Problem Solving on Simple Games via BDDs

Rudolf Berghammer and Stefan Bolus

University of Kiel, Germany

13. Sept. 2010

Yes-No-Voting

A pair $(N, \mathcal{W})$ is called a **simple game**, if N is a set of **players** and $\mathcal{W} \subseteq 2^N$, s.t.

$$\forall S, T \in 2^N : S \in \mathcal{W} \wedge S \subseteq T \Rightarrow T \in \mathcal{W} .$$

Subsets of N are called **coalitions**. Coalitions in $\mathcal{W}$ are called **winning**, and **losing** otherwise.

Weighted Voting Game (WVG): There has to be a weight $w_i \in \mathbb{N}$ for each player i, a quota $Q \in \mathbb{N}$ s.t. a coalition S wins, iff

$$\sum_{i \in S} w_i \geq Q .$$

Multiple Weighted Voting Game (MWVG): There are multiple WVGs and a winning coalition has to win in all of them.

Representation of Simple Games

Representations include the enumeration of ...

- ... Winning coalitions $\mathcal{W}$.
- ... Minimal winning coalitions $\mathcal{W}_{\text{min}}$.
- ... Shift-minimal winning coalitions $\mathcal{W}_{\text{shift}}$.

Gap

- Weights and quota(s) of (multiple) weighted voting games.

Problems: Enumeration of coalitions is practically impossible for many real world games. There is no apparent and exploitable relation between structure of e.g. winning coalitions and classes of simple games which is usable in algorithms.

Representation of Simple Games

We want something that is **explicit** like the winning coalitions, that is **compact in size** and that **maintains properties** (at least some) of the simple game's class.

Binary Decision Diagrams (BDDs)

- Can be used to represent Boolean functions and sets of subsets ($\Rightarrow$ simple games).
- Is a labeled, binary and directed acyclic graph $G = (V \cup \{\mathbb{I}, \mathbb{O}\}, E)$ with exactly one source (*r*oot) and two designated sinks.
- Labels are names of Boolean variables $1, \ldots, n$ (and $n + 1$ for the sinks).
- Sinks are called the *1-sink* and *0-sink* (denoted by $\mathbb{I}, \mathbb{O}$).
- Non-sinks $v \in V$ (*inner nodes*) have a *yes-edge* and a *no-edge*.
- Can have exponential size in n. But: Restricted classes can have polynomial size.

Special class: QOBDD.

Quasi-Reduced and Ordered BDDs (QOBDDs) by Example

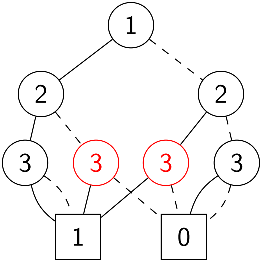
We start with Boolean variables $N = \{1, 2, 3\}$ and a truth table.

variable	1	2	3	f-value
	0	0	0	0
	0	0	1	0
	0	1	0	0
	0	1	1	1
	1	0	0	0
	1	0	1	1
	1	1	0	1
	1	1	1	1

Quasi-Reduced and Ordered BDDs (QOBDDs) by Example

This corresponds to the following binary decision **tree** with $2^{|N|} - 1$ nodes.

variable	1	2	3	f-value
	0	0	0	0
	0	0	1	0
	0	1	0	0
	0	1	1	1
	1	0	0	0
	1	0	1	1
	1	1	0	1
	1	1	1	1

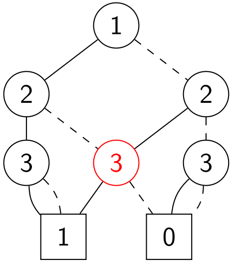


The red nodes are *equivalent* and could be merged.

Quasi-Reduced and Ordered BDDs (QOBDDs)

We obtain the QOBDD from an (ordered) binary decision tree by applying the following rule until no longer applicable.

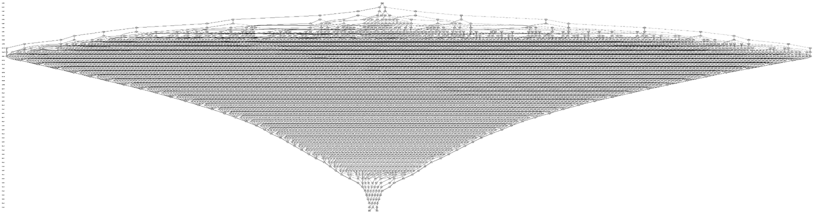
Merging Rule: If two nodes u, v have the same label, the same yes- and no-successors, all edges leading to v can be redirected to lead to u and v can be eliminated.



