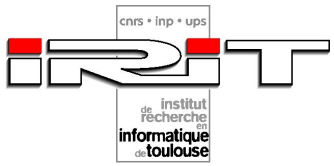
A Knowledge Compilation Map of Set-labeled Diagrams

Alexandre Niveau
Hélène Fargier Cédric Pralet



July 16th, 2011

Outline

Outline

Introductory Example

- A product configuration problem: customized tee-shirts.
  [Hadzic et al., 2008]

[Hadzic et al., 2008] Hadzic, T., Hansen, E., and B. O'Sullivan, B. (2008). On Automata, MDDs and BDDs in Constraint Satisfaction. In *ECAI Workshop on Inference methods based on Graphical Structures of Knowledge (WIGSK)*.

Introductory Example

- A product configuration problem: customized tee-shirts.
  [Hadzic et al., 2008]
- parameters:
  - print – "Men in Black" (MiB) or "Save the Whales" (StW)
  - color – black or blue
  - size – small or large
  - sleeves – with or without

[Hadzic et al., 2008] Hadzic, T., Hansen, E., and B. O'Sullivan, B. (2008). On Automata, MDDs and BDDs in Constraint Satisfaction. In *ECAI Workshop on Inference methods based on Graphical Structures of Knowledge (WIGSK)*.

Introductory Example

- A product configuration problem: customized tee-shirts.
  [Hadzic et al., 2008]
- parameters:
  - print – "Men in Black" (MiB) or "Save the Whales" (StW)
  - color – black or blue
  - size – small or large
  - sleeves – with or without
- rules:
  - "MiB" tee-shirts must be black
  - "StW" picture does not fit on small sized tee-shirts
  - large sized tee-shirts can't be sleeveless

[Hadzic et al., 2008] Hadzic, T., Hansen, E., and B. O'Sullivan, B. (2008). On Automata, MDDs and BDDs in Constraint Satisfaction. In *ECAI Workshop on Inference methods based on Graphical Structures of Knowledge (WIGSK)*.

Introductory Example

- A product configuration problem: customized tee-shirts.
  [Hadzic et al., 2008]
- parameters:
  - print – "Men in Black" (MiB) or "Save the Whales" (StW)
  - color – black or blue
  - size – small or large
  - sleeves – with or without
- rules:
  - "MiB" tee-shirts must be black
  - "StW" picture does not fit on small sized tee-shirts
  - large sized tee-shirts can't be sleeveless
- the program must be able to tell whether each choice respects the rules

---

[Hadzic et al., 2008] Hadzic, T., Hansen, E., and B. O'Sullivan, B. (2008). On Automata, MDDs and BDDs in Constraint Satisfaction. In *ECAI Workshop on Inference methods based on Graphical Structures of Knowledge (WIGSK)*.

Problematic

- the configurable product is represented by a Boolean formula: each variable corresponds to a choice; models are possible configurations

Problematic

- the configurable product is represented by a Boolean formula: each variable corresponds to a choice; models are possible configurations
- configuration process:
  - each choice fixes a value for some variable → *restriction*
  - is the function still *satisfiable*? → SAT problem

Problematic

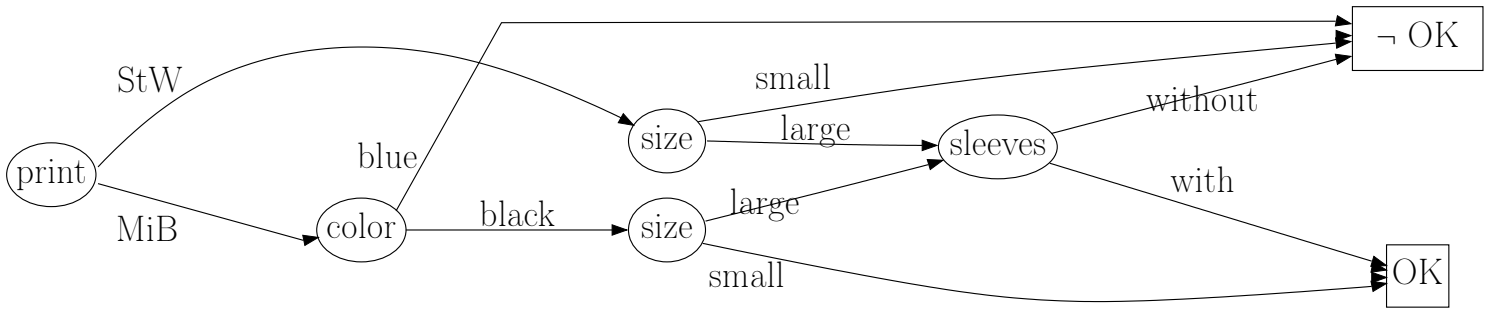
- the configurable product is represented by a Boolean formula: each variable corresponds to a choice; models are possible configurations
- configuration process:
  - each choice fixes a value for some variable → *restriction*
  - is the function still *satisfiable*? → SAT problem
- NP-complete, but the user doesn't want to wait too long after each choice (real-life configuration problems can have hundreds of multi-valued variables!...)

