Optimal Dynamic Promotion and the Knapsack Problem for Perishable Items

Peter Jacko

INFORMS RMP, Montréal
June 19, 2008

Motivation

- Perishable product
  - ▷ deteriorating product with associated deadline after which it becomes worthless, if not sold
  - ▷ arises in food industry ("best before" date), fashion industry (seasonal goods), etc.
- How to select perishable products to be promoted?
  - ▷ cannot ignore time to go!
  - ▷ likely to be PSPACE-hard
- Similar problems in task management, project selection

Perishable Products

- With “increasing” demand
  - ▷ utility obtained at or after the deadline
  - ▷ e.g., transportation tickets, concert tickets, trips
  - ▷ promoted at early periods, to stimulate later demand
  - ▷ promoted at very final periods (last-minute)
- With “decreasing” demand
  - ▷ utility obtained before the deadline
  - ▷ e.g., grocery items, seasonal goods
  - ▷ promoted at final periods, to correct for wrong planning and pricing

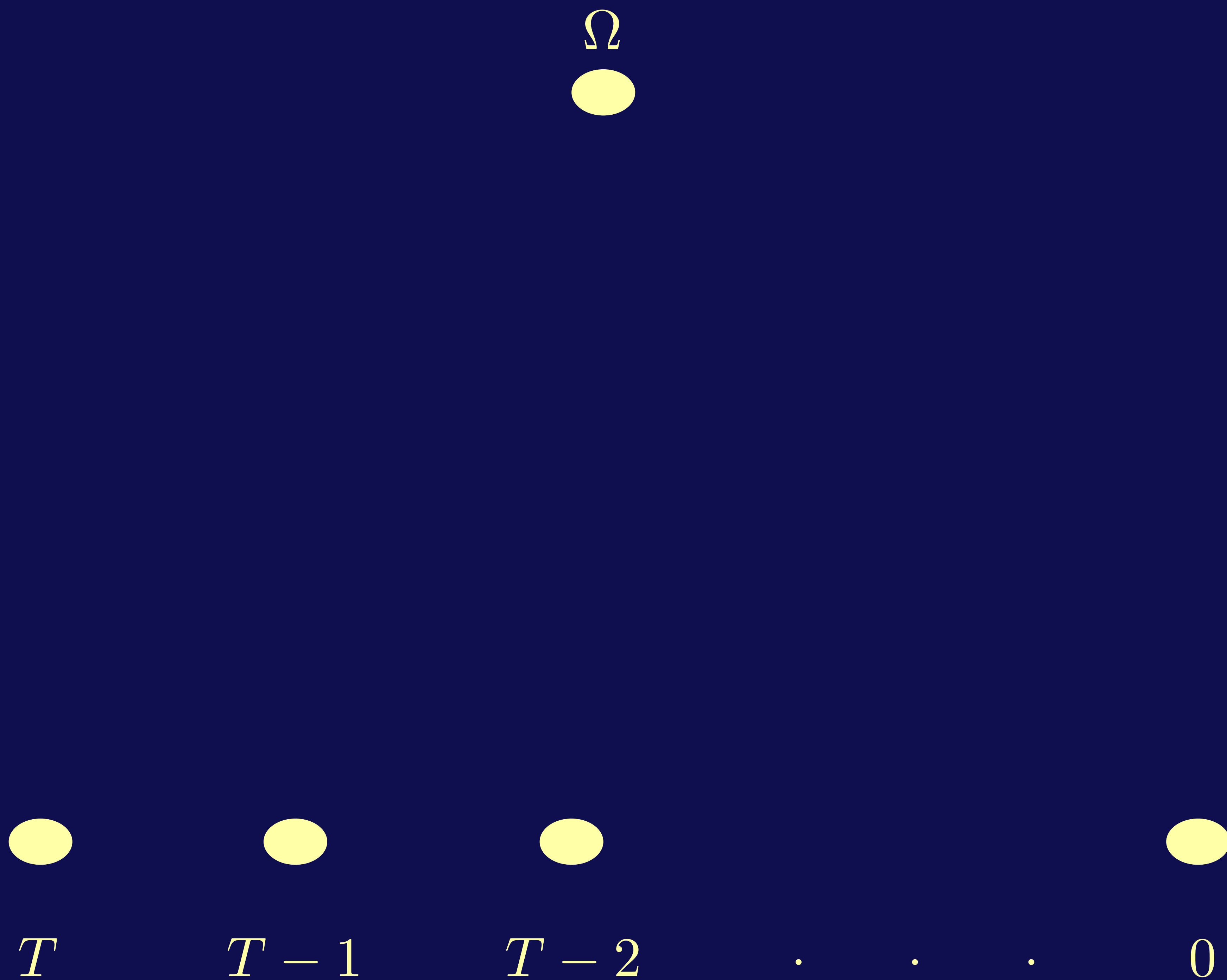
Modeling Outline

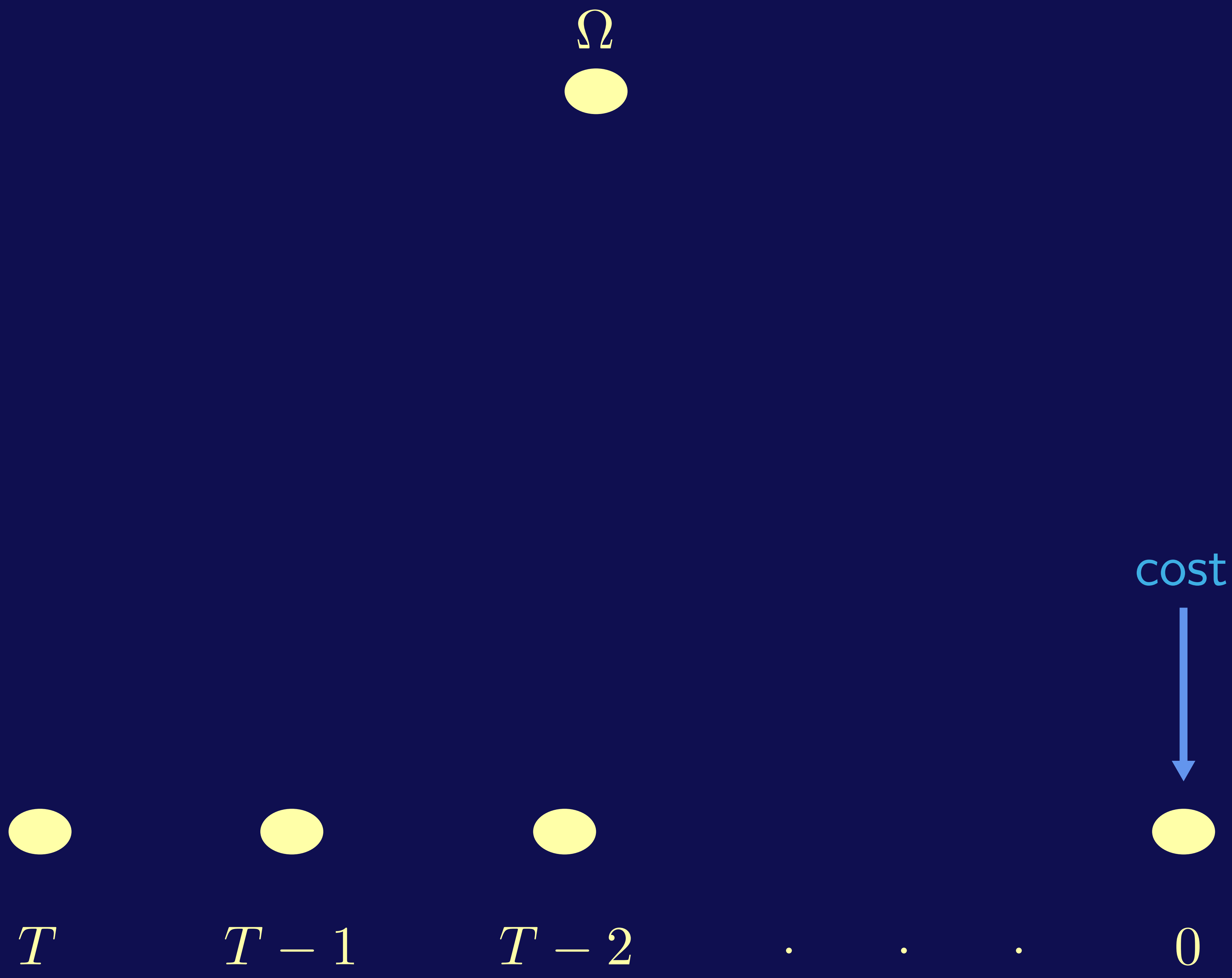
- Single-item case: Optimal Dynamic Promotion
  - ▷ marginal productivity indices (MPI)
  - ▷ promote iff MPI is larger than promotion cost
- Inventory case
  - ▷ MPI policy: calculate MPI of each unit and promote iff MPI is larger than promotion cost
- Network case: Knapsack Problem for Perishable Items
  - ▷ MPI policy: calculate MPI of each unit and solve a knapsack problem with MPIs as item values

