Bâtiment IMAG, Université Grenoble Alpes
700, avenue centrale
38401 Saint Martin d'Hères, France
http://www-verimag.imag.fr

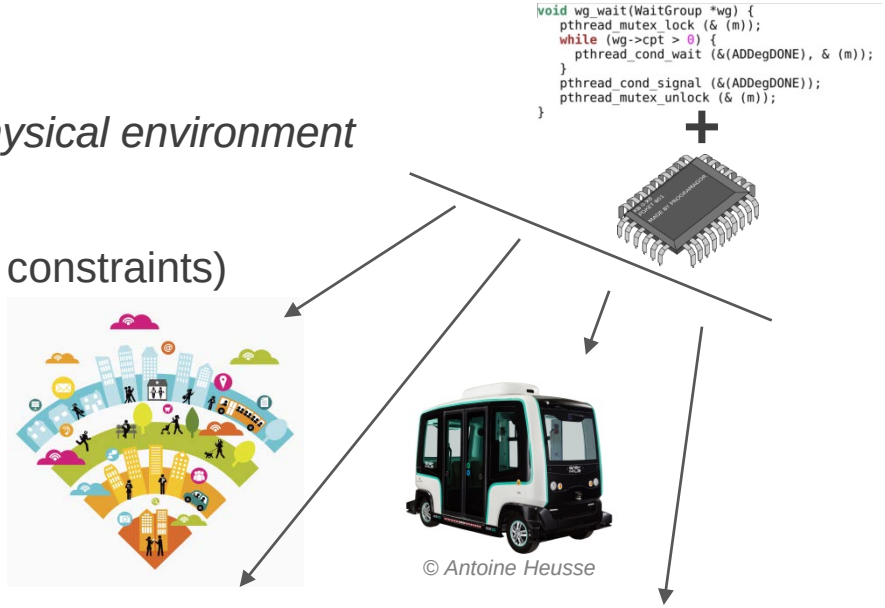
ParetoLib: A Python Library for Parameter Synthesis

José-Ignacio Requeno
Alexey Bakhirkin
Nicolas Basset
Oded Maler*

Context

CYBER-PHYSICAL SYSTEMS

- **Cyber-Physical Systems:**
  - *hardware/software interacting with the physical environment*
  - computerized control systems
  - uncertain / changing environment (timing constraints)
  - large scale, distributed / networked
- **Criticality:**
  - safety (absence of errors)
  - security (resistance to attacks)
  - certification
- **Validation & Verification:**
  - **hybrid: discrete + real-time systems**

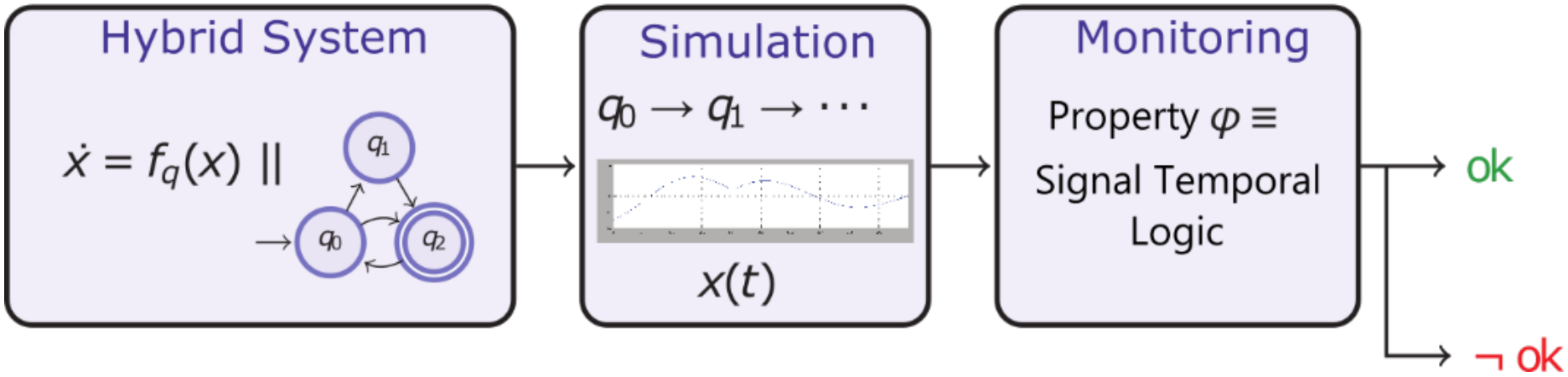


© Antoine Heusse

© Sancio83

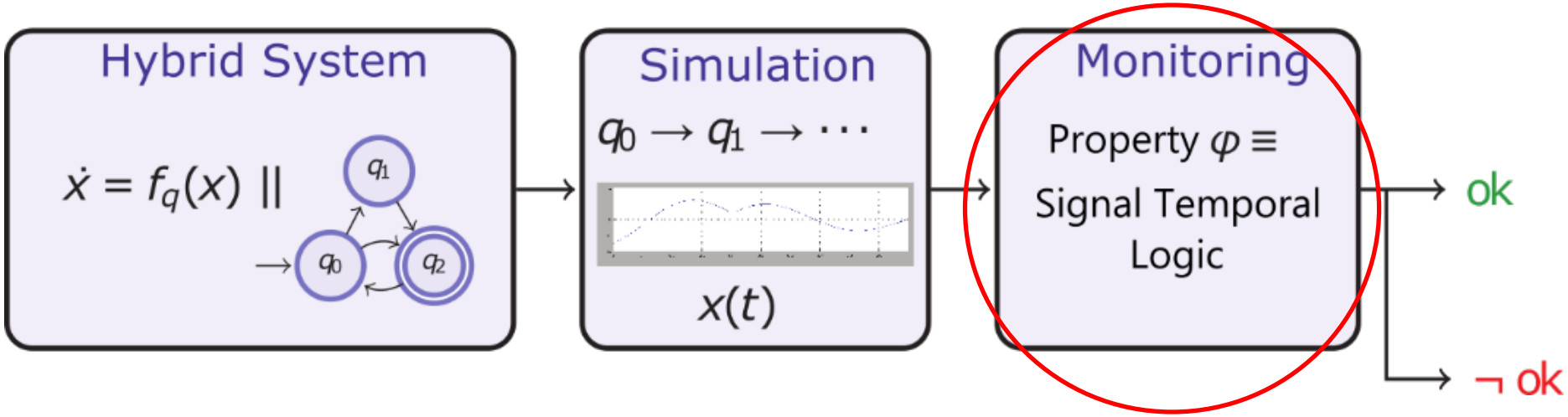
MONITORING

- Models are generally hybrid systems producing hybrid **traces**
- Runtime verification (aka monitoring) analyses a **single** behavior at a time



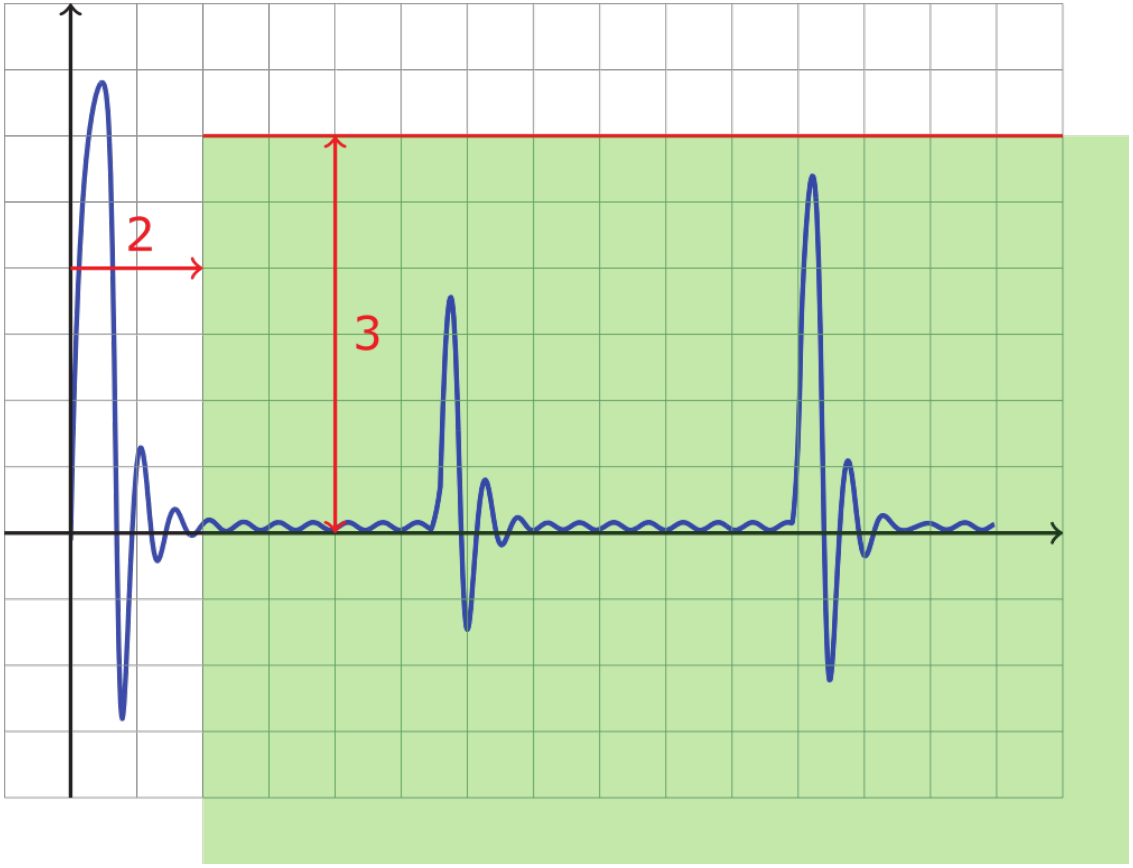
MONITORING

- Models are generally hybrid systems producing hybrid **traces**
- Runtime verification (aka monitoring) analyses a **single** behavior at a time



