B Thermodynamics of computation

B.1 Von Neumann-Landaur Principle

¶1. **Entropy:** A quick introduction/review of the entropy concept. We will look at it in more detail soon (Sec. B.4).

¶2. **Information content:** The information content of a signal (message) measures our "surprise," i.e., how unlikely it is.
$I(s) = -\log_b \mathcal{P}\{s\}$, where $\mathcal{P}\{s\}$ is the probability of s.
We take logs so that the information content of independent signals is additive.
We can use any base, with corresponding units *bits*, *nats*, and *dits* (also, hartleys, bans) for $b = 2, e, 10$.

¶3. **1 bit:** Therefore, if a signal has a 50% probability, then it conveys one bit of information.

¶4. **Entropy of information:** The *entropy of a distribution* of signals is their average information content:

$$H(S) = \mathcal{E}\{I(s) \mid s \in S\} = \sum_{s \in S} \mathcal{P}\{s\} I(s) = -\sum_{s \in S} \mathcal{P}\{s\} \log \mathcal{P}\{s\}.$$

Or more briefly, $H = -\sum_k p_k \log p_k$.

¶5. **Shannon's entropy:** According to a well-known story, Shannon was trying to decide what to call this quantity and had considered both "information" and "uncertainty." Because it has the same mathematical form as statistical entropy in physics, von Neumann suggested he call it "entropy," because "nobody knows what entropy really is, so in a debate you will always have the advantage."[8]
(This is one version of the quote.)

¶6. **Uniform distribution:** If there are N signals that are all equally likely, then $H = \log N$.
Therefore, if we have eight equally likely possibilities, the entropy is $H = \lg 8 = 3$ bits.
A uniform distribution maximizes the entropy (and minimizes the ability to guess).

[8] `https://en.wikipedia.org/wiki/History_of_entropy` (accessed 2012-08-24).

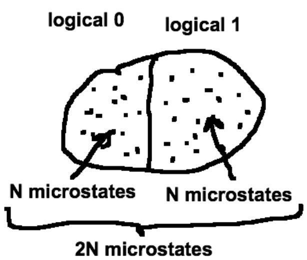

Figure II.3: Physical microstates representing logical states. Setting the bit decreases the entropy: $\Delta H = \lg N - \lg(2N) = -1$ bit. That is, we have one bit of information about its microstate.

¶7. **Macrostates and microstates:** Consider a macroscopic system composed of many microscopic parts (e.g., a fluid composed of many molecules). In general a very large number of *microstates* (or *microconfigurations*) — such as positions and momentums of molecules — will correspond to a given *macrostate* (or *macroconfiguration*) — such as a combination of pressure and termperature.
For example, with $m = 10^{20}$ particles we have $6m$ degrees of freedom, and a $6m$-dimensional phase space.

¶8. **Thermodynamic entropy:** Macroscopic thermodynamic entropy S is related to microscopic information entropy H by Boltzmann's constant, which expresses the entropy in thermodynamical units (energy over temperature).
$S = k_{\mathrm{B}} H$.
(There are technical details that I am skipping.)

¶9. **Microstates representing a bit:** Suppose we partition the microstates of a system into two macroscopically distinguishable macrostates, one representing 0 and the other representing 1.
Suppose N microconfigurations correspond to each macroconfiguration (Fig. II.3).
If we confine the system to one half of its microstate space, to represent a 0 or a 1, then the entropy (average uncertainty in identifying the microstate) will decrease by one bit.

We don't know the exact microstate, but at least we know which half of the statespace it is in.

¶10. **Overwriting a bit:** Consider the erasing or overwriting of a bit whose state was originally another known bit.

¶11. We are losing one bit of physical information. The physical information still exists, but we have lost track of it.

