My Projects Using Game Theory

Robert Holte

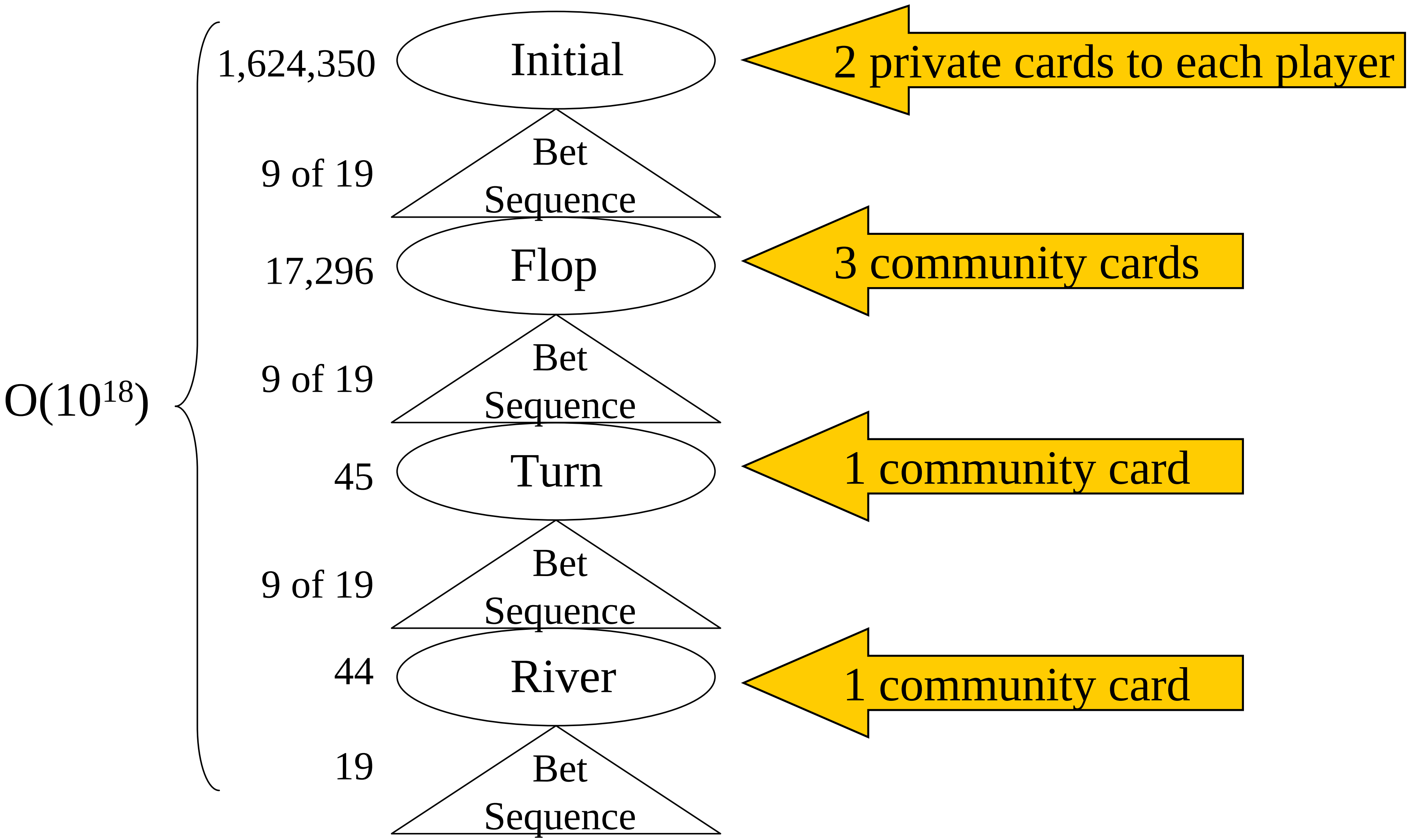
Poker

The Challenges

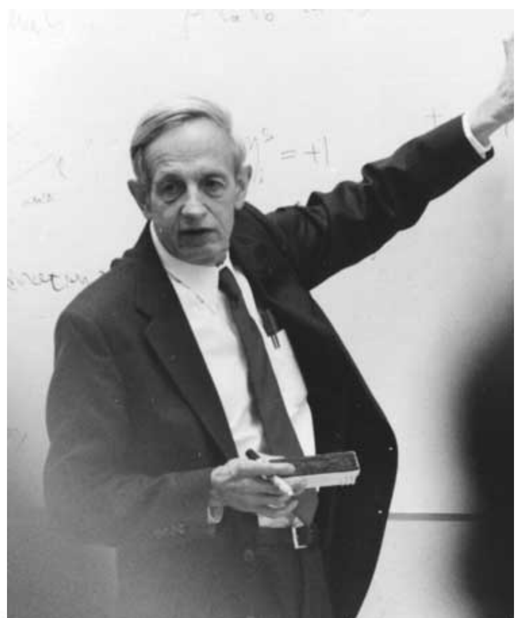
- Large game tree
- Stochastic element
- Imperfect information (during hand, and after)
- Variable number of players (2–10)
- Aim is not just to win, but to maximize winnings
  - Need to exploit opponent weaknesses

