Collusion in online competitions using covert channels

Inference Group Meeting
2 July 2003

Steven J. Murdoch

http://www.cl.cam.ac.uk/users/sjm217/

Security Group, Computer Laboratory, University of Cambridge

Summary

- Connect-4 competition
- Advantages of collusion
- Multi Level Security
- Covert channels for authentication
- Choosing a collusion strategy
- Results

Introduction

- Collusion in games can give a significant advantage
- In Bridge partners should not be able to know the other’s cards
    - Many rules on what is permitted
    - Hard to enforce, often policed on the instincts of experts
- In sports sometimes teams have colluded (normally against the rules)

Running example

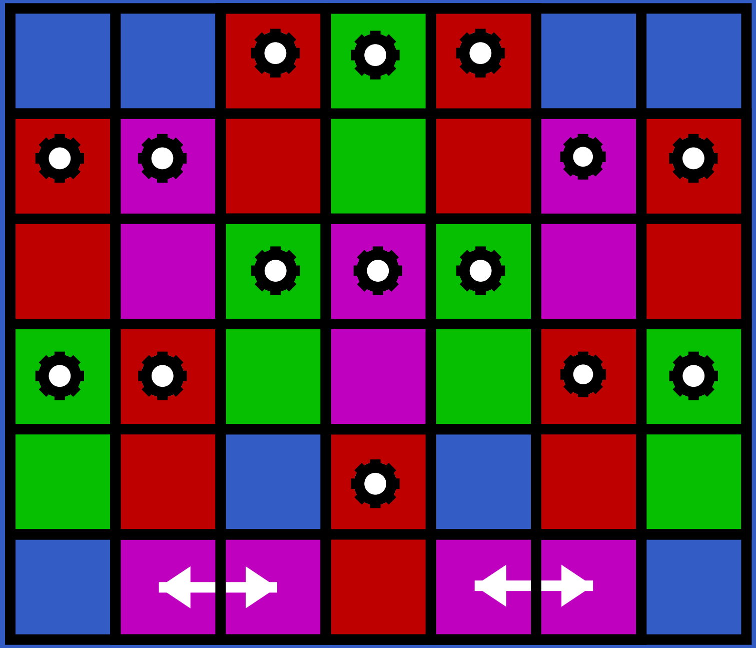
- Rules of game like Connect-4
- However players can pass
- First stage of competition is a league
    - 2 points for a win, 1 point for a draw
- Top five go into a knockout stage
    - If there is a draw, CPU time and memory usage decide

Problem

- Because of passing rule no winning strategy exists
- However there is a fast drawing strategy (Piotr Zieliński)
- Solution: Enter multiple “drones”, draw against everyone except one player, who will get more points

Game strategy

- Based on one of the techniques used by L. Victor Allis for the original Connect-4 game.



Authentication

- For collusion an authentication mechanism is needed
- Normally easy, but harder in programming competitions
- Low bandwidth — 1 bit for simple case
- Must be high reliability
- Policy: Overt channels, such as sockets/IPC not permitted under competition rules

Multi Level Security (MLS)

- Bell-LaPadula policy used in military time sharing systems
- Risk of trojans being inserted at high privilege level
  - All data has a level, all programs have a level, all users have a clearance
  - No read up (program cannot read data with a higher level than itself)
  - No write down (program cannot write data with a lower level than itself)

Covert channels

- Channel which violates (write-down) security policy
- Timing Channels
  - CPU Load
  - Disk head scheduling
- Storage Channels
  - Process table exhaustion
  - Disk space

Covert channels in Connect-4

- Standard covert channels can be used
  - Complicated, noisy
  - Uses CPU time
- Move timing
  - Easier, less CPU time (sleep not counted)
  - Could be noisy

Moves as a covert channel 1

- For most strategies there is more than one equivalent move
- Till late in the game the outcome is undecided
- Use redundancy in move to send message
- Noise difficult to model
- So use (fast) PRNG to select move
- Linear congruential