Suppose we have N physical microstates per logical macrostate (0 or 1). Therefore, there are N states in the bit we are copying and N in the bit to be overwritten. But there can be only N in the rewritten bit, so N must be dissipated into the environment. $\Delta S = k \ln(2N) - k \ln N = k \ln 2$ dissipated. (Fig. II.3)

¶12. The increase of entropy is $\Delta S = k \ln 2$, so the increase of energy in the heat reservoir is $\Delta S \times T_{\text{env}} = kT_{\text{env}} \ln 2 \approx 0.7kT_{\text{env}}$. (Fig. II.4)
$kT_{\text{env}} \ln 2 \approx 18$ meV $\approx 3 \times 10^{-9}$pJ.

¶13. **von Neumann – Landauer bound:** This is the *von Neumann – Landauer (VNL) bound*. VN suggested the idea in 1949, but it was published first by Rolf Landauer (IBM) in 1961.

¶14. "From a technological perspective, energy dissipation per logic operation in present-day silicon-based digital circuits is about a factor of 1,000 greater than the ultimate Landauer limit, but is predicted to quickly attain it within the next couple of decades." (Berut et al., 2012)
That is, current circuits are about 18 eV.

¶15. **Experimental confirmation:** In March 2012 the Landauer bound was experimentally confirmed (Berut et al., 2012).

B.2 Erasure

This lecture is based on Charles H. Bennett's "The Thermodynamics of Computation — a Review" (Bennett, 1982). Unattributed quotations are from this paper.

"Computers may be thought of as engines for transforming free energy into waste heat and mathematical work." (Bennett, 1982)

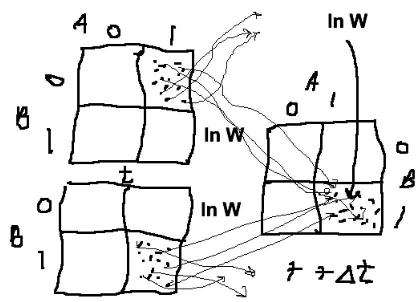

Figure II.4: Bit $A = 1$ is copied over bit B (two cases: $B = 0$ and $B = 1$). In each case there are $W = N^2$ micro states representing each prior state, so a total of $2W$ logically meaningful microstates. However, at time $t + \Delta t$ the two-bit system must be on one of W posterior microstates. Therefore W of the trajectories have exited the $A = B = 1$ region of phase space, and so they are no longer logically meaningful. The entropy of the environment must increase by $\Delta S = k \ln(2W) - k \ln W = k \ln 2$. We lose track of this information because it passes into uncontrolled degrees of freedom.

¶1. As it turns out, it is not measurement or copying that necessarily dissipates energy, it is the erasure of previous information to restore it to a standard state so that it can be used again.
Thus, in a TM writing a symbol on previously used tape requires two steps: erasing the previous contents and then writing the new contents. It is the former process that increases entropy and dissipates energy.
Cf. the mechanical TM we saw in CS312; it erased the old symbol before it wrote the new one.
Cf. also old computers on which the load instruction was called "clear and add."

¶2. A bit can be copied reversibly, with arbitrarily small dissipation, if it is initially in a standard state (Fig. II.5).
The *reference bit* (fixed at 0, the same as the moveable bit's initial state) allows the initial state to be restored, thus ensuring reversibility.

¶3. Susceptible to degradation by thermal fluctuation and tunneling. However, the error probability η and dissipation ϵ/kT can be kept arbitrarily small and much less than unity.

