# Bounded Partial-Order Reduction

**Proof Companion Source Material**

Katherine Coons* Madanlal Musuvathi† Kathryn McKinley*†

University of Texas, Austin* Microsoft Research†

This companion source material contains the proofs for all the theorems in the main paper. For completeness, it repeats the definitions, theorems, and lemmas.

## 1. Definitions

**Definition 1.1. Traces** [1].
Equivalence classes of $\equiv_\Lambda$ are *traces* over $\Lambda$. The term $[\omega]$ denotes the trace that contains the sequence of transitions $\omega$.

**Definition 1.2. *Prefix*$([\omega])$** [1].
*Prefix*$([\omega])$ returns the set containing all prefixes of all sequences in the Mazurkiewicz trace defined by $\omega$.

**Definition 1.3. Local sufficient.**
A nonempty set $T \subseteq \mathcal{T}$ of transitions enabled in a state $s$ in $A_{G(Bv,c)}$ is *local sufficient* in $s$ if and only if for all sequences $\omega$ of transitions from $s$ in $A_{G(Bv,c)}$, there exists a sequence $\omega'$ from $s$ in $A_{G(Bv,c)}$ such that $\omega \in$ *Prefix*$([\omega'])$ and $\omega'_1 \in T$.

**Definition 1.4. *ext*$(s, t)$.**
Given a state $s =$ *final*$(S)$ and a transition $t \in$ *enabled*$(s)$, *ext*$(s, t)$ returns the unique sequence of transitions $\beta$ from $s$ such that

1. $\forall i \in dom(\beta) : \beta_i.tid = t.tid$
2. $t.tid \notin enabled(final(S.\beta))$

### 1.1 Preemption-bounded search

**Definition 1.5. Preemption bound** [2].

$$Pb(t) = 0$$

$$Pb(S.t) = \begin{cases} Pb(S) + 1 & \text{if } t.tid \neq \text{last}(S).tid \text{ and} \\ & \text{last}(S).tid \in enabled(final(S)) \\ Pb(S) & \text{otherwise} \end{cases}$$

**Definition 1.6. Preemption-bound persistent sets.**
A set $T \subseteq \mathcal{T}$ of transitions enabled in a state $s =$ *final*$(S)$ is *preemption-bound persistent* in $s$ iff for all nonempty sequences $\alpha$ of transitions from $s$ in $A_{G(Pb,c)}$ such that $\forall i \in dom(\alpha), \alpha_i \notin T$ and for all $t \in T$,

1. $Pb(S.t) \leq Pb(S.\alpha_1)$
2. if $Pb(S.t) < Pb(S.\alpha_1)$, then $t \leftrightarrow \text{last}(\alpha)$ and $t \leftrightarrow next(final(S.\alpha), \text{last}(\alpha).tid)$
3. if $Pb(S.t) = Pb(S.\alpha_1)$, then $ext(s, t) \leftrightarrow \text{last}(\alpha)$ and $ext(s, t) \leftrightarrow next(final(S.\alpha), \text{last}(\alpha).tid)$

**Definition 1.7. *PC* for Explore$(S)$ – Preemption bound.**
$\forall u \forall \omega :$ **if** $Pb(S.\omega) \leq c$ **then** $Post(S.\omega, len(S), u)$

**Definition 1.8. *Post*$(S, k, u)$ – Preemption bound.**
$\forall v :$ **if** $i = max(\{i \in dom(S) \mid S_i \not\leftrightarrow next(final(S), u)$ **and** $S_i.tid = v\})$ **then**

1. **if** $i \leq k$ **then**
   **if** $u \in enabled(pre(S, i))$ **then** $u \in backtrack(pre(S, i))$
   **else** $backtrack(pre(S, i)) = enabled(pre(S, i))$
2. **if** $j = max(\{j \in dom(S) \mid j = 0$ **or** $S_{j-1}.tid \neq S_j.tid$ **and** $j < i\})$ **and** $j < k$ **then**
   **if** $u \in enabled(pre(S, j))$ **then** $u \in backtrack(pre(S, j))$
   **else** $backtrack(pre(S, j)) = enabled(pre(S, j))$

### 1.2 Fair-bounded search

**Definition 1.9. Fair bound (*Fb*).**
Let $Y(S, u)$ return Thread $u$'s yield count in *final*$(S)$.

$$Fb(t) = 0$$
$$Fb(S.t) = max(Fb(S), \max_{u \in enabled(final(S))}(Y(S, t.tid) - Y(S, u)))$$

**Definition 1.10. Fair-bound persistent sets.**
A set $T \subseteq \mathcal{T}$ of transitions enabled in a state $s =$ *final*$(S)$ is *fair-bound persistent* in $s$ if and only if for all nonempty sequences $\alpha$ of transitions from $s$ in $A_{G(Fb,c)}$ such that $\forall i \in dom(\alpha) : \alpha_i \notin T$ and for all $t \in T$,

1. $Fb(S.t) \leq c$
2. if $t$ is a release operation, then $\forall u \in enabled(s)$ : $next(s, u) \in T$
3. $t \leftrightarrow \text{last}(\alpha)$

**Definition 1.11. *PC* for Explore$(S)$ - Fair bound.**
$\forall u \forall \omega :$ **if** $Fb(S.\omega) \leq c$ **and** $len(S.\omega) \leq MAX$ **then** $Post(S.\omega, len(S), u)$

**Definition 1.12.** ***Post***$(S, k, u)$ **- Fair bound.**
$\forall v$ : **if** $i = max(\{i \in dom(S) \mid S_i \nleftrightarrow next(final(S), u)$ **and** $S_i.tid = v\})$ **and** $i \leq k$ **then**
  **if** $u \in enabled(pre(S, i))$ and $S_i$ is not a release **then**
    $u \in backtrack(pre(S, i))$
  **else** $backtrack(pre(S, i)) = enabled(pre(S, i))$

## 2. Proofs

Let $A_{R(Bv,c)}$ be the reduced state space explored by a selective search that explores a nonempty local sufficient set in each state.

**Theorem 1.** *Let $s$ be a state in $A_{R(\mathrm{Bv},c)}$, and let $l$ be a local state reachable from $s$ in $A_{G(\mathrm{Bv},c)}$ by a sequence $\omega$ of transitions. Then, $l$ is also reachable from $s$ in $A_{R(\mathrm{Bv},c)}$.*

*Proof.* The proof is by induction on the length of the longest sequence of transitions that leads to $l$ from $s$ in $A_{G(Bv,c)}$.

**Case 1.1. Base Case.**
For $len(\omega) = 0$ the result is immediate.

**Case 1.2. Inductive case.**
Let $l$ be a local state such that the longest sequence of transitions $\omega$ from $s$ to $l$ has length $n + 1$. Let $u$ be a thread such that $l = local(final(S.\omega), u)$. Let $T$ be the nonempty local sufficient set explored from $s$ in $A_{R(Bv,c)}$.

By Definition 1.3 of local sufficient sets, there exists a sequence $\omega'$ of transitions from $s$ in $A_{G(Bv,c)}$ such that $\omega'_1 \in T$ and $\omega \in Prefix([\omega'])$. Thus, by Definition 1.2 of the prefix function, there exists a sequence $\beta$ of transitions from $final(S.\omega)$ such that $\omega.\beta \in [\omega']$. Assume that none of the transitions in $\omega$ are by $u$. Then, by definition of local states,

$$local(final(S.\omega), u) = local(final(S), u)$$

and the result is immediate.

