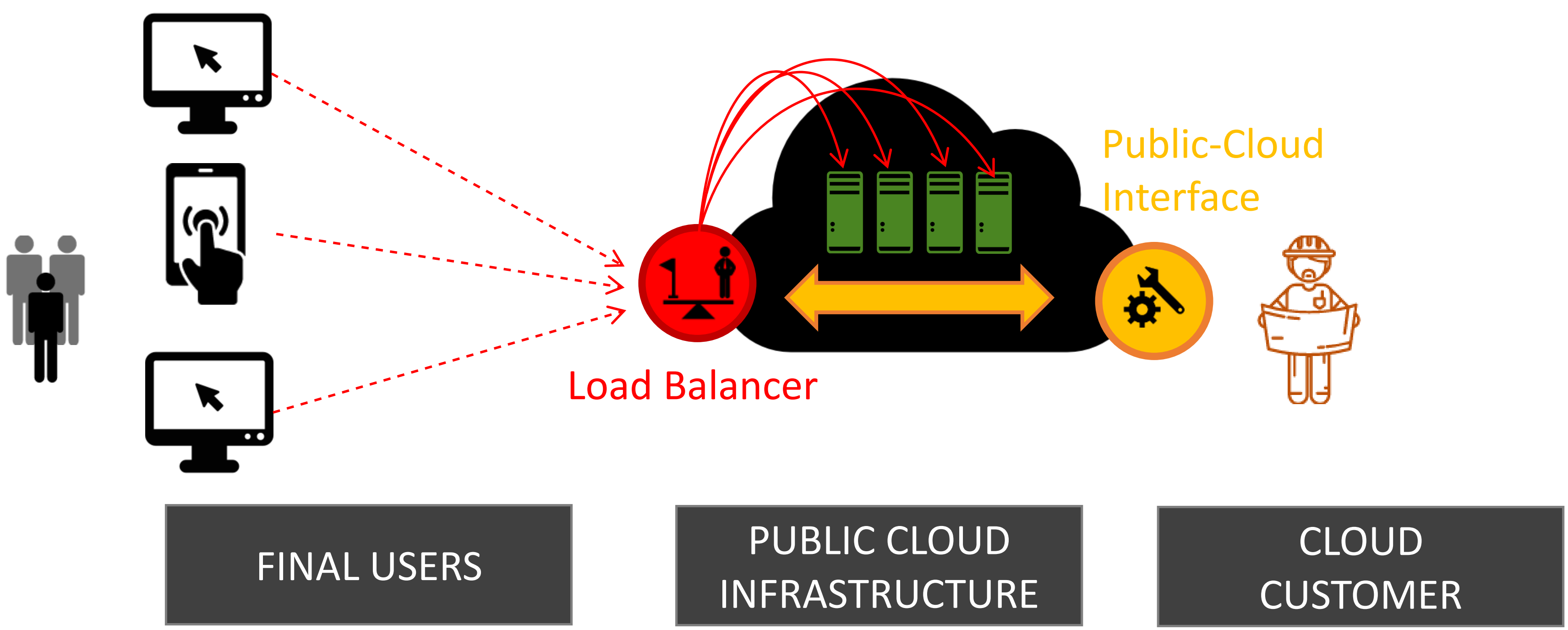


- The number of VMs activated or terminated at time interval k (VM_k) depends on the actual value of the control signal u_k