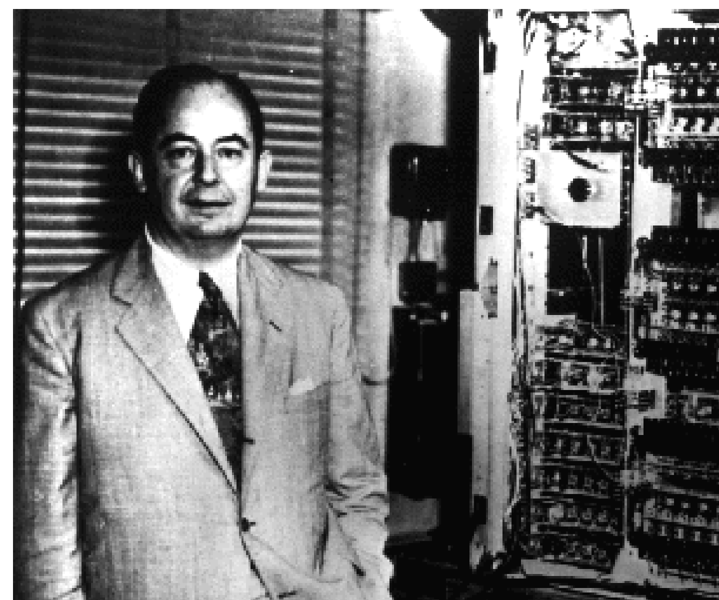
ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

2-player, limit, Texas Hold'em



Game-Theoretic Approach





ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

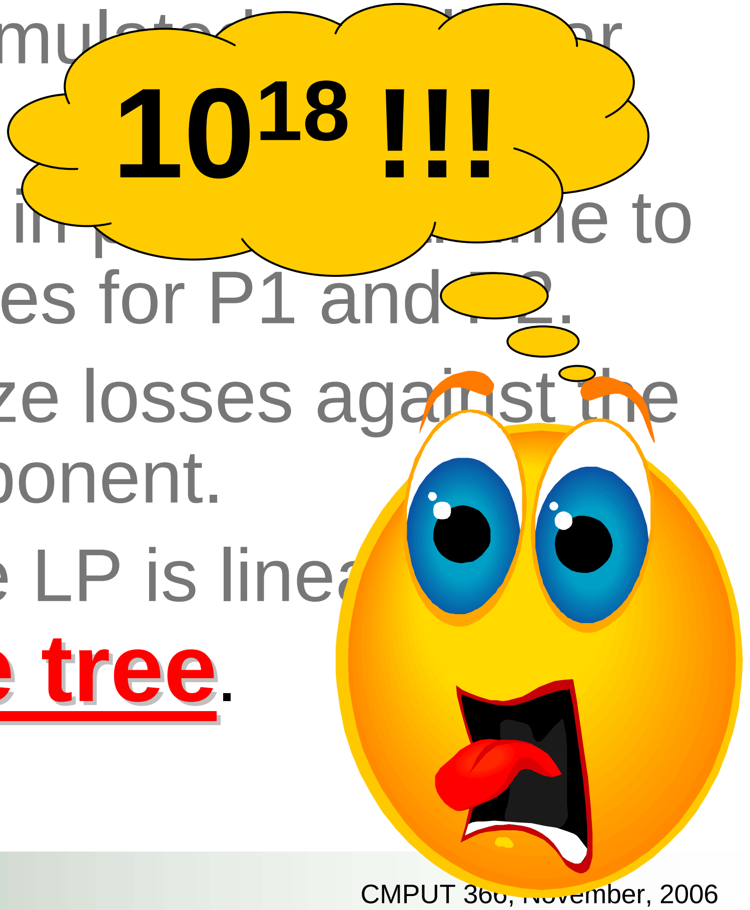
Linear Programming

- ...layer, zero-sum game with chance ... mixed strategies, and imperfect ...tion can be formulated as a linear ...n (LP).
- ...P can be solved in polynomial time to ...duce Nash strategies for P1 and P2.
- Guaranteed to minimize losses against the strongest possible opponent.
- “Sequence form” – the LP is linear in the size of the game tree

(Koller, Megiddo, and von Stengel)

Linear Programming

- A 2-player, zero-sum game with chance events, mixed strategies, and imperfect information can be formul[illegible]ar program (LP).
- The LP can be solved in [illegible]me to produce Nash strategies for P1 and [illegible].
- Guaranteed to minimize losses aga[illegible]st the strongest possible opponent.
- “Sequence form” – the LP is linea[illegible] **size of the game tree**.