MOTIVATION

After 2s, the signal is never above 3

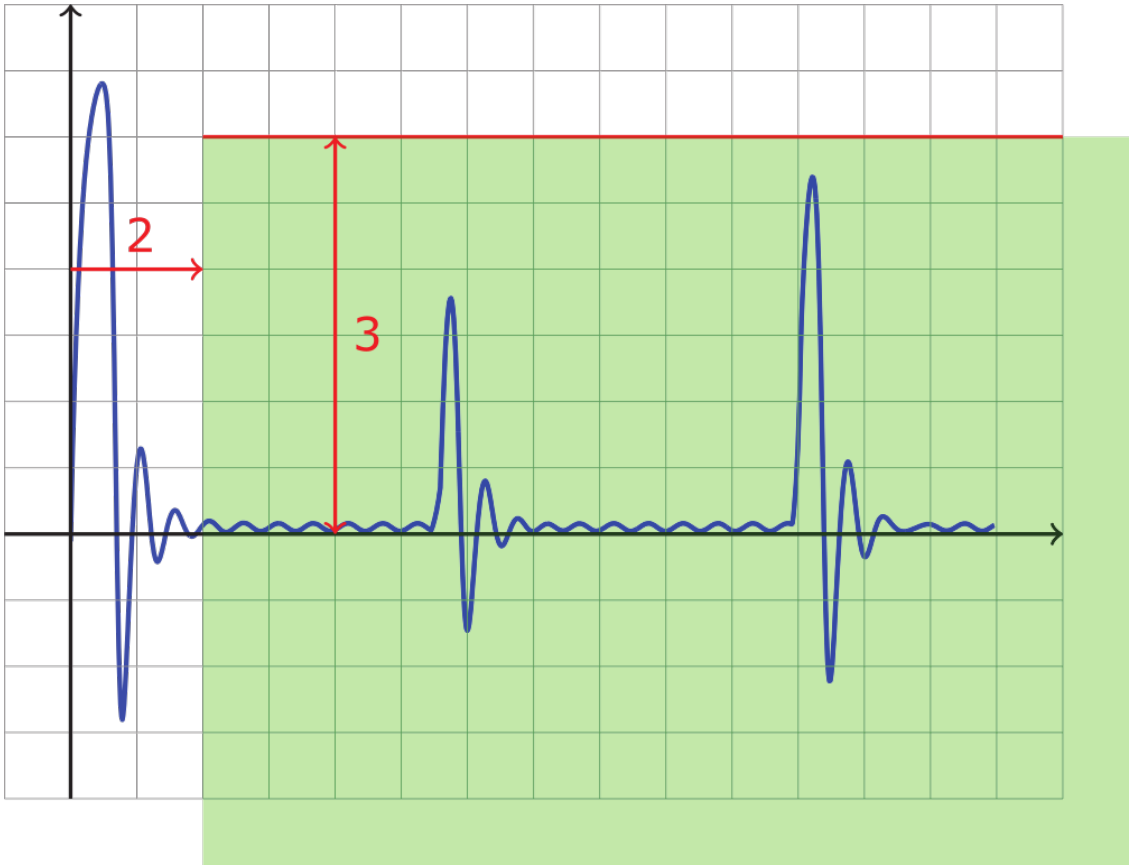


*Donzé, A.: On signal temporal logic. In: Legay, A., Bensalem, S. (eds.) RV 2013. LNCS, vol. 8174, pp. 382–383. Springer, Heidelberg (2013).

STL EXAMPLE

After 2s, the signal is never above 3

$$\varphi := \mathsf{F}_{[2,\infty]} \ (x[t] < 3)$$



PROBLEM

Parameter Synthesis

PARAMETER SYNTHESIS

Given a Parametric STL formula

$$\boldsymbol{\varphi(p_1, \dots, p_n)}$$

identify the valuations $\boldsymbol{v(p_i)}$ such that

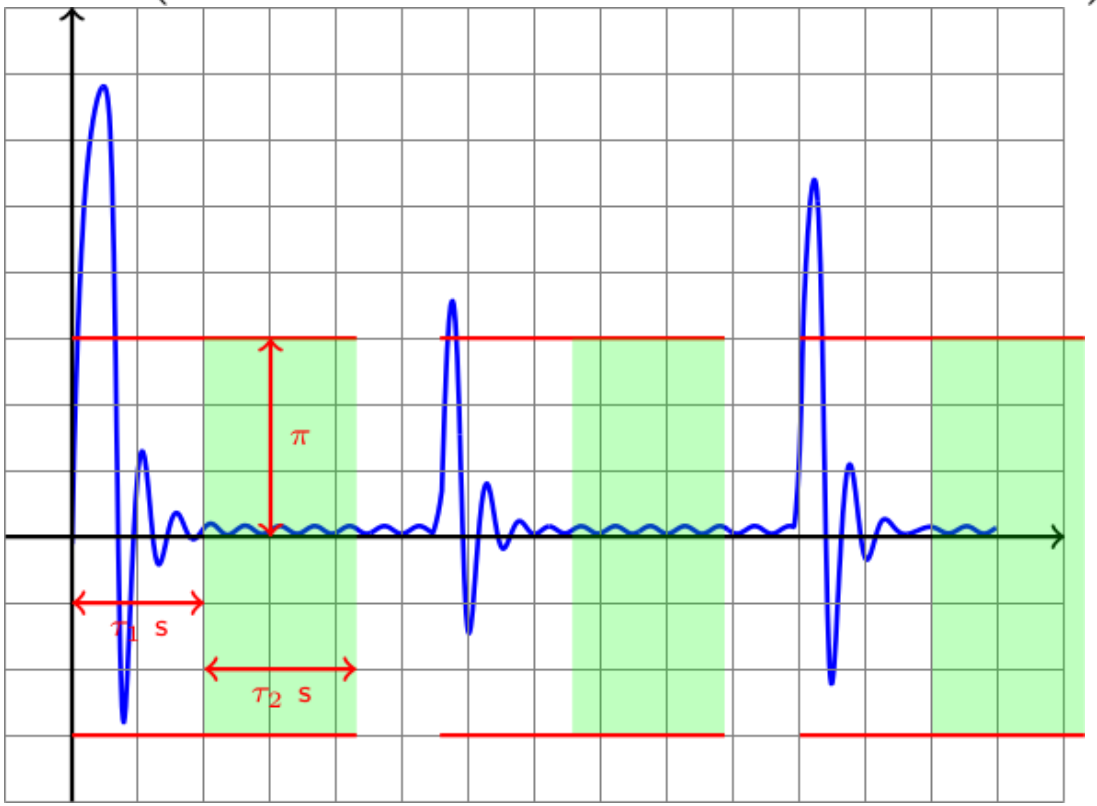
$$\boldsymbol{x \vDash \varphi(v(p_1), \dots, v(p_n))}$$