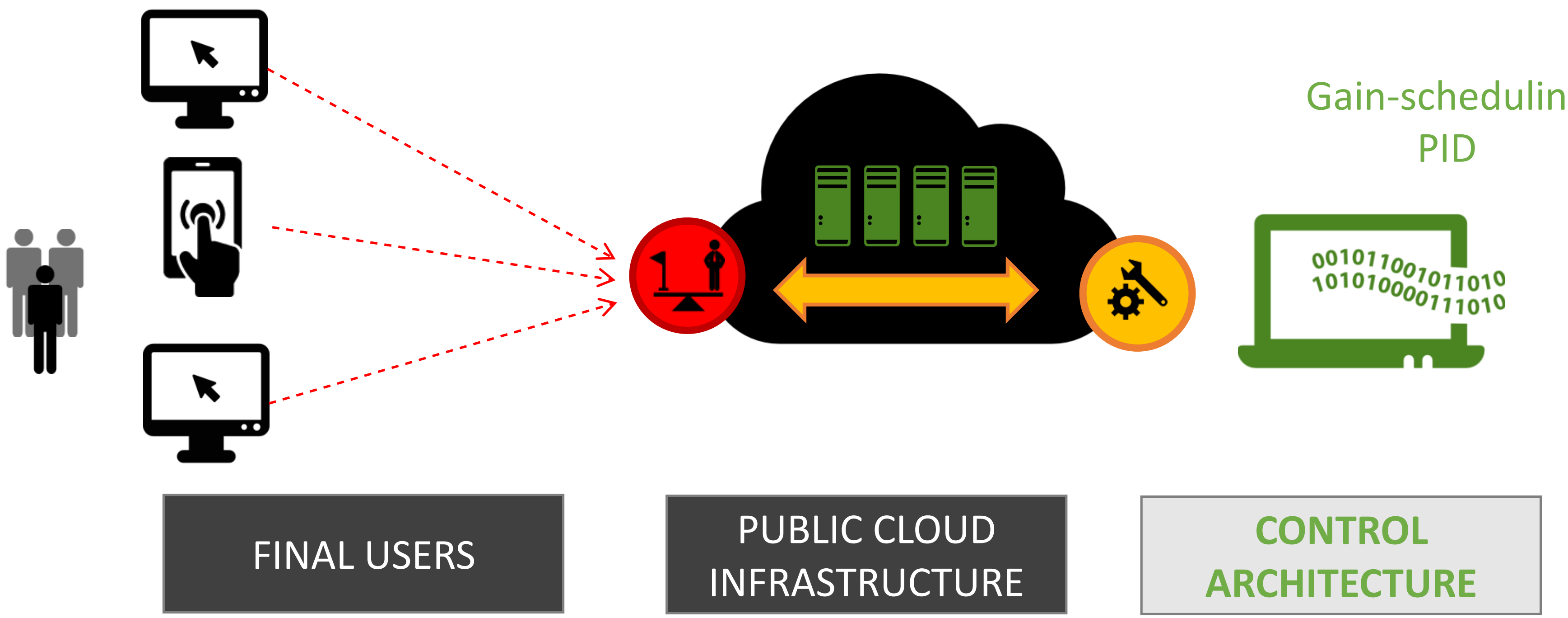
Experimental Setup*

*Code available at:
http://www.traffic.comics.unina.it/cloud



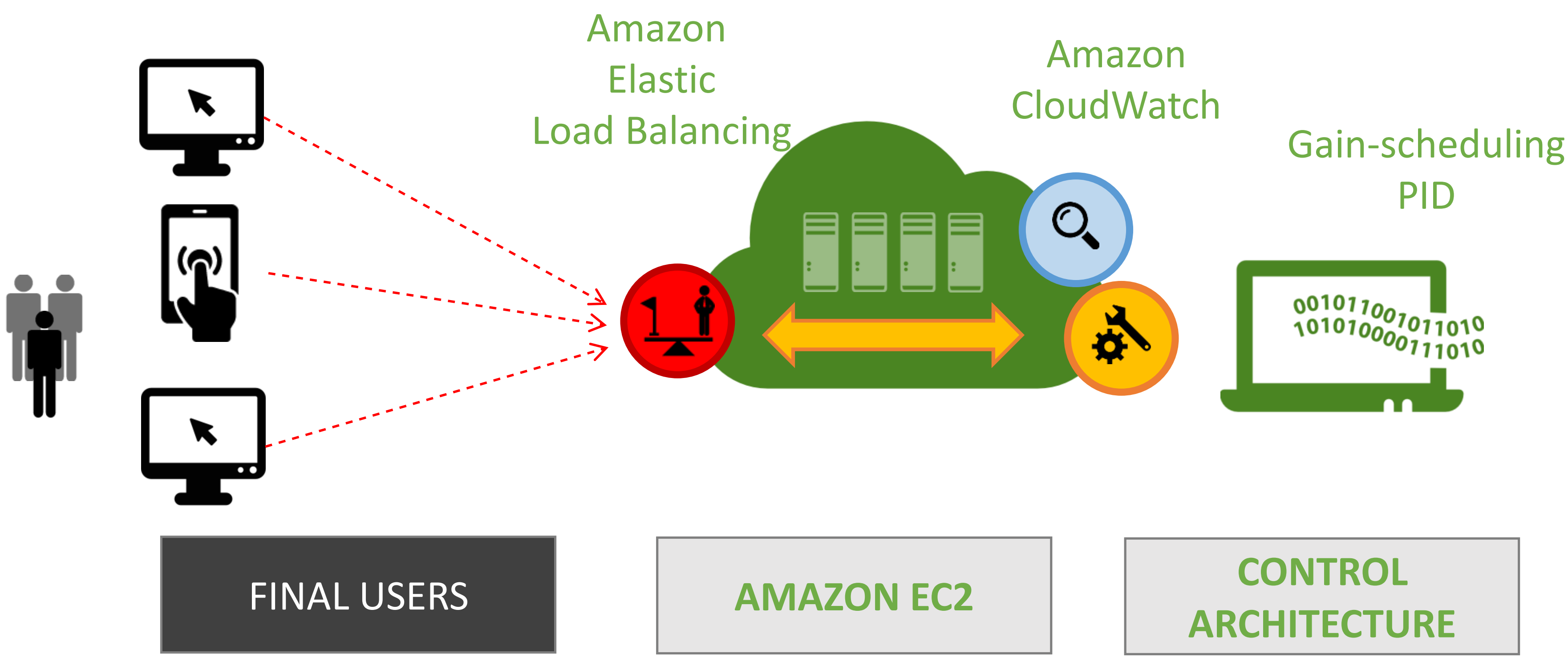