Assume that a transition in $\omega$ is by $u$. Let $i \in dom(\omega)$ be the maximum value of $i$ such that $\omega_i.tid = u$. Because $\omega.\beta \in [\omega']$, there must exist $j \in dom(\omega')$ such that $\omega'_j = \omega_i$. Let $\omega' = \alpha.t.\gamma$ such that $t = \omega'_j$. Because $\omega.\beta \in [\omega']$,

$$local(final(S.\omega), u) = local(final(S.\alpha.t), u)$$

Thus, $\omega'$ leads to $l$. Because $\omega'_1$ is in $T$, it is explored from $s$ and the state $final(S.\omega'_1)$ is reachable in $A_{R(Bv,c)}$. Because $\omega$ is the longest sequence of transitions that leads to $l$ in $A_{G(Bv,c)}$, $len(S.\alpha.t) \leq len(\omega)$. Thus, from $final(S.\omega'_1)$, $l$ is reachable via a sequence of transitions of length $n$. By the inductive hypothesis, $l$ is also reachable from $s$ in $A_{R(Bv,c)}$.

□

### 2.1 Preemption-bounded search

Let $A_{R(Pb,c)}$ be the reduced state space for a selective search that explores a preemption-bound persistent set in each state. We provide two lemmas to manage the bound, and a theorem stating that a nonempty preemption-bound persistent set is local sufficient.

**Lemma 2.** *Let $\alpha$ and $\beta$ be nonempty sequences of transitions from $s = \mathrm{final}(S)$ in $A_{G(\mathrm{Pb},c)}$ such that*

1. $\beta \leftrightarrow \alpha$
2. $\mathrm{Pb}(S.\beta_1) \leq \mathrm{Pb}(S.\alpha_1)$
3. $\forall i \in \mathrm{dom}(\beta) : \beta_i.tid = \beta_1.tid$
4. $\beta \leftrightarrow \mathrm{next}(\mathrm{final}(S.\alpha_1 \ldots \alpha_i), \alpha_i.tid), 1 \leq i < \mathrm{len}(\alpha)$
5. *if* $\mathrm{Pb}(S.\beta_1) = \mathrm{Pb}(S.\alpha_1)$, *then* $\beta_1.tid \notin \mathrm{enabled}(\mathrm{final}(S.\beta))$

*Then, $\beta.\alpha$ is a sequence of transitions from $s$ in $A_{G(\mathrm{Pb},c)}$.*

*Proof.* By Assumption 1, $\beta.\alpha$ is a sequence of transitions from $s$ in $A_G$. For each preemption in $S.\beta.\alpha$, from left to right, show that there exists a unique preemption in $S.\alpha$. Assume that $\beta_1$ requires a preemption from $final(S)$. By Assumption 2, $\alpha_1$ also requires a preemption from $final(S)$. By Assumption 3, no transition in $\beta$ after $\beta_1$ requires a preemption.

Assume that $\alpha_1$ requires a preemption from $final(S.\beta)$. Then,

$$\beta_1.tid \in enabled(final(S.\beta))$$

and thus by Assumptions 2 and 5, $Pb(S.\beta_1) < Pb(S.\alpha_1)$. Thus, $\alpha_1$ requires a preemption from $final(S)$ and $\beta_1$ does not, so this preemption is unique. Assume that a transition $\alpha_i$, $2 \leq i \leq len(\alpha)$, requires a preemption in $S.\beta.\alpha$. By Assumption 4, $\alpha_i$ also requires a preemption in $S.\alpha$. Thus, for each preemption in $S.\beta.\alpha$ there exists a unique preemption in $S.\alpha$ and

$$Pb(S.\beta.\alpha) \leq Pb(S.\alpha) \leq c$$

Thus, $\beta.\alpha$ is a sequence of transitions from $s$ in $A_{G(Pb,c)}$.

□

**Lemma 3.** *Let $T$ be a nonempty preemption-bound persistent set in a state $s = \mathrm{final}(S)$ in $A_{R(\mathrm{Pb},c)}$ and let $\alpha.\beta.\gamma$ be a sequence of transitions from $s$ in $A_{G(\mathrm{Pb},c)}$ such that $\alpha$ and $\beta$ are nonempty and*

1. $\forall i \in \mathrm{dom}(\alpha) : \alpha_i \notin T$
2. $\beta_1 \in T$
3. $\forall i \in \mathrm{dom}(\beta) : \beta_i.tid = \beta_1.tid$
4. *if* $\mathrm{Pb}(S.\beta_1) < \mathrm{Pb}(S.\alpha_1)$ *then* $\mathrm{len}(\beta) = 1$
5. *if* $\mathrm{Pb}(S.\beta_1) = \mathrm{Pb}(S.\alpha_1)$ *and $\gamma$ is empty, then* $\beta_1.tid \notin \mathrm{enabled}(\mathrm{final}(S.\beta))$
6. *if* $\mathrm{Pb}(S.\beta_1) = \mathrm{Pb}(S.\alpha_1)$ *and $\gamma$ is nonempty, then* $\gamma_1.tid \neq \beta_1.tid$

*Then, $\beta.\alpha.\gamma$ is a sequence of transitions from $s$ in $A_{G(\mathrm{Pb},c)}$.*

*Proof.* By Assumptions 1-4 and by Requirements 2 and 3 of Definition 1.6 of preemption-bound persistent sets, $\beta \leftrightarrow \alpha$

and

$$\forall i \in \textit{dom}(\alpha) : \beta \leftrightarrow \textit{next}(\textit{final}(S.\alpha_1 \ldots \alpha_i), \alpha_i.tid) \quad (1)$$

Thus, $\beta.\alpha.\gamma$ is a sequence of transitions from $s$ in $A_G$. For each preemption in $S.\beta.\alpha.\gamma$, from left to right, show that there exists a unique preemption in $S.\alpha.\beta.\gamma$. Assume that $\beta_1$ requires a preemption from *final*$(S)$. Then, by Requirement 1 of Definition 1.6 of preemption-bound persistent sets, $\alpha_1$ also requires a preemption from *final*$(S)$. By Assumption 3, no transition in $\beta$ after $\beta_1$ requires a preemption.

Assume that $\alpha_1$ requires a preemption from *final*$(S.\beta)$. If $\textit{Pb}(S.\beta_1) < \textit{Pb}(S.\alpha_1)$, then $\alpha_1$ requires a preemption from *final*$(S)$ and $\beta_1$ does not, so this preemption is unique. Otherwise, by Requirement 1 of Definition 1.6 of preemption-bound persistent sets, $\textit{Pb}(S.\beta_1) = \textit{Pb}(S.\alpha_1)$. Because $\alpha_1$ requires a preemption from *final*$(S.\beta)$,

$$\beta_1.tid \in \textit{enabled}(\textit{final}(S.\beta)) \quad (2)$$

By Assumption 5, $\gamma$ is nonempty, and by Assumption 6 $\gamma_1.tid \neq \beta_1.tid$. By Equation 2 and Requirement 3 of Definition 1.6 of preemption-bound persistent sets,

$$\beta_1.tid \in \textit{enabled}(\textit{final}(S.\alpha.\beta))$$

Thus, $\gamma_1$ requires a preemption from *final*$(S.\alpha.\beta)$. Assume that a transition $\alpha_i$, $2 \leq i \leq \textit{len}(\alpha)$, requires a preemption in $S.\beta.\alpha.\gamma$. By Equation 1, $\alpha_i$ also requires a preemption in $S.\alpha.\beta.\gamma$.