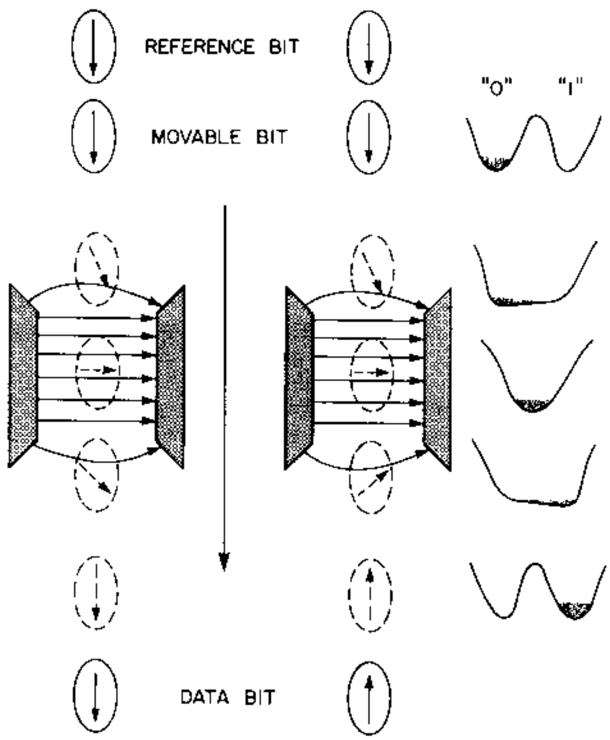

Fig. 14. Reversible copying using a one-domain ferromagnet. The movable bit, initially zero, is mapped into the same state as the data bit (zero in left column; one in center column). Right column shows how the probability density of the movable bit's magnetization, initially concentrated in the "0" minimum, is deformed continuously until it occupies the "1" minimum, in agreement with a "1" data bit.

Figure II.5: [from Bennett (1982)]

ϵ here is the driving force.

¶4. Whether erasure must dissipate energy depends on the prior state (Fig. II.6).

(B) The initial state is genuinely unknown. This is reversible (steps 6 to 1). $kT \ln 2$ work was done to compress it into the left potential well (decreasing the entropy). Reversing the operation increases the entropy by $kT \ln 2$.
That is, the copying is reversible.

(C) The initial state is known (perhaps as a result of a prior computation or measurement). This initial state is lost (forgotten), converting $kT \ln 2$ of work into heat with no compensating decrease of entropy. The irreversible entropy increase happens because of the expansion at step 2. This is where we're forgetting the initial state (vs. case B, where there was nothing known to be forgotten).

¶5. In research reported in March 2012 (Berut et al., 2012) a setup very much like this was used to confirm experimentally the Landauer Principle and that it is the erasure that dissipates energy.

"incomplete erasure leads to less dissipated heat. For a success rate of r, the Landauer bound can be generalized to"

$$\langle Q \rangle^r_{\text{Landauer}} = kT[\ln 2 + r \ln r + (1-r)\ln(1-r)] = kT[\ln 2 - H(r, 1-r)].$$

"Thus, no heat needs to be produced for $r = 0.5$" (Berut et al., 2012).

¶6. We have seen that erasing a random register increases entropy in the environment in order to decrease it in the register: $NkT \ln 2$ for an N-bit register.

¶7. Such an initialized register can be used as "fuel" as it thermally randomizes itself to do $NkT \ln 2$ useful work.
Keep in mind that these 0s (or whatever) are physical states (charges, compressed gas, magnetic spins, etc.).

¶8. Any computable sequence of N bits (e.g., the first N bits of π) can be used as fuel by a slightly more complicated apparatus. Start with a reversible computation from N 0s to the bit string.
Reverse this computation, which will transform the bit string into all

