Randomized Algorithms for Design of Uncertain Complex Systems

Roberto Tempo

IEIIT-CNR
Politecnico di Torino
tempo@polito.it

Randomized Algorithms: A Success Story

A Success Story

- **Randomized Algorithms** (RAs) are successfully used in various areas outside control

1. CS: Sorting problems (e.g., QuickSort algorithm)
2. CS: Data structuring, search trees, graph algorithms
3. Mathematics of finance: Computation of integrals
4. Genomics: String matching and classification
5. Robotics: Motion and path planning problems

Powerful Tool

- Randomized Algorithms are powerful tools for solving problems which are computationally intractable (in a classical sense)
- We introduce a different notion of *problem tractability*
- For uncertain complex systems break the curse of dimensionality[1]

[1] R. Bellman (1957)

Randomized Algorithms and Control

- Remarkably, randomized methods and algorithms are not used *systematically* in systems and control

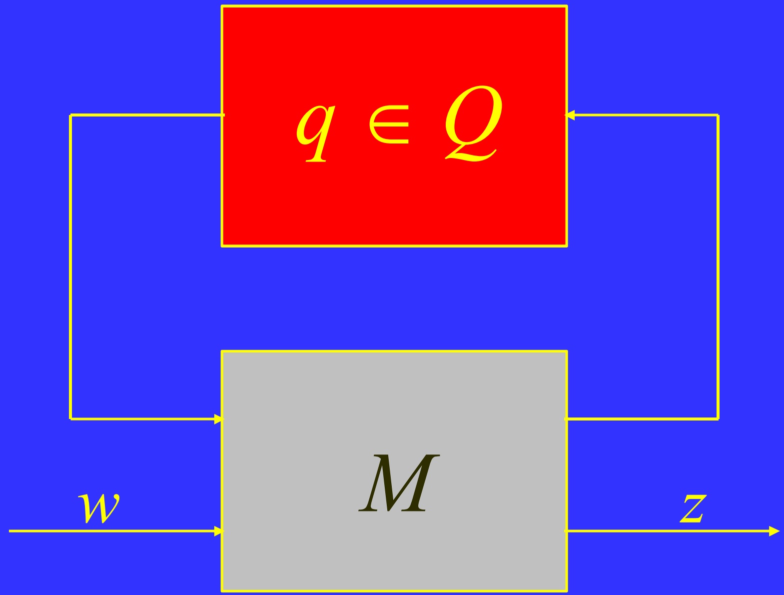
- **Objective:** Develop mathematically rigorous methods, not straightforward use of Monte Carlo simulations

Analysis Paradigm: Understanding Phenomena

Probabilistic Analysis

IEIIT-CNR

Probabilistic Analysis Paradigm



q represents random *uncertainty* bounded in a set Q

M is the known part of the system

w and z are disturbances and errors

IEIIT-CNR

Example[1]: Measure of Performance - 1

- Consider the linear system

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -a_0 & -a_1 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} w \qquad z = \begin{bmatrix} 1 & 0 \end{bmatrix} x$$

with parameters

$$a_0 = 1 + q_0$$

$$a_1 = 0.8 + q_1$$

and bounding set $Q = \{q = [q_0\ q_1]^T : \|q\|_\infty \leq \rho\}$

[1] G. Calafiore, F. Dabbene, R. Tempo (2010)

Example: Measure of of Performance - 2

- Compute the peak of the modulus of the frequency response on the *w*-*z* channel

$$z = G(s,q)\ w$$

- If the system is stable, this peak is given by the $\mathcal{H}_\infty$ norm of the transfer function

$$||G(s,q)||_\infty = \sup_\omega |G(j\omega,q)|$$

IEIIT-CNR

Example: Measure of Performance - 3

- Given a performance level γ, the **objective** is to compute the maximal radius $\bar{\rho}$ of Q such that

$$G(s,q) \text{ is stable and } \|G(s,q)\|_\infty \leq \gamma$$

for all $q \in Q$

- Let $\gamma = \sqrt{2}$
- Then $G(s,q)$ is stable and $\|G(s,q)\|_\infty \leq \gamma$ if and only if

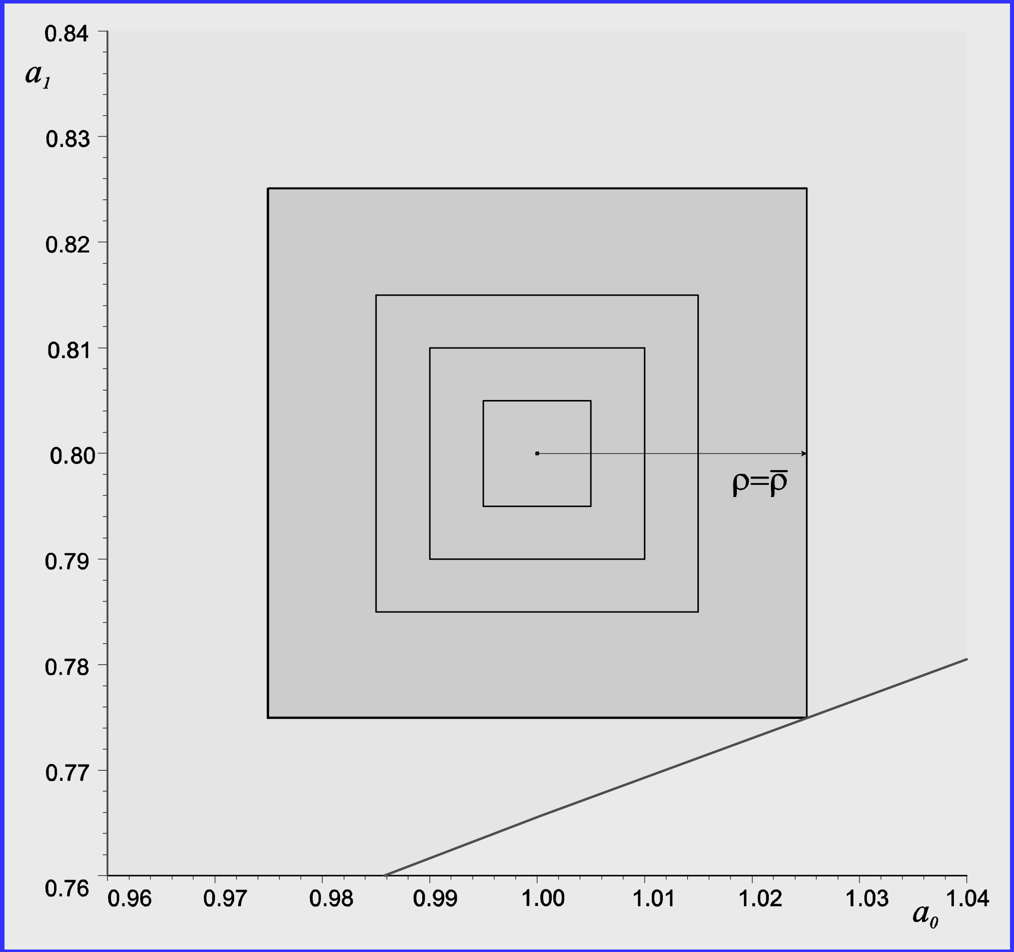
$$\rho < 0.8 \quad \text{and} \quad \frac{(0.8-\rho)^2}{2-\sqrt{2}} > 1+\rho$$