A Solution: Knowledge Compilation

- configurable product = a non-varying, huge function, usually represented in a compact way, *e.g.* as a CSP

A Solution: Knowledge Compilation

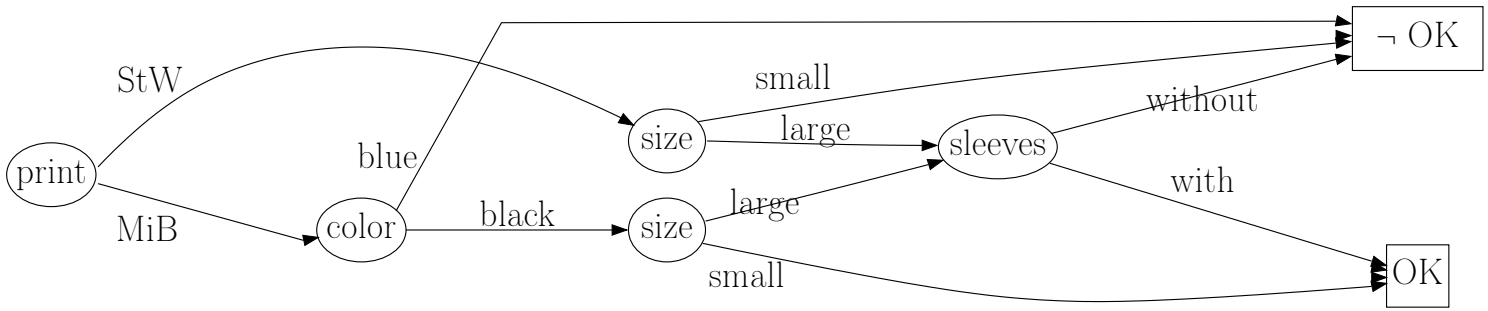
- configurable product = a non-varying, huge function, usually represented in a compact way, *e.g.* as a CSP
- *compile* it as an OBDD



- fixing variables values (*conditioning*) and SAT are polytime on OBDDs → the user's wait is reduced!

A Solution: Knowledge Compilation

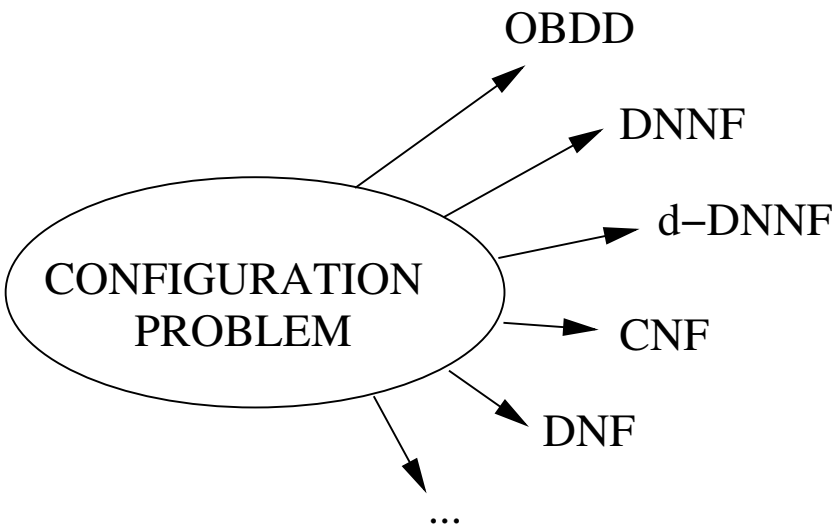
- configurable product = a non-varying, huge function, usually represented in a compact way, *e.g.* as a CSP
- *compile* it as an OBDD



- fixing variables values (*conditioning*) and SAT are polytime on OBDDs → the user's wait is reduced!
- *knowledge compilation*: offline translating the fixed part of a problem (pre-processing, which may be hard) for online operations to be tractable

Outline

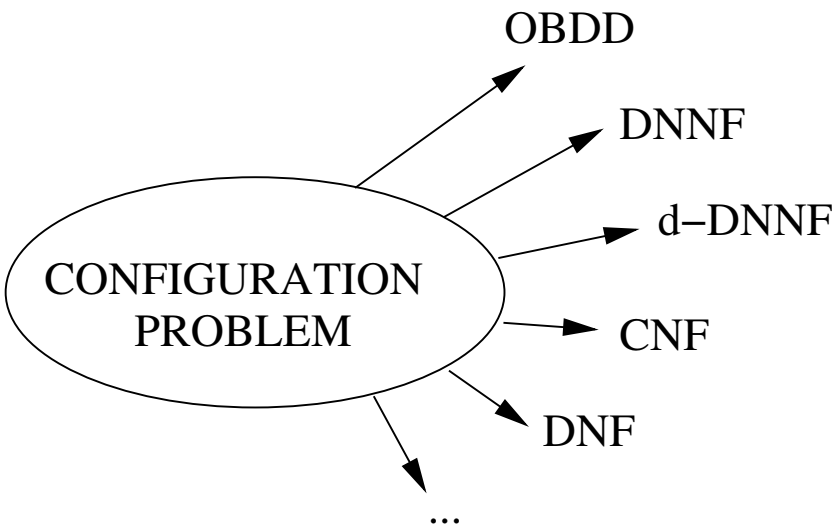
Choosing a Target Language



- what is the most appropriate for my application?

[Darwiche and Marquis, 2002] Darwiche, A. and Marquis, P. (2002). A Knowledge Compilation Map. *JAIR*, 17:229–264.

Choosing a Target Language



- what is the most appropriate for my application? → use the knowledge compilation map [Darwiche and Marquis, 2002]
- helps to choose languages by comparing them according to two criteria:
  - capacity to answer requests efficiently
  - succinctness.