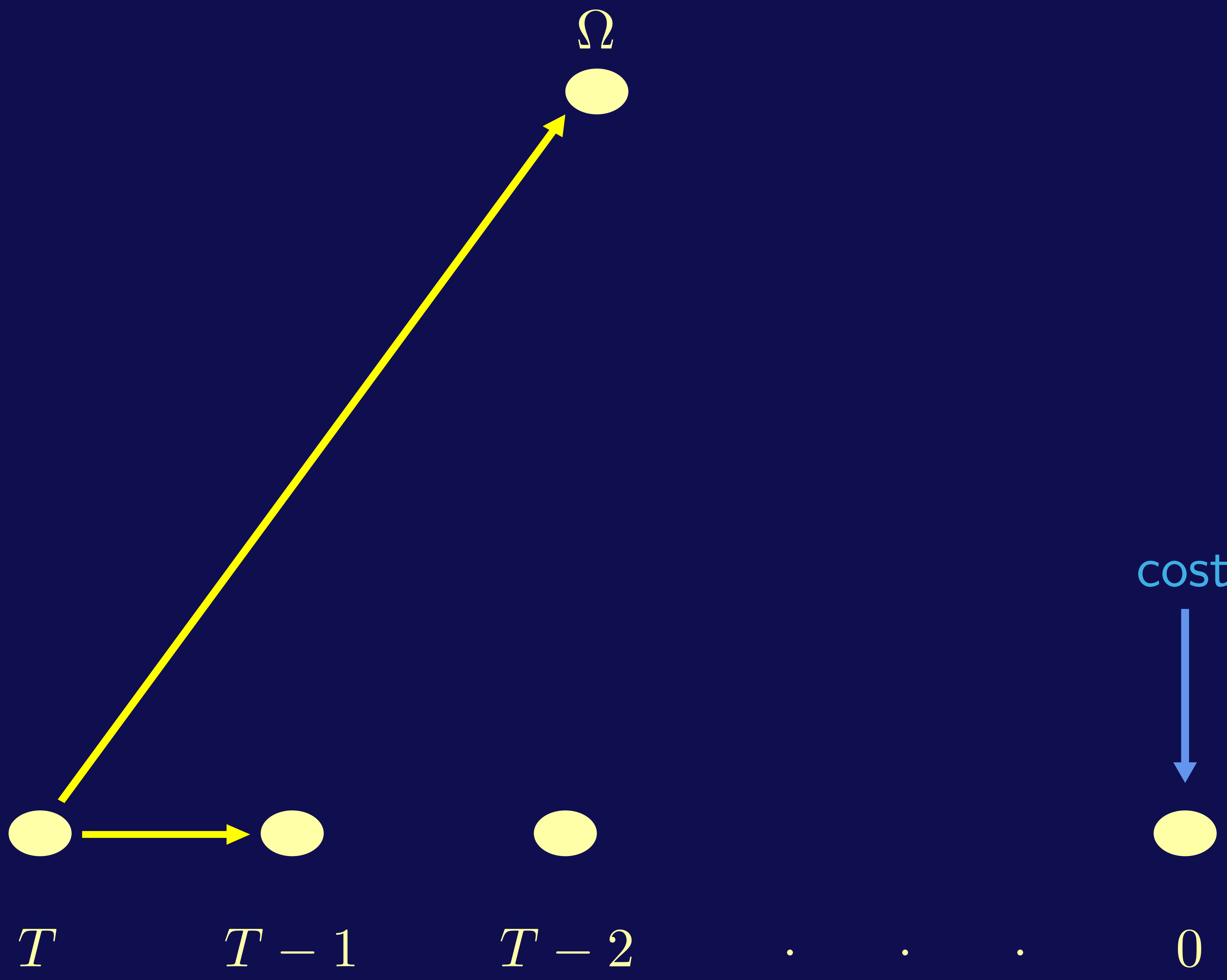
Characterization of a Perishable Item

- Decision moments: $s = T, T-1, \ldots, 1$
  - ▷ occupies space w
  - ▷ if promoted, it remains unsold with probability p
  - ▷ if not promoted, it remains unsold with probability $q_s > p$
  - ▷ once sold, it never resurrects

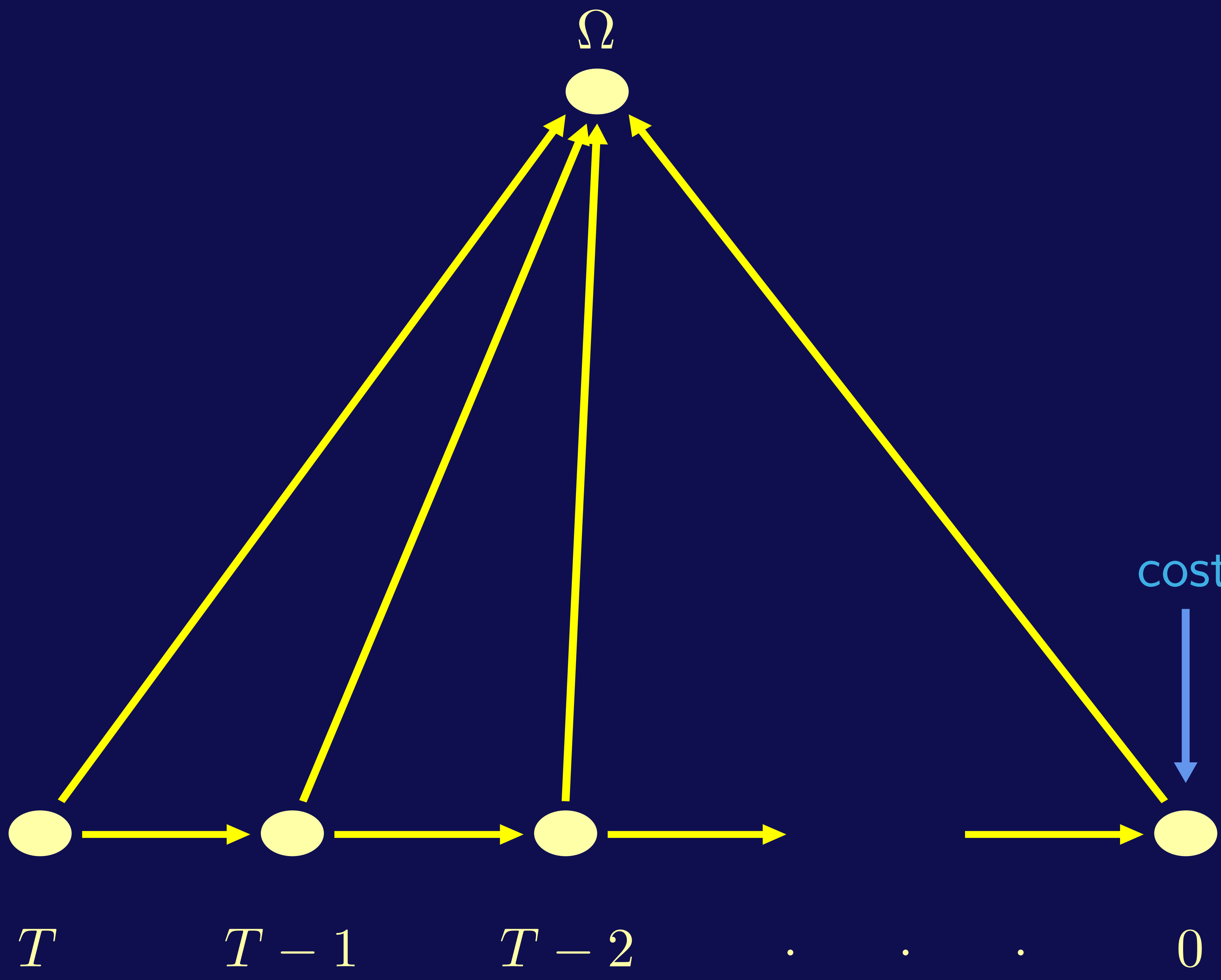
- Deadline: $s = 0$
  - ▷ pay cost $c > 0$ if not sold ("bad" state)
  - ▷ no cost if sold ("good" state)


Ω
T
$T-1$
$T-2$
$\cdot \quad \cdot \quad \cdot$
0


Ω
cost
T
$T-1$
$T-2$
$\cdot \quad \cdot \quad \cdot$
0



Ω
cost
T
$T-1$
$T-2$
. . .
0


Ω
cost
T
$T-1$
$T-2$
$\cdot \quad \cdot \quad \cdot$
0

Perishable Item as Markov Decision Chain

- States:
  - ▷ $t \in \{T, T-1, \ldots, 1\}$: unsold and t periods before deadline
    - – actions to choose: promote/don't promote
    - – no cost
  - ▷ 0: unsold and perishing (exactly at deadline)
    - – no action to choose
    - – cost c
  - ▷ Ω: sold or perished (terminal state)
    - – no action to choose
    - – no cost

The Problem

- Consider promotion cost ν per period if promoting
- Minimize the expected total β-discounted cost:

$$\min_{\pi} \mathbb{E}_T^{\pi} \left[-\beta^T r(0) + \nu \sum_{s=0}^{T-1} \beta^s w(T-s) \right]$$

or simply $$\min_{\pi} -R_T^{\pi} + \nu W_T^{\pi}$$

- $r(0)$ is the deadline reward ($-c$ if unsold, 0 if sold)
- $w(s)$ is the "work" at time s (1 if promoting, 0 if not)

Intuitive Solution

- Expected properties of optimal solution:
  - ▷ if optimally promoted for ν,
    then optimally promoted for $\nu' < \nu$
  - ▷ if optimally promoted at t,
    then optimally promoted at $t-1$

