Boolean Leximax Optimisation using Iterative SAT Solving

Miguel Cabral[1] Mikoláš Janota[2] Vasco Manquinho[1]

SAT 2022

[1] Universidade de Lisboa
[2] Czech Technical University in Prague

Package Upgradeability Problem

```
$ apt install libreoffice
```

Package Upgradeability Problem

`$ apt install libreoffice`

`libreoffice` requires:

- `libreoffice-writer`
- `libreoffice-calc`

Package Upgradeability Problem

`$ apt install libreoffice`

`libreoffice` requires:

- `libreoffice-writer`
- `libreoffice-calc`

`libreoffice-calc` requires:

- `ure`

Package Upgradeability Problem

`$ apt install libreoffice`

`libreoffice` requires:

- `libreoffice-writer`
- `libreoffice-calc`

`libreoffice-calc` requires:

- `ure` ← several versions

Package Upgradeability Problem

`$ apt install libreoffice`

`libreoffice` requires:

- `libreoffice-writer`
- `libreoffice-calc`

`libreoffice-calc` requires:

- `ure` $\leftarrow$ several versions

`ure`, 7645 requires:

- `debconf` ≥ 1675 or `debconf-2.0`

Package Upgradeability Problem

`$ apt install libreoffice`

`libreoffice` **requires:**

- `libreoffice-writer`
- `libreoffice-calc`

`libreoffice-calc` **requires:**

- `ure` $\leftarrow$ several versions

`ure`, 7645 **requires:**

- `debconf` ≥ 1675 **or** `debconf-2.0`

`ure`, 7645 **conflicts with:**

- `cli-uno-bridge` < 16229

Package Upgradeability Problem

System	Manager
ubuntu	`apt`
fedora	`dnf`
python	`pip`
OCaml	`opam`

Package Upgradeability Problem

Package Upgradeability $\rightsquigarrow$ SAT

Package Upgradeability Problem

Package Upgradeability $\rightsquigarrow$ SAT

Minimize:

Package Upgradeability Problem

Package Upgradeability $\rightsquigarrow$ SAT

Minimize:

- Number of **removed** packages
- Number of not **up-to-date** packages
- ...

Package Upgradeability Problem

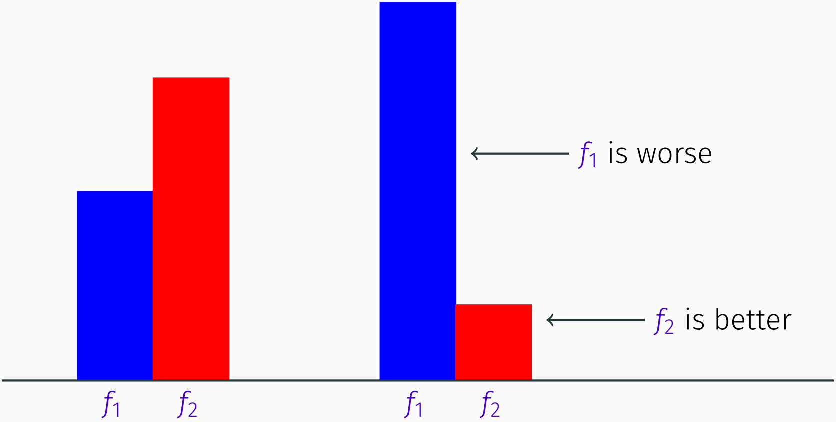
Package Upgradeability $\rightsquigarrow$ SAT

Minimize:

- Number of **removed** packages
- Number of not **up-to-date** packages
- ...

Multi-Objective MaxSAT

Hard clauses + **Multiple** sets of Soft clauses



What is the optimum?