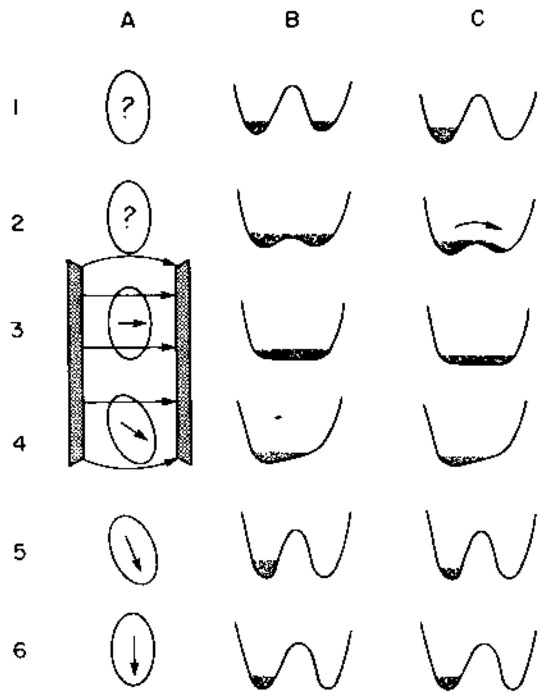

Fig. 16. Erasure of a bistable one-domain ferromagnet. Column A: the bistable element, which may be magnetized either up or down (1), is moved gradually (2) into a transverse field that abolishes its bistability symmetrically (3). It is then rotated slightly counterclockwise (4), to bias the soft mode downward, removed from the field (5), and rotated back again, leaving it in the down or zero state (6). Column B: Evolution of the probability density when the manipulation just described is used to erase a random, unknown bit. Column C: Behavior of the probability density when the manipulation is applied to a known bit (here zero). An irreversible entropy increase of $k \ln 2$ occurs at stage 2, when the probability density leaks out of the initially occupied minimum.

Figure II.6: [from Bennett (1982)]

0s, which can be used as fuel. This "uncomputer" puts it in usable form.

¶9. Suppose we have a random bit string, initialized, say, by coin tossing. Because it has a specific value, we can in principle get $NkT \ln 2$ useful work out of it, but we don't know the mechanism (the "uncomputer") to do it. The apparatus would be specific to each particular string.

B.3 Algorithmic entropy

¶1. *Algorithmic information theory* was developed by Ray Solomonoff c. 1960, Andrey Kolmogorov in 1965, and Gregory Chaitin, around 1966.

¶2. **Algorithmic entropy:** The algorithmic entropy $H(x)$ of a microstate x is the number of bits required to describe x as the output of a universal computer, roughly, the size of the program required to compute it.
Specifically the smallest "self-delimiting" program (i.e., its end can be determined from the bit string itself).

¶3. Differences in machine models lead to differences of $\mathcal{O}(1)$, and so $H(x)$ is well-defined up to an additive constant (like thermodynamical entropy).

¶4. Note that H is not computable (Rice's Theorem).

¶5. **Algorithmically random:** A string is *algorithmically random* if it cannot be described by a program very much shorter than itself.

¶6. For any N, most N-bit strings are algorithmically random.
(For example, "there are only enough $N - 10$ bit programs to describe at most 1/1024 of all the N-bit strings.")

¶7. Deterministic transformations cannot increase algorithmic entropy very much.
Roughly, $H[f(x)] \approx H(x) + |f|$, where $|f|$ is the size of program f.
Reversible transformations also leave H unchanged.

¶8. A transformation must be probabilistic to be able to increase H significantly.

¶9. **Statistical entropy:** Statistical entropy in units of bits is defined:

$$S_2(\mathbf{p}) = -\sum_x p(x) \lg p(x).$$

¶10. Statistical entropy (a function of the macrostate) is related to algorithmic entropy (an average over algorithmic entropies of microstates) as follow:

$$S_2(\mathbf{p}) < \sum_x p(x) H(x) \leq S_2(\mathbf{p}) + H(\mathbf{p}) + \mathcal{O}(1).$$

¶11. A macrostate $\mathbf{p}$ is *concisely describable* if, for example, "it is determined by equations of motion and boundary conditions describable in a small number of bits."
In this case S_2 and H are closely related as given above.

¶12. For macroscopic systems, typically $S_2(\mathbf{p}) = \mathcal{O}(10^{23})$ while $H(\mathbf{p}) = \mathcal{O}(10^3)$.

¶13. If the physical system increases its H by N bits, which it can do only by acting probabilistically, it can "convert about $NkT \ln 2$ of waste heat into useful work."

¶14. "[T]he conversion of about $NkT \ln 2$ of work into heat in the surroundings is necessary to decrease a system's algorithmic entropy by N bits."

B.4 Mechanical and thermal modes

These lecture is based primarily on Edward Fredkin and Tommaso Toffoli's "Conservative logic" (Fredkin & Toffoli, 1982).

¶1. Systems can be classified by their size and completeness of specification:

specification:	complete	incomplete	
size:	~ 1	~ 100	$\sim 10^{23}$
laws:	dynamical	statistical	thermodynamical
reversible:	yes	no	no

¶2. **Dynamical system:** Some systems with a relatively small number of particles or degrees of freedom can be completely specified.
E.g., 6 DoF for each particle (x, y, z, p_x, p_y, p_z).