Assume that $\gamma_1$ requires a preemption from *final*$(S.\beta.\alpha)$. Then,

$$\text{last}(\alpha).tid \in \textit{enabled}(\textit{final}(S.\beta.\alpha))$$

By Equation 1,

$$\text{last}(\alpha).tid \in \textit{enabled}(\textit{final}(S.\alpha))$$

Because $\beta \leftrightarrow \alpha$, $\beta_1.tid \neq \text{last}(\alpha).tid$. Thus, $\beta_1$ requires a preemption from *final*$(S.\alpha)$. Assume that a transition $\gamma_i$, $2 \leq i \leq \textit{len}(\gamma)$, requires a preemption in $S.\beta.\alpha.\gamma$. Because $\beta \leftrightarrow \alpha$, $\textit{final}(S.\alpha.\beta.\gamma_1) = \textit{final}(S.\beta.\alpha.\gamma_1)$. Thus, by Definition 1.5 of the preemption bound, $\gamma_i$ also requires a preemption in $S.\alpha.\beta.\gamma$. Thus, for each preemption in $S.\beta.\alpha.\gamma$ there exists a unique preemption in $S.\alpha.\beta.\gamma$ and

$$\textit{Pb}(S.\beta.\alpha.\gamma) \leq \textit{Pb}(S.\alpha.\beta.\gamma) \leq c$$

Thus, $\beta.\alpha.\gamma$ is a sequence of transitions from $s$ in $A_{G(\textit{Pb},c)}$.

□

**Theorem 4.** *If $T$ is a nonempty preemption-bound persistent set in a state $s$ in $A_{R(\text{Pb},c)}$, then $T$ is local sufficient in $s$.*

*Proof.* Let $s$ be a state in $A_{R(\textit{Pb},c)}$ and let $l$ be a local state reachable from $s$ in $A_{G(\textit{Pb},c)}$ via a nonempty sequence $\omega$ of transitions.

**Case 4.1.** $\forall i \in \textbf{\textit{dom}}(\omega) : \omega_i \notin T$**.**
Let $t$ be any transition in $T$. By Requirement 1 of Definition 1.6 of preemption-bound persistent sets, $\textit{Pb}(S.t) \leq \textit{Pb}(S.\omega_1)$. Let $\beta = t$ if $\textit{Pb}(S.t) < \textit{Pb}(S.\omega_1)$, and let $\beta = \textit{ext}(s,t)$ otherwise. Consider the sequence $\omega' = \beta.\omega$. By Requirements 2 and 3 of Definition 1.6 of preemption-bound persistent sets, $\beta \leftrightarrow \omega$ and $\forall i \in \textit{dom}(\omega) : \beta \leftrightarrow \textit{next}(\textit{final}(S.\omega_1 \ldots \omega_i), \omega_i.tid)$. Thus, by Lemma 2 $\beta.\omega$ is a sequence of transitions from $s$ in $A_{G(\textit{Pb},c)}$ and by Definition 1.1 of a trace, $\omega.\beta \in [\omega']$. By Definition 1.2 of the prefix function, $\omega \in \textit{Prefix}([\omega'])$. Thus, $T$ is local sufficient in $s$.

**Case 4.2.** $\exists i \in \textbf{\textit{dom}}(\omega) : \omega_i \in T$**.**
Let $\omega = \alpha.\beta.\gamma$ such that

1. $\forall i \in \textit{dom}(\alpha) : \alpha_i \notin T$
2. $\beta_1 \in T$
3. $\forall i \in \textit{dom}(\beta) : \beta_i.tid = \beta_1.tid$
4. if $\textit{Pb}(S.\beta_1) < \textit{Pb}(S.\alpha_1)$ then $\textit{len}(\beta) = 1$
5. if $\textit{Pb}(S.\beta_1) = \textit{Pb}(S.\alpha_1)$ and $\gamma$ is nonempty, then $\gamma_1.tid \neq \beta_1.tid$

Assume that $\alpha$ is empty. Then, $T$ is local sufficient in $s$ because $\omega_1 \in T$ and $l$ is reachable via $\omega$. Assume that $\alpha$ is nonempty. By Requirement 1 of Definition 1.6 of preemption-bound persistent sets, $\textit{Pb}(S.\beta_1) \leq \textit{Pb}(S.\alpha_1)$.

**Case 4.2a.** $\gamma$ **is nonempty, or** $\gamma$ **is empty and** $\beta_1.tid \notin \textbf{\textit{enabled}}(\textbf{\textit{final}}(S.\beta))$**, or** $\textbf{\textit{Pb}}(S.\beta_1) < \textbf{\textit{Pb}}(S.\alpha_1)$**.**
Consider the sequence $\omega' = \beta.\alpha.\gamma$, i.e., $\omega$ with $\beta$ moved to the beginning. By Requirements 2 and 3 of Definition 1.6 of preemption-bound persistent sets, $\beta \leftrightarrow \alpha$ and $\forall i \in \textit{dom}(\alpha) : \beta \leftrightarrow \textit{next}(\textit{final}(S.\alpha_1 \ldots \alpha_i), \alpha_i.tid)$. Thus, by Lemma 3 $\omega'$ is a sequence of transitions from $s$ in $A_{G(\textit{Pb},c)}$ and by Definition 1.1 of a trace $\omega' \in [\omega]$. By Definition 1.2 of the prefix function $\omega \in \textit{Prefix}([\omega'])$, so $T$ is local sufficient in $s$.

**Case 4.2b.** $\gamma$ **is empty,** $\beta_1.tid \in \textbf{\textit{enabled}}(\textbf{\textit{final}}(S.\beta))$**, and** $\textbf{\textit{Pb}}(S.\beta_1) = \textbf{\textit{Pb}}(S.\alpha_1)$**.**
Let $\beta' = \textit{ext}(s, \beta_1)$. Consider the sequence $\omega' = \beta'.\alpha$. By Requirement 3 of Definition 1.6 of preemption-bound persistent sets, $\beta' \leftrightarrow \alpha$ and $\forall i \in \textit{dom}(\alpha) : \beta' \leftrightarrow \textit{next}(\textit{final}(S.\alpha_1 \ldots \alpha_i), \alpha_i.tid)$. Thus, by Lemma 2 $\beta'.\omega$ is a sequence of transitions from $s$ in $A_{G(\textit{Pb},c)}$ and by Definition 1.1 of a trace $\omega.\beta' \in [\omega']$. By Definition 1.2 of the prefix function $\omega \in \textit{Prefix}([\omega'])$, so $T$ is local sufficient in $s$.

□

**Lemma 5.** *Whenever a state $s = \text{final}(S)$ is backtracked by Algorithm 1, the set $T$ of transitions explored from $s$ is preemption-bound persistent in $s$, provided that postcondition* PC *holds for every recursive call* ***Explore***$(S.t)$ *for all $t \in T$.*

**Algorithm 1** BPOR with bound function *Bv* and bound $c$

```
1:  Initially, Explore(ε) from s_0
2:  procedure Explore(S) begin
3:    Let s = final(S)
      # Add backtrack points
4:    for all (u ∈ Tid) do
5:      for all (v ∈ Tid | v ≠ u) do
          # Find most recent dependent transition
6:        if (∃i = max({i ∈ dom(S) | S_i ↮ next(s, u) and S_i.tid = v})) then
7:          Backtrack(S, i, u)
      # Continue the search by exploring successor states
8:    Initialize(S)
9:    Let visited = ∅
10:   while (∃u ∈ (enabled(s) ∩ backtrack(s) \ visited)) do
11:     add u to visited
12:     if (Bv(S.next(s, u)) ≤ c) then
13:       Explore(S.next(s, u))
```

**Algorithm 2** BPOR for preemption-bounded search

```
1:  procedure Initialize(S) begin
2:    if (last(S).tid ∈ enabled(final(S))) then
3:      add last(S).tid to backtrack(final(S))
4:    else
5:      add any u ∈ enabled(final(S)) to backtrack(final(S))
6:  procedure Backtrack(S, i, u) begin
7:    AddBacktrackPoint(S, i, u)
8:    if (j = max({j ∈ dom(S) | j = 0 or S_{j-1}.tid ≠ S_j.tid and j < i})) then
9:      AddBacktrackPoint(S, j, u)
10: procedure AddBacktrackPoint(S, i, u) begin
11:   if (u ∈ enabled(pre(S, i))) then
12:     Add u to backtrack(pre(S, i))
13:   else
14:     backtrack(pre(S, i)) = enabled(pre(S, i))
```

*Proof.* Let $T = \mathit{next}(s, u) \mid u \in \mathit{backtrack}(s)$. Show that if $T$ violates any requirement in Definition 1.6 of preemption-bound persistent sets, then we have a contradiction.

