Consistency Maintenance: Propagation
Consistency Maintenance: Propagation

Conflict Resolution
Challenges

- Activating inactive features
- Fix incompleteness
Use_Pre_Allocation -> (Pre_Allocation_Size <= Buffer_Size * 1024 / Object_Size)
Use_Pre_Allocation -> (Pre_Allocation_Size <= Buffer_Size * 1024 / Object_Size)

By how much?

Fixes:
- **Decrease** Pre_Allocation_Size
- **Increase** Buffer_Size
- **Decrease** Object_Size
- **Disable** Use_Pre_Allocation
Typed variables
- Pre_Allocation_Size: Int
- Buffer_Size: Int
- Object_Size: Int
- Use_Pre_Allocation: Bool

Assigned values
- Pre_Allocation_Size = 10
- Buffer_Size = 4
- Object_Size = 512
- Use_Pre_Allocation = true

Logic constraint
- Use_Pre_Allocation -> (...)

Complete fix list
- Pre_Allocation_Size <= 8
- Buffer_Size >= 5
- Object_Size <= 409,6
- Use_Pre_Allocation = false
Constraint violation

\[ V : \{ m: \text{Bool}, a: \text{Int}, b: \text{Int} \} \rightarrow \text{Typed variables} \]

\[ e : \{ m=\text{true}, a=6, b=5 \} \rightarrow \text{Configuration (assignment)} \]

\[ c : (m \rightarrow a>10) \land (\neg m \rightarrow b>10) \land (a < b) \rightarrow \text{Violated constraint} \]

Range Fixe

Assignment

Unit

Range

Unit

[\[ m := \text{false}, b : b > 10 \]\]

[\[ (a,b) : a > 10 \land a < b \]\]

Range Fix

Range Fix
Constraint violation

\[ V : \{ m: \text{Bool}, a: \text{Int}, b: \text{Int} \} \]

\[ e : \{ m=\text{true}, a=6, b=5 \} \]

\[ c : (m \to a>10) \land (\neg m \to b>10) \land (a < b) \]

Typed variables

Configuration (assignment)

All the configuration that can be produced satisfy the constraint

Minimality of variables

A fix should change a minimal set of variables

Range Fixe

Maximality of ranges

The range of a fix should be maximal
1. Find minimal diagnoses

HS-DAG algorithm

Hard constraint (c)

\((m \rightarrow a > 10) \land (\neg m \rightarrow b > 10) \land (a < b)\)

Soft constraints (e)

[1] \(m = \text{true}\)
[2] \(a = 6\)
[3] \(b = 5\)
Fix generation algorithm

1. Find minimal diagnoses  →  HS-DAG algorithm

Hard constraint (c)

\[(m \rightarrow a > 10) \land (\neg m \rightarrow b > 10) \land (a < b)\]

Soft constraints (e)

[1] m = true
[2] a = 6
[3] b = 5
Fix generation algorithm

HS-DAG algorithm

Hard constraint (c)
\[(m \rightarrow a > 10) \land (\neg m \rightarrow b > 10) \land (a < b)\]

Soft constraints (e)
[1] \(m = \text{true}\)
[2] \(a = 6\)
[3] \(b = 5\)
Fix generation algorithm

1. Find minimal diagnoses → HS-DAG algorithm

2. Replace unchanged variables

\{m = true, b=5\}

\((m \rightarrow 6 > 10) \land (\neg m \rightarrow b > 10) \land (6 < b)\)
Fix generation algorithm

1. Find minimal diagnoses  {m = true, b=5}
2. Replace unchanged variables  (m→6>10) ∧ (¬m→b>10) ∧ (6 < b)
3. Generate the fixes  
   (m→6>10) ∧ (¬m→b>10) ∧ (6 < b)
   (¬m ∨ 6>10) ∧ (m ∨ b>10) ∧ (6 < b)
   (¬m) ∧ (b>10) ∧ (6 < b)
   (¬m) ∧ (b>10 ∧ 6 < b)
Multi-Constraint violation

\[ V : \{m:\text{Bool}, n:\text{Bool}, x:\text{Bool}, y:\text{Bool}, z:\text{Bool} \} \]

\[ e : \{m=\text{true}, n=\text{false}, x=\text{false}, y=\text{false}, z=\text{false} \} \]

\[ c : (m \land n) \]

\[ S : \{n \rightarrow (x \lor y), x \rightarrow z \} \quad \text{Satisfied constraints} \]

