Airplane model contains multiple environment components as *M3_Planes* and SUT *M3_Controller*. To apply the proposed pattern on this model, it is duplicated and merged into one system. And the new system contains two bisimilar controllers called *M3_Controller* and *M3_Controller_* and duplicated environments *M3_Planes* (Figure 3) and *M3_Planes_* (Figure 4). After merging the models, the pattern is applied to synchronize *M3_Controller* (Figure 1) and *M3_Controller_* (Figure 2) based on IO actions and similarly on the environments.

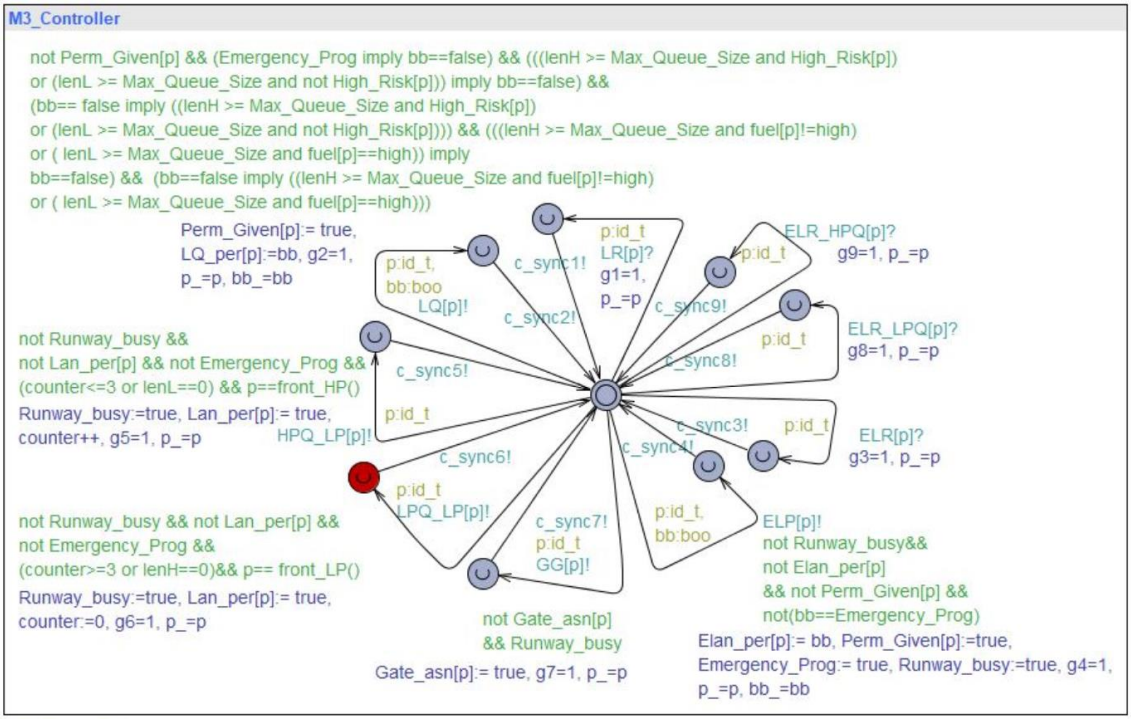

Figure 1. Airplane System M3_Controller

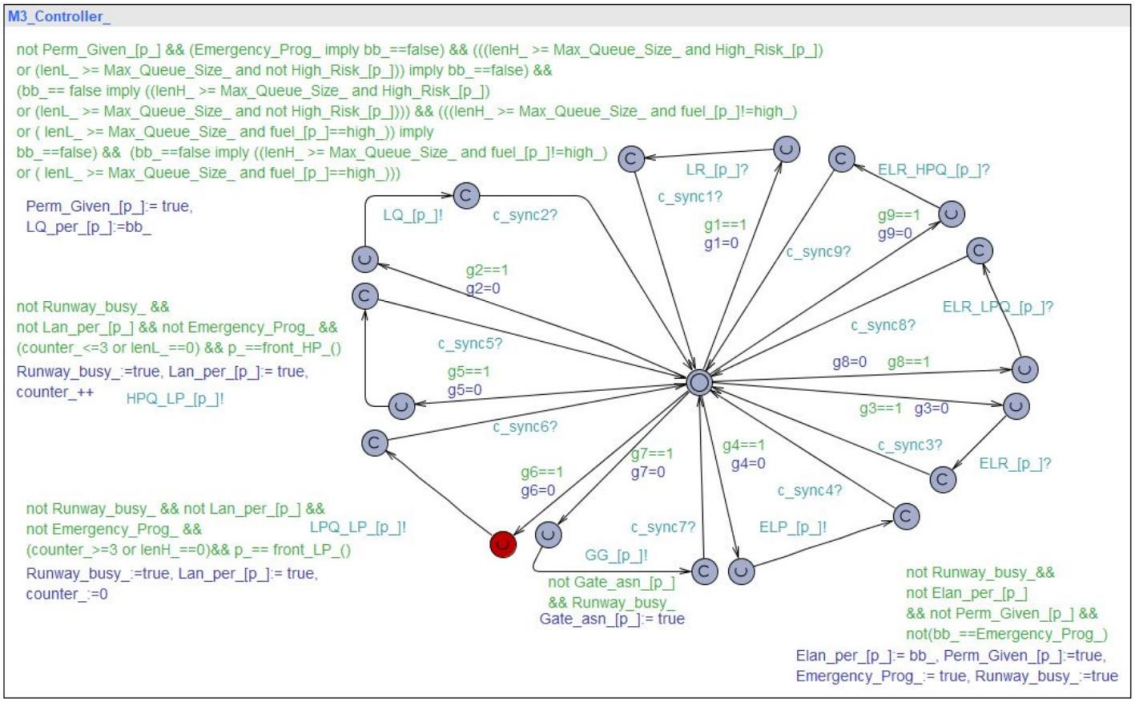

Figure 2. Airplane System M3_Controller_

In this example, for a stronger bisimulation, the transitions where there are no I/O actions, a synchronization channel is used on the original transition as shown in Figure 24 and 25. Currently, there is a deadlock in the system when the last two pair of *M3_Planes* and *M3_Planes_* instances are queued after *Waiting_L_Permission* state in both environments. The pattern tries to execute the *LPQ_LP* channel on *Low_PQ* state and synchronize one pair of Planes instances but at the same time another pair is given permission to proceed towards *High_PQ*. The transition from *Waiting_L_Permission* to *High_PQ* increments the value of *LenH* variable which is used in a guard condition in both controllers when queuing the *LPQ_LP* but *M3_Controller_* (See Figure 2) gets stuck there because of the *LenH* value change in the middle of the pattern execution for the other pair and vice versa scenario when *Waiting_L_Permission* to *Low_PQ* changes *LenL* and pattern is executing *HPQ_LP*.

