

Appendix D: Instruction Graph Proofs

1 cfgvalid

cfg cfgvalid means that the configuration cfg is a valid configuration.

$$\frac{\mathbf{P}(v, vs) \text{ valid} \quad \mathbf{V}(n, c) \in v :: vs}{(n, v :: vs, I, O) \text{ cfgvalid}} (\text{cfg}_{\text{cfgvalid}})$$

2 Progress

Theorem 1. *If cfg cfgvalid , then either*

1. cfg terminated
2. cfg waiting
3. $\exists \text{cfg}' \text{ s.t. } \text{cfg} \mapsto \text{cfg}'$

2.1 Proof of Progress

We proceed by case analysis on the judgment cfg cfgvalid . There is only one rule that concludes cfg cfgvalid :

$$\frac{\mathbf{P}(v, vs) \text{ valid} \quad \mathbf{V}(n, c) \in v :: vs}{(n, v :: vs, I, O) \text{ cfgvalid}}$$

So we know cfg is of the form $(n, v :: vs, I, O)$ and $\mathbf{V}(n, c) \in v :: vs$. We continue by structural induction on c , which is of the sort **Content**.

Case c is **do** a **then** n' :

Then by the rule

$$\frac{\mathbf{V}(n, \text{do } a \text{ then } n') \in vs}{(n, vs, I, O) \mapsto (n', vs, I, a :: O)}$$

we can conclude $(n, v :: vs, I, O) \mapsto (n', v :: vs, I, a :: O)$.

Case c is **do** a **until** end **then** n' :

We use structural induction on I .

Subcase I is $[]$:

Then by the rule

$$\frac{\mathbf{V}(n, \text{do } a \text{ until } end \text{ then } n') \in vs}{(n, vs, [], O) \text{ waiting}}$$

we can conclude $(n, v :: vs, I, O) \text{ waiting}$.

Subcase I is $true :: I'$:

Then by the rule

$$\frac{\mathbf{V}(n, \text{do } a \text{ until } end \text{ then } n') \in vs}{(n, vs, true :: I, O) \mapsto (n', vs, I, a :: O)}$$

we can conclude $(n, v :: vs, I, O) \mapsto (n', v :: vs, I', a :: O)$.

Subcase I is $false :: I'$:

Then by the rule

$$\frac{\mathbf{V}(n, \text{do } a \text{ until } cnd \text{ then } n') \in vs}{(n, vs, false :: I, O) \mapsto (n, vs, I, a :: O)}$$

we can conclude $(n, v :: vs, I, O) \mapsto (n, v :: vs, I', a :: O)$.

Case c is **if** cnd **then** n' **else** n'' :

We use structural induction on I .

Subcase I is $[]$:

Then by the rule

$$\frac{\mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in vs}{(n, vs, [], O) \text{ waiting}}$$

we can conclude $(n, v :: vs, I, O) \text{ waiting}$.

Subcase I is $true :: I'$:

Then by the rule

$$\frac{\mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in vs}{(n, vs, true :: I, O) \mapsto (n', vs, I, O)}$$

we can conclude $(n, v :: vs, I, O) \mapsto (n', v :: vs, I', O)$.

Subcase I is $false :: I'$:

Then by the rule

$$\frac{\mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in vs}{(n, vs, false :: I, O) \mapsto (n'', vs, I, O)}$$

we can conclude $(n, v :: vs, I, O) \mapsto (n'', v :: vs, I', O)$.

Case c is **goto** n' :

Then by the rule

$$\frac{\mathbf{V}(n, \text{goto } n') \in vs}{(n, vs, I, O) \mapsto (n', vs, I, O)}$$

we can conclude $(n, v :: vs, I, O) \mapsto (n', v :: vs, I, O)$.

Case c is **end**:

Then by the rule

$$\frac{\mathbf{V}(n, \text{end}) \in vs}{(n, vs, I, O) \text{ terminated}}$$

we can conclude $(n, v :: vs, I, O) \text{ terminated}$.