Example: Measure of Performance - 4

IEIIT-CNR

Largest radius of Q is
$\overline{\rho} = 0.025$

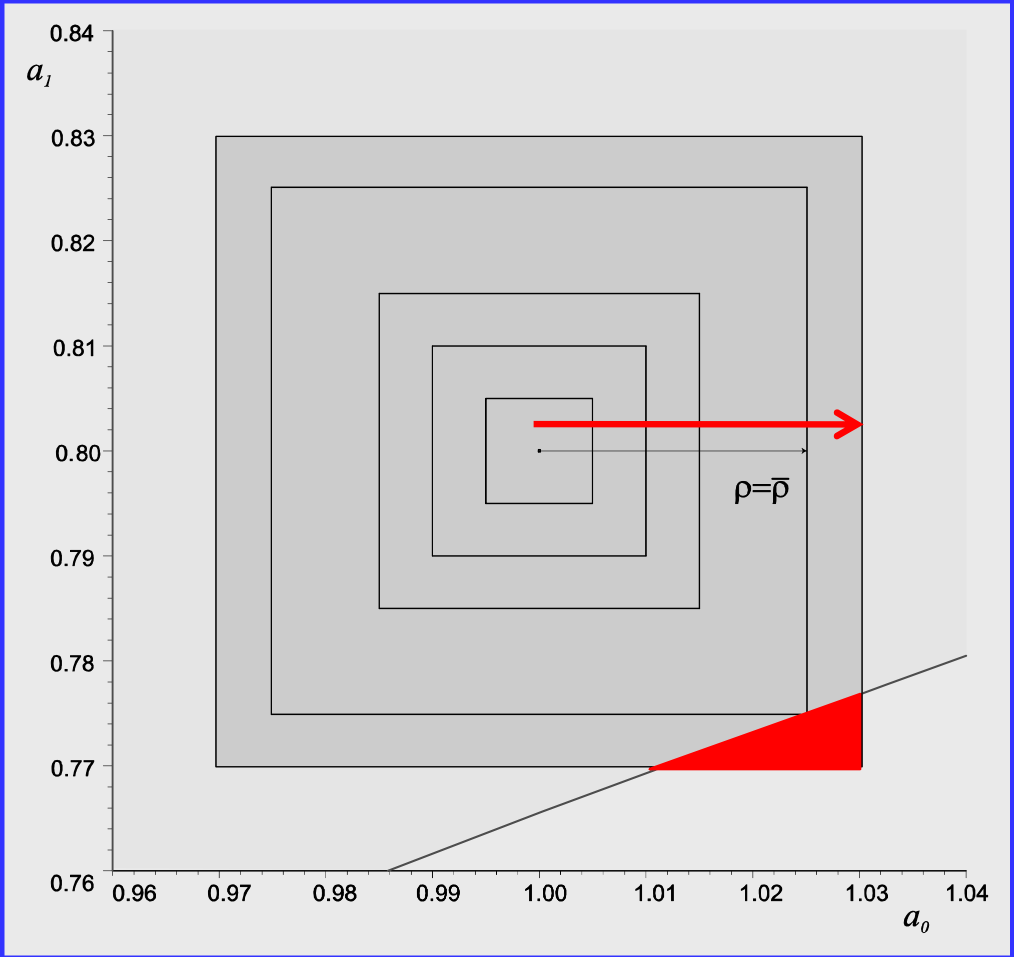
Conclusion: Stability and performance are satisfied for all $q \in Q$ with radius $\overline{\rho}$



IEIIT-CNR

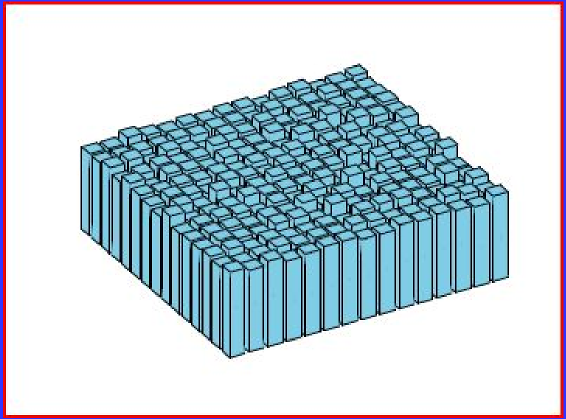
Example: Measure of Performance - 5

Observation: If we allow a small violation we may increase the radius ρ significantly



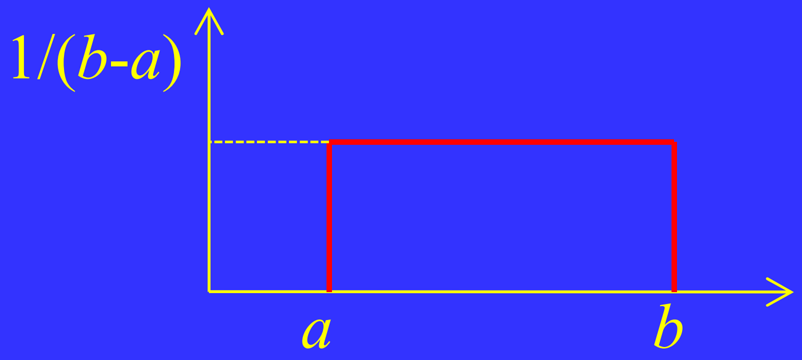
Probabilistic Model of Uncertainty

- Probability density function associated to Q
- Assume that q is a random vector/matrix with given density function and support Q
- **Example:** Uniform density $\mathcal{U}[Q]$ within Q



Uniform Density $\mathcal{U}[Q]$

- Univariate uniform density



- Multivariate uniform density

$$\mathcal{U}[Q] = \begin{cases} \dfrac{1}{vol(Q)} & \text{if } q \in Q \\ 0 & \text{otherwise} \end{cases}$$

IEIIT-CNR

Probability of Performance

- Define a performance function

$$J(q): Q \rightarrow \mathbf{R}$$

- Given a level γ, the probability of performance is

$$\text{Prob}\{q \in Q: J(q) \leq \gamma\}$$

- **Example:** If $G(s,q)$ is stable and $J(q) = \|G(s,q)\|_\infty$

$$\text{Prob}\{q \in Q: J(q) \leq \gamma\} = \text{Prob}\{q \in Q: \|G(s,q)\|_\infty \leq \gamma\}$$

Reliability and Measure of Violation

- *Reliability* (probability of performance) is denoted as

$$R = \mathrm{Prob}\{q \in Q : J(q) \leq \gamma\}$$

- We also define the measure of *violation*

$$V = 1 - \mathrm{Prob}\{q \in Q : J(q) \leq \gamma\} = \mathrm{Prob}\{q \in Q : J(q) > \gamma\}$$

- **Objective:** Achieve a small measure of violation

$$V \leq \varepsilon$$

where $\varepsilon \in (0,1)$ is the *accuracy*

Computation of Reliability and Violation

- Computing R and V requires to solve a difficult integration problem
- Taking uniform density $\mathcal{U}[Q]$

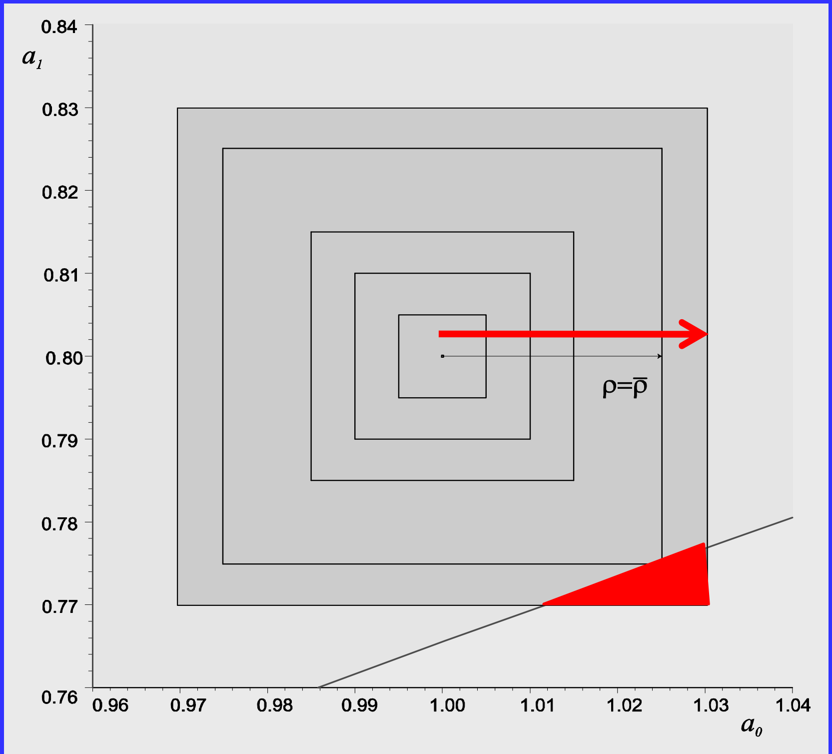
$$R = \text{Prob}\{q \in Q : J(q) \le \gamma\} = \frac{\int_{J(q)\le\gamma} \mathrm{d}\,q}{vol(Q)}$$