**Case 5.1.** $T$ **violates Requirement 1.**
Proceed by contradiction. Assume that there exist transitions $t \in T$ and $t' \notin T$ such that $t$ and $t'$ are both enabled in $s$ and $\mathit{Pb}(S.t') < \mathit{Pb}(S.t)$. By Definition 1.5 of the preemption bound

$$t'.tid = \text{last}(S).tid$$

Thus, by Line 3 of Algorithm 2, $t'.tid \in \mathit{backtrack}(s)$ and thus $t' \in T$, and we have a contradiction.

**Case 5.2.** $T$ **violates Requirement 2.**
Proceed by contradiction. Assume that there exists a nonempty sequence $\alpha$ of transitions from $s$ in $A_{G(\mathit{Pb},c)}$ and a transition $t \in T$ such that, if we let $u = \text{last}(\alpha).tid$:

1. $\forall i \in \mathit{dom}(\alpha) : \alpha_i \notin T$
2. $\mathit{Pb}(S.t) < \mathit{Pb}(S.\alpha_1)$
3. $t$ is dependent with last$(\alpha)$ or with $\mathit{next}(\mathit{final}(S.\alpha), u)$

Let $n = \mathit{len}(\alpha)$ and let $\omega = \alpha_1 \ldots \alpha_{n-1}$, i.e., $\alpha$ with its last transition removed. Let there be no prefixes of $\alpha$ that also meet the criteria above, and thus

4. $t \leftrightarrow \omega$ and $\forall i \in \mathit{dom}(\omega)$ :
 $t \leftrightarrow \mathit{next}(\mathit{final}(S.\omega_1 \ldots \omega_i), \omega_i.tid)$

Assume that $t.tid = u$. Because $t \leftrightarrow \omega$,

$$t = \mathit{next}(\mathit{final}(S), u) = \mathit{next}(\mathit{final}(S.\omega), u) = \text{last}(\alpha)$$

Thus, last$(\alpha) \in T$ and we have a contradiction.

Assume that $t.tid \neq u$. Let $\omega' = \omega$ if $t$ is dependent with last$(\alpha)$, and let $\omega' = \alpha$ if $t \leftrightarrow \alpha$ and $t$ is dependent with $\mathit{next}(\mathit{final}(S.\alpha), u)$. Consider the postcondition

$$\mathit{Post}(S.t.\omega', \mathit{len}(S) + 1, u)$$

for the recursive call **Explore**$(S.t)$. By Lemma 2, $t.\omega'$ is a sequence of transitions from $s$ in $A_{G(\mathit{Pb},c)}$. Because $t \leftrightarrow \omega'$, $t$ is the most recent transition by $t.tid$ that is dependent with $\mathit{next}(\mathit{final}(S.t.\omega'), u)$. Thus, by Definition 1.8 of *Post*, either $u \in \mathit{backtrack}(s)$, or $\mathit{backtrack}(s) = \mathit{enabled}(s)$ and thus $\alpha_1 \in T$. In either case, we have a contradiction.

**Case 5.3.** $T$ **violates Requirement 3.**
Proceed by contradiction. Assume that there exists a nonempty sequence $\alpha$ of transitions from $s$ in $A_{G(\mathit{Pb},c)}$ and a transition $t \in T$ such that, if we let $u = \text{last}(\alpha).tid$ and let $\beta = \mathit{ext}(s, t)$:

1. $\mathit{Pb}(S.t) = \mathit{Pb}(S.\alpha_1)$
2. $\forall i \in \mathit{dom}(\alpha) : \alpha_i \notin T$
3. a transition in $\beta$ is dependent with last$(\alpha)$ or with $\mathit{next}(\mathit{final}(S.\alpha), u)$

Let $n = \mathit{len}(\alpha)$, and let $\omega = \alpha_1 \ldots \alpha_{n-1}$, i.e., $\alpha$ with its last transition removed. Let there be no prefixes of $\alpha$ that also meet the criteria above, and thus

4. $\beta \leftrightarrow \omega$ and $\forall i \in \mathit{dom}(\omega) : \beta \leftrightarrow \mathit{next}(\mathit{final}(S.\omega_1 \ldots \omega_i), \omega_i.tid)$

Assume that $\beta_1.tid = u$. Because $\beta \leftrightarrow \omega$,

$$\beta_1 = \mathit{next}(\mathit{final}(S), u) = \mathit{next}(\mathit{final}(S.\omega), u) = \text{last}(\alpha)$$

Thus, last$(\alpha) \in T$ and we have a contradiction.

Assume that $\beta_1.tid \neq u$. Let $\beta_k$ be the last transition in $\beta$ that is dependent with last$(\alpha)$ or with $\mathit{next}(\mathit{final}(S.\alpha), u)$. Let $\omega' = \omega$ if $\beta_k$ is dependent with last$(\alpha)$, and let $\omega' = \alpha$ if $\beta \leftrightarrow \alpha$ and $\beta_k$ is dependent with $\mathit{next}(\mathit{final}(S.\alpha), u)$.

By Lemma 2, $\beta.\omega'$ is a sequence of transitions from $s$ in $A_{G(Pb,c)}$. Consider the postcondition

$$Post(S.\beta.\omega', len(S) + 1, u)$$

for the recursive call **Explore**$(S.\beta_1)$. Because $\beta \leftrightarrow \omega'$, $\beta_k$ is the most recent transition by $\beta_1.tid$ that is dependent with $next(final(S.\beta.\omega'), u)$. Because $Pb(S.\beta_1) = Pb(S.\alpha_1)$, by Definition 1.5 of the preemption bound either $\beta_1.tid \neq \text{last}(S).tid$, or $S$ is empty. Because all transitions in $\beta$ are by the same thread, $\beta_1$ is the most recent such location to $\beta_k$. Thus, by Requirement 2 of Definition 1.8 of postcondition *Post*, either $u \in backtrack(s)$, or $backtrack(s) = enabled(s)$ and thus $\alpha_1 \in T$. In either case, we have a contradiction.

□

Thus, if postcondition *PC* holds in each state $s$ that Algorithm 1 explores with the **Backtrack** procedure from Algorithm 2, then the set of transitions Algorithm 1 explores from $s$ is preemption-bound persistent in $s$.

Next, we prove that postcondition *PC* holds in each state $s$ that Algorithm 1 explores. First, we prove a lemma that simplifies the inductive step. Lemma 6 differs from the similar lemma used in depth-bounded and context-bounded search because it must account for the more complex postcondition that preemption-bounded search requires.

**Lemma 6.** *Let $s = \text{final}(S)$ be a state in $A_{R(\text{Pb},c)}$, let $\omega$ and $\omega'$ be nonempty sequences of transitions from $s$ in $A_{G(\text{Pb},c)}$ such that $\text{Pb}(S.\omega'_1) \leq \text{Pb}(S.\omega_1)$, and let $u$ be a thread such that*

1. $\exists\beta : \omega.\beta \in [\omega']$ ***and*** $\beta \leftrightarrow \text{next}(\text{final}(S.\omega), u)$, *or*
2. $\exists\beta : \omega'.\beta \in [\omega]$ ***and*** $\beta \leftrightarrow \text{next}(\text{final}(S.\omega), u)$

*Then,* $\text{Post}(S.\omega', \text{len}(S) + 1, u) \implies \text{Post}(S.\omega, \text{len}(S), u)$.