3 Preservation

Theorem 2. *If cfg cfgvalid and $cfg \mapsto cfg'$ then cfg' cfgvalid .*

3.1 Lemma 1

If (vs, U_v, n, U) **connected** then $\forall n' \in U . \exists U'_v, U'$ such that U' is nonempty and (vs, U'_v, n', U') **connected**.

Proof: We proceed by rule induction on (vs, U_v, n, U) **connected**.

$$\text{Case } \frac{(vs, U) \text{ defined} \quad U_v \subseteq U \quad n \in U_v}{(vs, U_v, n, \emptyset) \text{ connected}}$$

Then U is the empty set so the lemma is vacuously true.

$$\text{Case } \frac{(vs, U) \text{ defined} \quad U_v \subseteq U \quad \mathbf{V}(n, \text{end}) \in vs \quad n \notin U_v}{(vs, U_v, n, \{n\}) \text{ connected}}$$

Then U contains exactly n . But we already have $(vs, U_v, n, \{n\})$ **connected** so the lemma is satisfied.

$$\text{Case } \frac{\mathbf{V}(n, \text{do } a \text{ then } n') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup \{n\}) \text{ connected}}$$

Then by the inductive hypothesis, since $(vs, U_v \cup \{n\}, n', U)$ **connected** we know the lemma is satisfied for all $n' \in U$. All that's left is to show it is satisfied for n , but we have $(vs, U_v, n, U \cup \{n\})$ **connected**.

$$\text{Case } \frac{\mathbf{V}(n, \text{do } a \text{ until } cnd \text{ then } n') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup \{n\}) \text{ connected}}$$

Analogous to the case above.

$$\text{Case } \frac{\mathbf{V}(n, \text{goto } n') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup \{n\}) \text{ connected}}$$

Analogous to the case above.

$$\text{Case } \frac{\mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad (vs, U_v \cup U \cup \{n\}, n'', U') \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup U' \cup \{n\}) \text{ connected}}$$

Then by the inductive hypothesis, since $(vs, U_v \cup \{n\}, n', U)$ **connected** and $(vs, U_v \cup U \cup \{n\}, n'', U')$ **connected** we know the lemma is satisfied for all $n' \in U \cup U'$. All that's left is to show it is satisfied for n , but we have $(vs, U_v, n, U \cup U' \cup \{n\})$ **connected**.

3.2 Lemma 2

If $\mathbf{V}(n, c) \in vs$ and (vs, U) **defined** then $n \in U$.

Proof: We proceed by rule induction on (vs, U) **defined**.

Case $\frac{}{(nil, \{ \}) \text{ defined}}$

But vs is nil , so the lemma is vacuously true.

Case $\frac{(vs, U) \text{ defined} \quad n' \notin U}{(\mathbf{V}(n', c) :: vs, U \cup \{n'\}) \text{ defined}}$

(n is replaced by n' in this statement of the rule to avoid ambiguity.)

The vertex in question, $\mathbf{V}(n, c)$, is either $\mathbf{V}(n', c)$ or $\in vs$. If it is the former, we are done since $n' \in U \cup \{n'\}$. If it is the latter, then by the inductive hypothesis, since (vs, U) **defined** we know $n \in U$ so $n \in U \cup \{n'\}$.

3.3 Lemma 3

If (vs, U) **defined** and $n \in U$ then $\mathbf{V}(n, c) \in vs$ for some c .

Proof: We proceed by rule induction on (vs, U) **defined**.

Case $\frac{}{(nil, \{ \}) \text{ defined}}$

But U is empty so the lemma is vacuously true.