- Aim: To each state t assign priority index ν_t so that it is optimal to promote at state t whenever $\nu_t > \nu$

- We expect $\nu_t < \nu_{t-1}$ (increasing as deadline approaches)

Marginal Productivity Index (MPI)

- Stationary policy $\pi \equiv$ promotion set $\mathcal{S} \subseteq \mathcal{T}$
- MPI ν_t for state t must satisfy: if $\nu = \nu_t$,
  both promoting and not promoting are optimal
- So, there is a promotion set $\mathcal{S}_t$ for state t such that

$$-R_t^{\mathcal{S}_t \cup \{t\}} + \nu_t W_t^{\mathcal{S}_t \cup \{t\}} = -R_t^{\mathcal{S}_t \setminus \{t\}} + \nu_t W_t^{\mathcal{S}_t \setminus \{t\}}$$

- Therefore, if denominator is nonzero,

$$\nu_t = \frac{R_t^{\mathcal{S}_t \cup \{t\}} - R_t^{\mathcal{S}_t \setminus \{t\}}}{W_t^{\mathcal{S}_t \cup \{t\}} - W_t^{\mathcal{S}_t \setminus \{t\}}} \quad \text{for some } \mathcal{S}_t$$

Interpretation of MPI

- Marginal rate of substitution for promoting
- Marginal productivity rate of promoting with respect to not promoting
- Expected marginal reward divided by marginal work
- Evolution of indices:
  - ▷ $c\mu$-rule (1960s); Gittins' index (1970s)
  - ▷ Whittle's index (1988)
  - ▷ MPI: Niño-Mora (2000s)

MPI for Perishable Item

- Under a regularity condition for $(q_s), p, \beta$ we have

$$\mathcal{S}_t = \{t, t-1, \ldots, 1\}$$

- So we have

$$\nu_t = \frac{R_t^{\mathcal{S}_t} - R_t^{\mathcal{S}_{t-1}}}{W_t^{\mathcal{S}_t} - W_t^{\mathcal{S}_{t-1}}}$$

- After some algebra, closed-form formula:

$$\nu_t = \frac{c\beta(q_t - p)(\beta p)^{t-1}}{1 - \beta(q_t - p)\frac{1-(\beta p)^{t-1}}{1-\beta p}}$$

MPI Properties (under Regularity Cond.)

- Positive and proportional to deadline cost c
- Increasing in q_t
- Depends only on q_t, not on whole sequence (q_s)
- Increasing as deadline approaches: $\nu_t < \nu_{t-1}$
- Extends to undiscounted case $(\beta = 1)$

Regularity Condition

- Regularity condition: for all $s \in \{T, T-1, \ldots, 2\}$,

$$\beta q_{s-1} - \beta p \geq (\beta q_s - \beta p)\beta q_{s-1}$$

- Holds if
  - ▷ demand is constant over time (q_s's are constant)
  - ▷ demand is nonincreasing over time ($q_s \leq q_{s-1}$)
  - ▷ demand is moderately increasing over time

Inventory of Perishable Items

- Consider J units of a perishable product
- Common demand (e.g., Poisson)
- Denote by
  - $d(j) = \mathbb{P}\{\#\ \text{customers} < j\}$
- Q: How many units should we promote?
- Try to use the MPI derived for single-unit case

Calculation of q's

- How to transform demand function into q's?
- Label the units as $1, 2, \ldots, J$
- The j-th customer buys unit labeled j (WLOG)
- Obviously $q_{j,T} = d(j)$
- $q_{j,t}$ is the conditional probability that unit j is unsold at $t-1$ given that it was unsold at t
- Therefore we have $q_{j,t} \geq d(1)$ for all j, t

MPI Policy for Inventory

- Units only differ in their q's, and

$$q_{j,t} > p \text{ for all } j, t \text{ whenever } d(1) > p$$

- So, we can assign MPI $\nu_{j,T}$ to every unit j
- By properties of MPI we have $\nu_{j,T} \geq \nu_{j-1,T}$
- Policy: Promote all units j with $\nu_{j,T} > \nu$
- If the Regularity Condition holds for each unit then this policy is optimal

Knapsack Problem for Perishable Items

- Consider I perishable products with inventories J_i
- Each unit of product i occupies space w_i
- Let W be the promotion space (knapsack)
- A dynamic and stochastic combinatorial problem
- Aim: Fill in the knapsack so that the expected aggregate total β-discounted cost is minimized

KPPI $\rightarrow$ KP Reduction

- KPPI reduces to Knapsack Problem
  when $T_i = J_i = q_i = 1$, $p_i = 0$, and $c_i = v_i$
- (KP) is NP-hard $\Longrightarrow$ KPPI is at least NP-hard
- In fact, KPPI seems to be PSPACE-hard

Dynamic Programming Formulation

$$D_T(\boldsymbol{z}_T) = \sum_{i \in \mathcal{I}_T^0} c_i z_{(T,i)}$$

$$D_s(\boldsymbol{z}_s) = \sum_{i \in \mathcal{I}_s^0} c_i z_{(s,i)} + \min_{\substack{\boldsymbol{y}_s \le \boldsymbol{z}_s^+ \\ \sum_{i \in \mathcal{I}_s^+} w_i y_{(s,i)} \le W}} \left\{ \sum_{\boldsymbol{m}_s \le \boldsymbol{z}_s^+} \mathbb{P}^{\boldsymbol{y}_s}[\boldsymbol{m}_s] D_{s+1}(\boldsymbol{z}_s^+ - \boldsymbol{m}_s) \right\}$$

- Solving a system of an exponential number of equations for an exponential number of vectors $\boldsymbol{z}_s$ at every stage
  - ▷ tractability problem: curse of dimensionality
  - ▷ no interpretation

MPI Policy for KPPI

- Solve 0-1 Knapsack Problem for items (i,j):

$$\max_{\boldsymbol{x}} \sum_{(i,j)} \nu_{(i,j),T} x_{ij}$$

$$\text{subject to } \sum_{(i,j)} w_i x_{ij} \leq W \qquad \text{(KP)}$$

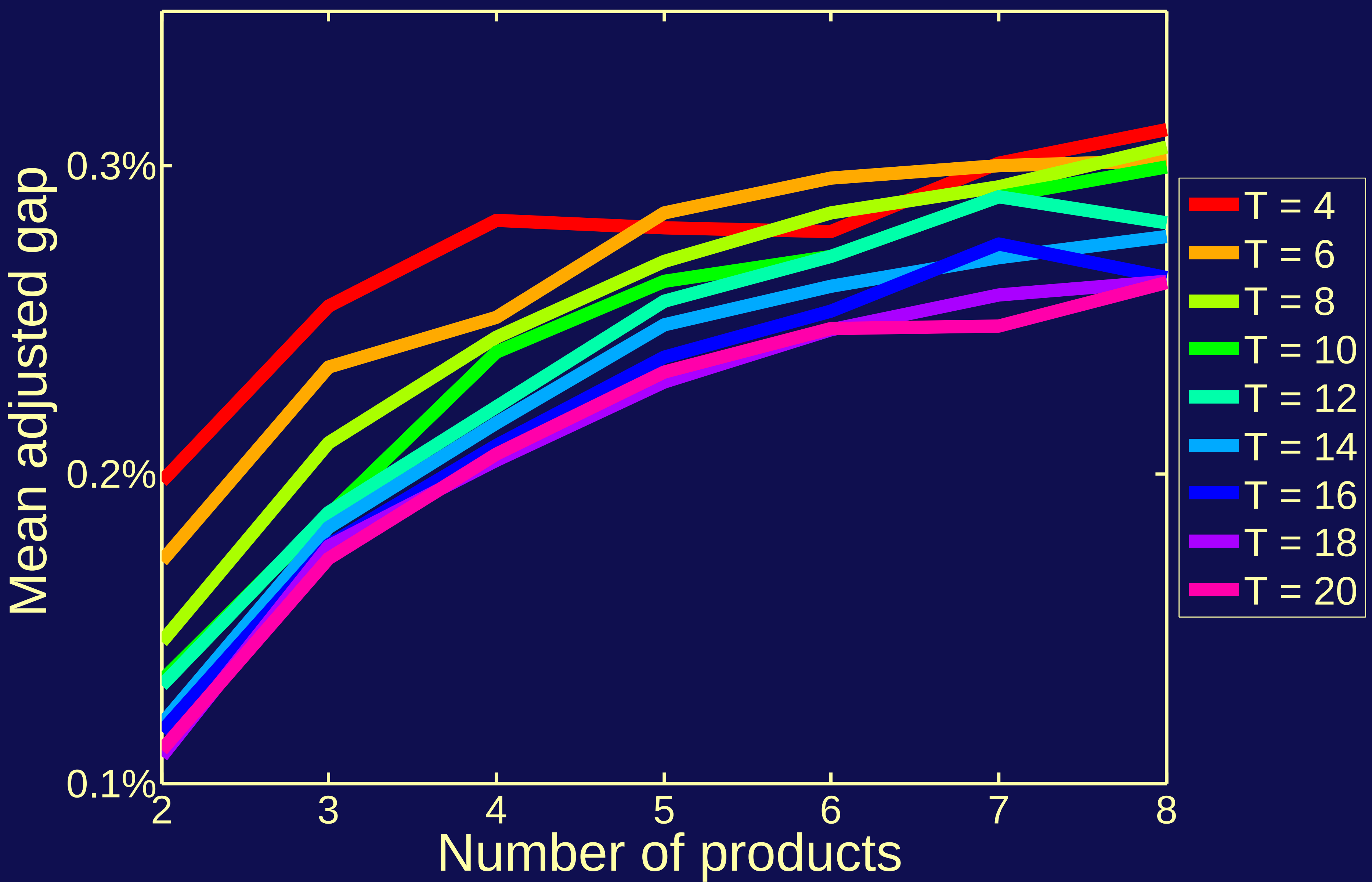
$$x_{ij} \in \{0,1\} \text{ for all } (i,j)$$

