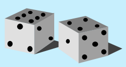
THE GAMES OF COMPUTER SCIENCE

© Games Workshop

TU DELFT
Feb 23 2001

© Games Workshop



Peter van Emde Boas
ILLC-FNWI-Univ. of Amsterdam

References and slides available at: http://turing.wins.uva.nl/~peter/teaching/thmod00.html

Topics

- Computation, Games and Computer Science
- Recognizing Languages by Games; Games as Acceptors
- Understanding the connection with PSPACE (The Holy Quadrinity)
- Interactive Protocols and Games
- Loose Ends in the Model ?

Games ???

Past (1980) position of Games in Mathematics & CS:

Study object for a marginally interesting part of AI (Chess playing programs)

Recreational Mathematics (cf. Conway, Guy & Berlekamp Theory)

Game Theory: von Neumann, Morgenstern, Aumann, Savage, ...

Games in Logic: Determinacy in foundation of set theory

Computer Science

- Computation Theory
- Complexity Theory
- Machine Models
- Algorithms
- Knowledge Theory
- Information Theory
- Semantics

WHERE ARE THE GAMES?

Games in Computer Science

- Evasive Graph properties (1972-74)
- Information & Uncertainty (Traub ea. - 1980+)
- Pebble Game (Register Allocation, Theory 1970+)
- Tiling Game (Reduction Theory - 1973+)
- Alternating Computation Model (1977-81)
- Interactive Proofs /Arthur Merlin Games (1983+)
- Zero Knowledge Protocols (1984+)
- Creating Cooperation on the Internet (1999+)
- E-commerce (1999+)
- Logic and Games (1950+)
- Language Games, Argumentation (500 BC)

Game Theory

- Theory of Strategic Interaction
- Attributes
  - Discrete vs. Continuous (state space)
  - Cooperative vs. Non-Cooperative (pay-off)
  - Perfect Information vs. Imperfect Information (Information sets) Knowledge Theory

PARTICIPANTS & MOVES

- Single player - no choices
- Single player - random moves
- Single player - choices : Solitaire
- Two players - choices
- Two players - choices and random moves
- Two players - concurrent moves
- More Players - Coalitions

COMPUTATION

- Deterministic
- Nondeterministic
- Probabilistic
- Alternating
- Interactive protocols
- Concurrency

COMPUTATION

- Notion of Configurations: Nodes
- Notion of Transitions: Edges
- Non-uniqueness of transition: Out-degree > 1 - Nondeterminism
- Initial Configuration : Root
- Terminal Configuration : Leaf
- Computation : ~~Branch~~ Tree
- Acceptance Condition: Property of trees

Linking Games and Computations

- Single player - no choices : Routine : Determinism
- Single player - choices : Solitaire : Nondeterminism
- Two players – choices : Finite Combinatorial Games : Alternating Computation
- Single player - random moves : Gambling : Probabilistic Algorithms
- Two players - choices and random moves : Interactive Proof Systems
- Several players & Coalitions - group moves : Multi Prover Systems

Introducing the Opponents

© Games Workshop

© Games Workshop

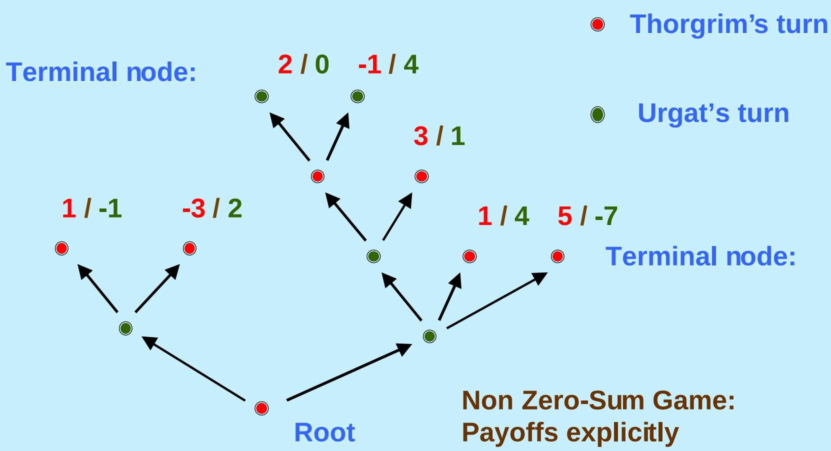
URGAT
Orc Big Boss

THORGRIM
Dwarf High King

Games involve strategic interaction ......