Case $\frac{(vs, U) \text{ defined} \quad n' \notin U}{(\mathbf{V}(n', c) :: vs, U \cup \{n'\}) \text{ defined}}$

(n is replaced by n' in this statement of the rule to avoid ambiguity.)

n is either $\in U$ or is n' . We know it's not both since $n' \notin U$. If it's the former, then by the inductive hypothesis, since (vs, U) **defined**, we know there is a $\mathbf{V}(n, c) \in vs$, which is also in $\mathbf{V}(n', c) :: vs$. If it's the latter, then clearly $\mathbf{V}(n', c) \in \mathbf{V}(n', c) :: vs$.

3.4 Lemma 4

If (vs, U) **defined** and $\mathbf{V}(n, c) \in vs$ and $\mathbf{V}(n, c') \in vs$ then $c = c'$.

Proof: We proceed by induction on (vs, U) **defined**.

Case $\frac{}{(nil, \{ \}) \text{ defined}}$

But vs is nil so the lemma is vacuously true.

Case $\frac{(vs, U) \text{ defined} \quad n' \notin U}{(\mathbf{V}(n', c) :: vs, U \cup \{n'\}) \text{ defined}}$

(n is replaced by n' in this statement of the rule to avoid ambiguity.)

Suppose that $\mathbf{V}(n, c)$ and $\mathbf{V}(n, c')$ are both $\mathbf{V}(n', c)$. Then clearly $c = c'$.

Suppose instead that one of the two is $\mathbf{V}(n', c)$ (so $n = n'$) and the other is $\in vs$. Then by Lemma 2, $n \in U$. But $n' \notin U$, contradiction.

Suppose lastly that both are $\in vs$. Then by the inductive hypothesis, $c = c'$.

3.5 Lemma 5

If (vs, U_v, n, U) **connected**, then $\forall n' \in U_v . \exists c'$ such that $\mathbf{V}(n', c') \in vs$.

Proof: We proceed by rule induction on (vs, U_v, n, U) **connected**.

$$\text{Case } \frac{(vs, U) \text{ defined} \quad U_v \subseteq U \quad n \in U_v}{(vs, U_v, n, \emptyset) \text{ connected}}$$

Since $U_v \subseteq U$, it suffices to check this is true for every element in U . Since we have (vs, U) **defined**, we have this property by Lemma 3.

$$\text{Case } \frac{(vs, U) \text{ defined} \quad U_v \subseteq U \quad \mathbf{V}(n, \text{end}) \in vs \quad n \notin U_v}{(vs, U_v, n, \{n\}) \text{ connected}}$$

Analogous to the case above.

$$\text{Case } \frac{\mathbf{V}(n, \text{do } a \text{ then } n') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup \{n\}) \text{ connected}}$$

Then by the inductive hypothesis, since $(vs, U_v \cup \{n\}, n', U)$ **connected** we know the lemma is satisfied for all $n' \in U_v \cup \{n\}$. Since $U_v \subseteq U_v \cup \{n\}$, this is also true for all $n' \in U_v$.

$$\text{Case } \frac{\mathbf{V}(n, \text{do } a \text{ until } cnd \text{ then } n') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup \{n\}) \text{ connected}}$$

Analogous to the case above.

$$\text{Case } \frac{\mathbf{V}(n, \text{goto } n') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup \{n\}) \text{ connected}}$$

Analogous to the case above.

$$\text{Case } \frac{\mathbf{V}(n, \text{ if } cnd \text{ then } n' \text{ else } n'') \in vs \quad (vs, U_v \cup \{n\}, n', U) \text{ connected} \quad (vs, U_v \cup U \cup \{n\}, n'', U') \text{ connected} \quad n \notin U_v}{(vs, U_v, n, U \cup U' \cup \{n\}) \text{ connected}}$$

Analogous to the case above.

3.6 Proof of Preservation

Recall that we stated preservation as “If cfg **cfgvalid** and $cfg \mapsto cfg'$ then cfg' **cfgvalid**.”

We begin by case analyzing on cfg **cfgvalid** in order to conclude some important facts.

