Simon Fraser University

Complexity Analysis of Optimal Stationary Call Admission Policy and Fixed Set Partitioning Policy for OVSF-CDMA Cellular Systems

Daniel Lee

Muhammad Naeem

Chingyu Hsu

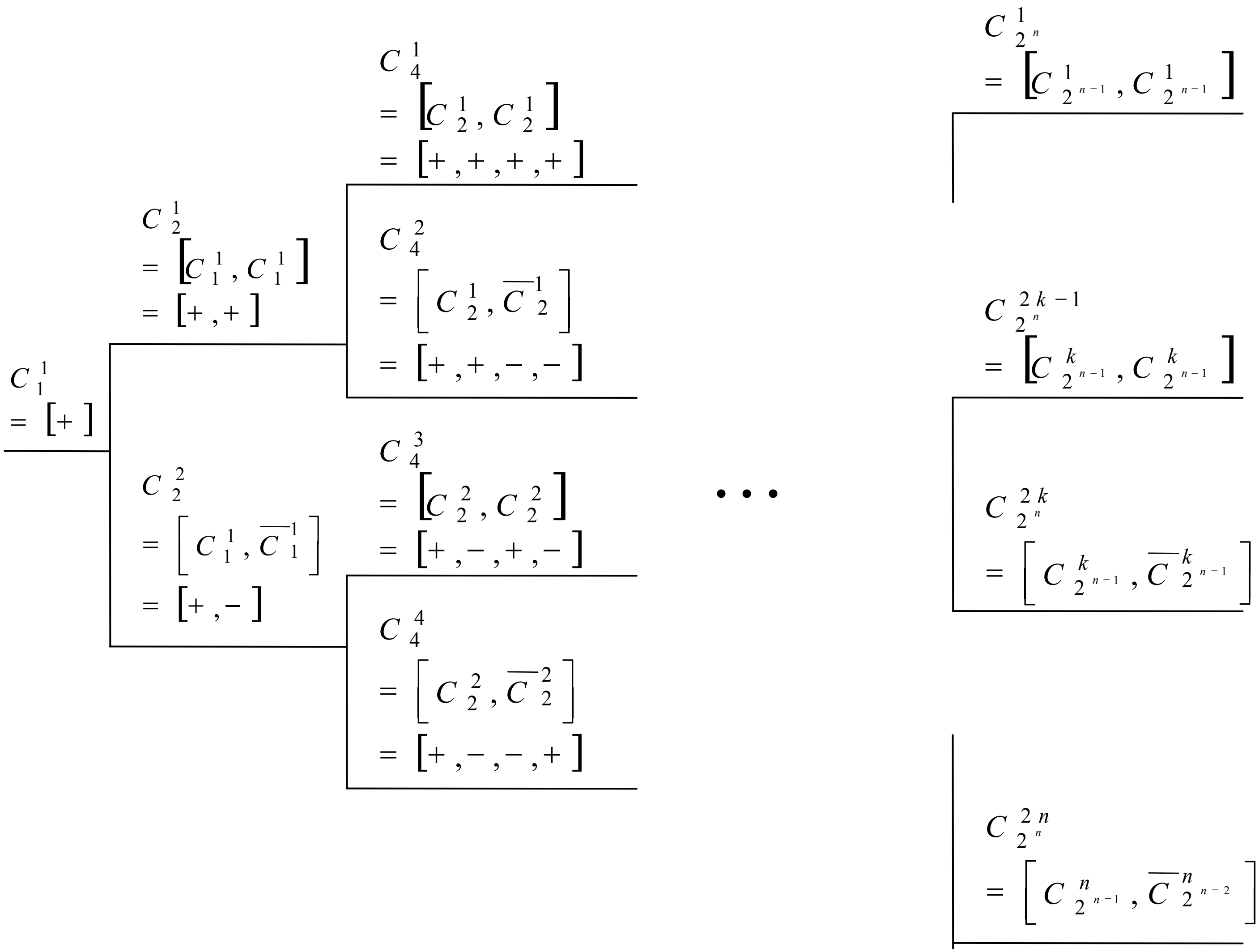
Presentation Outline

- Background
- System Model
- Call Admission Control Schemes
- Complexity Of Proposed Schemes
- Numerical Results
- Summary