for each parameter $\boldsymbol{p_i}$

SIGNAL STABILIZATION

Always |x| > _ then after _ s, |x| settles under _ for _ s

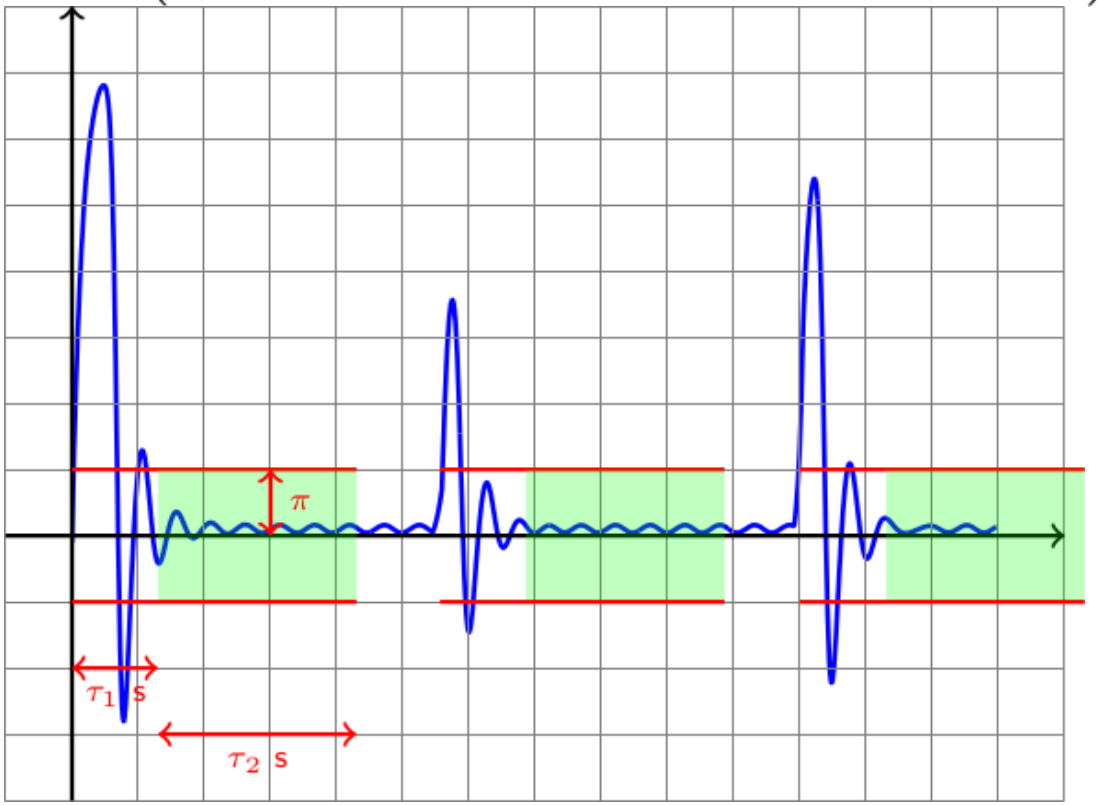
$$\varphi := \mathsf{G}\left(x[t] > \pi \rightarrow \mathsf{F}_{[0,\tau_1]}\ \left(\mathsf{G}_{[0,\tau_2]}\ x[t] < \pi\right)\right)$$



SIGNAL STABILIZATION

Always |x| > _ then after _ s, |x| settles under _ for _ s

$$\varphi := \mathsf{G}\left(x[t] > \pi \rightarrow \mathsf{F}_{[0,\tau_1]} \left(\mathsf{G}_{[0,\tau_2]}\ x[t] < \pi\right)\right)$$

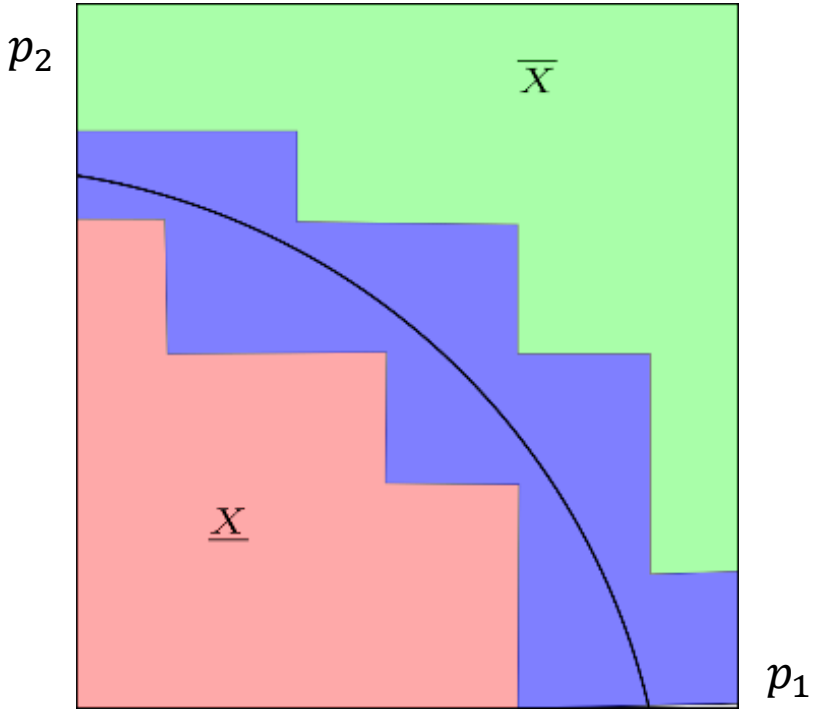


PARAMETER SYNTHESIS

- **Challenges**
  - Multiple solutions: which one to chose?
  - Tightness implies to “optimize” the valuation $v(p_i)$ for each parameter p_i
- **The multi-criteria optimization problem can be greatly simplified if the formula is monotonic in each p_i**
  - In case of monoticity, the border of the validity domain has the structure of a **Pareto front**

VALIDITY DOMAINS

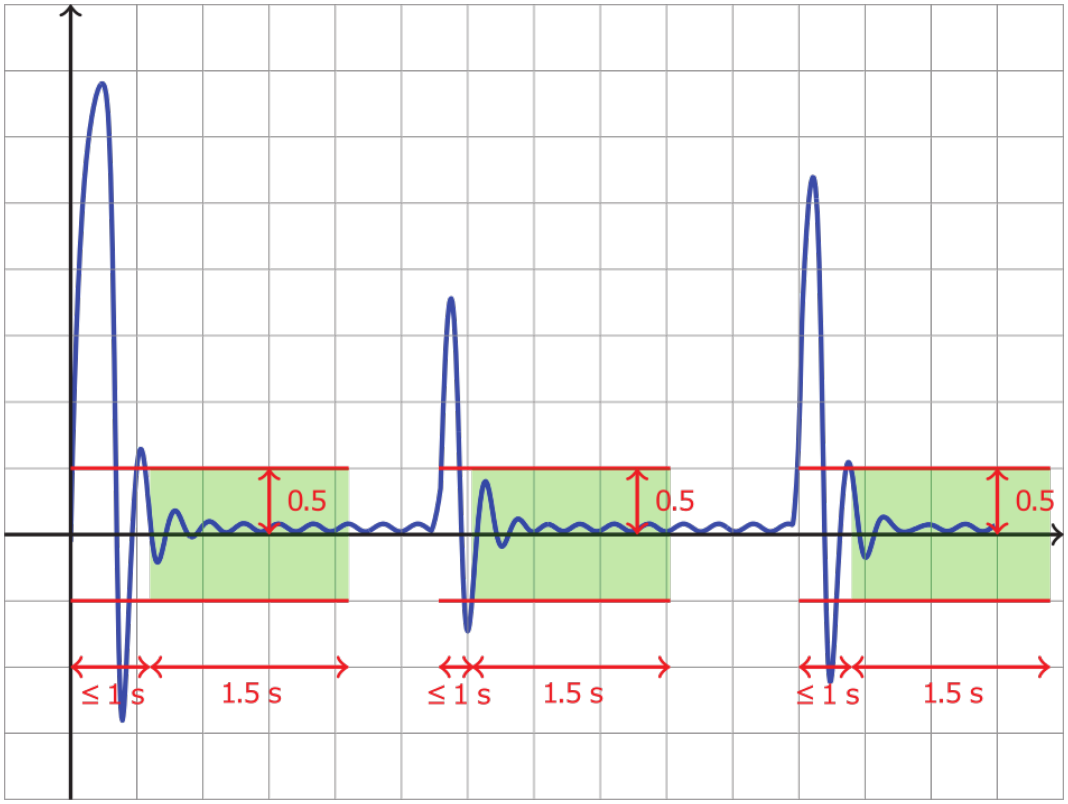
- The validity domain $\boldsymbol{X}$ of formula $\boldsymbol{\varphi}$ and signal $\boldsymbol{x}$ is the set of valuations $\boldsymbol{v}(\boldsymbol{p_i})$ such that $\boldsymbol{x} \vDash \boldsymbol{\varphi}(\boldsymbol{v}(\boldsymbol{p_1}), \dots, \boldsymbol{v}(\boldsymbol{p_n}))$ for each parameter $\boldsymbol{p_i}$
- In case of monoticity, the border of the validity domain has the structure of a **Pareto front**



SIGNAL STABILIZATION

Always |x| > 0.5 then after 1s, |x| settles under 0.5 for 1.5 s

$$\varphi := \mathsf{G}(x[t] > .5 \rightarrow \ \mathsf{F}_{[0,.6]} \ \ (\ \mathsf{G}_{[0,1.5]} \ x[t] < 0.5))$$