Alice & Bob Pay a Fine

- $\mathcal{A}$: 10₪, $\mathcal{B}$: 50₪ is worse than $\mathcal{A}$: 0₪, $\mathcal{B}$: 0₪

Alice & Bob Pay a Fine

- $\mathcal{A}$: 10₪, $\mathcal{B}$: 50₪ is worse than $\mathcal{A}$: 0₪, $\mathcal{B}$: 0₪
- What about

  $\mathcal{A}$: 0₪, $\mathcal{B}$: 30 versus $\mathcal{A}$: 20₪, $\mathcal{B}$: 20₪?

Alice & Bob Pay a Fine

- $\mathcal{A}$: 10₪, $\mathcal{B}$: 50₪ is worse than $\mathcal{A}$: 0₪, $\mathcal{B}$: 0₪
- What about

  $\mathcal{A}$: 0₪, $\mathcal{B}$: 30 versus $\mathcal{A}$: 20₪, $\mathcal{B}$: 20₪?
- **Leximax**: "Minimize the higher fine."

Alice & Bob Pay a Fine

- $\mathcal{A}$: 10₪, $\mathcal{B}$: 50₪ is worse than $\mathcal{A}$: 0₪, $\mathcal{B}$: 0₪
- What about

  $\mathcal{A}$: 0₪, $\mathcal{B}$: 30 versus $\mathcal{A}$: 20₪, $\mathcal{B}$: 20₪?
- **Leximax**: "Minimize the higher fine."
- Prefer:

  $\mathcal{A}$: 20₪, $\mathcal{B}$: 20₪ to $\mathcal{A}$: 0₪, $\mathcal{B}$: 30₪

Alice & Bob Pay a Fine

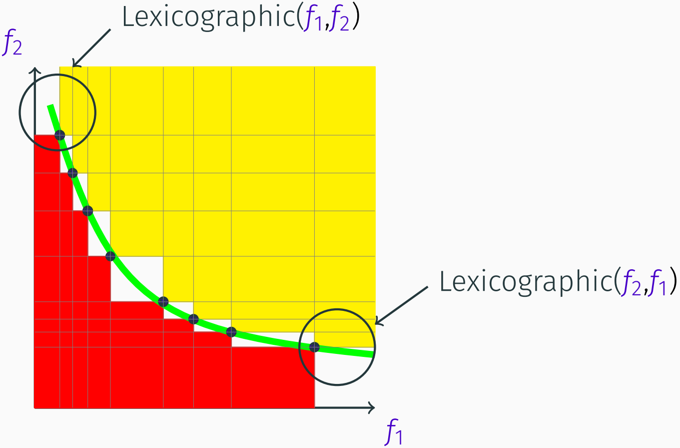
- $\mathcal{A}$: 10₪, $\mathcal{B}$: 50₪ is worse than $\mathcal{A}$: 0₪, $\mathcal{B}$: 0₪
- What about

  $\mathcal{A}$: 0₪, $\mathcal{B}$: 30 versus $\mathcal{A}$: 20₪, $\mathcal{B}$: 20₪?
- **Leximax**: "Minimize the higher fine."
- Prefer:

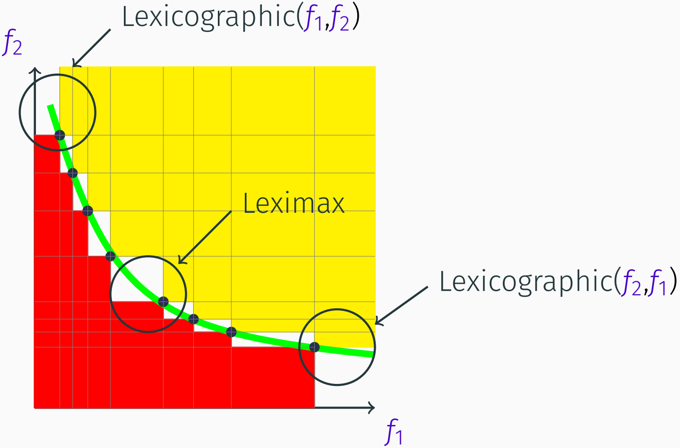
  $\mathcal{A}$: 20₪, $\mathcal{B}$: 20₪ to $\mathcal{A}$: 0₪, $\mathcal{B}$: 30₪
- **Generalize by**:

  Sort decreasingly and compare lexicographically

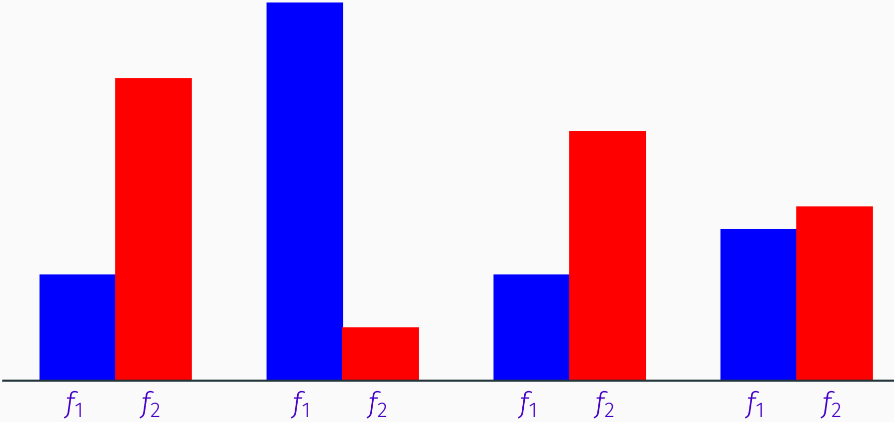
Lexicographic vs Leximax



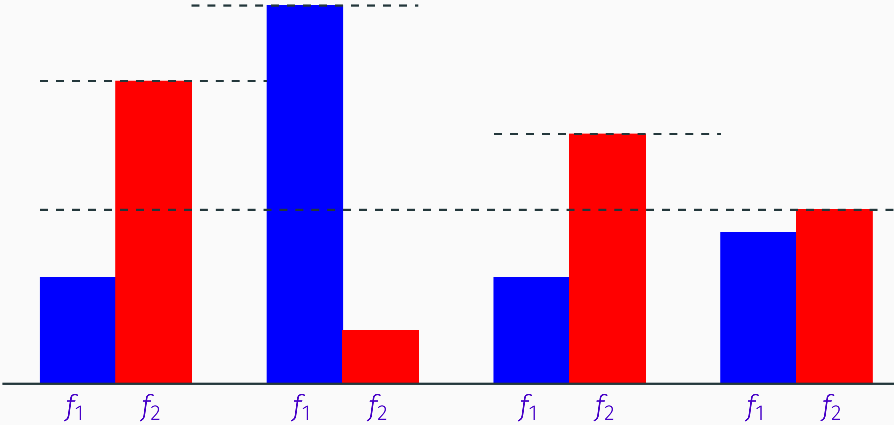
Lexicographic vs Leximax



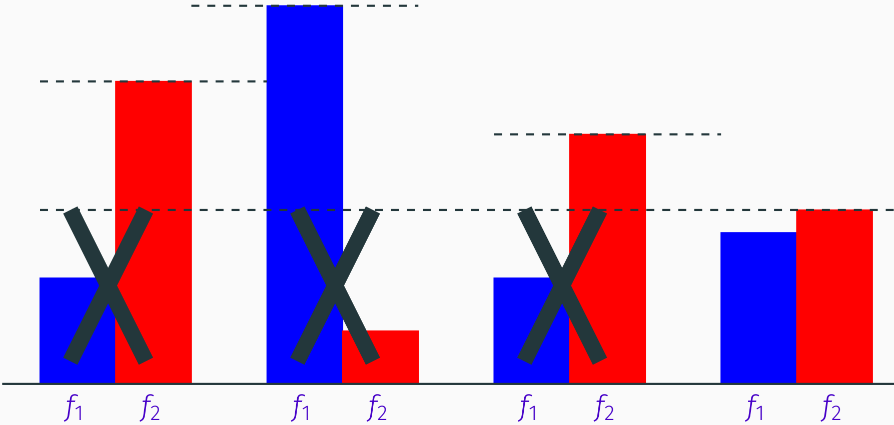
Leximax-optimum



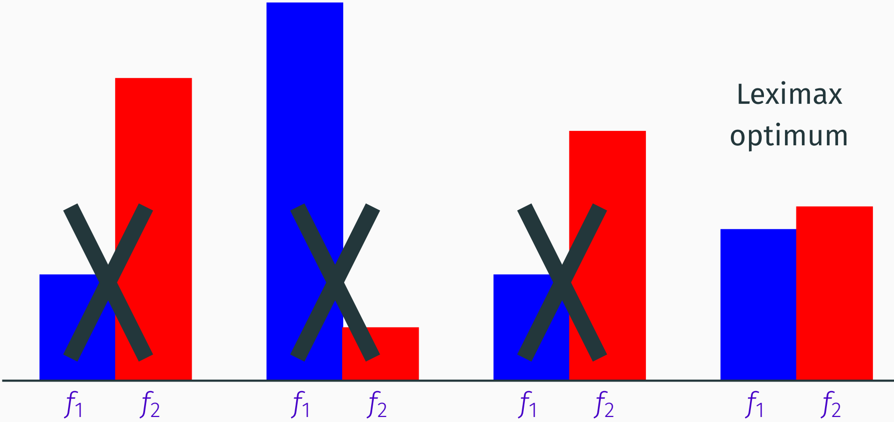
Leximax-optimum



Leximax-optimum



Leximax-optimum



SAT-based Leximax Optimisation

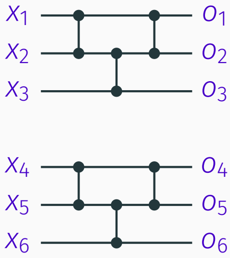
$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$

SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

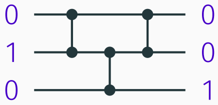