[Darwiche and Marquis, 2002] Darwiche, A. and Marquis, P. (2002). A Knowledge Compilation Map. *JAIR*, 17:229–264.

Knowledge Compilation Map: Requests

- all online manipulations boil down to elementary *queries* and *transformations*
- Most usual ones:
  - **CO**: consistency (does the function have a model?)
  - **VA**: validity (does the function have a countermodel?)
  - **CE**: clausal entailment (does $x = \top$ hold for all models?)
  - **CD**: conditioning (add the restriction $x = \top$ to the function)
  - **FO**: forgetting (project the function on a subset of variables)
  - $\wedge$**C**, $\vee$**C**: closure under conjunction, disjunction (compute the conjunction or disjunction of some functions)

Knowledge Compilation Map: Succinctness

- Succinctness relation: L_1 is *at least as succinct* as L_2 ($\mathrm{L}_1 \leq_s \mathrm{L}_2$) if and only if for every L_2 formula, there exists an equivalent polysize L_1 formula.

Knowledge Compilation Map: Succinctness

- Succinctness relation: L_1 is *at least as succinct* as L_2 ($\mathrm{L}_1 \leq_s \mathrm{L}_2$) if and only if for every L_2 formula, there exists an equivalent polysize L_1 formula.

- $\mathtt{DNNF} <_s \mathtt{OBDD}$: no formula is exponentially bigger when compiled as a DNNF than when compiled as an OBDD; there exists formulæ being exponentially bigger as an OBDD than as a DNNF

Knowledge Compilation Map: Succinctness

- Succinctness relation: L_1 is *at least as succinct* as L_2 ($\mathrm{L}_1 \leq_s \mathrm{L}_2$) if and only if for every L_2 formula, there exists an equivalent polysize L_1 formula.

- $\mathtt{DNNF} <_s \mathtt{OBDD}$: no formula is exponentially bigger when compiled as a DNNF than when compiled as an OBDD; there exists formulæ being exponentially bigger as an OBDD than as a DNNF

- $\mathtt{CNF}$ and $\mathtt{DNF}$ are incomparable w.r.t. succinctness: sometimes $\mathtt{CNF}$ is better, sometimes it is $\mathtt{DNF}$

Example of Use

L	CO	VA	CE	IM	EQ	SE	CT	ME
NNF	○	○	○	○	○	○	○	○
DNNF	√	○	√	○	○	○	○	√
d-DNNF	√	√	√	√	?	○	√	√
BDD	○	○	○	○	○	○	○	○
OBDD	√	√	√	√	√	○	√	√
DNF	√	○	√	○	○	○	○	√
CNF	○	√	○	√	○	○	○	○

L	CD	FO	SFO	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
NNF	√	○	√	√	√	√	√	√
DNNF	√	√	√	○	○	√	√	○
d-DNNF	√	○	○	○	○	○	○	?
BDD	√	○	√	√	√	√	√	√
OBDD	√	•	√	•	○	•	○	√
DNF	√	√	√	•	√	√	√	•
CNF	√	○	√	√	√	•	√	•

Example of Use

L	CO	VA	CE	IM	EQ	SE	CT	ME
NNF	○	○	○	○	○	○	○	○
DNNF	√	○	√	○	○	○	○	√
d-DNNF	√	√	√	√	?	○	√	√
BDD	○	○	○	○	○	○	○	○
OBDD	√	√	√	√	√	○	√	√
DNF	√	○	√	○	○	○	○	√
CNF	○	√	○	√	○	○	○	○

L	CD	FO	SFO	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
NNF	√	○	√	√	√	√	√	√
DNNF	√	√	√	○	○	√	√	○
d-DNNF	√	○	○	○	○	○	○	?
BDD	√	○	√	√	√	√	√	√
OBDD	√	•	√	•	○	•	○	√
DNF	√	√	√	•	√	√	√	•
CNF	√	○	√	√	√	•	√	•

Example of Use

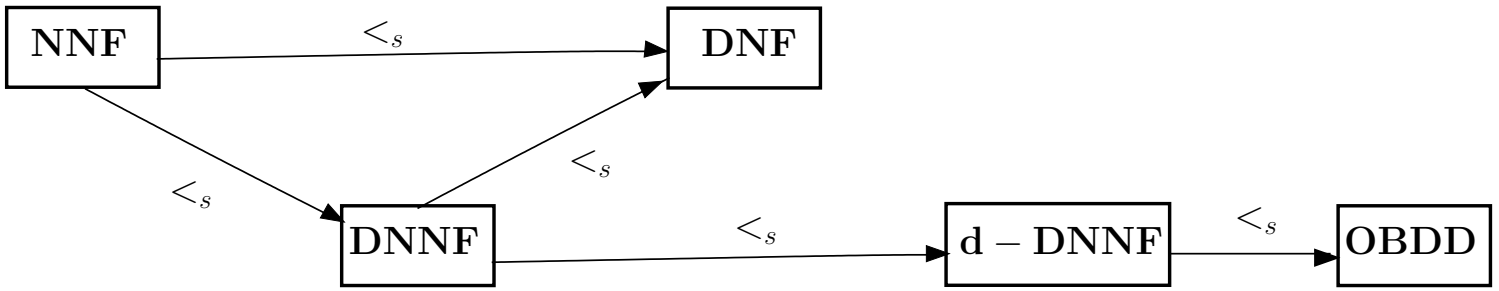
L	CO	VA	CE	IM	EQ	SE	CT	ME
NNF	○	○	○	○	○	○	○	○
DNNF	√	○	√	○	○	○	○	√
d-DNNF	√	√	√	√	?	○	√	√
BDD	○	○	○	○	○	○	○	○
OBDD	√	√	√	√	√	○	√	√
DNF	√	○	√	○	○	○	○	√
CNF	○	√	○	√	○	○	○	○

