Biological and Computer Designs Minimize Energy and Time. NOTE change title and abstract for Roy Soc

George Bezerra,[1,2*] James Brown,[3] Melanie Moses,[2,3] Stephanie Forrest[2]

[1]Computer Science and Artificial Intelligence Laboratory
Massachusetts Institute of Technology, Cambridge, MA, USA.
[2]Department of Computer Science
University of New Mexico, Albuquerque, NM, USA.
[3]Department of Biology
The University of New Mexico, Albuquerque, NM, USA.

[*]To whom correspondence should be addressed
E-mail: gbezerra@csail.mit.edu
Address: The Sata Center Building 32-G778
32 Vassar Street, Cambridge, Massachusetts 02139 USA
Phone: 617-324-8434

Keywords: network scaling, power consumption, metabolic rate, complex systems, energy-time product, organism, computer chip

Classification: Major: ; Minor:

Abstract

Metabolic rate in animals and power consumption in computers are analogous quantities that scale similarly with size. We analyze vascular systems of vertebrates and on-chip networks of microprocessors, where natural selection and human engineering respectively have produced optimized networks. Both network designs simultaneously reduce energy costs and increase flow rates. Using a simple network model, our analysis explains empirically observed trends in the scaling of metabolic rate in organisms and power consumption in chips across several orders of magnitude in size. This result suggests that a single principle governs the designs of complex systems that process energy, materials, and information.

1 Introduction

Both organisms and computers have evolved from relatively simple beginnings into complex systems that vary by orders of magnitude in size and number of components. Evolution, by natural selection in organisms and by human engineering in computers, required critical features of architecture and function to be scaled up as size and complexity increased. In Biology, Kleiber's law describes the empirical relation between metabolic rate and many other traits, such as lifespan, heart rate, and number of offspring, with body size [**?**]. Similarly, computer architecture has Moore's law to describe scaling of transistor density and performance [**?**], Koomey's law for the energy cost per computation [**?**], and Rent's rule for the external communication per logic block [**?**].

We posit that these empirical patterns originate from a common principle: Networks that deliver resources are optimized to reduce energy dissipation and increase flow rates, expressed here as minimizing the energy-time product. That is, both living systems and computer chips are designed to maximize the rate at which resources are delivered to terminal nodes of a network and to minimize the energy cost in the network. In biology, the vascular network of vertebrate animals supplies oxygen and nutrients to every cell, fueling metabolism for maintenance, growth and reproduction. Since energy is a limited resource, we assume that organisms are selected to minimize the energy dissipated in the network [**?**]. Similarly, computation in microprocessors relies on a network of microscopic wires that transmits bits of information between transistors on a chip. In order to maximize computation speed and minimize power consumption, this network is designed to deliver the maximum information flow at the lowest possible energy cost.

Here, we model vertebrates as composed of regions of tissue that receive oxygen carried by blood via a hierarchical vascular network of pipes, and we model microprocessors as composed

of transistors that perform computation, exchanging information over a modular network of wires. As each system scales up in size, we consider: 1) the rate at which resources are delivered by the network and processed in the nodes; and 2) the energy dissipated during these processes. Despite the obvious differences between organisms and chips, we present a general model and derive energy and time scaling relations from physical principles applicable to each system. Using these relations, we express the optimal network design as a tradeoff between energy cost and processing speed.

Earlier models in biology have assumed either energy minimization, e.g., [**?**] or optimal resource delivery rate [**?**], but they have not formalized the tradeoffs between them, as we do here. This formalization helps explain how nature and engineering are able to produce designs that approach pareto-optimal along the energy-time tradeoff. Thus, in biology evolution has produced mammals ranging in size from mice to elephants, rather than converging on a single optimal size, and in computer architecture engineers have designed processors ranging from X to Y, each of which fills a specific computational niche.

In the rest of the paper, we first present the unified model of network scaling (Section 2) and the basic assumptions underlying the model (Section **??**). We then use the model to derive a series of predictions about how time and energy scale with system size first for organisms (Section 3.1) and then for computers (Section 3.2). We then discuss new insights into previously analyzed scaling relationships in biology that we gain from the time-energy minimization framework, and we test scaling predictions with empirical power and performance data on computer chips. Finally, in Section 4 we discuss the implications of these results for evolutionary transitions in nature and engineering.

