

Chapter 7

Proofs of Compositionality and Program-Based Satisfaction

7.1 Introduction

We now prove the theorems of compositionality and program-based satisfaction presented in chapter 6. We first prove these theorems for the linear case, and then outline the proofs for the mesh and partial-service mesh cases.

7.2 Compositionality for linear case

We start by restating Theorem 6.1, which is to be proved:

Let M and N be distinct systems, and U , V and W be distinct services. Let M satisfy U above and V below. Let N satisfy V above and W below. Let MN be the composite system $\{M, N\}$ with xc events corresponding to events of V hidden. Then MN satisfies U above and W below. Furthermore, this holds even if the services were to have internal nondeterminism.

The above theorem does not say anything about relationship between the composite system MN and the enclosed service V . In fact, we expect the executions of MN to satisfy V provided the environment satisfies the encapsulating services U and W . Furthermore, exploiting this is essential to establishing the theorem. We next formalize the notion of satisfying enclosed services.

Definition (external-internal service satisfaction) Let M be a system, U and V be services, and I be a set of services, such that U , V , and the services in I are distinct. M **satisfies U above, V below, and I internally** if

- M satisfies U above and V below.
- M encloses I .
- For every finite execution x of M , if x is safe wrt U and V then x is safe wrt I .
- For every execution x of M , if x is safe wrt $\{U, V\}$ and complete wrt $\{M, V\}$, then x is complete wrt I .

Furthermore, this definition holds if the services were to have internal nondeterminism. ■

If I is empty, the above definition reduces to the definition of M satisfying U above and V below.

The following lemma generalizes theorem 6.1.

Lemma 7.1 Let M and N be distinct systems, and U , V and W be distinct services. Let N satisfy U above and V below. Let M satisfy V above and W below. Then the composite system $\{M, N\}$, with xc events corresponding to events of V hidden, satisfies U above, W below, and V internally. Furthermore, this holds even if the services were to have internal nondeterminism. ■

It suffices to prove Lemma 7.1. We first restate it, making explicit the desired signature, safety and progress conditions:

Lemma 7.2 Let systems M and N and services U , V and W be such that N satisfies V above W below and M satisfies U above V below. Let MN denote the composite system of M and N , with xc events corresponding to events of V hidden. Then the following hold:

- **Signature:** MN is encapsulated by U above W below, and has V as a fully-internal service.
- **Safety:** For every finite execution x of MN such that x is safe wrt $\{U, W\}$:
 - x is fault-free
 - x is safe wrt V .
 - For every call e of an input of MN : if $x \circ \langle e \rangle$ is safe wrt $\{U, W\}$, then e is enabled in the last state of x and its execution is fault-free and nonblocking. If e 's execution returns a value, say g , then $x \circ \langle e, g \rangle$ is safe wrt $\{U, W\}$.
 - For every execution y of MN such that y is x extended by an internal or output transition: y is fault-free and safe wrt $\{U, W\}$.
- **Progress:** For every execution x of MN such that x is safe wrt $\{U, W\}$: if x is complete wrt $\{MN, W\}$, then x is complete wrt $\{U, V\}$.

Furthermore, this holds even if the services were to have internal nondeterminism. ■

Relating executions of MN , M , and N

We first state some lemmas relating the executions of MN to those of M and N . Lemmas 7.3 and 7.4 below follow from Theorem 4.1, and Lemma 7.5 follows from Theorem 5.4.

Lemma 7.3 If x is a fault-free execution of MN then $x.M$ is a fault-free execution of M and $x.N$ is a fault-free execution of N . ■

Lemma 7.4 If x is a complete execution of MN then $x.M$ is a complete execution of M and $x.N$ is a complete execution of N . ■

Lemma 7.5 Let x be a fault-free execution of MN and Y be a safety or progress assertion with no free variables of N (or M). Then x satisfies Y iff $x.M$ (or $x.N$) satisfies Y . ■

Lemmas 7.6 and 7.7 hold from the definition of MN and the signature conditions of M offers U uses V and N offers V uses W .