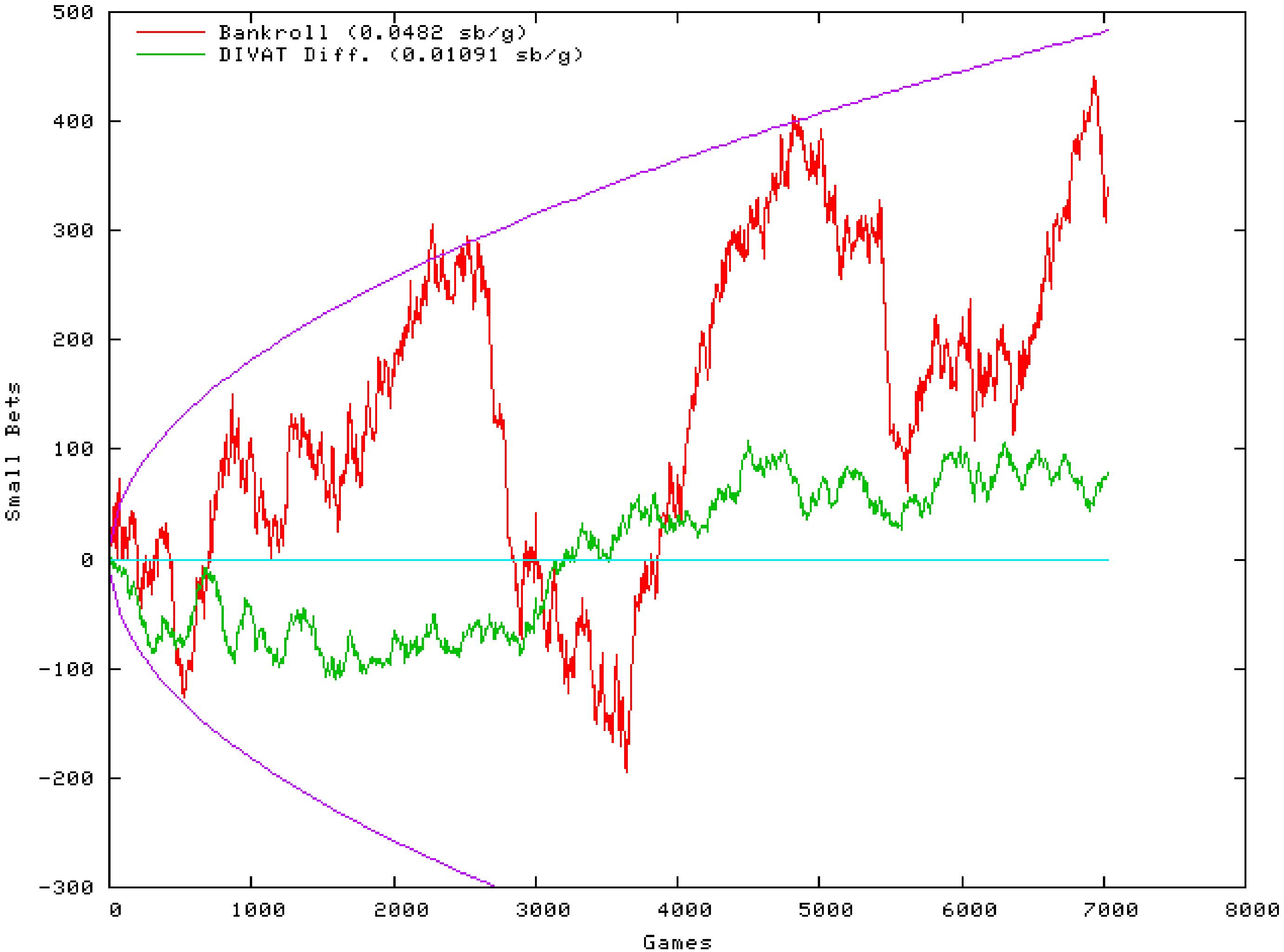


PsOpti (Sparbot) – IJCAI'03

- Abstract game tree of size 10^7
- Bluffing, slow play, etc. fall out from the mathematics.
- Best 2-player program to date
- Has held its own against 2 world-class humans
- Won the AAAI'06 poker-bot competitions

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

PsOpti2 vs. “theCount”



PsOpti's Weaknesses

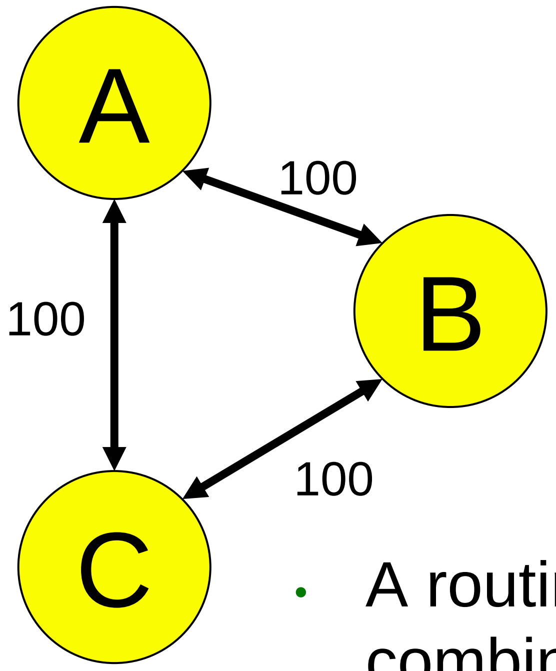
- The equilibrium strategy for the highly abstract game is far from perfect.
- No opponent modelling.
  - Nash equilibrium not the best strategy:
    - Non-adaptive
    - Defensive
  - Even the best humans have weaknesses that should be exploited

http://www.poker-academy.com

Network Routing when Traffic Demands are Uncertain

Recent Ph.D. thesis by Yuxi Li

Example



TRAFFIC DEMAND

To / From	A	B	C
A	0	80	20
B	80	0	10
C	20	10	0

- A routing is a set of fractions for all meaningful combinations of H, O, D, N in:

  At node H, $f_{H,O,D,N}$ of the packets going from node O to node D should be sent next to node N

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

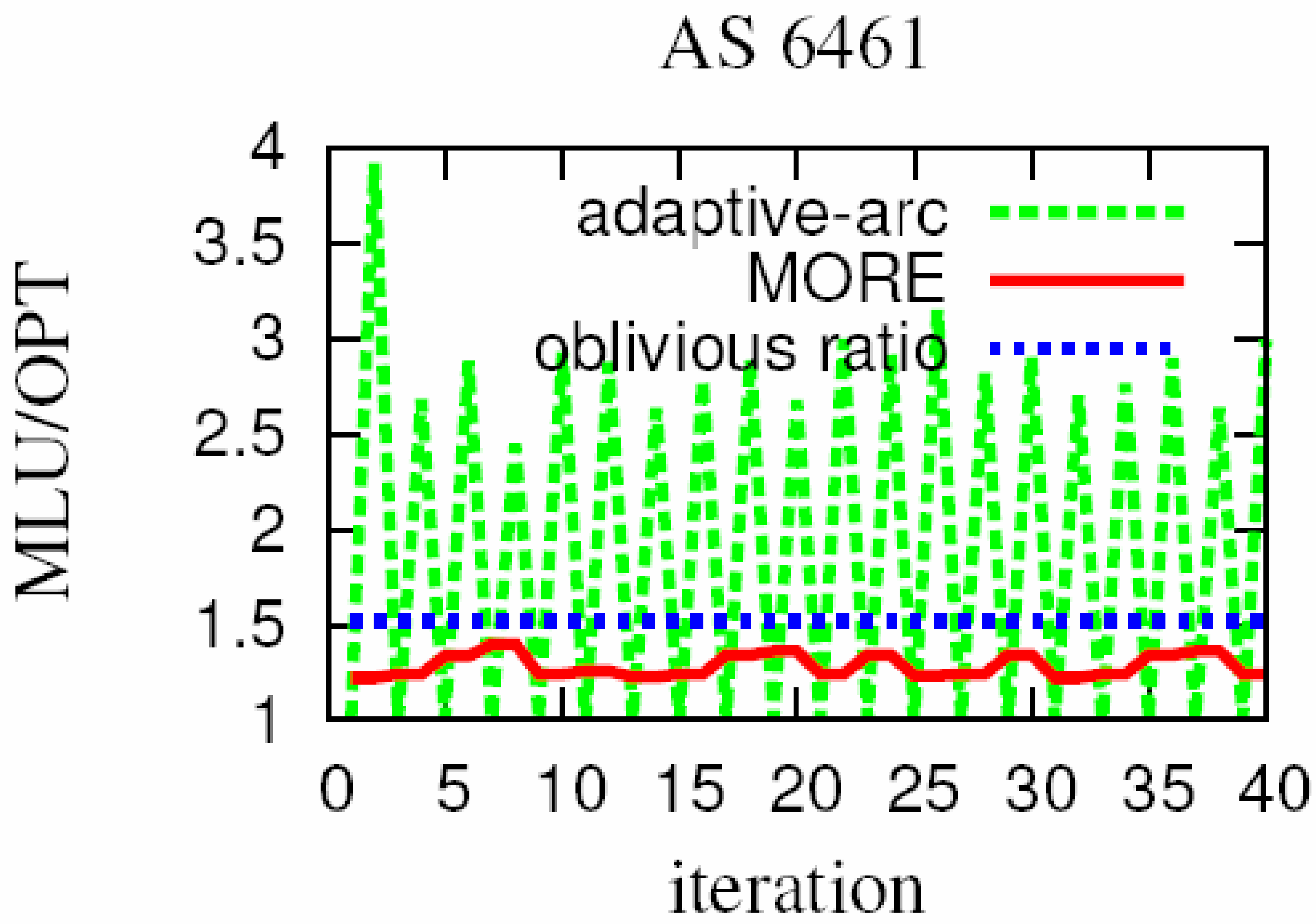
Treat this as a 2-player game

- Objective function:
  - lifetime of an energy-constrained network (e.g. sensor network)
- Player 1 (max): choose a routing
- Player 2 (min): choose a traffic demand matrix

- The Nash equilibrium of this game is a routing that is resistant to adversarial attacks

