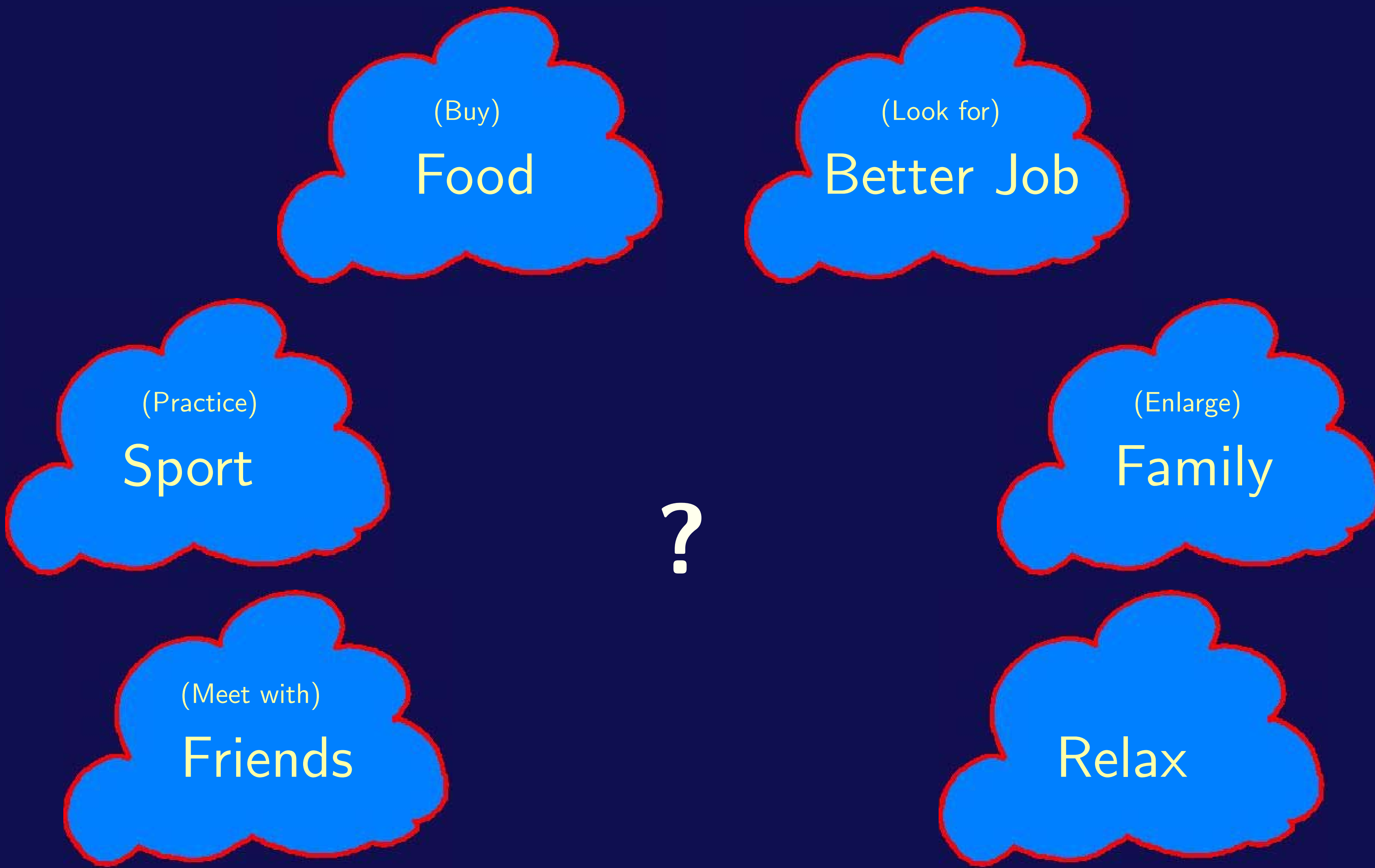
Index Policies for Stochastic Dynamic Optimization

Peter Jacko*

BCAM SC meeting 2011, December 12

*BCAM — Basque Center for Applied Mathematics, Spain

Everyday Decision-Making



Academic Task Management

(Prepare)
Classes
Due: tomorrow

(Have)
Lunch
Due: before 2pm

Investigate
Due: one month

?

(Evaluate)
Homeworks
Due: one week

(Write)
Paper
Due: two weeks

(Look for)
Funding
Due: next year

Motivation

- Problems intractable for finding an optimal solution
- Use of dynamic priorities in daily decision making
  - ▷ easy to interpret
  - ▷ easy to implement
  - ▷ often well-performing
- A divide and conquer solution approach
- Model: multi-armed restless bandit problem
  - ▷ Markov decision process with special structure
  - ▷ optimizing under the discounted or average criterion
  - ▷ subject to a sample path capacity constraint

Multi-Armed Restless Bandit Problem

Index Policies

- Priorities defined by dynamic index values
- Index policy: assign the resource to the competitor with highest actual index
- Proposed in increasingly more general settings by
  - Smith (1956): job scheduling (optimal)
  - Gittins (1970's): classic bandits (optimal)
  - Whittle (1988): restless bandits
  - Niño-Mora (2000's): index existence and computation
  - Jacko (2005-): scheduling and resource allocation
- Index policy is a tractable heuristic in general

Talk Outline

- Resource allocation MDP framework
- Decomposition and indexability
- Selected applications
  - ▷ control of Internet flows
  - ▷ knapsack problem for perishable products
  - ▷ scheduling of impatient customers
  - ▷ user scheduling in wireless networks
- Open problems

Resource Allocation Problem (RAP)

- Stochastic and dynamic
- There is a number of independent competitors
- Constraint: resource capacity W at any time
- Objective: maximize expected "reward"
- Captures the exploitation vs. exploration trade-off
  - ▷ always exploiting (being myopic) is not optimal
  - ▷ always exploring (being utopic) is not optimal
- This framework models learning by doing!

