



Command Line Interface

Jimmy Lee & Peter Stuckey



香港中文大學
The Chinese University of Hong Kong

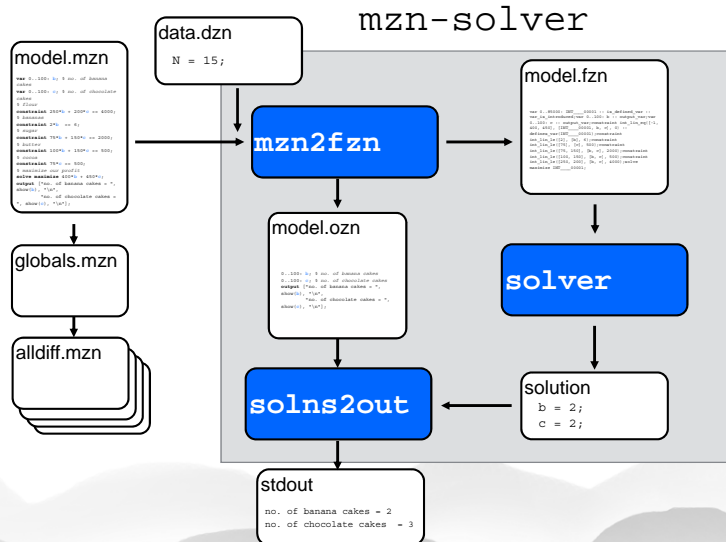


THE UNIVERSITY OF
MELBOURNE

Overview

- ⌘ The course is designed to be undertaken using the IDE, but
 - This is not preferred by some people
 - For solving real world problems we need to use MiniZinc in a tool chain
- ⌘ How do you use MiniZinc from the command line?

The MiniZinc Tool Chain



3

The bundled executables

- The MiniZincIDE comes bundled with a number of integrated solvers you can use from the command line
 - `mzn-gecode`: minizinc with Gecode (the default solver for the course, **recommended**)
 - `mzn-chuffed`: minizinc with chuffed (a learning CP solver)
 - `mzn-cbc`: minizinc with COIN OR CBC (a free MIP solver)
 - `mzn-gurobi`: minizinc with Gurobi (a commercial MIP solver - **but you need to install the DLL**)

4



The bundled executables

- The MiniZincIDE comes bundles with a number of integrated solvers you can use from the command line
 - `mzn-g12fd`: minizinc with G12 FD (a rather old CP solver)
 - `minizinc`: minizinc with G12 FD (same as above)
 - `mzn-g12lazy`: minizinc with lazyfd (an older learning CP solver)
 - `mzn-g12mip`: minizinc with a free older MIP solver

5

Command Line Usage

- Usage is given by

```
mzn-gecode [<options>] <model>.mzn [<data>.dzn ...]
```

- The executable expects
 - a single model file (.mzn)
 - any number of data files(.dzn)
 - possibly some options

6

Default Behaviour

⌘ By default

```
mzn-gecode model.mzn data.dzn
```

⌘ For a satisfaction problem: `solve satisfy`

- search for a solution
- stop and print the first solution found, or
- print `====UNSATISFIABLE=====`

⌘ For optimisation problem: `solve minimize` or `solve maximize`

- search for the optimal solution
- print the optimal solution, or
- print `====UNSATISFIABLE=====`
- beware may take a **long time** to find the optimal

7

All Solutions

⌘ An important option for MiniZinc is

- `-all-solutions`, or `-a`

```
mzn-gecode model.mzn data.dzn -a
```

⌘ For satisfaction problems

- print all solutions found, with `———` separators
- or, print `====UNSATISFIABLE=====`

⌘ For optimisation problems

- print all solutions found on the way to the optimal
 - only one optimal solution, it will be the last printed
- or, print `====UNSATISFIABLE=====`
- Note: here you can see progress to optimal

8

All Solutions

- ⌘ Note that only CP solvers tend to support the all solutions flag
- ⌘ MIP solvers such as
 - `mzn-cbc`
 - `mzn-gurobi`
 - `mzn-gl2mip`
 - only ever return one (optimal) solution

9

Statistics

- ⌘ An important option for MiniZinc is
 - `-statistics, or -s`
- `mzn-gecode model.mzn data.dzn -s`
- ⌘ The solver will print statistics about the solving process
- ⌘ The statistics printed depends on the solver
- ⌘ Examples
 - runtime, solvertime,
 - number of solutions found
 - size of problem: vars + constraints
 - search stats: nodes, failures, restarts

10



Data on the Command Line

- One can include data on the command line using the flag: `-D`
`mzn-gecode model.mzn data.dzn -D "<dzn content>"`
- The string after `-D` is treated like it were part of the model.
- This is useful for running models while varying one or two critical parameters

11

Changing the Output

- `-output-to-file, or -o <file>`
 - send output to file rather than stdout
- `-soln-sep <string>`
 - change the solution separator string (usually `"—"`)
 - useful for e.g. printing all solutions in CSV by setting this to the empty string
- `-search-complete-msg <string>`
 - change the string printed after all solutions (or the optimal) is found

12



Other Interesting Options

- ⌘ `-keep-files`, or `-k`
 - keep the temporary files constructed (.fzn)
 - useful for examining the flattening and debugging some error messages
- ⌘ `-parallel`, or `-p <n>`
 - run the solver using *n* threads during search
 - supported by Gecode and Gurobi
- ⌘ `-fzn-flags <flags>`
 - pass the `<flags>` to the solver executable, e.g.
`mzn-gecode model.mzn -fzn-flags "-time 1000"`
 - runs the gecode solver with 1000ms timeout

13

Overview

- ⌘ MiniZinc has a lot of command line options
- ⌘ We have covered the most important here

14



EOF

15