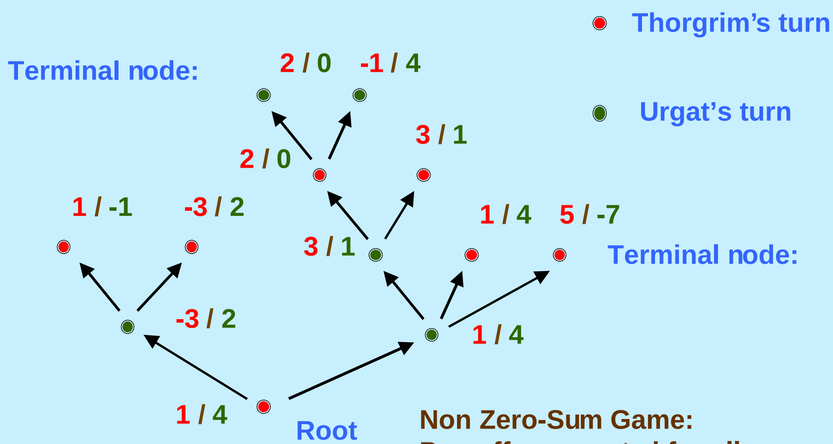
Game Trees

(*Extensive Form - close to Computation*)



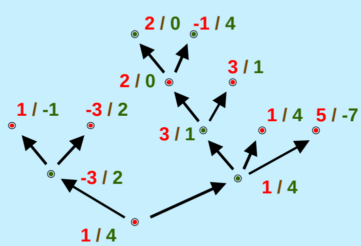
Non Zero-Sum Game:
Payoffs explicitly
designated at terminal node

Backward Induction



Non Zero-Sum Game:
Pay-offs computed for all game nodes including the Root.

Backward Induction



At terminal nodes: Pay-off as explicitly given

At Thorgrim's nodes: Pay-off inherited from Thorgrim's optimal choice

At Urgat's nodes: Pay-off inherited from Urgat's optimal choice

For strictly competetive games this is the Max-Min rule

Bi-Matrix Games

© Games Workshop

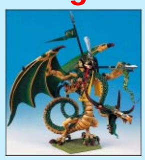
	O	S
R	-1/1	1/-1
D	1/-1	-1/1

© Games Workshop

Runesmith

© Games Workshop

Dragon



© Games Workshop

Ogre

© Games Workshop

Squigg

© Games Workshop

A Game specified by describing the Pay-off Matrix ....

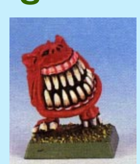
Peter van Emde Boas: The Games of Computer Science.

Von Neumann's Theorem

© Games Workshop

	O	S
R	-1/1	1/-1
D	1/-1	-1/1

© Games Workshop

R ( + D )/2 : (O  + S )/2

© Games Workshop © Games Workshop © Games Workshop © Games Workshop

Mixed Strategy Nash Equilibrium; no player can improve his pay-off by deviation.

Peter van Emde Boas: The Games of Computer Science.

A Game

Starting with 15 matches players alternatively take 1, 2 or 3 matches away until none remain. The player ending up with an odd number of matches wins the game

A Game specified by describing the rules of the game ....

Questions about this Game

- What if the number of matches is even?
- Can any of the two players force a win by clever playing?
- How does the winner depend on the number of matches
- Is this dependency periodic? If so WHY?

The Mechanism

Several of the results encountered in Computation Theory and in the Logic and Games Community are of the form:

Formula Φ is OK (true, provable, valid) iff the game G(Φ) has a winning strategy for the first player, where G(Φ) is obtained by some explicit construction.

Topic in these talks: This Reduction Mechanism
Which properties can be characterized this way ??

An Unfair Reduction

©Games Workshop

©Games Workshop

If Hoatlacotlincotitli faces an Opponent which is Worthy he will challenge her to a game of HEX where she moves first (and consequently she can win). Otherwise she is the First Player in a game of NIM with piles of sizes 5,6,9 and 10 (which she will loose if Hoatl plays well).

Hence: Only Worthy Opponents have a winning stategy ....

Peter van Emde Boas: The Games of Computer Science.