Questions to Answer

- [Economic] For a given joint goal, is it possible to define sound dynamic quantities for each competitor that can be interpreted as priorities? And if yes,
- [Algorithmic] How to calculate such priorities quickly?
- [Mathematical] Under what conditions is there a priority rule that achieves optimal resource capacity allocation?
- [Experimental] If priority rules are not optimal, how close to optimality do they come? And how do they compare to alternative policies?

MDP Framework

- Markov Decision Processes
- Discrete time model $(t = 0, 1, 2, \dots)$
- Competitor $k \in \mathcal{K}$ is defined by
  - ▷ states $\mathcal{N}_k$, actions $\mathcal{A} := \{0, 1\}$
  - ▷ expected one-period capacity consumption $\boldsymbol{W}_k^a$
  - ▷ expected one-period reward $\boldsymbol{R}_k^a$
  - ▷ one-period transition probability matrix $\boldsymbol{P}_k^a$
- State process $X_k(t) \in \mathcal{N}_k$
- Action process $a_k(t) \in \mathcal{A}$ – to be decided

Resource Allocation Problem

- Formulation under the β-discounted criterion:

$$\max_{\pi \in \Pi} \sum_{k \in \mathcal{K}} \mathbb{E}^{\pi} \left[\sum_{t=0}^{\infty} \beta^t R_{k, X_k(t)}^{a_k(t)} \right]$$

subject to $\sum_{k \in \mathcal{K}} W_{k, X_k(t)}^{a_k(t)} = W,$ for all $t = 0, 1, 2, \ldots$

- Analogously under the time-average criterion

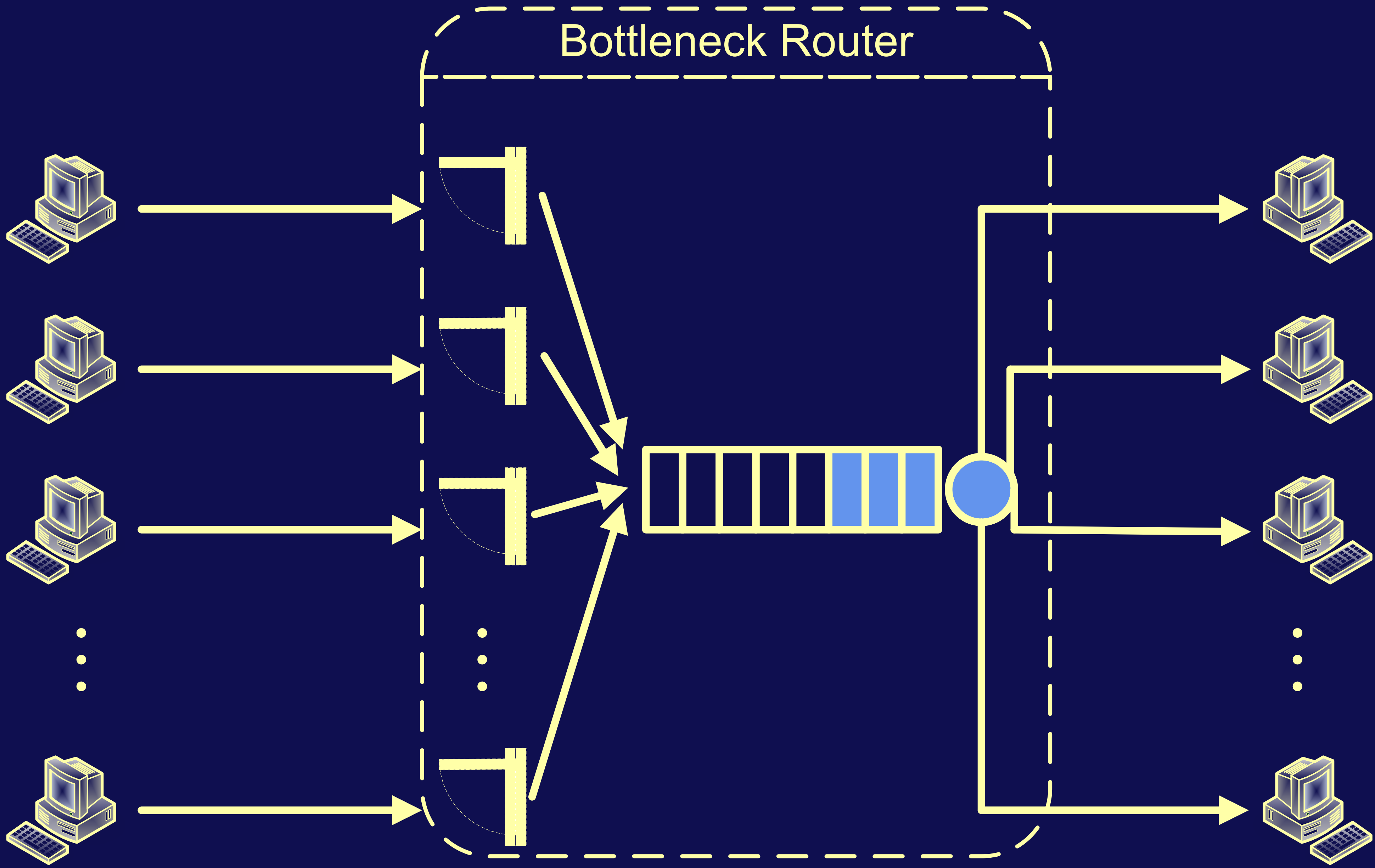
- PSPACE-hard (Papadimitriou & Tsitsiklis 1999)
  - ▷ intractable to solve exactly by Dynamic Programming
  - ▷ instead, we relax and decompose the problem

Relaxations and Decomposition

- 1. Whittle's (1988): Use resource W in expectation
  - ▷ infinite number of constraints is replaced by one
  - ▷ sort of perfect market assumption
- 2. Lagrangian: Pay cost ν for using the resource
  - ▷ the constraint is moved into the objective
- Decomposes due to competitor independence into single-competitor parametric subproblems
  - ▷ solved by identifying the efficiency frontier
  - ▷ indexability $\approx$ threshold policies are optimal
  - ▷ math $+$ art $=$ characterize index values

