Entropic Lower Bounds for Sorting

"I play K. 330 slowly and evenly, as if X-raying a mollusc. — Glenn Gould"
— Sam Tenka

The $n \lg n$ lower bound on comparison sorts allures me with its beauty. It goes like this: to identify a total order on a size-n set is to distinguish between $n!$ possibilities. Each interesting query to $\leq$ has two possible outcomes and hence yields at most one bit of information. A sorting algorithm based only on comparisons thus requires at least $\lg(n!) \sim n \lg n$ many queries; merge sort realizes this bound.[1]

[1] We consider complexities up to $\sim$, where $f \sim g$ means $\lim f/g = 1$. We thus attend to constants but not to lower order terms. Here, $\lg = \log_2$.

This note discusses some lower bounds for related problems.

Counting Queries

Decision Trees

The concept of *decision trees* abstracts our notion of algorithm to a level convenient for information-theoretic reasoning. An algorithm's complexity is then its height as a tree. We'll rely only on an intuitive understanding of such algorithms and their complexity; still, to fix terminology we give a formal definition here. We consider the problem of implementing a map $\textsc{task} : \mathcal{X} \to \mathcal{Z}$ in terms of queries — labeled by $q \in \mathcal{Q}$ — whose meanings are defined by $\textsc{ask} : \mathcal{Q} \to \mathcal{X} \to \mathcal{A}$. (We'll consider only finite $\mathcal{X}, \mathcal{Q}, \mathcal{A}, \mathcal{Z}$s and surjective $\textsc{task}$s.) For a fixed problem $(\textsc{task}, \textsc{ask})$, an algorithm of complexity c is a pair $(\textsc{next} : \mathcal{A}^\star \to \mathcal{Q}, \textsc{read} : \mathcal{A}^{\times c} \to \mathcal{Z})$ that is correct on all inputs:

$$(\textsc{task}(x), x) = ((\textsc{read} \times \mathrm{id}_{\mathcal{X}}) \circ \textsc{body}^{\circ c})\ ([], x)$$
$$\textsc{body}\ (\ell, x) = (\textsc{push}\ \ell\ (\textsc{ask}\ (\textsc{next}\ \ell)\ x), x)$$

Here, $\mathcal{A}^\star = \bigsqcup_n \mathcal{A}^{\times n}$ is the type of $\mathcal{A}$-valued lists; its constructors are $[] : \mathcal{A}^\star$ (the empty list) and $\textsc{push} : \mathcal{A}^\star \to \mathcal{A} \to \mathcal{A}^\star$.

We're interested in lower bounds on c for various problems. For example, we may model comparison sorting as the problem of computing the identity function $\textsc{task} = \mathrm{id}_{\mathcal{X}}$ on the set $\mathcal{X}$ of total orders on a size-n set $\mathcal{S}$ by querying $\textsc{ask} : (\mathcal{S}^2 \setminus$

diagonal) $\to \mathcal{X} \to$ {less, more}. As another example, the field of *communication complexity* studies problems where $\textsc{task} : \text{Alice} \times \text{Bob} \to 2$ is a joint predicate and $\textsc{ask} : (2^{\text{Alice}} \sqcup 2^{\text{Bob}}) \to \mathcal{X} \to 2$ is the canonical evaluation map.

A Basic Bound

Recall the $n \lg n$ argument we started with. We isolate its essence as follows. A probability distribution on $\mathcal{X}$ induces (via $\textsc{task}$) a distribution on $\mathcal{Z}$ and (for any fixed algorithm of complexity c) on $\mathcal{Q}^c, \mathcal{A}^c$. Let's abuse notation by writing T for the evident random variable of type T; for instance, $H(\mathcal{Z})$ is the Shannon entropy of the random variable $z = \textsc{task}(x)$. When we choose $\mathcal{X}$'s distribution so that $\mathcal{Z}$'s distribution is uniform, the data processing inequality immediately gives the following complexity bound:

Lemma (Counting Bound). $c \lg |\mathcal{A}| \geq H(\mathcal{A}^c) \geq H(\mathcal{Z}) = \lg |\mathcal{Z}|$.

As is usual, each problem we consider is actually part of a natural-number indexed family of problems and thus induces a sequence $(c_n : n \in \mathbb{N})$ of complexities. We study these sequences up to the preorder $f \lesssim g$ defined by $\lim f/g \leq 1$. With this in mind, we plug in $|\mathcal{X}| = |\mathcal{Z}| = n!$ and $|\mathcal{A}| = 2$ to recover the $c \gtrsim n \lg n$ bound for comparison-based identification of total orders.

The counting bound gives some interesting results for related identification (i.e., $\textsc{task} = \mathrm{id}_{\mathcal{X}}$ problems:

Puzzle (Merge). Let's merge two sorted lists of sizes m, n with $1 \ll k \ll n$. More precisely, we fix $\mathcal{S} = [k] \sqcup [n]$, set $\mathcal{X}$ to the set of total orders on $\mathcal{S}$ that restrict to the standard orders on $[k], [n]$, and let $\textsc{ask}$ compare distinct pairs in $\mathcal{S}^2$. Show that $c \gtrsim n \lg(n/k)$ and that this bound is achieved. For example, if $k \sim n/\lg n$, then $c \sim n \lg \lg n / \lg n$ is optimal — strictly better than the "zip" or "search" strategies! This win-win prefigures *fractional cascading*.

Puzzle (Ballots). Let's sort potentially tied elements. So $\mathcal{X}$ contains the ballots (a.k.a.: total preorders) on $\mathcal{S}$ that have k equivalence classes. Here, $\textsc{ask}$ maps to {less, tied, more}. The counting bound says $c \gtrsim n \lg k / \lg 3$. Improve this bound to $c \gtrsim n \lg k$ and show that the latter is tight. Hint: a routine transformation gives for any complexity-c' algorithm a complexity-$c \leq c'$ algorithm that for any x makes fewer than n queries answered by "tied".

Examples in Communication

Communicating an Equality.
Communicating a Comparison.

Convex Cohorts

Distinct Elements

Strictly Orderable.

Counterfeit Coins. We have n coins, some of which may be counterfeit. Not all of the coins are counterfeit. The counterfeit coins weigh $1+\epsilon$ while the ordinary coins weigh 1 for $\epsilon < 1/n$. We have a balance that tells us for any two subsets of the n coins which, if any, is heavier. In how few comparisons may determine whether or not there are any counterfeit coins?

Now $\mathcal{X} = 2^n \setminus [n]$ and $\textsc{task}\ x$ indicates whether or not $\mathcal{X}$ is empty. Now $\mathcal{Q} = \sum_k \binom{n}{k}^2$ and $\textsc{ask}\ (a,b)\ x$ indicates whether or not $|x \cap a| = |x \cap b|$. What sorts of lower bounds on c can we come up with?

Range

Minimum.

Range.

Constraints of no Consequence

Length

Bits

problem	counting	lower	upper	direct
Merge			$n \lg\lg n / \lg n$	
Ballots			$n \lg n / 2$	
Strictly			$n \lg n$	
Counterfeit			$2 \lg n / \lg\lg n$	
Minimum			n	
Range			$3n/2$	
Length			$(\lg n)^2/2$	
Bits			$O(n \lg\lg n)$	