Background

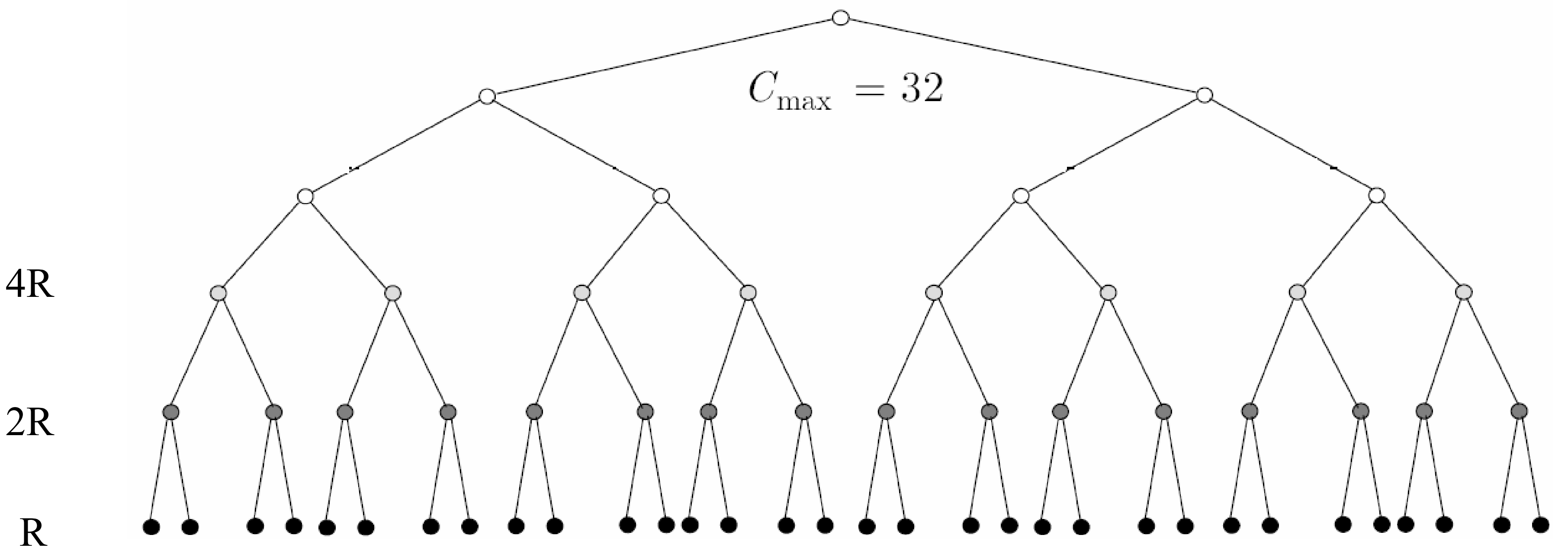
- In Wideband CDMA orthogonal variable spreading factor (OVSF) are codes for providing different channels for different users with different data rates
-