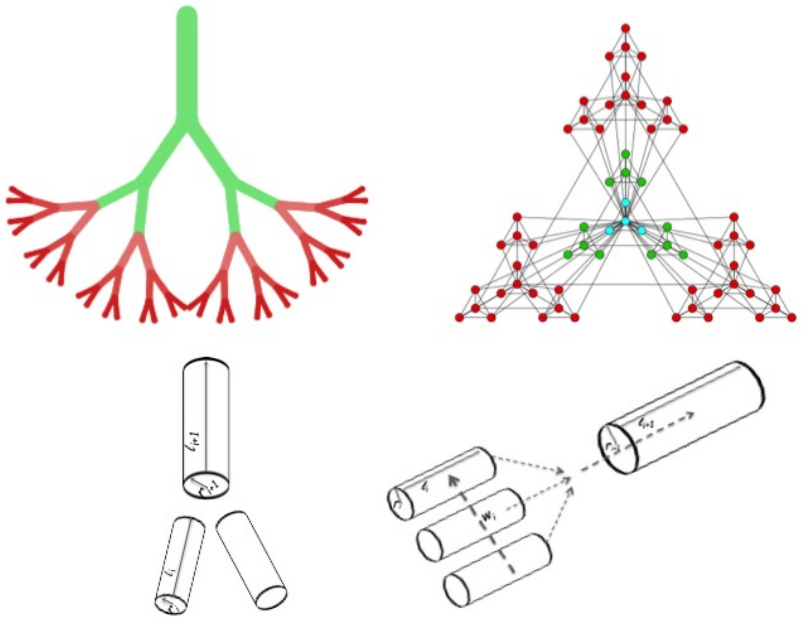

Figure 1: Panel a: idealized fractal branching model of the cardiovascular network. Explain Dl, Dr, lambda H N and slowing by referring to the picture. Panel c illustrates the communication dimension, D_w, which measures how the ratio of internal (intra-module) communication per node to external (inter-module) communication per node scales with the level of the module in the hierarchy. Keep b and d?

2 A Unified Model of Network Scaling

Vascular systems are hierarchical branching networks where blood vessels (pipes) become thicker and longer as hierarchy increases from the capillaries to the aorta. Similarly, microprocessor chips are organized hierarchically into a nested structure of modules and submodules, where wires become longer and thicker as the hierarchical level of a module increases. See Figure 1. These wires are organized into metal layers, where short, thin wires are routed on the lowest layers, and long, thick wires are placed on the top layers. We model the scaling of length (l) and thickness (r) of both pipes and wires as

$$l_i = l_0 \lambda^{i/D_l} \tag{1}$$

and

$$r_i = r_0 \lambda^{i/D_r}, \tag{2}$$

where i is the hierarchical level of a branch or module, λ is the branching factor, and D_l and D_r are the length and thickness dimensions. This model is akin to the hierarchical pipe model of vascular systems proposed in [**?**], where $1/D_l$ corresponds to the scaling exponent for γ and D_r corresponds to the scaling exponent forβ, but both are negative because this paper labels the lowest level of the hierarchy (the capillaries) level 0, while [**?**] labelled the highest level (the aorta) to be level 0. In vascular networks, r represents the radius of cylindrical pipes, and in computer interconnects, r represents the width of wires with aspect ratio 1. D_r describes the relative radius of pipes between successive hierarchical levels.

The length parameter D_l is set by the spatial dimension occupied by the nodes of the network [**?**]. For organisms, $D_l = 3$ [**?**, **?**], since capillaries distribute blood to cells in three dimensions. For chips, $D_l = 2$, since transistors are placed on a single two-dimensional layer [**?**, **?**]. Thus, D_l describes both the relative length of pipe between successive hierarchical levels and the external dimension of the system. The smallest edges occur at $i = 0$, and have length and thickness, l_0 and r_0.

Digital circuits scale in a third way in addition to length and radius, which has no direct analog in organisms. In vascular networks, each pipe branches at each hierarchical level (generally bifurcating so that $\lambda = 2$ CITE SavageCompBio2015.) Chips have multiple wires per module at each hierarchical level, and their number increases with the hierarchical level. This difference arises from the fact that digital circuits are decentralized networks connecting multiple sources and destinations, while vascular networks are centralized, with blood and oxygen flowing from a single heart. To account for this difference, we introduce a new equation, in which the communication (or number of wires) per module increases with the hierarchical level

as

$$w_i = w_0 \lambda^{i/D_w}, \tag{3}$$

where D_w is the communication dimension and w_0 is the average number of wires per node. This hierarchical scaling of communication is a well-known pattern in circuit design called Rent's rule [**?**], where $p = 1/D_w$ is the Rent's exponent.* This pattern is not unique to circuits and has been shown to occur in many biological networks [**?**, **?**]. Vascular systems correspond to a special case where $w_i = 1$ for all i.