- In some special cases we can easily compute violation and reliability

IEIIT-CNR

Example: Measure of Performance - 6

- Take uniform pdf in Q
- Allowing a 5% violation, we increase ρ of 54% obtaining 0.038 (instead of 0.025)

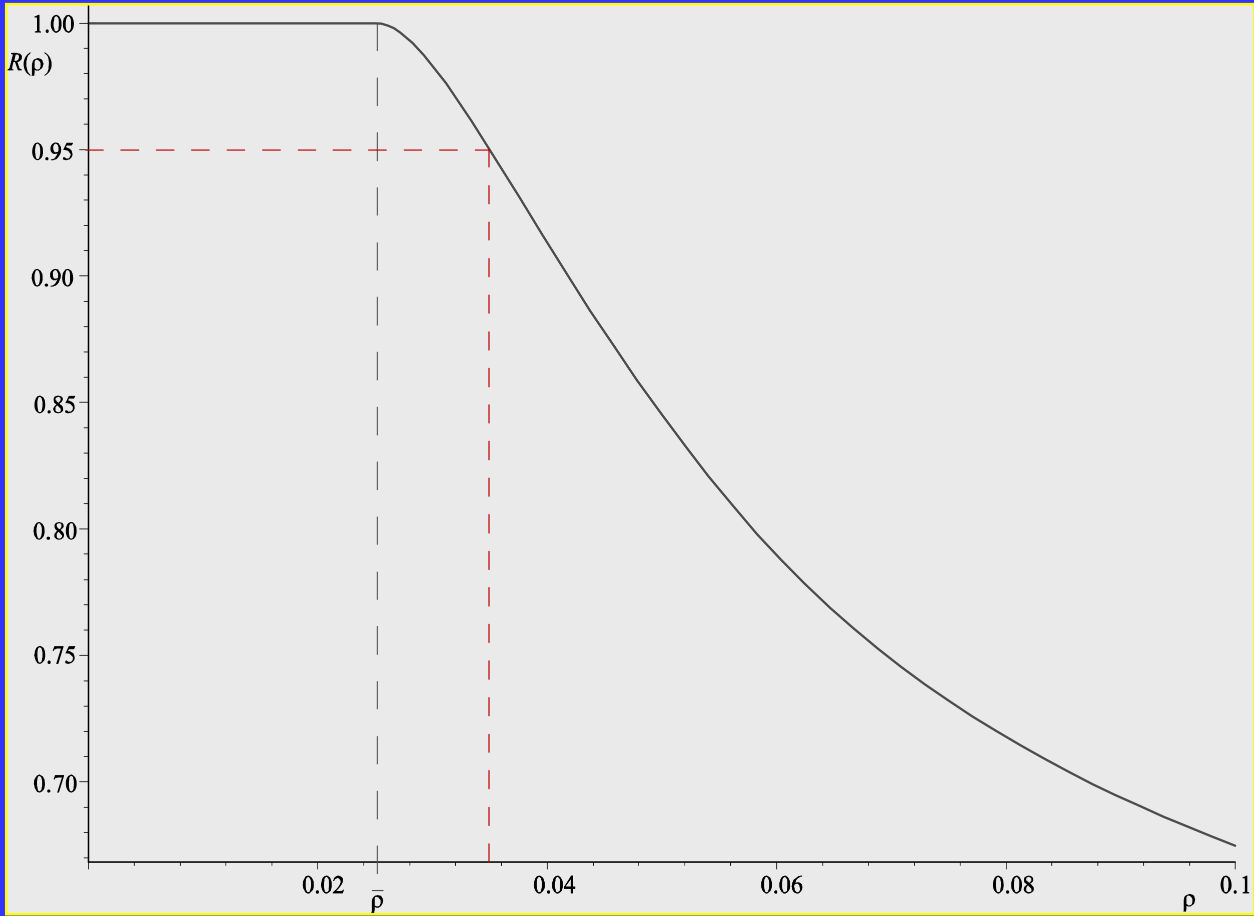


- For several values of ρ we compute R obtaining the *probability degradation function*

Probability Degradation Function

Reliability and violation: If a 5% violation is allowed we increase ρ of 54%

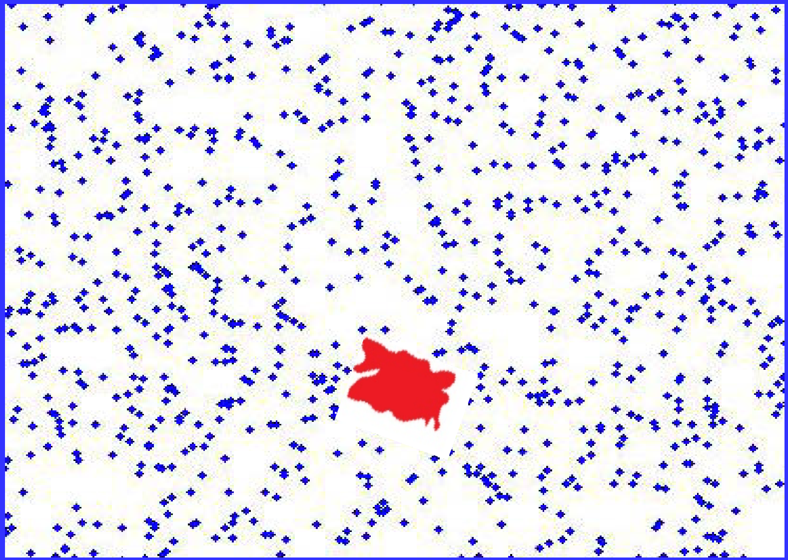
Obtain a radius 0.038 compared to $\bar{\rho} = 0.025$



Computation of Reliability and Violation

IEIIT-CNR

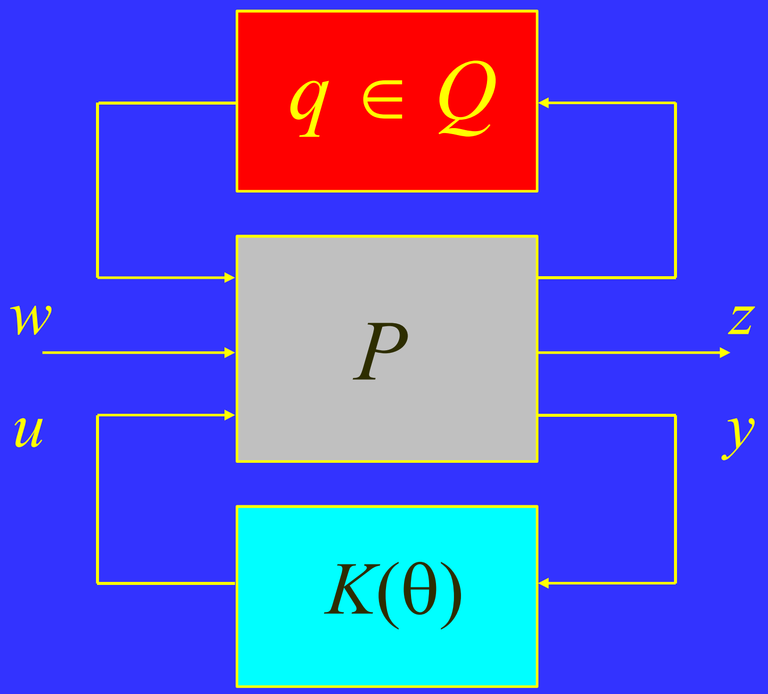
- In very special cases we compute *R* and *V* in closed form
- In general we need to solve a difficult integration problem
- Use randomized algorithms to determine probabilistic *estimates* of *R* and *V*
- Simulation-based approach
- Monte Carlo or Las Vegas
- Sample complexity
- Design a controller



