Unsolvability results for Turing machines arise because Turing machines can accept Turing machine descriptions as inputs. Paradoxes arise because a Turing machine can read its own description.

These unsolvability results require Turing machines to be encoded as Turing machine inputs. This can easily be done as follows:

We consider an integer $i$ as an encoding of a Turing machine $T_i$. This is done by considering $i$ as a binary number and breaking this binary number into 8 bit bytes. Each byte is interpreted as a character, and then $i$ is read as a Turing machine description in some language (such as the book gives). If $i$ does not encode a Turing machine then $T_i$ is some fixed Turing machine, possibly the Turing machine that immediately halts when it starts.

In this formalism, a **universal Turing machine** $U$ takes as input $i$ and $j$ and simulates $T_i$ on input $j$.

It’s actually easier conceptually to let $\text{encode}(M)$ be an encoding of a Turing machine as a string, and $\text{encode}(x)$ be the encoding of a string $x$. Then the universal Turing machine takes as input $\text{encode}(M)\text{encode}(x)$ and simulates $M$ on input $x$.

We can use this formalism to show that some problems are undecidable. Consider the language $L = \{\text{encode}(M) : M \text{ loops on input } \text{encode}(M)\}$. A simple argument shows that there is no Turing machine that partially decides $L$. For suppose $T$ partially decided $L$. What would $T$ do on input $\text{encode}(T)$? If $T$ halts on input $\text{encode}(T)$ then (since $T$ partially decides $L$), $\text{encode}(T) \in L$, so by definition of $L$, $T$ does not halt on input $\text{encode}(T)$. If $T$ does not halt on input $\text{encode}(T)$ then (since $T$ partially decides $L$) $\text{encode}(T) \notin L$, so $T$ halts on input $\text{encode}(T)$ by definition of $L$. Either choice leads to a contradiction. So $L$ is not partially decidable.

The **halting problem** is, given a Turing machine and an input, to decide whether the Turing machine halts on the input. If the halting problem were decidable, $L$ would be decidable. Since $L$ is not even partially decidable, $L$ is not decidable, so the halting problem is not decidable, either.