Experimental Setup*

*Code available at:
http://www.traffic.comics.unina.it/cloud



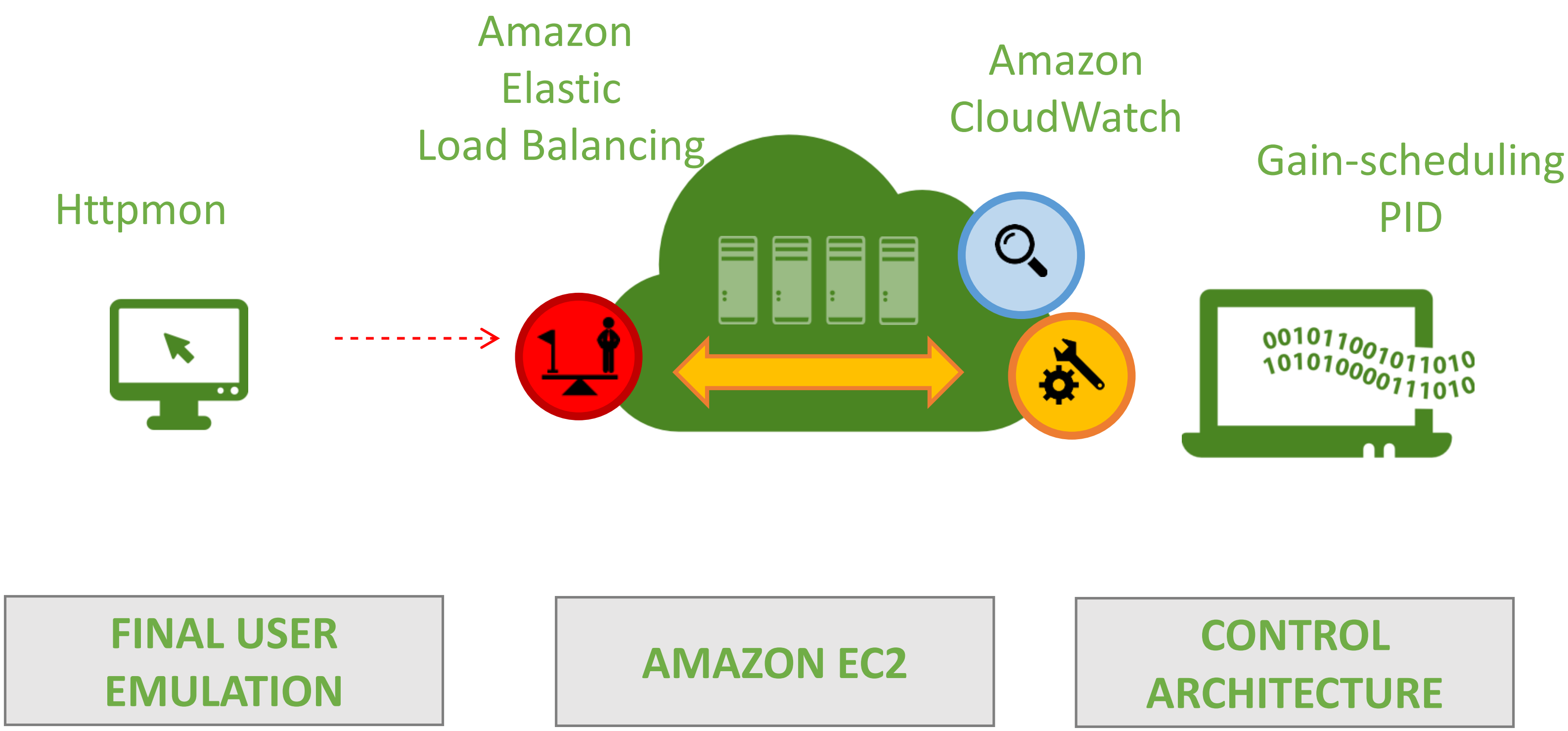
Experimental Setup*

*Code available at:
http://www.traffic.comics.unina.it/cloud



Experimental Setup*

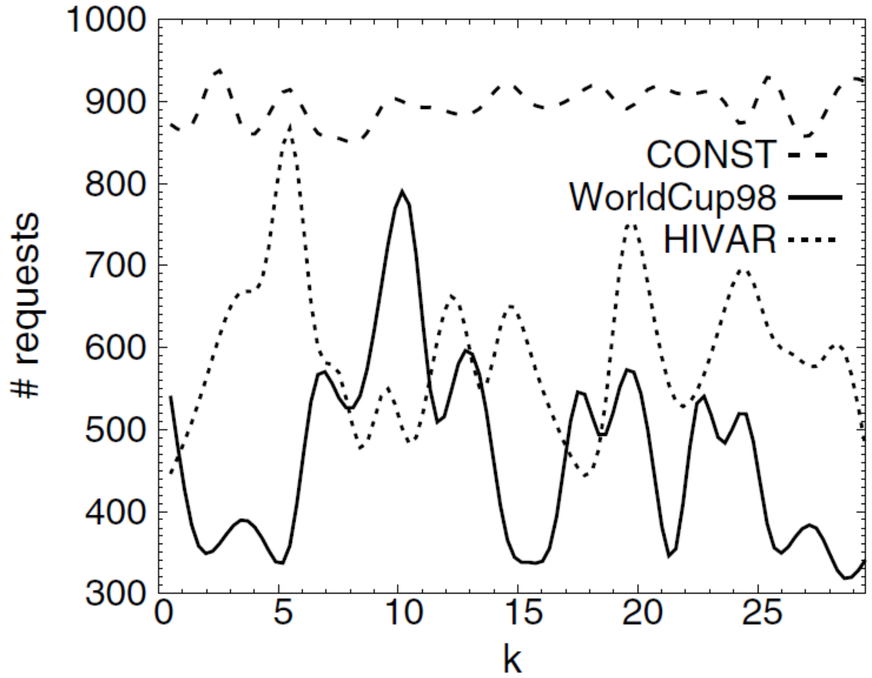
*Code available at:
http://www.traffic.comics.unina.it/cloud



