Verification mini-project

Crossing the River

Jacob Karsten Wortmann
Sam Sepstrup Olesen
Nicklas Andersen
sw805f14

May 14, 2014

Introduction

The purpose of the exercise is to model a game, Crossing the River, in UPPAAL. The goal of the game is to get a group of people from one riverside to the other. The group consists of two daughters, two sons, a mom, a dad, a police officer, and a thief. The game has a list of rules that must be followed at all times to complete the game:

- Max 2 persons on the boat,
- Mom not alone with boys,
- Dad not alone with girls,
- Thief not alone with family,
- Only police officer, dad and mom can handle the boat.

Boat

Figure 1 shows the automata of the boat. Since the rules of the game defines that an adult needs to be on the boat for it to sail. To conform to this rule our automata require an adult to embark to boat before a second person is allowed to embark. When people on the boat disembark, we check whether this new state conforms to the rules. If it is an allowed state we update the location of the boat. If it is not a valid state the automata goes to a "gameover" location, which will cause a deadlock.

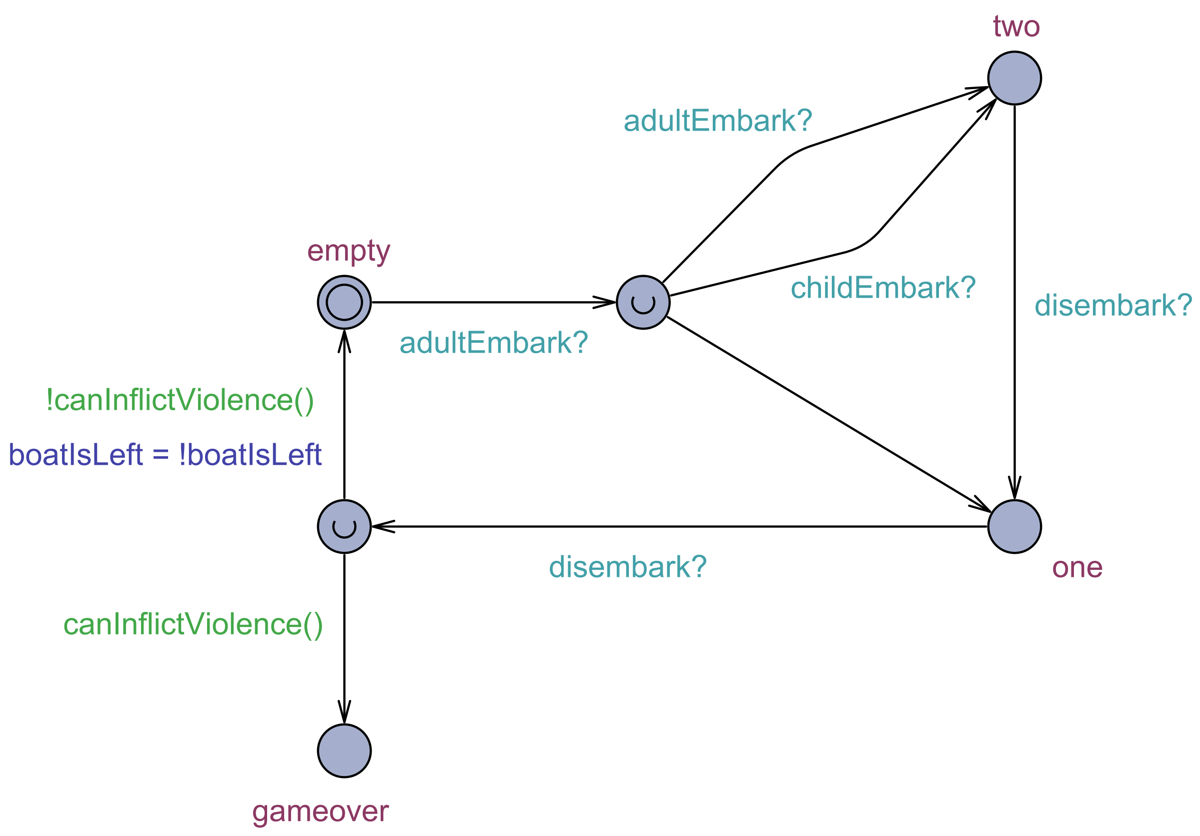

Figure 1: Boat automata.

```