L	CD	FO	SFO	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
NNF	√	○	√	√	√	√	√	√
DNNF	√	√	√	○	○	√	√	○
d-DNNF	√	○	○	○	○	○	○	?
BDD	√	○	√	√	√	√	√	√
OBDD	√	•	√	•	○	•	○	√
DNF	√	√	√	•	√	√	√	•
CNF	√	○	√	√	√	•	√	•

Example of Use

L	CO	VA	CE	IM	EQ	SE	CT	ME
NNF	○	○	○	○	○	○	○	○
DNNF	√	○	√	○	○	○	○	√
d-DNNF	√	√	√	√	?	○	√	√
BDD	○	○	○	○	○	○	○	○
OBDD	√	√	√	√	√	○	√	√
DNF	√	○	√	○	○	○	○	√
CNF	○	√	○	√	○	○	○	○

L	CD	FO	SFO	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
NNF	√	○	√	√	√	√	√	√
DNNF	√	√	√	○	○	√	√	○
d-DNNF	√	○	○	○	○	○	○	?
BDD	√	○	√	√	√	√	√	√
OBDD	√	•	√	•	○	•	○	√
DNF	√	√	√	•	√	√	√	•
CNF	√	○	√	√	√	•	√	•



Multivalued Languages

- the map is drawn for languages with Boolean variables only

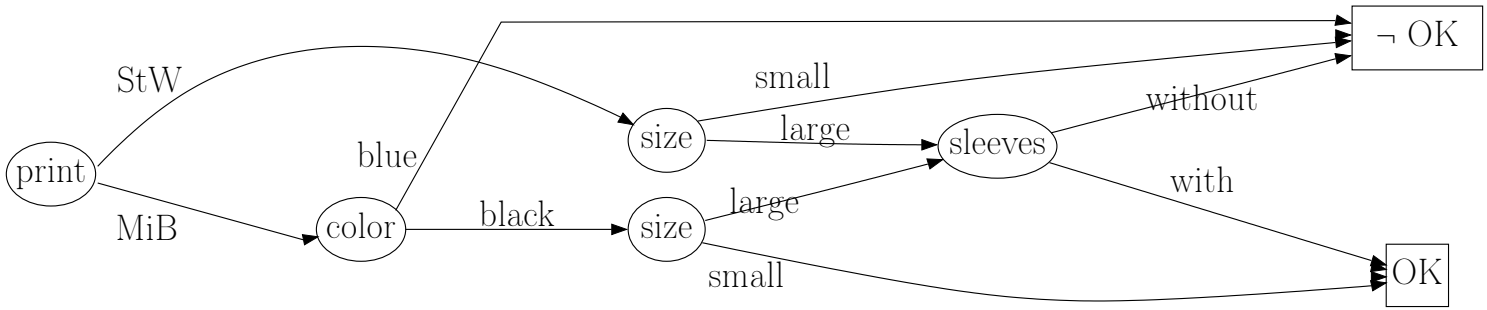
Multivalued Languages

- the map is drawn for languages with Boolean variables only
- multivalued variables: useful *e.g.* for configuration problems
  - Multivalued Decision Diagrams (MDDs)
  - AND/OR MDDs
  - Arithmetic Circuits...
- *our work*: adding multivalued languages to the map.

Outline

Multivalued Decision Diagrams

- MDDs: direct generalization of OBDDs to multivalued variables [Srinivasan et al., 1990]

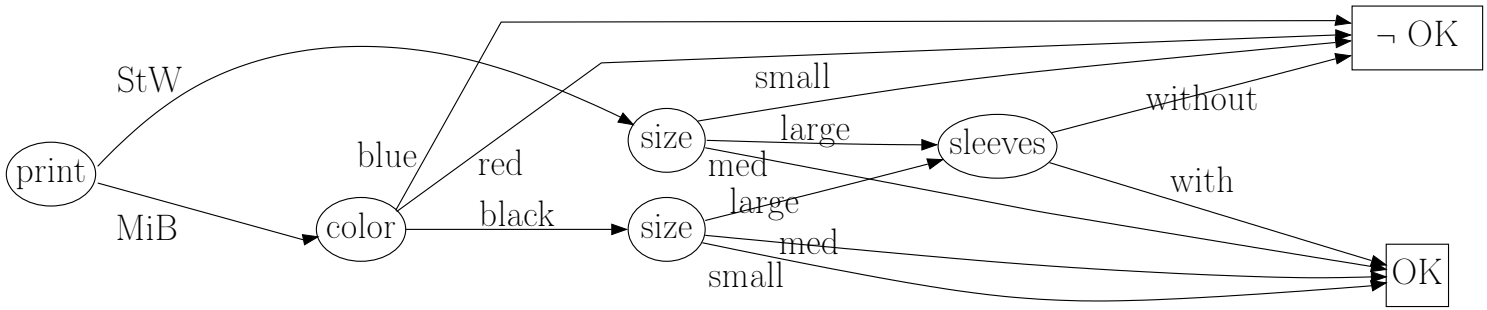


[Srinivasan et al., 1990] Srinivasan, A., Kam, T., Malik, S., and Brayton, R. K. (1990). Algorithms for Discrete Function Manipulation. In *ICCAD*, pages 92–95.