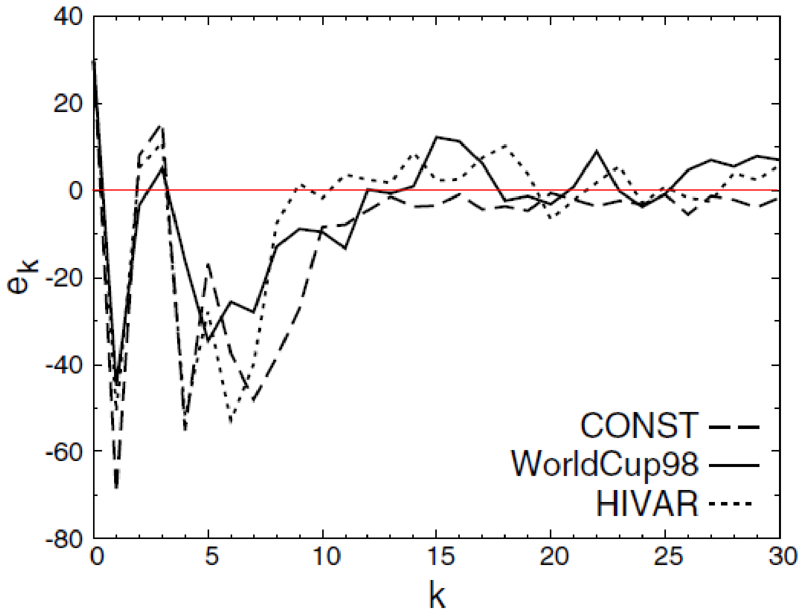
Workloads (WLs)

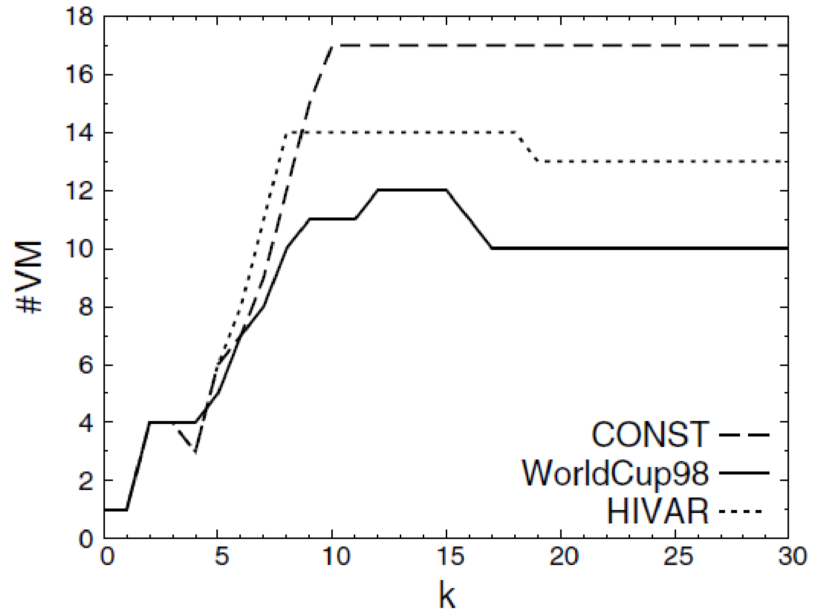
3 different workloads

- **CONST**
  900 reqs/time interval
- **WorldCup98**
  1998 world cup web site
- **HIVAR**
  synthetic workload with high variability