Lemma 7.6 MN satisfies the signature condition in Lemma 7.2. ■

Lemma 7.7 For any sequence x of states, input calls, output calls, and event returns of MN :

- a. x is safe (complete) wrt U iff $x.M$ is safe (complete) wrt U .

- b. x is safe (complete) wrt V iff $x.M$ is safe (complete) wrt V .
- c. x is safe (complete) wrt V iff $x.N$ is safe (complete) wrt V .
- d. x is safe (complete) wrt W iff $x.N$ is safe (complete) wrt W .

■

Establishing safety condition for MN

Lemma 7.8 For any finite execution x of MN that is safe wrt $\{U, W\}$:

- a. x is fault free
- b. x is safe wrt V .

Proof The proof is by induction on the number of transitions in x . The base case is x with no transitions, i.e., x consists only of an initial state; clearly, x is fault free and vacuously safe wrt V .

For the induction step, we assume the lemma holds for a finite execution x . Let $y = x \circ \langle \bar{p} \rangle$ be an execution of MN such that y is safe wrt $\{U, W\}$. Since x is a prefix of y , x is safe wrt $\{U, W\}$. From the induction hypothesis, x is fault free and safe wrt V . Hence $x.M$ and $x.N$ are fault-free executions (Lemma 7.3). Thus

- a1. $x.M$ is a fault-free execution of M that is safe wrt $\{U, V\}$
- a2. $x.N$ is a fault-free execution of N that is safe wrt V and W .

The transitions of MN consist of internal transitions of M , internal transitions of N , composite internal transitions between M and N , input transitions of M driven by U dnw events, output transitions of M that call U upw events, input transitions of N driven by W upw events, and output transitions of N that call W dnw events.

Suppose \bar{p} is derived from an input, internal, or output transition of M . Because M offers U uses V , condition a1 implies that $\bar{p}.M$ is well-formed (i.e., non-blocking and fault-free) and $x.M \circ \langle \bar{p}.M \rangle$ is safe wrt $\{U, V\}$. If $\bar{p}.M$ does not call an event f of N , then N 's state in \bar{p} is unchanged from the end of x and so \bar{p} , and hence y , is fault free. If $\bar{p}.M$ calls an event f of N then $x.M \circ \langle f \rangle$ is safe wrt V (because M offers U uses V) and event f is enabled and its execution is well-formed (because of a2 and because N offers V uses W). Hence y is fault free and safe wrt U, V and W (Lemma 7.7).

The argument is analogous if \bar{p} is derived from an input, internal, or output transition of N .

■

Lemma 7.9 Let x be a finite execution of MN that is safe wrt $\{U, W\}$. Then

- a. For every input call f of MN (corresponding to a dnw event of U or a upw event of W): if $x \circ \langle f \rangle$ is safe wrt $\{U, W\}$, then f is enabled and its execution is well-formed at the end of x . If f returns a value, say g , then $x \circ \langle e, g \rangle$ is safe wrt $\{U, W\}$.
- b. For every output call f of MN (corresponding to a upw event of U or a dnw event of W): if MN can call f at the end of x then $x \circ \langle f \rangle$ is safe wrt $\{U, W\}$.

■

Proof Because x is an execution of MN that is safe wrt $\{U, W\}$, we have

- a1. x is fault free and safe wrt V (from Lemma 7.8).
- a2. $x.M$ is a fault-free execution of M and is safe wrt $\{U, V\}$ (from a1, Lemmas 7.3 and 7.7).
- a3. $x.N$ is a fault-free execution of N and is safe wrt V and W (from a1, Lemmas 7.3 and 7.7).