¶3. That is, we can prepare an *individual* system in an initial state and expect that it will behave according to the dynamical laws that describe it.

¶4. Think of billiard balls or pucks on a frictionless surface, or electrons moving through an electric or magnetic field.

¶5. So far as we know, the laws of physics at this level (classical or quantum) are reversible.

¶6. **Statistical system:** If there are a large number of particles with many degrees of freedom (several orders of magnitude), then it is impractical to specify the system completely.

¶7. Small errors in the initial state will have a larger effect, due to complex interaction of the particles.

¶8. Therefore we must resort to *statistical laws*.

¶9. They don't tell us how an individual system will behave (there are too many sources of variability), but they tell us how *ensembles* of similar systems (or preparations) behave.

¶10. We can talk about the average behavior of such systems, but we also have to consider the variance, because unlikely outcomes are not impossible.
For example, tossing 10 coins has a probability of $1/1024$ of turning up all heads.

¶11. Statistical laws are in general irreversible (because there are many ways to get to the same state).

¶12. **Thermodynamical system:** Macroscopic systems have a very large number of particles ($\sim 10^{23}$) and a correspondingly large number of DoF. "Avogadro scale" numbers.

¶13. Obviously such systems cannot be completely specified (we cannot describe the initial state and trajectory of every atom).

¶14. We can derive statistical laws, but in these cases most macrostates become so improbable that they are virtually impossible: Example:

the cream unmixing from your coffee.
The central limit theorem shows that the variance decreases with n.

In the thermodynamic limit, the *likely* is *inevitable*, and the *unlikely* is *impossible*.

¶15. In these cases, *thermodynamical laws* describe the virtually deterministic (but *irreversible*) dynamics of the system.

¶16. **Mechanical vs. thermal modes:** Sometimes in a macroscopic system we can separate a small number of *mechanical modes* (DoF) from the *thermal modes*.

"Mechanical" includes "electric, magnetic, chemical, etc. degrees of freedom."

¶17. The mechanical modes are strongly coupled to each other but weakly coupled to the thermal modes.
(e.g., bullet, billiard ball)

¶18. Thus the mechanical modes can be treated exactly or approximately independently of the thermal modes.

¶19. **Conservative mechanisms:** In the ideal case the mechanical modes are completely decoupled from the thermal modes, and so the mechanical modes can be treated as an isolated (and reversible) dynamical system.

¶20. The energy of the mechanical modes (once initialized) is independent of the energy ($\sim kT$) of the thermal modes.

¶21. The mechanical modes are *conservative*; they don't dissipate any energy.

¶22. This is the approach of reversible computing.

¶23. **Damped mechanisms:** Suppose we want irreversible mechanical modes, e.g., for implementing irreversible logic.

¶24. The physics is reversible, but the information lost by the mechanical modes cannot simply disappear; it must be transferred to the thermal modes. This is *damping*.

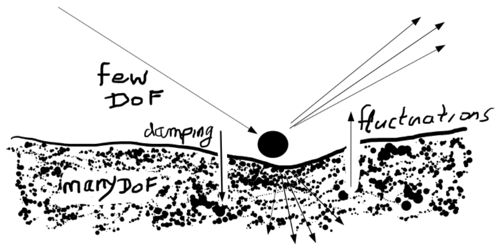

Figure II.7: Complementary relation of damping and fluctuations.

¶25. But the transfer is reversible, so *noise* will flow from the thermal modes back to the mechanical modes, making the system nondeterministic.

¶26. "If we know where the *damping* comes from, it turns out that that is also the source of the *fluctuations*" [Feynman, 1963].

Think of a bullet ricocheting off a flexible wall filled with sand. It dissipates energy into the sand and also acquires noise in its trajectory (see Fig. II.7).

¶27. To avoid nondeterminacy, the information may be encoded redundantly so that the noise can be filtered out.

I.e., signal is encoded in multiple mechanical modes, on which we take a majority vote or an average.