The Model: Games as Acceptors

Input X is mapped to some game G(X)

The mapping X → G(X) is easy to compute
(computable in Polynomial Time or Logarithmic Space)

Consequence: G(X) has a Polynomial Size Description.
(Leaving open what the Proper Descriptions are.)

L_G := { X | G(X) has a winning strategy for
the first player }

Which Languages L can be characterized in this way ?

COMPLEXITY ENDGAME ANALYSIS

Input Data:
Game G , Position p in G

Question: Is position p a winning position
for Thorgrim ?
for Urgat ?
a Draw ?

Relevant Issues: Game presentation,
Game structure (tree, graph, description)
Determinacy (Imperfect Information!)

The Impact of the Format

Thinking about simple games like Tic-Tac-Toe one considers the size of the game to be indicated by measures like:

-- size configuration (9 cells possibly with marks)

-- depth (duration) game (at most 9 moves)

The full game tree is much larger : 986410 nodes
(disregarding early terminated plays)

The size of the strategic form is beyond
imagination.....

What size measure should we use for complexity theory estimates ??

The Impact of the Format

The gap between the experienced size (Wood Measure : configuration size & depth) and the size of the game tree is Exponential !
Another Exponential Gap between the game tree and the strategic form.
These Gaps are highly relevant for Complexity!

Here: use configuration size and depth as size measures for input games. Estimate complexity of endgame analysis in terms of the Wood Measure.

Backward Induction in PSPACE?

The Standard Dynamic Programming Algorithm for Backward Induction uses the entire Configuration Graph as a Data Structure: Exponential Space !

Instead we can Use Recursion over Sequences of Moves:

This Recursion proceeds in the game tree from the Leaves to the Root.

Relevant issues: Draws possible? Terminating Game?
Loops ?

Backward Induction in PSPACE?

This Recursive scheme combines recursion (over move sequence) with iteration (over locally legal moves). Correct only for determinated games!

Space Consumption =
O(| Stackframe | . Recursion Depth)

| Stackframe | =
O(| Move sequence | + | Configuration|)

Recursion Depth = | Move sequence | =
O(Duration Game)

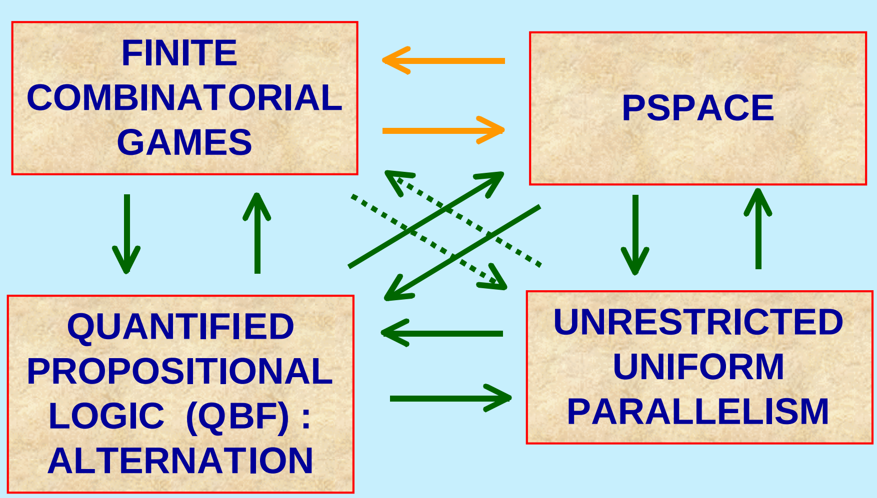
So the game duration should be polynomial!

REASONABLE GAMES

Assumptions for the sequel:

Finite Perfect Information (Zero Sum) Games
Structure: tree given by description,
where deciding properties like
is p a position ?, is p final ? is p starting
position ?, who has to move in p ?, and the
generation of successors of p are all trivial
problems ..... The tree can be generated
in time proportional to its size.....
Moreover the duration of a play is polynomial.

THE HOLY QUADRINITY



Peter van Emde Boas: The Games of Computer Science.

Known Hardness results on Games in Complexity Theory (1980+)

- QBF (PSPACE) (the “mother game”)
- Tiling Games (NP, PSPACE, NEXPTIME,....)
- Pebbling Game (PSPACE) (solitaire game!)
- Geography (PSPACE)
- HEX (generalized or pure) (PSPACE)
- Checkers, Go (PSPACE)
- Block Moving Problems (PSPACE)
- Chess (EXPTIME) (repetition of moves !)

