TABLE I: Symbolic constraints. This table shows how we generate symbolic constraints according to the type of a Condition and whether it is satisfied. $IllegalOps$ denotes the bytes which cannot correspond to an opcode. $Types$ denotes all valid types. $Op_{defined}$is the correct opcode of the instruction. $S[i]$ refers to the i_{th} operand on the top of the stack.

Constraint	Meaning	Symbolic expression
$StackCond(N, T)$, T is specific	There are N operands with type T on the stack top	$S[i].type = T, i \in \{1, ...N\}$
$StackCond(N, T)$, T is unspecific	There are N operands on the stack top	$S[i].type \in Types, i \in \{1, ...N\}$
$\neg StackCond(N, T)$, T is specific	There are N operands with unexpected type $\overline{T}$ on the stack top	$S[i].type = \overline{T}, \overline{T} \neq T, \overline{T} \in Types, i \in \{1, ...N\}$
$\neg StackCond(N, T)$, T is unspecific	The number of operands on the stack top is smaller than N	$S[i].type \in Types, i \in \{1, ...N-1\}$
$SameTypeCond(N)$	N operands on the stack top are of the same type	$\forall S[i].type = S[1].type, S[i].type \in Types, i \in \{1, 2, ..., N\}$
$\neg SameTypeCond(N)$	N operands on the stack top are of different types	$\exists S[i].type \neq S[1].type, S[i].type \in Types, i \in \{1, 2, ..., N\}$
$OpDefinedCond(Op)$	Op is defined	$Op = Op_{defined}$
$\neg OpDefinedCond(Op)$	Op is undefined	$Op \in IllegalOps$
$EqualCond(V_1, V_2)$	V_1 is equal to V_2	$V_1.value = V_2.value$
$\neg EqualCond(V_1, V_2)$	V_1 is not equal to V_2	$V_1.value \neq V_2.value$
$ExprCond(Expr)$	An equation or inequality $Expr$ holds	$Expr$
$\neg ExprCond(Expr)$	An equation or inequality $Expr$ does not hold	$\neg Expr$
$ExistCond(Elem(Instance, idx))$	The $Instance[idx]$ exists	$idx < Instance.len$
$\neg ExistCond(Elem(Instance, idx))$	The $Instance[idx]$ does not exist	$idx >= Instance.len$
$CompareCond(V1, V2, R)$	$V1$ and $V2$ hold the comparison relation R	$V1$ R $V2$ hold the comparison relation R
$\neg CompareCond(V1, V2, R)$	$V1$ and $V2$ do not hold the comparison relation R	$V1$ and $V2$ do not hold the comparison relation R