The only case is

$$\frac{\mathbf{P}(v, vs) \text{ valid} \quad \mathbf{V}(n, c) \in v :: vs}{(n, v :: vs, I, O) \text{ cfgvalid}}$$

so we can conclude:

$$cfg = (n, v :: vs, I, O) \tag{1}$$

$$\mathbf{P}(v, vs) \text{ valid} \tag{2}$$

$$\mathbf{V}(n, c) \in v :: vs \tag{3}$$

We case analyze on $\mathbf{P}(v, vs) \text{ valid}$. The only case is

$$\frac{(\mathbf{V}(s, c_s) :: vs, U) \text{ defined} \quad (\mathbf{V}(s, c_s) :: vs, \emptyset, s, U) \text{ connected}}{\mathbf{P}(\mathbf{V}(s, c_s), vs) \text{ valid}}$$

so we can conclude:

$$v = \mathbf{V}(s, c_s) \quad (4)$$

$$(\mathbf{V}(s, c_s) :: vs, U) \text{ defined} \quad (5)$$

$$(\mathbf{V}(s, c_s) :: vs, \emptyset, s, U) \text{ connected} \quad (6)$$

From (3), (4), and (5), using Lemma 2 we can conclude

$$n \in U \quad (7)$$

From (6) and (7), using Lemma 1 we can conclude $\exists U'_v, U'$ where U' is not empty such that

$$(v :: vs, U'_v, n, U') \text{ connected} \quad (8)$$

We continue by case analyzing on $cfg \mapsto cfg'$.

$$\text{Case } \frac{\mathbf{V}(n, \text{do } a \text{ then } n') \in vs'}{(n, vs', I, O) \mapsto (n', vs', I, a :: O)}$$

So taking into account (1), cfg is of the form $(n, v :: vs, I, O)$ and cfg' is of the form $(n', v :: vs, I, a :: O)$. We do another case analysis on (8).

$$\text{Subcase } \frac{(v :: vs, U) \text{ defined} \quad U'_v \subseteq U \quad n \in U'_v}{(v :: vs, U'_v, n, \emptyset) \text{ connected}}$$

But U' is not empty, contradiction.

All but one of all of the other subcases have a premise of the form $\mathbf{V}(n, c') \in v :: vs$ where c' is not **do a then** n' . This is a contradiction by Lemma 4 since we have $\mathbf{V}(n, \mathbf{do\ a\ then\ } n') \in v :: vs$.

Thus the only subcase left is

$$\frac{\mathbf{V}(n, \mathbf{do\ a\ then\ } n') \in vs \quad (v :: vs, U'_v \cup \{n\}, n', U'') \text{ connected} \quad n \notin U'_v}{(v :: vs, U'_v, n, U'' \cup \{n\}) \text{ connected}}$$

If we let U''_v be $U'_v \cup \{n\}$, then we can case analyze on $(v :: vs, U''_v, n', U'') \text{ connected}$:

$$\text{Sub-subcase } \frac{(v :: vs, U) \text{ defined} \quad U''_v \subseteq U \quad n' \in U''_v}{(v :: vs, U''_v, n', \emptyset) \text{ connected}}$$

Then since we have $n' \in U''_v$, by Lemma 5, $\exists c'$ such that $\mathbf{V}(n', c') \in v :: vs$.

In any of the other sub-subcases, one of the premises gives us $\mathbf{V}(n', c') \in v :: vs$ for some c' .