$f_2 : x_4 + x_5 + x_6$

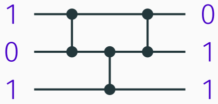


SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$

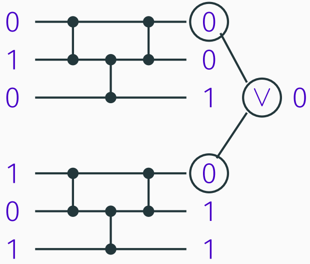




SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

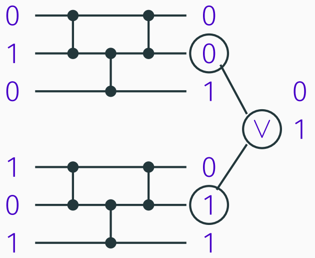
$f_2 : x_4 + x_5 + x_6$



SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

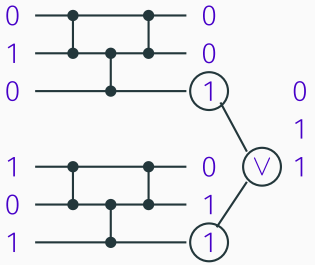
$f_2 : x_4 + x_5 + x_6$



SAT-based Leximax Optimisation

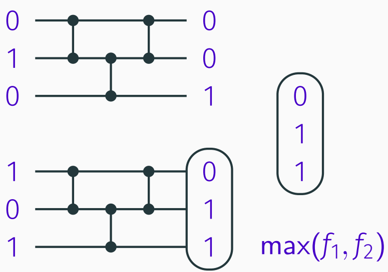
$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$