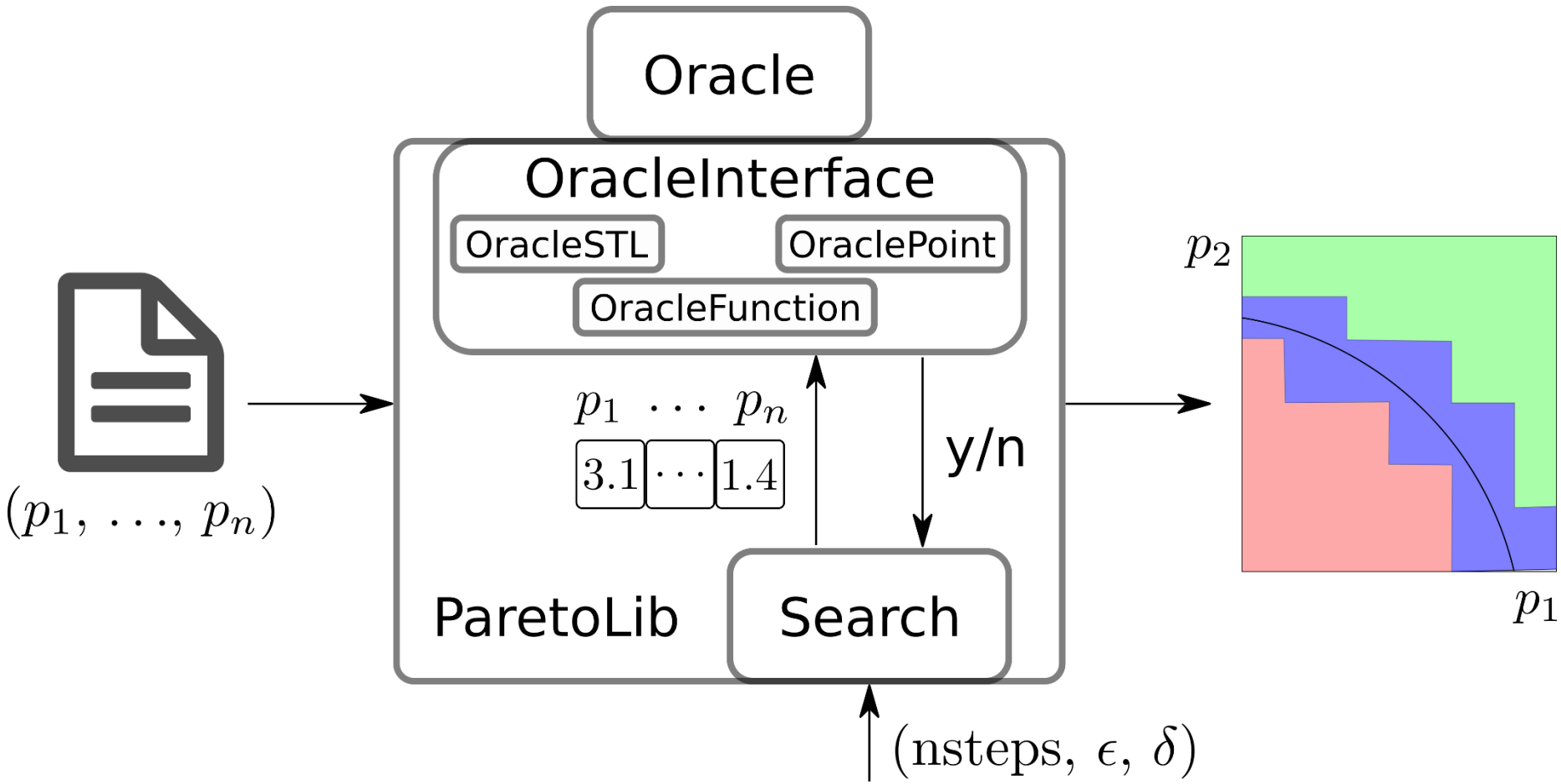


ParetoLib

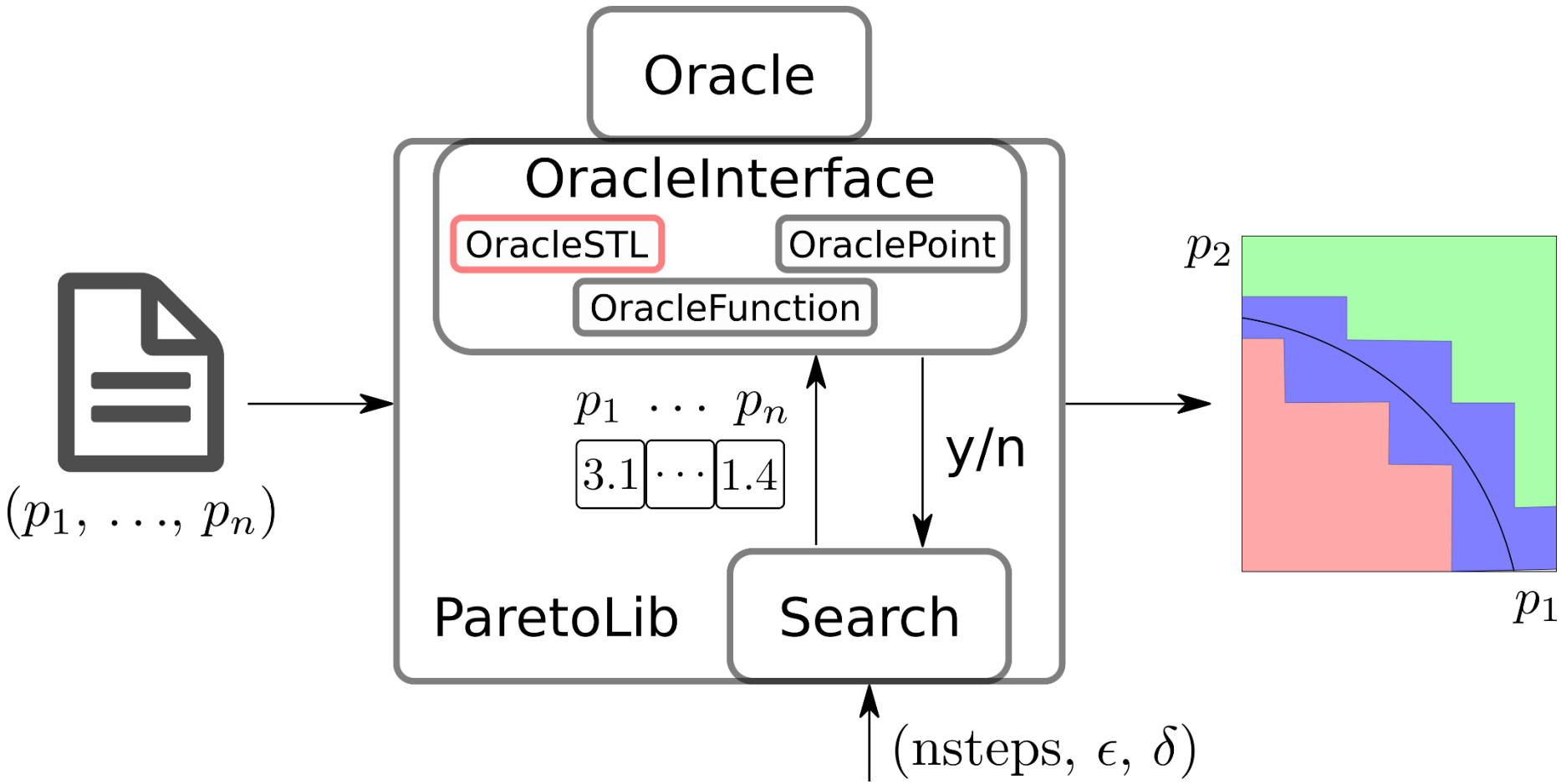
PARETOLIB

- **Python library for solving multi-criteria optimization problems:**
  - Faster convergence than pre-dating similar algorithms
  - Parallel computation in multi-core CPU's
  - Generic interface for domain-specific decision procedures (Oracle)
- **Application to parameter synthesis:**
  - Integration with Signal Temporal Logic runtime monitors
  - Experiments with parametric Signal Temporal Logic specifications