- Policy: Promote $y_i = \sum_{j=1}^{J_i} x_{ij}$ units of each product $i \in \mathcal{I}$

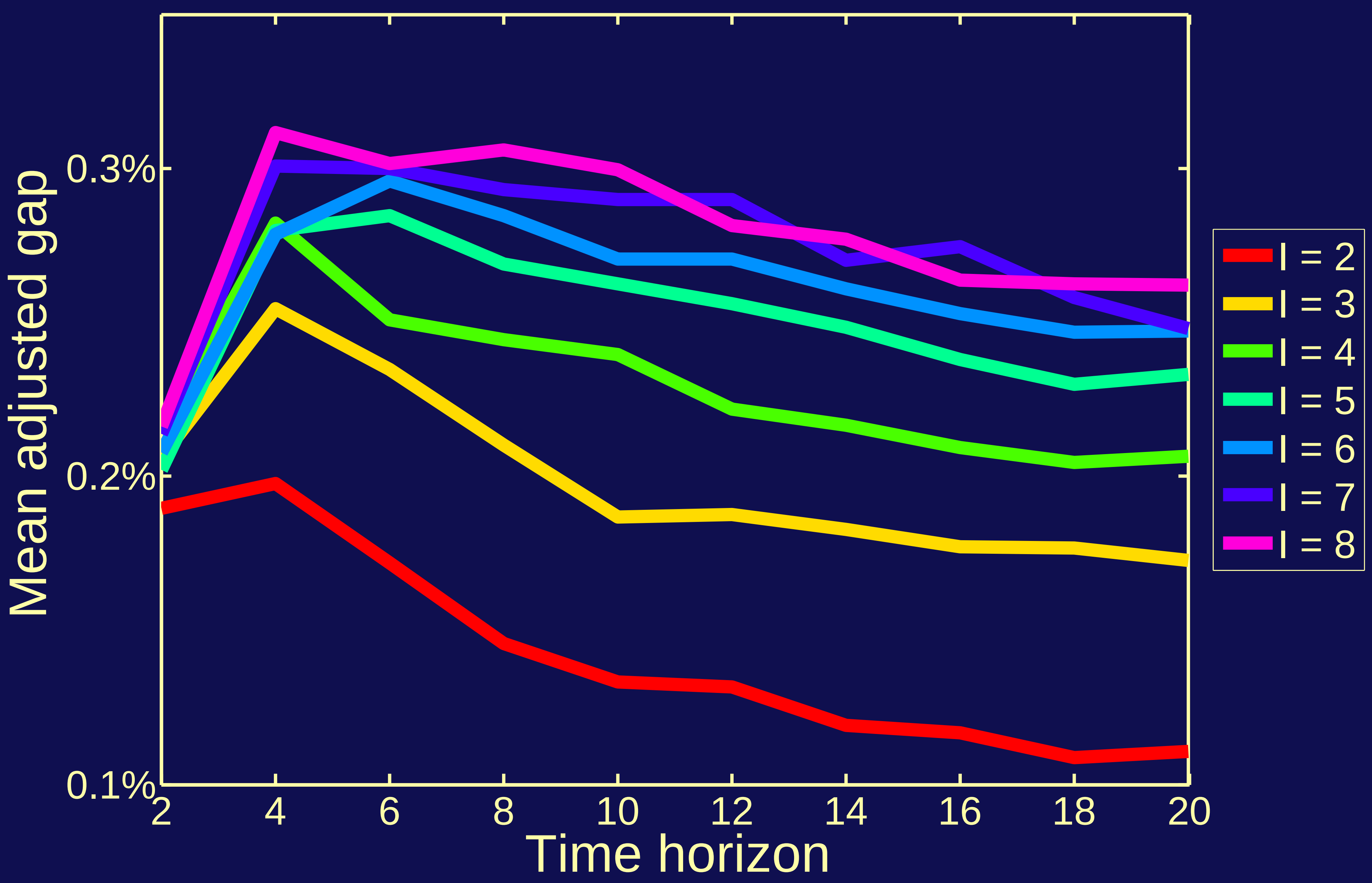
Simulation Study

- Randomly generated instances, $c_i, w_i \in [10, 50]$
- Let $T = \max\{T_i\}$ be the time horizon
- Non-conservative inventory: $\frac{1}{2}\lambda_i T_i \leq J_i \leq \frac{3}{2}\lambda_i T_i$
  and $1 \leq J_i \leq J$
- Knapsack volume W less than 30% of total volume
- Experiment 1: $J = 1$ and I, T varying (10000 instances)
- Experiment 2: $I = 2$ and J, T varying (10000 instances)
- Experiment 3: $I = 5, J = 9, T = 10$ (20 instances)

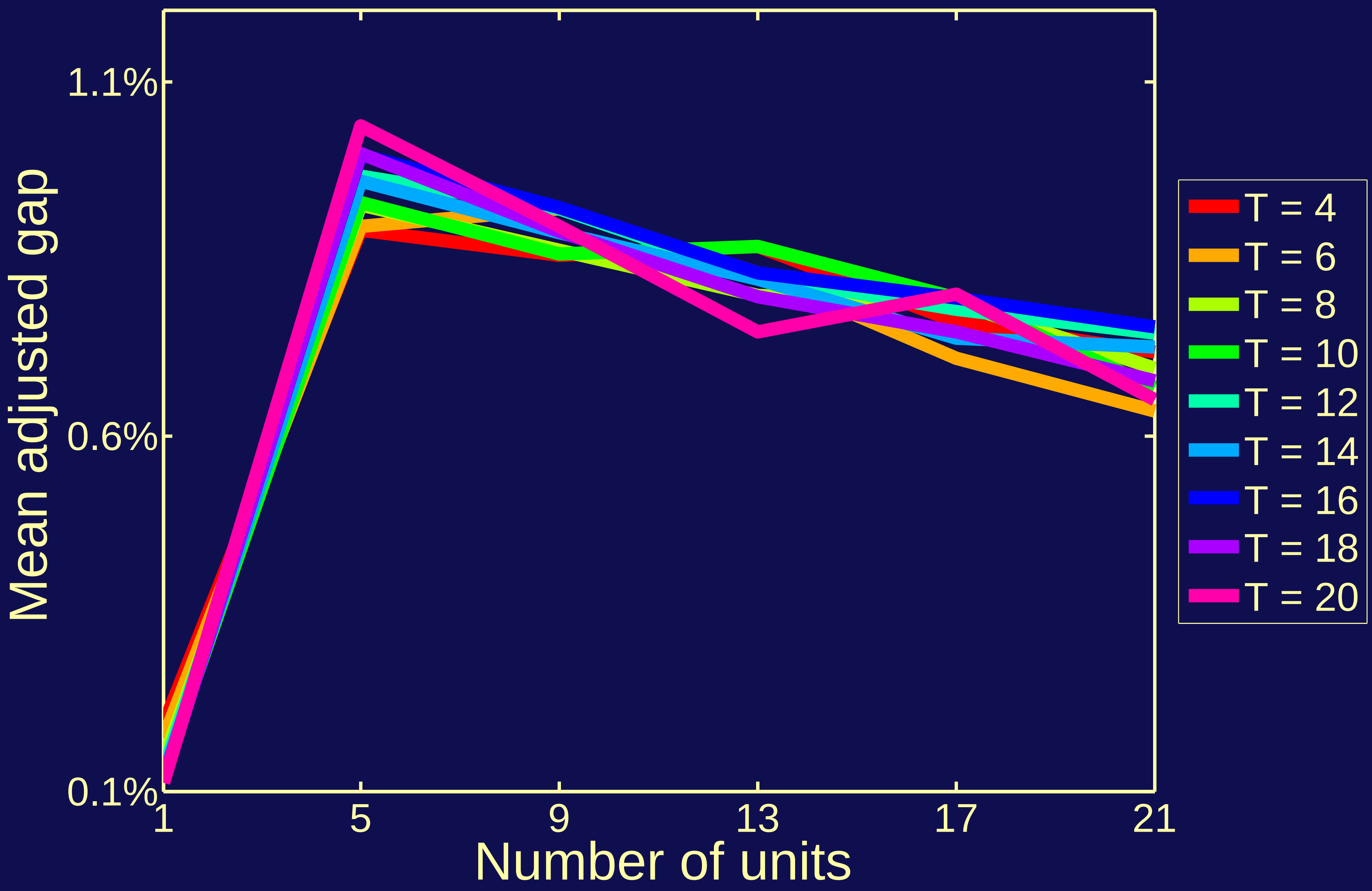
Performance of MPI Policy ($J = 1$)