2.1 Assumptions of the Unified Model

Before deriving scaling predictions from the model, we make explicit its assumptions and how they relate to earlier models, both in computation and biology:

1. **Time and energy are equally important constraints:** System designs seek to deliver the maximum amount of resource per unit time for the minimum amount of energy overhead. In computer architecture this relationship is expressed as the *energy-delay product*, which formalizes the insight that a chip that is ten times faster or ten times more energy efficient is ten times better.

2. **Steady state**: Resource supply matches processing demand [**?**]. That is, the network supplies resources continually to the terminal nodes and is always filled to capacity. This avoids network delays and the need to store resources in the system. Specifically,

   (a) System designs balance network delivery rates with node processing speeds, so that resources are delivered at exactly the same rate that they are processed.

*Rent's rule is typically expressed as $C(n) = kn^p$, where C_n is the external communication of a module, n is the size of the module (number of nodes), k is the average external communication of a module with size 1, and p is the Rent's exponent. For a hierarchy with branching factor of λ, the size of a module is given as $n = \lambda^i$, where i is the hierarchical level. Therefore, we can rewrite Rents rule as $c_i = c_0 \times \lambda^{ip}$, where $c_0 = w_0$ and $p = 1/D_w$.

(b) Pipelining: A concept from computer architecture in which resources, e.g., computer instructions, leave the source at the same rate that they are delivered to the terminal nodes, and the network is always full. In the hierarchical networks considered here, the source is the highest branch (root), and the principle holds within every level of the hierarchy. Consequently, resources flow through the network continually at maximum speed, and they do not accumulate at source, sink, or intermediate locations.

3. **Terminal units and service volumes:** In contrast to Ref. [**?**] we do not assume that terminal units have fixed size. In chips it is well known that transistor size has shrunk over many orders of magnitude. Previous scaling models of biology posit that the service volume (region served by terminal units of the network) increases with system size [**?**, **?**]. Folowing Ref. [**?**], we assume that capillaries have fixed radius but that their length is proportional to the radius of the service volume. Similarly, the radius of the isochronic region (service volume) for chips scales proportionally with process size which describes the length or width of a transistor which has decreased many orders of magnitude over the last 40 years. NOTE: double check the dimensions as described in the computer section.

4. **New assumption statement needed about velocity**: Computer design has been all about increasing clock speed as more components are added to chips, so our general framework needs to address changes in speed across size. In biology, while it is clear that blood flow slows by several orders of magnitude as it flows from the aorta to the capillaries (CITE somebody), scaling models have largely ignored this slowing. WBE assumed constant speed in their model. Banavar assumed that speed increases in larger animals, but that it is constant within an animal. Here we do XXX???

In addition to these general assumptions, we make the following refinements to accommodate salient differences between biology and computer architecture.

1. In biology, the energy processed by a service volume, E_{node}, is invariant with system size.[†] That is, as the service volume increases with body size, the total amount of energy processed remains constant. We do not make this assumption for chips.

2. Component packing: In chips, we assume that total chip area is constant, and the number of transistors (N) is inversely proportional to the square of the process size (process size is the length of one side of a transistor).

3 Predictions of the Model for Organisms and Computers

We define E_{net} and T_{net} respectively to be the energy dissipated and the time taken by the network to deliver a unit of resource to a node. For organisms the resource is oxygen (in mammals, carried by a unit volume of blood), and for computers the fundamental resource is a bit of information. Similarly, we define E_{node} and T_{node} as the energy dissipated and the time taken by a node to process that resource.

For organisms, the node is the service volume corresponding to a region of tissue supplied by a single capillary. For computers, the nodes are transistors, and E_{node} and T_{node} represent the energy dissipated and the time taken to process a bit at the node (i.e., transistor switching energy and delay). In the following, we derive general scaling relationships between E_{net}, T_{net}, E_{node} and T_{node} and the number of nodes in the network N, under the condition that the energy time product is minimized. N is a measure of system size (i.e. the number of capillaries or services volumes in an organism and the number of transistors on a chip). In organisms, larger N implies larger organism volume and mass. For computer chips, N increases by shrinking

[†]IS E_{NODE} THE METABOLIC OUTPUT (MAXIMIZE) OR COST OF METABOLISM (MINIMIZE)?

components, and so increasing N does not imply increasing chip area, which we assume to be constant.

The goal of minimizing the energy time product means that system designs seek to deliver the maximum amount of resource per unit time for the minimum amount of energy overhead. This is equivalent to minimizing the energy dissipated to deliver and process a unit of resource, while simultaneously minimizing the time to deliver and process that resource. Assuming that time and energy are equally important, we seek to minimize the energy-time product, a well-known concept in computer architecture referred to as the the energy-delay product.

We express the optimal network design as a constraint optimization problem in which the whole system's energy-time product is minimized:

$$Minimize(E_{sys} \times T_{sys}) \tag{4}$$

where the system's energy is given as $E_{sys} = E_{net} + E_{node}$, and the system's time as $T_{sys} = max(T_{net} + T_{node})$ due to pipelining.