$f_1 : x_1 + x_2 + x_3$

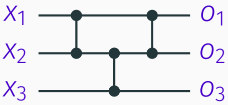
$f_2 : x_4 + x_5 + x_6$

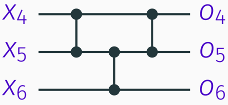


SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$



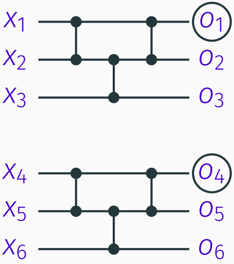


SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$

$o_1 \vee o_4 \Leftrightarrow y_1$



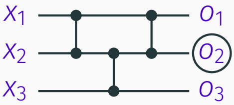
SAT-based Leximax Optimisation

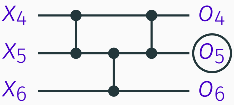
$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$

$o_1 \vee o_4 \Leftrightarrow y_1$

$o_2 \vee o_5 \Leftrightarrow y_2$

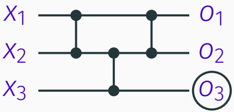


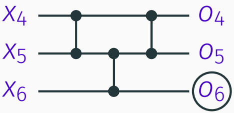


SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$





$o_1 \vee o_4 \Leftrightarrow y_1$

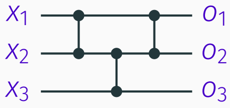
$o_2 \vee o_5 \Leftrightarrow y_2$

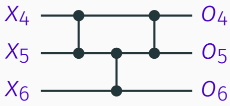
$o_3 \vee o_6 \Leftrightarrow y_3$

SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$





$o_1 \vee o_4 \Leftrightarrow y_1$

$o_2 \vee o_5 \Leftrightarrow y_2$

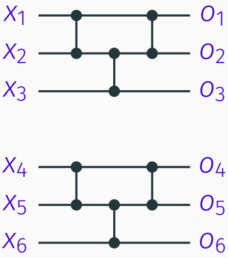
$o_3 \vee o_6 \Leftrightarrow y_3$

$y_1 + y_2 + y_3 = \max(f_1, f_2)$

SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$



$o_1 \vee o_4 \Leftrightarrow y_1$

$o_2 \vee o_5 \Leftrightarrow y_2$

$o_3 \vee o_6 \Leftrightarrow y_3$

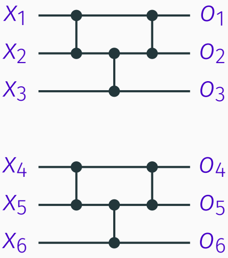
$y_1 + y_2 + y_3 = \max(f_1, f_2)$

Minimize $y_1 + y_2 + y_3$

SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$



$o_1 \vee o_4 \Leftrightarrow y_1$

$o_2 \vee o_5 \Leftrightarrow y_2$

$o_3 \vee o_6 \Leftrightarrow y_3$

$y_1 + y_2 + y_3 = \max(f_1, f_2)$

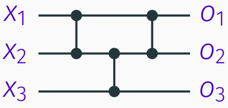
Minimize $y_1 + y_2 + y_3$

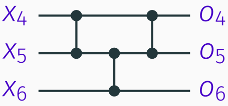
Optimum = k

SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$





$o_1 \vee o_4 \Leftrightarrow y_1$

$o_2 \vee o_5 \Leftrightarrow y_2$

$o_3 \vee o_6 \Leftrightarrow y_3$

$y_1 + y_2 + y_3 = \max(f_1, f_2)$

Minimize $y_1 + y_2 + y_3$

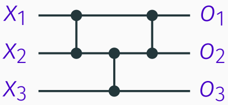
Optimum = k

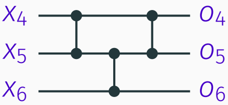
Fix $y_1 + y_2 + y_3 = k$

SAT-based Leximax Optimisation

$f_1 : x_1 + x_2 + x_3$

$f_2 : x_4 + x_5 + x_6$





$o_1 \vee o_4 \Leftrightarrow y_1$

$o_2 \vee o_5 \Leftrightarrow y_2$

$o_3 \vee o_6 \Leftrightarrow y_3$

$y_1 + y_2 + y_3 = \max(f_1, f_2)$

Minimize $y_1 + y_2 + y_3$

Optimum = k

Fix $y_1 + y_2 + y_3 = k$

Repeat with **second** maximum.