So regardless of the sub-subcase, we have $\mathbf{V}(n', c') \in v :: vs$. Combining with (2), we can conclude $(n', v :: vs, I, a :: O) \text{ cfgvalid}$ as desired.

$$\text{Case } \frac{\mathbf{V}(n, \mathbf{do\ a\ until\ } cnd \text{ then } n') \in vs}{(n, vs, true :: I, O) \mapsto (n', vs, I, a :: O)}$$

So taking into account (1), cfg is of the form $(n, v :: vs, true :: I', O)$ and cfg' is of the form $(n', v :: vs, I', a :: O)$. The rest of the proof for this case is analogous to above except c is **do a until** cnd **then** n' and the end conclusion is that $(n', v :: vs, I', a :: O) \text{ cfgvalid}$.

$$\text{Case } \frac{\mathbf{V}(n, \mathbf{do\ a\ until\ } cnd \text{ then } n') \in vs}{(n, vs, false :: I, O) \mapsto (n, vs, I, a :: O)}$$

So taking into account (1), cfg is of the form $(n, v :: vs, false :: I', O)$ and cfg' is of the form $(n, v :: vs, I', a :: O)$. The rest of the proof for this case is analogous to above except c is **do** a **until** cnd **then** n' and the end conclusion is that $(n, v :: vs, I', a :: O)$ **cfgvalid**.

$$\text{Case } \frac{\mathbf{V}(n, \text{goto } n') \in vs}{(n, vs, I, O) \mapsto (n', vs, I, O)}$$

So taking into account (1), cfg is of the form $(n, v :: vs, I, O)$ and cfg' is of the form $(n', v :: vs, I, O)$. The rest of the proof for this case is analogous to above except c is **goto** n' and the end conclusion is that $(n', v :: vs, I, O)$ **cfgvalid**.

$$\text{Case } \frac{\mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in vs}{(n, vs, true :: I, O) \mapsto (n', vs, I, O)}$$

So taking into account (1), cfg is of the form $(n, v :: vs, true :: I', O)$ and cfg' is of the form $(n', v :: vs, I', O)$. The rest of the proof for this case is analogous to above except c is **if** cnd **then** n' **else** n'' and the end conclusion is that $(n', v :: vs, I', a :: O)$ **cfgvalid**.

$$\text{Case } \frac{\mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in vs}{(n, vs, false :: I, O) \mapsto (n'', vs, I, O)}$$

So taking into account (1), cfg is of the form $(n, v :: vs, false :: I', O)$ and cfg' is of the form $(n'', v :: vs, I', O)$. We do another case analysis on (8).

$$\text{Subcase } \frac{(v :: vs, U) \text{ defined} \quad U'_v \subseteq U \quad n \in U'_v}{(v :: vs, U'_v, n, \emptyset) \text{ connected}}$$

But U' is not empty, contradiction.

All but one of all of the other subcases have a premise of the form $\mathbf{V}(n, c') \in v :: vs$ where c' is not **if** cnd **then** n' **else** n'' . This is a contradiction by Lemma 4 since we have $\mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in v :: vs$.

Thus the only subcase left is

$$\frac{\begin{array}{l} \mathbf{V}(n, \text{if } cnd \text{ then } n' \text{ else } n'') \in v :: vs \\ (v :: vs, U'_v \cup \{n\}, n', U) \text{ connected} \\ (v :: vs, U'_v \cup U \cup \{n\}, n'', U'') \text{ connected} \quad n \notin U'_v \end{array}}{(v :: vs, U'_v, n, U \cup U'' \cup \{n\}) \text{ connected}}$$

If we let U''_v be $U'_v \cup U \cup \{n\}$, then we can case analyze on $(v :: vs, U''_v, n'', U'') \text{ connected}$:

$$\text{Sub-subcase } \frac{(v :: vs, U) \text{ defined} \quad U''_v \subseteq U \quad n'' \in U''_v}{(v :: vs, U''_v, n'', \emptyset) \text{ connected}}$$

Then since we have $n'' \in U''_v$, by Lemma 5, $\exists c'$ such that $\mathbf{V}(n'', c') \in v :: vs$.

In any of the other sub-subcases, one of the premises gives us $\mathbf{V}(n'', c') \in v :: vs$ for some c' .

So regardless of the sub-subcase, we have $\mathbf{V}(n'', c') \in v :: vs$. Combining with (2), we can conclude $(n'', v :: vs, I', O) \text{ cfgvalid}$ as desired.