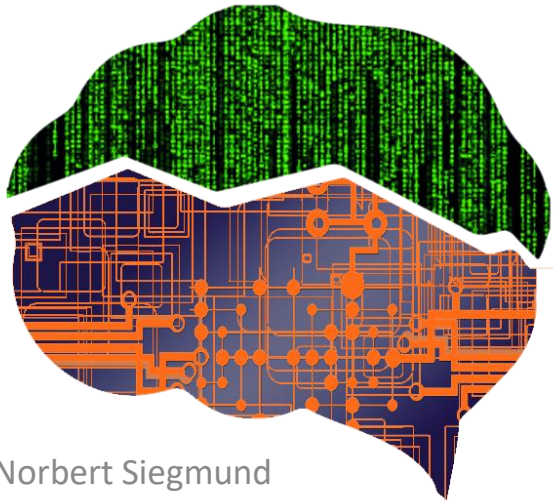


Machine Learning for Software Engineering

Project Description



Prof. Dr.-Ing. Norbert Siegmund
Intelligent Software Systems

**Bauhaus-Universität
Weimar**

Goal of the Project

- Apply meta-heuristics for single- and multi-objective optimization of a constrained optimization space
- Develop an algorithm to find an (or a set in the multi-objective scenarios) optimal configuration of a configurable software system
- Tasks:
 - Parse several files (feature model, files for fitness assessment)
 - Implement one or multiple algorithms that find (near) optimal solutions given the constraints in the feature model
 - Visually present the optimization process and output the solutions

Context: Configurable Systems

BMW Deutschland - Mozilla Firefox

File Edit View Go Bookmarks Tools Help

http://www.bmw.de/de/index_narrowband.html?content=../de/de/general/configurations_center/configurator.htm

Getting Started Latest Headlines

Home 1 3 5 6 7 X3 X5 Z4 M

Mein BMW Kontakt BMW Konfigurator BMW Händler

BMW Konfigurator

Zurück

BMW Konfigurator

Tipps zum Car Configurator

Modell	Grundausrüstung
	Getriebe Klima, Heizung, Kommunikation, Info
	↕ Ausstattung
Getriebe	
<input checked="" type="checkbox"/>	Automatic Getriebe
Klima, Heizung	
<input checked="" type="checkbox"/>	Standheizung mit
<input type="checkbox"/>	Sonnenschutzze
<input checked="" type="checkbox"/>	Klimaautomatik m
Komfort/Nutzen	
<input type="checkbox"/>	Ablagenpaket
<input type="checkbox"/>	Armauflage vorn
<input type="checkbox"/>	Außenheizung

Waiting for ecom.bmwgroup.com...

Beispiel BMW X3



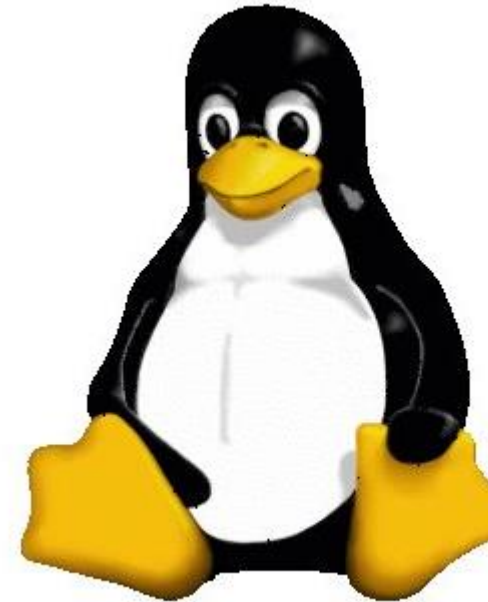
Dachhimmel:
90.000
Varianten-
möglichkeiten

Autotüren:
3.000
Varianten-
möglichkeiten

Hinterachsen:
324
Varianten-
möglichkeiten

Linux-Kernel

- ca. 6,000,000 LOC
- Highly configurable
 - > 10,000 configuration options! (x86, 64bit,
 - Nearly all source code is optional



