システムと制御のための
決定論的/ランダマイズド混合アルゴリズム
– 恩恵と罠 –

ロベルト　テンポ

IEIIT-CNR
トリノ工科大学，イタリア
roberto.tempo@polito.it

Deterministic and Randomized Algorithms for Systems and Control: Benefits and Pitfalls

Roberto Tempo

IEIIT-CNR
Politecnico di Torino, Italy
roberto.tempo@polito.it

Overview

- Preliminaries
- Randomized Algorithms
- One-in-a-Box Problem
- Deterministic and Randomized Algorithms
- Fixed Order Controller Design
- Some Recent Research Directions
- Conclusions: Benefits or Pitfalls?

Preliminaries

© RT - 2006

IEIIT-CNR

Computationally Intractable Problems

- Many control problems are computationally intractable
- This is generally meant for *NP*-hard problems

- Examples:
  - fixed order controller design
  - static output feedback
  - stability of interval matrices
  - μ computation

© RT - 2006

IEIIT-CNR

Deterministic Models

- Deterministic (worst case) models can be successfully used only for computationally tractable problems
- Example: Convex problems
- Otherwise, the goal is to find local solutions instead of global solutions (non-smooth optimization)
- Use of relaxation, overbounding methods, SOS, and other techniques

IEIIT-CNR

Probabilistic Models

- Probabilistic models naturally lead to Randomized Algorithms (RAs) which are used in various areas

1. Combinatorial optimization, computational geometry
2. Data structuring, search trees, graph algorithms, sorting, …
3. Mathematics of finance: Computation of path integrals
4. Genomics: String matching and classification
5. Motion and path planning problems

© RT - 2006

Randomized Algorithms

- Remarkably, randomized methods and algorithms are not used *systematically* in systems and control
- Objective of this research is the development of mathematically rigorous methods, not straightforward use of Monte Carlo simulations

© RT - 2006

IEIIT-CNR

References

- "*Randomized Algorithms*" by R. Motwani and P. Raghavan, Cambridge University Press, 1995
- "*Randomized Algorithms for Analysis and Control of Uncertain Systems*" by R. Tempo, G. Calafiore and F. Dabbene, Springer-Verlag, 2005
- Additional documents, papers, MATLAB™ codes, etc, please consult

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

© RT - 2006

Randomized Algorithms

IEIIT-CNR

Monte Carlo Randomized Algorithm

- Definition of a randomized algorithm: An algorithm that makes random choices during execution to produce a result

- For hybrid systems, “random choices” could be switching between different states or logical operations
- In other situations, “random choices” may require random sample generation

Monte Carlo Randomized Algorithm

- Definition of a Monte Carlo randomized algorithm: A randomized algorithm that may produce incorrect results, but with bounded error probability

- There are other classes of randomized algorithms

© RT - 2006

IEIIT-CNR

Example: H_∞ Performance - 1

- Consider sensitivity function $S(s,\Delta)$ with random structured uncertainty Δ, bounding set B and weighting function $W(s)$
- H_∞ performance problem: Given a level $\gamma > 0$, check if

$$\| W(s)\, S(s,\Delta) \|_\infty \leq \gamma$$

for all $\Delta \in B$

- This is an uncertain decision problem

Example: H_∞ Problem - 2

- Monte Carlo RAs can be immediately constructed
  - Take N random samples Δ^i of Δ within B
  - Compute the empirical maximum

$$\max_{i=1,\ldots,N} \| W(s)\, S(s,\Delta^i) \|_\infty \le \gamma$$

- The algorithm may provide an incorrect answer
- The probability of error can be made arbitrarily small if N is sufficiently large (Law of Large Numbers)
- Specific bounds on N which make the probability of error smaller than a given level are available

One-in-a-Box Problem

© RT - 2006

Interval Polynomials

- An interval polynomial is

$$p(s,k)= k_0 + k_1 s + k_2 s^2 + \ldots + k_n s^n$$

where $k_i \in [k_i^-, k_i^+]$, $i = 0, 1, 2, \ldots, n$

- An interval polynomial is a box of polynomials, i.e., the coefficients $k \in K$ where

$$K = \{k\colon k_i \in [k_i^-, k_i^+],\ i = 0, 1, 2, \ldots, n\}$$

© RT - 2006

IEIIT-CNR

Robust Stability of Interval Polynomials

- **Kharitonov Theorem**[1]

The interval polynomial $p(s,k)$ is stable for all $k \in K$ if and only if the four Kharitonov polynomials

$$p_1(s) = k_0^+ + k_1^+ s + k_2^- s^2 + k_3^- s^3 + k_4^+ s^4 + \ldots$$

$$p_2(s) = k_0^- + k_1^- s + k_2^+ s^2 + k_3^+ s^3 + k_4^- s^4 + \ldots$$

$$p_3(s) = k_0^+ + k_1^- s + k_2^- s^2 + k_3^+ s^3 + k_4^- s^4 + \ldots$$

$$p_4(s) = k_0^- + k_1^+ s + k_2^+ s^2 + k_3^- s^3 + k_4^- s^4 + \ldots$$

are Hurwitz stable

[1] V.L. Kharitonov (1978)

IEIIT-CNR

One-in-a-Box Problem

- Design counterpart of Kharitonov problem: Does there exist a stable polynomial $p(s,k)$ in the interval family?
- One-in-a-box problem: Find $k \in K$ such that $p(s,k)$ is Hurwitz stable
- One-in-a box is a special case of fixed order stabilization and static output feedback problems
- Compute the volume of Hurwitz polynomials within K

© RT - 2006

Looking for a Needle in a Haystack

- In some cases, we can “easily” find a Hurwitz polynomial in the family
- In general, one-in-a-box problem is like looking for a needle in a haystack
- One-in-a-box is (probably) *NP*-hard, but no definitive answer is known

© RT - 2006

IEIIT-CNR

Volume of Stable Polynomials

- Consider an interval polynomial

$$p(s,k)= k_0 + k_1 s + k_2 s^2 + \dots + k_n s^n$$

where $k \in K$

- Volume V_{stab} of Hurwitz region

$$H = \{k \in K : p(s,k) \text{ is Hurwitz}\}$$

is defined as

$$V_{stab} = \int_H d\mu$$

where $d\mu$ is the Lebesgue measure

Volume of Stable Polynomials for Unit Box K

- Theorem 2[1]

For $n \geq 3$ and $K = [0,1]^{n+1}$, we have

$$V_{stab} \leq 1/((n+1)/2)!$$

In addition

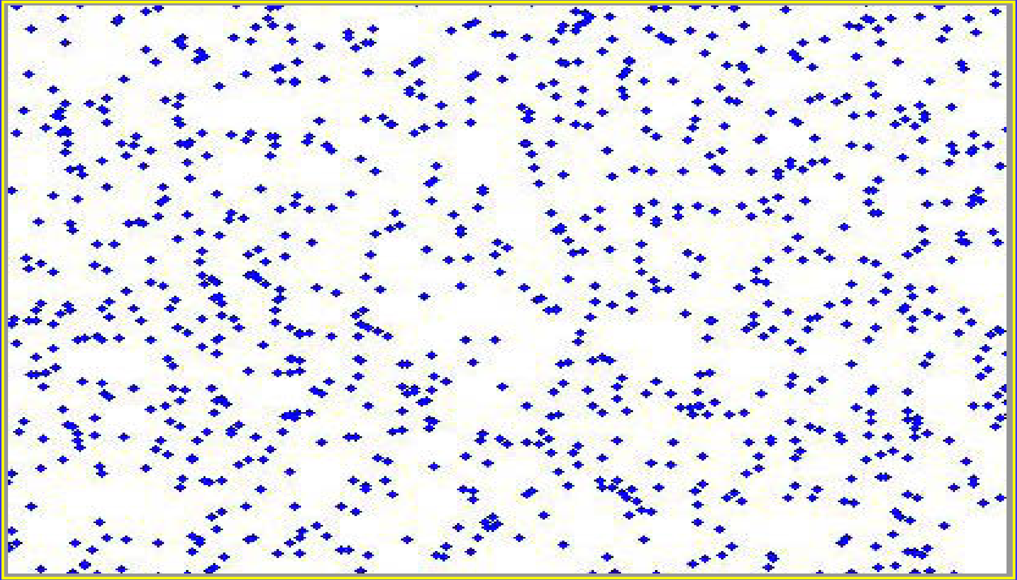
$$V_{stab} \approx e^{-n^2} \text{ for } n \to \infty$$