We derive expressions for E_{sys} and T_{sys} for organisms (Sec. 3.1) and computers (Sec. 3.2) in terms of the dimensions D_l, D_r, and D_w.

3.1 Organisms

In this section, we derive general energy and time scaling relations for the network and nodes in organisms in order to minimize Eq. 4. We first define scaling relationships for the four key quantities E_{net}, T_{net}, E_{node} and T_{node}. We then show how they scale with N, in order to minimize Eq. 4. We note that, in contrast to computer scaling, biological scaling relationships have been examined in many different theoretical models (CITE lots of theoretical models). We do not recapitulate this work, but instead highlight how two different models [**?**, **?**] respectively determined scaling relationships by minimizing energy dissipation and metabolic delivery time

separately. Here we emphasize the consequences of incorporating these approaches into the broader computational scaling framework of minimizing the energy time product.

E_{net}: From basic principles of hydraulics, the energy dissipated to transport a constant volume of blood through the network is given by the loss in pressure from the aorta to the capillaries multiplied by the volume being transported. Pressure is the product between hydraulic resistance (R) and flow (Q), so the loss in pressure $\Delta P = RQ$. Thus $E_{net} \propto \Delta P \propto RQ$. NOTE: this calculation for E_{net} actually is the total for the network, not the energy per unit because it doesn't make sense to calculate R for only one oxygen molecule.

E_{node}: Following [**?**] and (CITE mosesamnat08) we assume that the quantity of energy dissipated to metabolize a fixed quantity of oxygen is independent of metabolic rate and, therefore, $E_{node} \propto N^0$. ADDITION: to calculate the energy consumed by all nodes $E_{node} \propto N$.

T_{net}: The time to deliver a constant volume of blood to a node is given by the volume of blood being transported divided by the flow (Q). Since this volume is constant, $T_{net} \propto 1/Q$. One might ask why there is no distance term in the T_{net} equation. Because of the steady-state assumption, T_{net} is the time it takes to get the 'next' oxygen molecule from a capillary. Thus T_{net} is not the time it takes a single molecule to traverse the network (i.e. it is not τ in [**?**], but rather the inverse of the rate of of delivery of oxygen molecules, analogous to the inverse of clock speed in computer chips.

T_{node}: Following [**?**] and under the pipelining and steady-state assumptions, $T_{node} = T_{net}$, i.e., supply matches demand and so $T_{node} \propto 1/Q$.

Substituting these relationships into Eq. 4 gives $(RQ + N) \times (Q^{-1})$ and the minimization simplifies to

$$Minimize(R + \frac{N}{Q}). \tag{5}$$

where N is the number of terminal units.

We now show how R and Q scale with N. The resistance of a pipe with non-pulsatile flow is given by given by the well-known Hagen-Poiseuille's equation. This applies in the lower region of the cardiovascular network in which the pulsatile flow from the heart beat is attenuated. (To understand how pulsatile flow affects R in the upper pulsatile region of the network, see [**?**], which shows that R is minimized under pulsatile flow when $D_r = 2$ which results in area preserving branching.)

R at hierarchical level i is $R_i = 8\mu l_i/\pi r_i^4$, where μ is the viscosity constant. The total network resistance R is given by [**?**]:

$$R = \sum_{i=0}^{H} \frac{8\mu l_i}{\pi r_i^4} \frac{1}{n_i} = \frac{8\mu l_0}{\pi r_0^4} \lambda^{-H} \sum_{i=0}^{H} \lambda^{i\left(\frac{1}{D_l} - \frac{4}{D_r} + 1\right)} \tag{6}$$

where $H+1$ is the number of hierarchical levels, and $n_i = \lambda^{H-i}$ is the total number of pipes at hierarchical level i. Recalling that $D_l = 3$ in 3 dimensional organisms and that $\lambda^{-H} = N^{-1}$ we now show how D_r affects the scaling of the energy-time product. We consider two relevant cases (derived in Appendix 1):

i) when $D_r <= 3$, the exponent in Eq. 6 is $<= 0$. Simplifying and ignoring constant and logarithmic terms in N gives $R \propto l_0 \lambda^{-H}$ or $R \propto l_0 N^{-1}$. We keep l_0 in the scaling equation even though it was assumed constant in [**?**] because it was predicted to scale with body size in [**?**].

ii) When $D_r > 3$, the the exponent in Eq. 6 is positive, resulting in $R \propto l_0 N^{(\frac{1}{3} - \frac{4}{D_r})}$.

Note that the scaling exponent for R is always minimized by minimizing D_r and the exponent is always smaller for case *i*.