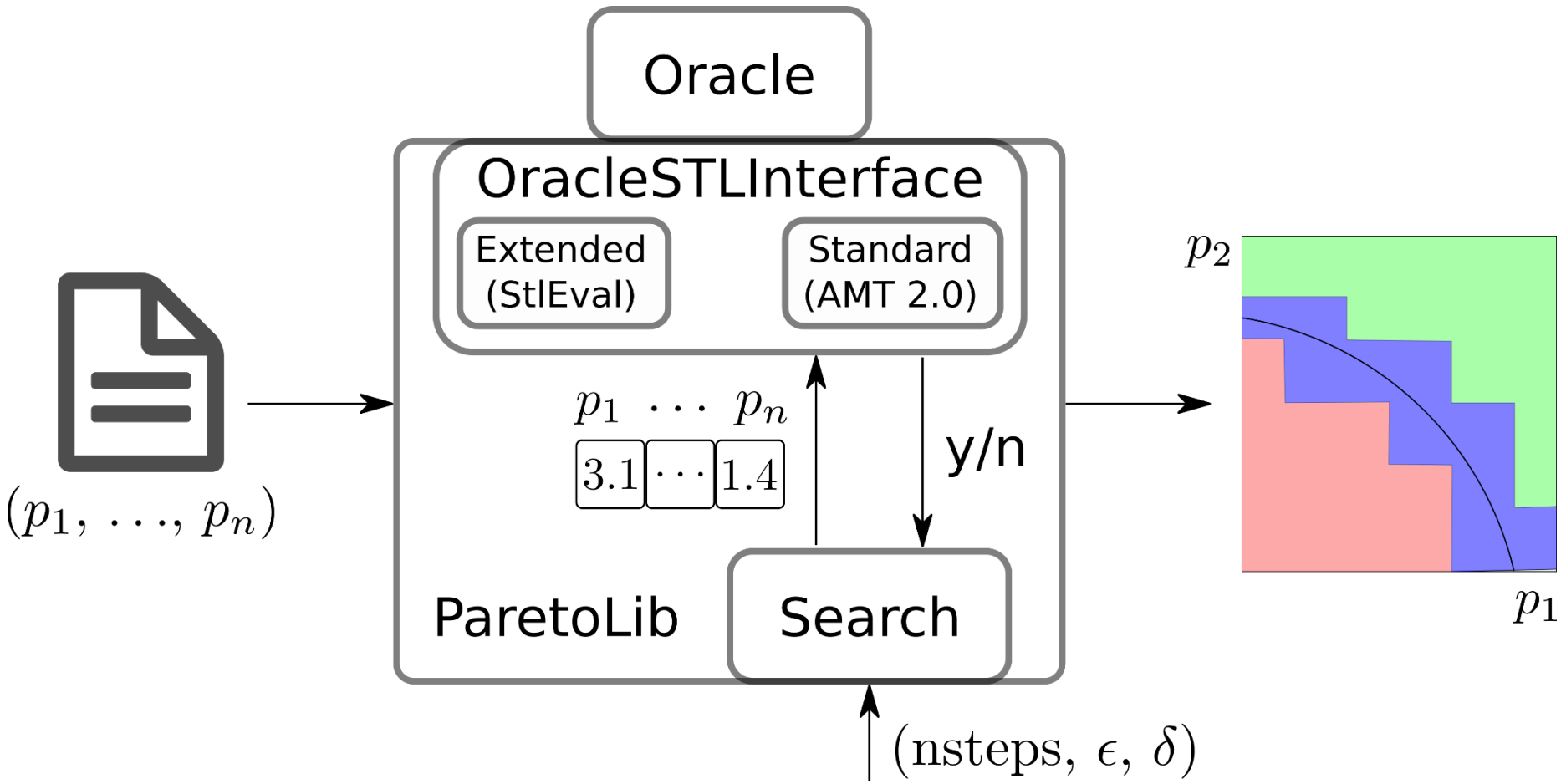
PARETOLIB LIBRARY



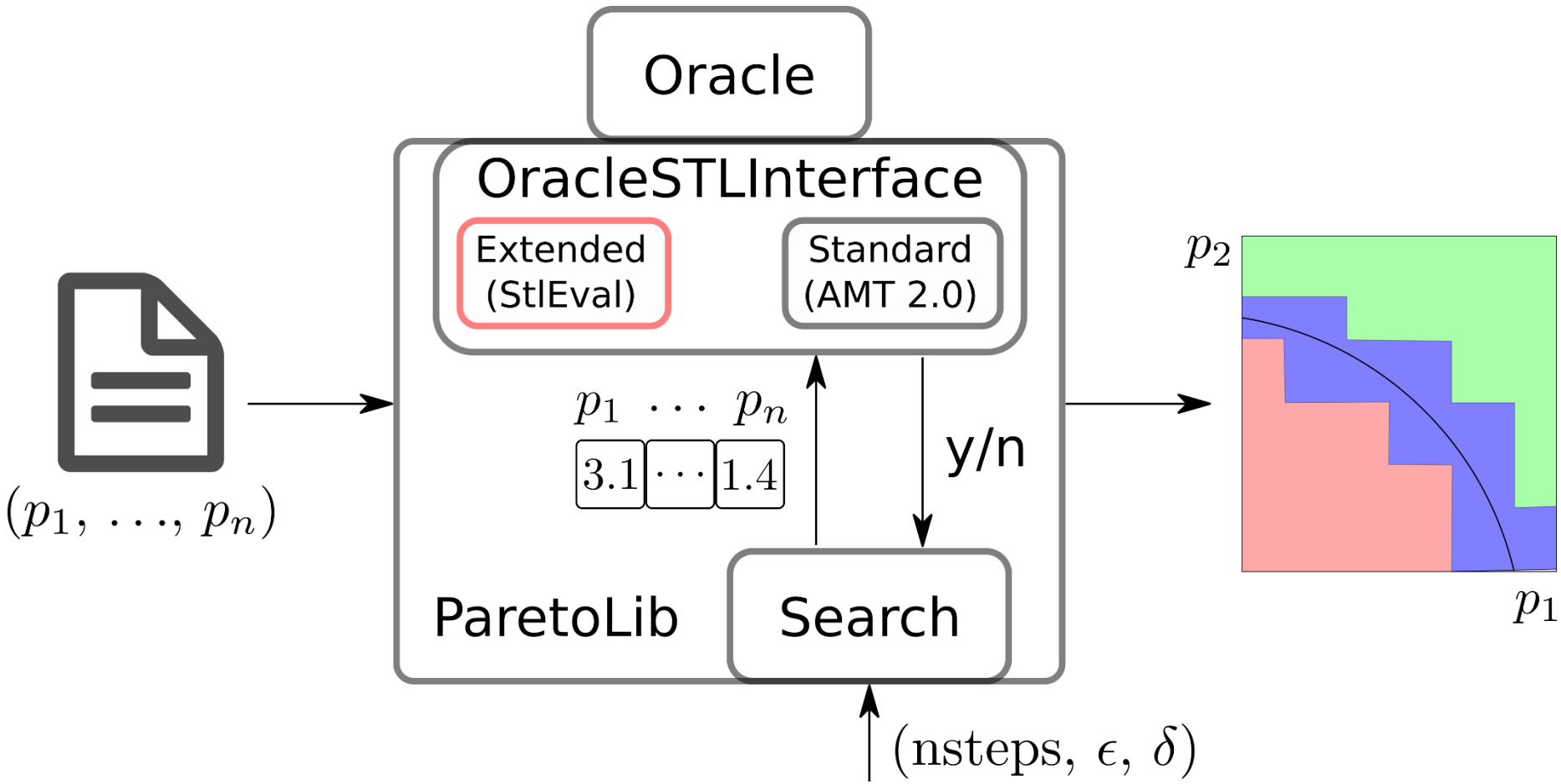
PARETOLIB LIBRARY



PARETOLIB LIBRARY



PARETOLIB LIBRARY



PARETOLIB LIBRARY

```
from ParetoLib.Oracle.OracleSTLe import OracleSTLeLib
from ParetoLib.Search.Search import Search2D, EPS, DELTA, STEPS

# Definition of the n-dimensional space
min_x, min_y = (0.0, 0.0)
max_x, max_y = (300.0, 1.0)

# File containing the definition of the Oracle
nfile='Tests/Search/OracleSTLe/2D/stabilization.txt'
oracle = OracleSTLeLib()
oracle.from_file(nfile, human_readable=True)

rs=Search2D(ora=oracle,
          min_cornerx=min_x,min_cornery=min_y,
          max_cornerx=max_x,max_cornery=max_y,
          epsilon=EPS, delta=DELTA, max_step=STEPS,
          opt_level=0,parallel=False)
rs.fo_file("result.zip")
```