[Vempaty, 1992] Vempaty, N. R. (1992). Solving Constraint Satisfaction Problems Using Finite State Automata. In *AAAI*, pages 453–458.

Multivalued Decision Diagrams

- MDDs: direct generalization of OBDDs to multivalued variables [Srinivasan et al., 1990]

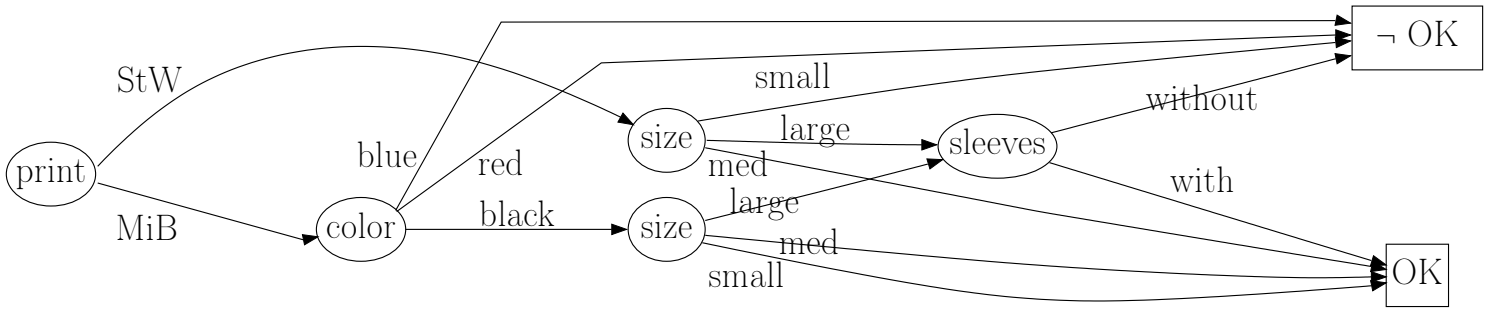


[Srinivasan et al., 1990] Srinivasan, A., Kam, T., Malik, S., and Brayton, R. K. (1990). Algorithms for Discrete Function Manipulation. In *ICCAD*, pages 92–95.

[Vempaty, 1992] Vempaty, N. R. (1992). Solving Constraint Satisfaction Problems Using Finite State Automata. In *AAAI*, pages 453–458.

Multivalued Decision Diagrams

- MDDs: direct generalization of OBDDs to multivalued variables [Srinivasan et al., 1990]



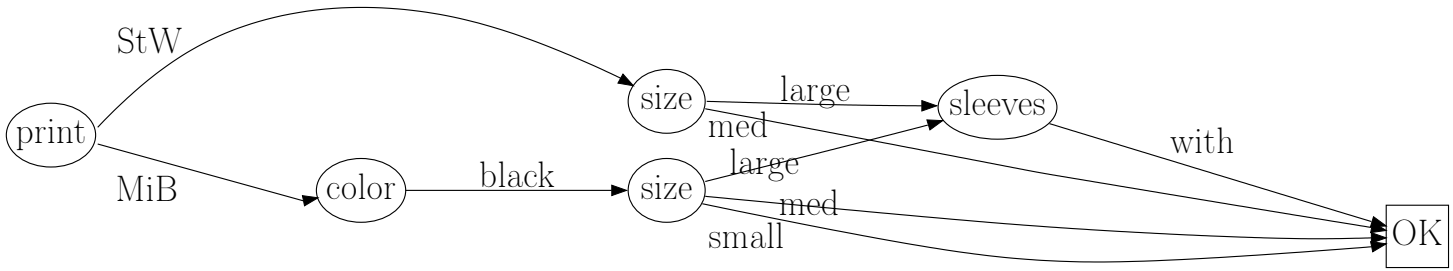
- we use the "automaton" format of MDDs [Vempaty, 1992]

[Srinivasan et al., 1990] Srinivasan, A., Kam, T., Malik, S., and Brayton, R. K. (1990). Algorithms for Discrete Function Manipulation. In *ICCAD*, pages 92–95.

[Vempaty, 1992] Vempaty, N. R. (1992). Solving Constraint Satisfaction Problems Using Finite State Automata. In *AAAI*, pages 453–458.

Multivalued Decision Diagrams

- MDDs: direct generalization of OBDDs to multivalued variables [Srinivasan et al., 1990]



- we use the "automaton" format of MDDs [Vempaty, 1992]

[Srinivasan et al., 1990] Srinivasan, A., Kam, T., Malik, S., and Brayton, R. K. (1990). Algorithms for Discrete Function Manipulation. In *ICCAD*, pages 92–95.

[Vempaty, 1992] Vempaty, N. R. (1992). Solving Constraint Satisfaction Problems Using Finite State Automata. In *AAAI*, pages 453–458.

KC Properties of MDDs

L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
MDD	√	√	√	√	√	√	∘	√	√	√	√
$MDD_<$	√	√	√	√	√	√	√	√	√	√	√

L	CD	SCD	$\wedge$tC	FO	SFO	EN	SEN	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\top$C
MDD	√	√	√	•	•	•	•	•	∘	•	∘	√
$MDD_<$	√	√	√	•	•	•	•	•	√	•	√	√

- Results mainly found in/deduced from the literature

  [Kam and Brayton, 1990] [Darwiche and Marquis, 2002]

[Kam and Brayton, 1990] Kam, T. and Brayton, R. K. (1990). Multi-valued decision diagrams. Master's thesis.