Selected Applications

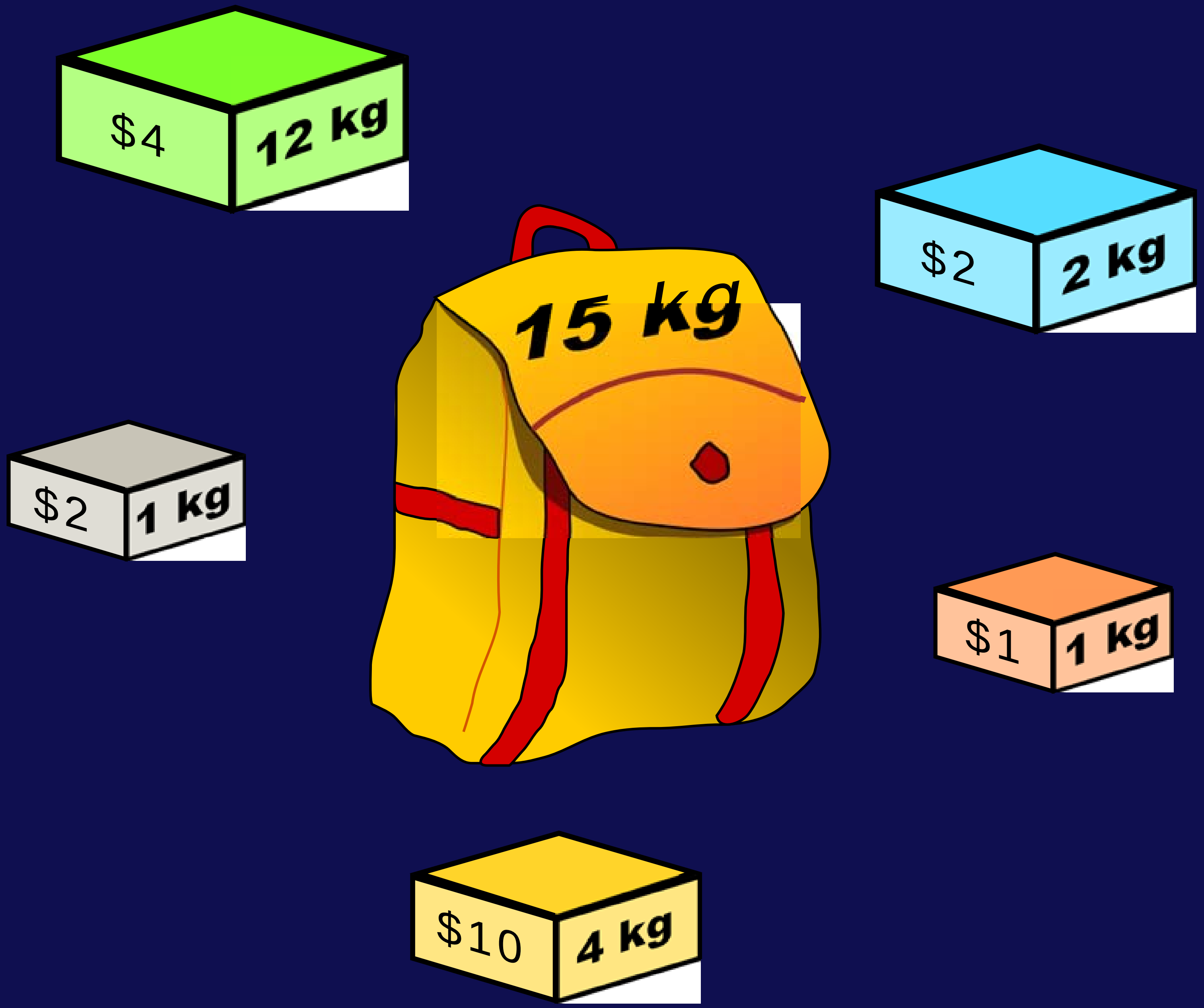
Control of Internet Flows



Control of Internet Flows

- Objective: fast and fair delivery of packets
- Difficulty: Different TCP variants, different round-trip times, aggressive flows
- J. & Sansó (Polytechnique de Montréal) (PEVA 2011)
- Doncel (internship) (2011): UPV master thesis
- Avrachenkov, Ayesta, Doncel, J. (submitted 2011)
- Avrachenkov (INRIA Sophia-Antipolis) & J. (in prep.)

