

Tabelas Comparativas – Mundo dos Blocos

Conceito	STRIPS	Prolog Estendido	NuSMV	Justificativa
Propriedades do bloco	block(X)	tamanho(X, W)	DEFINE size_a := 1;	Definir tamanhos fixos como constantes melhora eficiência (não variam no tempo)
Mesa / Slots	place(N)	table_slot(N), table_width(M)	{left, middle, right} ou array[slots]	Mapear slots como valores enumerados/arrays para raciocinar sobre posições
Posição de blocos	on(Block, Obj)	pos(Block, table(X)) ou pos(Block, on(OtherBlock))	VAR on_a : {table1, table2, b, c};	Substitui on/2 abstrato por variáveis enumeradas que assumem valores discretos de suporte
Espaço livre	clear(Object)	clear(Block), is_free(Slot, State)	clear_a, clear_b, etc. (booleanos)	Simplifica checagem de pré-condições no modelo de transição
Operador de movimento	move(Block, From, To)	move(Block, table(X)) ou move(Block, on(OtherBlock))	move : {move_a.b, move_c.table2, none,...}	No NuSMV ações são enumeradas; cada transição define pré-condições e efeitos
Pré-condições – Mobilidade	clear(Block)	clear(Block) deve ser verdade	TRANS exige clear_x = TRUE	Verifica se o bloco pode ser movido
Pré-condições – Validade lógica	Block \neq Target-Block	neq(Block, TargetBlock)	Implícito (TRUE)	Evita que um bloco seja colocado sobre si mesmo
Pré-condições – Acessibilidade do alvo	clear(Target)	clear(TargetBlock)	TRANS exige clear_y = TRUE	Garante que o destino está livre
Pré-condições – Estabilidade	Não existe	tamanho(Block) \leq tamanho(Target)	Regras de transição restringem	Impede blocos maiores sobre menores
Pré-condições – Ocupação espacial	Não existe	is_free(Slot, State)	Arrays de posição e regras de exclusão	Garante que há espaço contíguo na mesa
Estados iniciais e meta	Lista de on/clear	EstadoInicial = [pos(...), clear(...)]	INIT ... e CTL-SPEC !EF(goal)	No NuSMV, a meta é formulada como propriedade a ser verificada
Estratégia de planejamento	Busca em espaço de estados	Regressão de metas	Contraexemplo de NuSMV = plano	O plano é o traço do modelo que viola a especificação

Tipo de Restrição	STRIPS	Prolog Estendido	NuSMV	Justificativa
Mobilidade	clear(Block)	clear(Block) deve estar no estado	TRANS exige clear_x = TRUE	Um bloco só pode ser movido se estiver livre
Acessibilidade do alvo	clear(Target)	clear(TargetBlock)	TRANS exige clear_y = TRUE	O destino deve estar livre
Validade lógica ()	Implícito, não suportado nativamente	neq(Block, TargetBlock)	Implícito (TRUE)	Evita movimentos inválidos (ex.: sobre si mesmo)

Estabilidade	Não suportado	$\text{tamanho}(\text{Block}) \leq \text{tamanho}(\text{Target})$	Restrições nas transições	Impede blocos maiores sobre menores
Ocupação espacial	Não suportado	$\text{is_free}(\text{Slot}, \text{State}), \text{space_check}$	Arrays e exclusão de posições	Evita sobreposição de blocos
Identidade dos blocos	Implícito	$\text{not}(\text{on}(\text{X}, \text{X}))$	TRANS impede self-loops	Bloco não pode estar sobre si mesmo
Unicidade de posição	Implícito (não representado)	$\text{unique_position}(\text{Block})$	Variável on_X assume um único valor	Um bloco só pode estar em um lugar por vez
Meta de atingibilidade	Não considerado	Definido por regressão de metas	CTLSPEC EF(goal) ou contraexemplo	Nem toda meta é atingível; precisa de checagem
Sequência válida	Estrutura de pré-condições	Pré-condições explícitas	Cada TRANS aplica apenas se válido	Garante que só ações coerentes sejam executadas
Limite da mesa	Não suportado	Representação de slots discretos	Array de slots $\{1..N\}$	Não permite colocar blocos fora da mesa
Integridade do estado	Não suportado	Regras adicionais no domínio	Invariantes lógicos INVAR	Evita contradições em estados
Consistência temporal	Não suportado	—	Temporal logic (CTL/LTL)	Garante que o plano seja uma sequência de transições válidas