Core-guided Approach

- Major difficulty is the encoding of sorting networks

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- large CNF encoding

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- large CNF encoding
- difficult for SAT solvers

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- **large** CNF encoding
- **difficult** for SAT solvers
- **Core-guided:**

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- **large** CNF encoding
- **difficult** for SAT solvers
- **Core-guided:**
  - Initially all the variables set to minimum.

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- large CNF encoding
- difficult for SAT solvers
- Core-guided:
    - Initially all the variables set to minimum.
    - Permit non-optimality if it appears in a core.

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- **large** CNF encoding
- **difficult** for SAT solvers
- **Core-guided:**
    - Initially all the variables set to minimum.
    - Permit non-optimality if it appears in a core.
    - Include variables in a core into the sorting network.

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- **large** CNF encoding
- **difficult** for SAT solvers
- **Core-guided:**
  - Initially all the variables set to minimum.
  - Permit non-optimality if it appears in a core.
  - Include variables in a core into the sorting network.
    - **Augment** old sorting network with new variables

Core-guided Approach

- Major difficulty is the encoding of sorting networks
- large CNF encoding
- difficult for SAT solvers
- Core-guided:
  - Initially all the variables set to minimum.
  - Permit non-optimality if it appears in a core.
  - Include variables in a core into the sorting network.
    - Augment old sorting network with new variables
    - Or Rebuild the network.

Evaluation

- Instances:
  Package Upgradeability benchmarks
  from the Mancoosi project

Evaluation

- Instances:
  Package Upgradeability benchmarks
  from the Mancoosi project
- `mccs` (ILP-based algorithm)

Evaluation

- Instances:
  Package Upgradeability benchmarks
  from the Mancoosi project
- `mccs` (ILP-based algorithm) with
  `CPLEX`, `Gurobi`, `SCIP`, `Cbc`, `GLPK`, `lpsolve`

Evaluation

- Instances:
  Package Upgradeability benchmarks
  from the Mancoosi project
- `mccs` (ILP-based algorithm) with
  <u>`CPLEX`</u>, <u>`Gurobi`</u>, `SCIP`, `Cbc`, `GLPK`, `lpsolve`

Evaluation

- Instances:
  Package Upgradeability benchmarks
  from the Mancoosi project
- `mccs` (ILP-based algorithm) with
  `CPLEX`, `Gurobi`, <u>`SCIP`</u>, <u>`Cbc`</u>, <u>`GLPK`</u>, <u>`lpsolve`</u>

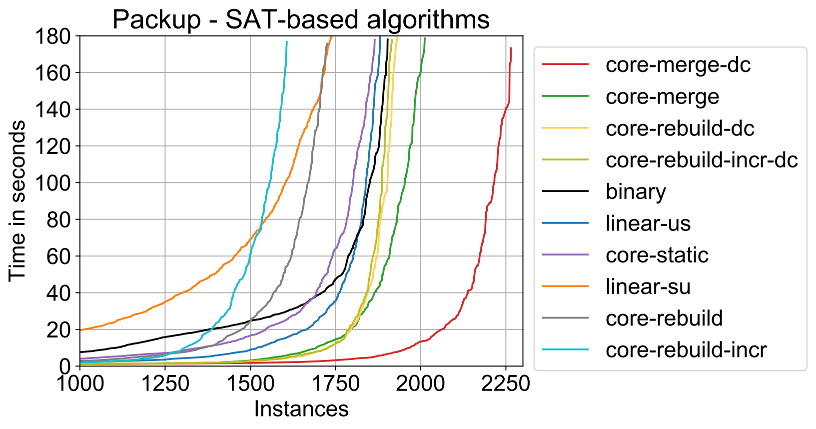
Evaluation

- Instances:
  Package Upgradeability benchmarks
  from the Mancoosi project
- `mccs` (ILP-based algorithm) with
  `CPLEX`, `Gurobi`, `SCIP`, `Cbc`, `GLPK`, `lpsolve`
- `packup` (SAT-based algorithms) with `CaDiCaL`