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Behaviour under Attack



Combinatorial Auctions

Rodent auction

Flopsy

Rodent auction

Flopsy

Mopsy →

Rodent auction

Flopsy

Mopsy →

Jack

Rodent auction

Flopsy

Mopsy →

Jack

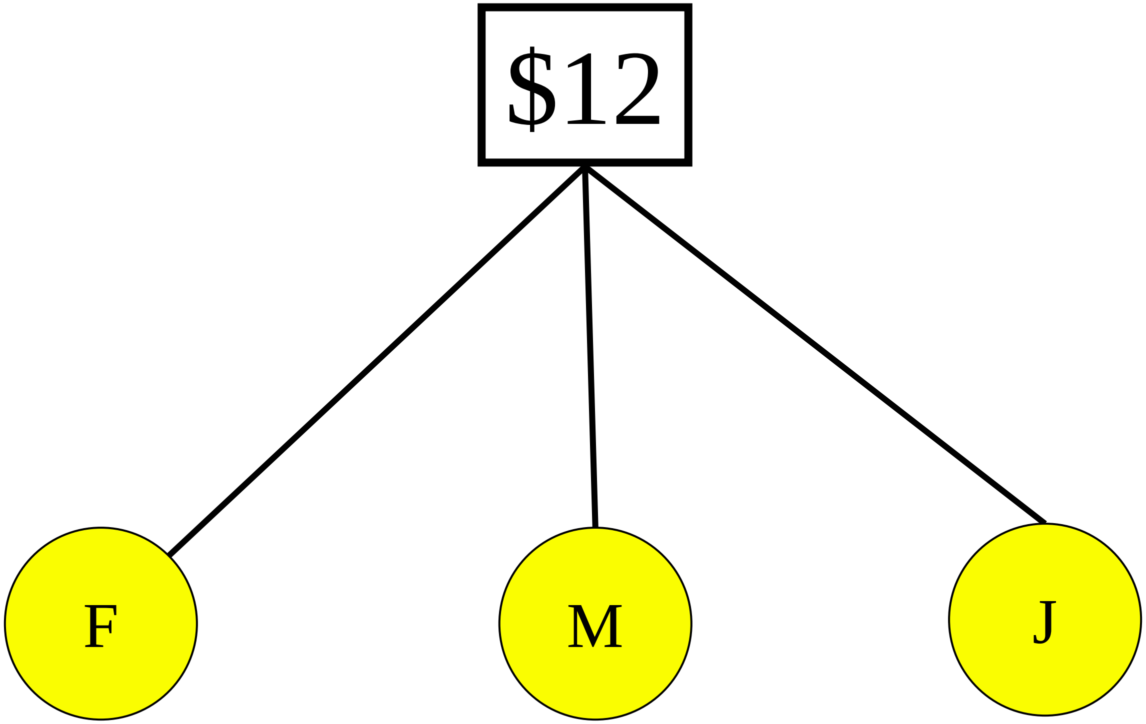
C1: will pay $5 for any one
C3: will pay $12 for all three
C2: will pay $9 for a breeding pair
(Flopsy and one of the others)

Combinatorial Auctions

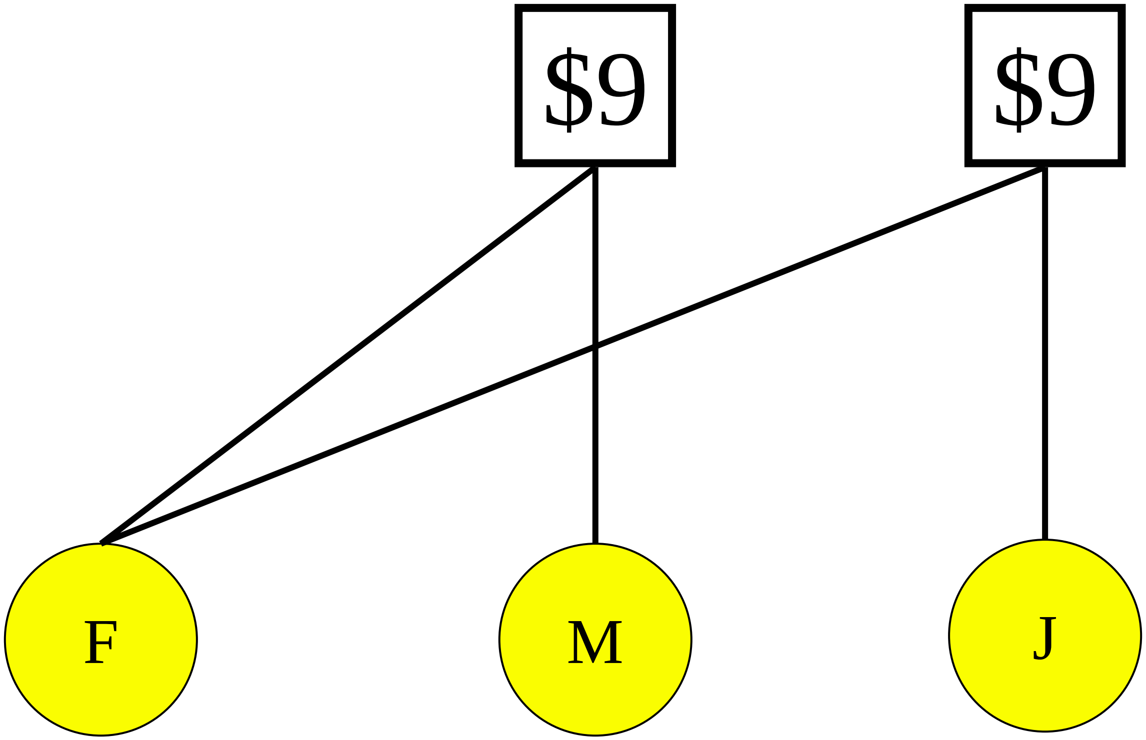
- Single Unit C.A. – one copy of each item
- Auction all items simultaneously
- Bid specifies a price and a set of items ("all or nothing")
- Exclusive-OR can be achieved by having a "dummy item" representing the bidder
- Multi-round or single-round