Synthesis Paradigm: Realization of Desired Behavior

Probabilistic Synthesis

Probabilistic Synthesis Paradigm



- Design the parameterized controller $K(\theta)$ to guarantee stability and performance

IEIIT-CNR

Synthesis Performance Function

- Study parameterized controller $K(\theta)$ where $\theta \in \Theta$ are the controller parameters to be determined and Θ is their bounding set
- Study a *synthesis* performance function

$$J = J(\theta, q)$$

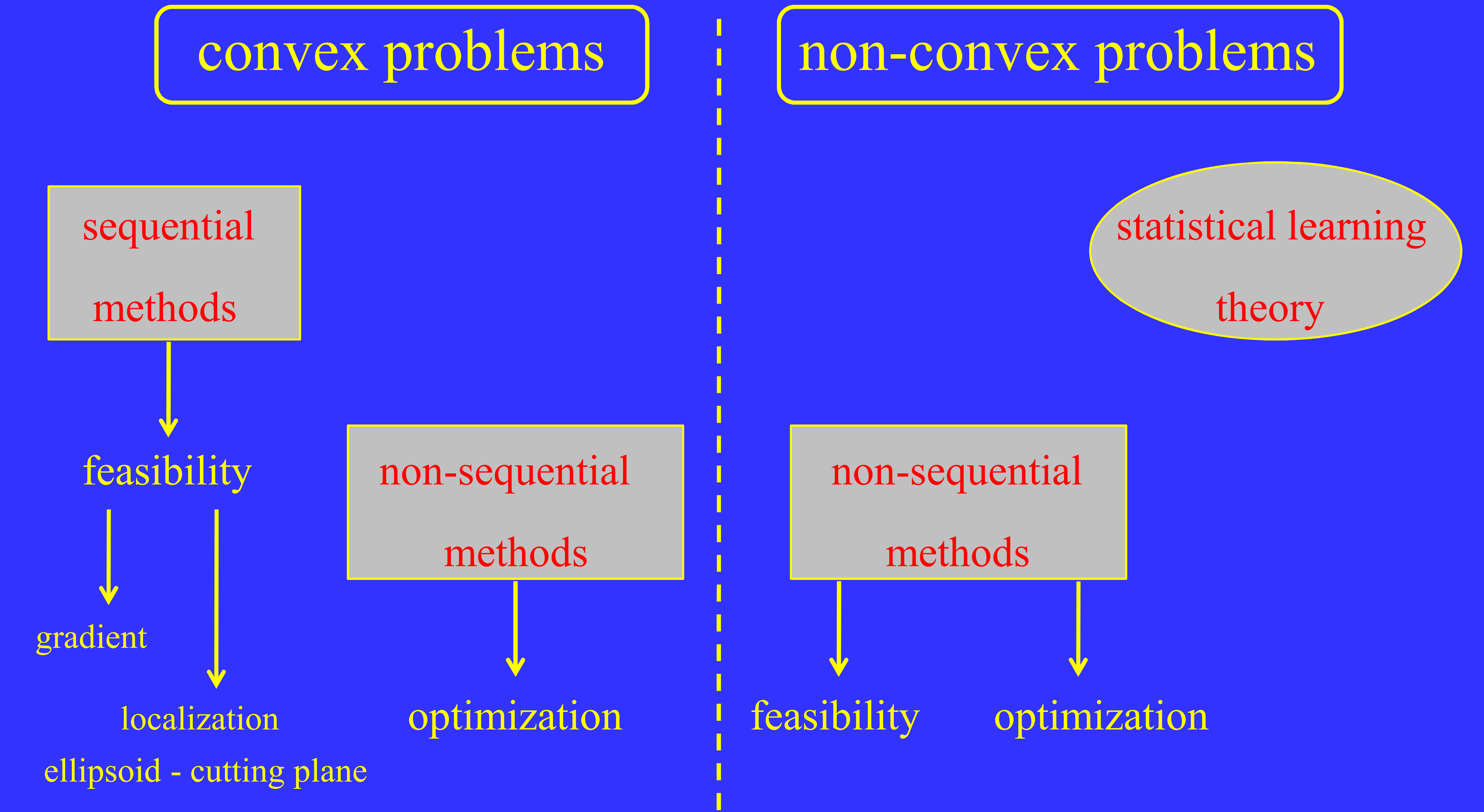
representing system constraints

- Replace

$J(q) \leftrightarrow J(\theta, q)$ | $R \leftrightarrow R(\theta)$ | $V \leftrightarrow V(\theta)$

IEIIT-CNR

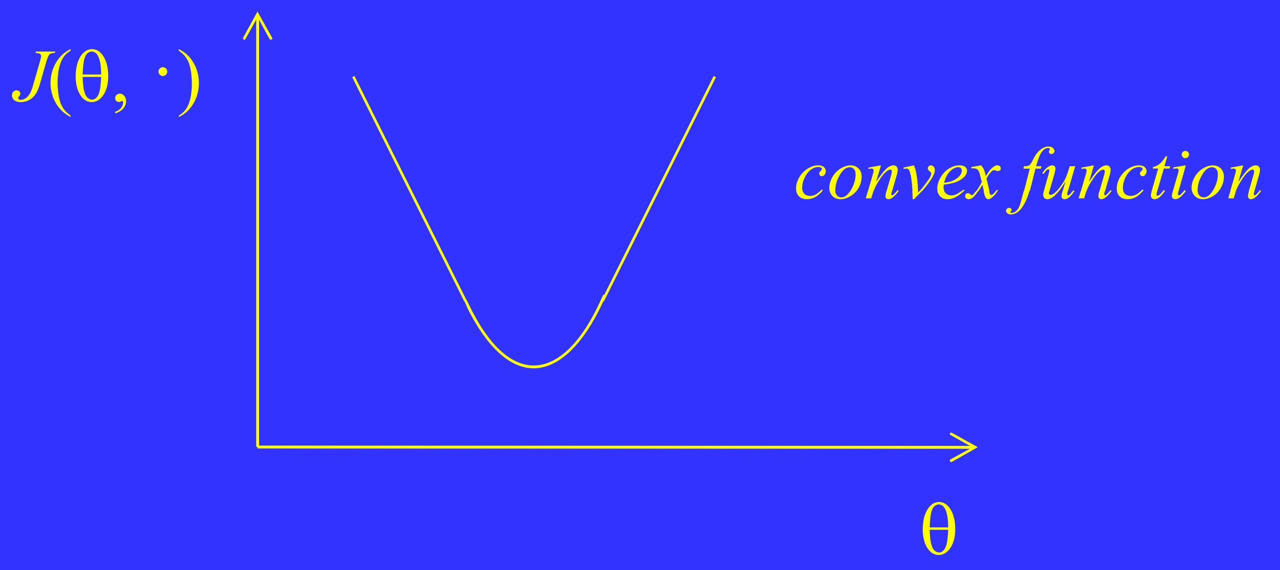
Probabilistic Design Methods: The Big Picture



IEIIT-CNR

Convexity Assumption for Design Parameters

- **Convexity:** The function $J(\theta, q)$ is convex in θ for any fixed value of $q \in Q$