[1] A.S. Nemirovskii and B.T. Polyak (1994)

Deterministic Approach

In general no conclusion can be drawn from vertices

Probabilistic Approach



Taking (uniform) random samples in *K*, we could miss the Hurwitz region (with high probability)

Deterministic and Randomized Algorithms

© RT - 2006

IEIIT-CNR

Even/Odd Polynomial Coefficients

even coefficients

odd coefficients

- Consider a polynomial

$$p(s,k)= k_0 + k_1 s + k_2 s^2 + \ldots + k_n s^n$$

- We have

- odd coefficients $k_o = [k_1, k_3, \ldots, k_{2n_o+1}]^T$

- even coefficients $k_e = [k_0, k_2, \ldots, k_{2n_e}]^T$

IEIIT-CNR

Mixed Deterministic/Randomized Methods - 1

k_o deterministic

k_e randomized

- deterministic methods for odd coefficients

$$k_o = [k_1, k_3, \ldots, k_{2n_o+1}]^T$$

- randomized methods for even coefficients

$$k_e = [k_0, k_2, \ldots, k_{2n_e}]^T$$

© RT - 2006

IEIIT-CNR

Mixed Deterministic/Randomized Methods - 2

k_o randomized

k_e deterministic

- randomized methods for odd coefficients

$$k_o = [k_1, k_3, \ldots, k_{2n_o+1}]^T$$

- deterministic methods for even coefficients

$$k_e = [k_0, k_2, \ldots, k_{2n_e}]^T$$

IEIIT-CNR

Odd Coefficients: Probabilistic Model

- Assume that the odd coefficients of $p(s,k)$ are random variables distributed in the intervals

$$k_{2i+1} \in [k_{2i+1}^-, k_{2i+1}^+], i = 0, 1, 2, \ldots, n_o$$

where $n_o = (n\text{-}1)/2$ if n is odd or $n_o = n/2 - 1$ if n is even

- Letting $k_o = [k_1, k_3, \ldots, k_{2n_o+1}]^T$, we define the odd box

$$K_o = \{k_o : k_{2i+1} \in [k_{2i+1}^-, k_{2i+1}^+], i = 0, 1, 2, \ldots, n_o\}$$

© RT - 2006

Even Coefficients: Deterministic Model

- Assume that the even coefficients of $p(s,k)$ vary in the intervals

$$k_{2i} \in [k_{2i}^{-}, k_{2i}^{+}],\ i = 0, 1, 2, \ldots, n_e$$

where $n_e = n/2$ if n is even or $n_o = (n\text{-}1)/2$ if n is odd

- Letting $k_e = [k_0, k_2, \ldots, k_{2n_e}]^T$, we define the even box

$$K_e = \{k_e : k_{2i} \in [k_{2i}^{-}, k_{2i}^{+}],\ i = 0, 1, 2, \ldots, n_e\}$$

- No probabilistic assumption is used in this case

IEIIT-CNR

Randomization for Odd Coefficients

- Uniform randomization of K_o (Monte Carlo approach) is extremely inefficient
- We use an *importance sampling* technique (providing non-uniform samples), i.e. we limit the search space
- Perform randomization within a subset K_o^{nec} of K_o which contains the Hurwitz region

$$K_o \supseteq K_o^{nec} \supseteq H = \{k \in K : p(s,k) \text{ is Hurwitz}\}$$

© RT - 2006

IEIIT-CNR

The Set K_o^{nec}

- Computation of K_o^{nec} is involved
- Requires establishing a necessary condition for stability based on the Hermite-Biehler Theorem and on a Newton condition for polynomials having real roots

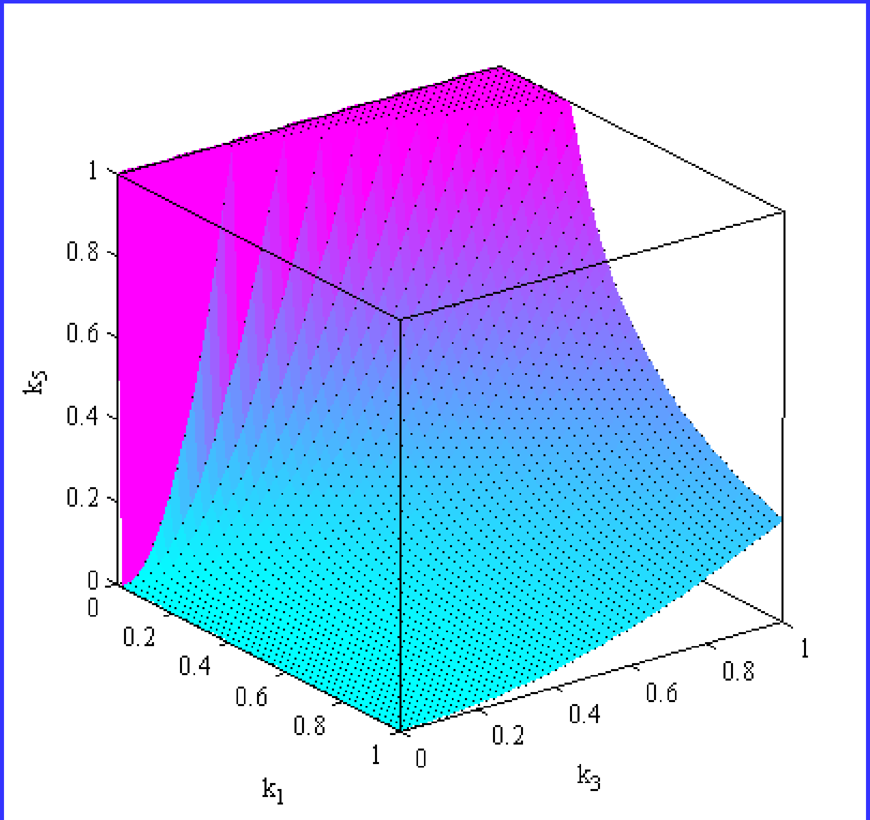
$$K_o^{nec} = \{k_o \in K_o : k_{2i+1}\, k_{2i-3} \leq C(i,n_o)\,(k_{2i-1})^2,\ i = 2, 3, \ldots, n_o\}$$

