Studies in Systems, Decision and Control 89

Calin Belta
Boyan Yordanov
Ebru Aydin Gol

Formal Methods for Discrete-Time Dynamical Systems

Springer

Chapter 7
Largest Satisfying Region

In this chapter, we develop a procedure that attempts to find the largest set of initial states from which an autonomous PWA system (Definition 6.3) satisfies an LTL formula over the set labeling the polytopes in its definition. The same problem was considered in Chap. 4 for a finite transition system and an LTL formula over its set of observations. Several methods were presented to find an exact solution to this problem. As expected, since PWA systems have infinitely many states, we are only able to find a subset of the largest satisfying region in this chapter. We formulate the problem for the general case of autonomous PWA systems with uncertain parameters, and we show that more efficient solutions can be found for the particular cases of autonomous PWA systems with fixed parameters and additive uncertainties. The problem that we consider in this chapter can be formally stated as follows:

Problem 7.1 (*Largest Satisfying Region for PWA Systems*) Given an autonomous PWA system $\mathscr{W}$ (Definition 6.3) and an LTL formula ϕ over $L \cup \{\text{Out}\}$, find the largest set of initial states from which all trajectories of $\mathscr{W}$ satisfy ϕ.

From Definition 6.7, solving Problem 7.1 involves working with the infinite embedding transition system $T_{\mathscr{W}}$ (Definition 6.6) and formula ϕ. In Chap. 3, we described LTL model checking as an algorithmic procedure for deciding whether a finite transition systems satisfies an LTL formula. Then, in Sect. 4.1, we used model checking to develop an analysis procedure for finite transition systems (Algorithm 3). Since the embedding $T_{\mathscr{W}}$ from Definition 6.6 is infinite, neither model checking, nor the analysis procedure from Algorithm 3 can be applied directly to solve Problem 7.1.

In Chap. 4, we also developed several methods for the analysis of potentially large transition systems through the construction and refinement of their quotients. In the following sections, we show that this theory can be extended to infinite transition systems such as $T_{\mathscr{W}}$, in order to address Problem 7.1.

First, we consider autonomous, fixed-parameter PWA systems (Definition 6.4) and autonomous, additive-uncertainty systems (Definition 6.5). For these systems, we show that the quotient construction and refinement procedures from Chap. 4

C. Belta et al., *Formal Methods for Discrete-Time Dynamical Systems*,
Studies in Systems, Decision and Control 89, DOI 10.1007/978-3-319-50763-7_7

are implementable, and use them to solve Problem 7.1 in Sect. 7.1. For general autonomous PWA systems with uncertain parameters (Definition 6.3), we develop a conservative procedure in Sect. 7.2.

7.1 PWA Systems with Fixed and Additive Uncertain Parameters

The following discussion applies to autonomous PWA systems with additive parameter uncertainty (Definition 6.5). All the results automatically apply to the subclass of autonomous fixed-parameter PWA systems (Definition 6.4). We will discuss the differences as appropriate.

We describe the construction of quotient $T_{\mathscr{W}}/_{\sim} = (X_{\mathscr{W}}/_{\sim}, \delta_{\mathscr{W},\sim}, O_{\mathscr{W}}, o_{\mathscr{W},\sim})$ through the construction of its sets of states and observations, and observation and transition maps. From the definition of the observational equivalence relation $\sim$ (Definition 1.2), induced by observation map $o_{\mathscr{W}}$ of $T_{\mathscr{W}}$ (Definition 6.6) and the definition of the quotient $T_{\mathscr{W}}/_{\sim}$ (Definition 1.3), the set of states $X_{\mathscr{W}}/_{\sim}$ of quotient $T_{\mathscr{W}}/_{\sim}$ is simply the set of observations $X_{\mathscr{W}}/_{\sim} = O_{\mathscr{W}} = L \bigcup \{\text{Out}\}$ of $T_{\mathscr{W}}/_{\sim}$, which is inherited from $T_{\mathscr{W}}$, and the observation map is identity. Given a state $l \in X_{\mathscr{W}}/_{\sim}$, where $l \neq$ Out, the set of all equivalent states from l is

$$con(l) = \mathbf{X}_l. \tag{7.1}$$

In other words, each equivalence class is a polytope from the PWA system definition (Definition 6.1), while the explicit representation of the set $con(\text{Out}) = \mathbb{R}^N \setminus \mathbf{X}$ is not required for our methods.

In order to complete the construction of quotient $T_{\mathscr{W}}/_{\sim}$, we need to compute the transition function $\delta_{\mathscr{W},\sim}$. In Sect. 1.3, we showed that through Eq. (1.7), transitions of the quotient $T_{\mathscr{W}}/_{\sim}$ can be found by computing the set of successors of a region in $T_{\mathscr{W}}$ using the $Post()$ operation defined in Eq. (1.4)—given states $l_1, l_2 \in X_{\mathscr{W}}/_{\sim}$, there exists a transition from l_1 to l_2 (i.e., $l_2 \in \delta_{\mathscr{W},\sim}(l_1)$) if and only if the intersection $Post(con(l_1)) \cap con(l_2)$ is non-empty. From the computation of the set of equivalent states $con(l)$ for an equivalence class $l \in X_{\mathscr{W}}/_{\sim}$ given in Eq. (7.1), checking if a transition between states $l_1, l_2 \in X_{\mathscr{W}}/_{\sim}$ exists amounts to checking the non-emptiness of the intersection $Post(\mathbf{X}_{l_1}) \cap \mathbf{X}_{l_2}$. Formally, the computation of transitions in the quotient $T_{\mathscr{W}}/_{\sim}$ for any states $l_1, l_2 \in X_{\mathscr{W}}/_{\sim}$, where $l_1 \neq$ Out and $l_2 \neq$ Out is summarized as

$$l_2 \in \delta_{\mathscr{W},\sim}(l_1) \text{ if and only if } Post(\mathbf{X}_{l_1}) \cap \mathbf{X}_{l_2} \neq \emptyset. \tag{7.2}$$

Given a polytope $\mathbf{X}_l$ for some $l \in L$, the set of successor states $Post(\mathbf{X}_l)$ is another polytope computable as[1]:

$$Post(\mathbf{X}_l) = A_l\mathbf{X}_l \oplus \mathbf{P}_l^c, \tag{7.3}$$

where $A_l\mathbf{X}_l$ is the image of polytope $\mathbf{X}_l$ through matrix A_l (see Appendix A.3) and "$\oplus$" denotes the Minkowski (set) sum (Definition A.7). Since $Post(\mathbf{X}_l)$ is a polytope, for any states $l_1, l_2 \in X_{\mathscr{W}}/_\sim$ the intersection $Post(\mathbf{X}_{l_1}) \cap \mathbf{X}_{l_2}$ is also a polytope and its non-emptiness can be checked easily using polyhedral operations. Note that, for the particular case of a fixed parameter PWA system, $\mathbf{P}_l^c$ in Eq. (7.3) is a singleton $c_l \in \mathbb{R}^N$.

Given a state $l \in X_{\mathscr{W}}/_\sim$, where $l \neq$ Out, a transition from state l to state Out is assigned in accordance to Definition 6.6 as

$$\text{Out} \in \delta_{\mathscr{W},\sim}(l) \text{ if and only if } Post(\mathbf{X}_l) \nsubseteq \mathbf{X}, \tag{7.4}$$

which is also checked easily, since both $Post(\mathbf{X}_l)$ and $\mathbf{X}$ are polytopic sets. To complete the construction of $\delta_{\mathscr{W},\sim}$, transitions for state Out $\in X_{\mathscr{W}}/_\sim$ must be assigned but, from Definitions 6.1 and 6.6, it only has a transition to itself (i.e., $\delta_{\mathscr{W},\sim}(\text{Out}) = \{\text{Out}\}$).

