Safety Properties of Session Guarantees

Yifei Sun

January 24, 2024

Base

$\{P\}C\{Q\}$

$\{P\}$: initial state (an empty history + set of assertions)

C: program

$\{Q\}$: eventual convergence (or C might never terminate)

Premitives

```
OperationType = z3.EnumSort("OperationType", ["rd", "wr"])
Operation.declare("cons",
  ("proc", z3.IntSort()),    # process id
  ("type", OperationType),   # operation type
  ("obj", z3.IntSort()),     # invoking object
  ("ival", z3.StringSort()), # input value
  ("oval", z3.StringSort()), # output value
  ("stime", z3.IntSort()),   # start time
  ("rtime", z3.IntSort())    # return time
)
```

Premitives

$$\text{rb} \triangleq \{(a,b) : a,b \in H \wedge \text{a.rtime} < \text{b.stime}\}\ [1]$$

$$\text{ss} \triangleq \{(a,b) : a,b \in H \wedge \text{a.proc} = \text{b.proc}\}\ [1]$$

$$\text{so} \triangleq \text{rb} \cap \text{ss}\ [1]$$

Premitives

vis: visibility $\wedge$ AC $\wedge$ TC

ar: strict total order (for conflic resolution)

* Need rework, in current representation, both vis and ar are modeled as rb $\wedge$ AC $\wedge$ TC

Example

https://github.com/stepbrobd/consistency#example

Models

- Monotonic Reads
- Monotonic Writes
- Read Your Writes
- Writes Follow Reads
- PRAM Consistency

Results

Satisfiability of models:

```
MonotonicReads: True
MonotonicWrites: True
PRAMConsistency: True
ReadYourWrites: True
WritesFollowReads: True
```

Results

Pairwise validity (check whether $\neg(\text{LHS} \Rightarrow \text{RHS}) \equiv \text{LHS} \wedge \neg\ \text{RHS}$ is unsatisfiable or not):

```
MonotonicReads <- MonotonicWrites: False
MonotonicReads <- PRAMConsistency: False
MonotonicReads <- ReadYourWrites: False
MonotonicReads <- WritesFollowReads: False
MonotonicWrites <- MonotonicReads: False
MonotonicWrites <- PRAMConsistency: False
MonotonicWrites <- ReadYourWrites: False
MonotonicWrites <- WritesFollowReads: False
```

```
PRAMConsistency <- MonotonicReads: True
PRAMConsistency <- MonotonicWrites: False
PRAMConsistency <- ReadYourWrites: True
PRAMConsistency <- WritesFollowReads: False
ReadYourWrites <- MonotonicReads: False
ReadYourWrites <- MonotonicWrites: False
ReadYourWrites <- PRAMConsistency: False
ReadYourWrites <- WritesFollowReads: False
WritesFollowReads <- MonotonicReads: False
WritesFollowReads <- MonotonicWrites: False
WritesFollowReads <- PRAMConsistency: False
WritesFollowReads <- ReadYourWrites: False
```

Results

Composition (conjunction of assertions):

```
PRAM <- {RYW, MR, MW}: False
{RYW, MR, MW} <- PRAM: False
PRAM <- {MR, RYW}: True
{MR, RYW} <- PRAM: False
```

References

[1] P. Viotti and M. Vukolić, “Consistency in Non-Transactional Distributed Storage Systems”. 2016.