Knapsack Problem for Perishable Products



Knapsack Problem for Perishable Products

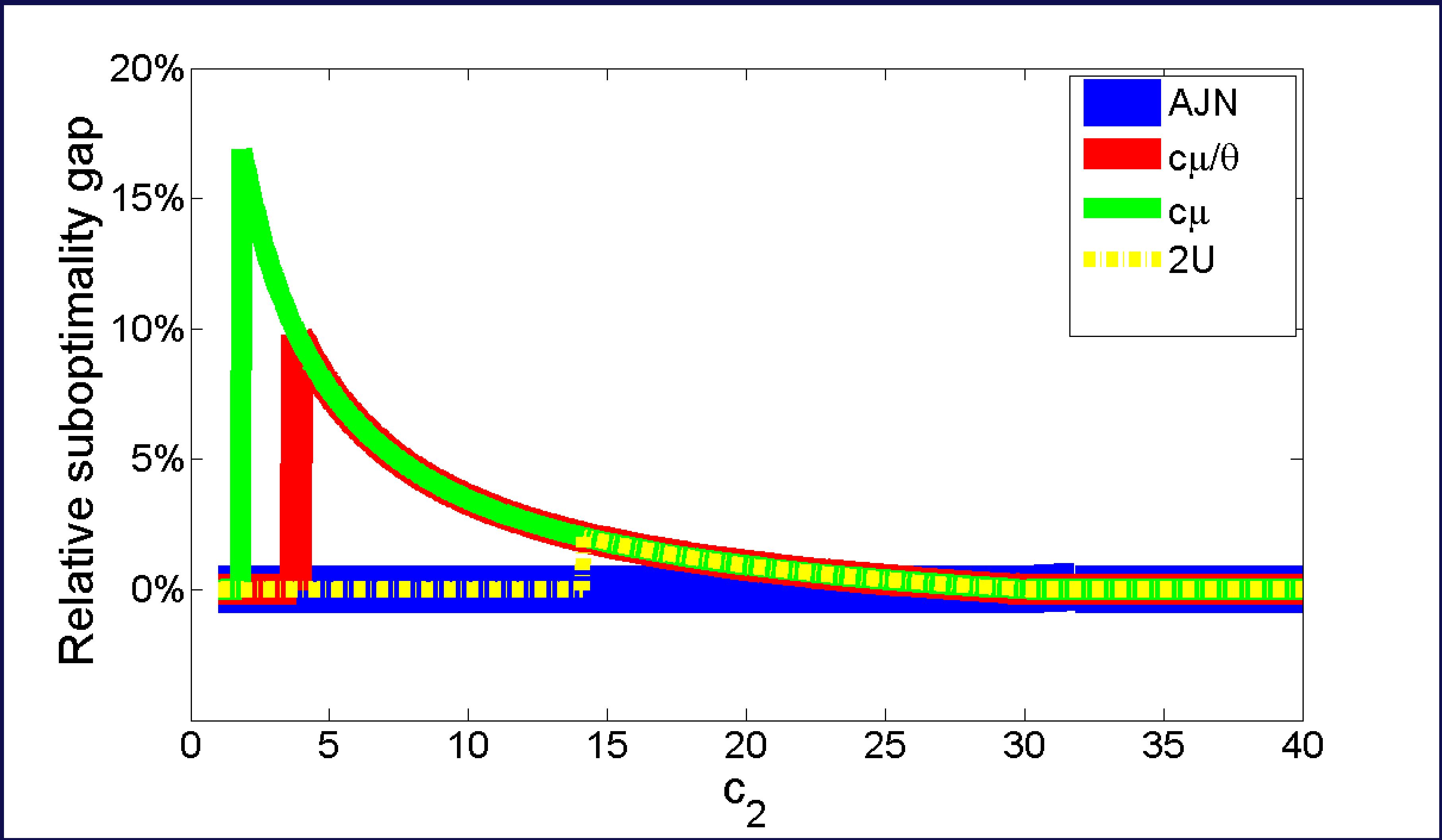
- Objective: maximize revenue
- Difficulty: different perishability dates, cross-dependent and time-varying demand
- J. (submitted 2011)
- Gráczová (PhD internship) & J. (submitted 2011)
- Possible applications in cloud computing, survey design

Scheduling of Impatient Customers

- Callers are willing to wait an average of 30-60 sec.
- Customer who just bought water in a supermarket
- Objective: avoid losing impatient customers and keep queues short
- Difficulty: classical queueing theory hard to apply (not work-conserving)
- Ayesta, J. & Novák (IEEE Infocom 2011)
- Novák (internship) (2011): Comenius bachelor thesis
  - ▷ best bachelor thesis, best research project

Scheduling of Impatient Customers

- Two customer classes:



User Scheduling in Wireless Networks

- CDMA 1xEV-DO
- Channel conditions vary randomly due to fading
- Channel conditions independent across users
- No interference
- Base station can serve W users per slot

User Scheduling in Wireless Networks

- Objective: keep waiting times short
- Difficulty: time-varying service rate and # users
- Ayesta & J. (patent filed 2010)
- Ayesta, Erausquin & J. (Performance 2010), 7 cit.
- Ayesta, Erausquin & J. (Allerton 2011), invited
- J. (Performance 2011), J. (2010)
- J., Morozov (Karelian) & Verloop (in prep.)
- Other NET papers...

User Scheduling in Wireless Networks

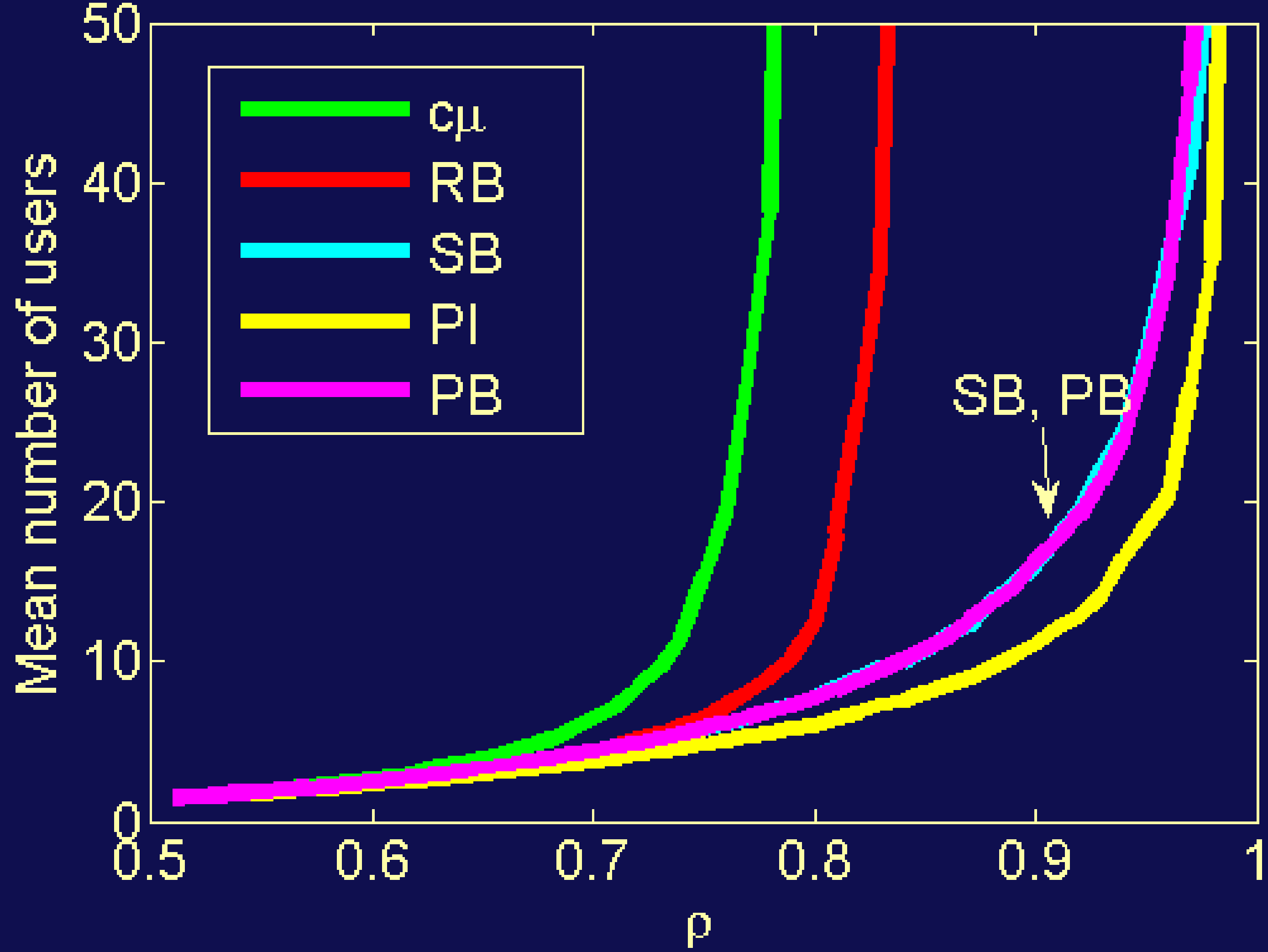
- Potential improvement (opportunistic) index