Possible fix \[ [n := \text{true}] \]

Violates \[ n \rightarrow (x \lor y) \]

Ignore

Eliminate fix

Propagate
Fix generation algorithm with propagation

\[ \text{S} \]

\[ c \]

\[ P : \{ m \land n \} \]

\[ P : \{ m \land n, n \rightarrow (x \lor y) \} \]

\[ P : \{ m \land n, n \rightarrow (x \lor y), x \rightarrow z \} \]

\[ c : (m \land n) \land (n \rightarrow (x \lor y)) \land (x \rightarrow z) \]

\[ [n = \text{true}, x = \text{true}, z = \text{true}] \]
\[ [n = \text{true}, y = \text{true}] \]

Shares variables with \( \{ n \rightarrow (x \lor y), x \rightarrow z \} \)?

Shares variables with \( \{ x \rightarrow z \} \)?

Replace \( c \) by \( P \)

Apply Fix generation algorithm
Number of variables per fix

Number of fixes

0
20
40
60
80

123456789

75%

Number of fixes

Number of violations

0
10
20
30
40
50

01234589

95%

Generation Time (ms)

Density of violations

0.000
0.005
0.010
0.015
0.020
0.025

50 100 150 200

73% coverage of error resolution by configtool

100% coverage of error resolution by our tool
Number of variables

Number of violations

Elimination
20ms
17 ignored

Ignorance
17ms
0 ignored
32% new errors

Propagation
50ms
1 ignored
USB Network Adapters

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 < >

< > USB CATC NetMate-based Ethernet device support (EXPERIMENTAL)
< > USB KLSI KL5USB101-based ethernet device support
< > USB Pegasus/Pegasus-II based ethernet device support
< > USB RTL8150 based ethernet device support (EXPERIMENTAL)
< > Multi-purpose USB Networking Framework
< > ASIX AX88xxx Based USB 2.0 Ethernet Adapters
-< > CDC Ethernet support (smart devices such as cable modems)
< > CDC EEM support
< > CDC NCM support (NEW)
< > Davicom DM9601 based USB 1.1 10/100 ethernet devices (NEW)
< > SMSC LAN75XX based USB 2.0 gigabit ethernet devices (NEW)
< > SMSC LAN95XX based USB 2.0 10/100 ethernet devices (NEW)
< > GeneSys GL620USB-A based cables (NEW)
< > NetChip 1080 based cables (Laplink, ...)
< > Prolific PL-2301/2302 based cables (NEW)
< > MosChip MCS7830 based Ethernet adapters (NEW)
< > Host for KNDIS and ActiveSync devices (EXPERIMENTAL)
< > Simple USB Network Links (CDC Ethernet subset)
< > Sharp Zaurus (stock ROMs) and compatible
< > Conexant CX82310 USB ethernet port (NEW)
< > Intellon PLC based usb adapter (NEW)
< > Apple iPhone USB Ethernet driver
< > USB-to-WWAN Driver for Sierra Wireless modems (NEW)
< > LG VL600 modem dongle (NEW)
Jack Purcell CP Canvas

$70.00

Choose a color:
- White
- Black
- Blue
- Green

MY CONFIGURATION

My preferences
- Fuel type: Electricity

My version
- Expression:

My options

Expression:
- From
- £22,495.00

CO2, benefit in kind and mpg figures may vary according to optional equipment.
Unify
Software & Product Configuration

A. Hubaux
K. Czarnecki
C. Drescher
V. Ganesh
P. Heymans
L. Hvam
D. Jannach
T. Mannisto
L. Murta
T. Nguyen
M. Zanker
Knowledge Modelling
Automated reasoning
Complexity
Life cycle coverage
Knowledge evolution
?
Knowledge Modelling
Automated reasoning
Complexity
Life cycle coverage
Knowledge evolution

Dagstuhl Seminar submitted