To determine Q, we define flow through a pipe as $Q = u\pi r^2$, where u is the fluid velocity. Flow through the aorta is $Q = u_H \pi r_H^2$, and substituting from Eq. 1, $Q = u_0 \pi r_0^2 \lambda^{2H/D_r} = u_0 \pi r_0^2 N^{\frac{2}{D_r}}$ and ignoring constants $Q \propto u_0 N^{\frac{2}{D_r}}$.

NOTE: Similarly to l_0, we keep u_0 in the scaling equation even though it was assumed

constant in [**?**] because it was predicted shown to scale with body size in [**?**].

Analysis

West et al [**?**] show that E_{net} is minimized when when $D_r = 3$ in the lower non-pulsatile region of the network where R is governed by Eq. 6. Our analysis concurs with this prediction: when $D_r > 3$ then $E_{net} \propto RQ \propto l_0 u_0 N^{(\frac{1}{3} - \frac{4}{D_r})} N^{\frac{2}{D_r}}$ which is minimized when D_r is as small as possible (i.e., approaches 3 from above). Additionally, when $D_r <= 3$ then $E_{net} \propto RQ \propto l_0 u_0 N^{\frac{2}{D_r} - 1}$ which is minimized when D_r is as large as possible (i.e., D_r approaches 3 from below).

However, minimizing E_{net} is only one component of minimizing the energy time product. It is clear that both terms of Eq. 5 (R and $1/Q$) are minimized by minimizing D_r: for $D_r <= 3$, Eq. 5 becomes $minimize(l_0 N^{-1} + u_0^{-1} N^{1 - \frac{2}{D_r}})$.

Thus, noting that the first term dominates if $D_r < 2$, the energy time product is minimized when $D_r = 2$. **ACK: this is no longer correct. The second term dominates now that we've set E_{node} to be the energy of all of the nodes, and therefore proportional to N** This is the area-preserving case in which blood velocity (u) is constant throughout the network. This was assumed by [**?**], and shown to minimize R only in the case of pulsatile flow by [**?**]. Here we show that that the area preserving $D_r = 2$ case optimally minimizes the energy time product even for nonpulsatile Poiseuille flow.

This analysis also sheds light on how evolution has met the additional constraint of slowing blood as it flows from the aorta (empirically at approximately 1 m/s) to the capillaries (less than 1mm/s) both to allow diffusion of oxygen from the capillaries and to avoid rupture of small vessels. Blood slowing requires area increasing branching ($D_r > 2$). $D_r = 3$ is an inflection point beyond which the scaling exponent increases rapidly. By incorporating pulsatile flow that attenuates to non-pulsatile flow, organisms minimize the energy time product under the constraint of needing to slow blood flow to the capillaries.

The final energy and time scaling relations for organisms are

(NOTE: these should go in a table after resolving whether the exponent on T_{node} and T_{net} is $1 - 2/D_r$ or $-2/D_r$)

1. Assume $D_r = 2$ which minimizes the energy time product

$E_{net} \propto RQ \propto l_0 u_0 N^{-1} N^{\frac{2}{D_r}} \propto l_0 u_0 N^0$

$E_{node} \propto N$ (by reference to WBE per node it's N^0, total it's N)

$T_{node} \propto T_{net} \propto 1/Q \propto u_0^{-1} N^{1-2/D_r} \propto u_0^{-1} N^0$

$E_{sys} \propto l_0 u_0 N$ (E_{sys} is dominated by E_{node}, now that we've made the change that I am now suspicious of)

$T_{sys} \propto u_0^{-1} N^0$

2. Assume $D_r = 3$ which allows blood to slow without blowing up the energy time product (or the capillaries)

$E_{net} \propto RQ \propto l_0 u_0 N^{-1} N^{\frac{2}{D_r}} \propto l_0 u_0 N^{(1/3)}$

$E_{node} \propto N$ (by reference to WBE per node it's N^0, total it's N)

$T_{node} \propto T_{net} \propto 1/Q \propto u_0^{-1} N^{-2/D_r} \propto u_0^{-1} N^{-2/3}$

$E_{sys} \propto N$ (now Esys is dominated by E_{node})

$T_{sys} \propto u_0^{-1} N^{-2/3}$

3.2 Computers

We now apply the same reasoning to computer chips. In computers, unlike biology, the nodes (transistors) are not of constant size and have shrunk by many orders of magnitude over 40 years of micro architecture evolution. During this time, chip area has grown much more slowly; we assume chip area to be constant for our calculations. Putting these two constraints together, the linear dimensions of transistors and wires decrease with transistor count as $N^{-1/2}$ (or, more generally, N^{-1/D_l}) [**?**]. This miniaturization process is well understood and is accurately mod-

eled by Dennard's scaling theory [**?**]. Thus, $r_0 \propto l_0 \propto N^{-1/D_l}$. Intuitively, this means that the number of nodes increases by placing smaller transistors closer together and connecting them with smaller and shorter wires.