KC Properties of MDDs

L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
MDD	√	√	√	√	√	√	○	√	√	√	√
$MDD_<$	√	√	√	√	√	√	√	√	√	√	√

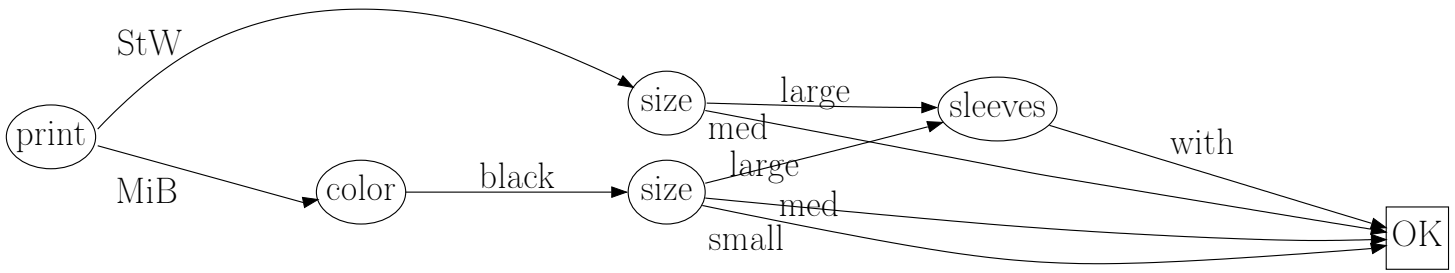
L	CD	SCD	$\wedge$tC	FO	SFO	EN	SEN	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
MDD	√	√	√	•	•	•	•	•	○	•	○	√
$MDD_<$	√	√	√	•	•	•	•	•	√	•	√	√

- Results mainly found in/deduced from the literature
  [Kam and Brayton, 1990] [Darwiche and Marquis, 2002]

- Basically the same as OBDDs except for **SFO** and **SEN**: due to domain size being unknown, the usual decomposition $\bigvee_{i \in \mathrm{Dom}(x)} \phi_{|x=i}$ gives $\phi_{|x=1} \vee \cdots \vee \phi_{|x=n}$
  $\rightarrow$ size depending on n

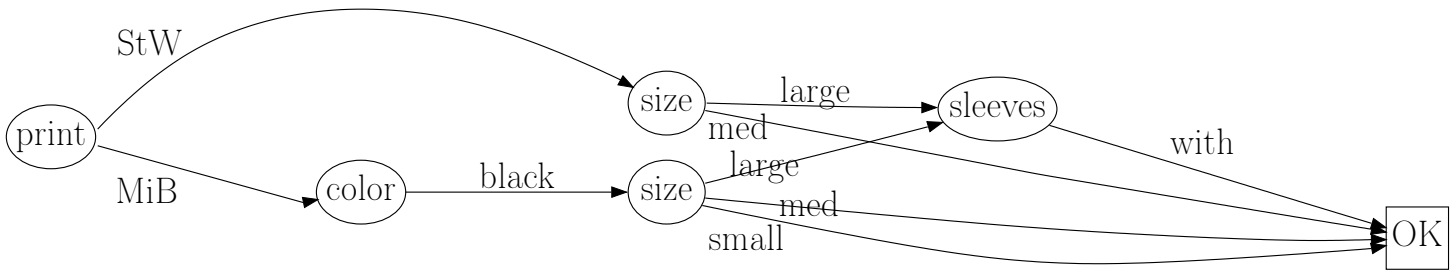
[Kam and Brayton, 1990] Kam, T. and Brayton, R. K. (1990). Multi-valued decision diagrams. Master's thesis.

Set-labeled Diagrams



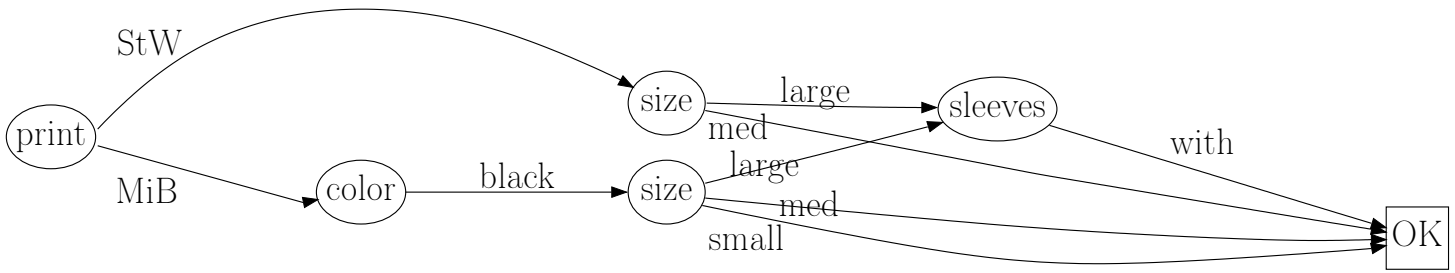
- usual definition requires ordering and determinism; we considered relaxing these constraints

Set-labeled Diagrams



- usual definition requires ordering and determinism; we considered relaxing these constraints
- we also allow edges to be labeled by *sets* rather than single values