where $C(i,n_o) = ((i-1)\,(n_o-i+1))/(i(n_o-i+2))$

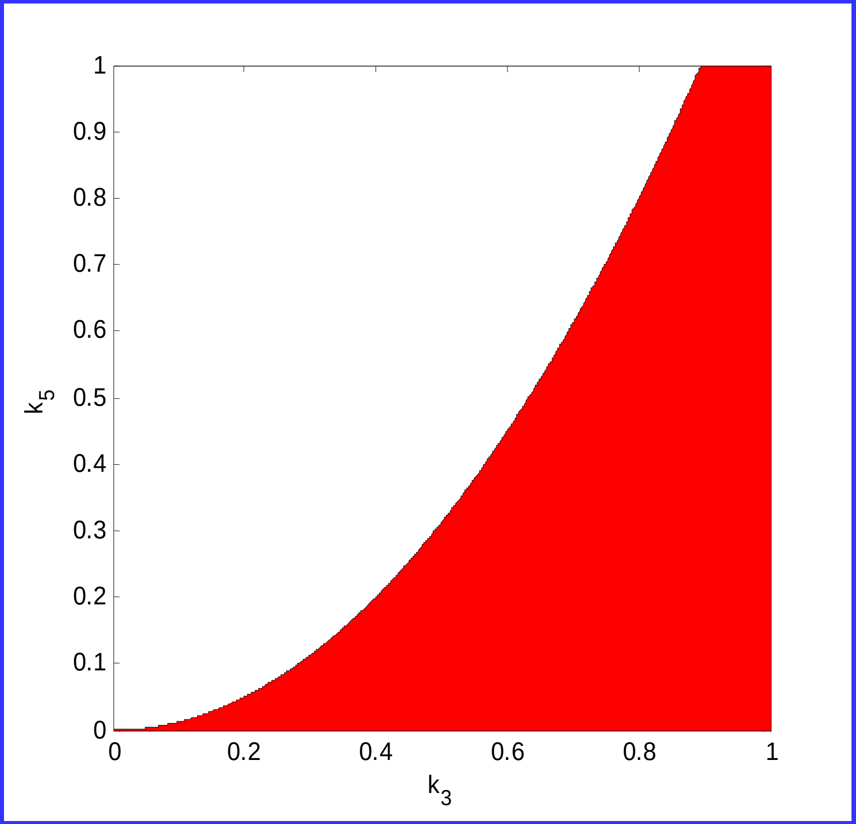
- This condition is recursive and useful for randomization

Sets K_o^{nec} and K_o

$n_o = 2$



2-D cross-section for $k_1 = 0.2$



© RT - 2006

IEIIT-CNR

Estimated Ratio between Volumes

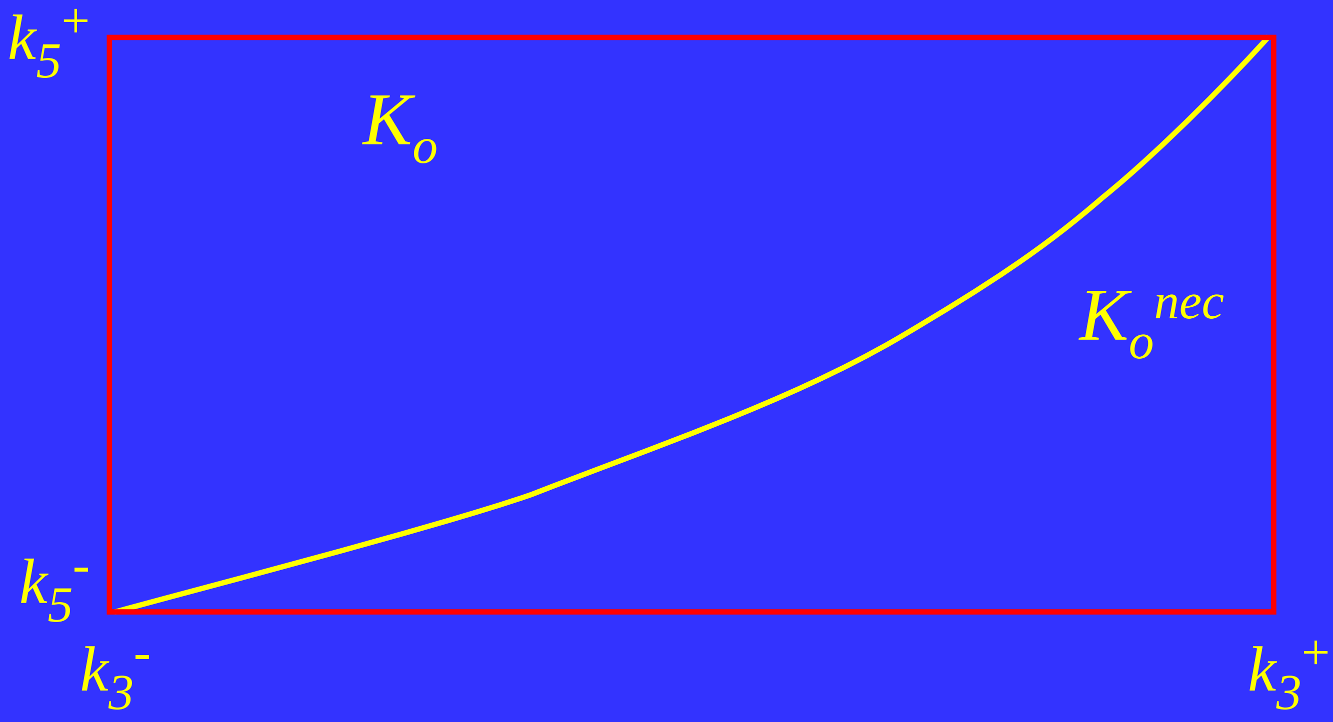
Estimated ratio ξ between volumes of the sets K_o^{nec} and $K_o = [0.001,1]^{n_o}$ (based on 10^5 uniform random samples)