Robustness to different WLs

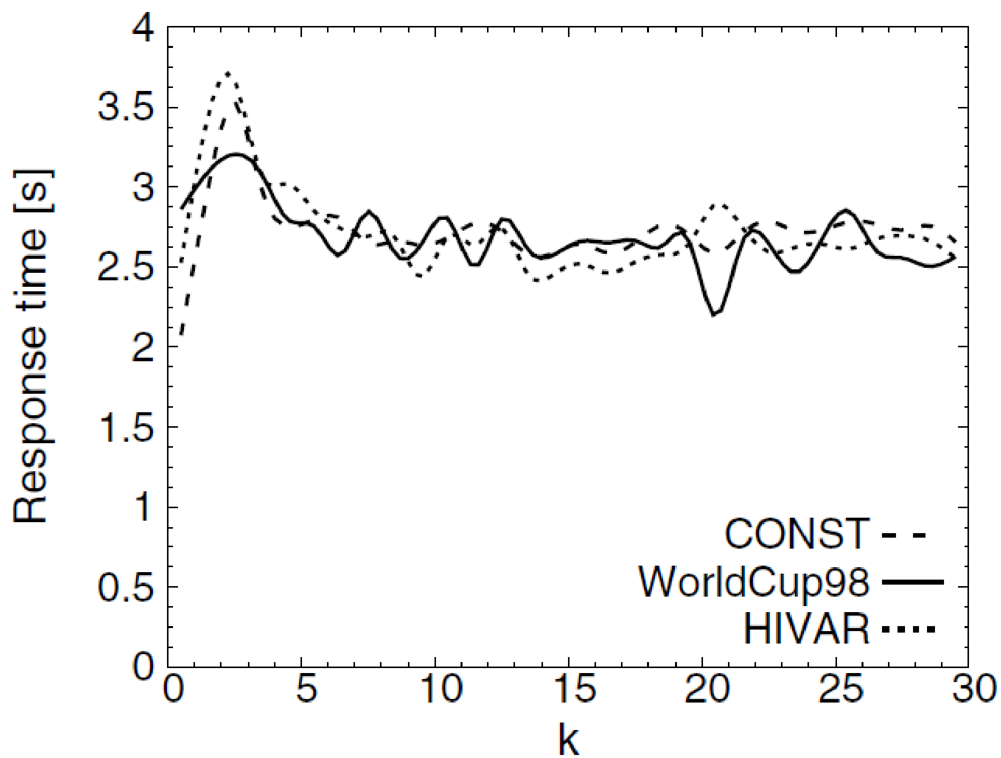




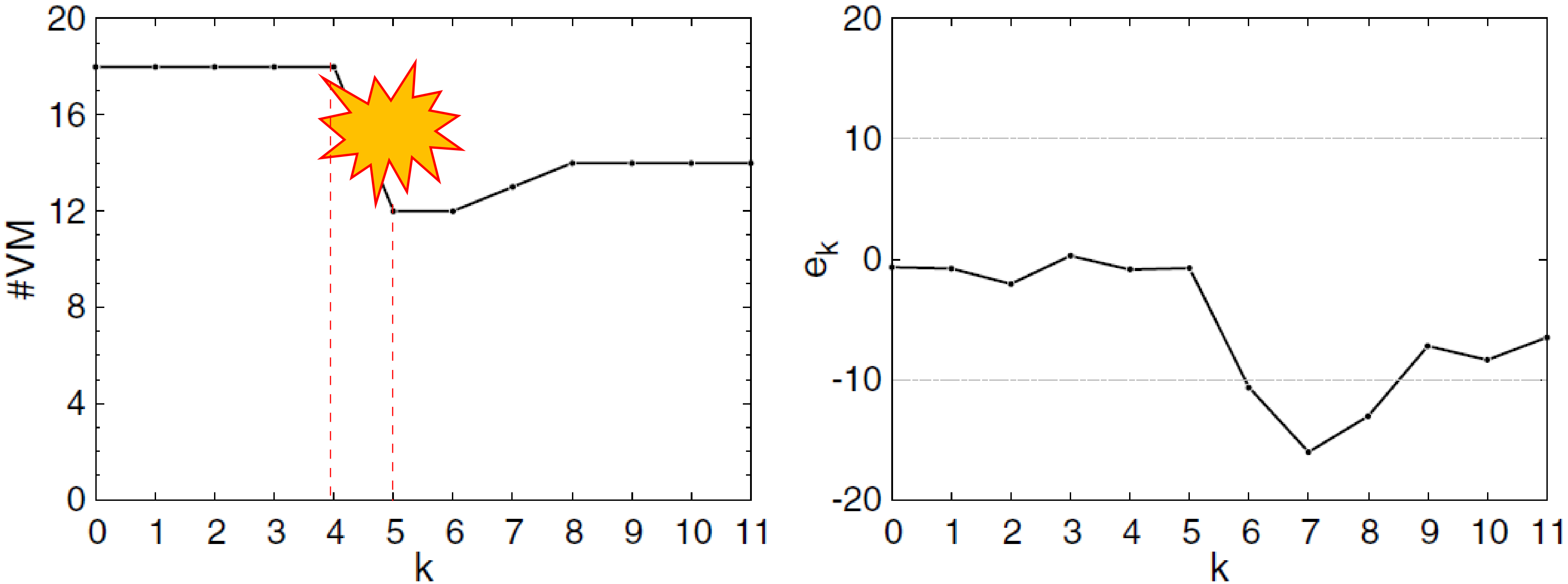
- $y_d = 30$
- Results confirm that also in cases where operating condition are highly varying, control objective is achieved with short-term performance degradation