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

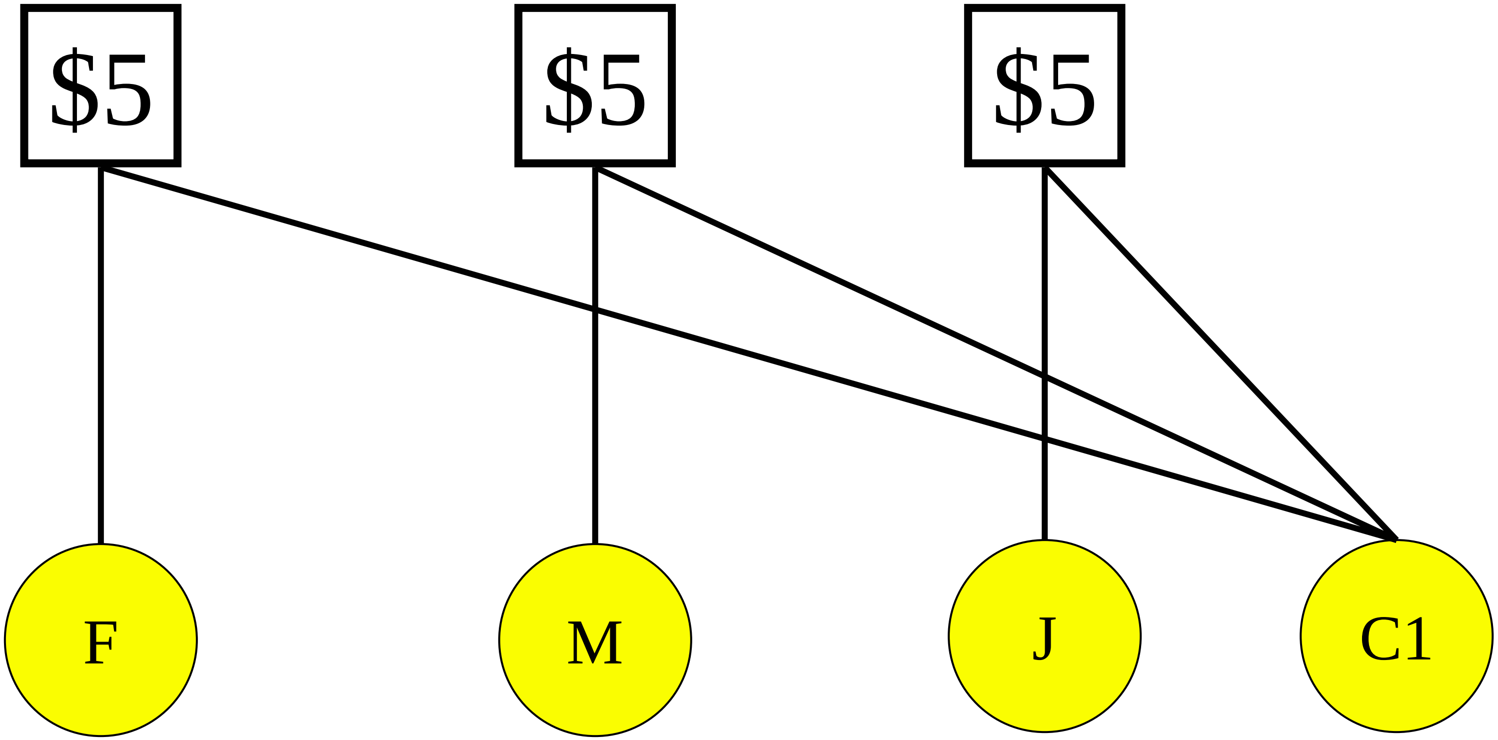
$12 for all three



$9 for a breeding pair



$5 for any one



ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Applications

- FCC spectrum auctions
- Goods distribution routes
- Airport gates, parcels of land
- eBay

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Example Application

SPOT: earth observation satellite

Requests are made for photographs.

Each photograph can be taken at several times by several different instruments/settings, but quality (profit) may vary.

Using one instrument/setting at a given time may prevent the use of another at the same or adjacent times.