In the following, we assume that all wires carry the same flow and that information is transferred synchronously. We first calculate how E_{net} scales with N. From basic principles of electronics, the energy dissipated to transmit a bit over a wire is given by the formula $CV^2/2$, where C is capacitance and V is voltage. Although energy depends on V^2, voltage has remained approximately constant over the last four decades (decreasing only by a factor of five while transistor count increased by six orders of magnitude [**?**]), so we estimate that the total energy to transmit all bits over the network scales as C [**?**]. Ignoring fringe effects and for an aspect ratio of 1, wire capacitance is $C_i = \epsilon l_i$ [**?**], where ϵ is the dielectric constant. The network capacitance is the sum of the capacitances of all wires, which is proportional to the total wire length of the network [**?**]:

$$C \propto Length \propto \sum_{i=0}^{H} l_i w_i n_i \propto l_0 w_0 \lambda^H \sum_{i=0}^{H} \lambda^{i\left(\frac{1}{D_l}+\frac{1}{D_w}-1\right)} \tag{7}$$

where l_i is the length of wire, w_i is the number of wires per module, and n_i is the number of modules, all at level i.

Recall that $l_0 \propto N^{-1/D_l}$ and $\lambda^H \propto N$ giving

$$C \propto N^{(1-1/D_l)} \sum_{i=0}^{H} \lambda^{i\left(\frac{1}{D_l}+\frac{1}{D_w}-1\right)} \tag{8}$$

We note that C, and consequently E_{net}, are minimized when the series converges which occurs when $D_w >= D_l/(D_l - 1)$. In this case $C \propto \lambda^H \propto N$. The average energy to transmit a bit over the network is then given by

$$E_{net} \propto C/N \propto N^{-1/D_l} \tag{9}$$

indicating that E_{net} decreases with the number of terminal units in the network. This makes sense intuitively because we assume chip area is constant, so that as N increases the distance between nodes is reduced. Additionally, Rent's rule, reflected in D_w, means that most bits move locally so that the distance distance between nearest nodes affects the average distance that bits are transmitted.

(Not sure if this is necessary): Note that if D_w were smaller than $D_l/(D_l - 1)$ then energy per bit would increase with N, and if D_w were larger than $D_l/(D_l-1)$ there would be no further reduction in the scaling of E_{net}. Because D_w is not a factor in any other term of the energy time product, we set $D_w = D_l/(D_l - 1)$, and given a 2 dimensional chip, $D_w = D_l = 2$.

We now calculate how T_{net}, the time to transmit a bit over a wire scales with N. T_{net} is equivalent to the wire latency which is $L \propto RC$, where R is the wire resistance. For wires with aspect ratio 1, $R_i = \rho l_i / r_i^2$, where ρ is the resistivity of the material. Thus,

$$L_i \propto R_i C_i = \rho\epsilon \frac{l_i^2}{r_i^2} = \rho\epsilon \frac{l_0^2}{r_0^2} \lambda^{i\left(\frac{2}{D_l} - \frac{2}{D_r}\right)}, \tag{10}$$

.

Because flow is synchronous (meaning each wire transfers one bit in parallel on each clock tick), the network must wait until all wires have finished their transmission before initiating the next cycle. Therefore, the time to transfer a bit equals the latency of the slowest wires. We consider 3 cases.

i) $D_r > D_l$ and latency is greatest at the highest hierarchical level, thus

$$T_{net} = L_H \propto \frac{l_0^2}{r_0^2} \lambda^{H\left(\frac{2}{D_l} - \frac{2}{D_r}\right)} = N^{\left(\frac{2}{D_l} - \frac{2}{D_r}\right)}. \tag{11}$$

In this case, because $D_r > D_l$ the exponent in Eqn. 11 is positive, and T_{net} increases with N.

ii) In the case where $D_r < D_l$, the thickness of the wire increases faster than the length at increasing hierarchical levels of the network. Thus, the fastest wires would be at the highest

level of the network, and latency would be dominated by wires at the lowest level: $T_{net} = L_0 \propto \frac{l_0^2}{r_0^2} \propto N^0$.

iii) In the case where $D_r = D_l$, the network has perfect pipelining and the latency is the same at every level of the network: $T_{net} = L \propto \frac{l_0^2}{r_0^2} \propto N^0$.