Algorithm 14 $T_{\mathscr{W}}/_\sim$ =QUOTIENT($\mathscr{W}$) : Compute the quotient $T_{\mathscr{W}}/_\sim$ of an additive uncertainty PWA system $\mathscr{W}$

```
$X_{\mathscr{W}}/_\sim := L \bigcup \{\text{Out}\}$
$O_{\mathscr{W}} := X_{\mathscr{W}}/_\sim$
for all $l \in X_{\mathscr{W}}/_\sim$ do
  $o_{\mathscr{W},\sim}(l) := l$
  $\delta_{\mathscr{W},\sim} := \emptyset$
  if $Post(\mathbf{X}_l) \nsubseteq \mathbf{X}$ then
    $\delta_{\mathscr{W},\sim}(l) := \delta_{\mathscr{W},\sim}(l) \cup \{\text{Out}\}$
  end if
  for all $l' \in X_{\mathscr{W}}/_\sim$ do
    if $Post(\mathbf{X}_l) \cap \mathbf{X}_{l'} \neq \emptyset$ then
      $\delta_{\mathscr{W},\sim}(l) := \delta_{\mathscr{W},\sim}(l) \cup \{l'\}$
    end if
  end for
end for
$\delta_{\mathscr{W},\sim}(\text{Out}) := \{\text{Out}\}$
return $T_{\mathscr{W}}/_\sim = (X_{\mathscr{W}}/_\sim, \delta_{\mathscr{W},\sim}, O_{\mathscr{W}}, o_{\mathscr{W},\sim})$
```

The transition map of quotient $T_{\mathscr{W}}/_\sim$ is constructed using the computation described above, which completes the quotient's construction. The computation of

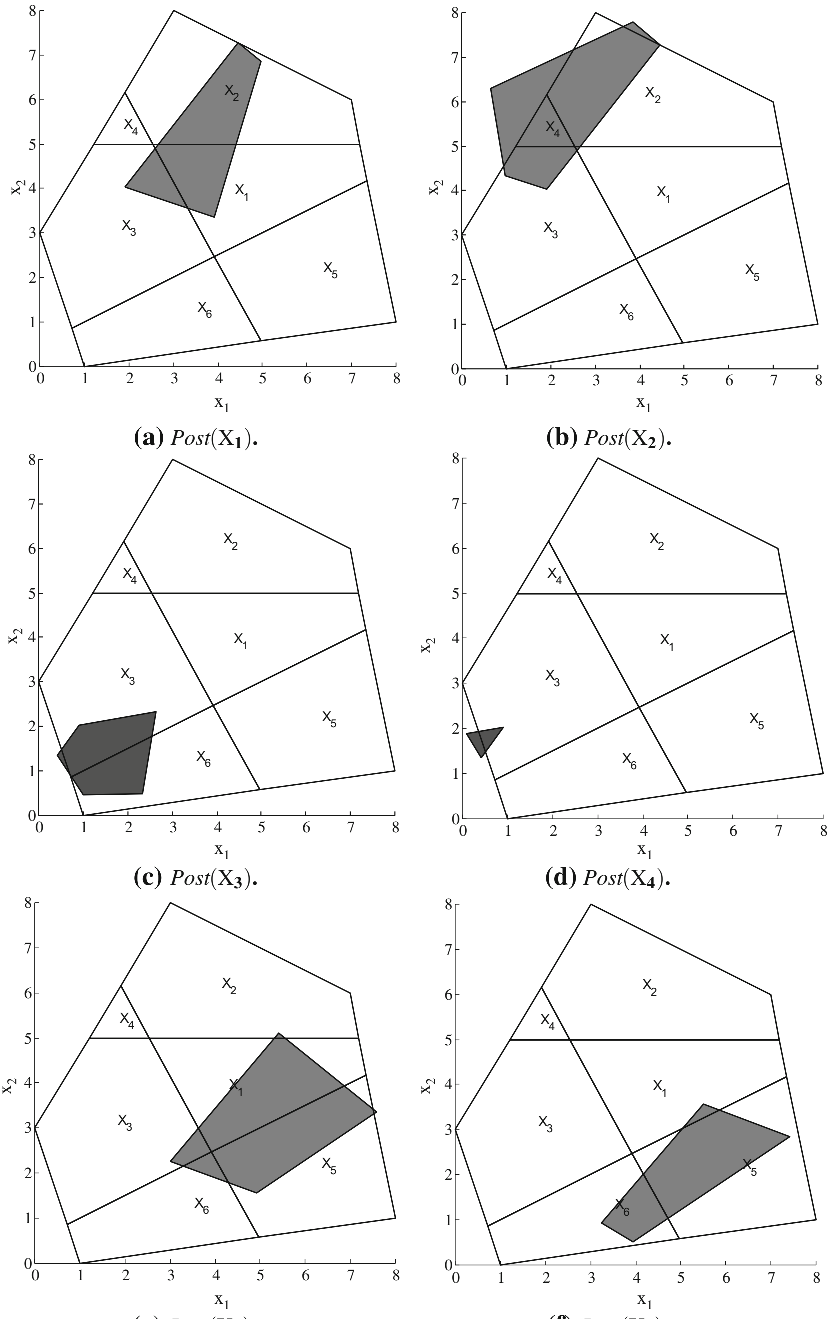
[1] In this chapter, we assume, for simplicity of presentation, that all matrices $A_l, l \in L$ are invertible. This assumption can be easily relaxed as discussed in Sect. 7.4. The technical details are included in Sects. A.3 and A.4.

$T_{\mathscr{W}}/_{\sim}$ is summarized in Algorithm 14, which is implementable using polyhedral operations on polytopes. Since the number of regions L of PWA system $\mathscr{W}$ is finite, $T_{\mathscr{W}}$ has a finite set of observations $O_{\mathscr{W}}$ and, as a result, the set of states $X_{\mathscr{W}}/_{\sim}$ of the quotient is also finite. This allows the application of model checking or analysis of $T_{\mathscr{W}}/_{\sim}$ through Algorithm 3 but the implementation of the more advanced analysis procedure from Chap. 4 requires additional operations, which will be discussed next.

Example 7.1 We apply Algorithm 14 to construct the quotient $T_{\mathscr{W}}/_{\sim}$ for the PWA system $\mathscr{W}$ defined in Example 6.2. Initially, the system had four regions (Fig. 6.2a) but additional partitioning of the state space was required to accommodate some specifications, resulting in a system with six regions denoted by $\mathbf{X}_1, \ldots, \mathbf{X}_6$ (Fig. 6.2b) with $L = \{1, \ldots, 6\}$. Therefore, the quotient $T_{\mathscr{W}}/_{\sim}$ has six states $X_{\mathscr{W}}/_{\sim} = \{1, \ldots, 6\}$ where, for each state $l \in X_{\mathscr{W}}/_{\sim}$, the set of equivalent states of $T_{\mathscr{W}}$ (and therefore $\mathscr{W}$) is given by $con(l) = \mathbf{X}_l$ as in Eq. (7.1).

To compute the transitions of $T_{\mathscr{W}}/_{\sim}$, we compute the set of successors $Post(\mathbf{X}_l)$ for each region $\mathbf{X}_l$ of $T_{\mathscr{W}}$ (see Fig. 7.1). Checking the non-emptiness of the intersection $Post(\mathbf{X}_{l_1}) \cap \mathbf{X}_{l_2}$ allows us to compute the transitions of $T_{\mathscr{W}}/_{\sim}$. Only the set of successors of region 6 is completely included within the defined state space $\mathbf{X}$ and, therefore, all other states have a transition to state Out (note that $Post(\mathbf{X}_1) \not\subset \mathbf{X}$, although this is not obvious from Fig. 7.1a). This leads to the inclusion of transitions $\delta_{\mathscr{W},\sim}(1) = \{1, 2, 3, \text{Out}\}$, $\delta_{\mathscr{W},\sim}(2) = \{2, 3, 4, \text{Out}\}$, $\delta_{\mathscr{W},\sim}(3) = \{3, 6, \text{Out}\}$, $\delta_{\mathscr{W},\sim}(4) = \{3, \text{Out}\}$, $\delta_{\mathscr{W},\sim}(5) = \{1, 2, 3, 5, 6, \text{Out}\}$ and $\delta_{\mathscr{W},\sim}(6) = \{1, 5, 6\}$. The resulting quotient $T_{\mathscr{W}}/_{\sim}$ is shown in Fig. 7.2, where the observations for each state are omitted but are clear from the state labels.

By embedding the PWA system $\mathscr{W}$ into an infinite transition system $T_{\mathscr{W}}$ (Definition 6.6), we reduced Problem 7.1 to Problem 4.1. However, since $T_{\mathscr{W}}$ was infinite, the analysis procedure outlined as Algorithm 3 in Chap. 4 could not be applied directly. So far, we showed that the quotient $T_{\mathscr{W}}/_{\sim}$ of the embedding $T_{\mathscr{W}}$ under the observational equivalence relation $\sim$ (Definition 1.2) can be constructed using polyhedral operations (Algorithm 14). Since $T_{\mathscr{W}}/_{\sim}$ is finite, this allows us to apply the analysis technique described in Sect. 1.3. However, as discussed there, such an approach leads to a conservative solution to Problem 7.1. In order to obtain less conservative results, bisimulation-based and formula-guided quotient refinement techniques were proposed in Sects. 4.3 and 4.5, respectively. Both methods were initialized by constructing a finite quotient such as $T_{\mathscr{W}}/_{\sim}$ but in addition required the implementation of a state refinement procedure.



(a) *Post*($\mathbf{X_1}$).

(b) *Post*($\mathbf{X_2}$).

(c) *Post*($\mathbf{X_3}$).

(d) *Post*($\mathbf{X_4}$).

(e) *Post*($\mathbf{X_5}$).

(f) *Post*($\mathbf{X_6}$).