Proof of part a. Consider the case where f corresponds to a dnw event of U . Because $x \circ \langle f \rangle$ is safe wrt $\{U, W\}$, $x.M \circ \langle f \rangle$ is safe wrt U, V and W . Because M offers U uses V , we have that f is enabled and its execution well-formed at the end of $x.M$ (this also handles the case where f returns a value). Because f (being an xc event) does not call any event, there is no change in N and f is enabled and its execution is well-formed at the end of x . The case where f corresponds to a upw event of W holds by symmetry (i.e., same argument with M, N, U, W , replaced by N, M, W, U , respectively).

Proof of part b. Consider the case where f corresponds to a upw event of U . If an event e of MN can call f at the end of x , then e is an event of M that can call f at the end of $x.M$. Because M offers U uses V and because $x.M$ is an execution of M that is safe wrt $\{U, V\}$, e 's execution is well-formed at the end of $x.M$ and $x.M \circ \langle f \rangle$ is safe wrt U . Because a transition calls at most one event, e does not call any event of N and N is unchanged by this transition. Hence e 's execution is well-formed at the end of x and $x \circ \langle f \rangle$ is safe wrt U . The case where f corresponds to a dnw event of W holds by symmetry. ■

Lemmas 7.8 and 7.9 establish the safety condition of Lemma 7.2.

Establishing progress condition for MN

Lemma 7.10 If x is an execution of MN that is safe wrt $\{U, W\}$ and complete wrt $\{MN, W\}$, then x is complete wrt $\{U, V\}$. ■

Proof

- a1. x is an execution of MN that is safe wrt $\{U, W\}$. (assumption)
- a2. x is fault-free execution of MN and is safe wrt V . (a1, Lemma 7.8)
- a3. $x.M$ is a fault-free execution of M and is safe wrt $\{U, V\}$. (a1, a2, Lemma 7.3)
- a4. $x.N$ is a fault-free execution of N and is safe wrt V and W . (a1, a2, Lemma 7.3)
- a5. x is a complete execution of MN . (assumption)
- a6. $x.M$ is a complete execution of M . (a5, Lemma 7.4)
- a7. $x.N$ is a complete execution of N . (a5, Lemma 7.4)
- a8. x is complete wrt W . (assumption)
- a9. $x.N$ is complete wrt W . (a8, Lemma 7.7)
- a10. $x.N$ is complete wrt V . (a4, a8, a9, N offers V uses W)
- a11. $x.M$ is complete wrt V . (a10, Lemma 7.7)
- a12. $x.M$ is complete wrt U . (a6, a11, M offers U uses V)
- a13. x is complete wrt $\{U, V\}$ (a12, a11, Lemma 7.7)

Lemma 7.10 establishes the progress condition of Lemma 7.2.

This completes the proof of Lemma 7.2, and hence of Theorem 6.1.

7.3 Proof of program-based satisfaction

We now prove Theorem 6.2. In the proof, we indicate where exactly the absence of internal nondeterminism in services is exploited (in Lemma 7.15). We first restate the theorem introducing some notation.

Theorem 7.11 (Program-based linear satisfaction) Let system M be encapsulated by service U above and service V below. Let M' denote system $M\text{-wrt-}\{U, V\}$. Let U' denote system $U\text{-wrt-}M$. Let V' denote system $V\text{-wrt-}M$. Let M^* be the closed composite system of M' , U' , and V' .

- The safety condition for M offers U uses V holds iff M^* is fault-free.
- The progress condition for M offers U uses V holds iff M^* satisfies $V.\text{progress} \Rightarrow U.\text{progress}$

We first state some lemmas relating executions of M^* to executions of M' , U' , V' , U , and V .