$$\frac{\text{actual transmission rate}}{\text{potential transmission rate improvement}}$$

- Scheduler: serve the job with highest actual PI index
  - ▷ tie-breaking in the best condition (index $= \infty$): serve the job with highest completion probability
- Outperforming other schedulers, maximally stable, fluid optimal, extensible to more general settings...

User Scheduling in Wireless Networks

- Varied arrival rate so that ϱ varies from 0.5 to 1



Conclusion

- Rich framework to study intractable problems
  - obtain elegant index rules
  - index policies optimal for relaxations
  - suggests structure of (asymptotically) optimal policies
- Open problems
  - general stability/optimality results
  - non-Markovian settings
  - what if indices do not exist
  - correlation among competitors

Thank you for your attention

Dynamic Prices (Index Values)

- We will assign a dynamic price to each user
- Arises in the solution of the parametric subproblem
  - ▷ optimal policy: use server iff price greater than ν
- Prices are values of ν when optimal solution changes
- However, such prices may not exist!
  - ▷ indexability has to be proved
- Price computation (if they exist):
  - ▷ in general, by parametric simplex method
  - ▷ by analysis sometimes obtained in a closed form

Optimal Solution to Subproblems

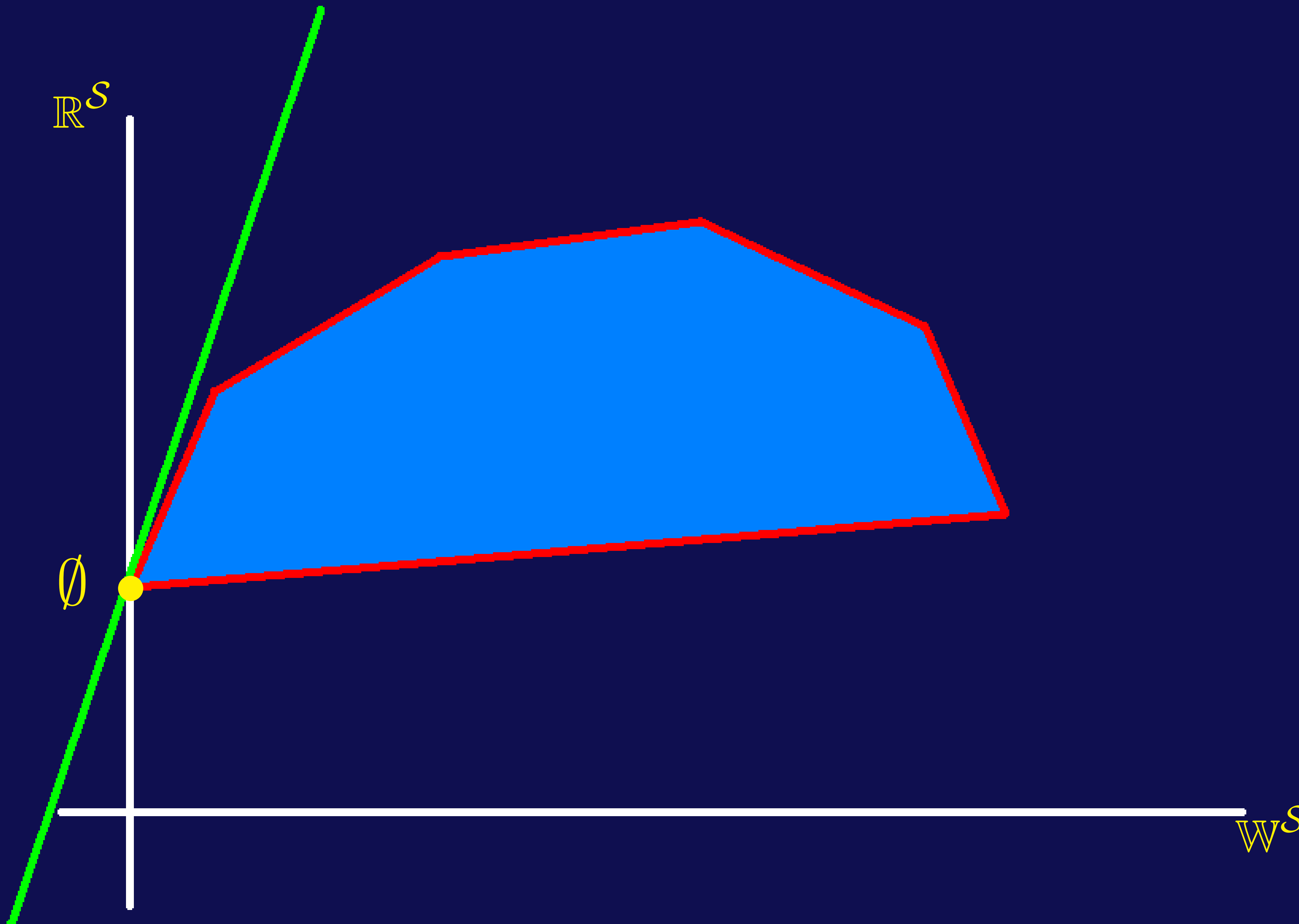
- For finite-state finite-action MDPs there exists an optimal policy that is deterministic, stationary, and independent of the initial state
  - ▷ we narrow our focus to those policies
  - ▷ represent them via serving sets $\mathcal{S} \subseteq \mathcal{N}$
  - ▷ policy $\mathcal{S}$ prescribes to serve in states in $\mathcal{S}$ and wait in states in $\mathcal{S}^{\mathsf{C}} := \mathcal{N} \setminus \mathcal{S}$
- Combinatorial ν-cost problem: $\max_{\mathcal{S} \subseteq \mathcal{N}} \mathbb{R}_n^{\mathcal{S}} - \nu \mathbb{W}_n^{\mathcal{S}}$, where