The **variable ordering** can make the QOBDD vary from polynomial to exponential size and vice-versa.

QOBDD Notions

The QOBDD G for the US Electoral College 2008 with 51 players, quota 270, 4558 nodes and $> 1.117 \cdot 10^{12}$ winning coalitions:

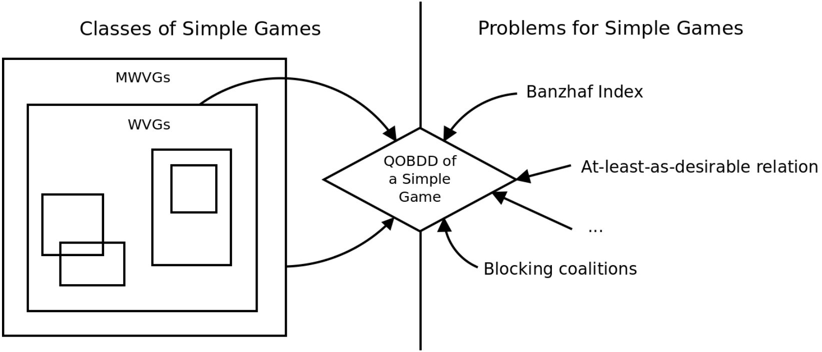


The rhombus like form is typical for QOBDDs. Some notions:

- **level** $i \triangleq$ all nodes with a given variable. E.g. level 1 has always one node (the root).
- **width(G)** $\triangleq$ maximum number of nodes over all levels.
- **size(G)** $\triangleq$ number of all inner nodes.

What can QOBDDs do for us?

Decouple classes of simple games and algorithms for solving problems. Building the QOBDD is now self-containted. Algorithms can be applied to **any** simple game:



Instead, use derived **properties** (like size, structure) of the simple game's class to prove something without changing the algorithm. Let's see ...

Some results

Given the QOBDD G of a simple game:

1. **Banzhaf indices** of all players is in $\mathcal{O}(\text{size}(G))$.
2. **Shapley-Shubik indices** of all players is in $\mathcal{O}(n^2 \cdot \text{size}(G))$.
3. Computation of the QOBDD for the blocking coalitions (**dual game**) is in $\mathcal{O}(\text{size}(G))$.
4. The test if player i **is at least as desirable as player** j is in $\mathcal{O}(n \cdot \text{width}(G)^2)$.
5. Computation of the QOBDD for the **minimal winning coalition** (in general) seems to be hard, though.

Some of them can be found in (Bolus 2010).

Some results (cont.)

Upper bounds of QOBDD sizes for simple game classes:

1. **WVG** with quota Q: $\text{size}(G) \leq n(Q+1)$.
2. **WVG** in general: $\text{size}(G) \in \mathcal{O}(2^{\frac{n}{2}})$ (Hosaka, Takenaga, and Yajima 1994)
3. **MWVG** with quotas $Q_1 \geq \cdots \geq Q_m$: $\text{size}(G) \leq n \prod_{t=1}^{m}(Q_t + 1)$.
4. If additionally any player in any sub-WVG has non-zero weight then $\text{size}(G) \leq nQ_1 \cdots Q_m$.
5. **Unbalanced WVGs***: $\text{size}(G) \in \mathcal{O}(n)$.
6. **Sequential WVGs***: $\text{size}(G) \in \mathcal{O}(n^2)$.

*: With a special ordering of the players.

Some results (cont.)

Structural properties of QOBDD for simple game classes:

1. Computation of the minimal winning coalitions for a WVGs (with a special ordering of the players) is in $\mathcal{O}(\text{size}(G))$.
2. Hence, **Public Good Index** of all players is in $\mathcal{O}(\text{size}(G))$ for WVGs and
3. **Deegan-Packel indices** of all players is in $\mathcal{O}(n^2 \cdot \text{size}(G))$ for WVGs.