- The function $J(\theta, q)$ is measurable in q for any fixed value of θ

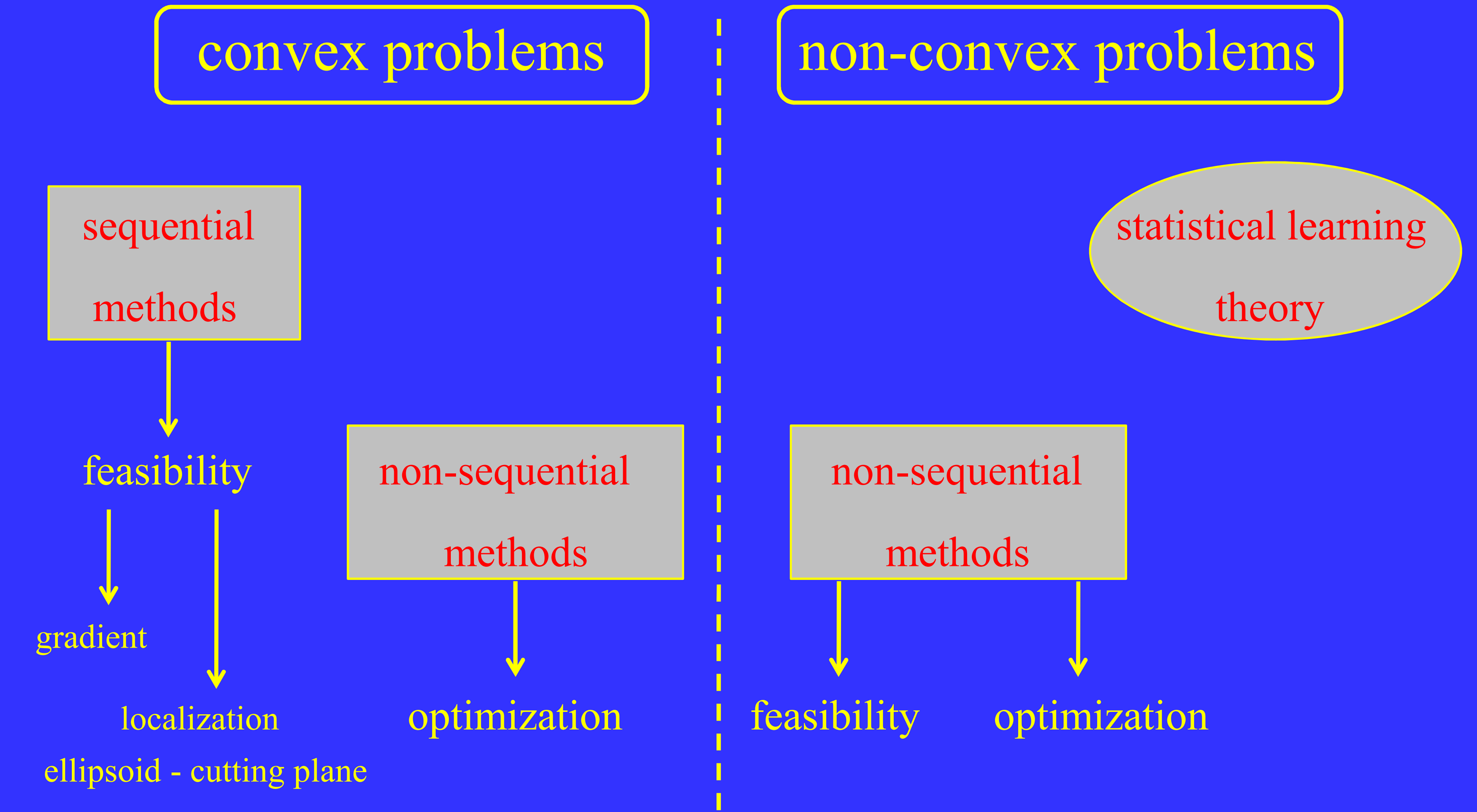
Convex Functions and LQ Regulators

- Examples of convex functions arise when considering various control problems, such as design of LQ regulators in the presence of uncertainty $q \in Q$

Sequential Methods for Convex Problems

Probabilistic Design Methods: The Big Picture

IEIIT-CNR



Sequential Methods for Design

- Objective is to determine θ satisfying the uncertain inequality

$$J(\theta, q) \leq 0$$

with some probability

- We study randomized sequential methods for finding a *probabilistic feasible* solution θ

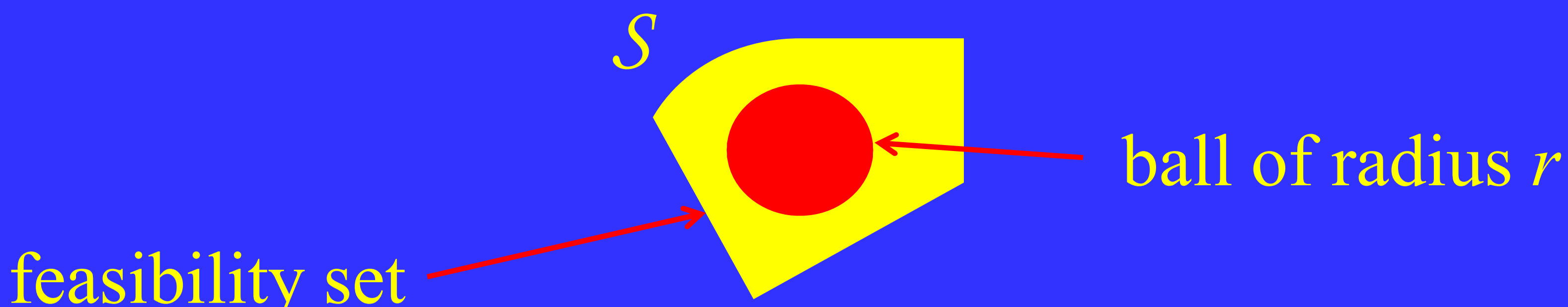
IEIIT-CNR

Definition of *r*-feasibility

- ***r*-feasibility**: For given $r > 0$, we say that $J(\theta, q) \leq 0$ is *r*-feasible if the set

$$S = \{\theta : J(\theta, q) \leq 0 \text{ for all } q \in Q\}$$

contains a (full-dimensional) ball of radius *r*



IEIIT-CNR

Controller Reliability

- Given probabilistic *accuracy* $\varepsilon \in (0,1)$, we search for controller θ such that

$$R(\theta) = \mathrm{Prob}\{q \in Q: J(\theta, q) \leq 0\} > 1 - \varepsilon$$

- Since this probability is not easily computable we need randomization to obtain an estimate $\hat{R}(\theta)$
- We introduce the *confidence* $\delta \in (0,1)$ which measures the probability of the event

$$\left| R(\theta) - \hat{R}(\theta) \right| \leq \varepsilon$$

IEIIT-CNR

Probabilistic Feasible Solution

- The probability of violation of the controller θ is

$$V(\theta) = \mathrm{Prob}\{q \in Q: J(\theta, q) > 0\}$$

- Find θ such that the probability of violation is small

$$V(\theta) < \varepsilon$$

- If such θ exists in the feasible set $\mathcal{S}$ we have a *probabilistic feasible* solution
- Equivalently we obtain a probabilistic robust design

IEIIT-CNR

Sequential Methods for Design[1]

- Randomized sequential algorithms for finding a probabilistic feasible solution θ are based on two fundamental ingredients

  i) Oracle checking probabilistic feasibility of a candidate solution

  ii) Update rule exploiting convexity to construct a new candidate solution based on the oracle outcome

[1] G. Calafiore, F. Dabbene, R. Tempo (2010)