Well known OVSF code generation

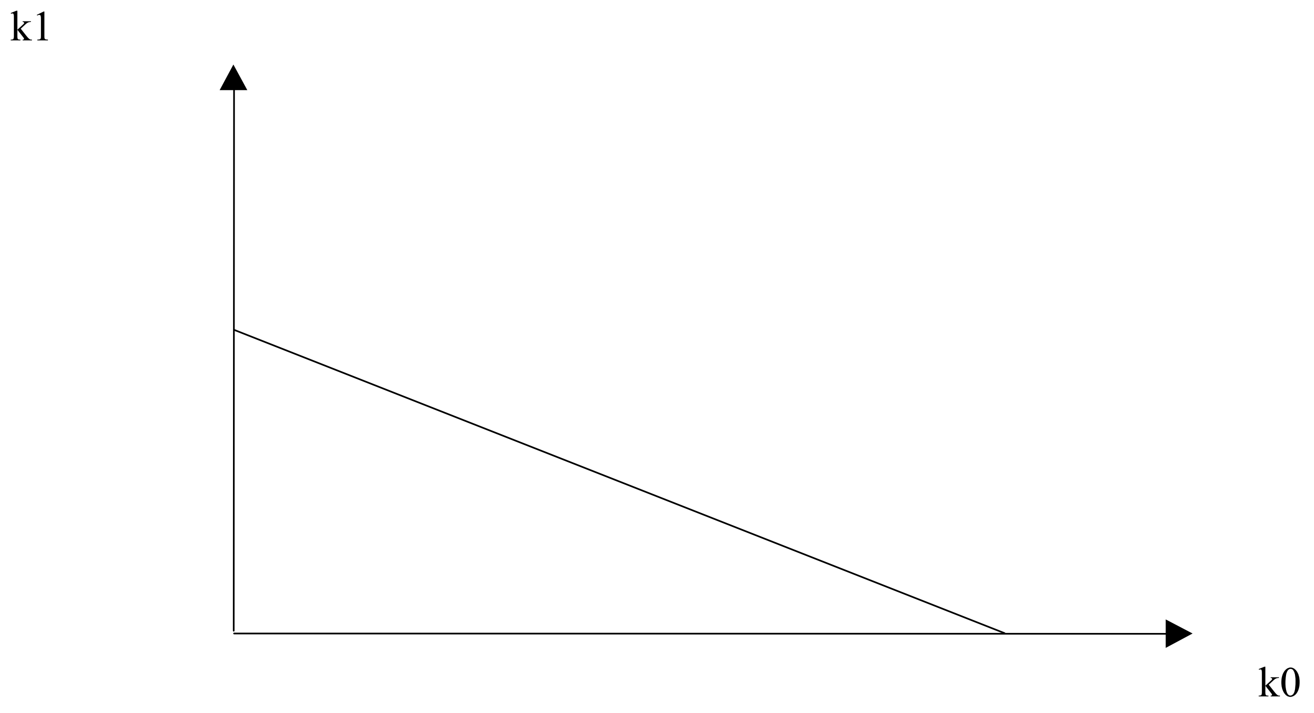


- Nodes with the depth = l from the root of the tree have code words of code length = 2^l.

Example



Resource constraint: $\sum_{j=0}^{M-1} 2^j k_j \le C$



Example of capacity region,, Cmax=32

System Model

- Random arrivals of call requests in each class can be modeled by a stochastic process – (Poisson process).
- For class *j*
  - Average arrival rate is λ_j
  - Average call duration μ_j.
- In response to each arrival of call request, the network operator must make a call admission control (CAC) decision (whether to accept or reject the call request), and the decision should be made based on the resource constraint.
- The objective can be to maximize average throughput, or minimize blocking probability, etc.

System Model

- We allow code re-assignment

Main point

- The greedy Call Admission Policy is not necessarily optimal.

Optimal policy

- The system can be modeled by a Markov decision process, so an optimal policy can be obtained by optimizing a Markov Decision Process.
  - Linear Programming
  - Policy iteration
  - etc.

$$\left\{ (k_0, k_1, \cdots, k_{M-1}) \middle| \sum_{i=0}^{M-1} 2^i k_i \le C \ , k_i \in Z^+ \right\}$$

Previous Results: optimal policy

- An optimal policy can have significantly better performance than the greedy policy
  - According to small-scale cases
- Large-scale: large *C*, large number of classes, etc.
  - Computational complexity of linear programming (e.g., number of variables)

$$\left\{ \left(k_0, k_1, \cdots, k_{M-1}\right) \middle| \sum_{i=0}^{M-1} 2^i k_i \le C \;, k_i \in Z^+ \right\}$$

Previous results: suboptimal policy

- Even a fixed set policy can have significantly better performance than the greedy policy.

