6.4 Physical Complexity

The obvious approach to assessing the complexity of an algorithm is counting the primitive operations that a customary processor executes, or, more exactly, counting the clock cycles. This would lead to concrete results like: "Computing ... costs at least (say) 10^{80} of the following steps: ...". For example we could count elementary arithmetical operations (additions, multiplications, ...), taking into account the word size of the processor (e. g. 32 bits) and the number of clock cycles for the considered operations. [Note that this number might not be uniquely defined on a modern CPU with pipeline architecture.]

For many concrete algorithms statements of this kind are possible, and often lead to interesting mathematical problems as abundantly demonstrated by D. KNUTH in his books.

Unfortunately no flavour of complexity theory yields results on the *minimum* number of steps that *each* algorithm for solving a certain problem must execute, except for extremely simple problems like evaluating a polynomial for a certain argument. If we knew results of this kind, we could mathematically prove the security of cryptographic procedures without recurring to unproven conjectures or heuristic arguments.

This kind of reasoning could take into account physical bounds that limit the resources computers in this universe can dispose of. A known estimate of this kind was proposed by Louis K. SCHEFFER in `sci.crypt`:

- Our universe contains at most 10^{90} elementary particles. This is certainly an upper bound for the number of available CPUs.
- Passing an elementary particle with the speed of light takes at least 10^{-35} seconds. This is certainly a lower bound for the time required by a single operation.
- Our universe has a life span of at most 10^{18} seconds ($\approx 30 \times 10^9$ years). This is certainly an upper bound for the available time.

Multiplying these bounds together we conclude that at most $10^{143} \approx 2^{475}$ operations can be executed in our universe. In particular 500-bit keys are secure from exhaustion ...

> ... until such time as computers are built from something other than matter, and occupy something other than space. (Paul CISZEK)

Note that this security bound holds for the one algorithm "exhaustion". It has no relevance for the security of even a single cryptographic procedure! (As long as there is no proof that no attack is faster than exhaustion.)

Needless to say that a realistic upper bound is smaller by several orders of magnitude.

For comparision we list some cryptologically relevant quantities:

seconds/year	3×10^7
CPU cycles/year (1 GHz CPU)	3.2×10^{16}
age of our universe (years)	10^{10}
CPU cycles since then (1 GHz)	3.2×10^{26}
atoms of the earth	10^{51}
electrons in our universe	8.37×10^{77}
ASCII strings of length 8 (95^8)	6.6×10^{15}
binary strings of length 56 (2^{56})	7.2×10^{16}
binary strings of length 80	1.2×10^{24}
binary strings of length 128	3.4×10^{38}
binary strings of length 256	1.2×10^{77}
primes with 75 decimal places (about 250 bits)	5.2×10^{72}