*Proof.* Because $\beta \leftrightarrow next(final(S.\omega), u)$,

$$next(final(S.\omega), u) = next(final(S.\omega'), u)$$

Assume that in Definition 1.8 of postcondition *Post*, $i \leq k$ for $Post(S.\omega, len(S), u)$. Then, $i$ and $j$ have the same values in $Post(S.\omega', len(S), u)$ that they have in $Post(S.\omega, len(S), u)$ because $\beta \leftrightarrow next(final(S.\omega), u)$.

Assume that $i > k$ for $Post(S.\omega, len(S), u)$. Because $Pb(S.\omega'_1) \leq Pb(S.\omega_1)$, by Definition 1.5 of the preemption bound either $S$ is empty or $\omega_1.tid \neq \text{last}(S).tid$. Thus, $j \geq k$ for $Post(S.\omega, len(S), u)$, so Definition 1.8 of *Post* does not require any backtrack points. In either case,

$$Post(S.\omega', len(S), u) \implies Post(S.\omega, len(S), u) \quad (3)$$

Because Requirement 1 of Definition 1.8 of *Post* requires that $i \leq k$ and Requirement 2 of Definition 1.8 of *Post* requires that $j < k$

$$Post(S.\omega', len(S) + 1, u) \implies Post(S.\omega', len(S), u)$$

Thus, by Equation 3,

$$Post(S.\omega', len(S) + 1, u) \implies Post(S.\omega, len(S), u)$$

□

**Theorem 7.** *Whenever a state $s = \text{final}(S)$ is backtracked during the search performed by Algorithm 1 in an acyclic state space, the postcondition* Post *for* ***Explore***$(S)$ *is satisfied, and the set $T$ of transitions explored from $s$ is preemption-bound persistent in $s$.*

*Proof.* The proof is by induction on the order in which states are backtracked.

**Base case.**
Because the search is acyclic, is performed in depth-first order, and the preemption bound provides a zero-cost transition in each state, the first backtracked state must be a deadlock state in which no transition is enabled. Thus, the postcondition for the first backtracked state is

$$\forall u : Post(S, len(S), u)$$

and is directly established by Lines 4-7 in Algorithm 1.

**Inductive case.**
Assume that each recursive call to **Explore**$(S.t)$ satisfies its postcondition. By Lemma 5, $T$ is preemption-bound persistent in $s$. Show that **Explore**$(S)$ satisfies its postcondition for any sequence $\omega$ of transitions from $s$ in $A_{G(Pb,c)}$ and for any thread $u$.

**Case 7.1.** $\forall i \in \boldsymbol{dom}(\omega) : \omega_i \notin T$ **and** $u \in \boldsymbol{backtrack}(s)$**.**
Because $u \in backtrack(s)$, $next(s, u) \in T$. By Definition 1.5 of preemption-bound persistent sets, $next(s, u) \leftrightarrow \omega$, and thus

$$next(final(S.\omega), u) = next(s, u)$$

Thus, $next(final(S.\omega), u) \leftrightarrow \omega$, and $Post(S.\omega, len(S), u)$ iff $Post(S, len(S), u)$. The latter is directly established by Lines 4-7 in Algorithm 1.

**Case 7.2.** $\forall i \in \boldsymbol{dom}(\omega) : \omega_i \notin T$ **and** $u \notin \boldsymbol{backtrack}(s)$**.**
Because $u \notin backtrack(s)$, $next(s, u) \notin T$. Let $t$ be any transition in $T$, and thus $t.tid \neq u$. Let $\beta = t$ if $Pb(S.t) < Pb(S.\omega_1)$, and let $\beta = ext(s, t)$ otherwise. Consider the sequence $\omega' = \beta.\omega$. By Definition 1.6 of preemption-bound persistent sets,

1. $Pb(S.t) \leq Pb(S.\omega_1)$
2. $\beta \leftrightarrow \omega$
3. $\forall i \in dom(\omega) : \beta \leftrightarrow next(final(S.\omega_1 \ldots \omega_i), \omega_i.tid)$

By Lemma 2, $\omega'$ is a sequence of transitions from $s$ in $A_{G(Pb,c)}$. Because $\beta \leftrightarrow \omega$, $\omega.\beta \in [\omega']$. By the inductive hypothesis for the recursive call **Explore**$(S.t)$,

$$Post(S.\omega', len(S) + 1, u)$$

Assume that *next*(*final*$(S.\omega'), u)$ is dependent with a transition in $\beta$. Because $\beta \leftrightarrow \omega$, the most recent dependent transition to *next*(*final*$(S.\omega'), u)$ by $\beta_1.tid$ must be in $\beta$. If $\beta_1$ is the most recent dependent transition, then by Requirement 1 of Definition 1.8 of *Post* either $u \in$ *backtrack*$(s)$, or *backtrack*$(s) =$ *enabled*$(s)$ and thus $\omega_1 \in T$. If the most recent dependent transition is another transition in $\beta$, then *Pb*$(S.t) =$ *Pb*$(S.\omega_1)$ because otherwise $\beta$ would contain only a single transition, and thus either $S$ is empty or last$(S).tid \neq \beta_1.tid$. Thus, $j$ must be *len*$(S)$ in Definition 1.8, and thus either $u \in$ *backtrack*$(s)$, or *backtrack*$(s) =$ *enabled*$(s)$ and thus $\omega_1 \in T$. In either case, we have a contradiction.

Assume that $\beta \leftrightarrow$ *next*(*final*$(S.\omega'), u)$. Because $\beta_1.tid \neq u$, *next*(*final*$(S.\omega), u) =$ *next*(*final*$(S.\omega'), u)$ and

$$\beta \leftrightarrow \mathit{next}(\mathit{final}(S.\omega), u)$$

Thus, by Lemma 6 where $\omega.\beta \in [\omega']$,

$$\mathit{Post}(S.\omega, \mathit{len}(S), u)$$

**Case 7.3.** $\exists i \in$ ***dom***$(\omega) : \omega_i \in T$**.**
Let $\omega = \alpha.\beta.\gamma$ such that

1. $\forall i \in \mathit{dom}(\alpha) : \alpha_i \notin T$
2. $\beta_1 \in T$
3. $\forall i \in \mathit{dom}(\beta) : \beta_i.tid = \beta_1.tid$
4. if *Pb*$(S.\beta_1) <$ *Pb*$(S.\alpha_1)$ then *len*$(\beta) = 1$
5. if *Pb*$(S.\beta_1) =$ *Pb*$(S.\alpha_1)$ and $\gamma$ is nonempty, then $\gamma_1.tid \neq \beta_1.tid$

Assume that $\alpha$ is empty. Then, $\omega_1 \in T$ and by the inductive hypothesis,

$$\mathit{Post}(S.\omega, \mathit{len}(S) + 1, u)$$

Because Requirement 1 of Definition 1.8 of *Post* requires that $i \leq k$ and Requirement 2 of Definition 1.8 of *Post* requires that $j < k$,

$$\mathit{Post}(S.\omega, \mathit{len}(S), u)$$

as required.

Assume that $\alpha$ is nonempty. By Requirement 1 of Definition 1.6 of preemption-bound persistent sets, *Pb*$(S.\beta_1) \leq$ *Pb*$(S.\alpha_1)$.

