Automatic Verification of Systems September 3, 2018

Symbolic Model checking

Lecturer: Dr. Sharon Shoham *Names: Omri Lifshitz, Idan Berkovits*

1 Abstract

2 Introduction

3 CTL and LTL Model Checking

4 Symbolic Model Checking - BDDs

5 Algorithm

5.1 Finding Maximal SCC

As taught in class, in order to assert that there is a fair path in the product structure, we need to find a strongly connected component (SCC) that is reachable from the initial state and that intersects with each one of the fairness constraints.

In this section we will present the algorithm use to find the SCCs in a given model using BDDs. Our algorithm is based the algorithm explained in [1].

Algorithm 1 SCC Decomposition

procedure SCC_Decomp(N,V)

$V' \leftarrow V$

while $V' \neq 0$ **do**

$v \leftarrow random\_take(V')$

$B(v) \leftarrow backward_set(v, V', N)$

$SCC\_DECOMP\_RECUR(v, B(v), N)$

$V' \leftarrow V' \wedge \overline{v \vee B(v)}$

Algorithm 2 Finite maximum distance predecessors

procedure FMD_PRED(W, U, N)

$pred \leftarrow 0$

$front \leftarrow W$

$bound \leftarrow U$

while $front \neq 0$ **do**

$x \leftarrow \exists_Y front(Y) \wedge N(X,Y) \wedge bound(X)$

$y \leftarrow \exists_Y bount(Y) \wedge N(X,Y) \wedge bound(X)$

$front \leftarrow x \wedge \overline{y}$

$pred \leftarrow pred \vee front$

$bount \leftarrow bounts \wedge \overline{front}$

return *pred*

Algorithm 3 Recursive method to find SCCs

procedure SCC_DECOMP_RECUR(v, B(v), N)

$F(v) \leftarrow forward\_set(v, B(v), N)$

if $F(v) \neq 0$ **then**

report $F(v)$ an SCC

else

report v non- SCC

$x \leftarrow F(v) \vee v$

$R \leftarrow B(v) \wedge \overline{x}$

$y \leftarrow$ FMD_PRED(x, R, N)

report y non-SCC

$R \leftarrow R \wedge \overline{y}$

$IP \leftarrow \exists_Y (y \vee x)(Y) \wedge N(X,Y) \wedge R(X)$

while $R \neq 0$ **do**

$v \leftarrow random\_take(IP)$

$B(v) \leftarrow backward_set(v, R, N)$

$SCC\_DECOMP\_RECURE(v, B(v), N)$

$R \leftarrow R \wedge \overline{v \vee B(v)}$

$IP \leftarrow IP \wedge \overline{v \vee B(v)}$

References

[1] Aiguo. Xie and P A. Beerel. Implicit enumeration of strongly connected components. pages 37–40, Nov 1999.