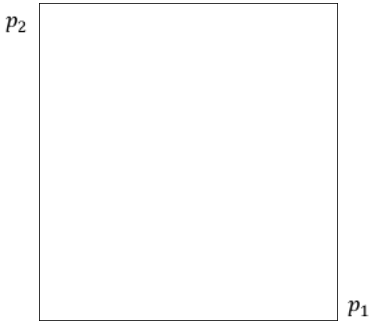
PARETOLIB LIBRARY

```
from ParetoLib.Oracle.OracleSTLe import OracleSTLeLib
from ParetoLib.Search.Search import Search2D, EPS, DELTA, STEPS

# Definition of the n-dimensional space
min_x, min_y = (0.0, 0.0)
max_x, max_y = (300.0, 1.0)

# File containing the definition of the Oracle
nfile='Tests/Search/OracleSTLe/2D/stabilization.txt'
oracle = OracleSTLeLib()
oracle.from_file(nfile, human_readable=True)

rs=Search2D(ora=oracle,
          min_cornerx=min_x,min_cornery=min_y,
          max_cornerx=max_x,max_cornery=max_y,
          epsilon=EPS, delta=DELTA, max_step=STEPS,
          opt_level=0,parallel=False)
rs.fo_file("result.zip")
```



PARETOLIB LIBRARY

```
from ParetoLib.Oracle.OracleSTLe import OracleSTLeLib
from ParetoLib.Search.Search import Search2D, EPS, DELTA, STEPS

# Definition of the n-dimensional space
min_x, min_y = (0.0, 0.0)
max_x, max_y = (300.0, 1.0)

# File containing the definition of the Oracle
nfile='Tests/Search/OracleSTLe/2D/stabilization.txt'
oracle = OracleSTLeLib()
oracle.from_file(nfile, human_readable=True)

rs=Search2D(ora=oracle,
          min_cornerx=min_x,min_cornery=min_y,
          max_cornerx=max_x,max_cornery=max_y,
          epsilon=EPS, delta=DELTA, max_step=STEPS,
          opt_level=0,parallel=False)
rs.fo_file("result.zip")
```

OracleSTLInterface

Extended (StlEval)

Standard (AMT 2.0)

PARETOLIB LIBRARY

```
from ParetoLib.Oracle.OracleSTLe import OracleSTLeLib
from ParetoLib.Search.Search import Search2D, EPS, DELTA, STEPS

# Definition of the n-dimensional space
min_x, min_y = (0.0, 0.0)
max_x, max_y = (300.0, 1.0)

# File containing the definition of the Oracle
nfile='Tests/Search/OracleSTLe/2D/stabilization.txt'
oracle = OracleSTLeLib()
oracle.from_file(nfile, human_readable=True)

rs=Search2D(ora=oracle,
            min_cornerx=min_x,min_cornery=min_y,
            max_cornerx=max_x,max_cornery=max_y,
            epsilon=EPS, delta=DELTA, max_step=STEPS,
            opt_level=0,parallel=False)
rs.fo_file("result.zip")
```

Search

$(\text{nsteps}, \epsilon, \delta)$

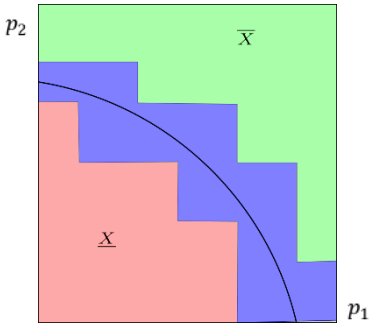
PARETOLIB LIBRARY

```
from ParetoLib.Oracle.OracleSTLe import OracleSTLeLib
from ParetoLib.Search.Search import Search2D, EPS, DELTA, STEPS

# Definition of the n-dimensional space
min_x, min_y = (0.0, 0.0)
max_x, max_y = (300.0, 1.0)

# File containing the definition of the Oracle
nfile='Tests/Search/OracleSTLe/2D/stabilization.txt'
oracle = OracleSTLeLib()
oracle.from_file(nfile, human_readable=True)

rs=Search2D(ora=oracle,
          min_cornerx=min_x,min_cornery=min_y,
          max_cornerx=max_x,max_cornery=max_y,
          epsilon=EPS, delta=DELTA, max_step=STEPS,
          opt_level=0,parallel=False)
rs.fo_file("result.zip")
```

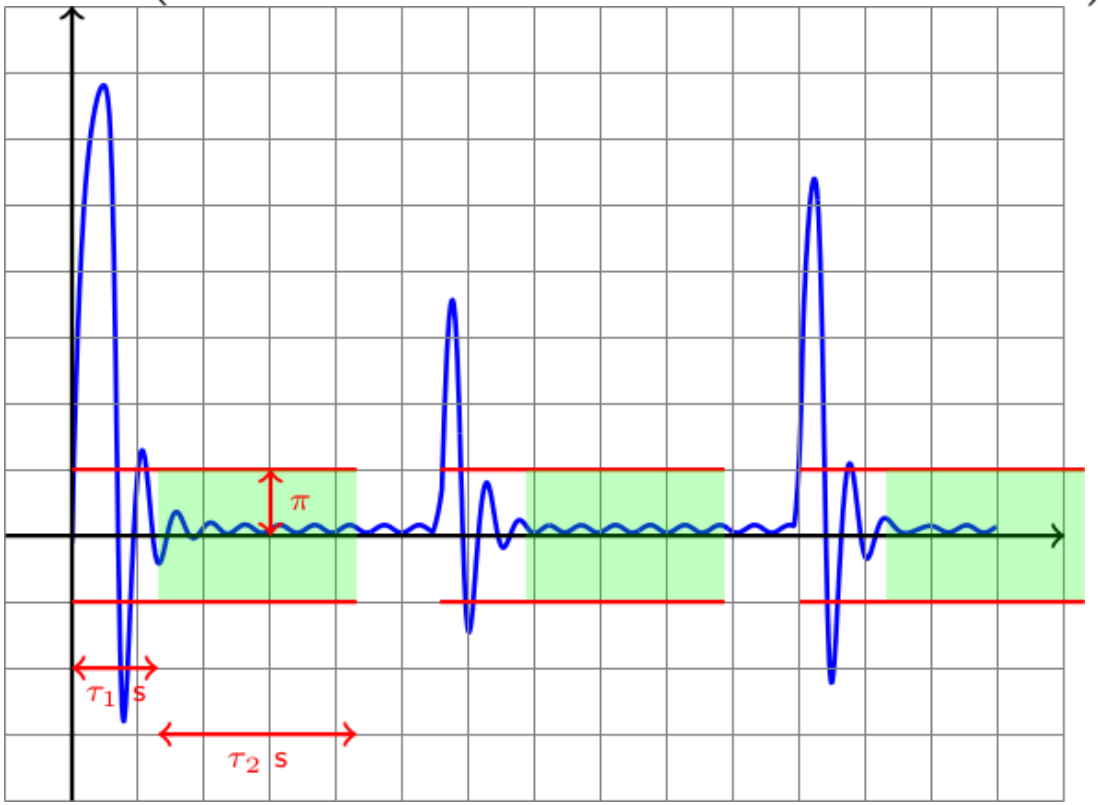


Experiments

EXPERIMENTS

Always |x| > _ then after _ s, |x| settles under _ for _ s

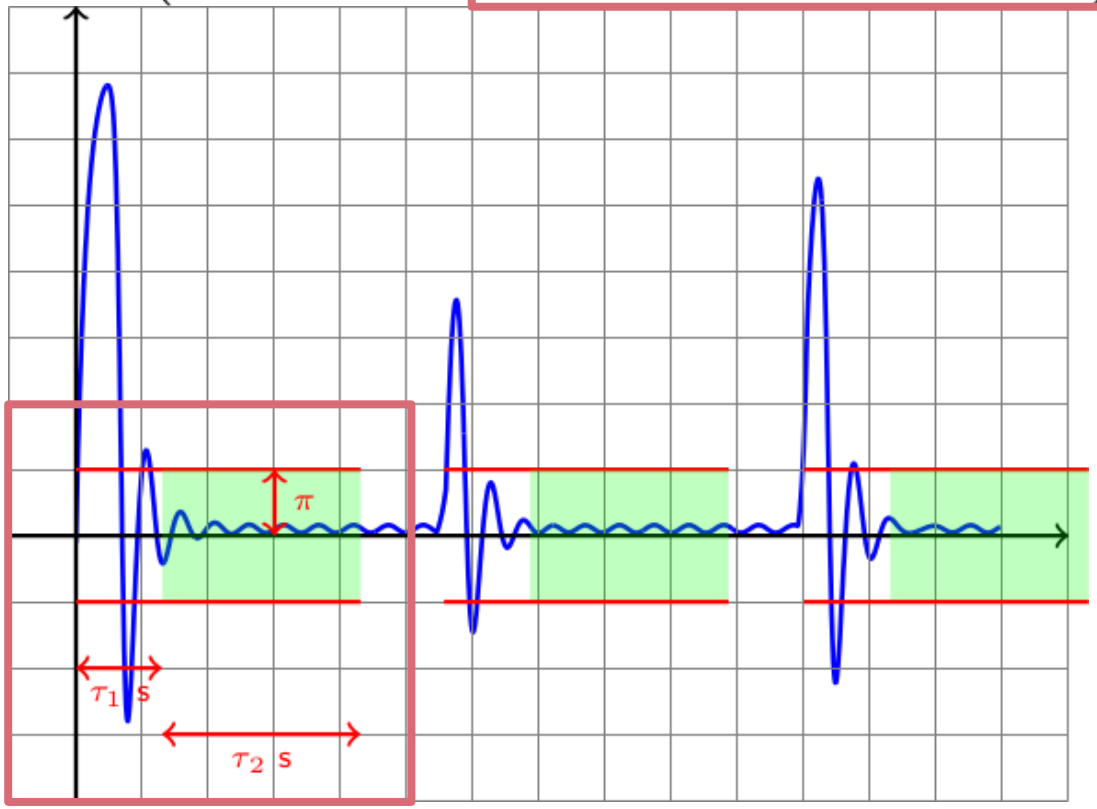
$$\varphi := \mathsf{G}\left(x[t] > \pi \rightarrow \mathsf{F}_{[0,\tau_1]} \left(\mathsf{G}_{[0,\tau_2]}\ x[t] < \pi\right)\right)$$