Fig. 7.1 Successor states (*shaded gray*) of different regions of PWA system $\mathscr{W}$ defined in Example 6.2 (Fig. 6.2b). See Example 6.2 for additional details

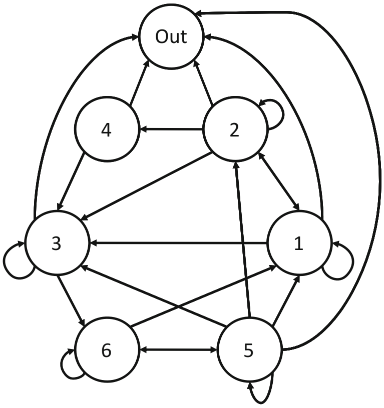


Fig. 7.2 Finite quotient $T_{\mathscr{W}}/_{\sim}$ of PWA system $\mathscr{W}$ defined in Example (6.2) (Fig. 6.2b). Observations of the states are omitted. See Example 7.1 for additional details

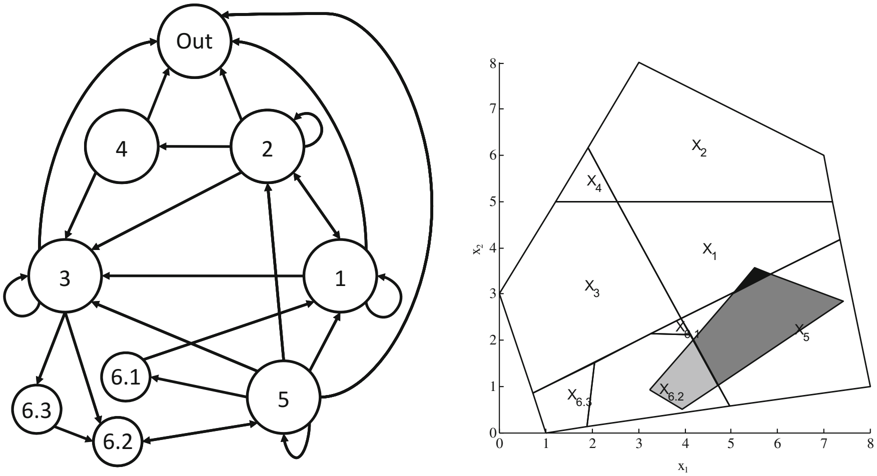
In the following, we focus on the implementation of the refinement procedure REFINE() (Algorithm 5, Chap. 4) and show that for autonomous additive uncertainty (and fixed parameter) PWA systems all its operations are computable through polyhedral operations. Specifically, as the embedding of a PWA system with additive parameter uncertainty is non-deterministic, we will refer to Algorithm 5. For the particular case of a PWA with fixed parameters, whose embedding is deterministic, the refinement procedure is described in Algorithm 6.

To implement function REFINE() for $T_{\mathscr{W}}$, given states $l_1, l_2 \in X_{\mathscr{W}}/_{\sim}$ such that $l_2 \in \delta_{\mathscr{W},\sim}(l_1)$ (i.e., l_2 is reachable from l_1 in $T_{\mathscr{W}}/_{\sim}$), we need to be able to construct a state l', such that $con(l') = con(l_1) \cap Pre(con(l_2))$ (see Algorithms 5 and 6). From Eq. (7.1), this computation reduces to the construction of a state l' where $con(l') = \mathbf{X}_{l_1} \cap Pre(\mathbf{X}_{l_2})$.

Under the invertibility assumption made earlier in this chapter, which, as stated, can be easily relaxed (see Sect. 7.4), this intersection is computable as

$$\mathbf{X}_{l_1} \cap Pre(\mathbf{X}_{l_2}) = \mathbf{X}_{l_1} \cap A_{l_1}^{-1}(\mathbf{X}_{l_2} \ominus \mathbf{P}_{l_1}^{c}), \tag{7.5}$$

where $\ominus$ denotes the Minkowski difference (Definition A.8). Note that while the $Pre()$ operation is applied to region $\mathbf{X}_{l_2}$, the parameters of region $\mathbf{X}_{l_1}$ are used for the computation, which is consistent with Definition 6.2. Using Eq. (7.5) to refine the states of $T_{\mathscr{W}}/_{\sim}$ and Eq. (7.3) to update its transitions wherever necessary (see Algorithms 5 and 6) allows the implementation of function REFINE() and all computation is performed using polyhedral operations.



(a) Refined quotient $T_{\mathscr{W}}/_{\sim_r}$ (b) Equivalence classes $X_{\mathscr{W}}/_{\sim_r}$

Fig. 7.3 Refined quotient $T_{\mathscr{W}}/_{\sim_r} = $ REFINE($T_{\mathscr{W}}/_{\sim}, 6$) (**a**) and equivalence classes $X_{\mathscr{W}}/_{\sim_r}$ (**b**) of PWA system $\mathscr{W}$ from Example 6.2 (Fig. 6.2b). The successor states $Post(con(6.1))$ (dark gray), $Post(con(6.2))$ (medium gray) and $Post(con(6.3))$ (light gray) are also shown for the refined subsets $6.1, 6.2, 6.3 \in X_{\mathscr{W}}/_{\sim_r}$, where $con(6.1) \cup con(6.2) \cup con(6.3) = con(6)$ for state $6 \in X_{\mathscr{W}}/_{\sim}$. See Example 7.2 for additional details

Example 7.2 We apply function REFINE() (Algorithm 6) to refine the quotient $T_{\mathscr{W}}/_{\sim}$ (constructed in Example 7.1 and shown in Fig. 7.2) of PWA system $\mathscr{W}$ defined in Example 6.2 (Fig. 6.2b). We target refinement to state $6 \in X_{\mathscr{W}}/_{\sim}$ and construct the refined quotient $T_{\mathscr{W}}/_{\sim_r} = $ REFINE($T_{\mathscr{W}}/_{\sim}, 6$). State 6 has three successors in $X_{\mathscr{W}}/_{\sim}$ (i.e., $\delta_{\mathscr{W},\sim} = \{1, 5, 6\}$) and, therefore, refinement results in three subsets in $X_{\mathscr{W}}/_{\sim_r}$ denoted as 6.1, 6.2 and 6.3, where $con(6.1) \cup con(6.2) \cup con(6.3) = con(6)$. Each subset has only a single outgoing transitions in $T_{\mathscr{W}}/_{\sim_r}$ (see the sets of successors shown in Fig. 7.3b), which is implicitly induced through the refinement and incoming transitions are recomputed (see Algorithm 6). This results in the construction of the refined quotient $T_{\mathscr{W}}/_{\sim_r}$ shown in Fig. 7.3a.

Note that the notation is abused in this example and in the rest of this chapter. As we assumed that all the polytopes are open, the equality $con(6.1) \cup con(6.2) \cup con(6.3) = con(6)$ does not hold precisely. Indeed, $con(6)$ contains some facets of $con(6.1)$, $con(6.2)$, and $con(6.3)$, which are not contained in $con(6.1) \cup con(6.2) \cup con(6.3)$. More discussions are included in Sect. 7.4.

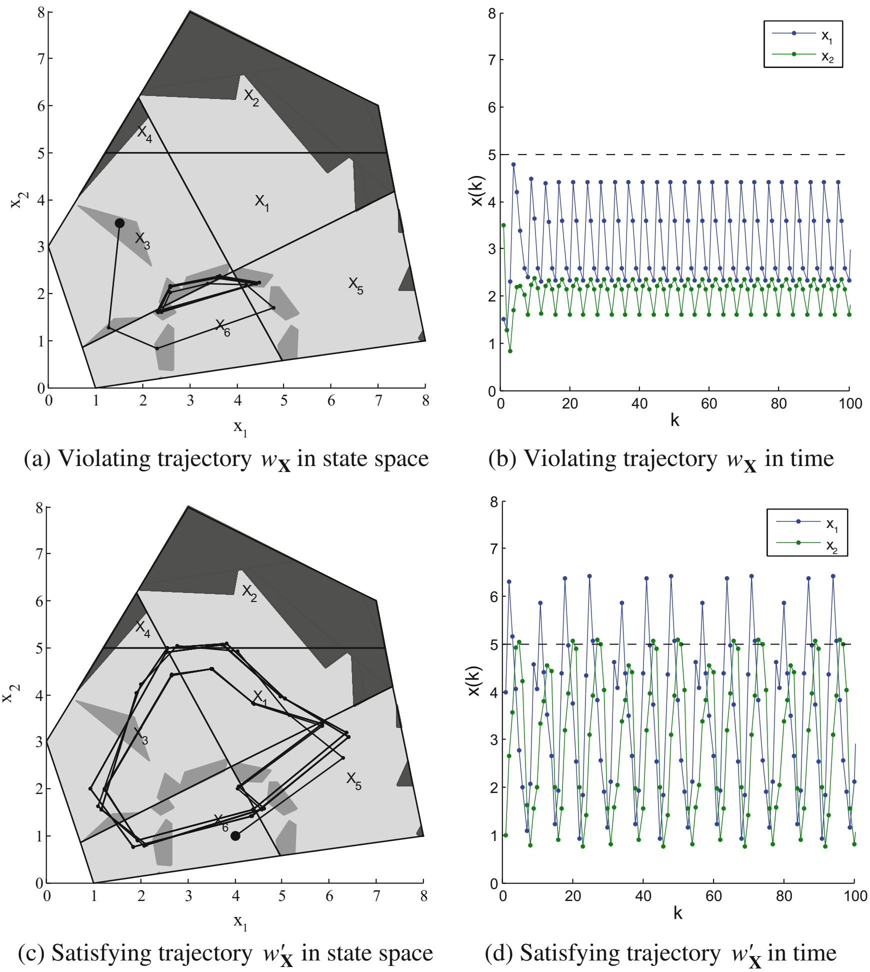
After a state $l \in X_{\mathscr{W}}/_{\sim}$ is refined into states l_1 and l_2 such that $con(l_1) \cup con(l_2) = con(l)$, the computation from Eqs. (7.3) and (7.5) can be applied to the subsets l_1 and l_2. This enables the iterative refinement of the quotient $T_{\mathscr{W}}/_{\sim}$ and allows the implementation of the bisimulation algorithm (Algorithm 1) and the analysis methods described in Sects. 4.3 (Algorithm 7) and 4.5 (Algorithm 8) for PWA systems.