**Case 7.3a.** $\gamma$ **is nonempty, or** $\gamma$ **is empty and** $\beta_1.tid \notin$ ***enabled***(***final***$(S.\beta)$)**, or** ***Pb***$(S.\beta_1) <$ ***Pb***$(S.\alpha_1)$**.**
Consider the sequence $\omega' = \beta.\alpha.\gamma$, i.e., $\omega$ with $\beta$ moved to the beginning. By Requirements 2 and 3 of Definition 1.6 of preemption-bound persistent sets, $\beta \leftrightarrow \alpha$ and $\forall i \in$ *dom*$(\alpha) : \beta \leftrightarrow$ *next*(*final*$(S.\alpha_1 \ldots \alpha_i), \alpha_i.tid)$. Thus, by Definition 1.1 of a trace, $\omega' \in [\omega]$. By Lemma 3, $\omega'$ is a sequence of transitions from $s$ in $A_{G(Pb,c)}$. By the inductive hypothesis for the recursive call **Explore**$(S.\beta_1)$,

$$\mathit{Post}(S.\omega', \mathit{len}(S) + 1, u)$$

and thus by Lemma 6 where $\beta$ is empty and $\omega' \in [\omega]$,

$$\mathit{Post}(S.\omega, \mathit{len}(S), u)$$

**Case 7.3b.** $\gamma$ **is empty,** $\beta_1.tid \in$ ***enabled***(***final***$(S.\beta)$)**,** ***Pb***$(S.\beta_1) =$ ***Pb***$(S.\alpha_1)$**, and** $u \in$ ***backtrack***$(s)$**.**
Because $\gamma$ is empty, $\omega = \alpha.\beta$. Consider the sequence $\omega' = \beta$. By Requirement 3 of Definition 1.6 of preemption-bound persistent sets, $\beta \leftrightarrow \alpha$ and thus

$$\omega'.\alpha \in [\omega]$$

Because $u \in$ *backtrack*$(s)$, *next*$(s, u) \in T$ and *next*$(s, u) \leftrightarrow \alpha$. If $\beta_1.tid = u$, then *next*(*final*$(S.\omega), u)$ is a transition in *ext*$(s, \beta_1)$ and by Requirement 3 of Definition 1.6 of preemption-bound persistent sets *next*(*final*$(S.\omega), u) \leftrightarrow \alpha$. If $\beta_1.tid \neq u$, then *next*$(s, u) =$ *next*(*final*$(S.\omega), u)$. In either case,

$$\mathit{next}(\mathit{final}(S.\omega), u) \leftrightarrow \alpha$$

Because *Pb*$(S.\beta_1) =$ *Pb*$(S.\alpha_1)$ and all transitions in $\beta$ are by the same thread and thus do not require a preemption, $\omega'$ is a sequence of transitions from $s$ in $A_{G(Pb,c)}$. By the inductive hypothesis for the recursive call **Explore**$(S.\beta_1)$,

$$\mathit{Post}(S.\omega', \mathit{len}(S) + 1, u)$$

and thus by Lemma 6 where $\beta = \alpha$ and $\omega'.\alpha \in \omega$,

$$\mathit{Post}(S.\omega, \mathit{len}(S), u)$$

**Case 7.3c.** $\gamma$ **is empty,** $\beta_1.tid \in$ ***enabled***(***final***$(S.\beta)$)**,** ***Pb***$(S.\beta_1) =$ ***Pb***$(S.\alpha_1)$**, and** $u \notin$ ***backtrack***$(s)$**.**
Because $\gamma$ is empty, $\omega = \alpha.\beta$. Let $\beta'$ be the unique, nonempty sequence of transitions from *final*$(S.\beta)$ such that $\beta.\beta' =$ *ext*$(s, \beta_1)$. Consider the sequence $\omega' = \beta.\beta'.\alpha$. By Requirement 3 of Definition 1.6 of preemption-bound persistent sets, $\beta.\beta' \leftrightarrow \alpha$ and $\forall i \in$ *dom*$(\alpha) : \beta.\beta' \leftrightarrow$ *next*(*final*$(S.\alpha_1 \ldots \alpha_i), \alpha_i.tid)$. Thus, by Lemma 2, $\omega'$ is a sequence of transitions from $s$ in $A_{G(Pb,c)}$. Because $\beta.\beta' \leftrightarrow \alpha$,

$$\omega.\beta' \in [\omega']$$

By the inductive hypothesis for **Explore**$(S.\beta_1)$,

$$\mathit{Post}(S.\omega', \mathit{len}(S) + 1, u)$$

Assume that *next*(*final*$(S.\omega'), u)$ is dependent with a transition in $\beta'$. Then, because $\beta.\beta' \leftrightarrow \alpha$, the most recent dependent transition to *next*(*final*$(S.\omega'), u)$ by $\beta_1.tid$ is in $\beta'$. Thus, by Definition 1.8 of *Post*, either $u \in$ *backtrack*$(s)$ or *backtrack*$(s) =$ *enabled*$(s)$ and thus $\omega_1 \in T$. In either case, we have a contradiction.

Assume that $\beta' \leftrightarrow$ *next*(*final*$(S.\omega'), u)$. Because $\beta_1 \in T$ and $u \notin$ *backtrack*$(s)$, $\beta_1.tid \neq u$. Thus, it must be the case that *next*(*final*$(S.\omega), u) =$ *next*(*final*$(S.\omega'), u)$, and

$$\beta' \leftrightarrow \mathit{next}(\mathit{final}(S.\omega), u)$$

Thus, by Lemma 6 where $\beta = \beta'$ and $\omega.\beta' \in [\omega']$,

$$\mathit{Post}(S.\omega, \mathit{len}(S), u)$$

□

## 2.2 Fair-bounded search

Let $A_{R(Fb,c)}$ be the reduced state space explored by a selective search that explores a fair-bound persistent set in each state. We provide two lemmas to manage the bound, and a theorem stating that a nonempty fair-bound persistent set is local sufficient.

**Lemma 8.** *Let $\alpha$ be a nonempty sequence of transitions from $s = \text{final}(S)$ in $A_{G(\text{Fb},c)}$ and let $t$ be a transition enabled in $s$ such that*

1. $\text{Fb}(S.t) \leq c$
2. *$t$ is not a release operation*
3. $t \leftrightarrow \alpha$

*Then, $t.\alpha$ is a sequence of transitions from $s$ in $A_{G(\text{Fb},c)}$.*

*Proof.* Because $t \leftrightarrow \alpha$, $t.\alpha$ is a sequence of transitions from $s$ in $A_G$. Because $t$ is not a release operation,

$$\forall i \in \mathit{dom}(\alpha) :$$
$$\mathit{enabled}(\mathit{final}(S.t.\alpha_1 \ldots \alpha_i)) \subseteq \mathit{enabled}(\mathit{final}(S.\alpha_1 \ldots \alpha_i))$$

Thus, by Definition 1.9 of the fair bound, the transitions in $\alpha$ cost no more in $S.t.\alpha$ than they do in $S.\alpha$. By Assumption 1, $t$ is within the bound from $s$. Thus, by Definition 1.9 of the fair bound,

$$\mathit{Fb}(S.t.\alpha) \leq c$$

and $t.\alpha$ is a sequence of transitions from $s$ in $A_{G(Fb,c)}$. □

**Lemma 9.** *Let $T$ be a nonempty fair-bound persistent set in a state $s = \text{final}(S)$ in $A_{R(\text{Fb},c)}$ and let $\alpha.t.\gamma$ be a sequence of transitions from $s$ in $A_{G(\text{Fb},c)}$ such that $\alpha$ is nonempty, $\forall i \in \text{dom}(\alpha) : \alpha_i \notin T$, and $t \in T$. Then, $t.\alpha.\gamma$ is a sequence of transitions from $s$ in $A_{G(\text{Fb},c)}$.*

