Appendix A

DTD Files

The Document Type Definition (DTD) files for the XML input files for PHAVer and Spin are presented here. These DTD files list out the necessary and possible elements for the given type of XML file. These files create a standard for the XML input files; they check that the necessary information is entered in the proper way.

A.1 PHAVer

```
<?xml version="1.0"?>

<!ELEMENT inputs (smc,cdsv,usv,goals,tcs,unsafe)>
<!ELEMENT smc (#PCDATA)>

<!ELEMENT cdsv (sv+)>
<!ELEMENT sv (name, constraint+, initial_condition)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT constraint (type_name,merge+,trans,dyn_eqn,reset?)>
<!ELEMENT type_name (#PCDATA)>
<!ELEMENT merge (constraint_type,merge_condition,merge_type?,
                  merged_constraint*)>
<!ELEMENT constraint_type (#PCDATA)>
<!ELEMENT merge_condition (#PCDATA)>
<!ELEMENT merge_type (#PCDATA)>
<!ELEMENT merged_constraint (#PCDATA)>
```

```
<!ELEMENT trans (entrylogic,exitlogic)>
<!ELEMENT entrylogic (#PCDATA)>
<!ELEMENT exitlogic (#PCDATA)>
<!ELEMENT dyn_eqn (#PCDATA)>
<!ELEMENT reset (#PCDATA)>
<!ELEMENT initial_condition (#PCDATA)>

<!ELEMENT usv (svu+)>
<!ELEMENT svu (name,trans_type,parameter*,input_var*,transition*,
               dynamics+,initial_condition)>
<!ELEMENT trans_type (#PCDATA)>
<!ELEMENT input_var (#PCDATA)>
<!ELEMENT parameter (#PCDATA)>
<!ELEMENT transition (start,end,condition)>
<!ELEMENT start (#PCDATA)>
<!ELEMENT end (#PCDATA)>
<!ELEMENT condition (#PCDATA)>
<!ELEMENT dynamics (value,eqn,reset?,condition)>
<!ELEMENT value (#PCDATA)>
<!ELEMENT eqn (#PCDATA)>


<!ELEMENT goals (goal+,pgoal+)>
<!ELEMENT goal (name,tp_start,tp_end,constraint_sv?,constraint_type?,
               constraint_value*,tactic*)>
<!ELEMENT pgoal (name,tp_start,tp_end,constraint_sv,constraint_type,
                   constraint_value)>
<!ELEMENT tp_start (#PCDATA)>
<!ELEMENT tp_end (#PCDATA)>
<!ELEMENT constraint_sv (#PCDATA)>
<!ELEMENT constraint_value (#PCDATA)>
<!ELEMENT tactic (goalname+,failure*)>
```

```
<!ELEMENT goalname (#PCDATA)>
<!ELEMENT failure (logic,destination)>
<!ELEMENT logic (#PCDATA)>
<!ELEMENT destination (#PCDATA)>

<!ELEMENT tcs (tconstraint*)>
<!ELEMENT tconstraint (start,end,min,max)>
<!ELEMENT max (#PCDATA)>
<!ELEMENT min (#PCDATA)>

<!ELEMENT unsafe (uset+)>
<!ELEMENT uset (ucons+)>
<!ELEMENT ucons (svc,type,constr)>
<!ELEMENT svc (#PCDATA)>
<!ELEMENT type (#PCDATA)>
<!ELEMENT constr (#PCDATA)>}
```