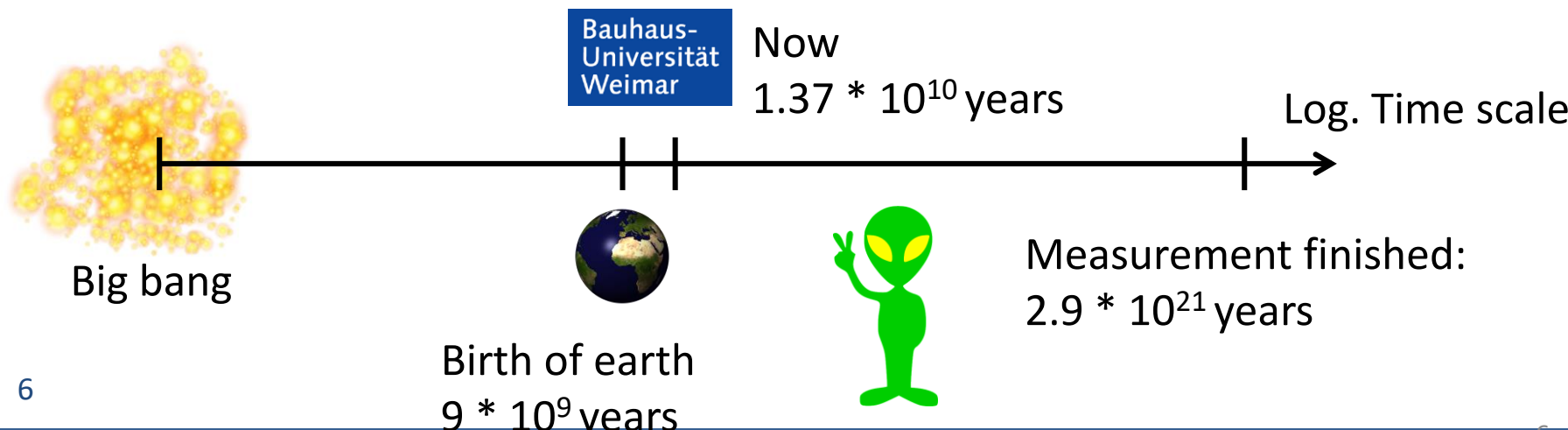
Printer firmware



Optimization of Non-Functional Properties

- ▶ SQLite is an embedded highly configurable database system
 - ▶ Deployed on over 500 million systems
 - ▶ Provides 88 compiler options for configuration
- ▶ Suppose we would measure each variant of SQLite:
 - ▶ 2^{88} variants and 5 minutes per measurement (compile + benchmark) = $2^{88} * 5\text{min} / 60 (\text{per h}) / 24 (\text{per d}) / 365 (\text{per y}) =$

2,944,111,585,058,457,655,296y

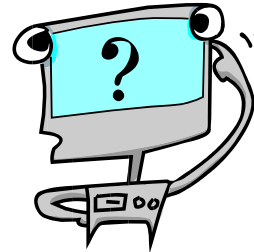


Feature Modelling

- Describe the features of a domain
- Used for visualization and communication
- A feature model describes...
 - The essential elements (functions, parameters, etc.) of a domain
 - All variation points and their constraints
 - The set of all valid variants of a configurable system

A Valid Feature Selection?

- Transaction management
- Log & Recovery
- Write access
- Persistency / In-Memory
- Page replacement strategies LRU / LFU / Clock / ...
- Sorting techniques
- Datentypes of variable length
- Grouping, aggregation, ...



```

karoma - Poderosa
File Edit Console Tools Window Plug-in Help
Line feed CR Encoding iso-8859-1 generic
1 karoma 2 karoma
.config - Linux Kernel v2.6.33.3 Configuration
===== Processor type and features =====
Arrow keys navigate the menu. <Enter> selects submenus --->. Highlighted letters
are hotkeys. Pressing <Y> includes, <N> excludes, <M> modularizes features.
Press <Esc><Esc> to exit, <?> for Help, </> for Search. Legend: [*] built-in [ ]
excluded <M> module < > module capable

[ ] Tickless System (Dynamic Ticks)
[ ] High Resolution Timer Support
[ ] Symmetric multi-processing support
[ ] Support for extended (non-PC) x86 platforms
[ ] Single-depth WCHAN output
[ ] Paravirtualized guest support --->
[ ] Memtest
  Processor family (Generic-x86-64) --->
  Preemption Model (No Forced Preemption (Server)) --->
[ ] Reroute for broken boot IRQs (NEW)
[ ] Machine Check / overheating reporting
[ ] Dell laptop support
[ ] /dev/cpu/microcode - microcode support
[ ] /dev/cpu/*/msr - Model-specific register support
[ ] /dev/cpu/*/cpuid - CPU information support
  Memory model (Sparse Memory) --->
[*] Sparse Memory virtual memmap (NEW)
[ ] Allow for memory hot-add (NEW)
[ ] Enable KSM for page merging
(4096) Low address space to protect from user allocation
[ ] Check for low memory corruption
[ ] Reserve low 64K of RAM on AMI/Phoenix BIOSen
-* MTRR (Memory Type Range Register) support
[ ] MTRR cleanup support
[ ] Enable seccomp to safely compute untrusted bytecode
[ ] Enable -fstack-protector buffer overflow detection (EXPERIMENTAL)
  Timer frequency (250 HZ) --->
[ ] kexec system call
v(+)

<Select> < Exit > < Help >

```

VEGETARIAN

WHICH WICH WOULD YOU LIKE?