A termination condition based on the sizes of equivalence classes was also proposed in Chap. 4 for the analysis procedures discussed there. To determine if a state $l \in X_{\mathscr{W}}/_{\sim}$ is “large enough” to undergo additional refinement, we compute the radius of the largest sphere inscribed in polytope $con(l)$ and apply the refinement procedure only if it is larger than a certain predefined limit ε. In other words, we apply the refinement procedure to state l only if $r(\mathbf{X}_l) > \varepsilon$, where $r(\mathbf{X}_l)$ is the radius of the Chebyshev ball of $\mathbf{X}_l$ (see Definition A.9 in the Appendix).

Example 7.3 We apply the analysis method from Sect. 4.3 summarized as Algorithm 7 to identify satisfying and violating regions of PWA system $\mathscr{W}$ defined in Example 6.2 and shown in Fig. 6.2a. We are interested in testing whether trajectories of the system keep reaching values over 5 in the second component x_2 of the system state x. Therefore, we introduce additional partitions to the states space of the system as shown in Fig. 6.2b and formulate the specification as the LTL formula $\phi = \Box\Diamond(2 \vee 4)$, requiring that states from regions $\mathbf{X}_2$ and $\mathbf{X}_4$ are visited infinitely often. Furthermore, we want to guarantee that trajectories of the system remain within the defined state space $\mathbf{X}$ and therefore augment the specification as $\phi' = \Box\Diamond(2 \vee 4) \wedge \Box\neg\text{Out}$.

We set $\varepsilon = 0.1$ as the limit on the states from $X_{\mathscr{W}}/_{\sim_r}$ that can undergo refinement, which guarantees the termination of the analysis procedure. While only an under-approximation of the largest satisfying and strictly violating regions of $T_{\mathscr{W}}$ (and therefore $\mathscr{W}$) is obtained as discussed in Chap. 4, most of the system’s state space is characterized as satisfying or violating (see Fig. 7.4a).

All trajectories originating in the regions shown in dark and medium gray in Fig. 7.4a violate specification ϕ'—trajectories originating in the dark gray region leave the defined state space $\mathbf{X}$, while trajectories originating in the light gray region oscillate but do not visit states where the values of the second component x_2 are above 5. However, all trajectories originating in the satisfying region shown in light gray in Fig. 7.4a are guaranteed to satisfy the specification.



(a) Violating trajectory $w_{\mathbf{X}}$ in state space (b) Violating trajectory $w_{\mathbf{X}}$ in time

(c) Satisfying trajectory $w'_{\mathbf{X}}$ in state space (d) Satisfying trajectory $w'_{\mathbf{X}}$ in time

Fig. 7.4 Satisfying (*light gray*) and violating (*medium gray*) regions of PWA system $\mathscr{W}$ defined in Example 6.2 (Fig. 6.2b) for specification "$\Box\Diamond(2 \vee 4) \wedge \Box\neg$Out" were identified using the analysis procedure described in Sect. 4.3 (Algorithm 7). Trajectories of $\mathscr{W}$ originating in the region shown in dark gray leave the defined state space of the system and therefore are also violating. A violating trajectory $w_{\mathbf{X}}$ (**a**) and (**b**) and a satisfying trajectory $w'_{\mathbf{X}}$ (**c**) and (**d**) were obtained by initializing $\mathscr{W}$ in the violating or satisfying regions (initial conditions are shown as large *circles*). See Example 7.3 for additional details

Note that for an autonomous, fixed-parameter PWA system $\mathscr{W}$ (Definition 6.4), the embedding $T_{\mathscr{W}}$ is deterministic, which allows the application of the more efficient refinement strategies from Algorithm 6. The computation from Eq. (7.5) is also sufficient to implement refinement strategies for autonomous, additive uncertainty PWA systems (Definition 6.5) through Algorithm 5. Refinement strategies for more general autonomous, uncertain parameters systems are discussed in the following Sect. 7.2.

7.2 PWA Systems with Uncertain Parameters

When the matrix component of the parameters is allowed to vary as in the autonomous PWA system from Definition 6.3, given a polytope $\mathbf{X}_l$, the set $Post_{T_{\mathscr{W}}}(\mathbf{X}_l)$ is not necessarily convex and, in general, there are no algorithms capable of its exact computation. Thus, the computational procedures for the construction and refinement of quotients described so far in this chapter do not apply directly to such systems. Instead, in the following we develop an analysis strategy for autonomous, uncertain parameter PWA systems based on the construction of over-approximation quotients.

Proposition 7.1 *Given a polytope $\boldsymbol{X}_l$, a convex over-approximation of $Post_{T_{\mathscr{W}}}(\boldsymbol{X}_l)$ can be computed as:*

$$\overline{Post}_{T_{\mathscr{W}}}(\boldsymbol{X}_l) = hull\{Ax \mid A \in V(\boldsymbol{P}_l^A), x \in V(\boldsymbol{X}_l)\} \oplus \boldsymbol{P}^c, \tag{7.6}$$

where hull() and $V()$ denote the convex hull and set of vertices, respectively (see Sect. A.1).

Proof Let $V(\mathbf{X}_l) = \{v_1, \ldots, v_R\}$ and $V(\mathbf{P}^A) = \{w_1, \ldots, w_M\}$. Let $x \in \mathbf{X}_l$ and $A \in \mathbf{P}_l^A$. Then $x = \sum_{r=1}^{R} \lambda_r v_r$, $A = \sum_{m=1}^{M} \mu_m w_m$, and

$$Ax = (\sum_{m=1}^{M} \mu_m w_m)(\sum_{r=1}^{R} \lambda_r v_r) = \sum_{m=1}^{M} \sum_{r=1}^{R} \mu_m \lambda_r w_m v_r$$

Since $\mu_m, \lambda_r \geq 0$ and $\sum_{m=1}^{M} \mu_m = \sum_{r=1}^{R} \lambda_r = 1$ then $\mu_m \lambda_r \geq 0$ for any m, r and $\sum_{m=1}^{M} \sum_{r=1}^{R} \mu_m \lambda_r = 1$. Therefore

$$Ax \in \text{hull}\{wv, w \in V(\mathbf{P}_l^A), v \in V(\mathbf{X})\}$$

and the rest of the proof follows from Definition A.7. ■

The computation from Eq. (7.6) produces an over-approximation of the reachable states from a given region and is the smallest convex set containing $Post_{T_{\mathscr{W}}}(\mathbf{X}_l)$:

$$Post_{T_{\mathscr{W}}}(\mathbf{X}_l) \subseteq \overline{Post}_{T_{\mathscr{W}}}(\mathbf{X}_l) \tag{7.7}$$

Although a precise distance between the real set and its over-approximation is hard to quantify, the volume of $Post_{T_{\mathscr{W}}}()$ was not significantly increased by the approximation for the systems we considered.

Using the over-approximation $\overline{Post}_{T_{\mathscr{W}}}(\mathbf{X}_l)$, instead of the exact $Post_{T_{\mathscr{W}}}(\mathbf{X}_l)$ an over-approximation quotient $\overline{T_{\mathscr{W}}/_\sim} = (Q_{\mathscr{W}}/_\sim, \overline{\delta_{\mathscr{W},\sim}}, O_{\mathscr{W}}, o_{\mathscr{W},\sim})$ can be constructed. From Eq. (7.7), it follows that for all $l \in Q_{\mathscr{W}}/_\sim$, we have $\delta_{\mathscr{W},\sim} \subseteq \overline{\delta_{\mathscr{W},\sim}}$, which leads to

$$\mathscr{L}_{T_{\mathscr{W}}} \subseteq \mathscr{L}_{T_{\mathscr{W}}/_\sim} \subseteq \mathscr{L}_{\overline{T_{\mathscr{W}}/_\sim}}. \tag{7.8}$$

Therefore, the over-approximation quotient $\overline{T_{\mathscr{W}}/_\sim}$ simulates the exact quotient $T_{\mathscr{W}}/_\sim$ and $\overline{T_{\mathscr{W}}/_\sim}$ can be used instead of $T_{\mathscr{W}}/_\sim$ for the methods we developed in Chap. 4 but the results become more conservative. The over-approximation quotient $\overline{T_{\mathscr{W}}/_\sim}$ can be computed through Algorithm 14 by substituting the $Post_{T_{\mathscr{W}}}()$ operation with its over-approximation $\overline{Post}_{T_{\mathscr{W}}}()$ when the matrix component of the parameters of $\mathscr{W}$ is uncertain. This leads to the computation of $X^{\phi}_{\overline{T_{\mathscr{W}}/_\sim}}$ where $X^{\phi}_{\overline{T_{\mathscr{W}}/_\sim}} \subseteq X^{\phi}_{T_{\mathscr{W}}/_\sim} \subseteq X^{\phi}_{T_{\mathscr{W}}}$, following from Eq. (7.8). While this is sufficient to apply some of the analysis strategies developed previously, additional refinement strategies are required to obtain less conservative analysis results but for autonomous, uncertain parameter PWA systems, the $Pre()$ operation is not easily computable. Instead, we apply a refinement approach, where a polytope $\mathbf{X}_l$ is split along each dimension e.g., through the center of the Chebyshev ball of $\mathbf{X}_l$ (Definition A.9). While this strategy is less efficient since the dynamics of the system are not taken into account during refinement, it allows an implementation through quad-tree data structures and their extensions into higher dimensions.