*Proof.* By Requirement 3 of Definition 1.10 of fair-bound persistent sets, $t \leftrightarrow \alpha$. Thus, $t.\alpha.\gamma$ is a sequence of transitions from $s$ in $A_G$. By Requirements 1 and 2 of Definition 1.10 of fair-bound persistent sets, $\mathit{Fb}(S.t) \leq c$ and $t$ is not a release operation. Thus, by Lemma 8,

$$\mathit{Fb}(S.t.\alpha) \leq \mathit{Fb}(S.\alpha)$$

Assume that $\gamma_1$ exceeds the bound from $\mathit{final}(S.t.\alpha)$, yet $t$ does not exceed the bound from $\mathit{final}(S.\alpha)$ and $\gamma_1$ does not exceed the bound from $\mathit{final}(S.\alpha.t)$. Then, $t$ must be a release operation that enables a transition $t'$ such that $t'.tid$ has a lower yield count than $\gamma_1.tid$ has in $\mathit{final}(S.t.\alpha)$, because otherwise $\gamma_1$ would also exceed the bound from $\mathit{final}(S.\alpha)$. Because $t$ is not a release operation, we have a contradiction. Thus,

$$\mathit{Fb}(S.t.\alpha.\gamma_1) \leq c$$

Because $t \leftrightarrow \alpha$, $\mathit{final}(S.t.\alpha.\gamma_1) = \mathit{final}(S.\alpha.t.\gamma_1)$ and thus each transition in $\gamma$ executes from exactly the same state in $S.t.\alpha.\gamma$ as it does in $S.\alpha.t.\gamma$. Thus, by Definition 1.9 of the fair bound,

$$\mathit{Fb}(S.t.\alpha.\gamma) \leq c$$

Thus, $t.\alpha.\gamma$ is a sequence of transitions from $s$ in $A_{G(Fb,c)}$. □

**Algorithm 3** BPOR procedures for fair-bounded search

**procedure Initialize**$(S)$ **begin**
  **if** ($\mathit{len}(S) > \mathit{MAX}$) **then**
    report livelock and exit
  **Backtrack**$(S, \mathit{len}(S), u)$ where $u$ is a lowest cost enabled thread in $\mathit{final}(S)$
**procedure Backtrack**$(S, i, u)$ **begin**
  **if** ($u \in \mathit{enabled}(\mathit{pre}(S,i))$ and $\mathit{next}(\mathit{pre}(S,i), u)$ is not a release operation) **then**
    add $u$ to $\mathit{backtrack}(\mathit{pre}(S,i))$
  **else**
    $\mathit{backtrack}(\mathit{pre}(S,i)) = \mathit{enabled}(\mathit{pre}(S,i))$

**Theorem 10.** *If $T$ is a nonempty fair-bound persistent set in a state $s$ in $A_{R(\text{Fb},c)}$, then $T$ is local sufficient in $s$.*

*Proof.* Let $s$ be a state in $A_{R(Fb,c)}$ and let $l$ be a local state reachable from $s$ in $A_{G(Fb,c)}$ via a nonempty sequence $\omega$ of transitions.

**Case 10.1.** $\forall i \in \boldsymbol{dom}(\omega) : \omega_i \notin T$.
Let $t$ be any transition in $T$. Consider the sequence $\omega' = t.\omega$. By Requirement 3 of Definition 1.10 of fair-bound persistent sets, $t \leftrightarrow \omega$. Thus, $\omega.t \in [\omega']$, and $\omega \in \mathit{Prefix}([\omega'])$. By Requirements 1 and 2 of Definition 1.10 of fair-bound persistent sets, $\mathit{Fb}(S.t) \leq c$ and $t$ is not a release operation. Thus, by Lemma 8, $t.\omega$ is a sequence of transitions from $s$ in $A_{G(Fb,c)}$ and $T$ is local sufficient in $s$.

**Case 10.2.** $\exists i \in \boldsymbol{dom}(\omega) : \omega_i \in T$.
Let $\omega = \alpha.t.\gamma$ such that $\forall i \in \mathit{dom}(\alpha) : \alpha_i \notin T$ and $t \in T$. Assume that $\alpha$ is empty. Then, $T$ is local sufficient in $s$ because $\omega_1 \in T$ and $l$ is reachable via $\omega$.

Assume that $\alpha$ is nonempty. Consider the sequence $\omega' = t.\alpha.\gamma$, i.e., $\omega$ with $t$ moved to the first position. By Requirement 3 of Definition 1.10 of fair-bound persistent sets, $t \leftrightarrow \alpha$. Thus, $\omega' \in [\omega]$ and $\omega \in \mathit{Prefix}([\omega'])$. By Lemma 9, $t.\alpha.\gamma$ is a sequence of transitions from $s$ in $A_{G(Fb,c)}$, and $T$ is local sufficient in $s$. □

**Lemma 11.** *Whenever Algorithm 1 backtracks a state $s = \text{final}(S)$, the set $T$ of transitions explored from $s$ is fair-bound persistent in $s$, provided that postcondition PC holds for every recursive call **Explore**$(S.t)$ for all $t \in T$.*

*Proof.* Let $T = next(s, u) \mid u \in backtrack(s)$. Show that if $T$ violates any requirement in Definition 1.10 of fair-bound persistent sets, then we have a contradiction.

**Case 11.1.** $T$ **violates Requirement 1.**
Proceed by contradiction. Assume that for some $t \in T$, $Fb(S.t) > c$. By Line 12 in Algorithm 1, the search explores only transitions that do not exceed the bound from $s$. Thus, we have a contradiction.

**Case 11.2.** $T$ **violates Requirement 2.**
Proceed by contradiction. Assume that there exists a transition $t \in T$ such that $t$ is a release operation and a thread $u \in enabled(s)$ such that $next(s, u) \notin T$. Because $t$ is a release operation Line 9 in Algorithm 3 must add it to $backtrack(s)$. Because $u \in enabled(s)$, Line 9 also adds $u$ to $backtrack(s)$ and thus $next(s, u) \in T$ and we have a contradiction.

**Case 11.3.** $T$ **violates Requirement 3.**
Proceed by contradiction. Assume that there exists a nonempty sequence $\alpha$ of transitions from $s$ in $A_{G(Fb,c)}$ such that $\forall i \in dom(\alpha) : \alpha_i \notin T$, and a transition $t \in T$ such that

1. $Fb(S.t) \leq c$
2. $t$ is not a release operation
3. $t$ is dependent with $\text{last}(\alpha)$

Let $n = len(\alpha)$ and let $\omega = \alpha_1 \ldots \alpha_{n-1}$, i.e., $\alpha$ with its last transition removed. Let there be no prefixes of $\alpha$ that also meet the criteria above, and thus

3. $t \leftrightarrow \omega$

Let $u = \text{last}(\alpha).tid$. Assume that $t.tid = u$. Because $t \leftrightarrow \omega$,

$$t = next(final(S), u) = next(final(S.\omega), u) = \text{last}(\alpha)$$

Thus, $\text{last}(\alpha) \in T$ and we have a contradiction.

Assume that $t.tid \neq u$. Consider the postcondition

$$Post(S.t.\omega, len(S) + 1, u)$$

for the recursive call **Explore**$(S.t)$. By Lemma 8, $t.\omega$ is a sequence of transitions from $s$ in $A_{G(Fb,c)}$. Because $t \leftrightarrow \omega$, $t$ is the most recent transition by $t.tid$ that is dependent with $next(final(S.t.\omega), u)$. Thus, by Definition 1.12 of *Post*, $u \in backtrack(s)$ and thus a transition in $\alpha$ must be in $T$ so we have a contradiction.

□

Thus, if postcondition *PC* holds in each state $s$ explored by Algorithm 1 with the **Backtrack** procedure from Algorithm 3, then the set of transitions explored from $s$ is fair-bound persistent in $s$. Next, we prove that postcondition *PC* holds in each state $s$ explored by Algorithm 1. First, we prove a lemma to simplify the inductive step.