n_o	ξ
2	0.0279
3	0.0011
4	2 e–5
≥ 5	0

IEIIT-CNR

Randomized Algorithm 1

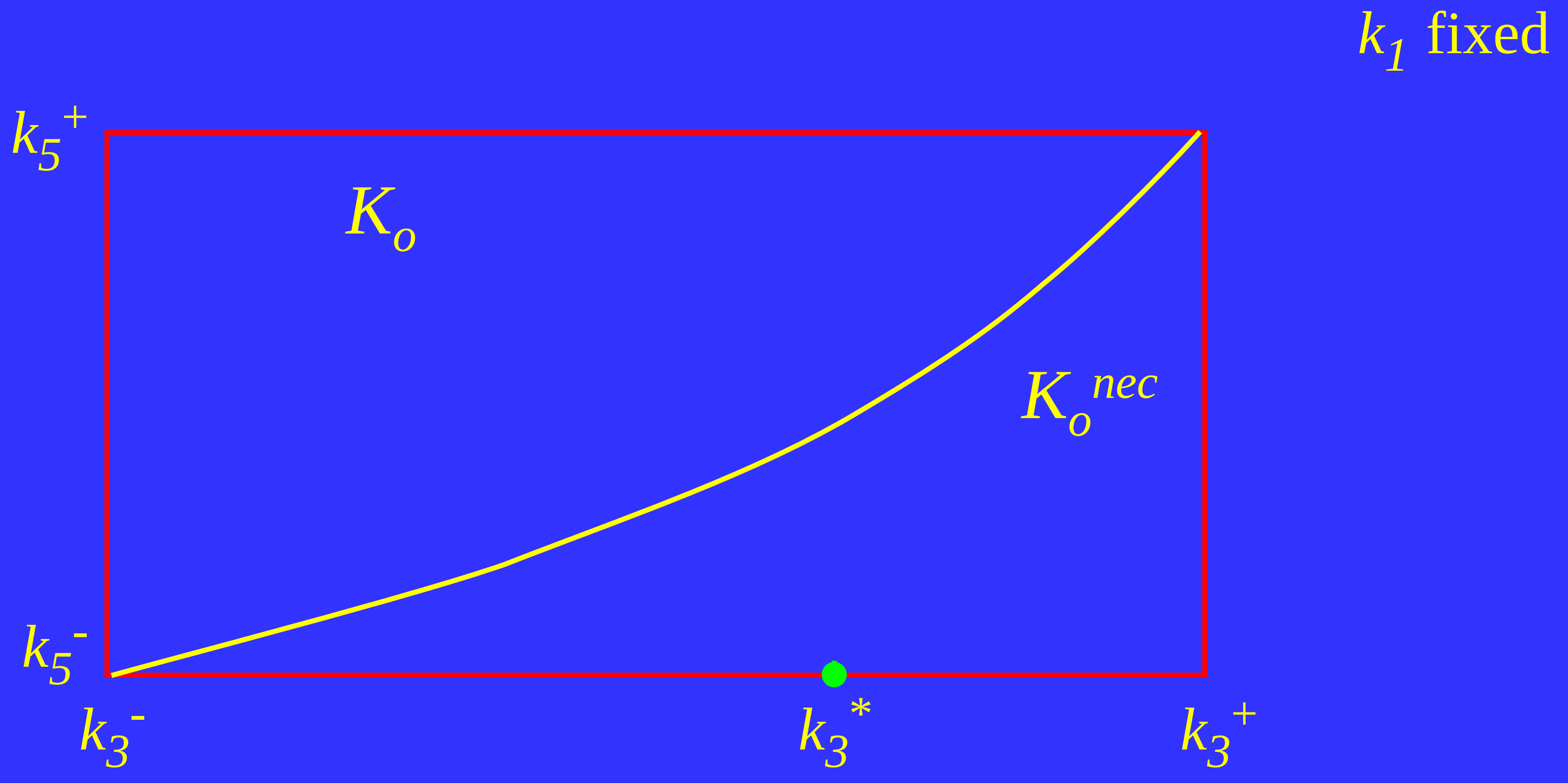
k_1 fixed



© RT - 2006

IEIIT-CNR

Randomized Algorithm 1

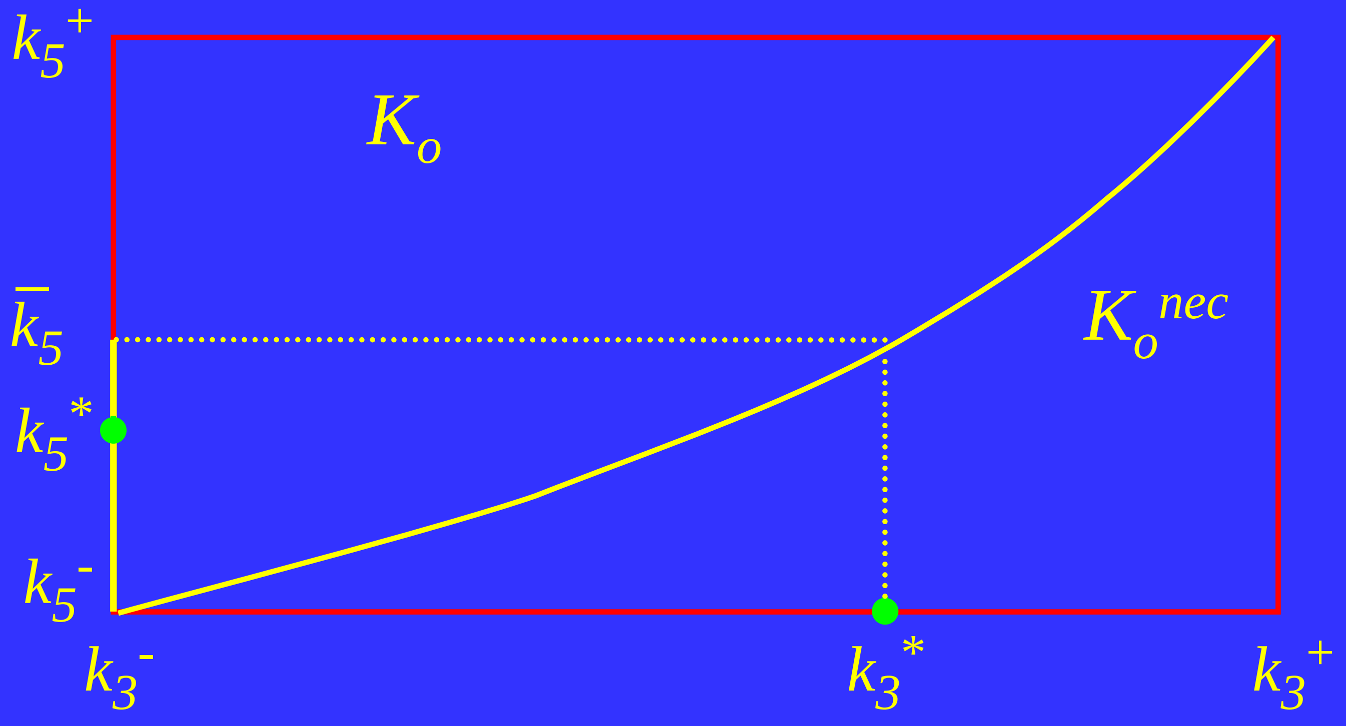


© RT - 2006

IEIIT-CNR

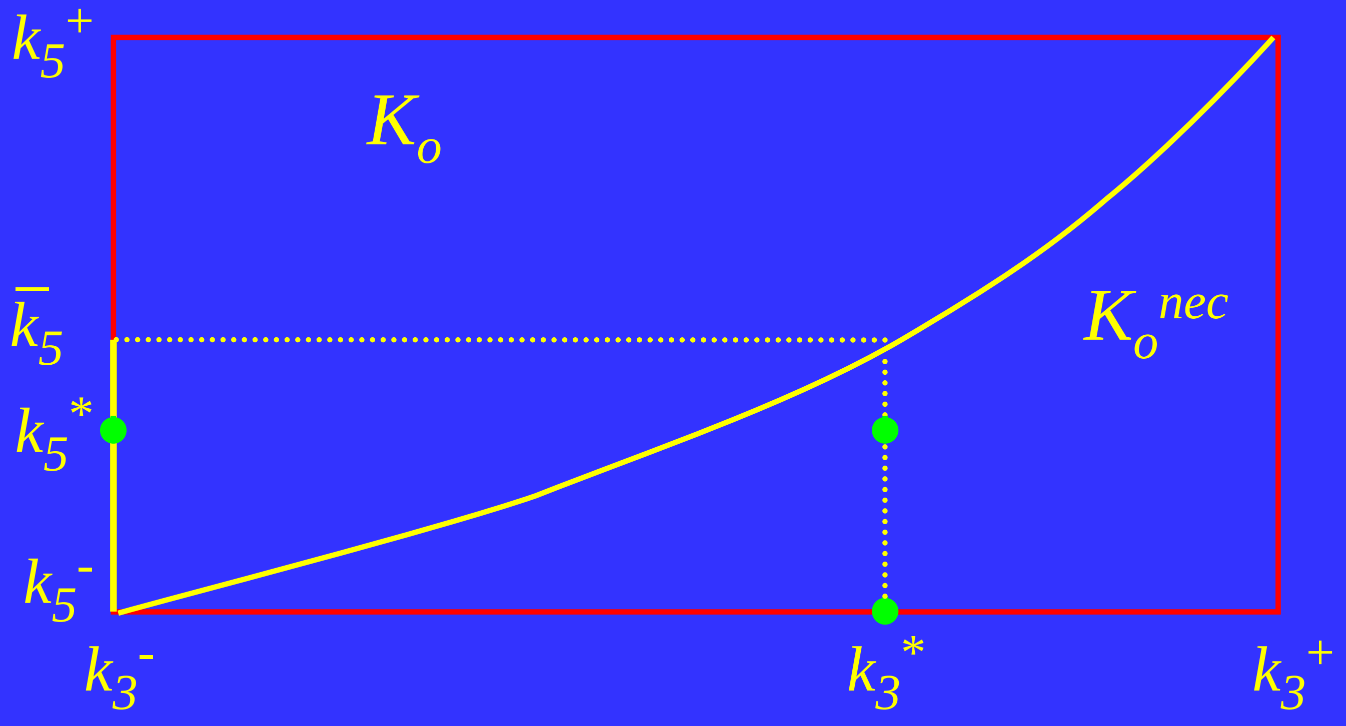
Randomized Algorithm 1

k_1 fixed



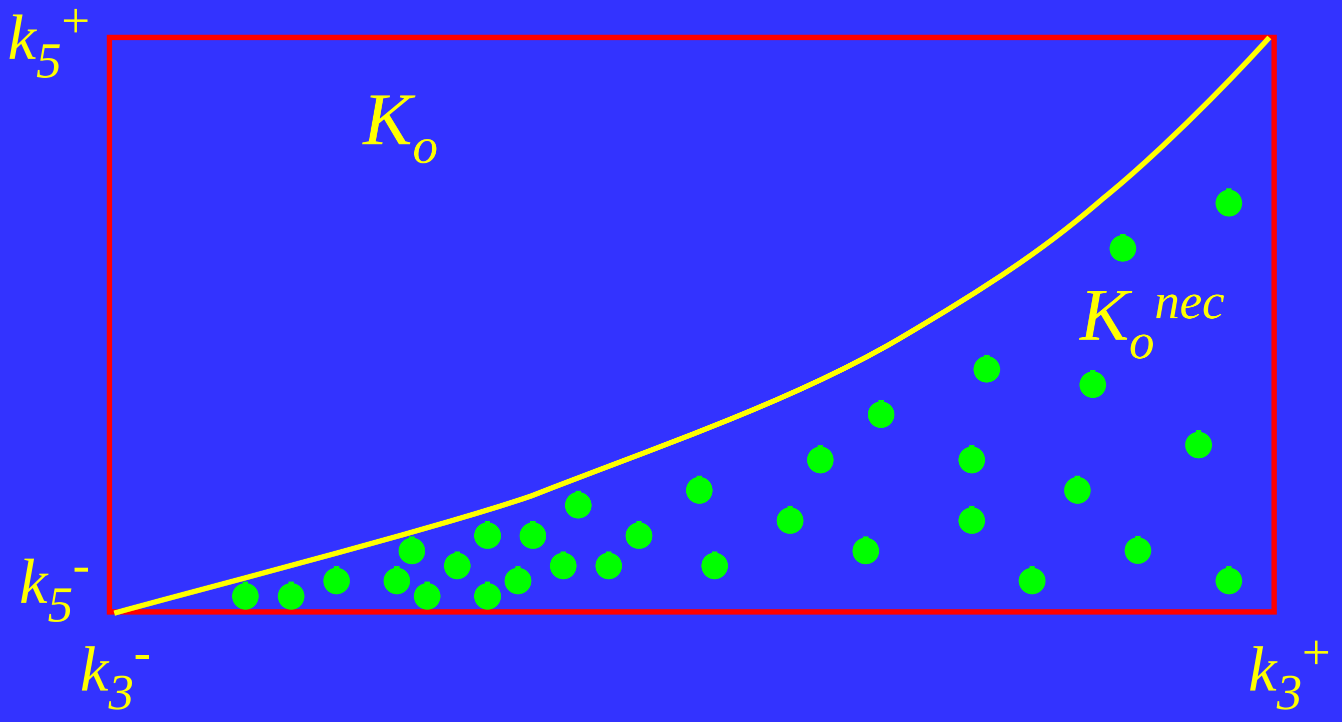
Randomized Algorithm 1

k_1 fixed



Randomized Algorithm 1

k_1 fixed



© RT - 2006

Randomized Algorithm 1

Generates (non-uniform) coefficients k_o in the set K_o^{nec}

1. generate k_1, k_3 uniformly in $[k_1^-, k_1^+]$, $[k_3^-, k_3^+]$
2. for $i = 2, 3, \ldots, n_o$

   - $I_{2i+1} = [k_{2i+1}^-, \min\{k_{2i+1}^+, C(i,n_o)\,(k_{2i-1})^2/k_{2i-3}\}]$

   - if I_{2i+1} is empty go to 1