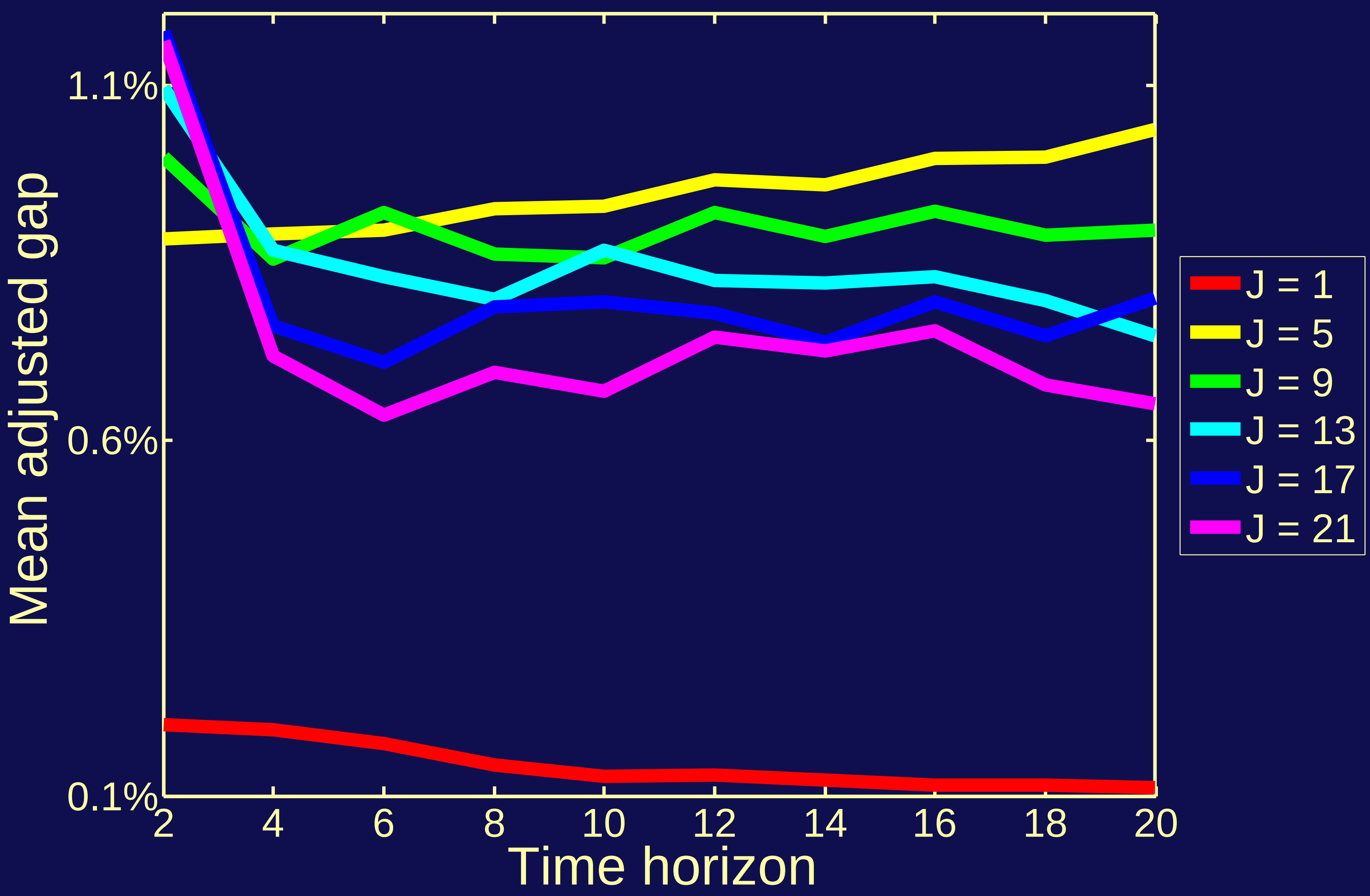
Performance of MPI Policy ($J = 1$)



Performance of MPI Policy ($I = 2$)



Performance of MPI Policy ($I = 2$)



Relative Suboptimality Gap

$$\mathrm{rsg}(\pi) = \frac{C^{\pi} - C^{\min}}{C^{\min}}$$

- Takes values between 0 (achievable) and ∞ (?)
- For what values of $\mathrm{rsg}(\pi)$ is π a "good" policy?
- Generally accepted: below 5%
- Is it a good measure for bounded-from-above problems?
- What if $\mathrm{rsg}(\max) = 10\%$? What if $C^{\min} \approx 0$?

Adjusted Relative Suboptimality Gap

$$\mathrm{arsg}(\pi) = \frac{C^{\pi} - C^{\min}}{C^{\max} - C^{\min}}$$

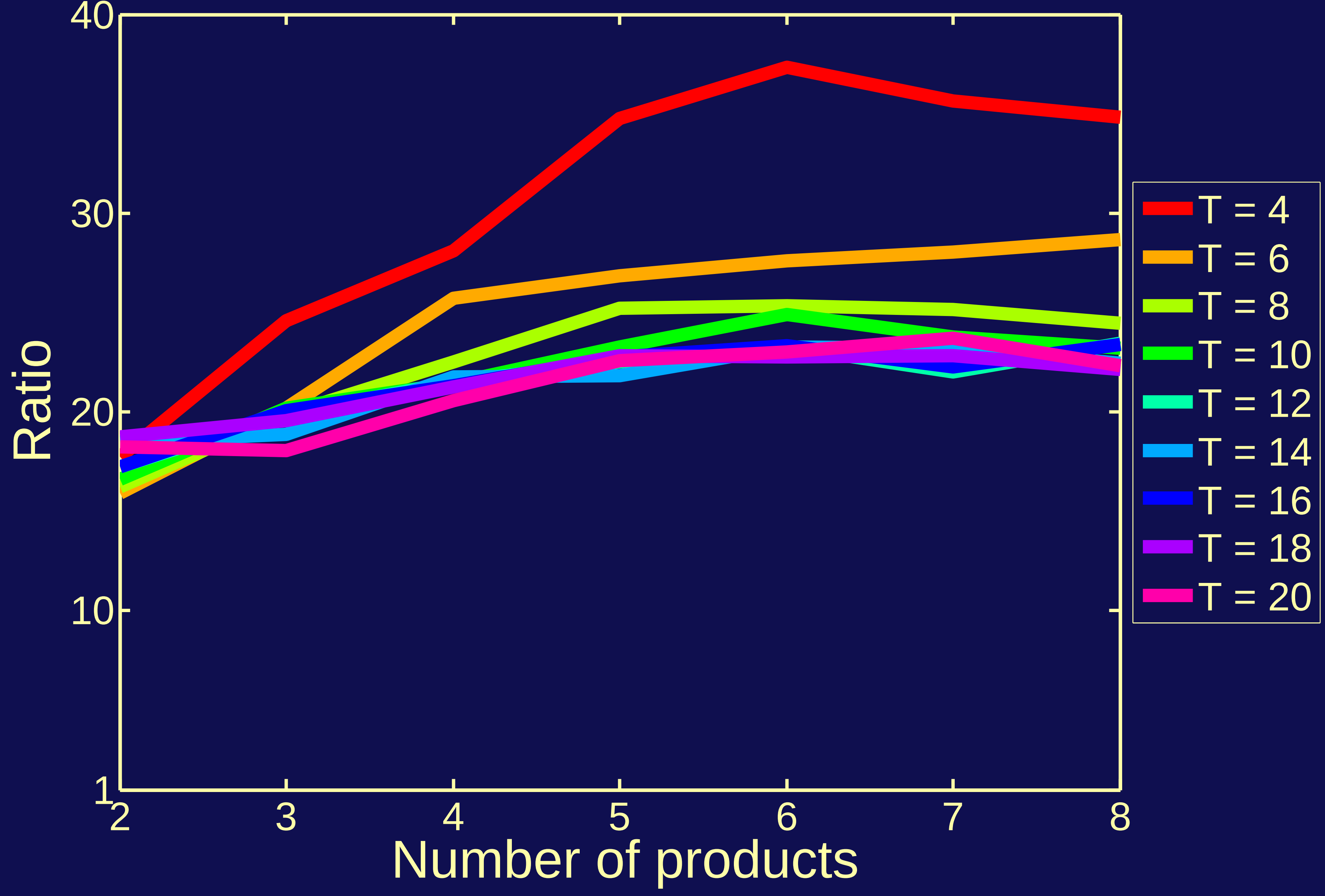
- Takes values between 0 and 1 (both achievable)
- Suitable if $C^{\max}$ can be calculated and is not ∞
- π_1 is better than π_2 following rsg $\equiv$ π_1 is better than π_2 following arsg
- Interpretation: Fraction of absolute gap $C^{\max} - C^{\min}$ that is not avoided