¶28. The signal can be encoded with energy much greater than any one of the thermal modes, $E \gg kT$, to bias the energy flow from mechanical to thermal preferring dissipation to noise).

¶29. **Signal regeneration:** Free energy must refresh the mechanical modes and heat must be flushed from the thermal modes.

¶30. "[I]mperfect knowledge of the dynamical laws leads to uncertainties in the behavior of a system comparable to those arising from imperfect knowledge of its initial conditions... Thus, the same regenerative processes which help overcome thermal noise also permit reliable operation in spite of substantial fabrication tolerances."

¶31. Damped mechanisms have proved to be very successful, but they are inherently inefficient.

¶32. "In a damped circuit, the rate of heat generation is proportional to the number of computing elements, and thus approximately to the useful volume; on the other hand, the rate of heat removal is only proportional to the free *surface* of the circuit. As a consequence, computing circuits using damped mechanisms can grow arbitrarily large in two dimensions only, thus precluding the much tighter packing that would be possible in three dimensions."

B.5 Brownian computers

¶1. Rather than trying to avoid randomization of kinetic energy (transfer from mechanical modes to thermal modes), perhaps it can be exploited. An example of *respecting the medium* in *embodied computation*.

¶2. **Brownian computer:** Makes logical transitions as a result of thermal agitation.
It is about as likely to go backward as forward.
It may be biased in the forward direction by a very weak external driving force.

¶3. **DNA:** DNA transcription is an example. It runs at about 30 nucleotides per second and dissipates about $20kT$ per nucleotide, making less than one mistake per 10^5 nucleotides.

¶4. **Chemical Turing machine:** Tape is a large macromolecule analogous to RNA.
An added group encodes the state and head location.
For each tradition rule there is a hypothetical enzyme that catalyzes the state transition.

¶5. Drift velocity is linear in dissipation per step.

¶6. We will look at molecular computation in much more detail later in the class.

¶7. **Clockwork Brownian TM:** He considers a "clockwork Brownian TM" comparable to the billiard ball computers. It is a little more realistic, since it does not have to be machined perfectly and tolerates environmental noise.

¶8. Drift velocity is linear in dissipation per step.

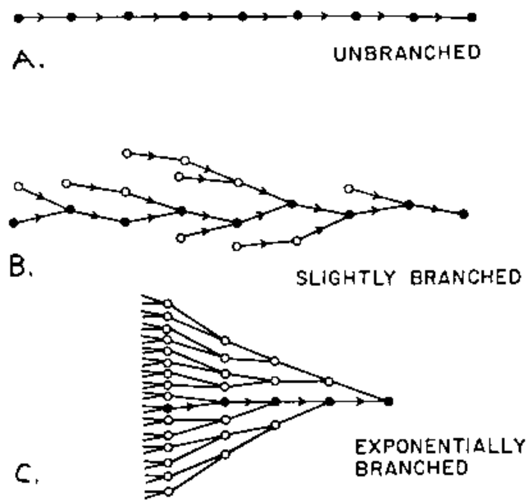

Figure II.8: Different degrees of logical reversibility. [from (Bennett, 1982)]

B.5.a Reversibility in computing

¶1. Bennett (1973) defined a reversible TM.

¶2. As in ballistic computing, Brownian computing needs logical reversibility.

¶3. A small degree of irreversibility can be tolerated (see Fig. II.8).
(A) Strictly reversible computation. Any driving force will ensure forward movement.
(B) Modestly irreversible computation. There are more backward detours, but can still be biased forward.
(C) Exponentially backward branching trees. May spend much more of its time going backward than going forward, especially since one backward step is likely to lead to more backward steps.

¶4. For forward computation on such a tree the dissipation per step must exceed kT times the log of the mean number of immediate predecessors to each state.

¶5. These undesired states may outnumber desired states by factors of 2^{100}, requiring driving forces on the order of $100kT$.

Why 2^{100}? Think of the number of possible predecessors to a state that does something like `x := 0`. Or the number of ways of getting to the next statement after a loop.