By constructing the over-approximation quotient in Algorithm 7 and using it within the analysis methods from Chap. 4 (e.g., Algorithm 8) together with the refinements strategy described above, a (more conservative) solution to Problem 7.1 is obtained even for autonomous, uncertain parameter PWA systems.

Example 7.4 Consider a two dimensional ($N = 2$) autonomous PWA system that has a total of nine rectangular regions $\mathbf{X}_1, \ldots, \mathbf{X}_9$ labeled by $L = \{1, 2, \ldots, 9\}$. The parameters for each region for an initial fixed parameter model (where $\mathbf{P}_l^A$, $\mathbf{P}_l^c$ are singletons A_l, c_l, respectively) are as follows:

$$A_1 = A_3 = A_9 = \begin{bmatrix} 0.82 & 0.00 \\ 0.00 & 0.67 \end{bmatrix}, \quad A_2 = A_8 = \begin{bmatrix} 0.82 & -0.37 \\ 0.00 & 0.67 \end{bmatrix},$$

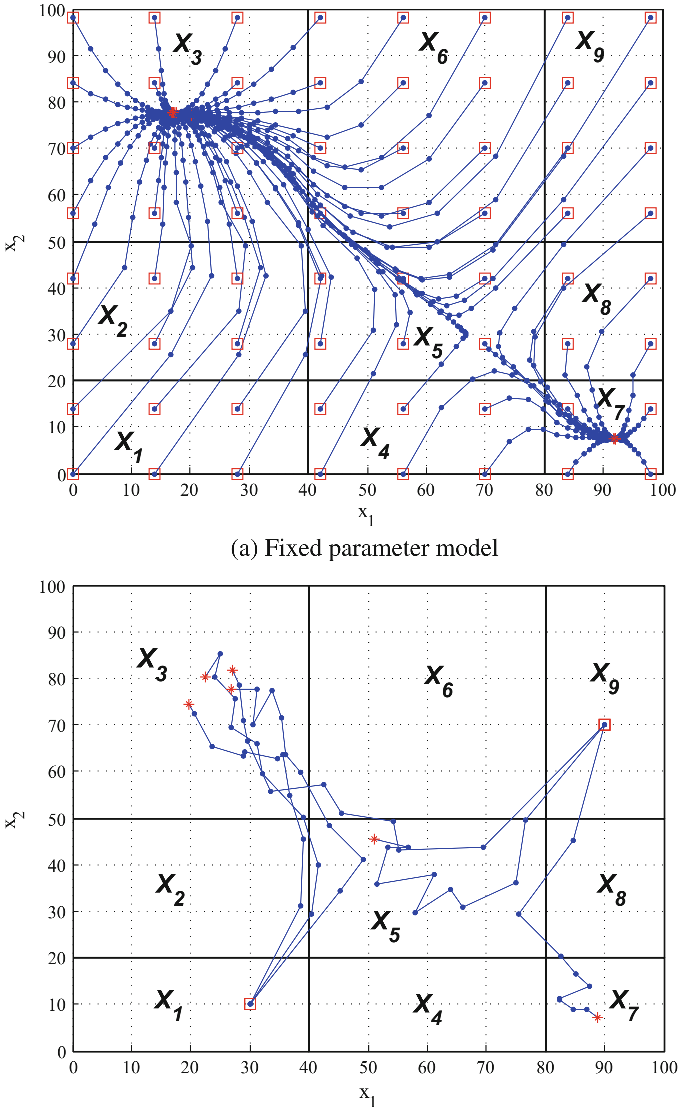
$$A_4 = A_6 = \begin{bmatrix} 0.82 & 0.00 \\ -0.52 & 0.67 \end{bmatrix}, \quad A_5 = \begin{bmatrix} 0.96 & -0.39 \\ -0.55 & 0.80 \end{bmatrix}, \quad A_7 = \begin{bmatrix} 0.82 & 0.00 \\ 0.00 & 0.67 \end{bmatrix},$$

$$c_1 = \begin{bmatrix} 16.68 \\ 25.55 \end{bmatrix}, \quad c_2 = \begin{bmatrix} 19.37 \\ 25.55 \end{bmatrix}, \quad c_3 = \begin{bmatrix} 3.08 \\ 25.55 \end{bmatrix},$$

$$c_4 = \begin{bmatrix} 16.68 \\ 43.34 \end{bmatrix}, \quad c_5 = \begin{bmatrix} 14.66 \\ 42.97 \end{bmatrix}, \quad c_6 = \begin{bmatrix} 3.08 \\ 47.65 \end{bmatrix},$$

$$c_7 = \begin{bmatrix} 16.68 \\ 2.47 \end{bmatrix}, \quad c_8 = \begin{bmatrix} 25.12 \\ 2.47 \end{bmatrix}, \quad c_9 = \begin{bmatrix} 3.08 \\ 2.47 \end{bmatrix}$$

Under the fixed parameters, dynamics 3 and 7 have unique, asymptotically stable equilibria inside rectangles $\mathbf{X}_3$ and $\mathbf{X}_7$ (see Fig. 7.5). An interesting problem is finding the regions of attraction for the two equilibria and exploring how those regions change when parameter uncertainty is introduced. By exploiting convexity properties of affine functions on polytopes, it can be easily proved that under the fixed parameters, $\mathbf{X}_3$ and $\mathbf{X}_7$ are invariants for dynamics 3 and 7, respectively. From this, we can immediately conclude that $\mathbf{X}_3$ and $\mathbf{X}_7$ are regions of attraction for the two equilibria. Therefore, our problem reduces to



(a) Fixed parameter model

(b) Uncertain parameter model

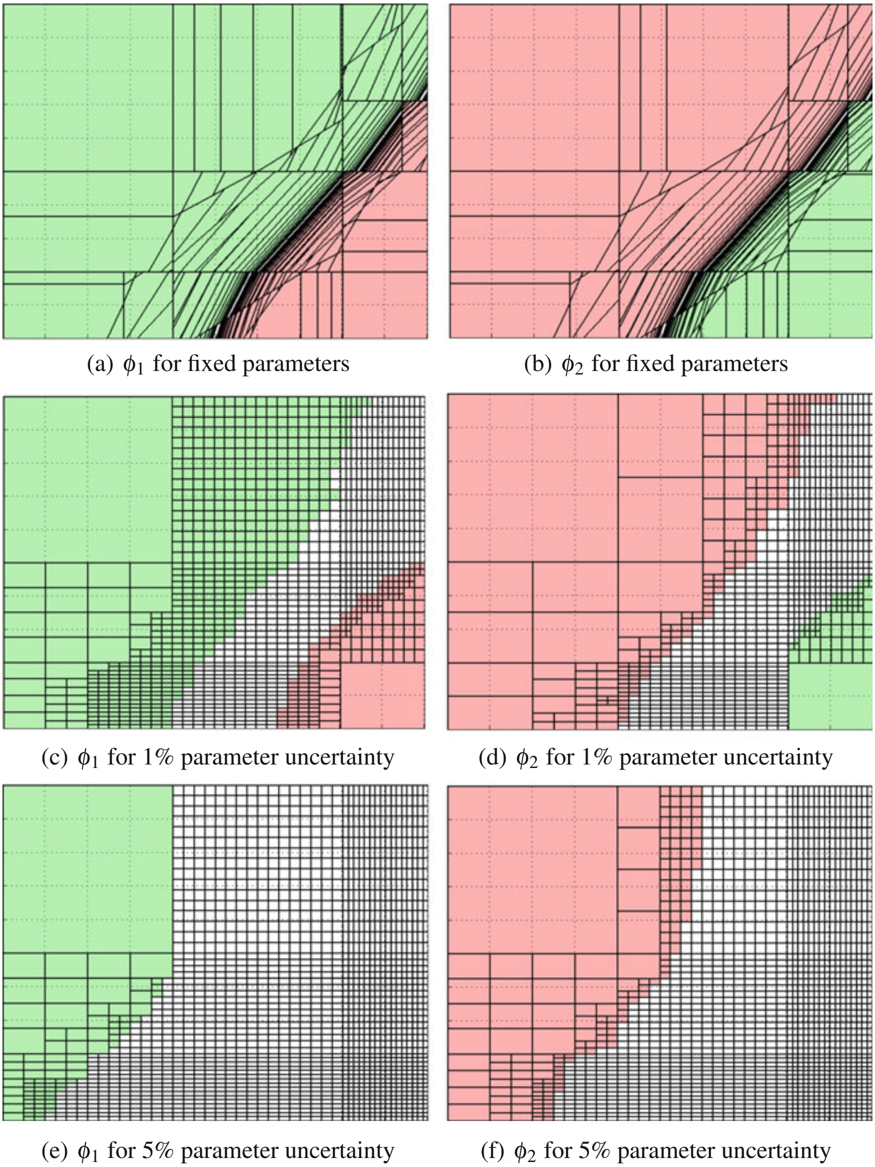
Fig. 7.5 Simulated trajectories of the autonomous PWA system from Example 7.4. Initial conditions are denoted by *red squares*