Of these cases, T_{net} is minimized when $D_l >= D_r$, and since D_r does not affect any of the other terms in the energy-time product (Table **??**), we assume this case giving $T_{sys} \propto N_0$. Thus, network latency, or T_{net} the time for each subsequent bit to be processed by the network is independent of the number of transistors in the network.

We now calculate the scaling of E_{node} and T_{node}. For a single node, computation energy is given by the transistor's (dynamic) energy dissipation as $CV^2/2$. Given that transistor size scales as N^{-1/D_l}, $E_{node} \propto N^{-1/D_l}$. Computation time is given by the transistor delay as CV/I [**?**], which also scales as N^{-1/D_l}. Thus, $T_{node} \propto N^{-1/D_l}$.

Summarizing the scaling relationships:

$E_{net} \propto N^{-1/D_l}$

$E_{node} \propto N^{-1/D_l}$

$T_{net} \propto N^0$

$T_{node} \propto N^{-1/D_l}$

Thus, for 2 dimensional chips in which $D_l = 2$, three components of the energy time product (E_{net}, E_{node}, and T_{node}) scale sub linearly as $N^{-1/2}$ but T_{net} is constant with respect to N.

Putting it all together,

$E_{sys} = E_{net} + E_{node} \propto N^{-1/D_l} \propto N^{-1/2}$

$T_{sys} = max(T_{net}, T_{node}) \propto N^0$.

4 Results and Discussion

The following 2 biology paragraphs need a total rewrite!!

Analysis of Equation **??** determines the optimal value of D_r for organisms. Although energy dissipated by the network decreases as D_r increases, we observe that for $D_r \geq 2$ the energy consumed by the nodes (N^0) dominates the energy dissipated by the network (N^{2/D_r-1}). Therefore, there is no additional benefit to setting $D_r > 2$. On the other hand, time is reduced by decreasing D_r and, therefore, the minimum energy-time product occurs at $D_r = 2$. This result shows that the optimal network design is area preserving, i.e., when the aggregate cross-sectional area of pipes is the same at all hierarchical levels. Various derivations show that this design leads to the 3/4 power scaling of metabolic rate known as Kleiber's law (e.g., [**?**, **?**]).

The model predicts that, for the optimal network design (i.e., $D_l = 3$ and $D_r = 2$), the system's energy-time product is invariant with respect to N. Since size increases with N, this suggests that the energy-time product is independent of body mass. This result has important implications for the energetic basis of fitness. Some have proposed that biological fitness maximizes metabolic power (energy/time) [**?**, **?**], whereas others have proposed that it minimizes biological times (e.g., generation times, which is equivalent to maximizing vital rates) [**?**, **?**]. The invariance of the energy-time product is consistent with the fact that fitness of organisms is largely independent of body mass. Organisms of all sizes, from small, fast, low-power microbes to large, slow, powerful mammals, coexist and, therefore, are likely nearly equally fit. This implies a direct trade-off between maximizing metabolic power and minimizing generation times that holds over the many orders of magnitude variation in body mass. The energy-time product reflects powerful geometric, physical and biological constraints on the evolution of organism designs.

For computers, minimizing the energy time product is trivial: also leading to $D_l = D_r = D_w = 2$,. This result corresponds to ideal scaling, as suggested by Dennard [**?**], where the linear dimensions of transistors and wires scale at the same rate, and wire delay is constant and the Rent's exponent is 1/2, consistent with observations (NEED citations.)

From this result, we make two important predictions. First, power consumption in chips (total energy dissipated per unit of time) scales as $Power = (N \cdot E_{sys})/T_{sys} \propto N^{1/2}$. Performance is given by the number of computations executed per unit of time, or throughput. Second, assuming that a constant fraction of the transistors is active at each cycle, throughput is predicted to scale linearly with N, i.e. $Throughput \propto N/T_{sys} \propto N$.

We compared the predictions for power consumption with data obtained for 523 different microprocessors over a range of approximately 6 orders of magnitude in transistor count. The data are shown in Figure 2, where the measured exponent was 0.495, which agrees closely with the prediction of 0.5. Consistent data on performance across many technology generations is difficult to obtain because the standards have changed over the years and their adoption by different vendors is not uniform. We were able to obtain normalized performance data for 16 Intel processors from different generations over a range of 6 orders of magnitude in transistor count. The exponent obtained for these data was 1.0, as shown in Figure 3, which agrees exactly with the predictions from our analysis. This suggests that engineered designs approach the theoretical optimal defined by the model. The final energy-time product predicted by the model scales as $N^{-1/2}$, showing that, unlike organisms, as size increases, the energy-delay product decreases systematically. This is a consequence of process technology improvement, since newer, larger processors use hardware that is smaller, faster and more energy efficient.