$$\mathbb{R}_n^{\mathcal{S}} := \mathbb{E}_n^{\mathcal{S}} \left[\sum_{t=0}^{\infty} \beta^t R_{X(t)}^{a(t)} \right], \quad \mathbb{W}_n^{\mathcal{S}} := \mathbb{E}_n^{\mathcal{S}} \left[\sum_{t=0}^{\infty} \beta^t W_{X(t)}^{a(t)} \right]$$

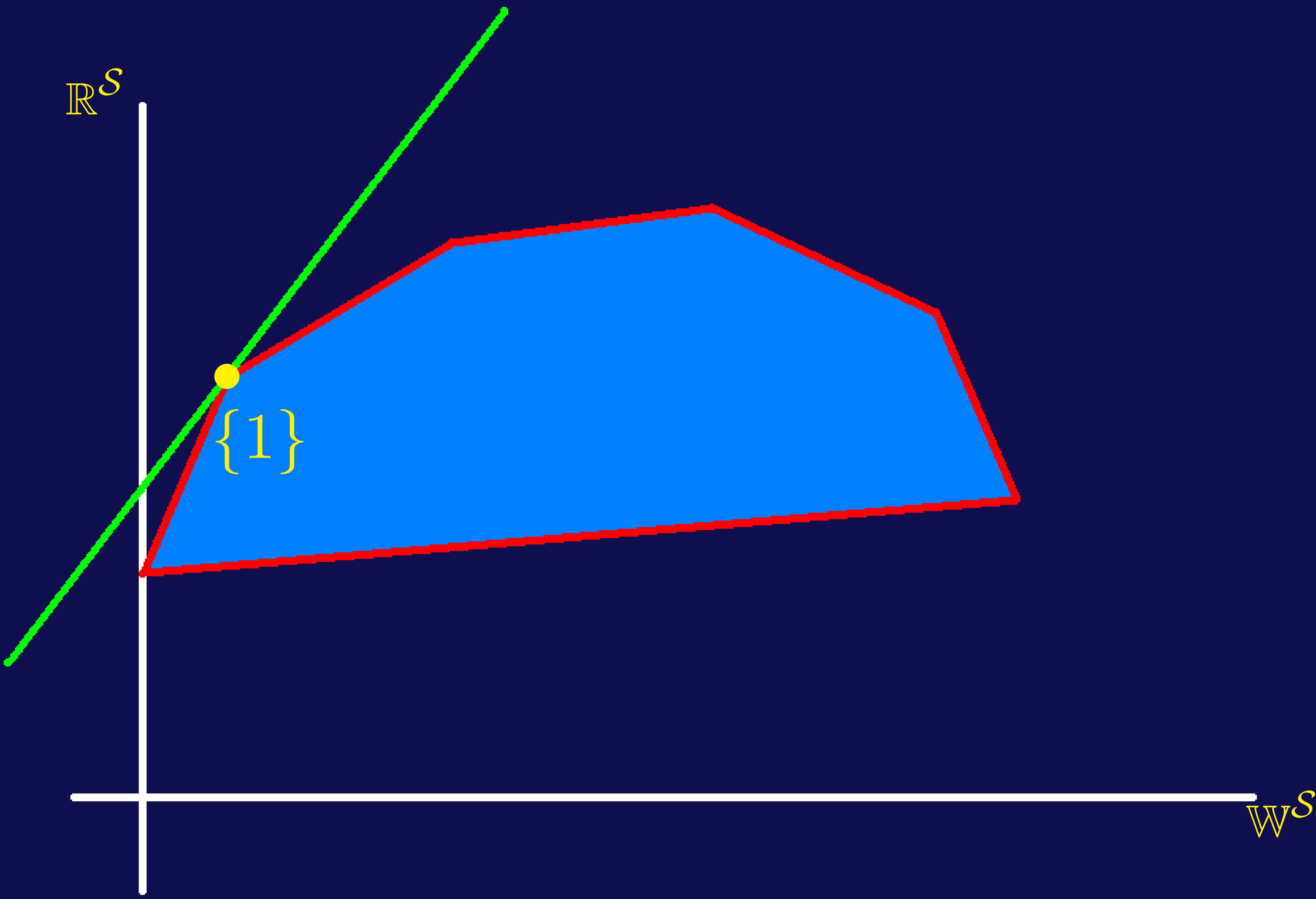
Geometric Interpretation

- $(\mathbb{W}_n^{\mathcal{S}}, \mathbb{R}_n^{\mathcal{S}})$ gives rise to 2-dim. performance region
- Indexability means the performance region is convex
- Optimal (threshold) policies are extreme points of the upper boundary of the performance region
- Index values are slopes of the upper boundary
- Indexability is sort of a dual concept to threshold policies
  - ▷ but not equivalent!