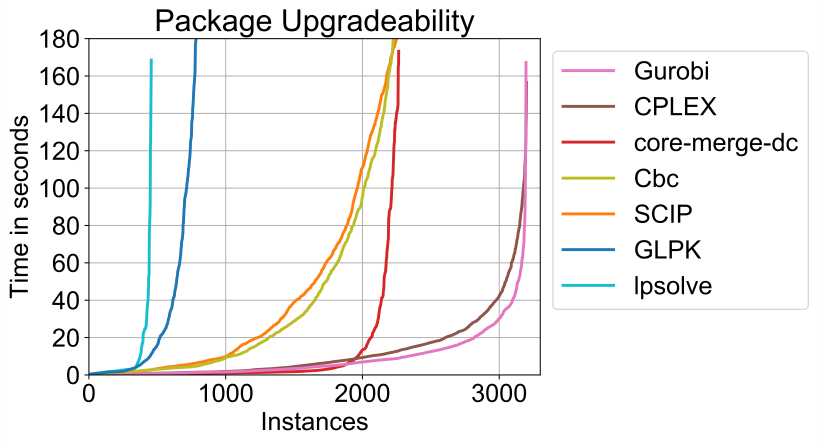
Evaluation

- Instances:
  Package Upgradeability benchmarks
  from the Mancoosi project
- `mccs` (ILP-based algorithm) with
  `CPLEX`, `Gurobi`, `SCIP`, `Cbc`, `GLPK`, `lpsolve`
- `packup` (SAT-based algorithms) with `CaDiCaL`
  - various non-core-guided search (binary, linear, etc.)
  - various core-guided search differing on how to augment the sorting network

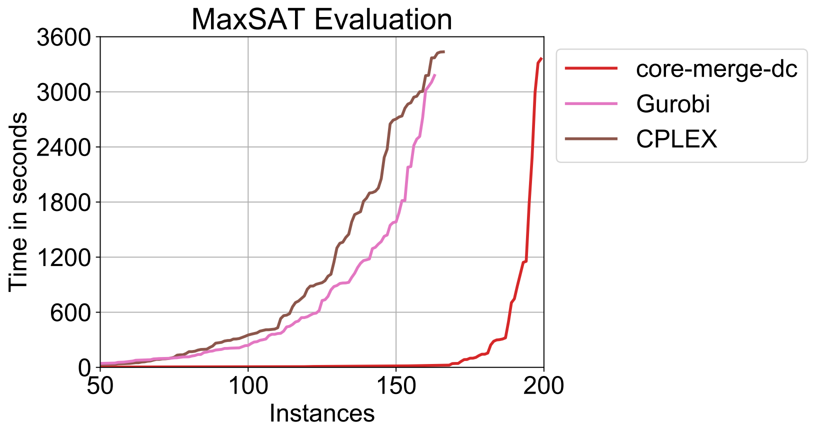
Comparison — SAT-based



Evaluation — Package Upgradeability



Evaluation — SAT Competition



Conclusions and Future Work

- Solving Multi-Objective Optimization for **Leximax**

Conclusions and Future Work

- Solving Multi-Objective Optimization for Leximax
- SAT-based solving

Conclusions and Future Work

- Solving Multi-Objective Optimization for **Leximax**
- **SAT**-based solving
- **Core**-based solving

Conclusions and Future Work

- Solving Multi-Objective Optimization for **Leximax**
- **SAT**-based solving
- **Core**-based solving
- Evaluation on Package Upgradeability

Conclusions and Future Work

- Solving Multi-Objective Optimization for **Leximax**
- **SAT**-based solving
- **Core**-based solving
- Evaluation on Package Upgradeability
- Core-guided the best out of the SAT-based