   else generate k_{2i+1} uniformly in the interval I_{2i+1}

Deterministic Methods for Even Coefficients

IEIIT-CNR

- Suppose that k_o is computed using Randomized Algorithm 1
- Theorem 3[1]

  The set H_e of all even coefficients providing a Hurwitz interval polynomial is either empty or is given by

$$H_e = K_e \cap C_e$$

  where C_e is the interior of a polyhedral cone

[1] F. Dabbene, B. Polyak and R. Tempo (2006)

Remarks

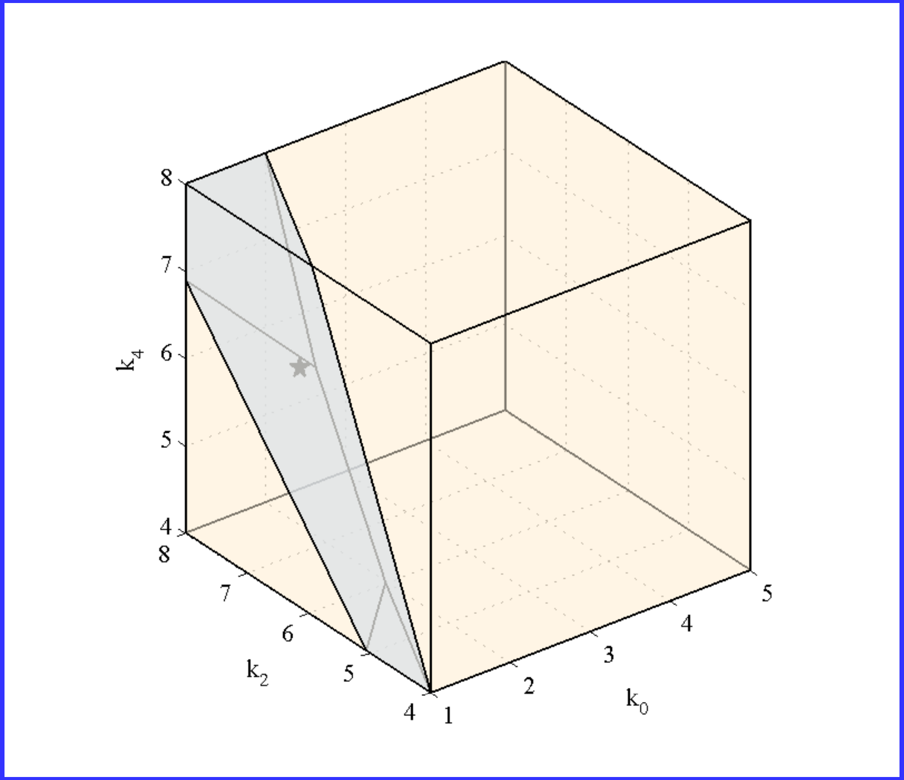
- The set H_e is a polytope given by the intersection of the even box K_e and a polyhedral cone C_e
- Checking if H_e is empty can be immediately accomplished solving only one linear program
- Specific points (e.g. analytic center or Chebichev center) within H_e can be also easily found by linear program