Optimal Fixed Set Partitioning

- OVSF codes are partitioned into mutually exclusive groups of codes and uniquely assigns a group to each service class.
- G_j the number of codes in the subset assigned with class j, where $j = 0, 1, 2, \ldots, M-1$.
- Once vector $G \equiv (G_0, G_1, \ldots, G_{M-1})$ is determined, the actual partition of the set is trivial.
- The numbers of codes in the subsets $G_0, G_1, \ldots, G_{M-1}$ satisfy

$$\sum_{j=0}^{M-1} 2^j G_j = C$$

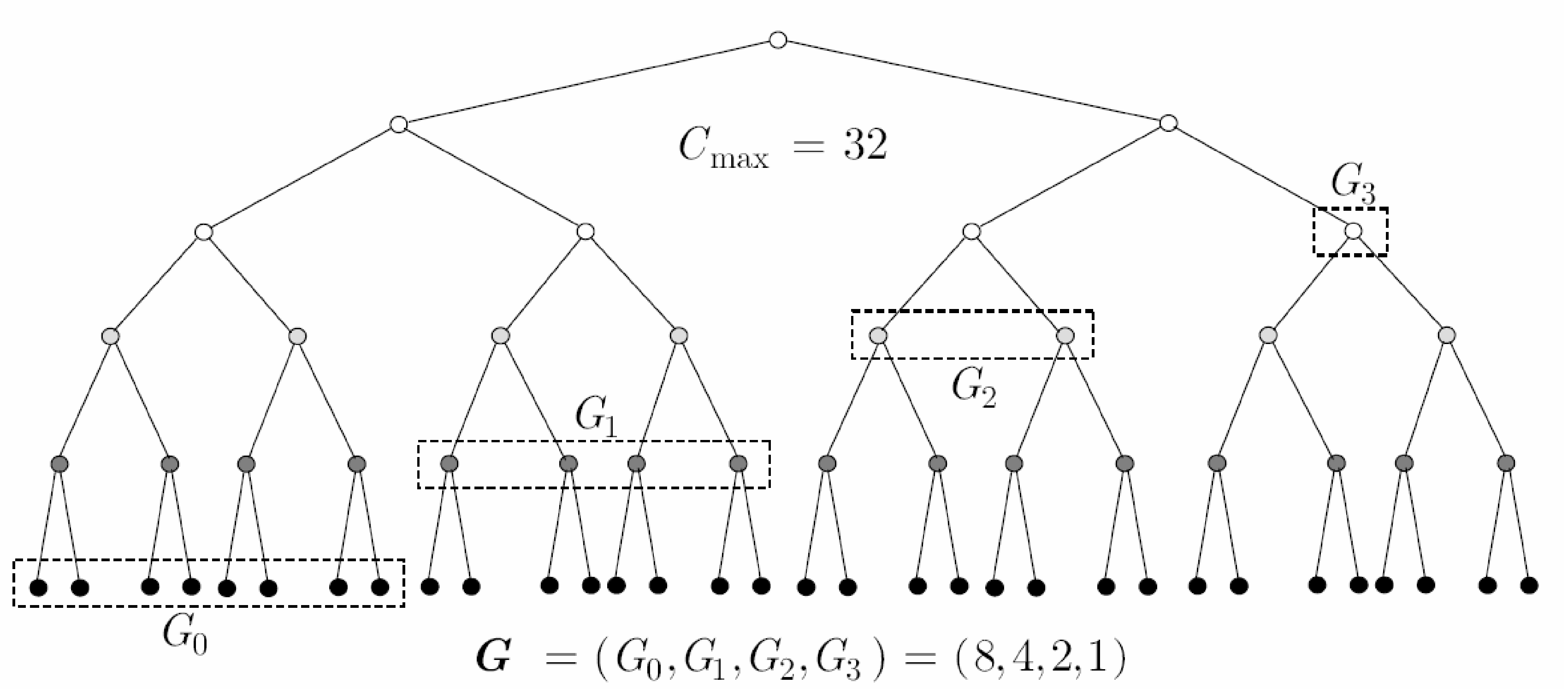
- Average Through put of this scheme

$$T_F(G) = \sum_{k=0}^{M-1} \left(2^k R\right)\left(1 - P_k\right) \lambda_k / \mu_k$$