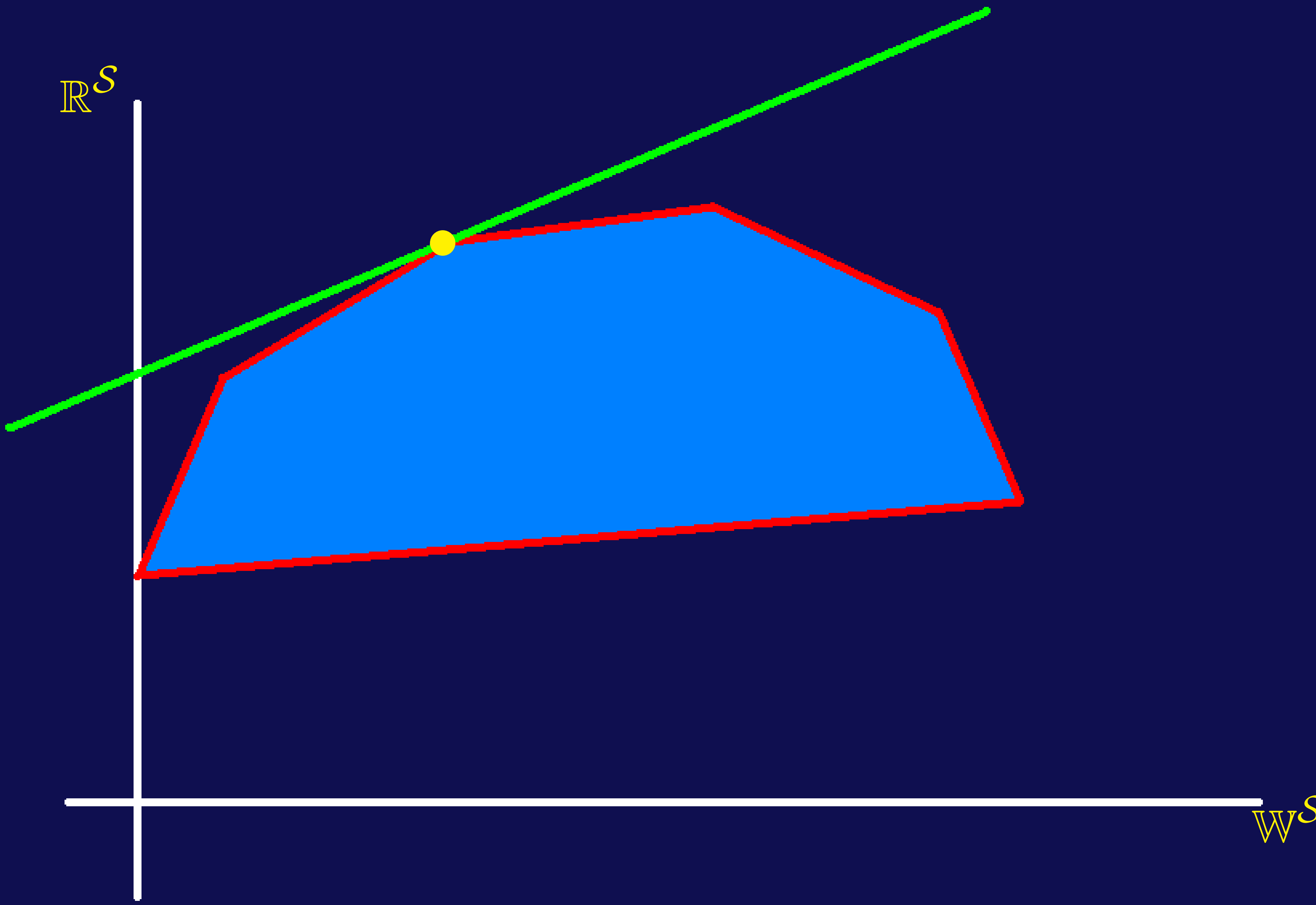
Performance Region



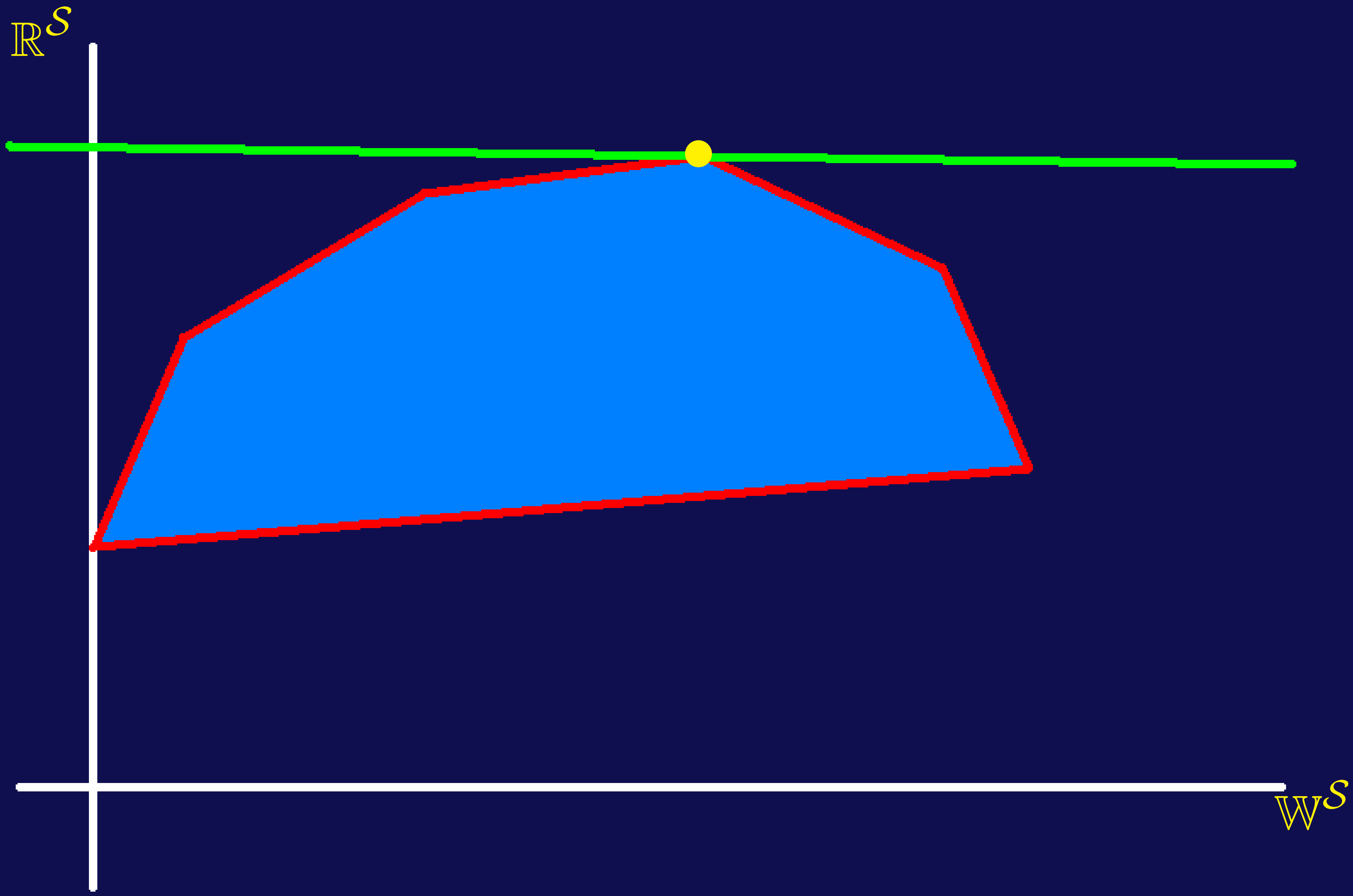
Performance Region