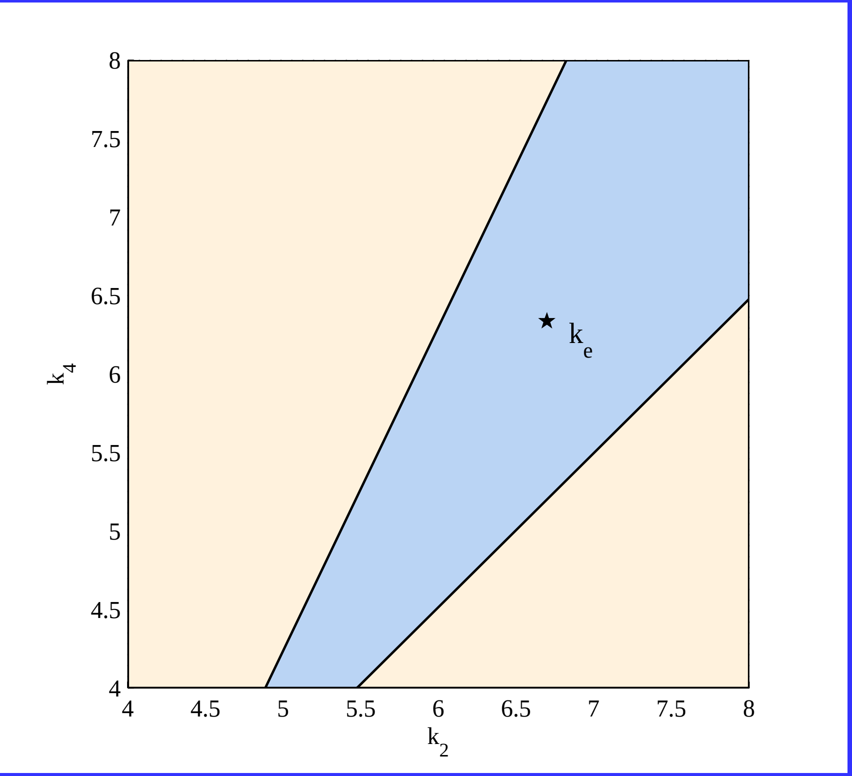
© RT - 2006

Polytope H_e and Analytic Center

$n_o = 3$

2-D cross-section





© RT - 2006

IEIIT-CNR

Simulations for Unit Box K - 1

polynomial degree	expected number of iterations (proposed method)	expected number of iterations (Monte Carlo)
3	1	2
4	1	62
5	1	50
6	1	1,108
7	2	28,334
8	2	∞
9	3	∞
10	5	∞

IEIIT-CNR

Simulations for Unit Box *K* - 2

polynomial degree	expected number of iterations (proposed method)	expected number of iterations (Monte Carlo)
11	7	∞
12	34	∞
13	80	∞
14	626	∞
15	4,099	∞
16	6,461	∞
17	76,968	∞
18	90,093	∞

© RT - 2006

IEIIT-CNR

Volume of Hurwitz Polynomials

- Volume of Hurwitz region V_{stab} is defined as

$$V_{stab} = \int_H d\mu$$

where $d\mu$ is the Lebesgue measure

- Probability one estimate of V_{stab} can be obtained (asymptotically) computing

- k_o (with Randomized Algorithm 1)

- volume of polytope H_e (with a deterministic algorithm)

© RT - 2006

Fixed Order Controller Design

IEIIT-CNR

Plant and Controller

- SISO strictly proper plant

$$P(s) = \frac{N_P(s)}{D_P(s)}$$

- Fixed order controller in even/odd form

$$C(s) = \frac{N_C(s)}{D_C(s)} = \frac{X(s^2)+sY(s^2)}{Z(s^2)+sV(s^2)}$$

© RT - 2006

Controller Polynomials

- The even-order controller polynomials are of the form

$$X(s^2) = \theta_0 + \theta_2 s^2 + \theta_4 s^4 + \ldots$$
$$Y(s^2) = \alpha_0 + \alpha_2 s^2 + \alpha_4 s^4 + \ldots$$
$$Z(s^2) = \beta_0 + \beta_2 s^2 + \beta_4 s^4 + \ldots$$
$$V(s^2) = \mu_0 + \mu_2 s^2 + \mu_4 s^4 + \ldots$$

- Closed-loop polynomial is given by

$$p(s) = N_P(s)\,(X(s^2)+sY(s^2)) + D_P(s)\,(Z(s^2)+sV(s^2))$$

- We assume that $p(s)$ has invariant degree (generic subset of controller parameters)

Mixed Deterministic/Randomized Methods - 1

Y, Z, V randomized

X deterministic

- randomized methods for

$$Y(s^2), Z(s^2), V(s^2)$$

- deterministic methods for

$$X(s^2)$$

© RT - 2006

Mixed Deterministic/Randomized Methods - 2

1. Deterministic methods for $X(s^2)$; randomized methods for $Y(s^2)$, $Z(s^2)$, $V(s^2)$
2. Deterministic methods for $Y(s^2)$; randomized methods for $X(s^2)$, $Z(s^2)$, $V(s^2)$
3. Deterministic methods for $Z(s^2)$; randomized methods for $X(s^2)$, $Y(s^2)$, $V(s^2)$
4. Deterministic methods for $V(s^2)$; randomized methods for $X(s^2)$, $Y(s^2)$, $Z(s^2)$

© RT - 2006

Mixed Deterministic/Randomized Methods - 3

IEIIT-CNR

- Consider the first case
- Then, we have

$$\theta = [\theta_0 \;\; \theta_2 \;\; \theta_4 \ldots]^T \qquad \text{deterministic methods}$$

$$\eta = [\alpha_0 \, \alpha_2 \;\; \alpha_4 \ldots \beta_0 \;\; \beta_2 \; \beta_4 \ldots \mu_0 \, \mu_2 \, \mu_4 \ldots]^T$$

randomized methods

- We assume that $\theta \in M$ and $\eta \in N$

$$n_\theta = \deg(X)/2 + 1$$

$$n_\eta = \deg(Y)/2 + \deg(V)/2 + \deg(Z)/2 + 3$$

Set of Stabilizing Controllers

- Suppose that η is selected using a randomized algorithm

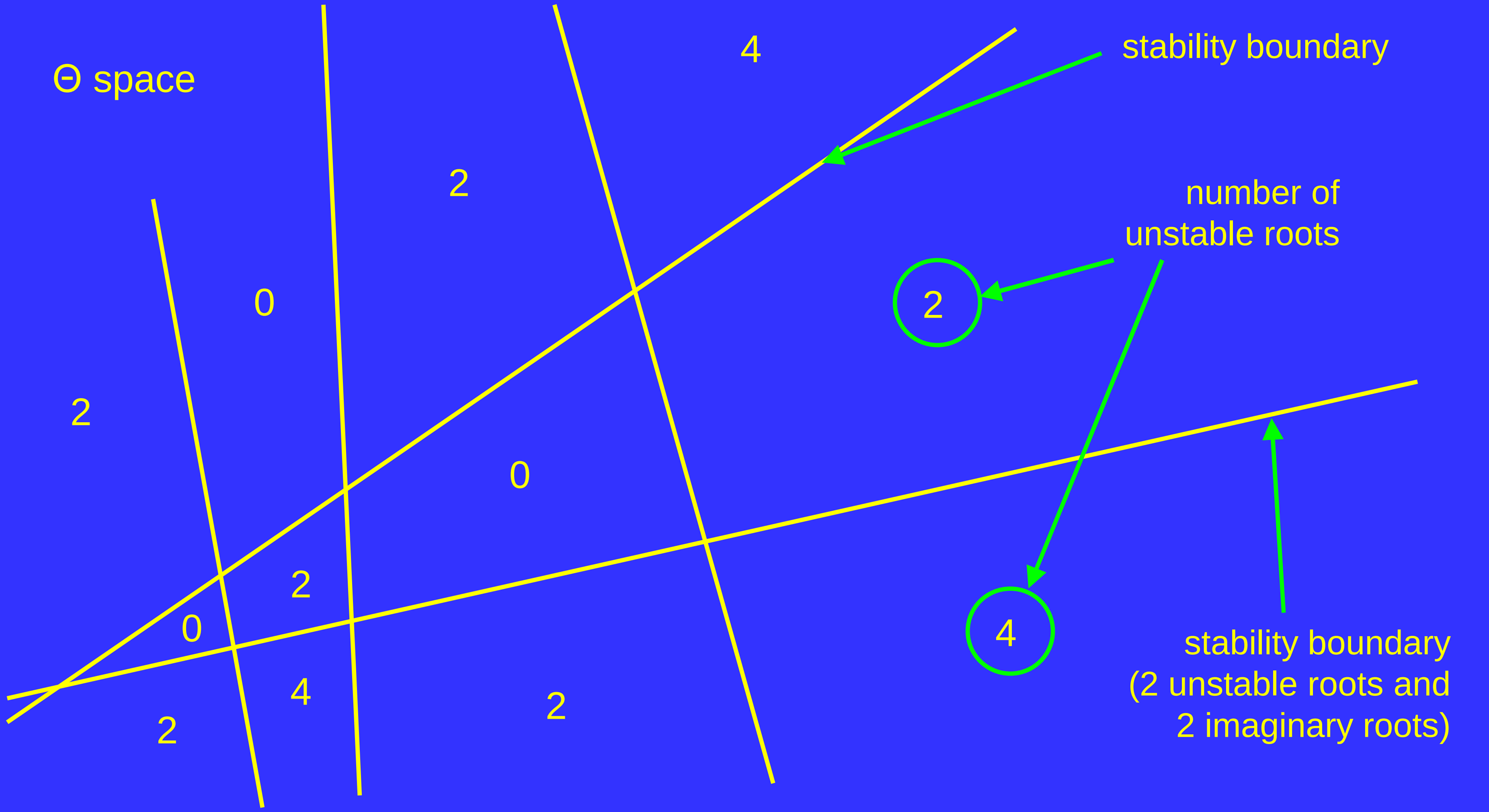
- Theorem 4[1]