Other Heuristics

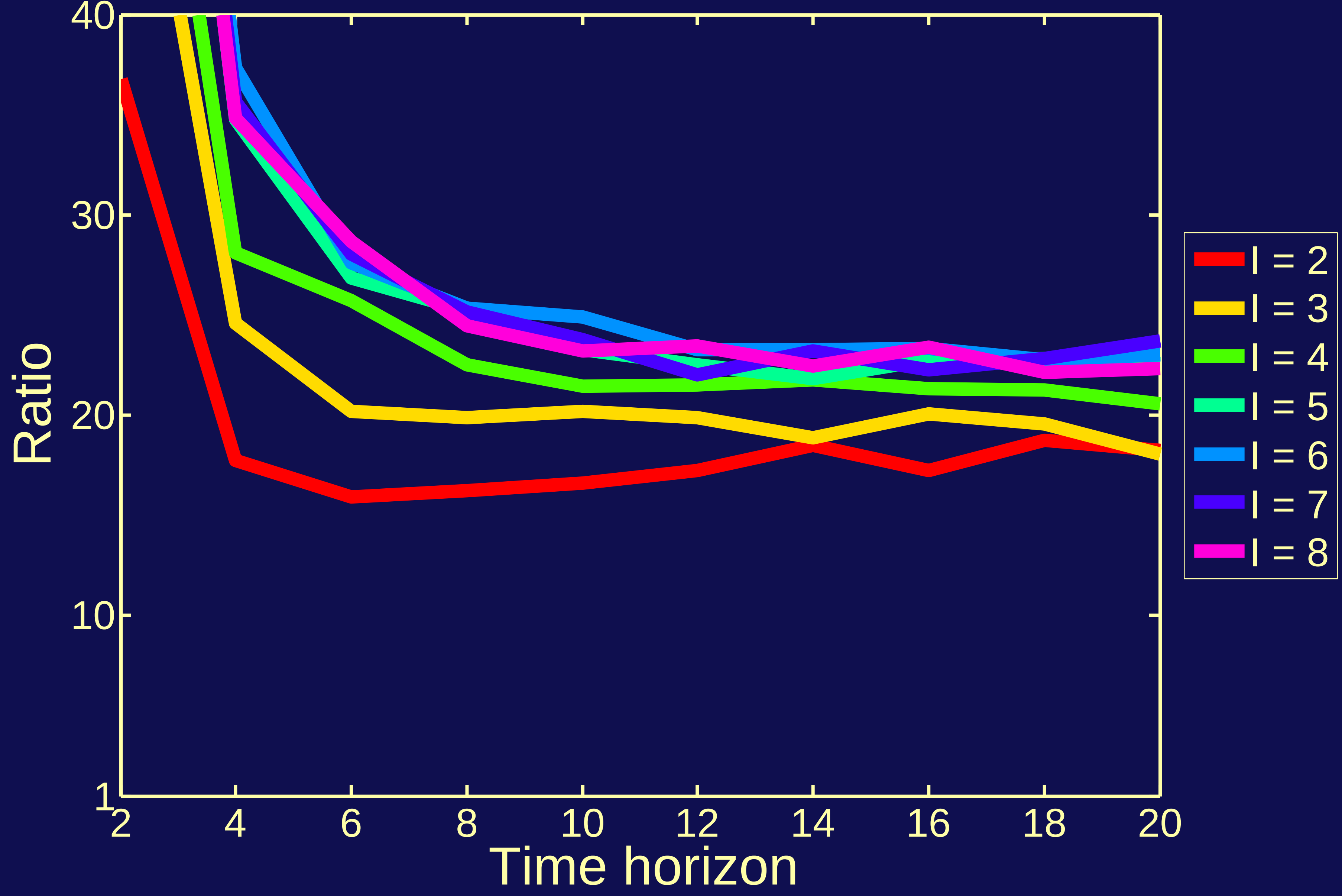
- EDF policy: Products with Earlier Deadline go First
  - ▷ naive benchmark policy
- GRE policy: Solving (KP) by greedy heuristic
  - ▷ to be used when (KP) is computationally intractable
  - ▷ based on Niño-Mora (2002)
- Define performance ratio of policy π

$$\mathrm{ratio}(\pi) = \frac{\mathrm{mean}\,(\mathrm{arsg}(\pi))}{\mathrm{mean}\,(\mathrm{arsg}(\mathsf{MPI}))}$$

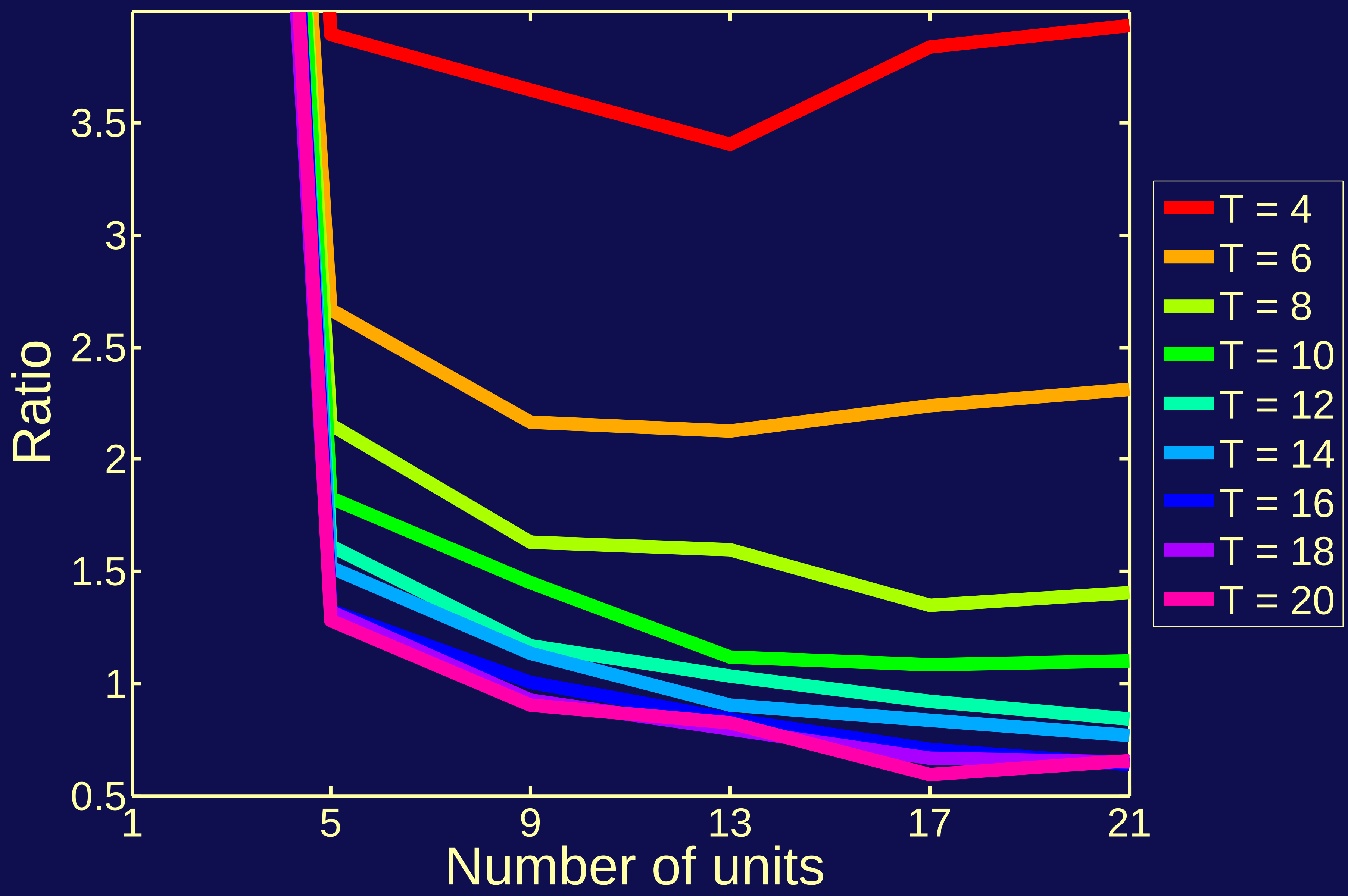
Performance Ratio of EDF Policy ($J = 1$)