Set-labeled Diagrams



- usual definition requires ordering and determinism; we considered relaxing these constraints
- we also allow edges to be labeled by *sets* rather than single values
- we call the general language "set-labeled diagrams" (SDs)

KC Properties of SDs and dSDs

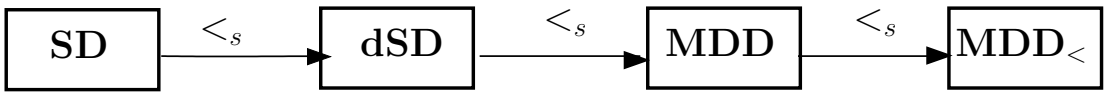
L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
SD	○	○	√	○	○	○	○	○	○	○	○
dSD	○	○	√	○	○	○	○	○	○	○	○
MDD	√	√	√	√	√	√	○	√	√	√	√
$MDD_<$	√	√	√	√	√	√	√	√	√	√	√

L	CD	SCD	$\wedge$tC	FO	SFO	EN	SEN	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
SD	√	√	√	○	√	○	√	√	√	√	√	?
dSD	√	√	√	○	√	○	√	√	√	√	√	√
MDD	√	√	√	•	•	•	•	•	○	•	○	√
$MDD_<$	√	√	√	•	•	•	•	•	√	•	√	√

KC Properties of SDs and dSDs

L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
SD	○	○	√	○	○	○	○	○	○	○	○
dSD	○	○	√	○	○	○	○	○	○	○	○
MDD	√	√	√	√	√	√	○	√	√	√	√
$\text{MDD}_<$	√	√	√	√	√	√	√	√	√	√	√

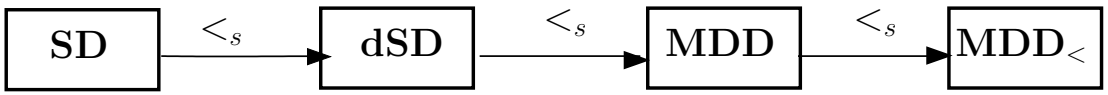
L	CD	SCD	$\wedge$tC	FO	SFO	EN	SEN	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
SD	√	√	√	○	√	○	√	√	√	√	√	?
dSD	√	√	√	○	√	○	√	√	√	√	√	√
MDD	√	√	√	•	•	•	•	•	○	•	○	√
$\text{MDD}_<$	√	√	√	•	•	•	•	•	√	•	√	√



KC Properties of SDs and dSDs

L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
SD	○	○	√	○	○	○	○	○	○	○	○
dSD	○	○	√	○	○	○	○	○	○	○	○
MDD	√	√	√	√	√	√	○	√	√	√	√
$\text{MDD}_<$	√	√	√	√	√	√	√	√	√	√	√

L	CD	SCD	$\wedge$tC	FO	SFO	EN	SEN	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\top$C
SD	√	√	√	○	√	○	√	√	√	√	√	?
dSD	√	√	√	○	√	○	√	√	√	√	√	√
MDD	√	√	√	•	•	•	•	•	○	•	○	√
$\text{MDD}_<$	√	√	√	•	•	•	•	•	√	•	√	√

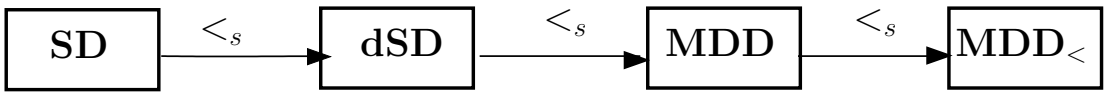


- $\text{dSD} <_s \text{MDD}$: adding ordering can lead to exponential loss in space
- does $\text{SD} <_s \text{dSD}$?

KC Properties of SDs and dSDs

L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
SD	○	○	√	○	○	○	○	○	○	○	○
dSD	○	○	√	○	○	○	○	○	○	○	○
MDD	√	√	√	√	√	√	○	√	√	√	√
$MDD_<$	√	√	√	√	√	√	√	√	√	√	√

L	CD	SCD	$\prec$tC	FO	SFO	EN	SEN	$\prec$C	$\prec$BC	$>$C	$>$BC	$\ulcorner$C
SD	√	√	√	○	√	○	√	√	√	√	√	?
dSD	√	√	√	○	√	○	√	√	√	√	√	√
MDD	√	√	√	•	•	•	•	•	○	•	○	√
$MDD_<$	√	√	√	•	•	•	•	•	√	•	√	√



- dSD $<_s$ MDD: adding ordering can lead to exponential loss in space
- does SD $<_s$ dSD?
- neither SD nor dSD support **CO**

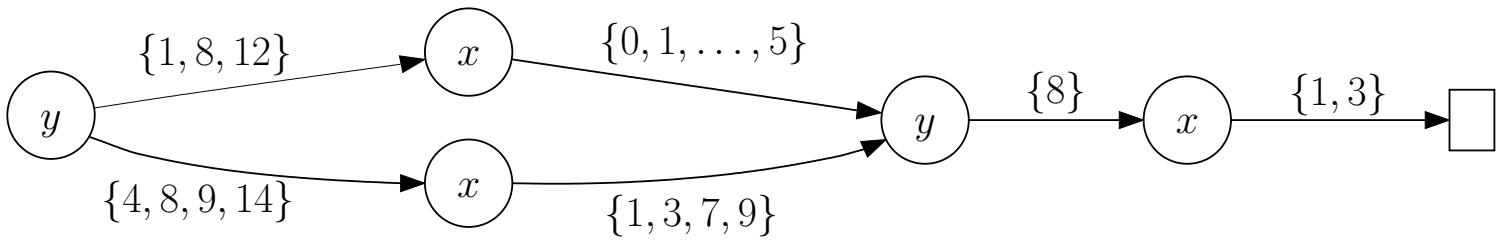
Focusingness

- the set labeling allows to define an interesting structural restriction allowing **CO** in polytime: *focusingness*