The algorithm for the minimal winning coalitions has slightly changed here. However, the idea hasn't changed.

Example: Minimal winning coalitions

Applicable to any QOBDD representing a simple game:

MinWin(v)
  if $v = \mathbb{O}$ **or** $v = \mathbb{I}$ **then return** v
  else if $T(v) = E(v)$ **then**
    return ite(var(v), $\mathbb{O}_{\text{var}(v)+1}$, MinWin($E(v)$))
  else return ite(var(v), MinWin($T(v)$) $\setminus e$), e)
    where $e :=$ MinWin($E(v)$))

Example: Minimal winning coalitions

Applicable only to QOBDDs representing a WVG:

MinWin(v)
  if $v = \mathbb{O}$ **or** $v = \mathbb{I}$ **then return** v
  else if $T(v) = E(v)$ **then**
    return ite(var(v), $\mathbb{O}_{\text{var}(v)+1}$, MinWin($E(v)$))
  else return ite(var(v), MinWin($T(v)$), e)
    where $e :=$ MinWin($E(v)$)

Open problems and future work

- Identification of key players (dominant, veto, dummy, ...).

Open problems and future work

- Identification of key players (dominant, veto, dummy, ...).
- Computation of shift-minimal winning coalitons and the recently proposed shift power index (Alonso-Meijide and Freixas 2010).

Open problems and future work

- Identification of key players (dominant, veto, dummy, ...).
- Computation of shift-minimal winning coalitons and the recently proposed shift power index (Alonso-Meijide and Freixas 2010).
- Power indices for simple games with a-priori unions (Alonso-Meijide and Fiestras-Janeiro 2002).

Open problems and future work

- Identification of key players (dominant, veto, dummy, ...).
- Computation of shift-minimal winning coalitons and the recently proposed shift power index (Alonso-Meijide and Freixas 2010).
- Power indices for simple games with a-priori unions (Alonso-Meijide and Fiestras-Janeiro 2002).
- A practicable test for being a WVG or trade-robustness (maybe similar to (Coates and Lewis 1961)).

Open problems and future work

- Identification of key players (dominant, veto, dummy, ...).
- Computation of shift-minimal winning coalitons and the recently proposed shift power index (Alonso-Meijide and Freixas 2010).
- Power indices for simple games with a-priori unions (Alonso-Meijide and Fiestras-Janeiro 2002).
- A practicable test for being a WVG or trade-robustness (maybe similar to (Coates and Lewis 1961)).
- Consider more classes of simple games. E.g., are there classes which are harder to manipulate?

Open problems and future work

- Identification of key players (dominant, veto, dummy, ...).
- Computation of shift-minimal winning coalitons and the recently proposed shift power index (Alonso-Meijide and Freixas 2010).
- Power indices for simple games with a-priori unions (Alonso-Meijide and Fiestras-Janeiro 2002).
- A practicable test for being a WVG or trade-robustness (maybe similar to (Coates and Lewis 1961)).
- Consider more classes of simple games. E.g., are there classes which are harder to manipulate?
- Consider *n*-ary decision diagrams. Especially 3-ary DDs for **ternary voting games** (Felsenthal and Machover 1997).

Thank you!

Alonso-Meijide, J. M. and M. G. Fiestras-Janeiro (2002).
Modification of the Banzhaf Value for Games with a Coalition Structure.
Annals of Operations Research 109(1), 213–227.

Alonso-Meijide, J. M. and J. Freixas (2010).
A new power index based on minimal winning coalitions without any surplus.
Decision Support Systems 49(1), 70–76.

Bolus, S. (2010).
Power indices of simple games and vector-weighted majority games by means of binary decision diagrams.
Submitted.

Coates, C. L. and P. M. Lewis (1961).
Linearly separable switching functions.
Journal of the Franklin Institute 272(5), 366–410.

Felsenthal, D. S. and M. Machover (1997, October).
Ternary voting games.
International Journal of Game Theory 26(3), 335–351.

Hosaka, K., Y. Takenaga, and S. Yajima (1994).
On the Size of Ordered Binary Decision Diagrams Representing Threshold Functions.
In *ISAAC '94: Proceedings of the 5th International Symposium on Algorithms and Computation*, London, UK, pp. 584–592. Springer-Verlag.