finding maximal regions satisfying LTL formulas $\phi_1 = \Diamond\Box 3$ and $\phi_2 = \Diamond\Box 7$. In other words, we want to find maximal sets of initial conditions, from which trajectories will eventually reach regions $\mathbf{X}_3$ or $\mathbf{X}_7$ and stay there forever.

To explore how the sizes of the attractor regions change, hyper-rectangular parameter uncertainty was introduced in the model by allowing each component of the parameters A_l and c_l for region $l \in L$ to vary in a range of size specified as a percentage of the fixed parameter value and centered around it (parameter components equal to 0 were also allowed to vary in a small range). Results from the computation with various levels of uncertainty are compared with the ones obtained under fixed parameters (Fig. 7.6). Because of the rectangular initial partition of the state space, 2^N-trees could be used as an efficient splitting strategy for the uncertain parameter case. Our method identifies only an attracting region for the equilibria at $\mathbf{X}_3$ for 5% uncertainty. As expected, increasing the level of uncertainty in the parameters decreases the size of the identified regions of state space (but a region identified at higher uncertainty is always a subset of the one identified at lower uncertainty).

Even if smaller limit ε is used, under parameter uncertainty it is possible that a subset of the state space is never included in the identified regions—a property resulting from nondeterminism introduced in the embedding transitions system. Even though complete partitioning of the state space might not be possible, decreasing ε provides further refinement and greater detail of the identified regions (initial iterations attempt to capture large satisfying (or violating) regions, while subsequent ones expand the solution less but provide greater resolution on its boundaries).

Example 7.5 Consider the PWA system from Example 7.4, where additive uncertainty is introduced by allowing each parameter c_l for region $l \in L$ to vary in a range of 10% of its original value (as before, parameter components equal to 0 were also allowed to vary in a small range). This corresponds to the PWA from Definition 6.5. As in Example 7.4, we are interested in identifying maximal regions satisfying LTL formulas $\phi_1 = \Diamond\Box 3$ and $\phi_2 = \Diamond\Box 7$. This problem can be approached by applying the refinement strategy for uncertain parameter systems as in Example 7.4, leading to the identification of the satisfying and violating regions presented in Fig. 7.7a, b. Alternatively, the specialized refinement strategy for additive uncertainty systems can be applied, resulting in the identification of the regions presented in Fig. 7.7c, d. Such a strategy leads to a finer partition, since at each step a region is refined by considering all combinations of regions reachable from it. However, this allows the identification of larger satisfying and violating regions for the same limit $\varepsilon = 5$, compared to the more general refinement strategy.


(a) ϕ_1 for fixed parameters

(b) ϕ_2 for fixed parameters

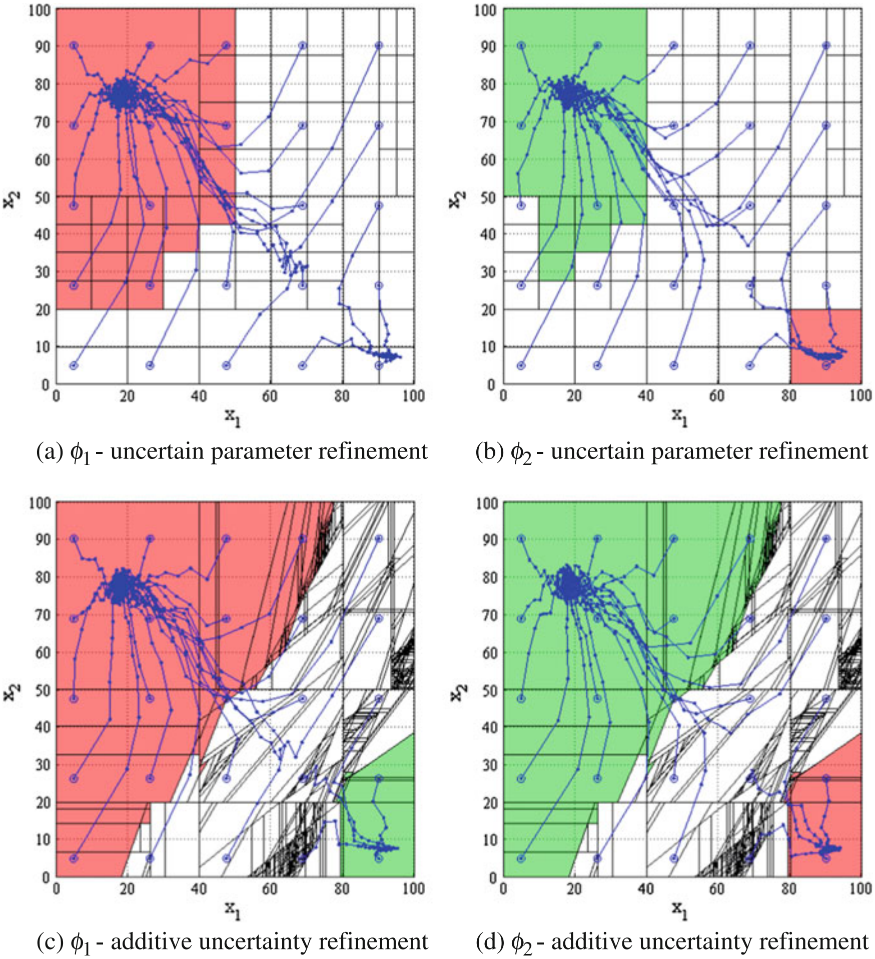
(c) ϕ_1 for 1% parameter uncertainty

(d) ϕ_2 for 1% parameter uncertainty

(e) ϕ_1 for 5% parameter uncertainty

(f) ϕ_2 for 5% parameter uncertainty

Fig. 7.6 Results for Example 7.4. Regions satisfying the formula are shown in *green*, while regions satisfying the negation of the formula are shown in *red*. The refinement exploits the system dynamics (using Pre) for the case when the parameters are fixed ((**a**) and (**b**)) and it uses a generic rectangular partitions when the parameters are uncertain ((**c**), (**d**), (**e**), and (**f**))



(a) ϕ_1 - uncertain parameter refinement

(b) ϕ_2 - uncertain parameter refinement

(c) ϕ_1 - additive uncertainty refinement

(d) ϕ_2 - additive uncertainty refinement

Fig. 7.7 The general refinement strategy for uncertain parameter systems and the specific refinement strategy for additive uncertainty systems were applied to the PWA system from Example 7.4 with 10% additive uncertainty. Regions satisfying the formula are shown in green, while regions satisfying the negation of the formula are shown in red. While the general strategy leads to fewer regions after refinement with the given limit $\varepsilon = 5$, the additive uncertainty refinement allows the identification of larger satisfying and violating regions