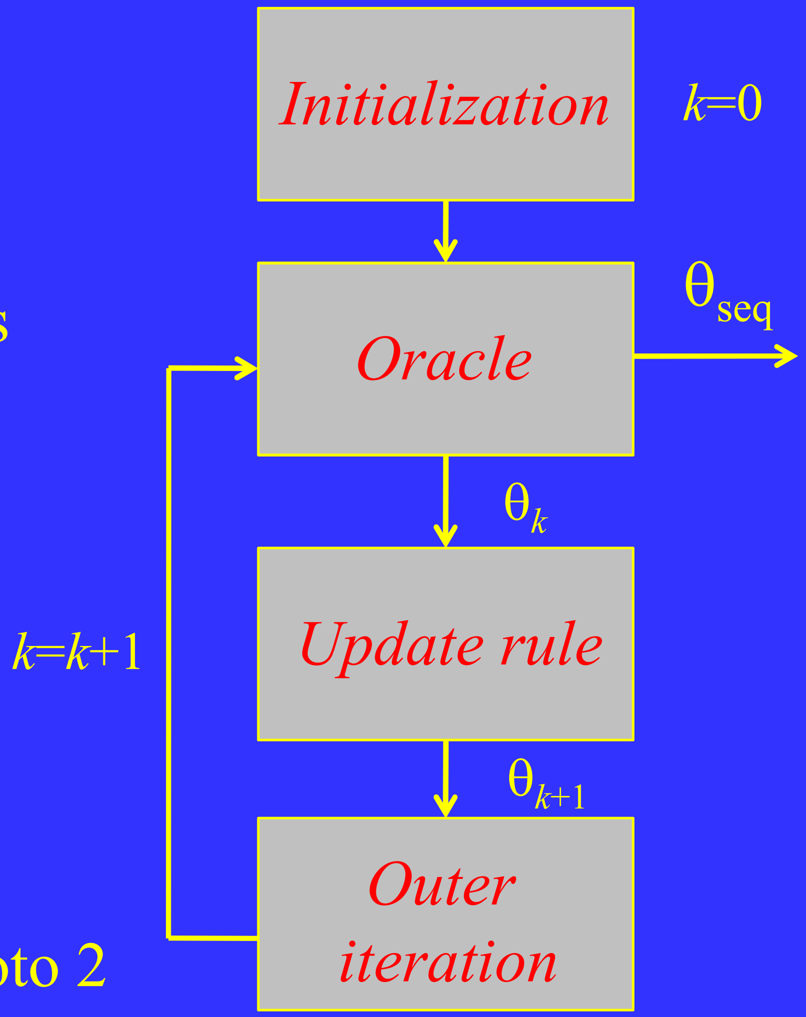
Meta-Algorithm

1. Initialization: set $k = 0$; choose θ_0

2. Oracle: return *true* if θ_k is probabilistic feasible; Exit returns $\theta_{seq} = \theta_k$
   Otherwise, return *false* and violation certificate

3. Update rule: Construct θ_{k+1} based on θ_k and on q_k

4. Outer iteration: Set $k=k+1$ and Goto 2



Probabilistic Oracle

- Oracle is the randomized part of the algorithm and decides probabilistic feasibility of the current solution
- Generate N_k i.i.d. samples of q within Q (multisample)

$$q^{(1)}, \ldots , q^{(N_k)} \in Q$$

- The candidate solution θ_k is *probabilistic feasible* if

$$J(\theta_k, q^{(i)}) \leq 0$$

for all $i = 1, \ldots , N_k$

- Otherwise if $J(\theta_k, q^{(i)}) > 0$ we set $q_k = q^{(i)}$

IEIIT-CNR

Oracle (Inner) Iterations

- Consider the multisample size[1]

$$N_k \geq N_{oracle} = \left\lceil \frac{\log \frac{\pi^2 (k+1)^2}{6\delta}}{\log \frac{1}{1-\varepsilon}} \right\rceil$$

where ε , $\delta \in (0,1)$ are *accuracy* and *confidence*

- N_k is the number of Oracle (inner) iterations

- Slightly better bound may be obtained using the Riemann function

[1] Y. Oishi (2007)

IEIIT-CNR

Update Rule: Gradient Method

- Update rule is a classical gradient step

$$\theta_{k+1} = \begin{cases} \theta_k - \eta_k \dfrac{\partial_k(\theta_k)}{\|\partial_k(\theta_k)\|} & \text{if } \partial_k(\theta_k) \neq 0 \\ \theta_k & \text{otherwise} \end{cases}$$

- Let $\alpha > 0$, then the stepsize η_k is given by

$$\eta_k = \begin{cases} \dfrac{J(\theta_k, q_k)}{\|\partial_k(\theta_k)\|} + \alpha & \text{if } \partial_k(\theta_k) \neq 0 \\ 0 & \text{otherwise} \end{cases}$$

IEIIT-CNR

Subgradient for LMIs

- For LMIs, find θ such that

$$F(\theta, q) = F_0(q) + \theta_1 F_1(q) + \ldots + \theta_n F_n(q) \leq 0$$

for all $q \in Q$ where $F_i(q)$ are real symmetric matrices depending (nonlinearly) on q

- A subgradient of the function

$$J(\theta, q_k) = \lambda_{\max} F(\theta, q_k)$$

can be readily computed obtaining

$$\partial_k(\theta_k) = [\xi_{\max}{}^T F_1(q_k)\, \xi_{\max} \cdots \xi_{\max}{}^T F_n(q_k)]^T$$

where $\xi_{\max}$ is a unit norm eigenvector corresponding to the largest eigenvalue of $F(\theta_k, q_k)$

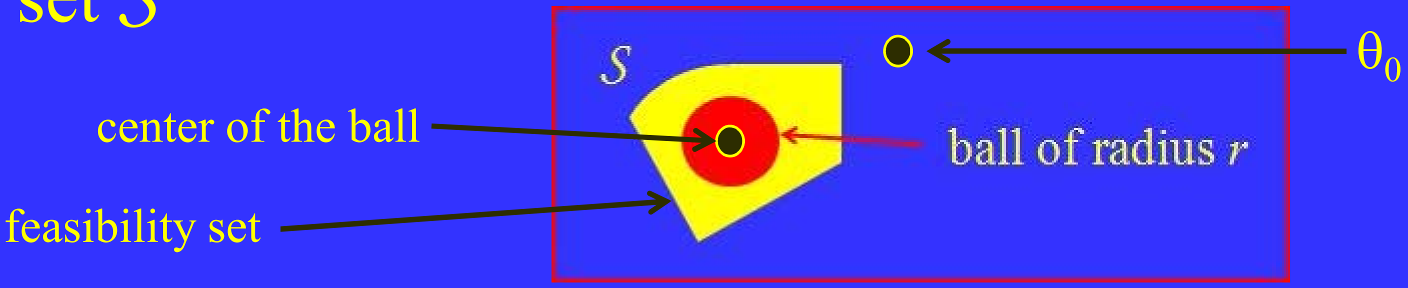
IEIIT-CNR

Outer Iterations

- Define

$$N_{outer} = \left\lceil \frac{D^2}{r^2} \right\rceil$$

where D is the distance between the initial solution θ_0 and the center of a ball of radius r contained in the solution set S



- r is imposed by the desired radius of feasibility
- If D is unknown, then we replace it with an upper bound which can be easily estimated

Successful/Unsuccessful Exit

- *Successful* exit: The algorithms returns a probabilistic controller θ_{seq}
- *Unsuccessful* exit: No solution has been found in N_{outer} iterations
- We have a *certificate of violation* q_k returned by the Oracle showing that the problem is not r-feasible

IEIIT-CNR

Probabilistic Properties of the Algorithm

- **Theorem**[1]

  Let Convexity Assumption hold and let $\varepsilon, \delta \in (0,1)$

- The probability that the algorithm terminates at some outer iteration $k < N_{\text{outer}}$ returning θ_{seq} having large violation (i.e. $V(\theta_{\text{seq}}) > \varepsilon$) is less than δ
- If the algorithm reaches the outer iteration N_{outer} then the problem is not r-feasible

[1] F. Dabbene and R. Tempo (2010)

Remarks

- **Remarks**
- Emphasis on finite termination criterion (key difference with classical stochastic approximation algorithms)
- Explicit use of convexity
- Closed-form computation of the gradient