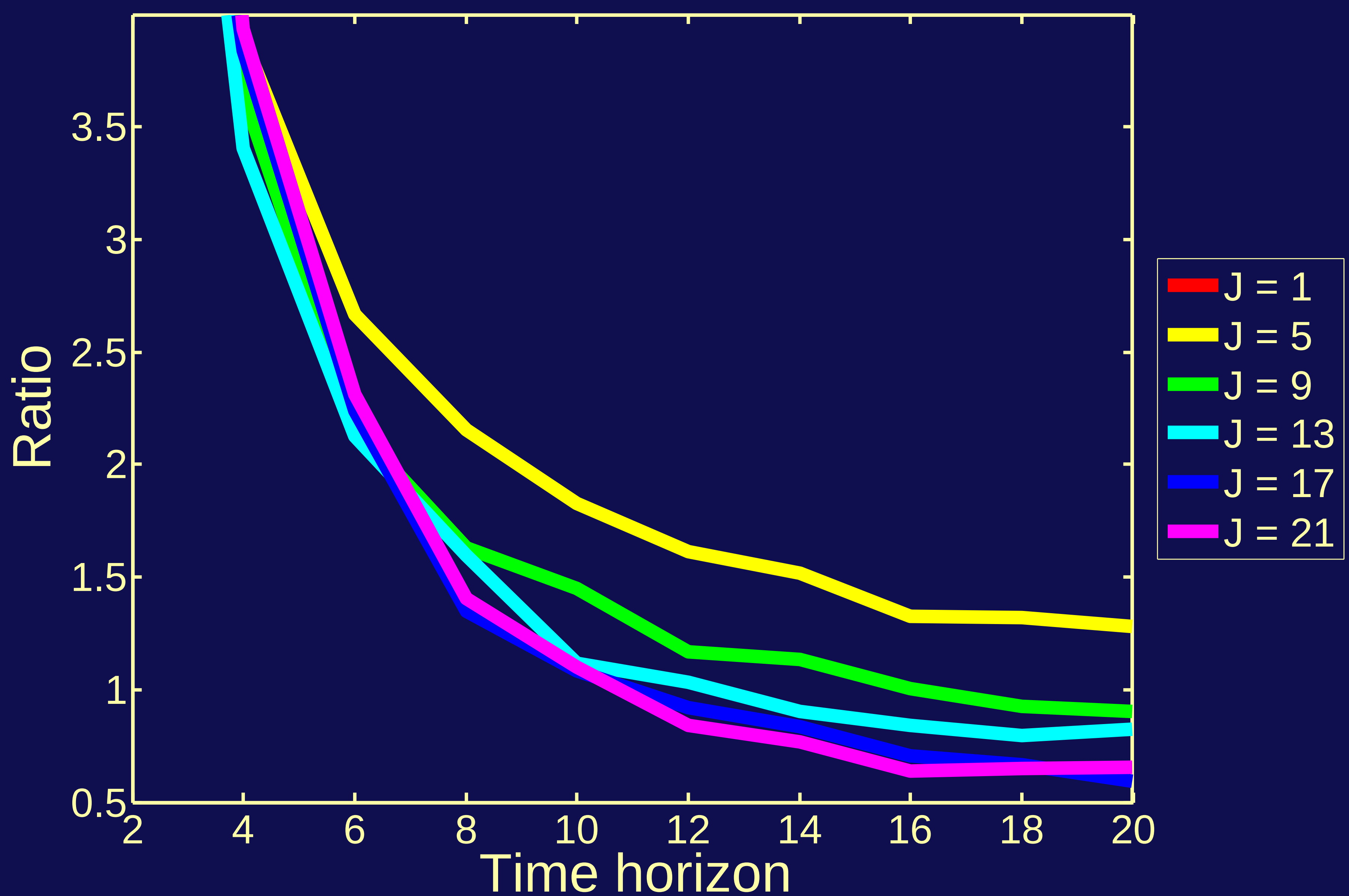
Performance Ratio of EDF Policy ($J = 1$)



Performance Ratio of EDF Policy ($I = 2$)



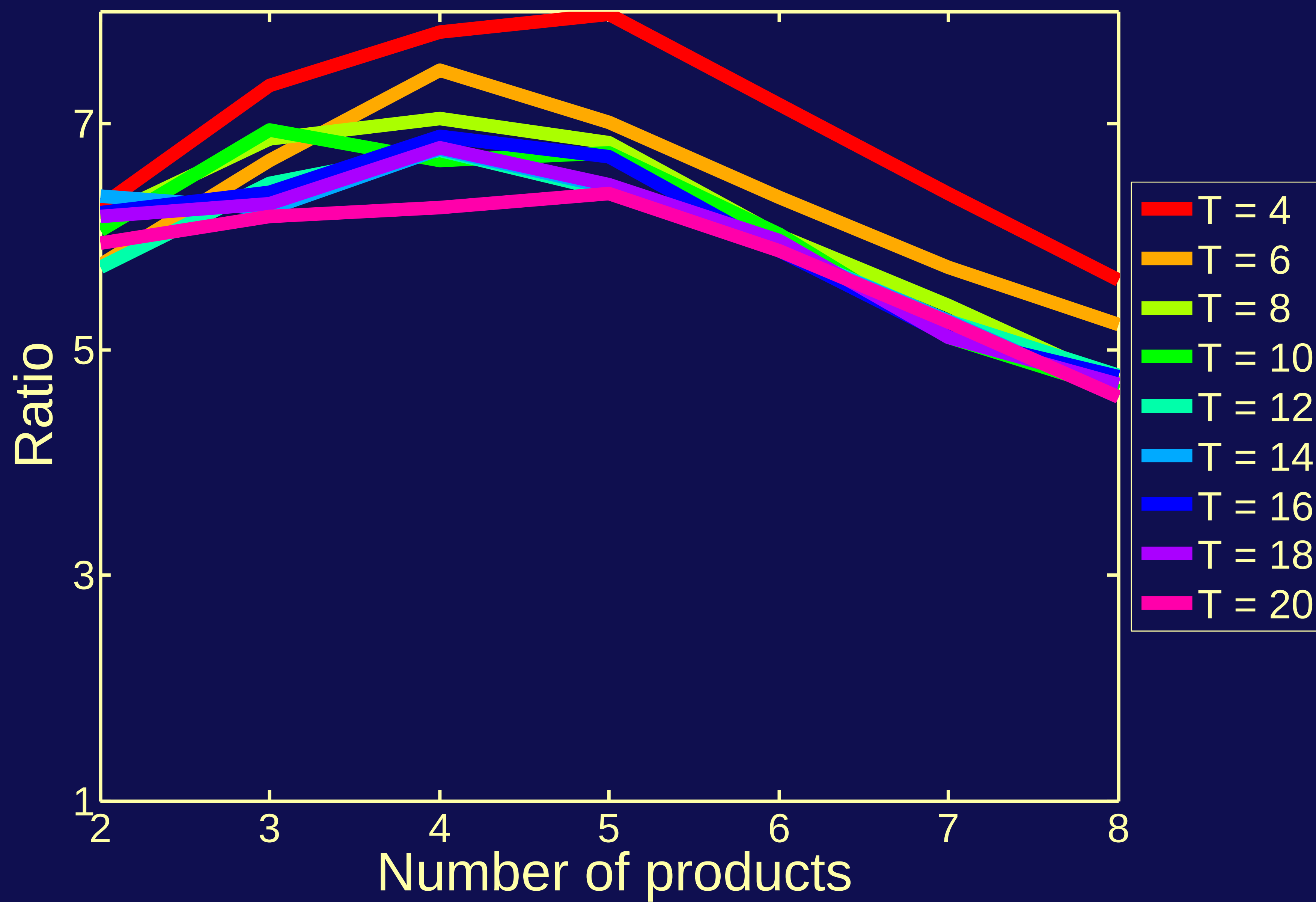
Performance Ratio of EDF Policy $(I = 2)$



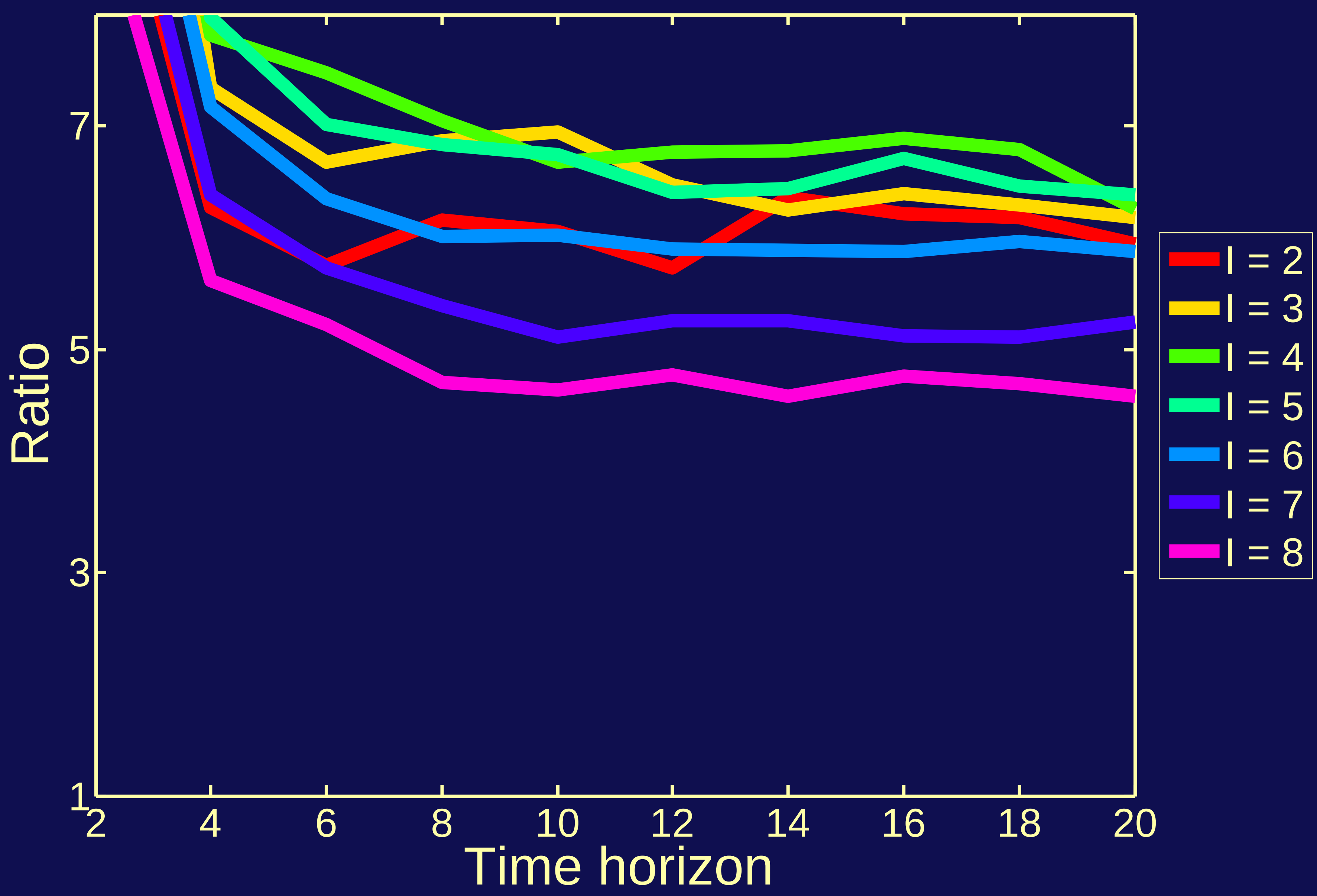
EDF Policy Summary

- Experiment 3: $I = 5, J = 9, T = 10$ (20 instances)
  - $\text{ratio}(\pi) = 3.58$
- Significantly inferior to MPI policy in all relevant cases