Example 7.6 Consider a three dimensional ($N = 3$) autonomous PWA system that has a total of 27 rectangular regions $\mathbf{X}_1, \ldots, \mathbf{X}_{27}$ labeled by $L = \{1, 2, \ldots, 27\}$. The parameters for each region for a fixed parameter model (where $\mathbf{P}_l^A$, $\mathbf{P}_l^c$ are singletons A_l, c_l, respectively) are as follows:

$$A_{1,3,7,9,19,21,25,27} = \begin{bmatrix} 0.67 & 0 & 0 \\ 0 & 0.67 & 0 \\ 0 & 0 & 0.67 \end{bmatrix}$$

$$A_{2,8,20,26} = \begin{bmatrix} 0.67 & 0 & 0 \\ 0 & 0.67 & 0 \\ 0 & -0.63 & 0.67 \end{bmatrix}$$

$$A_{4,6,22,24} = \begin{bmatrix} 0.67 & 0 & 0 \\ -0.63 & 0.67 & 0 \\ 0 & 0 & 0.67 \end{bmatrix}$$

$$A_{5,23} = \begin{bmatrix} 0.67 & 0 & 0 \\ -0.63 & 0.67 & 0 \\ 0.29 & -0.63 & 0.67 \end{bmatrix}$$

$$A_{10,12,16,18} = \begin{bmatrix} 0.67 & 0 & -0.63 \\ 0 & 0.67 & 0 \\ 0 & 0 & 0.67 \end{bmatrix}$$

$$A_{11,17} = \begin{bmatrix} 0.67 & 0.29 & -0.63 \\ 0 & 0.67 & 0 \\ 0 & -0.63 & 0.67 \end{bmatrix}$$

$$A_{13,15} = \begin{bmatrix} 0.67 & 0 & -0.63 \\ -0.63 & 0.67 & 0.29 \\ 0 & 0 & 0.67 \end{bmatrix}$$

$$A_{14} = \begin{bmatrix} 0.58 & 0.29 & -0.6 \\ -0.6 & 0.58 & 0.29 \\ 0.29 & -0.6 & 0.58 \end{bmatrix}$$

$$c_1 = \begin{bmatrix} 26 \\ 26 \\ 26 \end{bmatrix} \quad c_2 = \begin{bmatrix} 26 \\ 26 \\ 38 \end{bmatrix} \quad c_3 = \begin{bmatrix} 26 \\ 26 \\ 3.3 \end{bmatrix} \quad c_4 = \begin{bmatrix} 26 \\ 38 \\ 26 \end{bmatrix} \quad c_5 = \begin{bmatrix} 26 \\ 38 \\ 31 \end{bmatrix} \quad c_6 = \begin{bmatrix} 26 \\ 38 \\ 3.3 \end{bmatrix}$$

$$c_7 = \begin{bmatrix} 26 \\ 3.3 \\ 26 \end{bmatrix} \quad c_8 = \begin{bmatrix} 26 \\ 3.3 \\ 48 \end{bmatrix} \quad c_9 = \begin{bmatrix} 26 \\ 3.3 \\ 3.3 \end{bmatrix} \quad c_{10} = \begin{bmatrix} 38 \\ 26 \\ 26 \end{bmatrix} \quad c_{11} = \begin{bmatrix} 31 \\ 26 \\ 38 \end{bmatrix} \quad c_{12} = \begin{bmatrix} 48 \\ 26 \\ 3.3 \end{bmatrix}$$

$$c_{13} = \begin{bmatrix} 38 \\ 31 \\ 26 \end{bmatrix} \quad c_{14} = \begin{bmatrix} 33 \\ 33 \\ 33 \end{bmatrix} \quad c_{15} = \begin{bmatrix} 48 \\ 28 \\ 3.3 \end{bmatrix} \quad c_{16} = \begin{bmatrix} 38 \\ 3.3 \\ 26 \end{bmatrix} \quad c_{17} = \begin{bmatrix} 28 \\ 3.3 \\ 48 \end{bmatrix} \quad c_{18} = \begin{bmatrix} 48 \\ 3.3 \\ 3.3 \end{bmatrix}$$

$$c_{19} = \begin{bmatrix} 3.3 \\ 26 \\ 26 \end{bmatrix} \quad c_{20} = \begin{bmatrix} 3.3 \\ 26 \\ 38 \end{bmatrix} \quad c_{21} = \begin{bmatrix} 3.3 \\ 26 \\ 3.3 \end{bmatrix} \quad c_{22} = \begin{bmatrix} 3.3 \\ 48 \\ 26 \end{bmatrix} \quad c_{23} = \begin{bmatrix} 3.3 \\ 48 \\ 28 \end{bmatrix} \quad c_{24} = \begin{bmatrix} 3.3 \\ 48 \\ 3.3 \end{bmatrix}$$

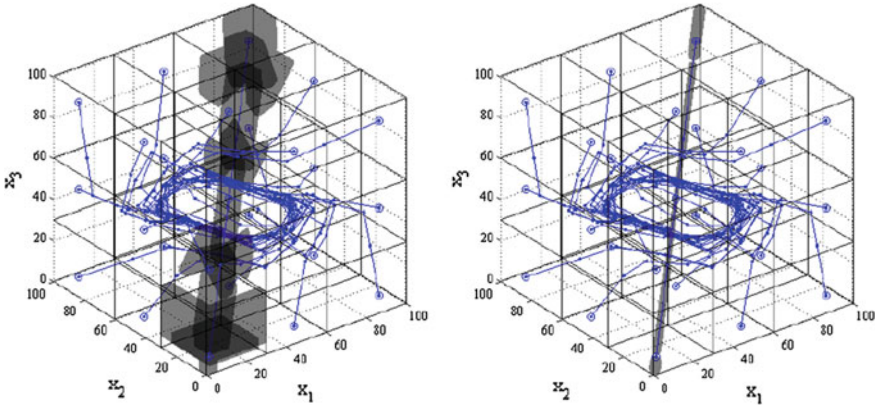
$$c_{25} = \begin{bmatrix} 3.3 \\ 3.3 \\ 26 \end{bmatrix} \quad c_{26} = \begin{bmatrix} 3.3 \\ 3.3 \\ 48 \end{bmatrix} \quad c_{27} = \begin{bmatrix} 3.3 \\ 3.3 \\ 3.3 \end{bmatrix}$$

The dynamics of the system are illustrated by simulated trajectories in Fig. 7.8. We are interested in testing whether all initial conditions lead to oscillatory behavior and define sub-formula ϕ, which is satisfied when the value of state variable 1 is above a certain threshold (i.e., ϕ is the disjunction of all regions $\mathbf{X}_l$ such that $\forall x \in \mathbf{X}_l$, $[1\ 0\ 0]x > 60$). By analyzing the system with LTL formula $\Box(\Diamond\phi \wedge \Diamond\neg\phi)$ we search for the maximal set of initial conditions guaranteeing that trajectories of the system keep oscillating between low and high values of the first component of the state. No violating region was identifying using the analysis approach, while most of the region $[0, 100]^3$ was identified as satisfying, with the exception of a small uncertain region shown in Fig. 7.8.

7.3 Formula-Guided Refinement

The analysis approach for PWA systems described in Sect. 7.1 is based on the construction and refinement of finite quotients inspired by bisimulation algorithms. As described in Sect. 4.5, an alternative approach to the analysis of large or infinite transition systems is to perform quotient refinement based on the exact specification for a particular problem. The goal in this case is to construct a formula-equivalent quotient (instead of a bisimilar one), which could be used for analysis instead of the original system, where all results are equivalent for the specific property of interest.

The approach from Sect. 4.5 can be implemented using polyhedral operations, utilizing the computations already described in Sect. 7.1, together with computational techniques described in Chap. 5. Using such a *formula-guided refinement* strategy,



(a) Refinement limit $\varepsilon = 5$

(b) Refinement limit $\varepsilon = 1$

Fig. 7.8 Results of the analysis in Example 7.6. Initial states of simulated trajectories are shown as circles. The specification is the system oscillates such that the value of the first component of the state goes above and under 60. Results for two refinement limits are shown. No violating region was identified for this property, while the largest satisfying region that was identified includes most of the state space $[0, 100]^3$, with the exception of the uncertain region shown in gray

the computation is performed on the product automaton and could lead to improved efficiency, since all refinement is guided by the specification, while the resulting refined system is simplified at each step. To illustrate this, the two approaches are compared in Example 7.7.

Example 7.7 To illustrate the differences between the analysis strategies described in Sects. 7.1 and 7.3, we apply these approaches to the two-dimensional ($N = 2$) PWA system introduced in Example 7.4 (denoted as $\mathscr{W}_2$ in the following) and the three-dimensional ($N = 3$) PWA system introduced in Example 7.6 (denoted as $\mathscr{W}_3$).

For $\mathscr{W}_2$, we seek the maximal sets of initial conditions guaranteeing that all trajectories of the system eventually reach regions $\mathbf{X}_3$ or $\mathbf{X}_7$, respectively, specified using LTL formulas $\Diamond 3$ and $\Diamond 7$.

For $\mathscr{W}_3$, we are interested in testing whether all initial conditions lead to oscillatory behavior and define sub-formulas ϕ_1 and ϕ_2, which are satisfied when state variable 3 is respectively low and high (i.e., ϕ_1 is the disjunction of all regions $\mathbf{X}_l$ such that $\forall x \in \mathbf{X}_l$, $[0\ 0\ 1]x < 30$ and, similarly, ϕ_2 is the disjunction of all regions $\mathbf{X}_l$ such that $\forall x \in \mathbf{X}_l$, $[0\ 0\ 1]x > 60$). By analyzing the system with LTL formula $\Box(\Diamond\phi_1 \wedge \Diamond\phi_2)$ we search for the maximal set of initial conditions guaranteeing that trajectories of the system keep oscillating between low and high values of state variable 3.

