Theory of Hybrid Systems
Informatik 2

RWTH AACHEN UNIVERSITY

gereon.kremer@cs.rwth-aachen.de
https://ths.rwth-aachen.de/teaching/

Satisfiability Checking - WS 2016/2017

Series 6

Exercise 1

You are given the following code and are asked if the functions `twice` and `twice_flat` are equivalent. Assume that `foo` is some function, model it as an uninterpreted function.

```
int foo(int x) { ... }
int twice(int n) {
        int out = n;
        for (int i = 0; i < 2; i++) {
                out = foo(out);
        }
        return out;
}
int twice_flat(int n) {
        return foo(foo(n));
}
```

1. Create a formula φ_1 modeling `twice`.
2. Create a formula φ_2 modeling `twice_flat`.
3. Create a formula φ_3 stating that the two functions are equivalent.
4. Apply Ackermann's reduction to φ_3.

Exercise 2

Let $a_{[l]}, b_{[l]}, c_{[l]}$ be bit vectors of size l in unsigned encoding.

- Give a propositional formula φ' that encodes the following finite-precision bit-vector arithmetic formula for $l = 3$:

$$\varphi : c_{[l]} = a_{[l]} \oplus b_{[l]} \wedge d_{[l]} = a_{[l]} +_U b_{[l]} \wedge e_{[l]} = a_{[l]} \cdot_U b_{[l]}$$

- Give the number of variables and clauses needed to express φ' in CNF.
- Give the space complexity (i.e. the growth of the number of variables and clauses for $l \to \infty$) of the encoding for $\oplus$, $+$ and $\cdot$ respectively in $\mathcal{O}$-notation.