Lemma 7.12 If x is a fault-free execution of M^* , then $x.M'$ is a fault-free execution of M' , $x.U'$ is a fault-free execution of U' , and $x.V'$ is a fault-free execution of V' . Furthermore, x and $x.M$ are safe wrt $\{U, V\}$. ■

Lemma 7.13 If x is a complete execution of M^* , then $x.M'$ is a complete execution of M' , $x.U'$ is a complete execution of U' , and $x.V'$ is a complete execution of V' . Furthermore, x and $x.M$ are complete wrt $\{U, V\}$. ■

Lemma 7.14 Let x be a fault-free execution of M^* and Y be a safety or progress assertion with no free variables of U', V' (or M', U' or M', V'). Then x satisfies Y iff $x.M'$ (or $x.V'$ or $x.U'$) satisfies Y . ■

Lemmas 7.12 and 7.13 follow from Theorem 4.1 applied to M^* , the definitions of M' , U' and V' , and the definition of “safe wrt” and “complete wrt”. Lemma 7.14 follows from Theorem 5.4 applied to M^* . (Note that Theorems 4.1 and 5.4 hold even if U' and V' have internal nondeterminism.)

For any sequence x and set E , let $\text{ssq}(x, E)$ denote the sequence of elements of E in x . For any system or service S , let $S.\text{svc}$ denote the union of the event calls and returns of S .

Trace-based implies program-based

We now establish one direction of Theorem 7.11.

Lemma 7.15 If the safety condition for M offers U uses V holds, then M^* is fault-free. ■

Proof We prove by contradiction. Assume M^* is not fault free. That is, M^* has an execution $y \circ \langle \bar{p} \rangle$ where y is fault free and \bar{p} is a faulty transition (assume, for notational convenience, that the last state of y is repeated as the first state of \bar{p}). Because y is fault free, $y.U'$ and $y.V'$ are fault-free executions of U' and V' respectively, and $y, y.U', y.M'$, and $y.V'$ are each safe wrt $\{U, V\}$ (from Lemma 7.12).

We consider the different cases of \bar{p} :

- a1. Let \bar{p} be due to an input call f of M' , where f matches a U dnw event, say f_U . Thus we have (b1) f_U is enabled at the end of y , and either (b2) f is not enabled at the end of y , or (b3) f is enabled at the end of y but its execution is faulty. In either case b2 or b3, we have that $y.M'$ is an execution of M' that cannot handle input f . But from b1, we have $(y \circ \langle \bar{p} \rangle).U'$ is an execution of U' , hence $(y \circ \langle f \rangle).U'$ is safe wrt U , hence $y.M' \circ \langle f \rangle$ is safe wrt U . Thus the safety condition of M offers U uses V does not hold.
- a2. Let \bar{p} be due to an input call f of M' , where f matches a V upw event. The argument is as in case a1.
- a3. Let \bar{p} be due to an lc event execution of M' that is faulty. Then M can extend the execution $y.M$ to a faulty execution even though $y.M$ is safe wrt $\{U, V\}$. Thus the safety condition of M offers U uses V does not hold.

- a4. Let \bar{p} be due to an lc event execution of M' that calls an event f of U such that f is not enabled at the end of y . We need to show that $y \circ \langle f \rangle$ is not safe wrt U , that is, there is no execution z of U such that $\text{ssq}(z, U.\text{evc})$ equals $\text{ssq}(y \circ \langle f \rangle, U.\text{evc})$, or equivalently, there is no execution q of U such that $\text{ssq}(q, U.\text{evc})$ equals $\text{ssq}(y, U.\text{evc})$ and f is enabled at the end of q . *Because U has no internal nondeterminism*, distinct executions of U have distinct external traces. Hence $\text{ssq}(q, U.\text{evc})$ equals $\text{ssq}(y, U.\text{evc})$ iff q equals y . Thus f would not be enabled at the end of q .
- a5. Let \bar{p} be due to an lc event execution of M' that is faulty or calls an event f of V such that f is not enabled at the end of y . The argument is as in cases a3 and a4.

