Exercise [16.18]

My answer to this exercise must be "NO" (I could not establish the assertion by myself).

I therefore looked for the proof in Penrose's book "The Emperor's New Mind", where it is however expressed with another formalism (without Turing machines). The central idea seems to be a sentence like *"This sentence cannot be proven in the formal system **F**"*.
So here is my proposal how to express this with Turing machines:

Assume that **P** is a Turing Machine[1] that proves all calculable Π_1-sentences of the form

"$T_w(w)$ does not terminate",

i.e. the result of **P** is "1" if the sentence is true. If the Π_1-sentence is false or if **P** cannot handle it, **P** shall not terminate[2]. This can be formulated as follows:

$$P(w) = \begin{cases} 1 & \Rightarrow T_w(w) \text{ does not terminate} \\ \text{does not terminate} & \Rightarrow T_w(w) \text{ terminates or not} \end{cases}$$

As **P** is a Turing Machine, it must appear in the list of all Turing Machines, i.e. there must be some index *k* such that **P** = $\mathbf{T}_k$. The Π_1-sentence

"$T_k(k)$ does not terminate" (#)

is then my candidate for the desired G(**F**) sentence:

- First, this sentence (#) cannot be proven by **P**, i.e. we have **P***(k)* ≠ 1, because **P***(k)* = 1 would require that **P***(k)* terminates. As **P***(k)*= $\mathbf{T}_k(k)$, this would however be a contradiction to what was allegedly proven.
- Accordingly, <u>**P***(k)* does not terminate</u>, i.e. the Proof-Machine **P** yields no result for this Π_1-sentence. However, the fact that "**P***(k)* does not terminate" just shows the truth of the Π_1-sentence (#) because **P***(k)*= $\mathbf{T}_k(k)$.

Is there any flaw in these considerations? I am grateful for any comments!

---

[1] I think that it is not necessary to consider several of such Turing Machines to cover all calculable Π_1-sentences because these could be combined into one Machine.
[2] If **P** should in some circumstances be able to find out that $T_w(w)$ terminates, this can be ignored and **P** can be driven into an endless loop.

Juergen Beckmann