where

$$P_k = \frac{\left(\lambda_k / \mu_k\right)^n / n!}{\sum_{n=0}^{G_k} \left(\lambda_k / \mu_k\right)^n / n!}$$

Optimal Fixed Set Partitioning



Greedy within fixed sets

Complexity Analysis

Size of the feasible set: the number of vectors $\left(G_0, G_1, \cdots, G_{M-1}\right)$ that satisfy constraint $\sum_{j=0}^{M-1} 2^j G_j = C$

Combinatorial Analysis and Recursion

- G_{M-1} can have any value from 0, 1,…, $C/2^{M-1}$. if G_{M-1} is x_M. Then, G_{M-2} can be 0, 1, 2, 3, …, $2(C/2^{M-1} - x_M)$ and so on.
- The relation between adjacent classes has a recursion

Let $\Gamma_m(v)$ be the number of non-negative integral vectors $(G_0, G_1, \cdots, G_{M-1})$ that can occupy the space taken by v codes of largest codes i.e., satisfying constraint $\sum_{i=0}^{m} 2^i G_i = v2^m$.

Then,

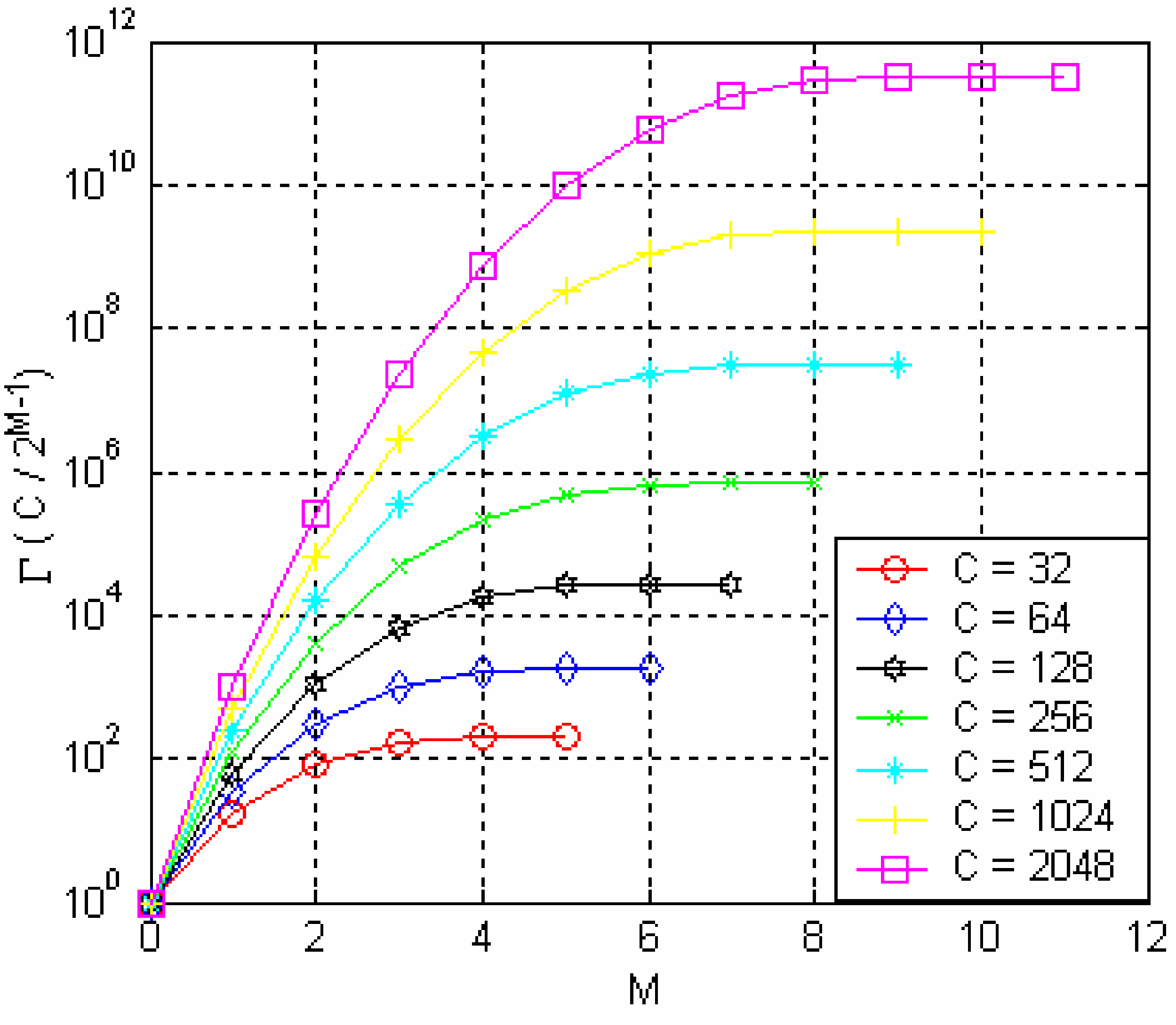
$$\Gamma_m(v) = \sum_{x=0}^{v} \Gamma_{m-1}(2(v-x)) = \sum_{i=0}^{v} \Gamma_{m-1}(2i)$$