  The set of all parameters θ providing a stabilizing controller is either empty or is a union of a finite number of polyhedral sets

[1] Y. Fujisaki, Y. Oishi and R. Tempo (2005)

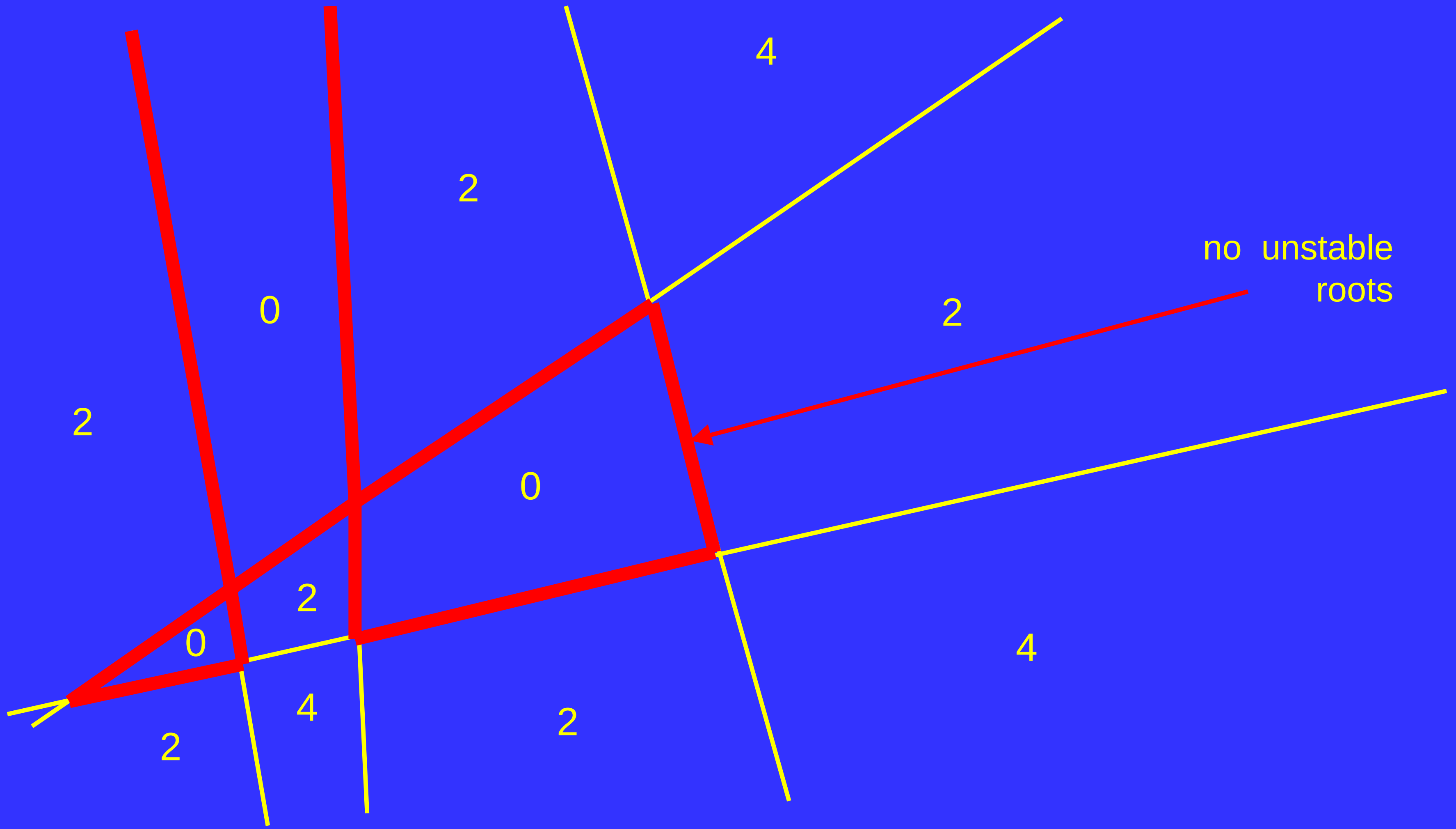
© RT - 2006

Parameter Space θ



IEIIT-CNR

Set of Stabilizing Controllers



Fixed Order Stabilizing Controller

- Combinatorial problem: Linear programming approach to find parameters θ may become inefficient because an exponential number of constraints need to be considered
- Different method (polynomial-time) based on vertices instead of hyperplanes is developed[1]

[1] Y. Fujisaki, Y. Oishi and R. Tempo (2005)

IEIIT-CNR

Extensions: H_∞ Performance

- Extensions of this method to H_∞ performance and stabilization of interval plants
- H_∞ performance of sensitivity function

$$S(s) = \frac{1}{1 + P(s)\, C(s)}$$

- Consider also a (stable) weighting function

$$W(s) = N_W(s)/D_W(s)$$

© RT - 2006

Set of H_∞ Controllers

- Suppose that η is selected using a randomized algorithm
- Theorem 5

The set S of all tractable parameter vectors θ giving an H_∞ controller satisfying

