Refuting the Halting Problem Diagonalization Argument

Every machine that halts in a reject state is a halting computation. At least two proofs ignore this when constructing Sipser's Figure 4.5. Because these two proofs ignore this when they insert machine D in Sipser's Figure 4.5 they do so incorrectly.

When machine D is inserted in both Figure 4.4 and Figure 4.5 correctly then the contradiction goes away. Since Sipser implicitly assumes that every blank entry of Figure 4.4 is a ~halt entry Figure 4.7a makes this explicit.

	⟨M1⟩	⟨M2⟩	⟨M3⟩	⟨M4⟩ ...	⟨D⟩
M1	accept	~halt	~halt	accept	reject
M2	accept	accept	accept	accept	reject
M3	~halt	~halt	~halt	~halt	accept
M4	accept	accept	~halt	~halt	accept
...					
D	reject	reject	accept	accept	reject

Figure 4.7a (corrected figure 4.6, inserting D into figure 4.4)

	⟨M1⟩	⟨M2⟩	⟨M3⟩	⟨M4⟩ ...	⟨D⟩
M1	accept	reject	reject	accept	reject
M2	accept	accept	accept	accept	reject
M3	reject	reject	reject	reject	accept
M4	accept	accept	reject	reject	accept
...					
D	accept	accept	accept	accept	accept

Figure 4.7b (corrected figure 4.6, inserting D into figure 4.5)

The following portions of pages 166-167 are directly relevant to the rebuttal.
Sipser, Michael 1997. Introduction to the Theory of Computation. Boston: PWS Publishing Company (165-167)

Where is the diagonalization in the proof of Theorem 4.9? It becomes apparent when you examine tables of behavior for TMs H and D. In these tables we list all TMs down the rows, $M_1, M_2, \ldots$ and all their descriptions across the columns, $\langle M_1\rangle, \langle M_2\rangle, \ldots$ The entries tell whether the machine in a given row accepts the input in a given column. The entry is *accept* if the machine accepts the input but is blank if it rejects or loops on that input. We made up the entries in the following figure to illustrate the idea.

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\cdots$
M_1	*accept*		*accept*		
M_2	*accept*	*accept*	*accept*	*accept*	
M_3					$\cdots$
M_4	*accept*	*accept*			
$\vdots$		$\vdots$			

FIGURE 4.4
Entry i, j is *accept* if M_i accepts $\langle M_j\rangle$

In the following figure the entries are the results of running H on inputs corresponding to Figure 4.4. So if M_3 does not accept input $\langle M_2\rangle$, the entry for row M_3 and column $\langle M_2\rangle$ is *reject* because H rejects input $\langle M_3, \langle M_2\rangle\rangle$.

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\cdots$
M_1	*accept*	*reject*	*accept*	*reject*	
M_2	*accept*	*accept*	*accept*	*accept*	$\cdots$
M_3	*reject*	*reject*	*reject*	*reject*	
M_4	*accept*	*accept*	*reject*	*reject*	
$\vdots$		$\vdots$			

FIGURE 4.5
Entry i, j is the value of H on input $\langle M_i, \langle M_j\rangle\rangle$

In the following figure, we added D to Figure 4.5. By our assumption, H is a TM and so is D. Therefore it must occur on the list $M_1, M_2, \ldots$ of all TMs. Note that D computes the opposite of the diagonal entries. The contradiction occurs at the point of the question mark where the entry must be the opposite of itself.

	$\langle M_1\rangle$	$\langle M_2\rangle$	$\langle M_3\rangle$	$\langle M_4\rangle$	$\cdots$	$\langle D\rangle$	$\cdots$
M_1	<u>*accept*</u>	*reject*	*accept*	*reject*		*accept*	
M_2	*accept*	<u>*accept*</u>	*accept*	*accept*	$\cdots$	*accept*	$\cdots$
M_3	*reject*	*reject*	<u>*reject*</u>	*reject*		*reject*	
M_4	*accept*	*accept*	*reject*	<u>*reject*</u>		*accept*	
$\vdots$		$\vdots$			$\ddots$		
D	*reject*	*reject*	*accept*	*accept*		<u>?</u>	
$\vdots$		$\vdots$					$\ddots$

FIGURE 4.6
If D is in the figure, a contradiction occurs at "?"