The results obtained by applying the analysis methods described in Sects. 7.1 and 7.3 to both systems are summarized in the following table (the numbers corresponding to the approach from Sect. 7.1 are given in parentheses). The reported computation times correspond to a 20-iteration limit. The relative volumes (as a percentage of the total state space) of the identified satisfying and violating regions are reported as "% Satisfaction" and "% Violation", respectively. The number of states in the initial quotient and the quotient after all refinement steps are reported as $|X/_\sim|$ and $|\hat{X}/_\sim|$, respectively. The number of states in the product automaton after all refinement, updating and simplification is reported as $|\hat{S}_P|$ (no product simplification is involved for the method described in Sect. 7.1 and the number of states of the product automaton with the specification and the negation of the specification are reported separately). The approach from Sect. 7.3 allows us to consider only a single product automaton, unlike the one from Sect. 7.1, where both the product automaton with the formula and its negation are constructed. In addition, the minimizations of this product automaton keep the number of the states to be considered low. As a result, the computation times are improved significantly by using the formula-guided approach, while the results (in terms of identified satisfying and violating regions) are comparable between the two methods.

System	$\mathscr{W}_2$		$\mathscr{W}_3$
Specification	$\diamond 3$	$\diamond 7$	$\square(\diamond \phi_1 \wedge \diamond \phi_2)$
Computation time	29 (97) s	28.9 (96) s	17 (153) min
% Satisfaction	79.7 (79.8)	20.1 (20.1)	99.62 (99.76)
% Violation	20.1 (20.1)	79.7 (79.8)	0 (0)
$\lvert X/_\sim \rvert$	9	9	27
$\lvert \hat{X}/_\sim \rvert$	423 (463)	419 (459)	2845 (2970)
$\lvert \hat{S}_P \rvert$	7 (926,463)	7 (918,459)	17 (8910,8910)

7.4 Notes

In this chapter, which is based on [177, 181–183, 185], we showed that the methods developed in Chap. 4 for finite transition systems can be extended to autonomous discrete-time PWA systems with parameter uncertainties. We developed a method that attempts to find the largest region of initial states from which such a system satisfies an LTL formula. Motivated by the fact that finite bisimulations only exist for very limited classes of dynamical systems [83–86], central to our approach is the notion of simulation. Related, probabilistic versions of this method can be found in [1, 96, 115].

There are several simplifying assumptions that we made for simplicity of computation and presentation. First, we assumed that the LTL specification is given over the set of symbols labeling the polytopes from the definition of the PWA system (Definition 6.3). Note that, as suggested in Example 7.1, this can be easily relaxed to allow for formulas over arbitrary predicates in the state variables of the system. Indeed, given a set of such predicates, a finer partition can be constructed by adding polytopes and labeling them according to the satisfaction of the predicates (see also Sect. 1.2).

Second, both in the definition of the system (Definitions 6.3, 6.4, and 6.5, which are particular cases of 6.1) and its semantics (Definitions 6.6 and 6.7), we considered only open, full dimensional polytopes as discussed in Sect. 6.3. Third, we assumed that the A_l-matrices of the system were non-singular. While seemingly restrictive, this assumption was made purely for simplicity of presentation. The implementation of the $Post$ and Pre operators can be easily extended to semi-linear sets and singular affine functions as discussed in Sect. A.4.

While in this chapter we focused on the problem of finding largest sets of satisfying initial states, the results presented here can be used to perform "classical" LTL model checking of PWA using standard tools such as SPIN [89], NuSMV [43], PRISM [114], or DiVinE [18]. Given a PWA system $\mathscr{W}$, the exact finite quotient $T_{\mathscr{W}}/_\sim$ or the over-approximation finite quotient $\overline{T_{\mathscr{W}}/_\sim}$ can be constructed as described in Sects. 7.1 and 7.2. Then, $T_{\mathscr{W}}/_\sim$ or $\overline{T_{\mathscr{W}}/_\sim}$ can be checked against an LTL formula ϕ over the index set L of $\mathscr{W}$. In addition, the special observation Out can be used as an atomic proposition in ϕ. As a simple example, consider the problem of guaranteeing that region $\mathbf{X}$ is an invariant for all trajectories of a PWA system $\mathscr{W}$. We formulate the specification $\phi = \square\neg$Out requiring that trajectories of the system never visit the region labeled by Out. In other words, satisfying trajectories of $\mathscr{W}$ will never leave $\mathbf{X}$. We can model check the quotient $T_{\mathscr{W}}/_{\sim}$ against ϕ using standard tools and if $T_{\mathscr{W}}/_{\sim}$ satisfies ϕ we can guarantee that all trajectories of $\mathscr{W}$ satisfy the specification (the same is true for the over-approximation quotient $\overline{T_{\mathscr{W}}/_{\sim}}$). In subsequent chapters we will use this strategy to guarantee that trajectories of $\mathscr{W}$ do not leave the defined state space of the system.

It is important to note that simply model checking the quotient $T_{\mathscr{W}}/_{\sim}$ (and especially the over-approximation $\overline{T_{\mathscr{W}}/_{\sim}}$) is restrictive as discussed in Chap. 4, which motivated the development of additional refinement strategies. While the satisfaction of the formula can be guaranteed for all trajectories of $\mathscr{W}$ when the quotient satisfies the formula, nothing can be guaranteed if the formula is violated. Since, in general, $T_{\mathscr{W}}/_{\sim}$ is coarse (it contains few states), positive verification results can be rarely obtained. In Chap. 4, we extended the standard model checking methods in order to obtain more informative results. In this chapter, we showed that the construction of quotients is possible for PWA systems and the techniques from Chap. 4 can, therefore, be extended to such systems.

For the computation of an over-approximation of the set of states reachable from a certain region in a PWA system with parameter uncertainty, we defined the $\overline{Post}$ operator. Intuitively, in this treatment we assume that the parameter uncertainty is inherent in the dynamics of the system and must be handled as such as part of the analysis problem. An alternative formulation, where such uncertainty is considered as an allowed range in which the system's parameters can be tuned (corresponding to a parameter synthesis problem) is presented in Chap. 8. A treatment related to Eq. (7.6) can be found in [16], where it is shown that this over-approximation is the smallest convex set containing the states reachable from a given region.

The methods presented in this chapter involve model checking of the finite quotient $T_{\mathscr{W}}/_{\sim}$ at each step of the iterative procedure. Even though the worst case complexity of LTL model checking is exponential in the size of the formula, this upper limit is rarely reached in practice. We use an in-house model checker, which allows us to model check $T_{\mathscr{W}}/_{\sim}$ from specific states only and perform computation (such as the construction of Büchi automata) only once instead of recomputing at each step.

The construction and refinement of finite quotients used in our approach is based on polyhedral operations, which also have an exponential upper bound. Therefore, the applicability of the method depends on controlling the number of states as refinement progresses. When applied to a state X, the refinement procedure Refine($T_{\mathscr{W}}/_{\sim}$, X) can, in general, produce a maximum of 2^k subsets, where $k = |Post_{T_{\mathscr{W}}/_{\sim}}(X)|$ is the number of states reachable from state X. In the particular case when the parameters of the PWA system are fixed, only k subsets can be produced. To limit the explosion in the number of states in the quotient, we only refine states when this can improve the solution. Even so, due to its inherent complexity, this method is not suitable for the analysis of systems in high dimensions or when many iterations are required to find a solution. As expected, the method performs best if large portions of the state space can be characterized as satisfying the formula or its negation during earlier iterations.

The formula-guided refinement method for PWA systems presented in Sect. 7.3 is based on the approach for constructing formula-equivalent quotients from Sect. 4.5. The main advantage of this approach is that the specification is incorporated directly as part of the refinement procedure and the computation is performed on the product automaton, which allows for certain optimizations. Compared to the iterative refinement approaches presented in Sect. 7.1, this procedure aims at constructing formula-equivalent abstractions that might be coarser than bisimulation for certain systems (conditions under which the analysis results from the formula-guided refinement approach are exact are described in Sect. 4.5 although, in general, the overall procedure is still conservative). Additional notes on the formula-guided refinement approach (in the context of transition systems rather than PWA systems) are available in Sect. 4.6.

The analysis of PWA systems for properties such as stability, invariance and reachability has also been considered elsewhere (e.g., [28]) but the approaches presented in this chapter allow greater expressivity through LTL specifications.

The algorithms presented in this chapter were implemented in Matlab and made available for download at http://sites.bu.edu/hyness/fapas/. The tools, called FAPAS (short for "Formal analysis of Piecewise Affine Systems") and FFAPAS (short for "Formula-guided Formal analysis of Piecewise Affine Systems"), use the MPT Toolbox [113] for polyhedral operations and the LTL2BA package [65] for the construction of Büchi automata from LTL formulas. For the two-dimensional ($N = 2$) case study presented in Example 7.4, the computation using FAPAS required under 20 s for the fixed parameter case, and under 10 min for all the uncertain parameter cases, where the limit on refinement was set to $\varepsilon = 1$ and $\varepsilon = 5$. For the three-dimensional ($N = 3$) case study presented in Example 7.6, the computation required under 20 min where $\varepsilon = 5$. This computation was performed on a 3.4 GHz machine with 1 GB of memory. The evaluation presented in Example 7.7, where the iterative and formula-guided refinement analysis approaches (described in Sects. 7.1 and 7.3) were compared, was performed on a 3.0 GHz machine with 4GB of memory.