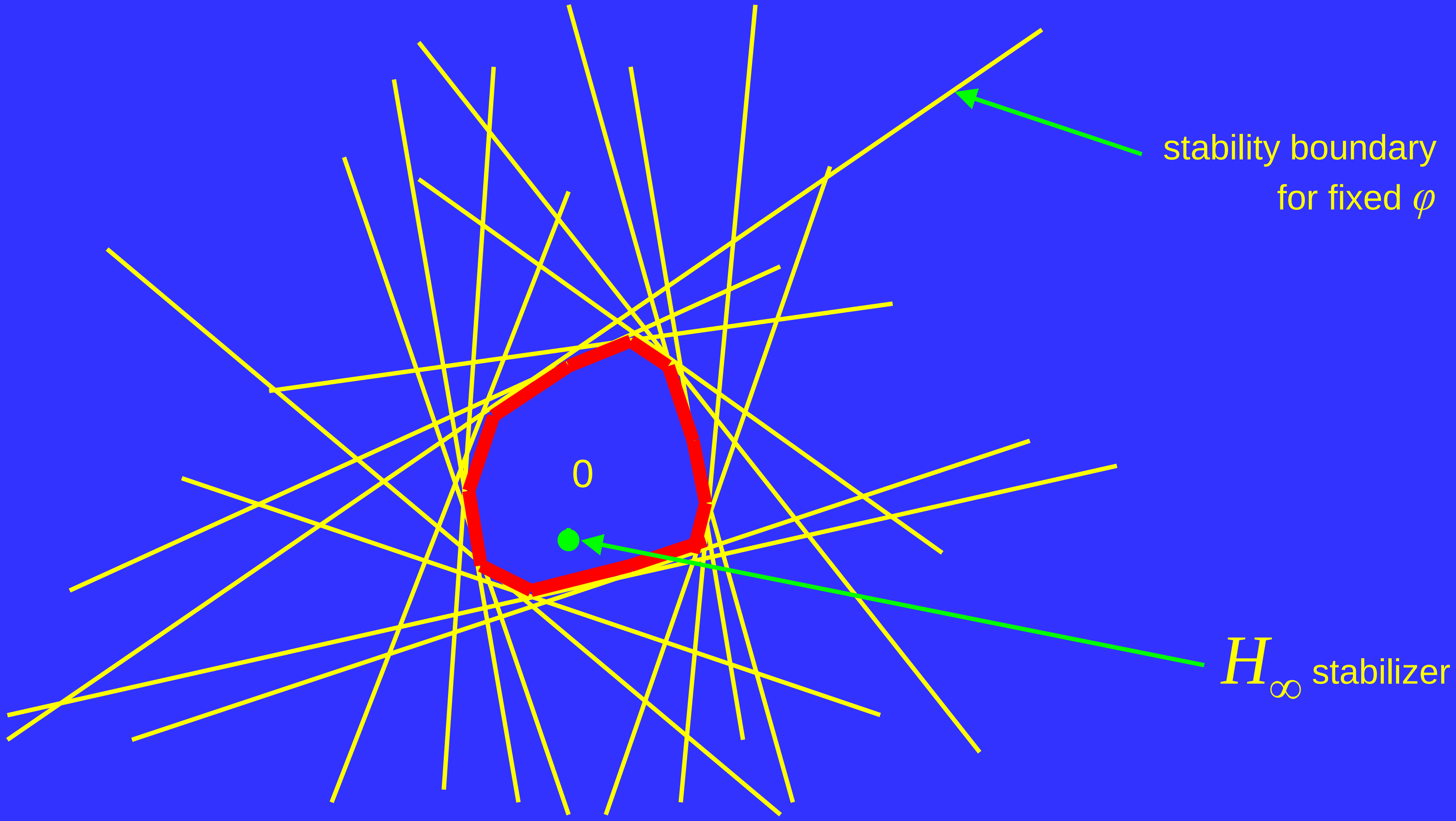
$$\| W(s)\, S(s) \|_\infty \leq \gamma$$

is either empty or is given by

$$S = \bigcap_{\varphi} \Gamma(\varphi)$$

where $\Gamma(\varphi)$ is the union of finite number of polyhedral sets for fixed $\varphi \in [0,2\pi)$

(Convex) Set of H_∞ Controllers



Some Recent Research Directions

Switched Systems, Optimization, System Identification, …

IEIIT-CNR

- Common Lyapunov functions for switched system[1]
- From common to piecewise Lyapunov functions[2]
- Ellipsoidal randomized algorithm[3] and stopping rules[4]
- RAs for semi-infinite programming[5]
- MRAS methods for global optimization[6]
- Estimation via MCMC[7]
- RAs for model validation[8] and system identification[9]

[1] D. Liberzon and R. Tempo (2004)
[2] H. Ishii, T. Basar and R. Tempo (2005)
[3] S. Kanev, B. De Schutter and M. Verhaegen (2002)
[4] Y. Oishi and H. Kimura (2003)
[5] V. B. Tadic, S. P. Meyn and R. Tempo (2006)
[6] J. Hu, M.C. Fu and S.I. Marcus (2005)
[7] J.C. Spall (2004)
[8] M. Sznaier, C. M. Lagoa and M.C. Mazzaro (2005)
[9] X. Bombois, G. Scorletti, M. Gevers, P. Van den Hof and R. Hildebrand (2006)

IEIIT-CNR

Applications of RAs

- RAs have been developed for many control applications
- Control of flexible structures
- Robustness of high speed networks
- Stability of quantized sampled-data systems
- Control design for brushless DC motors
- Synthesis of real time embedded controllers
- Mini-UAV control design

Applications of RAs

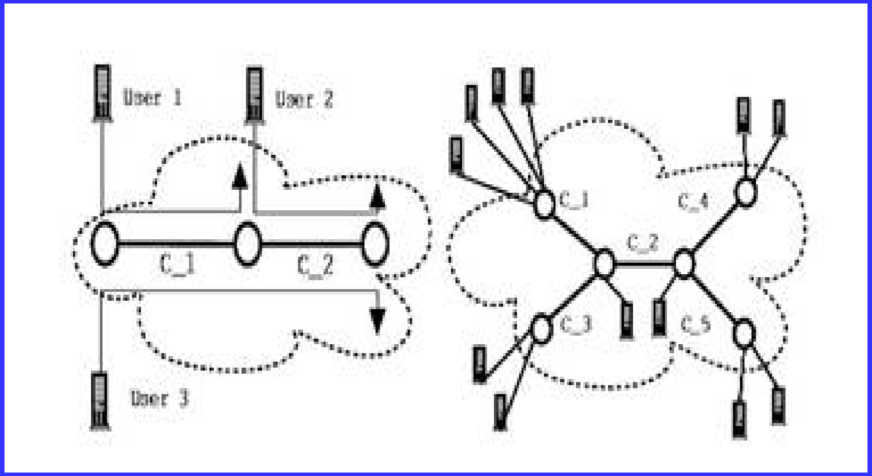
- RAs have been developed for many control applications
- Control of flexible structures
- Robustness of high speed networks
- Stability of quantized sampled-data systems
- Control design for brushless DC motors
- Synthesis of real time embedded controllers
- Mini-UAV control design

© RT - 2006

Stability and Robustness of High-Speed Networks

IEIIT-CNR

- Analysis of network topology
- Source and destination nodes, links (with buffer and capacity)
- Bottleneck link
- Symmetric case: Stability and robustness can be studied analytically



- Non-symmetric case: Use of randomized algorithms

[1] T. Alpcan, T. Basar and R. Tempo (2005)

IEIIT-CNR

Applications of RAs

- RAs have been developed for many control applications
- Control of flexible structures
- Robustness of high speed networks
- Stability of quantized sampled-data systems
- Control design for brushless DC motors
- Synthesis of real time embedded controllers
- Mini-UAV control design

© RT - 2006

Mini-UAV Control Design

- Study and development of a real-time land control and monitoring system for fire prevention in Sicily
- Uncertainty description
- Development of three RAs for gain synthesis and robustness analysis (according to flying quality military specs)

[1] L. Lorefice, B. Pralio and R. Tempo (2006)

Conclusions: Benefits or Pitfalls?

© RT - 2006

Benefits or Pitfalls?

- Randomized algorithms are Probably Approximately Correct (PAC)
- We give up a guaranteed deterministic solution
- This implies accepting a “small” risk of giving a wrong solution
- The risk can be made arbitrarily small (but not zero) taking suitable values of so-called confidence and accuracy

Benefits or Pitfalls?

- We provide an efficient (polynomial-time) solution to many problems which are computationally intractable in a (classical) worst case sense
- RAs presented in this talk fall in the class of Monte Carlo methods
- Different class of RAs are Las Vegas algorithms
- Las Vegas RAs always provide a correct solution, but they work only for limited classes of problems

© RT - 2006

Mini-Symposium MoP08 titled

Randomized and Probabilistic Techniques for Complex Systems Design

Organizers

Yasuaki Oishi and Yasumasa Fujisaki

Room I at 3:20

Don’t miss it!

Acknowledgments

Thanks to

- Fabrizio Dabbene, IEIIT-CNR, Italy
- Yasumasa Fujisaki, Kobe University, Japan
- Yasuaki Oishi, The University of Tokyo, Japan
- Boris Polyak, Russian Academy of Sciences, Russia

© RT - 2006