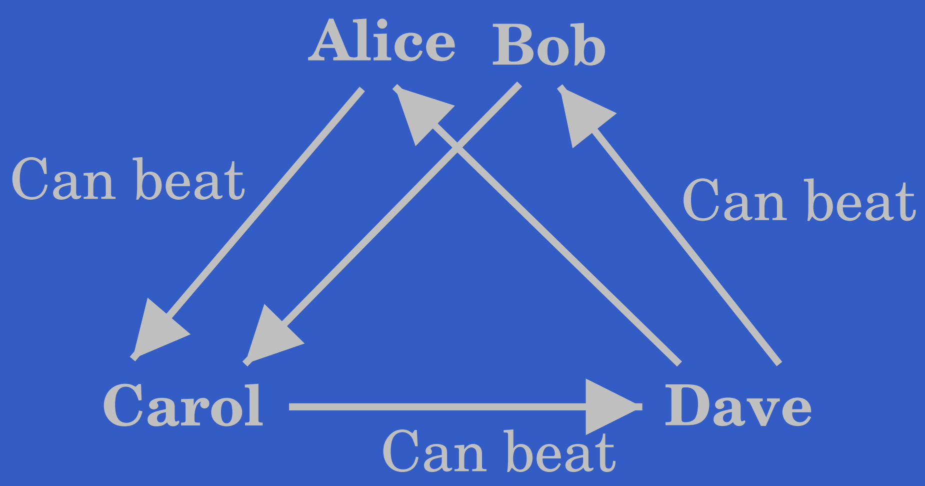
  $$X_{n+1} = (A * X_n + C) \bmod Z$$

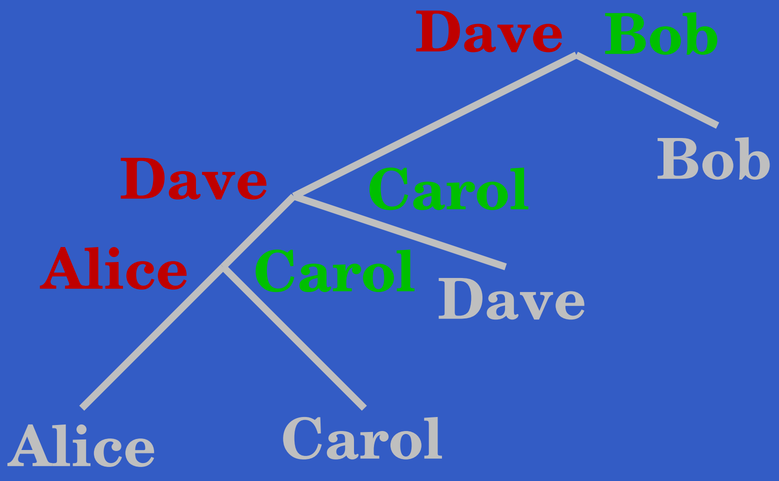
Moves as a covert channel 2

- Receiver knows:
  - Move number
  - Alternative moves
  - Opponents choice if it was a “friend (by using same PRNG)
  - If previous moves matched expectation
- Winning player needs no detection system, both faster and less suspicious

Strategy 1

- For league stage, collusion works
- For knockout tournaments, it doesn't
  - Exception where order is known and cycle in graph of results