So in any case, the safety condition of M offers U uses V does not hold. ■

Lemma 7.16 If the progress condition for M offers U uses V holds, then M^* satisfies $V.\text{progress} \Rightarrow U.\text{progress}$ ■

Proof We prove by contradiction. Assume M^* is fault free but does not satisfy the progress condition. That is, there is a complete execution of M^* that satisfies $V.\text{progress}$ but not $U.\text{progress}$, or equivalently, an execution y of M^* that satisfies $M.\text{progress}$ and $V.\text{progress}$ but not $U.\text{progress}$. So we have the following (from Lemmas 7.13 and 7.12): (a1) $y.M'$ is an execution of M' that satisfies $M.\text{progress}$, (a2) $y.V'$ is an execution of V' that satisfies $V.\text{progress}$, and (a3) $y.U'$ is an execution of U' that does not satisfy $U.\text{progress}$. Hence we have: $y.M$ is a complete execution of M (from a1); $y.V$, and hence $y.M$, is complete wrt V ; $y.U$, and hence $y.M$, is safe but not complete wrt V . So $y.M$ is an execution of M that is safe wrt $\{U, V\}$, complete wrt $\{M, V\}$ but not complete wrt U . Thus the progress condition of M offers U uses V does not hold. ■

Program-based implies trace-based

We now establish the other direction of Theorem 7.11. We first state a lemma that allows us to “stitch” together signature-compatible behaviors of M , U , and V to yield an execution of M^* .

Lemma 7.17 Let x , y , and z be fault-free executions of M , U , and V , respectively, such that $\text{ssq}(y, U.\text{evc}) = \text{ssq}(x, U.\text{evc})$ and $\text{ssq}(z, V.\text{evc}) = \text{ssq}(x, V.\text{evc})$ hold. Let x' be the execution of M' corresponding to x . Let y' be the execution of U' corresponding to y . Let z' be the execution of V' corresponding to z . Then there exists a fault-free execution w of M^* such that $w.M' = x'$, $w.U' = y'$, $w.V' = z'$ hold and $w.M = x$, $w.U = y$, $w.V = z$ hold. ■

Proof From $\text{ssq}(y, U.\text{evc}) = \text{ssq}(x, U.\text{evc})$, $\text{ssq}(z, V.\text{evc}) = \text{ssq}(x, V.\text{evc})$, and the definitions of M' , U' , and V' , we have that $\text{ssq}(y', U'.\text{evc}) = \text{ssq}(x', U'.\text{evc})$ and $\text{ssq}(z', V'.\text{evc}) = \text{ssq}(x', V'.\text{evc})$ hold. Thus x' , y' , and z' are signature-compatible. So from Theorem 4.2 applied to M^* , there exists a fault-free execution w of M^* such that $w.M' = x'$, $w.U' = y'$, and $w.V' = z'$ hold. From the definitions of M' , U' , and V' , we have that $w.M = x$, $w.U = y$, and $w.V = z$ hold. ■

Lemma 7.18 If M^* is fault-free, then the safety condition of M offers U uses V holds. ■

Proof We prove by contradiction. Assume that the safety condition of M offers U uses V does not hold. So there exists a fault-free finite execution x of M that is safe wrt $\{U, V\}$ and at least one of the following hold:

- a1. For some input call f of M , $x \circ \langle f \rangle$ is safe wrt $\{U, V\}$, but f is not enabled or its execution the end of x is faulty. Without loss of generality, assume f matches an event of U (the argument is symmetric if f matches an event of V).

Because $x \circ \langle f \rangle$ is safe wrt U , there exists an (fault-free) execution p of U such that $\text{ssq}(x \circ \langle f \rangle, U.\text{evc}) = \text{ssq}(p, U.\text{evc})$ holds and p 's last transition matches f . Let y be p without the last transition. Then $\text{ssq}(x, U.\text{evc}) = \text{ssq}(y, U.\text{evc})$.