- TRIPLE CHEESE MELT
- ELVIS WICH (P, Honey & Banana)
- TOMATO & AVOCADO
- BLACK BEAN PATTY
- HUMMUS & BELL PEPPERS

CHOOSE YOUR BREAD

- WHITE
- WHEAT

CHOOSE YOUR CHEESE (Optional)

- AMERICAN
- SWISS
- PROVOLONE
- CHEDDAR
- PEPPER JACK
- MOZZARELLA

How Would You Like Your WICH Worked?

MUSTARDS

- Yellow
- Dijon
- Honey
- Deli

MAYOS

- Regular
- Lite
- Horseradish
- Spicy

SPREADS & SAUCES

- BBQ
- Buffalo
- Marinara
- 1000 Island
- Ranch

ONIONS

- Red
- Grilled
- Crispy Strings

VEGGIES

- Lettuce
- Tomato
- Pickles
- Jalapenos
- Olive Salad
- Mushrooms
- Sauerkraut
- Coleslaw
- Bell Peppers

OILS & SPICES

- Oil
- Vinegar
- Salt
- Pepper
- Oregano
- Parmesan

EXTRAS (.75¢ Each)

- Bacon
- Avocado
- Pickle (Whole)
- More Meat
- More Cheese

Feature Model as Propositional Formula

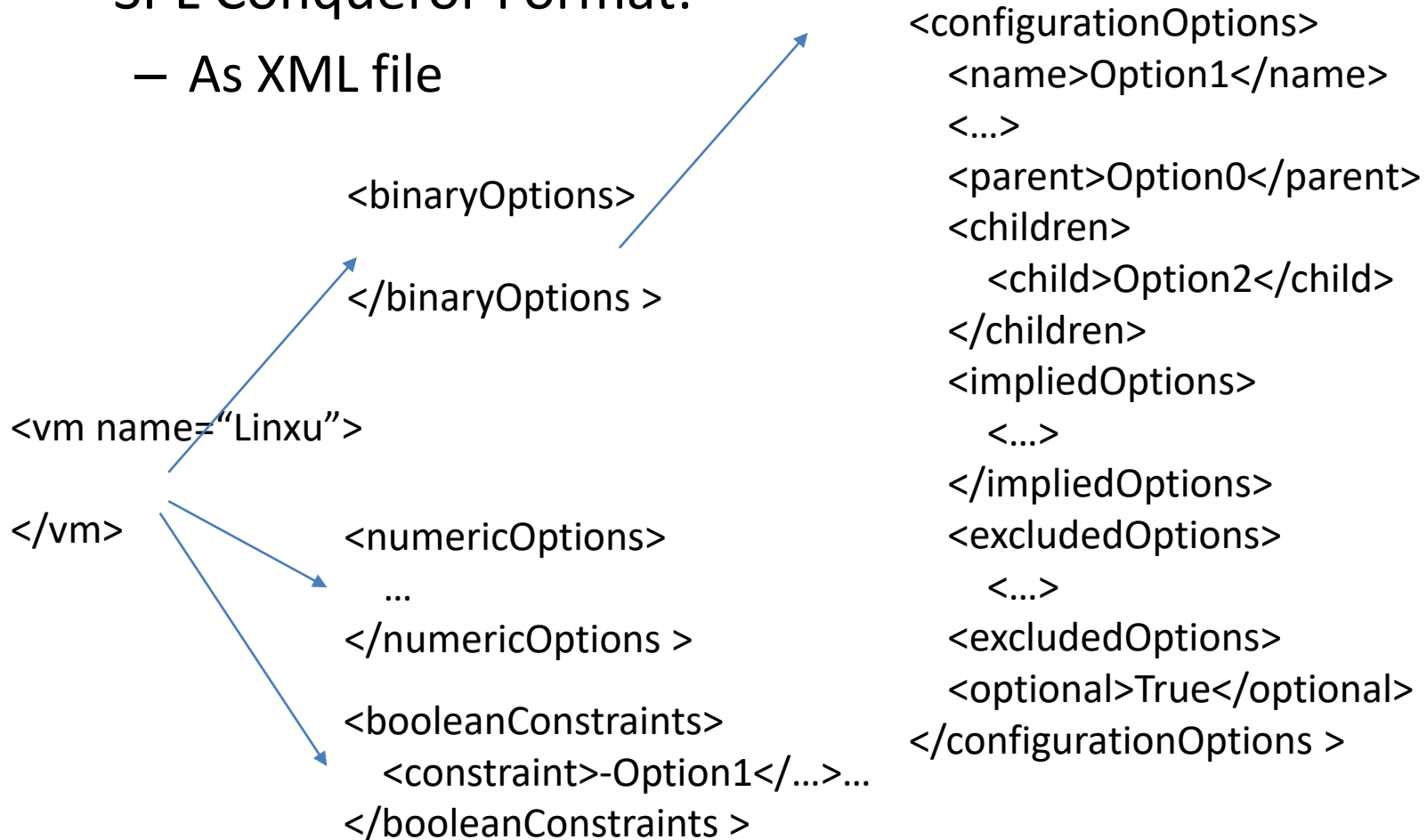
- Variable per feature (true if selected, false otherwise)
- Formula describes the feature model
- Formula returns true for a valid configuration