A.2 Spin

```
<?xml version="1.0"?>

<!ELEMENT inputs (smc,cdsv,usv,goals,tcs)>
<!ELEMENT smc (#PCDATA)>

<!ELEMENT cdsv (dt?,sv+)>
<!ELEMENT dt (#PCDATA)>
<!ELEMENT sv (name, constraint+, initial_condition)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT constraint (type_name,merge+,trans,dyn_eqn,reset?)>
<!ELEMENT type_name (#PCDATA)>
<!ELEMENT merge (constraint_type,merge_condition,merge_type?,
                    merged_constraint*)>
```

```
<!ELEMENT constraint_type (#PCDATA)>
<!ELEMENT merge_condition (#PCDATA)>
<!ELEMENT merge_type (#PCDATA)>
<!ELEMENT merged_constraint (#PCDATA)>
<!ELEMENT trans (entrylogic,exitlogic)>
<!ELEMENT entrylogic (#PCDATA)>
<!ELEMENT exitlogic (#PCDATA)>
<!ELEMENT dyn_eqn (#PCDATA)>
<!ELEMENT reset (#PCDATA)>
<!ELEMENT initial_condition (#PCDATA)>

<!ELEMENT usv (svu+)>
<!ELEMENT svu (name,trans_type,input_var*,transition*,dynamics+,
               initial_condition)>
<!ELEMENT trans_type (#PCDATA)>
<!ELEMENT input_var (#PCDATA)>
<!ELEMENT transition (start,end,condition)>
<!ELEMENT start (#PCDATA)>
<!ELEMENT end (#PCDATA)>
<!ELEMENT condition (#PCDATA)>
<!ELEMENT dynamics (value,eqn?,reset?,condition?)>
<!ELEMENT value (#PCDATA)>
<!ELEMENT eqn (#PCDATA)>


<!ELEMENT goals (goal+)>
<!ELEMENT goal (name,tp_start,tp_end,constraint_sv?,constraint_type?,
               constraint_value*,tactic*)>
<!ELEMENT tp_start (#PCDATA)>
<!ELEMENT tp_end (#PCDATA)>
<!ELEMENT constraint_sv (#PCDATA)>
<!ELEMENT constraint_value (#PCDATA)>
```

```
<!ELEMENT tactic (goalname+,startsin,failure+)>
<!ELEMENT goalname (#PCDATA)>
<!ELEMENT startsin (#PCDATA)>
<!ELEMENT failure (logic,destination)>
<!ELEMENT logic (#PCDATA)>
<!ELEMENT destination (#PCDATA)>

<!ELEMENT tcs (tconstraint*)>
<!ELEMENT tconstraint (start,end,min,max)>
<!ELEMENT max (#PCDATA)>
<!ELEMENT min (#PCDATA)>
```

Glossary

branch goal goal with no child goals in group. 35

compatible goals that are not elaborated into different tactics of the same parent goal. 31

complete system state state that includes the actual and estimated state of each uncertain state variable. 77

completion goal controlled goal with a transition constraint. 11

completion time minimum length of a nominal execution path for a group. 81

consistent constraints that can be executed concurrently. 31

contribution value normalized contribution of a location towards a completion task. 82

controllable state variable state variable associated directly with a command class. 21

controlled goal constraint that causes a command to be issued to the system. 10

dependent state variable state variable not associated with a command class but dependent on controllable or dependent state variables. 21

elaboration act of choosing a control tactic out of those available. 9

executable branch set of goals in a goal tree that can be executed concurrently. 61

executable set set of goals that can execute concurrently. 32

failure path execution path that includes a complete system state that is unsafe. 82

goal constraint on a state variable in time. 9

group set of goals that are active between consecutive time points or the corresponding set of locations. 30

linear hybrid automaton systems with discrete modes of execution that have different continuous behavior. 15

location discrete mode of execution. 15

nominal execution path set of nominal complete system states that represents the complete, safe execution of a group. 81

non-uniform completion group whose execution time depends on the states visited. 82

passive goal constraint with no associated command. 10

root goal goal with no parent. 24

sibling goal goals elaborated into the same tactic. 24

state variable states of the system or environment. 9

state-based transitions goal network or hybrid system that has control tactics or modes for every modeled passive state. 58

tactic control mode or method. 11

uncontrollable state variable state variable not associated with any command class and not dependent on any controllable or dependent state variable. 21

uniform completion group whose execution time does not depend on the states visited. 82

unsafe set set of conditions that should never be reached by a goal network or hybrid system execution. 79