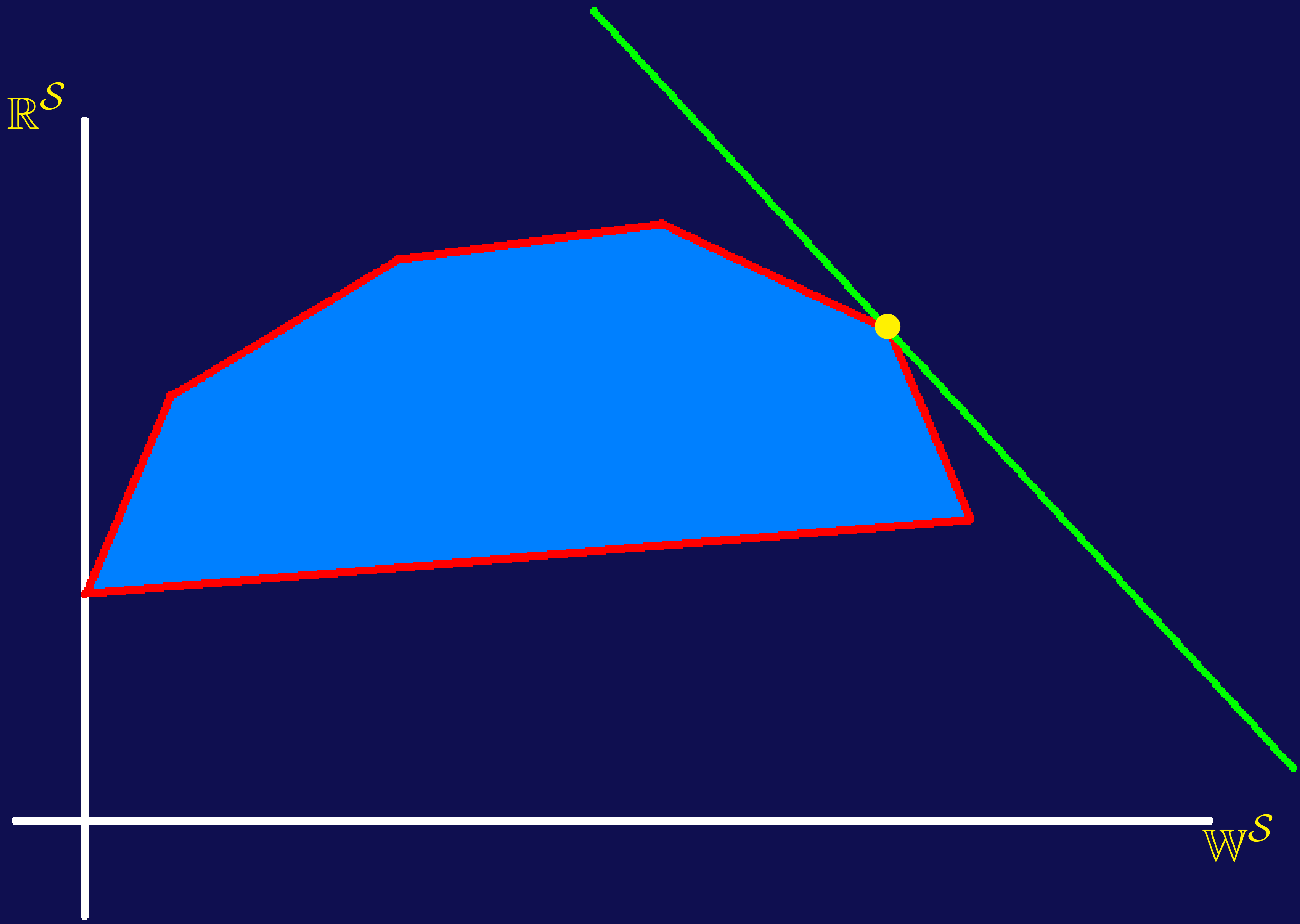
Performance Region



Performance Region



Performance Region



Performance Region