Impact on Latency

Low variability of response time despite the high variability of the operating conditions



Robustness against failures

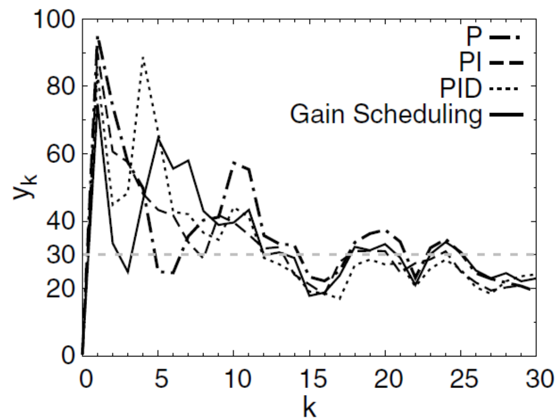


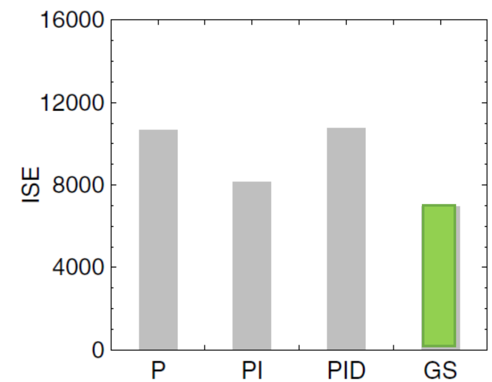
- 1/3 of the VMs fail between k=4 and k=5
- The control action adapts its gains and counteracts the effect of the failures
- At k=9 the error is within 10% bound

Comparison against fixed-gain controllers

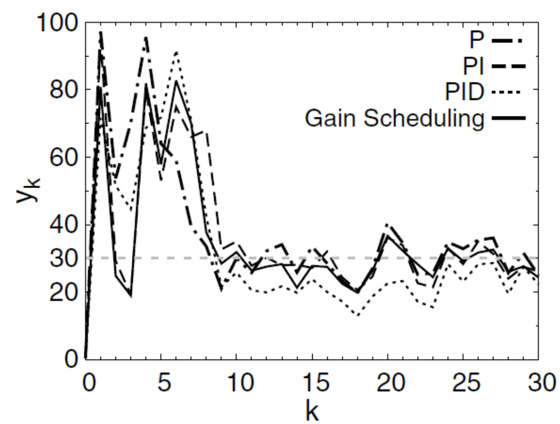
Gain Scheduling *VS* **P, PI, PID**

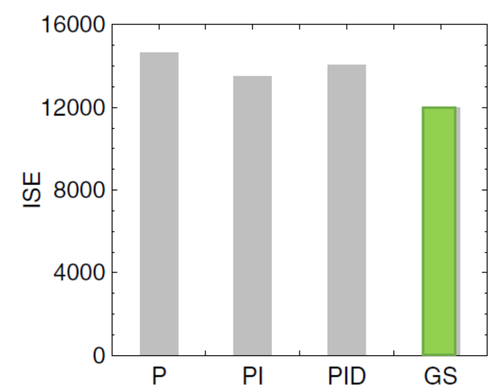
WorldCup98 workload





HIVAR workload





ISE: Integral of Squared Errors

Conclusion

- PID control approach + *Gain Scheduling* policy
  - No need of a priori knowledge of the system or of the workload
  - Tested on AWS EC2
  - Robust against different workloads and VM failure
  - Performs better than previously proposed approaches

- Future work
  - Merging multiple metrics
    (e.g. throughput, CPU load,etc.)
  - Implementing other sophisticated control approaches
    (e.g. fuzzy logic)

Questions?

valerio.persico@unina.it

http://wpage.unina.it/valerio.persico

DIETI. UNIVERSITA' DEGLI STUDI DI NAPOLI FEDERICO II