EXPERIMENTS

Always |x| > _ then after _ s, |x| settles under _ for _ s

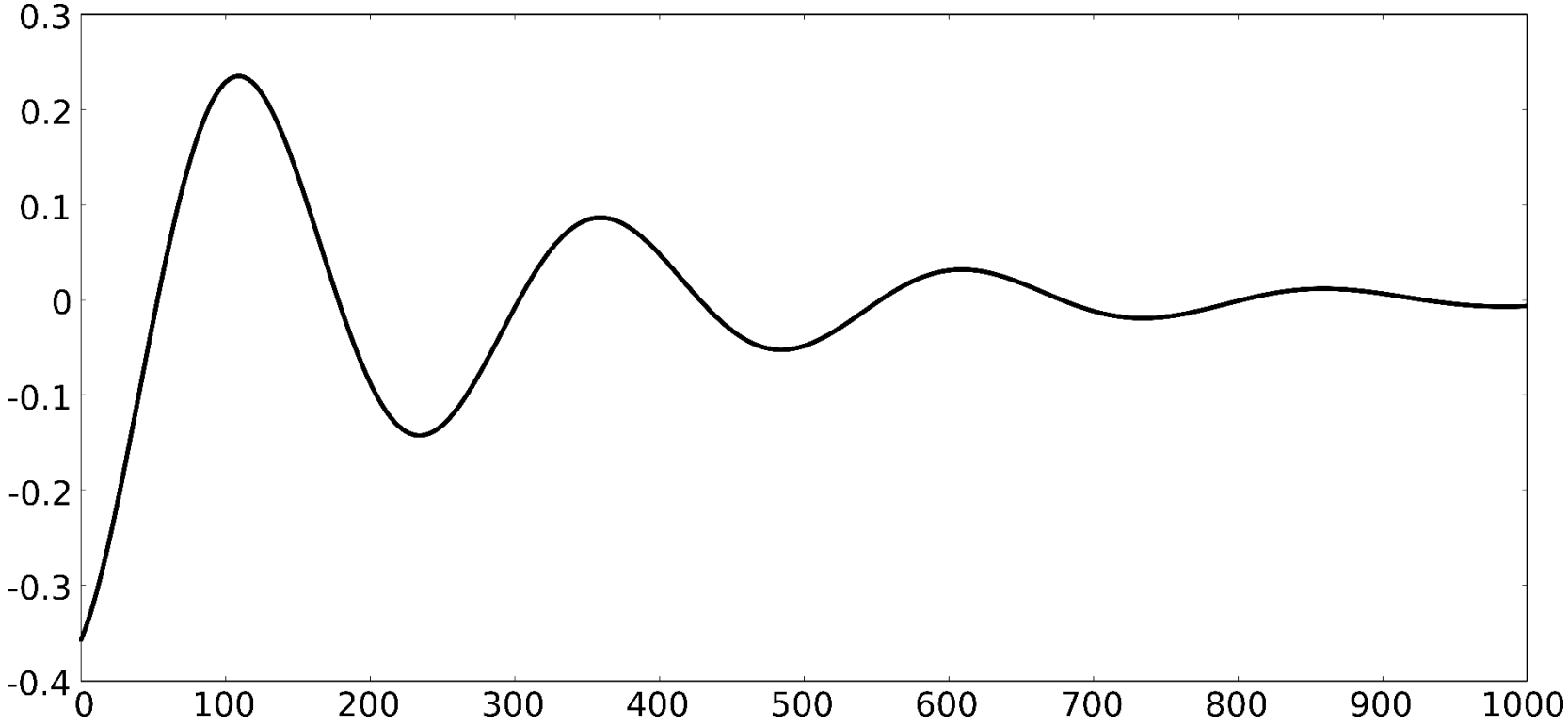
$$\varphi := \mathsf{G}\left(x[t] > \pi \rightarrow \boxed{\mathsf{F}_{[0,\tau_1]}\ \left(\mathsf{G}_{[0,\tau_2]}\ x[t] < \pi\right)}\right)$$



EXPERIMENTS

Always |x| > p2 then after p1 s, |x| settles under p2 indefinitely

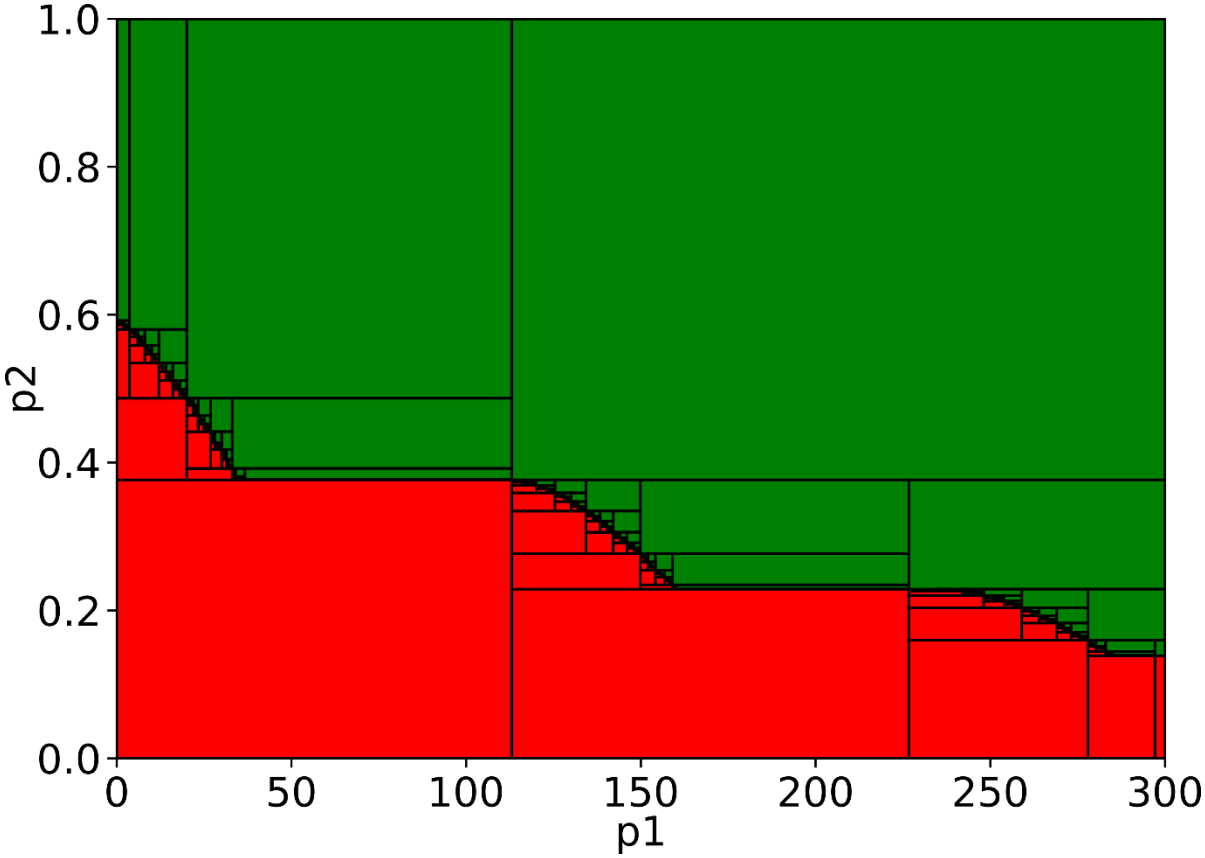
$$\varphi := \mathrm{F}_{[0,p_1]}\mathrm{G}_{[0,\infty]}(\mathrm{On}_{[0,\infty]}\mathrm{Max}\, x - \mathrm{On}_{[0,\infty]}\mathrm{Min}\, x)) < p_2$$