clock time;

chan adultEmbark, childEmbark, disembark;

bool boatIsLeft = true;

bool policeIsLeft = true;
bool thiefIsLeft  = true;
bool dadIsLeft    = true;
bool momIsLeft    = true;
bool boy1IsLeft   = true;
bool boy2IsLeft   = true;
bool girl1IsLeft  = true;
bool girl2IsLeft  = true;


bool canInflictViolence() {
    if (((momIsLeft == boy1IsLeft)
      || (momIsLeft == boy2IsLeft))
      && (momIsLeft != dadIsLeft))
        return true;

    if (((dadIsLeft == girl1IsLeft)
      || (dadIsLeft == girl2IsLeft))
      && (dadIsLeft != momIsLeft))
        return true;

    if (((thiefIsLeft == boy1IsLeft)
      || (thiefIsLeft == boy2IsLeft)
      || (thiefIsLeft == girl1IsLeft)
      || (thiefIsLeft == girl2IsLeft)
      || (thiefIsLeft == dadIsLeft)
      || (thiefIsLeft == momIsLeft))
      && (thiefIsLeft != policeIsLeft))
        return true;

    return false;
}
```

Listing 1: Global declaration.

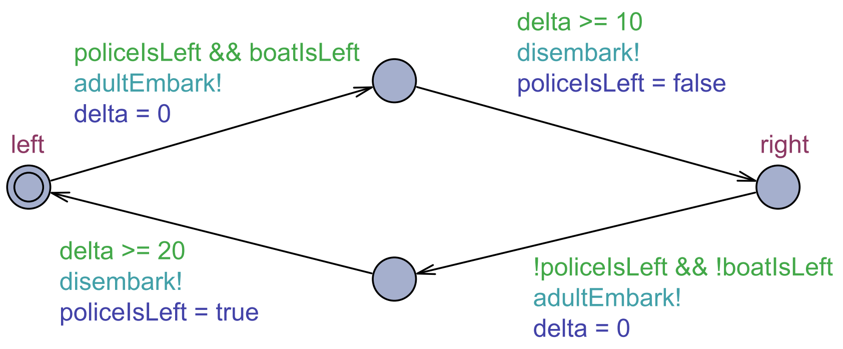

Figure 2: Police automata.

```
clock delta;

bool isLeft() {
    if (isDad)
        return dadIsLeft;
    else
        return momIsLeft;
}

void setLeft(bool left) {
    if (isDad)
        dadIsLeft = left;
```

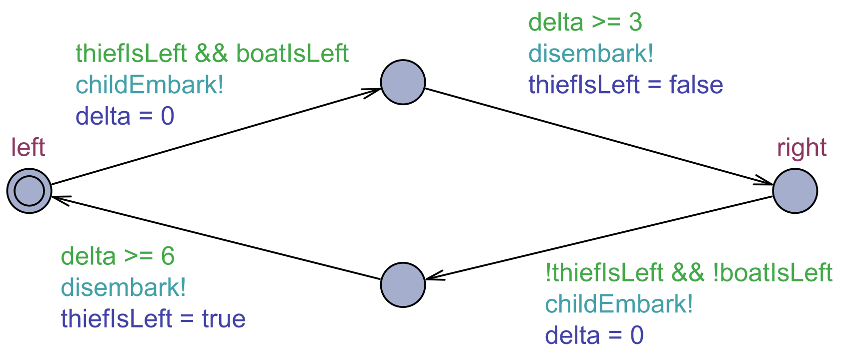

Figure 3: Thief automata.