IEIIT-CNR

Advanced Techniques for Update Rule

- More advanced techniques falling in the class of localization methods can be used instead of gradient update
- In probabilistic *cutting plane* methods the localization set is a polytope and the update rule computes the analytic center
- In the probabilistic *ellipsoid algorithm* the localization set is an ellipsoid and the update rule computes the center of the ellipsoid

Discussions and Related Literature

IEIIT-CNR

Research on Sequential Methods - 1

- Sequential approach for designing uncertain LQ regulators (gradient Update Rule)[1]
- Extensions to uncertain LMIs (gradient Update Rule)[2]
- Ellipsoid Update Rule[3]
- Design of LPV systems (multiple gradient rules)[4]

[1] B.T. Polyak and R. Tempo (2001)
[2] G. Calafiore and B.T. Polyak (2001)
[3] S. Kanev, B. De Schutter and M. Verhaegen (2003)
[4] Y. Fujisaki, F. Dabbene and R. Tempo (2003)

IEIIT-CNR

Research on Sequential Methods - 2

- Design of common Lyapunov functions for switched systems[1]
- From common to piecewise Lyapunov functions[2]
- Sample size of inner iteration[3,5]
- Cutting plane and localization methods[4]

[1] D. Liberzon and R. Tempo (2004)
[2] H. Ishii, T. Basar and R. Tempo (2005)
[3] Y. Oishi (2007)
[4] G. Calafiore and F. Dabbene (2007)
[5] Y. Fujisaki and Y. Oishi (2008)

A Posteriori Analysis

Controller Reliability

- Need to evaluate a posteriori the goodness of the probabilistic controller θ_{seq} we have obtained
- Reliability for the controller $K(\theta_{seq})$ is

$$R(\theta_{seq}) = \text{Prob}\{q \in Q: J(\theta_{seq}, q) \leq 0\} = 1 - V(\theta_{seq})$$

- Computing $R(\theta_{seq})$ exactly requires to solve a difficult integration problem
- Compute a probabilistic estimate of reliability setting a simple Monte Carlo experiment

Monte Carlo Experiment

- Take N (sample complexity) i.i.d. random samples of q according to the given probability measure

$$q^{(1)}, q^{(2)}, \ldots, q^{(N)} \in Q$$

- Given controller θ_{seq} evaluate

$$J(\theta_{\text{seq}}, q^{(1)}), J(\theta_{\text{seq}}, q^{(2)}), \ldots, J(\theta_{\text{seq}}, q^{(N)})$$

IEIIT-CNR

Estimated Probability of Reliability

- Given controller θ_{seq} construct a probabilistic estimate of reliability

$$\hat{R}_N(\theta_{seq}) = \frac{1}{N}\sum_{i=1}^{N} \mathbf{I}\left(J(\theta_{seq}, q^{(i)})\right)$$

where $\mathbf{I}(\cdot)$ denotes the indicator function

$$I(J(\theta_{seq}, q^{(i)})) = \begin{cases} 1 & \text{if } J(\theta_{seq}, q^{(i)}) \le \gamma \\ 0 & \text{otherwise} \end{cases}$$

Law of Large Numbers

- Monte Carlo analysis (Law of Large Numbers) studies the *sample complexity* such that for *fixed* θ_{seq} the probability inequality

$$\left| R(\theta_{seq}) - \hat{R}_N(\theta_{seq}) \right| \leq \varepsilon$$

holds with probability at least 1- δ

(Additive) Chernoff Bound[1]

- For fixed θ_{seq}, given $\varepsilon, \delta \in (0,1)$, if

$$N \geq N_{ch} = \left\lceil \frac{\log \frac{2}{\delta}}{2\varepsilon^2} \right\rceil$$

then the probability inequality

$$\left| R(\theta_{seq}) - \hat{R}_N(\theta_{seq}) \right| \leq \varepsilon$$

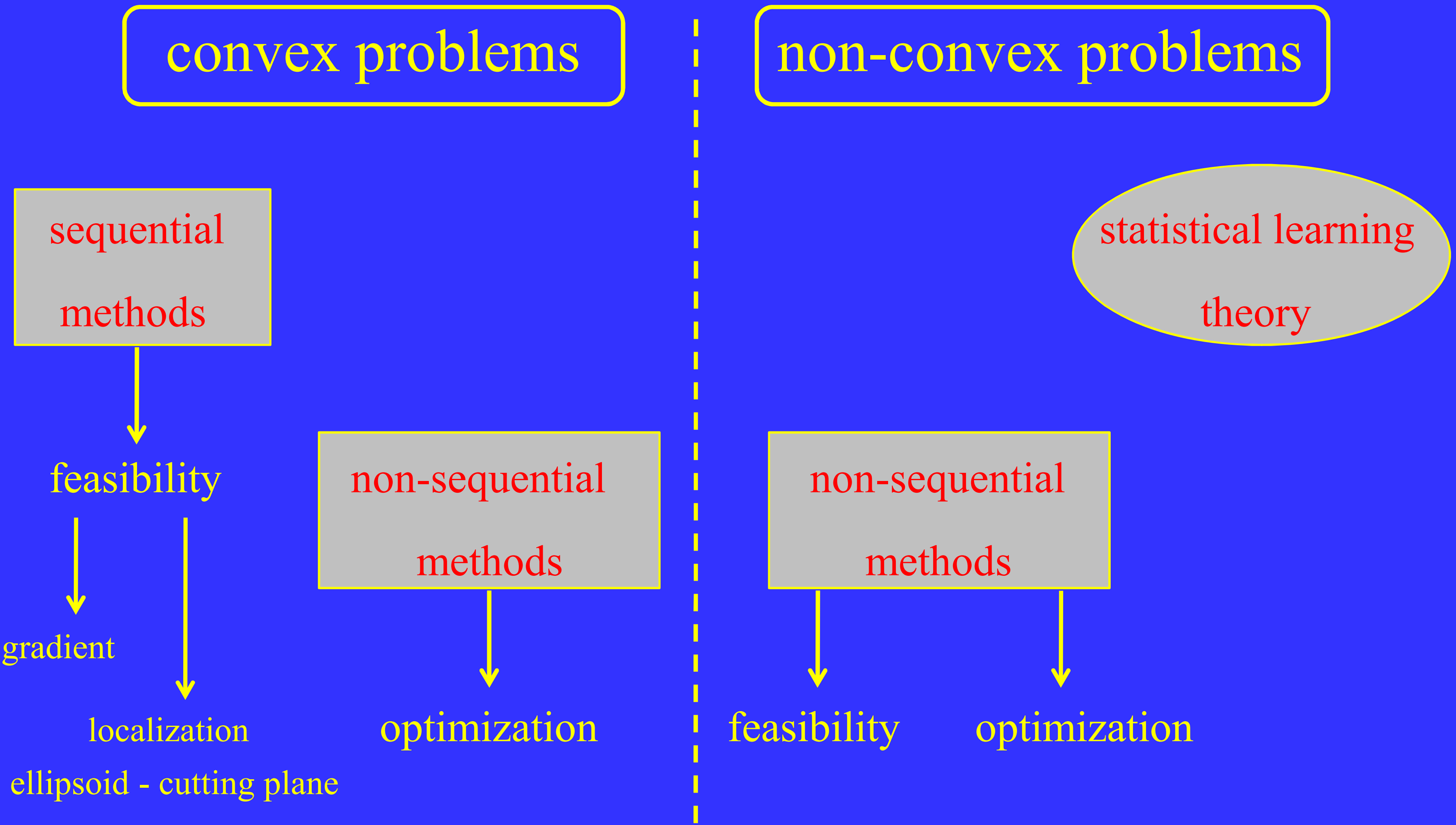
holds with probability at least 1- δ

[1] H. Chernoff (1952)