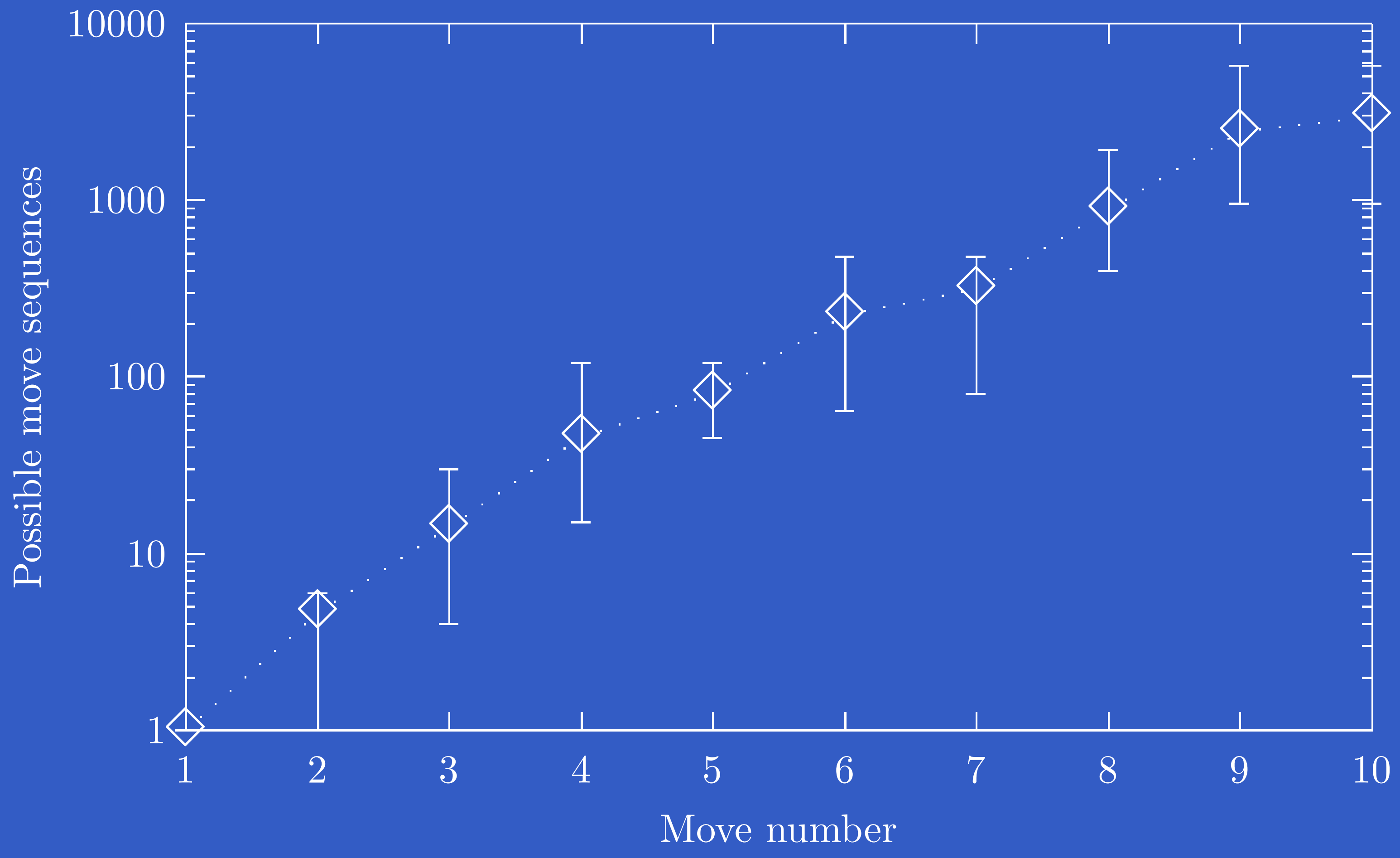




Strategy 2

- So enter 5 players to win rather than 1
- Also have multiple classes, to give advantage to optimised programs
  - However the more classes the higher the risk of false positives
- Risk of very poor programs
  - Programs that try to win will win against the bad players more than our players (which almost always draw)

Results 1


10000
1000
100
10
1
Possible move sequences
1
2
3
4
5
6
7
8
9
10
Move number

Results 2

Name	Played	Won	Drew	Lost	Points	Class
Maleficent	84	58	26	0	142	0
NCC1703	84	58	26	0	142	0
Flounder	84	51	29	4	131	1
NCC1707	84	49	31	4	129	1
Ratigan	84	49	31	4	129	1
NCC1705	84	48	32	4	128	1
7 others	⋮	⋮	⋮	⋮	88–99	—
24 drones	⋮	⋮	⋮	⋮	72–75	2
6 “rabbits”	⋮	⋮	⋮	⋮	54–71	—

Results 3