$$\Gamma_0(v) = 1, \quad v = 0, 1, 2, 3, \ldots.$$

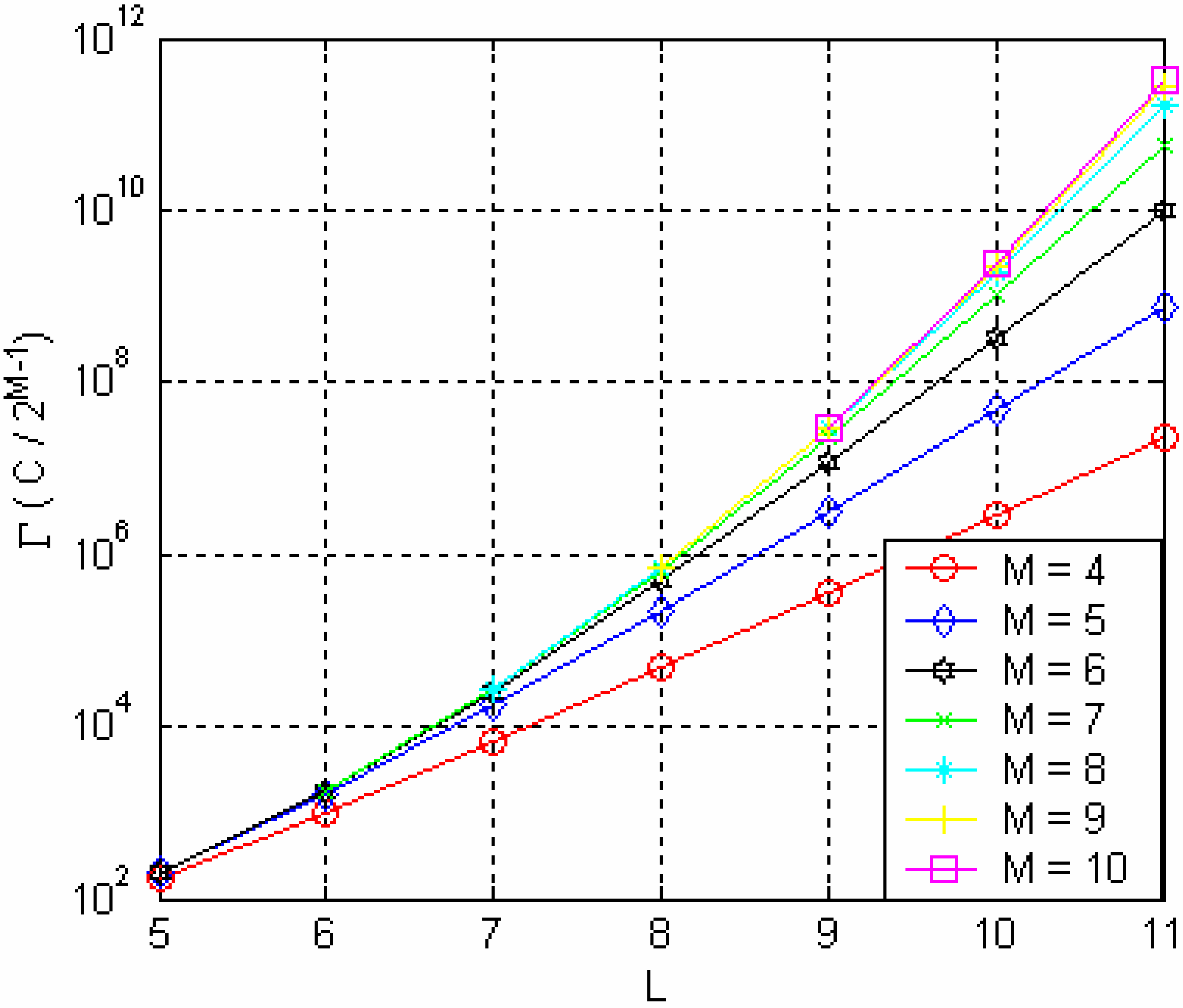
Functions $\Gamma_0(v), \Gamma_1(v), \ldots, \Gamma_{M-1}(v)$ can be evaluated from the recursion.

