Hierarchy Theorems

Idea: Similar to arbitrary complex functions. Just need to be careful on how much space/time is needed to construct the diagonalizing function.

Space hierarchy

Theorem: Suppose L is accepted by a $S(n) \geq \log n$ space bounded machine. Then L can be accepted by a $S(n)$ space bounded machine which halts on all inputs (i.e. it either accepts or rejects every string).

Proof: Suppose M is $S(n)$ space bounded and accepts L.
wlog assume that M uses only one work tape.
Alphabet size of M: r.
Number of states of M: s.

If M uses space l on input x and accepts x, then it accepts within
$(|x| + 2) \cdot s \cdot l \cdot r^l < c^{\max(l, \log|x|)}$ steps, for some constant c.
(since otherwise an ID must repeat and the machine diverges)
Machine M' simulating M is constructed as follows.
M' uses two working tapes. The first work tape is used for simulation of M, and the 2nd tape keeps number of steps used by M (in base c).

On input x,

- Initially Mark the length of the counter (2nd tape) as $\log|x|$.
- Simulate M, tracking the number of steps used by M.
- If M uses more space than in the counter, increase the length of the counter.
- If M accepts, then accept.
- If M does not accept or if counter overflows, then reject.

(Note that overflow of counter indicates that some ID must have been repeated since M has used more than $c^{\max(l, \log|x|)}$ time steps, where l is the space used by M on the input). QED

Let $M_0, M_1, \ldots$ denote some recursive ordering of all the TMs using 2-tapes and alphabet $0, 1, B$.

Theorem: Suppose $S_2(n)$ and $S_1(n)$ are both $\geq \log n$. Suppose that $S_2(n)$ is fully space constructible and

$$\lim_{n \to \infty} \frac{S_1(n)}{S_2(n)} = 0$$

Then there is a language in DSPACE$(S_2(n))-$DSPACE$(S_1(n))$.

Proof: We construct a machine M which is $S_2(n)$ space bounded.
M has fixed number of tapes (atleast 3).
Let L denote the language accepted by M.
L will not be in DSPACE$(S_1(n))$.

M rejects all inputs of the form 1^k.

M on input of the form 1^k0x works as follows:

- Mark out space $S_2(|1^k0x|)$ on the work tape. (Note that S_2 is fully space constructible)
- Simulate machine M_x on input 1^k0x. If this simulation of machine M_x attempts to use more space than $S_2(|1^k0x|)$, then M rejects the input.
- If M_x halts on input 1^k0x in the above simulation (without using more than $S_2(|1^k0x|)$ space) then M accepts iff M_x did not accept the input.

In the above simulation, one can assume that M_x uses only 2 tapes and uses alphabet $\{0, 1, B\}$.

Claim: Language accepted by M is in DSPACE$(S_2(n))$ and not in DSPACE$(S_1(n))$.

M is $S_2(n)$ space bounded, thus L is in DSPACE$(S_2(n))$.

Suppose by way of contradiction (Sbwoc) that M' is $S_1(n)$ space bounded and accepts L. Then, there exists another 2-tape machine M_x which accepts L using alphabet $0, 1, B$ and uses space $\leq dS_1(n)$, for some constant d. Without loss of generality assume that M_x halts on all inputs.

Note that the simulation of M_x by M above requires space $cS_1(n)$, for some constant c.
c may depend on x but does not depend on k;

Let k be large enough so that $c * S_1(|1^k 0x|) < S_2(|1^k 0x|)$. Then the simulation of M_x by M on input $1^k 0x$ must complete, and thus M accepts $1^k 0x$ iff M_x did not.

Thus M_x accepts a different language than the language L accepted by M. QED

Time Hierarchy

The proof for time hierarchy theorem is similar. The machine M as in the space hierarchy theorem, must have a constant number of tapes, whereas it must simulate an arbitrary machine M'.
This causes a slack factor in the time. (Recall that we lose a factor of $\log$ in speed when we simulate using two tapes).

Theorem: Suppose $T_2(n)$ is fully time constructible and $T_2(n), T_1(n) \geq (1+\epsilon)n$. Suppose that

$$\lim_{n \to \infty} \frac{T_1(n) * \log(T_1(n))}{T_2(n)} = 0$$

Then there exists a language in DTIME$(T_2(n))$ which is not in DTIME$(T_1(n))$. (Note that $T_1(n), T_2(n) \geq n$ by our assumption on time.)

Proof: We construct a machine M which is $O(T_2(n))$ time bounded. Let L be the language accepted by M. Clearly, L is in DTIME$(T_2(n))$ (by linear speed up theorem).
Consider a machine M with at least 5 tapes (it may be more depending on number of tapes needed for fully time constructibility of $T_2(n)$).
M on any input, of length n, first marks out time $T_2(n)$. In the construction it simultaneously counts the number of steps taken in the construction. If the number of steps taken reaches $T_2(n)$, then M stops and the input is rejected. (Note that this is why we need $T_2(n)$ to be fully time constructible).