EXPERIMENTS

Always |x| > p2 then after p1 s, |x| settles under p2 indefinitely

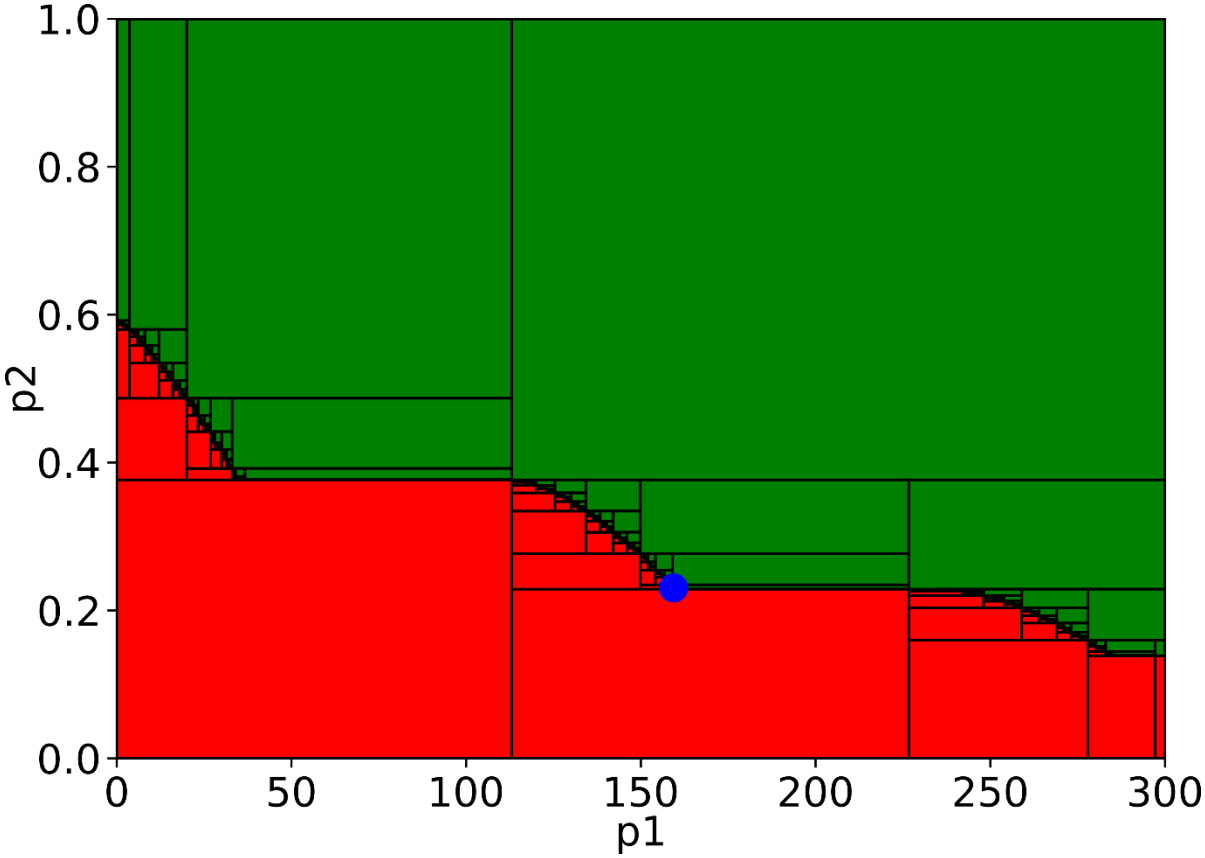
$$\varphi := \mathrm{F}_{[0,p_1]}\mathrm{G}_{[0,\infty]}(\mathrm{On}_{[0,\infty]}\mathrm{Max}\, x - \mathrm{On}_{[0,\infty]}\mathrm{Min}\, x)) < p_2$$



EXPERIMENTS

Always |x| > 0.25 then after 155 s, |x| settles under 0.25 indefinitely

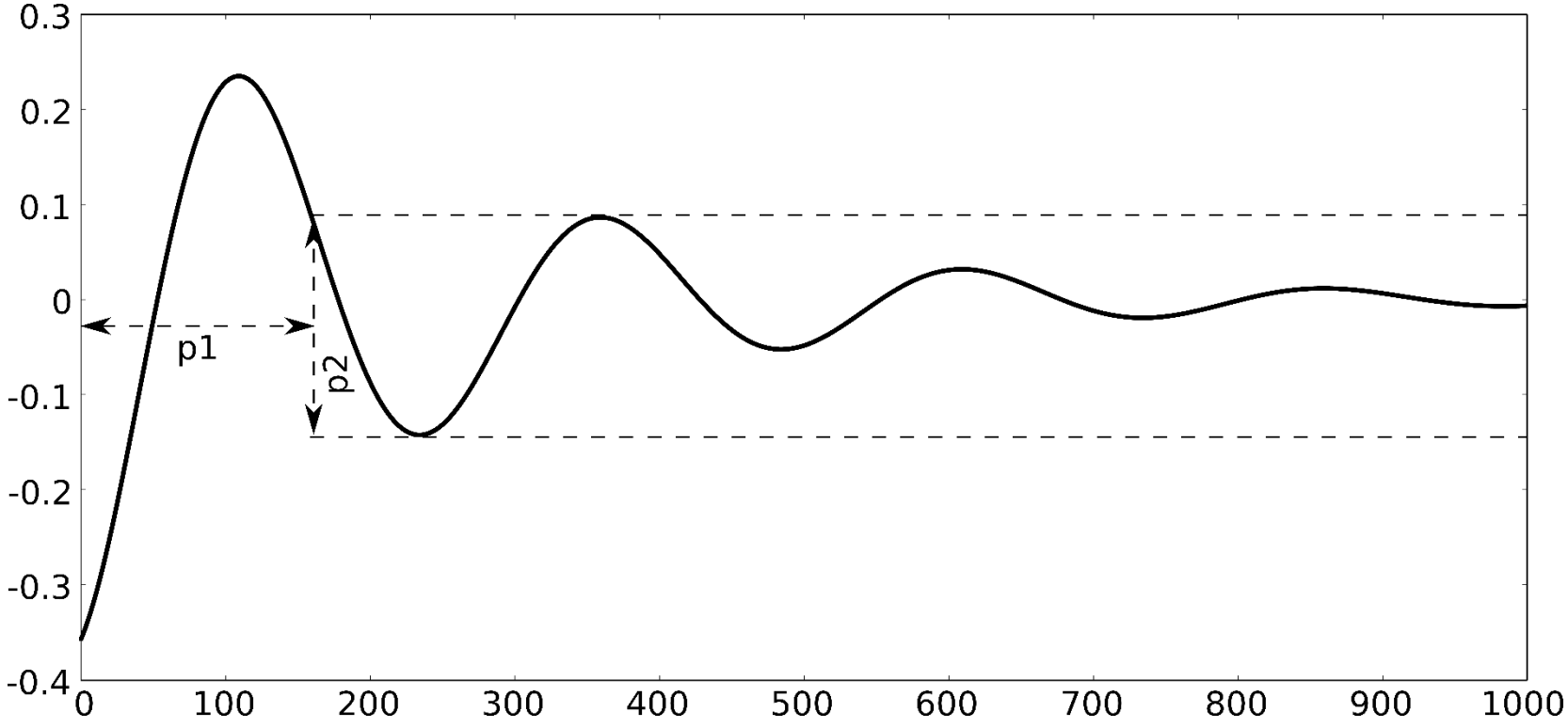
$$\varphi := \mathrm{F}_{[0,155]}\mathrm{G}_{[0,\infty]}(\mathrm{On}_{[0,\infty]}\mathrm{Max}\ x - \mathrm{On}_{[0,\infty]}\mathrm{Min}\ x)) < 0.25$$



EXPERIMENTS

Always |x| > 0.25 then after 155 s, |x| settles under 0.25 indefinitely

$$\varphi := \mathrm{F}_{[0,155]}\mathrm{G}_{[0,\infty]}(\mathrm{On}_{[0,\infty]}\mathrm{Max}\ x - \mathrm{On}_{[0,\infty]}\mathrm{Min}\ x)) < 0.25$$



Conclusions

CONCLUSIONS

- **Python library for solving multi-criteria optimization problems:**
  - Faster convergence than pre-dating similar algorithms
  - Parallel computation in multi-core CPU's
  - Generic interface for domain-specific decision procedures (Oracle)
- **Application to parameter synthesis:**
  - Integration with Signal Temporal Logic runtime monitors
  - Experiments with parametric Signal Temporal Logic specifications

FUTURE WORK

- **Python library for solving multi-criteria optimization problems:**
  - Keep on improving the tool performance
  - Introduction of new features
- **Application to parameter synthesis:**
  - Integration with new temporal logics

THE END