Focusingness

- the set labeling allows to define an interesting structural restriction allowing **CO** in polytime: *focusingness*

- focusingness imposes that sets pertaining to a given variable have a "nested" structure



KC Properties of FSDs and dFSDs

L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
SD	○	○	√	○	○	○	○	○	○	○	○
dSD	○	○	√	○	○	○	○	○	○	○	○
FSD	√	○	√	√	○	○	○	√	√	○	√
dFSD	√	√	√	√	√	?	○	√	√	?	√
MDD	√	√	√	√	√	√	○	√	√	√	√
$MDD_<$	√	√	√	√	√	√	√	√	√	√	√

L	CD	SCD	$\wedge$tC	FO	SFO	EN	SEN	$<$C	$\wedge$BC	$>$C	$>$BC	$\top$C
SD	√	√	√	○	√	○	√	√	√	√	√	?
dSD	√	√	√	○	√	○	√	√	√	√	√	√
FSD	√	√	√	√	√	○	○	○	○	√	√	○
dFSD	√	√	√	○	○	○	○	○	○	○	○	?
MDD	√	√	√	•	•	•	•	•	○	•	○	√
$MDD_<$	√	√	√	•	•	•	•	•	√	•	√	√

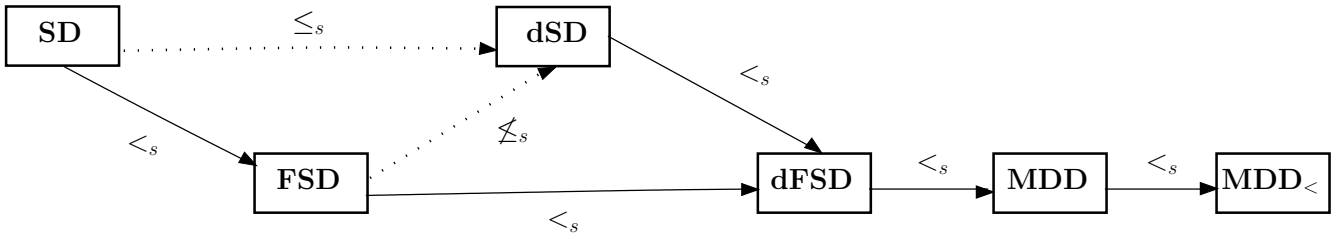
- exact same results as DNNF and d-DNNF. However, neither FSD nor dFSD are decomposable

KC Properties of FSDs and dFSDs

L	CO	VA	MC	CE	IM	EQ	SE	MX	CX	CT	ME
SD	○	○	√	○	○	○	○	○	○	○	○
dSD	○	○	√	○	○	○	○	○	○	○	○
FSD	√	○	√	√	○	○	○	√	√	○	√
dFSD	√	√	√	√	√	?	○	√	√	?	√
MDD	√	√	√	√	√	√	○	√	√	√	√
$MDD_<$	√	√	√	√	√	√	√	√	√	√	√

L	CD	SCD	$\wedge$tC	FO	SFO	EN	SEN	$\wedge$C	$\wedge$BC	$\vee$C	$\vee$BC	$\neg$C
SD	√	√	√	○	√	○	√	√	√	√	√	?
dSD	√	√	√	○	√	○	√	√	√	√	√	√
FSD	√	√	√	√	√	○	○	○	○	√	√	○
dFSD	√	√	√	○	○	○	○	○	○	○	○	?
MDD	√	√	√	•	•	•	•	•	○	•	○	√
$MDD_<$	√	√	√	•	•	•	•	•	√	•	√	√

- exact same results as DNNF and d-DNNF. However, neither FSD nor dFSD are decomposable



- SD $<_s$ FSD, dSD $<_s$ dFSD: imposing focusingness leads to exponential loss in space
- FSD $<_s$ dFSD: relaxing determinism on focusing structures leads to exponential gain in space

Summary

- we started to explore the compilation map of multivalued-languages
- include MDD in the map, as well as languages obtained by relaxing structural constraints: SD and dSD
- introduced languages based on focusingness

Perspectives

- almost done: explore KC properties of non-deterministic MDDs, and free MDDs/read-once SDs

  difficult remaining questions: determine whether $\mathtt{dSD} \leq_s \mathtt{SD}, \mathtt{FSD}$ and especially whether $\mathtt{FSD}$ is strictly more succinct than $\mathtt{RSD}$ (read-once SDs)

Perspectives

- almost done: explore KC properties of non-deterministic MDDs, and free MDDs/read-once SDs

  difficult remaining questions: determine whether $\mathtt{dSD} \leq_s \mathtt{SD}, \mathtt{FSD}$ and especially whether $\mathtt{FSD}$ is strictly more succinct than $\mathtt{RSD}$ (read-once SDs)

- to do: extend the map to capture tree-ordered decision diagrams, *i.e.* AOMDDs/tree-driven automata

Thanks for your attention!