Example

Photograph 1

- $4 if taken on instrument A at time T1
- $3 if taken in instrument A at time T2
- $12 is taken on instrument B at time T2

Photograph 2

- $7 if taken on instrument B at time T2

If B is being used, A cannot used at the same time.

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

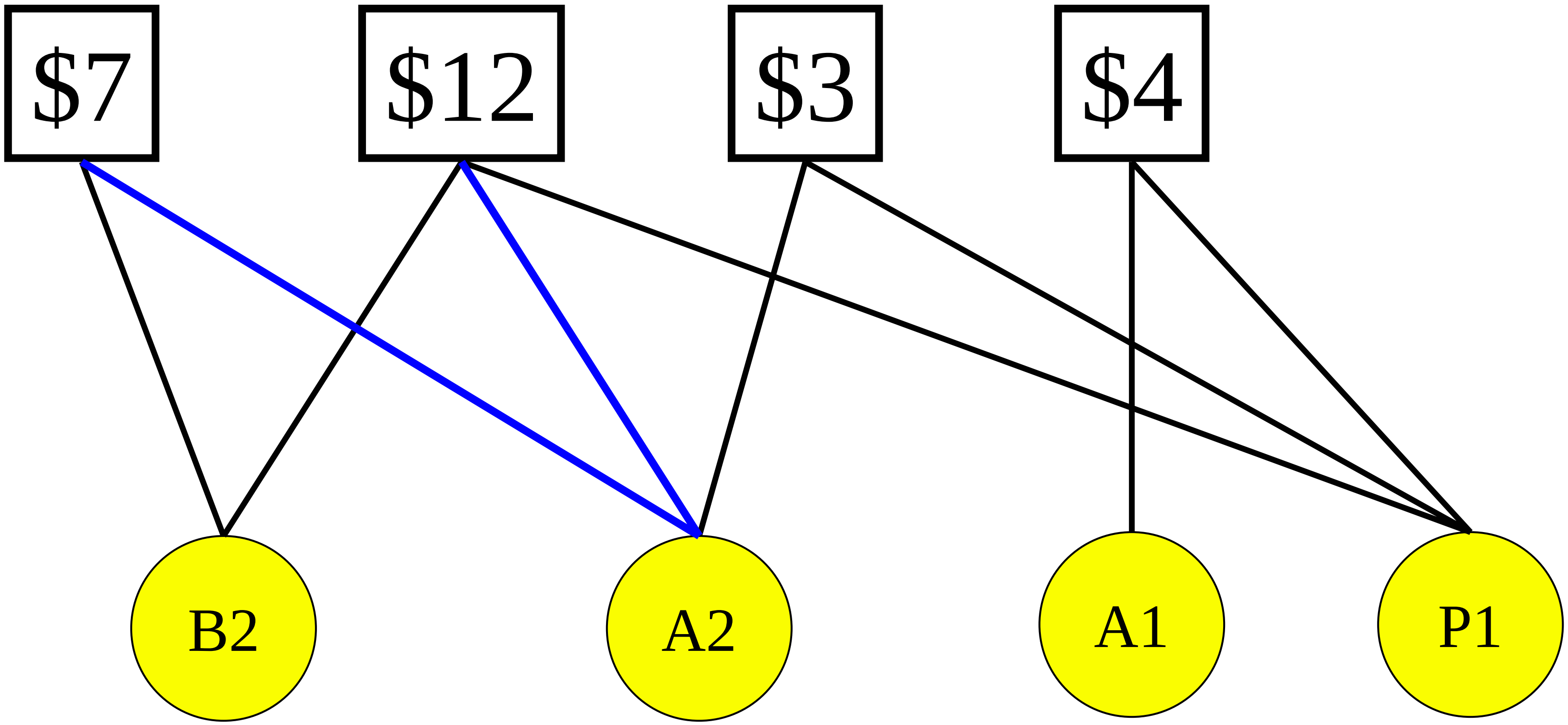
Formulation as a C.A.

Each instrument/setting/time is a separate item for auction.

Each photograph is a bidder, with XOR bids for each of the different ways of achieving the photograph.

If “items” A and B are mutually exclusive, any bid for A also includes B.

Example as a C.A.



Winner Determination

Problem: how to determine who wins ?

Choose a set of bids that are feasible (disjoint) and maximize the auctioneer's profit.

NP-complete (set packing problem)

Multi-unit Combinatorial Auctions

- There are b_i copies of item i
- A bid specifies a quantity for each item
(is a vector length *m* if there are *m* items)
- A set of bids is feasible if its total demand for each item does not exceed the number of available copies of the item

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Manufacturing Application 1

- Inventory consists of *m* types of parts, with b_i instances of type *i.*
- A customer order requests a certain quantity of each part, and offers a price.

Determine which orders to fill to maximize profit.

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Auctions and Knapsacks

Winner determination in a multi-unit combinatorial auction **with a single item** is the classic NP-complete Knapsack problem.

With more than one item, it is the **multidimensional Knapsack** problem (MDKP) – much less studied.