M rejects all inputs of the form 1^k.

M on input of the form 1^k0x works as follows:

- M simulates M_x on input 1^k0x. If this simulation takes more than $T_2(|1^k0x|)$ time then M rejects the input.
- If M_x halts on input 1^k0x in the above simulation then M accepts iff M_x rejected the input.

In above construction, we consider only M_x which use 2 tapes and fixed alphabet say $\{0, 1, B\}$.

Let L be the language accepted by M.

M is $O(T_2(n))$ time bounded, thus L is in DTIME$(T_2(n))$ (by linear speed up theorem).

Suppose by way of contradiction (Sbwoc) that M' is $T_1(n)$ time bounded and accepts L.

Then, there exists another 2-tape machine M_x which accepts L using alphabet $0, 1, B$ and is $d * T_1(n) \log T_1(n)$-time bounded, for some constant d.

Note that the simulation of M by M_x above requires time $c * T_1(n) \log(T_1(n))$, for some constant c (which may depend on x but does not depend on k). To do the simulation, M copies x into a new tape, and then simulates M_x step by step on input $1^k 0x$ using the input and another tape.

Let k be large enough so that
$c * T_1(|1^k 0x|) * \log(T_1(|1^k 0x|) < T_2(|1^k 0x|)$.
Then the simulation of M_x by M on input $1^k 0x$ must complete, and thus M accepts $1^k 0x$ iff M_x rejects it.
Thus M_x accepts a different language than the language L accepted by M. QED

Relationship among complexity classes

Theorem: (a) DTIME($S(n)$) $\subseteq$ DSPACE($S(n)$).
(b) If L is in DSPACE($S(n)$), and $S(n) \geq \log n$, then there exsits a constant c, which depends on L, such that L is in DTIME$(c^{S(n)})$.
(c) If L is in NTIME($T(n)$) then there exists a constant c, which depends on L, such that L is in DTIME$(c^{T(n)})$.
Proof: (a) is trivial.
(b) We have seen the proof in the proof for space hierarchy theorem.
(c) was proved earlier.

Translation Lemma

Lemma: Suppose $S_1(n)$, $S_2(n)$ and $f(n)$ are fully space constructible with $S_1(n) \geq \log n$, $S_2(n) \geq n$ and $f(n) \geq n$. Then $NSPACE(S_1(n)) \subseteq NSPACE(S_2(n))$ implies $NSPACE(S_1(f(n))) \subseteq NSPACE(S_2(f(n)))$.

Proof: Suppose L is accepted by a $S_1(f(n))$ space bounded nondeterministic machine N.
We want to accept L using a $S_2(f(n))$ space bounded nondeterministic machine.
Let $L' = \{x\#^i : N$ accepts x in space $S_1(|x| + i)\}$
Clearly L' is in NSPACE$(S_1(n))$.
By hypothesis, L' is also in NSPACE$(S_2(n))$.
Suppose N' witnesses above.
We now construct a nondeterministic machine N'', which is $S_2(f(n))$ space bounded, and which accepts L.

N'' on input x:

Mark $S_2(f(n))$ space down on its tapes. (Note that this can be done within space $S_2(f(n))$, since $f(n)$ and $S_2(n)$ are fully space constructible, and $S_2(f(n)) \geq f(n) \geq n$). Guess i.

Simulate N' on inputs $x\#^i$.

For this simulation we need to keep track of the head of N' on the input tape. If the head is in the portion of x then it is done using the input head of N''. If head is in the portion $\#^i$, then we use a counter on a separate tape (note that length of this counter is at most $\log i$). Note that head on non-input tapes can be kept track of by just simulating those heads on the respective tapes.

If in the simulation above N' accepts then N'' accepts.
If N' uses more space than $S_2(f(n))$ or the counter value for i overflows the space $S_2(f(n))$, then N'' rejects.

Note that if x is in L, N accepts x in space $S_1(f(|x|)) = S_1(|x| + f(|x|) - |x|)$. Thus for $i = f(|x|) - |x|$, N' must accept $x\#^i$ within space $S_2(f(|x|))$. Moreover, since $i \leq f(|x|) \leq S_2(f(|x|))$, i can also fit in space $S_2(f(|x|)$. Thus N'' will accept x. QED

Note that we can relax $S_2(n) \geq n$ to $S_2(n) \geq \log n$, if we assume that $S_2(f(n))$ is fully space constructible, since then we can lay out $S_2(f(n))$ space without first laying down $f(n)$ space.
Essentially the argument also works for deterministic space, time and nondeterministic time also.