**Lemma 12.** *Let* $s = \text{final}(S)$ *be a state in* $A_{R(\text{Fb},c)}$*, let* $\omega$ *and* $\omega'$ *be nonempty sequences of transitions from* $s$ *in* $A_{G(\text{Fb},c)}$*, and let* $u$ *be a thread such that*

1. $\exists\beta : \omega.\beta \in [\omega']$ ***and*** $\beta \leftrightarrow \text{next}(\text{final}(S.\omega), u)$*, or*
2. $\exists\beta : \omega'.\beta \in [\omega]$ ***and*** $\beta \leftrightarrow \text{next}(\text{final}(S.\omega), u)$

*Then,* $\text{Post}(S.\omega', \text{len}(S) + 1, u) \implies \text{Post}(S.\omega, \text{len}(S), u)$.

*Proof.* Because $\beta \leftrightarrow next(final(S.\omega), u)$,

$$next(final(S.\omega), u) = next(final(S.\omega'), u)$$

Assume that in Definition 1.12 of $Post(S.\omega, len(S), u)$ for some thread $v$, $i > k$. Then, *Post* does not require any backtrack points for $v$.

Assume that for some thread $v$ in Definition 1.12 of $Post(S.\omega, len(S), u)$, $i \leq k$. Then, $i$ is the same for thread $v$ in $Post(S.\omega', len(S), u)$ because $\beta \leftrightarrow next(final(S.\omega), u)$. Because $i \leq len(S)$, the yield counts for all threads are the same in $pre(S, i)$, as well. Thus, by Definition 1.12 of *Post*,

$$Post(S.\omega, len(S), u) \text{ iff } Post(S.\omega', len(S), u) \quad (4)$$

Because Definition 1.12 of *Post* requires that $i$ be less than or equal to $k$,

$$Post(S.\omega', len(S) + 1, u) \implies Post(S.\omega', len(S), u)$$

Thus, by Equation 4,

$$Post(S.\omega', len(S) + 1, u) \implies Post(S.\omega, len(S), u)$$

□

**Theorem 13.** *Whenever a state* $s = \text{final}(S)$ *is backtracked during the search performed by Algorithm 1, the postcondition* Post *for* ***Explore***$(S)$ *is satisfied, and the set* $T$ *of transitions explored from* $s$ *is fair-bound persistent in* $s$*.*

*Proof.* The proof is by induction on the order in which states are backtracked.

**Base case.**
If the stack depth exceeds *MAX*, then the search terminates and reports a livelock. Thus, the state space that the search may explore without reporting a livelock is a subset of the cyclic state space. Assume that the test does not contain a livelock. Because the search is performed in depth-first order, and the fair bound always provides a zero-cost transition, the first backtracked state must be a deadlock state in which no transition is enabled. Thus, the postcondition for the first backtracked state is

$$\forall u : Post(S, len(S), u)$$

and is directly established by Lines 4-7 in Algorithm 1.

**Inductive case.**

Assume that each call to **Explore**$(S.t)$ satisfies its postcondition. By Lemma 11, $T$ is fair-bound persistent in $s$. Show that **Explore**$(S)$ satisfies its postcondition for any sequence $\omega$ of transitions from $s$ in $A_{G(Fb,c)}$ and for any thread $u$. If $\omega$ is empty then the postcondition is directly established by Lines 4-7 in Algorithm 1, so assume that $\omega$ is nonempty.

**Case 13.1.** $\forall i \in \textbf{\textit{dom}}(\omega) : \omega_i \notin T$ **and** $u \in \textbf{\textit{backtrack}}(s)$**.**
Because $u \in \textit{backtrack}(s)$, $\textit{next}(s, u) \in T$. Thus, by Requirement 3 of Definition 1.10 of fair-bound persistent sets, $\textit{next}(s, u) \leftrightarrow \omega$, and thus

$$\textit{next}(\textit{final}(S.\omega), u) = \textit{next}(s, u)$$

Thus, $\textit{next}(\textit{final}(S.\omega), u) \leftrightarrow \omega$, and thus $\textit{Post}(S.\omega, \textit{len}(S), u)$ iff $\textit{Post}(S, \textit{len}(S), u)$. The latter is directly established by Lines 4-7 in Algorithm 1.

**Case 13.2.** $\forall i \in \textbf{\textit{dom}}(\omega) : \omega_i \notin T$ **and** $u \notin \textbf{\textit{backtrack}}(s)$**.**
Let $t$ be any transition in $T$. Consider the sequence $\omega' = t.\omega$. By Definition 1.10 of fair-bound persistent sets, $\textit{Fb}(S.t) \leq c$ and $t \leftrightarrow \omega$. Because $\omega$ is nonempty and $\omega_1 \notin T$, by Requirement 2 of Definition 1.10 of fair-bound persistent sets, $t$ is not a release operation. Thus, by Lemma 8, $\omega'$ is a sequence of transitions from $s$ in $A_{G(Fb,c)}$. Because $t \leftrightarrow \omega$,

$$\omega.t \in [\omega']$$

By the inductive hypothesis for **Explore**$(S.t)$,

$$\textit{Post}(S.\omega', \textit{len}(S) + 1, u)$$

If $t$ is dependent with $\textit{next}(\textit{final}(S.\omega'), u)$, then because $t \leftrightarrow \omega$, $\omega'_1$ must be the most recent dependent transition to $\textit{next}(\textit{final}(S.\omega'), u)$ by $t.tid$. Thus, by Definition 1.12 of *Post*, either $u \in \textit{backtrack}(s)$ or $\textit{backtrack}(s) = \textit{enabled}(s)$, in which case $\omega_1 \in T$. In either case, we have a contradiction. Thus, $t \leftrightarrow \textit{next}(\textit{final}(S.\omega'), u)$ and additionally, $t \leftrightarrow \textit{next}(\textit{final}(S.\omega), u)$. Thus, by Lemma 12 where $\beta = t$ and $\omega.t \in [\omega']$,

$$\textit{Post}(S.\omega, \textit{len}(S), u)$$

**Case 13.3.** $\exists i \in \textbf{\textit{dom}}(\omega) : \omega_i \in T$**.**
Let $\omega = \alpha.t.\gamma$ such that

1. $\forall i \in \textit{dom}(\alpha) : \alpha_i \notin T$
2. $t \in T$

Assume that $\alpha$ is empty. Then, $\omega_1 \in T$, and by the inductive hypothesis

$$\textit{Post}(S.\omega, \textit{len}(S) + 1, u)$$

Thus, because Definition 1.12 of *Post* requires that $i \leq k$,

$$\textit{Post}(S.\omega, \textit{len}(S), u)$$

as required.

Assume that $\alpha$ is nonempty. Consider the sequence $\omega' = t.\alpha.\gamma$, i.e., $\omega$ with $t$ moved to the beginning. By Definition 1.10 of fair-bound persistent sets, $\textit{Fb}(S.t) \leq c$ and $t \leftrightarrow \alpha$. Thus, by Definition 1.1 of a trace,

$$\omega' \in [\omega]$$

By Lemma 9, $\omega'$ is a sequence of transitions from $s$ in $A_{G(Fb,c)}$. By the inductive hypothesis for the recursive call **Explore**$(S.t)$,

$$\textit{Post}(S.\omega', \textit{len}(S) + 1, u)$$

and thus by Lemma 12 where $\beta$ is empty and $\omega' \in [\omega]$,

$$\textit{Post}(S.\omega, \textit{len}(S), u)$$

□

## References


[1] GODEFROID, P. *Partial-Order Methods for the Verification of Concurrent Systems: An Approach to the State-Explosion Problem.* Springer-Verlag, 1996.

[2] MUSUVATHI, M., AND QADEER, S. Partial-order reduction for context-bounded state exploration. Tech. Rep. MSR-TR-2007-12, Microsoft Research, 2007.