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

The Knapsack Problem

- An “easy” NP-complete problem
- Garey & Johnson: knapsack is considered “solved” by many (by branch-and-bound)
- n (#bids) can be reduced by decomposition and preprocessing
- Solvable in pseudo-polynomial time by dynamic programming

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Knapsack Algorithms

- Dave Pisinger's PhD (1995) - publicly available, very fast code (d.p.)
- There is a very simple greedy algorithm guaranteed to be ½ optimal or better

The Multidimensional Knapsack Problem (MDKP)

- An hard NP-complete problem
- Not solvable in pseudo-polynomial time.
- No greedy (polynomial) algorithm can guarantee an approximation that is better than $OPT/k^{½}$ where k is the sum of the b_i.

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Why Try Hill-climbing ?

- It is part of branch&bound search and some heuristic search algorithms
- Try simple, generic search algorithms before complex ones and problem-specific variants

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Deterministic Hill-Climbing

Start with the empty bid-set

REPEAT:

- Consider adding one bid to the current bid-set
- Prune bid-sets that are infeasible

- Add the bid with the highest "score"

UNTIL pruning eliminates all alternatives

Report the highest price seen during search, not the price of the final local "score" maximum

Deterministic Hill-Climbing

Start with the empty bid-set

REPEAT:

- Consider adding one bid to the current bid-set
- Prune bid-sets that are infeasible **or cannot be extended to improve the best price seen so far**
- Add the bid with the highest "score"

UNTIL pruning eliminates all alternatives

Report the highest price seen during search, not the price of the final local "score" maximum

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Scoring functions

- Price
- N2norm = Price/size
  - Single-unit, size = $(\text{\# items in the bid})^{1/2}$
  - Multi-unit, size = $(\sum f(i)^2)^{1/2}$

    f(i) is the fraction of the remaining quantity of item i required by the bid
- KO = Price/(price of contending bids KO'd by the bid)

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Randomized Hill-climbing

Instead of adding the bid with the best score, choose among the alternative bids (after pruning) randomly, with probability proportional to score.

Restart (with the empty bid-set) several times on a given problem and report best price seen on any restart.

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Single-Unit Test Problems

CATS test suite from Stanford

- new
- Problem generators for 5 different scenarios

  e.g. airport takeoff and landing time-slots
- Realistic (e.g. airports are Chicago, LaGuardia, etc.)
- Numerous Parameters – defaults used except
  - 3 variations on “regions” (default + 2 others)
  - scaled down “paths” and “scheduling”

Experimental Setup

- For each problem type randomly generate 100 different problems

- On each problem run
  - the 3 deterministic hill-climbers

    "Best DHC" = best of these prices
  - the 3 randomized hill-climbers (20 restarts each)

    "Best RHC" = best of these prices

Average Solution Quality (% of optimal)

problem type	best DHC	best RHC
path	98	98
match	99	99
sched	96	98
r75P	83	92
r90P	90	96
r90N	89	96
arb	87	95

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Observations

- Randomized (20 restarts) better than deterministic
- Randomized finds very good solutions
  (always > 80%, average > 92%)
- Problem ratings:

easy: path, match, sched
harder: r90P, r90N, arb
hardest: r75P

- On the easy problems, deterministic finds very good solutions (almost as good as randomized)

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Which Scoring Function ?

N2norm and KO about the same, better than Price.

Which Scoring Function ?

N2norm and KO about the same, better than Price.

But are they better than chance ?

Blind Hill-climbing

Choose bid randomly with uniform probability

- still prunes
- still reports best price seen throughout search

Repeat 200 times on each problem to measure its solution quality distribution

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Percentage of Blind HC solutions worse than Deterministic solutions

problem type	N2norm	KO
path	100	100
match	100	100
sched	99	99
r75P	76	63
r90P	16	7
r90N	23	6
arb	40	20

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Single-unit CA – Conclusions

- hard for Blind HC, easy for DHC and RHC
  - problems of this type are solved well by HC
  - success is due to the scoring functions
  - not good testbeds for comparative experiments allowing suboptimal solutions
- easy for Blind HC, hard for DHC
  - scoring functions alone no better than chance
  - numerous good solutions throughout the search space
  - good testbeds as long as the Blind HC baseline is taken into account

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Multidimensional Knapsack Test Problems

ORLIB test suite from J. Beasley

- mknap1, mknap2
  - real-world, optimal values known
  - widely used
  - now considered “too easy”

- artificial, larger, harder problems
  - only the smallest is solvable by CPLEX
  - best known solutions very close to LP-optimal

Average Solution Quality (% of optimal or of best known)

test set	Price	N2norm	KO	Blind
mknap1	90	98.99	83	84
mknap2	94	99.00	79	58
mknapcb1	89	98.94	85	82
mknapcb2	89	99.03	85	83
mknapcb3	89	99.21	85	83
mknapcb7	93	98.35	85	81

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING

Conclusion – MDKP

- deterministic hill-climbing with N2norm competitive with all previous work, and much better than previous greedy approaches
- other scoring functions relatively poor
- randomized hill-climbing only better on the easy problems

ALBERTA INGENUITY CENTRE FOR MACHINE LEARNING