Statistical Learning Theory

IEIIT-CNR

Probabilistic Design Methods: The Big Picture



Statistical Learning Theory for Control Design of Uncertain Systems

- Statistical learning theory is a branch of the theory of empirical processes
- Significant results and applications have been obtained in various areas, including neural networks, system identification, pattern recognition, …
- We study statistical learning theory for control design of uncertain systems

IEIIT-CNR

Statistical Learning Theory

- Main objective is to derive uniform convergence laws (for all controller parameters) and the sample complexity
- Powerful methodology for control synthesis (feasibility and optimization) which is not based upon a convexity assumption on the controller parameters
- The sample complexity is larger[3] than that derived in the convex case for the scenario approach[1,2]

[1] G. Calafiore, M. C. Campi (2006)
[2] M. C. Campi, S. Garatti (2008)
[3] T. Alamo, R. Tempo and E.F. Camacho (2009)

IEIIT-CNR

Uniform Convergence Law

- Statistical learning theory studies the sample complexity such that the probability inequality

$$\left| R(\theta) - \hat{R}_N(\theta) \right| \leq \varepsilon$$

holds *uniformly* for all θ with probability at least 1- δ

- Recall that Monte Carlo analysis deals with *fixed controller* θ or with *finite families* of controllers

IEIIT-CNR

RACT
Randomized Algorithms Control Toolbox

http://ract.sourceforge.net

IEIIT-CNR

RACT

- **RACT: Randomized Algorithms Control Toolbox** for Matlab
- RACT has been developed at IEIIT-CNR and at the Institute for Control Sciences-RAS, based on a bilateral international project
- Members of the project

Andrey Tremba (Main Developer and Maintainer)
Giuseppe Calafiore
Fabrizio Dabbene
Elena Gryazina
Boris Polyak (Co-Principal Investigator)
Pavel Shcherbakov
Roberto Tempo (Co-Principal Investigator)

IEIIT-CNR

RACT

- Study a variety of *uncertain objects*: scalar, vector and matrix uncertainties, with different pdfs
- Easy and fast *sampling* of uncertain objects of almost any type
- *Sequential randomized algorithms* for feasibility of uncertain LMIs using stochastic gradient and localization methods (ellipsoid or cutting plane)
- *Non-sequential randomized algorithms* for optimization of convex problems

- Under construction: Non-sequential randomized algorithms for feasibility and optimization of non-convex problems

RACT

- RACT: Randomized Algorithms Control Toolbox for Matlab

http://ract.sourceforge.net

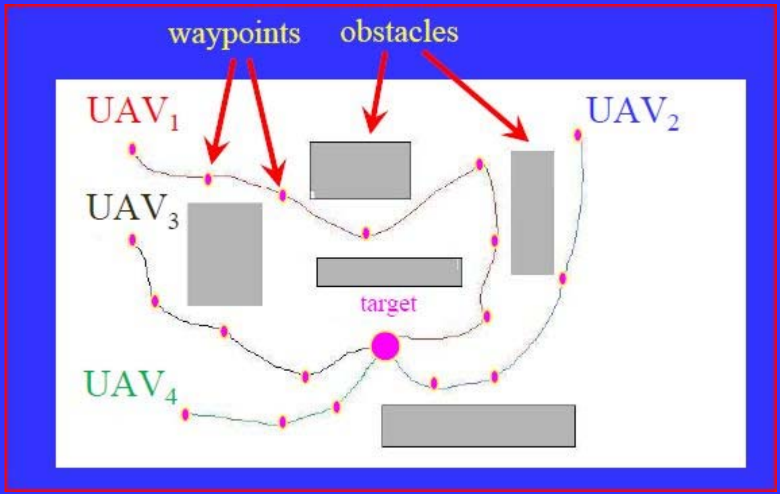
Randomized Algorithms for Systems and Control Applications

Randomized Algorithms for Systems and Control Applications - 1

IEIIT-CNR

- *Aerospace control and unmanned aerial vehicles* (UAVs)[1,2,3]

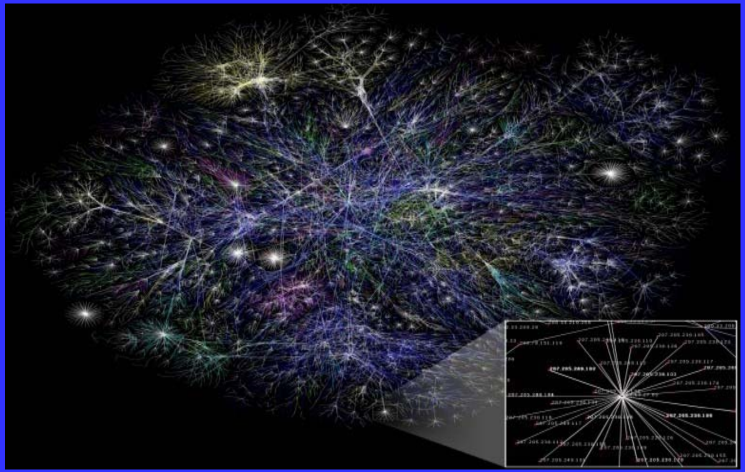
- *Multi-agent systems and consensus*[4,5]



[1] C.I. Marrison and R.F. Stengel (1998)
[2] B. Lu and F. Wu (2006)
[3] L. Lorefice, B. Pralio and R. Tempo (2009)
[4] H. Ishii and R. Tempo (2010)
[5] L. Pallottino, V.G. Scordio, E. Frazzoli and A. Bicchi (2007)

IEIIT-CNR

Randomized Algorithms for Systems and Control Applications - 2



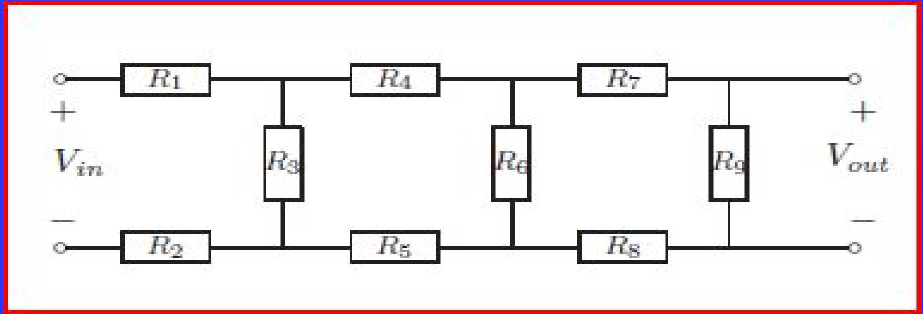
- *Network congestion control*[1]
- *Quantized and switched systems*[2-3]
- *Fault detection, isolation, vision-based control*[4-5]

[1] T. Alpcan, T. Basar and R. Tempo (2005)
[2] H. Ishii, T. Basar and R. Tempo (2004)
[3] H. Ishii, T. Basar and R. Tempo (2005)
[4] S. Kanev and M. Vehaegen (2006)
[5] W. Ma, M. Sznaier and C.M. Lagoa (2007)

IEIIT-CNR

Randomized Algorithms for Systems and Control Applications - 3

- *Embedded and electric circuits*[1,2]



- *Advanced driver assistance systems*[3]

[1] C. Alippi (2002)
[2] C.M. Lagoa, F. Dabbene and R. Tempo (2008)
[3] O.J. Gietelink, B. De Schutter, and M. Verhaegen (2005)

Main References

- R. Tempo, G. Calafiore and F. Dabbene, "Randomized Algorithms for Analysis and Control of Uncertain Systems," *Springer-Verlag*, London, 2005
- F. Dabbene and R. Tempo, "Probabilistic and Randomized Tools for Control Design," The Control Handbook (W. S. Levine Ed.), *Taylor & Francis*, 2010 (to appear)

http://staff.polito.it/roberto.tempo/

Conclusions

Randomized Algorithms: A Success Story

Randomized algorithms are a success story for systems and control