The Common View is that Games Characterize PSPACE

Peter van Emde Boas: The Games of Computer Science.

Walter Savitch

© Peter van Emde Boas

© Peter van Emde Boas

© Peter van Emde Boas

ICSOR; CWI, Aug 1976 San Diego, Oct 1983

Proved PSPACE = NPSPACE around 1970

Peter van Emde Boas: The Games of Computer Science.

Polynomial Space Configuration Graph

- **Configurations & Transitions:**
  - (finite) State, Focus of Interaction & Memory Contents
  - Transitions are Local (involving State and Memory locations in Focus only; Focus may shift). Only a Finite number of Transitions in a Configuration
  - Input Space doesn´t count for Space Measure

Polynomial Space Configuration Graph

- **Exponential Size Configuration Graph:**
  - input lenght: $|x| = k$; Space bound: $S(k)$
  - Number of States: q (constant)
  - Number of Focus Locations: $k.S(k)^t$ (where t denotes the number of “heads”)
  - Number of Memory Contents: $C^{S(k)}$
  - Together: $q.k.S(k)^t.\ C^{S(k)} = 2^{O(S(k))}$ (assuming $S(k) = \Omega(\log(k))$)

Peter van Emde Boas: The Games of Computer Science.

The Savitch Game

Given: some input x for a PSPACE acceptor M
(M can be nondeterministic)

To Construct: a 2 person Complete Information reasonable Game G(M,x) such that
x is accepted by M iff the first player has a winning strategy in G(M,x)

WLOG: time accepting computation $\leq 2^{(S(|x|))}$

Peter van Emde Boas: The Games of Computer Science.

The Savitch Game

© Games Workshop

Aethis

Thorgrim

© Games Workshop

Typical Position:
Configurations C_1 , C_2 and Time Interval $t_1 < t_2$
$| C_1 | , | C_2 | \leq (S(|x|))$, $0 \leq t_1 < t_2 \leq 2^{(S(|x|))}$

ROUND of the Game :
Thorgrim chooses t_3 such that $t_1 < t_3 < t_2$
Aethis chooses C_3 at t_3
Thorgrim decides to continue with either
C_1 , C_3 and $t_1 < t_3$ or C_3 , C_2 and $t_3 < t_2$

Peter van Emde Boas: The Games of Computer Science.

The Savitch Game

Initial Position:
C_1 is the starting position and C_2 the (unique) accepting Configuration. $0 = t_1$ and $t_2 = 2^{(S(|x|))}$

Final Position: $t_2 - t_1 = 1$

Aethis wins if C_1 ---> C_2 is a legal transition; otherwise Thorgrim wins the game

Polynomial duration enforced by requiring
$(t_2 - t_1).\varepsilon \leq (t_3 - t_1) \leq (t_2 - t_1).(1-\varepsilon)$ for some fixed ε satisfying $0 < \varepsilon \leq 1/2$

The Savitch Game

Winning Strategies:

If x is accepted Aethis can win the game by being truthful (always play the true configuration in some Accepting Computation...)

If x is not accepted the assertion entailed by the initial position is false. Regardless the configuration C_3 chosen by Aethis he must make a false assertion either on the first or on the second interval (or both). Thorgim wins by always attacking the false interval....

Peter van Emde Boas: The Games of Computer Science.

The Savitch Game

The Punchline:

Endgame Analysis of the Savitch Game
is in Deterministic PSPACE,
even if the original acceptor was
Nondeterministic:

NPSPACE = PSPACE !

an Alternative (direct) proof of the Savitch Theorem....