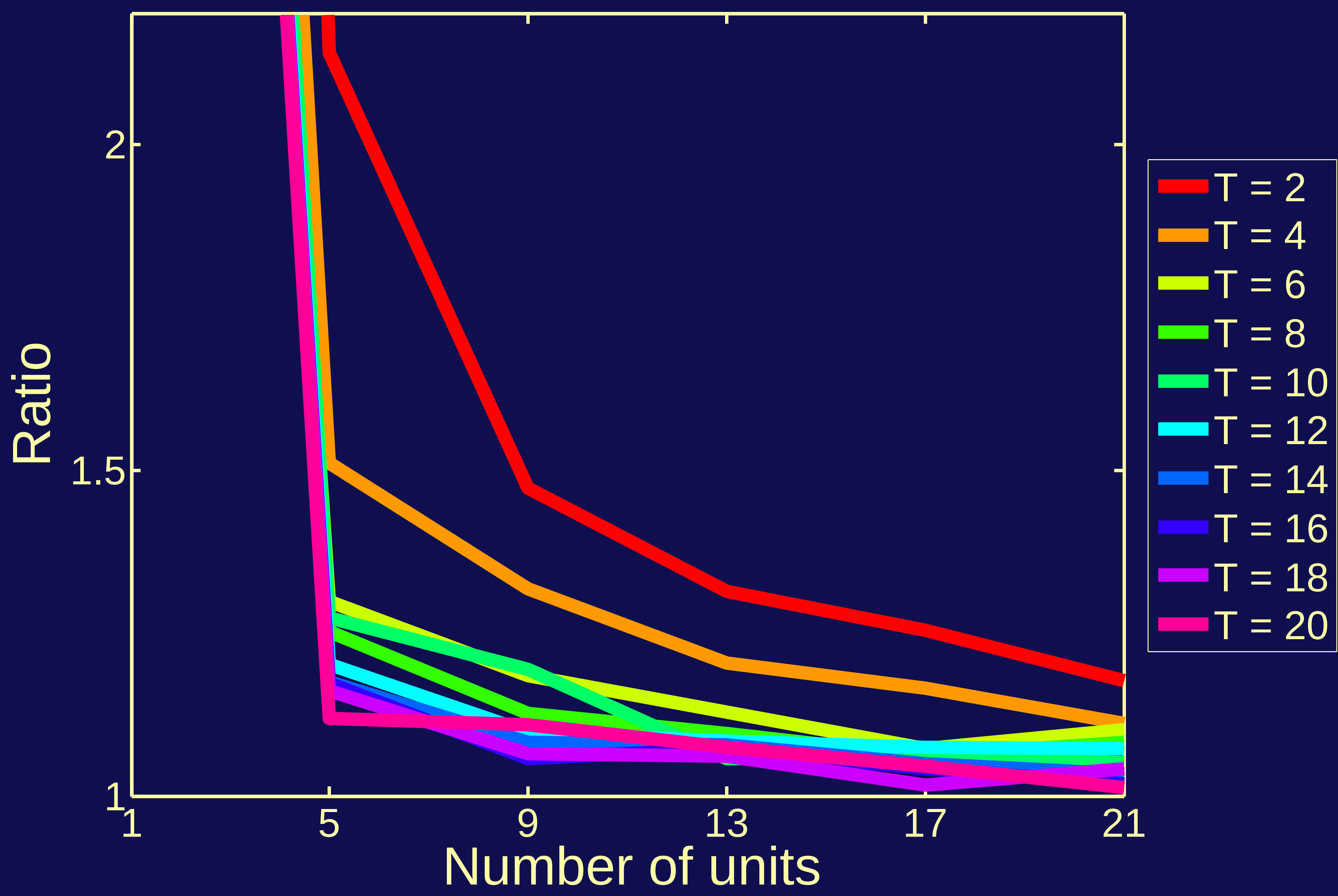
Performance Ratio of GRE Policy $(J = 1)$



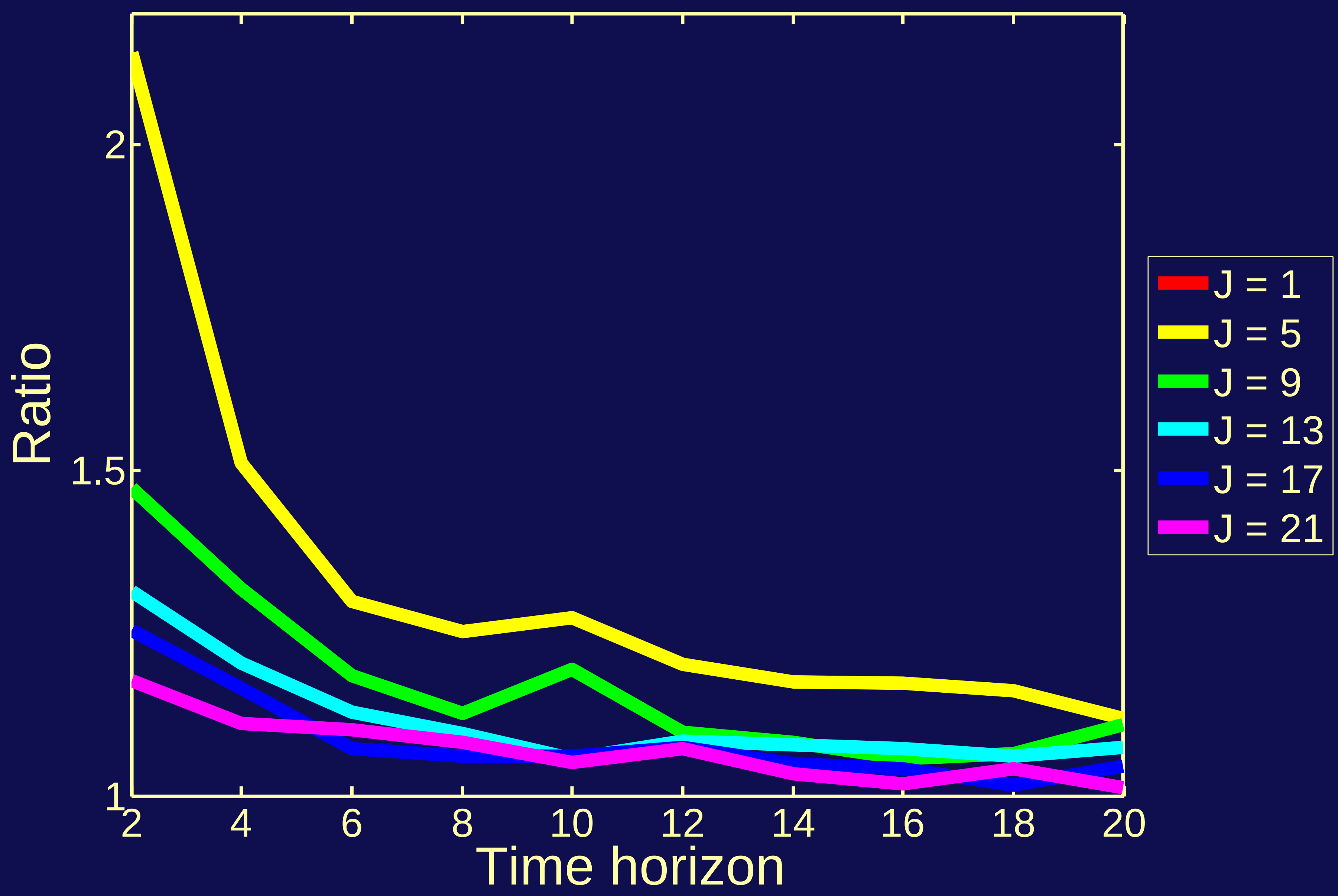
Performance Ratio of GRE Policy $(J = 1)$



Performance Ratio of GRE Policy $(I = 2)$



Performance Ratio of GRE Policy $(I = 2)$



GRE Policy Summary

- Experiment 3: $I = 5, J = 9, T = 10$ (20 instances)
  - $\mathrm{ratio}(\pi) = 1.59$
- Outperformed by MPI policy in all cases
- Suggesting convergence to MPI policy
  for large values of parameters

Summary

- We have presented:
  - ▷ a nontrivial problem with closed-form MPI
  - ▷ an optimal policy for inventory of perishable items
  - ▷ a new index-policy heuristic achieving nearly-optimal performance for KPPI
  - ▷ applicable to a variety of ad-hoc restrictions
  - ▷ new policy performance measure for bounded problems
- What to do: extensions and other applications

Thank you for your attention