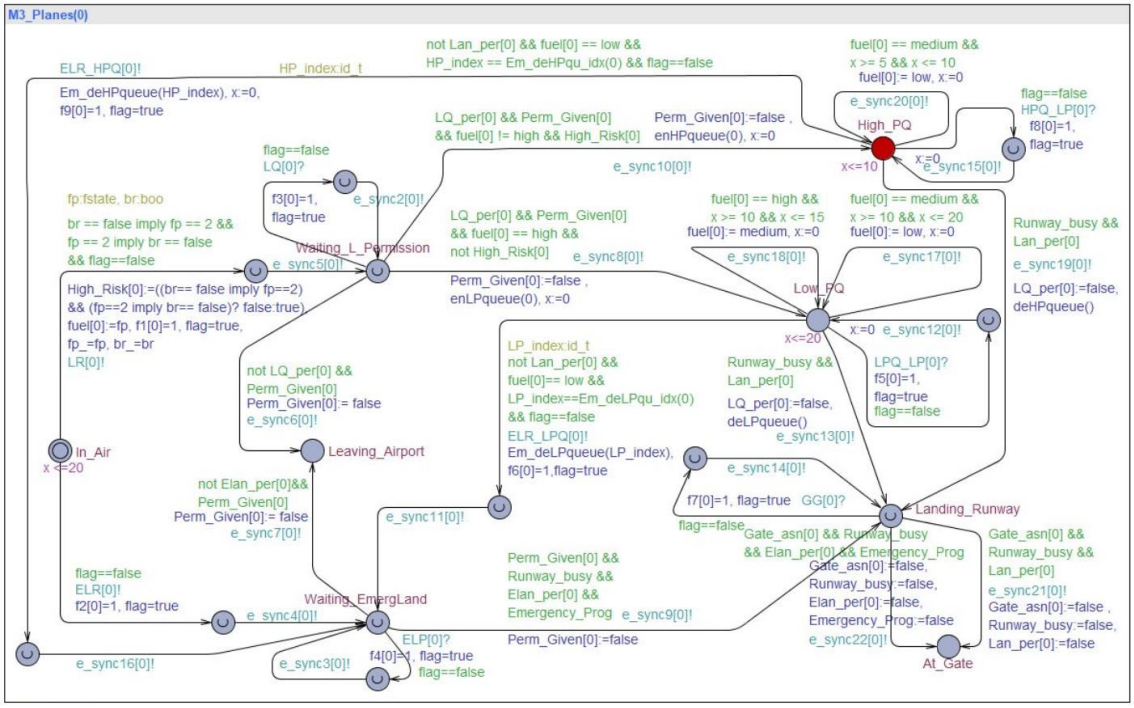

Figure 3. Airplane System M3_Planes

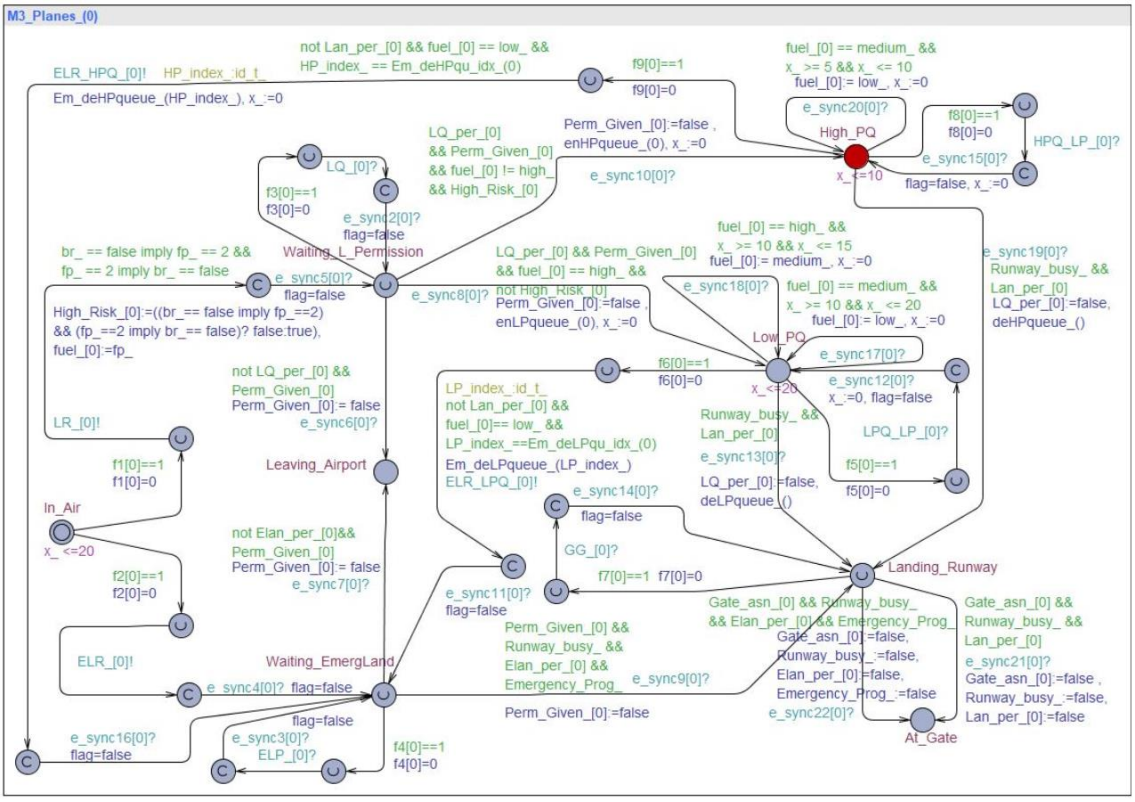

Figure 4. Airplane System M3_Planes_

Shokri-Manninen, Fatima & Tsiopoulos, Leonidas & Vain, Juri & Waldén, Marina. (2020). Integration of iUML-B and UPPAAL Timed Automata for Development of Real-Time Systems with Concurrent Processes. 10.1007/978-3-030-48077-6_13.