The Savitch Game

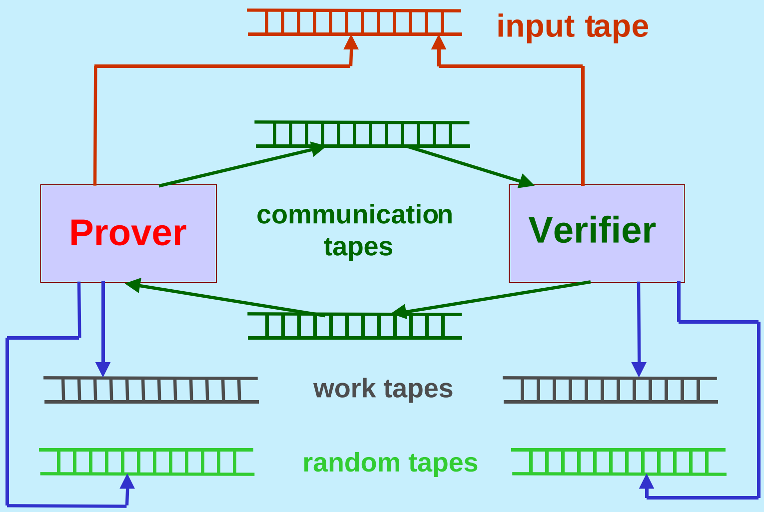
Final remarks:

Aethis can play his winning strategy if he knows the accepting computation.

Thorgrim can play his winning strategy if he can locate errors. Utterly unfeasible....

COMPARE THIS WITH INTERACTIVE PROTOCOLS: PSPACE = IP

The Basic Interactive Model



Using the Interactive Model

One of P or V opens the Communication

Next both Participants Exchange a Sequence of Messages, based on:

- Contents Private Memory
- Input
- Visible Coin Flips
- Earlier Messages (Send and) Received so far
- Current Message

At some point V decides to Accept the input (I am convinced - you win) or to Reject it (I don't Believe you - you loose)

Peter van Emde Boas: The Games of Computer Science.

Computational Assumptions

- Verifier is a P-time bounded Probabilistic Device
- Prover (in principle) can do everything (restrictions => feasibility)
- All messages and the number of messages are P-bounded.

Consequently, even if P can perform arbitrarily complex computations, it makes no sense to use these in order to generate complex messages, since V has to read them, and P could generate them using nondeterminism as well.

Accepting a Language L

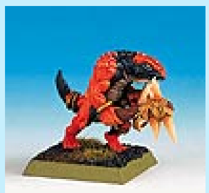
- For every x in L the Prover P has a Strategy which with High Probability will convince the Verifier
- For every x outside L , regardless the strategy followed by the Prover, the Verifier will reject with High Probability

IP = class of languages accepted by Interactive Proof Systems

The Participants

Thorgrim; our wise Prover

Urgat; our sceptical Verifier



Stragtos; fully deterministic

Orion; Random moves only

Various Models

Verifier vs. Prover

Stragtos vs. Orion:	Probabilistic Computation Rabin, Strassen Solovay
Orion vs. Thorgrim: unbounded error	Games against Nature Papadimitriou's model
Orion vs. Thorgrim:	Arthur Merlin Games Babai & Moran
Urgat vs. Thorgrim:	Interactive Protocols Goldwasser Micali Rackoff

Peter van Emde Boas: The Games of Computer Science.

Where is the Beef ?

The name of the area: Interactive Protocols, suggests that Interaction is the newly added ingredient.

Interaction already resides in the Alternating Computation Model!

The Key Addition therefore is Randomization.

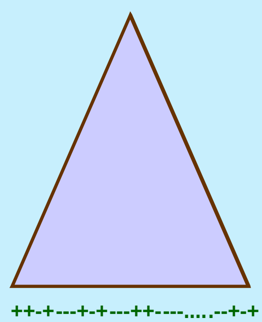
Leaf Languages

Nondeterministic Computation Tree with Ordered Binary Choices Everywhere.

Yields string of 2^T labels at leafs.

Accepts on the basis of some property of this string.

Backward Induction only for Regular properties (but where is the Game??)



Can Leaf Languages be analyzed by Games?

Incomplete Information Games

Things which go wrong:

-- Simple games no longer are determinated

-- Information sets capture uncertainty

-- Nodes may belong to multiple information sets: disambiguation causes exponential blow-up in size....

-- Uniform strategies are required

-- Earlier algorithms become incorrect if used on nodes without disambiguation

WANTED: a complexity theory for Incomplete Information Games......

CONCLUSIONS

- There exists a Link between Games and Computational Models
- Reasonable Games have PSPACE complete endgame analysis (but this tells more about reasonability than about PSPACE....)
- This Theory already existed around 1980 (but at that time Games were not taken serious....)
- Theory fails for Imperfect Information Games
- Unclear position Leaf Languages

Peter van Emde Boas: The Games of Computer Science.