The number of vectors $(G_0, G_1, \cdots, G_{M-1})$ that satisfy constraint $\sum_{j=0}^{M-1} 2^j G_j = C$ is $\Gamma_{M-1}\left(C/2^{M-1}\right)$.

Numerical Results



Numerical Results



Optimization of Markove Decision process through LP

In the case of LP, its complexity is closely related to the size of state space,

$$\left|\left\{\left(k_0, k_1, \cdots, k_{M-1}\right) \middle| \sum_{i=0}^{M-1} 2^i k_i \le C \ , k_i \in Z^+ \right\}\right|$$

- Let $\chi_m(v)$ be the number of non-negative integer vectors $(k_0, k_1, \ldots, k_m)$ that satisfy the constraint $\sum_{i=0}^{m} 2^i k_i \le v 2^m$
- The number of vectors $(k_0, k_1, \ldots, k_{m-1})$ that satisfy this constraint is $\chi_{m-1}(2(v-x_m))$. Therefore, we have the following recursion:

$$\sum_{i=0}^{m-1} 2^i k_i \le (v - x_m) 2^m = 2(v - x_m) 2^{m-1}$$

$$\chi_m(v) = \sum_{i=0}^{v} \chi_{m-1}(2i)$$

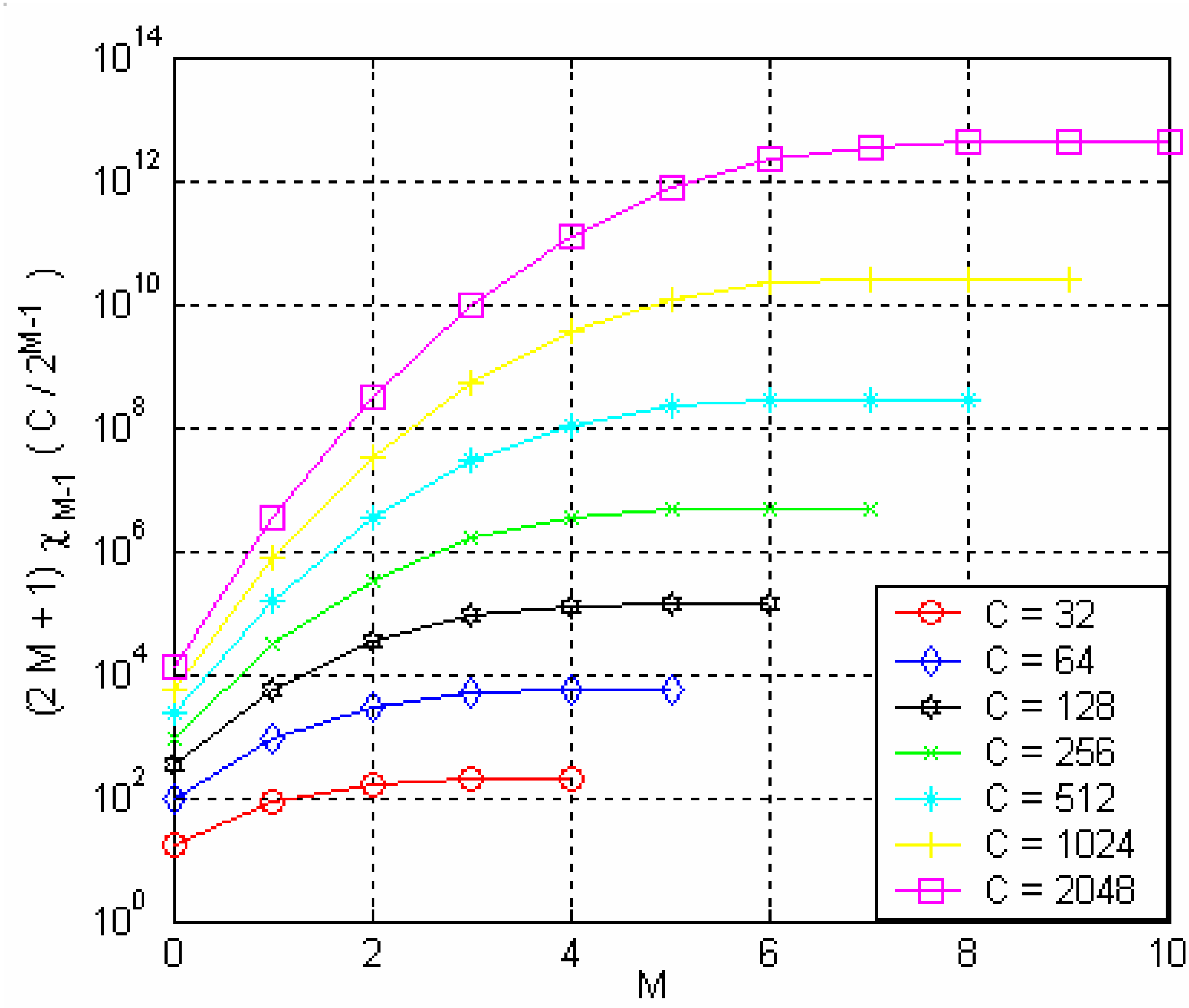
$\chi_m(v)$ and $\Gamma_m(v)$ have an identical recursion

The size of the state space

$$\left|\left\{\left(k_0, k_1, \cdots, k_{M-1}\right) \middle| \sum_{i=0}^{M-1} 2^i k_i \le C \ , k_i \in Z^+ \right\}\right|$$

$$= \chi_{M-1}\left(C/2^{M-1}\right) = \Gamma_M\left(C/2^{M-1}\right)$$

Numerical Results



Complexity

- Fixed Set Partitioning (FSP)

The complexity of FSP operation is characterized as $O(1)$ lookups

- DCA-CAC

The complexity of DCA-CAC is also $O(1)$

In summary, in both FSP and DCA, the computational complexity of online operations is $O(1)$ per call arrival.

Thank You.