A Cutoff Rule For Parameterized Distributed Protocols in Prenex Normal Form

Ian Dardik

March 4, 2022

1 Introduction

In this note, we consider the verification problem of a transition system $T = (I, \Delta)$ parameterized by a single sort P of identical elements. We assume that a candidate inductive invariant Φ (which implies our key safety property) is given. Φ universally quantifies over one or more variables, Δ (the transition relation) exitentially quantifies over one or more variables, and and both Φ and Δ are in Prenex Normal Form (PNF). We adopt the convention of [2] where $T(P)$ is the template of T, and $T(|P|)$ is a finite instantiation. We also will consider the prime (') symbol to be an operator that can be recursively applied to a formula, only affecting (sticking to) state variables.

In this note, we will build several lemmas that lead to an interesting result: $\Phi(P)$ is an inductive invariant for $T(P)$ iff $\Phi(m + n)$ is an inductive invariant for $T(m + n)$, where m is the number of variables that Φ quantifies over and n is the number of variables that Δ quantifies over. This result is useful for the verification problem laid out above because it reduces the burden to model checking the single finite instance $T(m + n)$. Essentially, $m + n$ is a cutoff instance size for proving that our inductive invariant holds.

2 Preliminaries

In this section we cover several preliminary items that we use to prove the MN Theorem.

2.1 Without Loss Of Generality

We will assume that the parameter $P = \{1, ..., |P|\}$. This assumption comes without loss of generality because each member of P is assumed to be identical. We make the notion of "identical" precise in Assumption 1.

2.2 Assumptions

This section contains the list of assumptions for the transition system we work with. In other words, these assumptions are the requirements for the M-N Theorem to hold.

Assumption 1 (P Has Identical Elements)**.** Let s be a state, f be a grounded formula, and g be a permutation. Then we assume that:

$$(s \models f) \leftrightarrow (g(s) \models g(f))$$

2.3 Definitions

Definition 1 (States)**.** Let $k \in \mathbb{N}$, then:

$$\text{States}(k) := \{\text{all states when } |P| = k\}$$

Definition 2 (Ground Formulas)**.** Let F be a quantified formula and $k \in \mathbb{N}$.

$$\text{Gr}(F, k) := \{f | (f \text{ is a ground formula of } F(k)) \wedge (f \models F(k))\}$$

Example 1. $\text{Gr}((\forall p, q, p = q), 2) := \{(1 = 1), (2 = 2)\}$

Example 2. $\text{Gr}((\forall p, q, p \neq q), 3) := \{(1 \neq 2), (1 \neq 3), (2 \neq 1), (2 \neq 3), (3 \neq 1), (3 \neq 2)\}$

3 MN

Theorem 1 (M-N)**.** Suppose that Φ is in PNF with only universal quantifiers, while Δ is in PNF with only existential quantifiers. Let m be the number of variables that Φ quantifies over and n be the number of variables that Δ quantifies over. If $\Phi(m + n)$ is an inductive invariant, then $\Phi(k)$ is also an inductive invariant for any $k > m + n$.

Proof. Let $k > m + n$ be given and assume that $[\Phi \wedge \Delta \to \Phi'](m + n)$ is valid. Let $s \in \text{States}(k)$ such that $s \models \Phi(k)$, and let δ be the next transition, i.e. $\delta \models \Delta(k)$. Finally, let $f' \in \text{Gr}(\Phi', k)$ be arbitrary, then we must show that $(s \wedge \delta) \models f'$.

Next, let g be a permutation such that $g(\delta) \models \Delta(m + n)$ and $g(f') \in \text{Gr}(\Phi', m + n)$, i.e. $g(f') \models \Phi'(m + n)$. We know that we can find such a g because δ will contain at most n distinct elements of P and f' will contain at most m distinct elements of P. Notice that $\text{Gr}(\Phi, m + n) \subset \text{Gr}(\Phi, k)$ so $s \models \Phi(m + n)$, and furthermore $g(s) \models \Phi(m + n)$. Thus $g(s \wedge \delta) \models \Phi'(m + n)$, and in particular, $g(s \wedge \delta) \models g(f')$. Therefore $s \wedge \delta \models f'$ by Assumption 1. $\square$

4 Case Studies

In this section we visit several (more coming soon) distributed protocols that are parameterized by a single sort and adhere to the property of Assumption 1.

4.1 Peterson's Mutex Protocol

Peterson's Mutex Protocol can be encoded with a transition function Δ in PNF that quantifies over two variables. A sample inductive invariant candidate is given in [1] that quantifies of two variables and works for $|P| = 2$:

```
Phi == \A p,q \in ProcSet :
   /\ pc[p] \in {"a3","a4","cs"} => flag[p]
   /\ (p#q /\ pc[p] = "cs" /\ pc[q] = "a4") => turn = p
   /\ (p # q) => ~(pc[p] = "cs" /\ pc[q] = "cs")
```

However, by the M-N Theorem, we must show that Φ is an inductive invariant for the case when $|P| = 4$. In fact, we easily see that Φ fails to be inductive in the case:

```
/\ turn = 1                 /\ turn = 2
/\ pc[1] = "cs"             /\ pc[1] = "cs"
/\ pc[2] = "a4"     ->      /\ pc[2] = "a4"
/\ pc[3] = "a3"             /\ pc[3] = "a4"
/\ a3(3,2)
```

This example uses states to describe the counterexample, but we can also describe it using the FIP $M_\Phi(1,2) \wedge M_\Delta(3,2)$ from $[\Phi \wedge \Delta](4)$. When this FIP is true, both $M_\Phi(1,2)$ and $M_\Phi(1,3)$ fail to hold in the next state, showing that $M_\Phi(1,2)$–and hence Φ–is not inductive.

This example shows how a FIP describes a specific relationship between Φ and Δ; in this case the specific relationship leads to a counterexample. It is important to note that it is only possible to describe this particular counterexample using a FIP with a minimum of three elements in P, which is precisely why we do not detect the counter example in Peterson's Protocol when $|P| = 2$.

It is also worthwhile to note that we could derive the same counterexample using an equivalent FIP, say $M_\Phi(3,2) \wedge M_\Delta(1,2)$. This shows how FIP equivalency partitions a formula $\Phi \wedge \Delta$ into classes of specific relationships that a transition system can exhibit.

References

[1] Parametric Peterson's Mutex Protocol. `https://github.com/iandardik/iinf/blob/master/ii_cutoff/mn_thm/PetersonParametric.tla`, 2022.

[2] Aman Goel and Karem Sakallah. On Symmetry and Quantification: A New Approach to Verify Distributed Protocols. In *NASA Formal Methods Symposium*, pages 131–150. Springer, 2021.