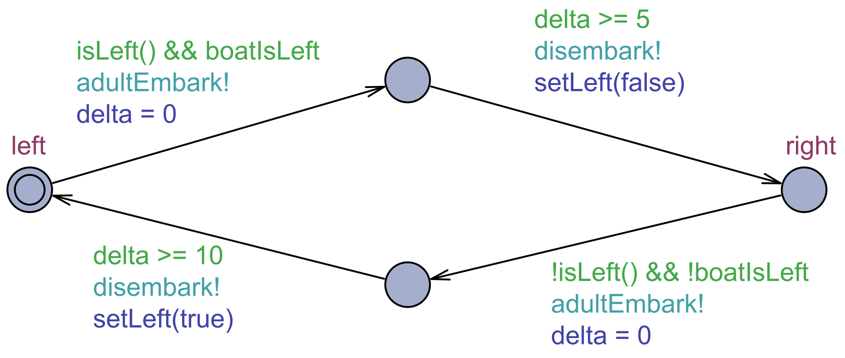

Figure 4: Parent automata.

```
    else
        momIsLeft = left;
}
```

Listing 2: Parent declaration.

```
clock delta;

bool isLeft() {
    if (isBoy)
        if (isFirst)
            return boy1IsLeft;
        else
            return boy2IsLeft;
    else
        if (isFirst)
            return girl1IsLeft;
        else
            return girl2IsLeft;
}

```

```
void setLeft(bool left) {
    if (isBoy)
        if (isFirst)
            boy1IsLeft = left;
        else
            boy2IsLeft = left;
    else
        if (isFirst)
            girl1IsLeft = left;
        else
            girl2IsLeft = left;
}
```

Listing 3: Child declaration.

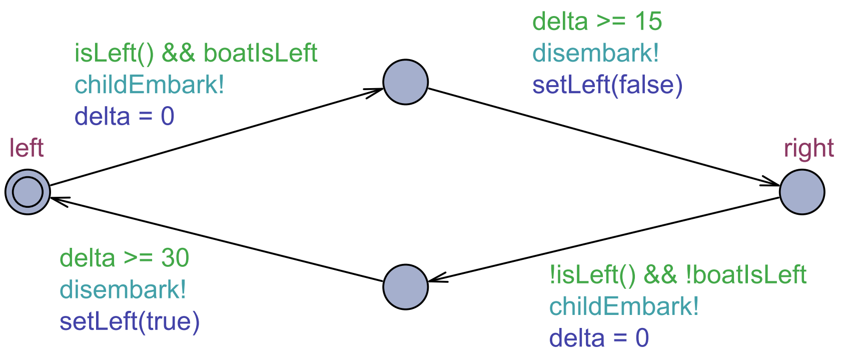

Figure 5: Child automata.

```
clock time;

chan adultEmbark, childEmbark, disembark;

bool boatIsLeft = true;

bool policeIsLeft = true;
bool thiefIsLeft  = true;
bool dadIsLeft    = true;
bool momIsLeft    = true;
bool boy1IsLeft   = true;
bool boy2IsLeft   = true;
bool boy3IsLeft   = true;
bool girl1IsLeft  = true;
bool girl2IsLeft  = true;


bool canInflictViolence() {
    if (((momIsLeft == boy1IsLeft) || (momIsLeft == boy2IsLeft) || (↩
        momIsLeft == boy3IsLeft)) && (momIsLeft != dadIsLeft))
        return true;

    if (((dadIsLeft == girl1IsLeft) || (dadIsLeft == girl2IsLeft)) && (↩
        dadIsLeft != momIsLeft))
        return true;

    if (((thiefIsLeft == boy1IsLeft) || (thiefIsLeft == boy2IsLeft) || (↩
        thiefIsLeft == boy3IsLeft) ||
         (thiefIsLeft == girl1IsLeft) || (thiefIsLeft == girl2IsLeft) ||
         (thiefIsLeft == dadIsLeft) || (thiefIsLeft == momIsLeft)) &&
         (thiefIsLeft != policeIsLeft))
        return true;

    return false;
}
```

Listing 4: Global declarations with an extra boy.