Because x is safe wrt V , there exist a finite execution z of V such that $\text{ssq}(x, V.\text{evc}) = \text{ssq}(z, V.\text{evc})$. Let x' be the execution of M' corresponding to x . Let y' be the execution of U' corresponding to y . Let z'

be the execution of V' corresponding to z . From Lemma 7.17, there exists a fault-free execution w of M^* such that $w.M' = x'$, $w.U' = y'$, $w.V' = z'$ hold and $w.M = x$, $w.U = y$, $w.V = z$ hold.

The U' event matching f , say $f_{U'}$, is enabled at the end of y , and hence at the end of y' , and hence at the end of w . But the execution of f (of M) is not well-formed at the end of x , and hence f 's execution is not well-formed at the end of w . So $f_{U'}$ is enabled at the end of w but its execution is faulty because it calls f . Thus M^* has a faulty execution.

a2. M can extend x (by an lc event execution) to q such that q is faulty or not safe wrt $\{U, V\}$.

Because x is safe wrt $\{U, V\}$, there exist finite executions y of U and z of V such that $\text{ssq}(x, U.\text{evc}) = \text{ssq}(y, U.\text{evc})$ and $\text{ssq}(x, V.\text{evc}) = \text{ssq}(z, V.\text{evc})$. Let x' be the execution of M' corresponding to x . Let y' be the execution of U' corresponding to y . Let z' be the execution of V' corresponding to z . From Lemma 7.17, there exists a fault-free execution w of M^* such that $w.M' = x'$, $w.U' = y'$, $w.V' = z'$ hold and $w.M = x$, $w.U = y$, $w.V = z$ hold.

Because M can extend x to q by an lc event execution, the same lc event is enabled at the end of x' . Suppose M' extends x' to q' by executing this lc event. If q is faulty, then q' is faulty. If q is not safe wrt $\{U, V\}$, then the lc event execution calls an event of U or V that is not enabled (otherwise q would be safe wrt $\{U, V\}$). In both cases, M^* has a faulty execution.

In both cases a1 and a2, M^* is faulty. ■

Lemma 7.19 If M^* is fault-free and satisfies $V.\text{progress} \Rightarrow U.\text{progress}$, then the progress condition of M offers U uses V holds. ■

Proof We prove by contradiction. Assume that the progress condition of M offers U uses V does not hold (and the safety condition holds). So there exists an execution x of M that is safe wrt $\{U, V\}$ and complete wrt $\{M, V\}$ but not complete wrt U . Because x is safe but not complete wrt U , there exists an execution y of U that does not satisfy $U.\text{progress}$ such that $\text{ssq}(x, U.\text{evc}) = \text{ssq}(y, U.\text{evc})$. Because x is complete wrt V , there exists an execution z of V that satisfies $V.\text{progress}$ such that $\text{ssq}(x, V.\text{evc}) = \text{ssq}(z, V.\text{evc})$. Let x' be the execution of M' corresponding to x . Let y' be the execution of U' corresponding to y . Let z' be the execution of V' corresponding to z . From Lemma 7.17, there exists a fault-free execution w of M^* such that $w.M' = x'$, $w.U' = y'$, and $w.V' = z'$ hold and $w.M = x$, $w.U = y$, and $w.V = z$ hold. Furthermore, w satisfies $M.\text{progress}$ (because x , and hence x' , satisfies it), w satisfies $V.\text{progress}$ (because z , and hence z' , satisfies it), and w does not satisfy $U.\text{progress}$ (because y , and hence y' , does not satisfy it). Thus w does not satisfy $M.\text{progress} \wedge V.\text{progress} \Rightarrow U.\text{progress}$. Hence M^* does not satisfy $V.\text{progress} \Rightarrow U.\text{progress}$. ■

Theorem 7.11 follows from Lemmas 7.15, 7.16, 7.18, and 7.19.

7.4 Compositionality for mesh and partial-service mesh

The above proofs extend in a straight-forward manner to mesh and partial-service layered systems. Just follow the same sequence of steps, appropriately substituting services with service sets or partial service sets. The details are omitted.