$$\begin{aligned} & \textit{Basis} \wedge (\textit{Unix} \vee \textit{Win}) \wedge \neg(\textit{Unix} \wedge \textit{Win}) \\ & \wedge (\textit{Txn} \Rightarrow \textit{Write}) \end{aligned}$$

- Enables automated check of configurations and listing of all valid configurations (e.g. using SAT, BDD, CSP, ...)

Different Formats of Feature Models

- SPL Conqueror Format:
 - As XML file



DIMACS Format

- A file can start with comments denoted by the character `c` at line start
- After the comments, each file contains the line `p cnf nbvar nbclauses`
 - Indicates the file is in CNF format; `nbvar` is the number of variables appearing in the file; `nbclauses` is the number of clauses in the file
- All clauses follow below
 - A clause is a sequence of distinct non-null numbers between `[-nbvar,nbvar]` with `0` as line ending symbol
 - A clause cannot contain the opposite literals `i` and `-i` simultaneously.
 - Positive numbers denote the corresponding variables
 - Negative numbers denote the negations of the corresponding variables.

`c`

`c start with comments`

`p cnf 5 3`

`1 -5 4 0`

`-1 5 3 4 0`

`-3 -4 0`

Fitness Assessment

- Build a fitness assessment function using the given files
 - System_feature.txt contains performance values for **features**
 - System_interactions.txt contains performance values for **combinations** of features



bdbc_feature.txt

```
root: 4.29699011265453
HAVE_CRYPT0: 0.0120768614910988
HAVE_HASH: 4.62346632789263
HAVE_REPLICATION: 0.274902720516604
```



bdbc_interactions.txt

```
PS16K#CS512MB: -4.05239323580753
PS1K#HAVE_REPLICATION: -0.732537201572133
HAVE_CRYPT0#PS8K: -0.833090135903526
HAVE_REPLICATION#PS16K#DIAGNOSTIC: 31.7326205862695
```

→ $4.29 + \text{HAVE_CRYPTO} * 0.01 + \text{HAVE_HASH} * 4.63 + \text{HAVE_REPLICATION} * 0.27 + \text{PS16K} * \text{CS512MB} * (-4.05) + \text{PS1K} * \text{HAVE_REPLICATION} * (-0.73) + \dots$

Multi-Objective Case

- You will get multiple `_feature.txt/_interaction.txt` files
- Each file represents a different property we might want to simultaneously optimize
- How to build the objective function?
 - See lecture on this topic!
- How to visualize this or what is the output?
 - Pareto fronts obviously...

Evaluation of Your Work

- I will make the following calls:
- Optimize model.xml model_feature.txt model_interactions.txt
- Optimize model.xml model_feature.txt model_interactions.txt
model_feature2.txt model_feature3.txt model_interactions2.txt
model_interactions3.txt
- The numbers at the end of the file name denote which pairs of feature values and interaction value belong together
- Submit either a python file or a jupyter notebook

Grading

- Functionality: 10% of final exam grade
 - Finds the optimal configuration in feasible amount of time (below brute force) for single objective case of easy cases
- Non-functional aspects: 10% of final exam grade
 - Performance
 - Bug freeness
 - Quality of solution
 - Visualization
- You can use off-the-shelf CSP solvers but you will get only the 10% of functionality when using also optimization capabilities (you always can use the constraint-checking functionality)

Related Work and Tools:

- Validity checking libraries (examples here):
 - `satisfy` (<https://pypi.python.org/pypi/satisfy>)
 - `Pycosat` (<https://pypi.python.org/pypi/picosat>)
 - Do it yourself (<http://sahandsaba.com/understanding-sat-by-implementing-a-simple-sat-solver-in-python.html>)
- Related approaches:
 - <http://www.dre.vanderbilt.edu/~jules/jianmei.pdf>
 - <https://pdfs.semanticscholar.org/3384/176ef4196797603ae2ca68ff353bb4233668.pdf>
 - <https://pdfs.semanticscholar.org/b156/63eb2e0a7f92dbb5364c5a8e0742105f4ebe.pdf>
 - https://www.researchgate.net/profile/Mike_Papadakis/publication/281584480_Combining_Multi-Objective_Search_and_Constraint_Solving_for_Configuring_Large_Software_Product_Lines/links/577e7f0308aeaa6988b0cb1d.pdf