Our model provides a simple theoretical explanation for the scaling of power and performance in computers over 40 years of microprocessor technology improvements. The excellent agreement between the theoretical optimal and experimental data suggests that through successive generations of trial-and-error, innovation and optimization, engineered designs are highly successful, approaching the theoretical limit predicted by the model.

These results also provide an explanation for Koomey's Law, which states roughly: "The number of computations per joule of energy dissipated has been doubling approximately every

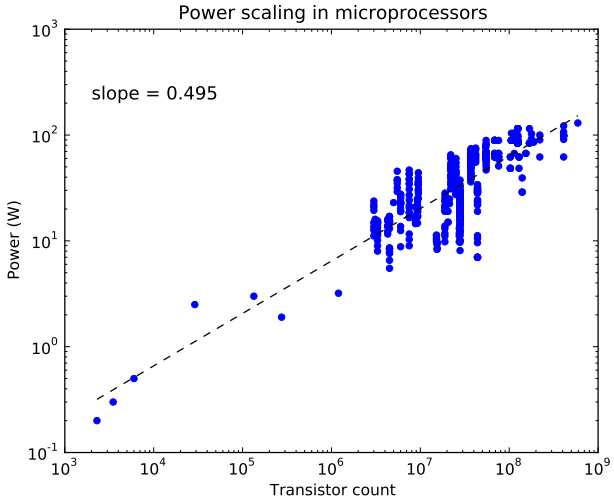

Figure 2:

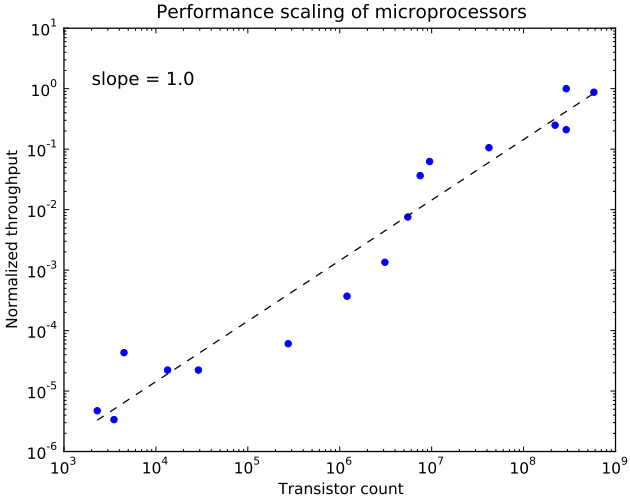

Figure 3:

1.57 years." This follows directly from minimizing the energy time product and Moore's Law, the empirical observation that the number of transistors (N) doubles every 2 years. NEED a figure that combines data from Fig 2 and Fig 3 to show this?

5 Discussion

- By unifying time and energy into a single model, the model accounts for the wide variation in size of organisms and computers, e.g., mouse to elephant and arm to multe-core.
- We provide an explanation for Koomey's Law (p. 13)
- The model shows why computers have given up on reducing clock speeds and gone to multi-core. Once you reach a minimum component size, then you are at a transition and a new scaling regime. What is the regime and why is biology the guide?
- The joule-second as a new fundamental unit of life and information.
- Theoretical explanation for well-known empirical patterns in computer architecture.
- Nature dealt with network design constraints by slowing metabolism as animal size increases. Computer architecture, at least until recently, focused on increase clock speeds by minimizing component size and incresing energy per chip. Also, using the third dimension to accommodate more wires,
- Computer transions: transistors; transistors to integrated circuits; single core to multicore; desk tops to data centers. Multicore is an evolutionary transition. IBM currently making 10 nm chips which are the limit for silicon. IBM recently announced a 7 nm transistor (more than 1000 times smaller than the diameter of a red blood cell) and only three times larger than a strand of DNS, using silicon germanium targeting a 50% power improvement. NYT July 9, 2015. Then going to carbon nanofibers.
- Because two additive terms in E_{sys}, we speculate that they explain the observed curvilinearity in the empirical scaling of metabolism vs. body mass. In organisms with low mass,

the E_{node} term dominates, while in larger animals, the E_{net} term dominates. Let's plot it to demonstrate this.

Similarly, in chips in the old days, when transistors were big, the energy consumption was all in the nodes, but today the energy is in the wires.

- Not sure if this is true until resolving the time exponent: Resistance switches from an increasing function of N to a $N - 4/3$ at $D_r = 3$.

  Time switches from an increasing function of N to a decreasing function of N at $D_r = 2$.

6 Conclusion

Our analysis provides a unifying explanation for the origin of scaling laws in biology and computing. Despite obvious differences in form and function, the scaling of organisms and computers is governed by the same simple principle. By minimizing the energy-time product, whether through natural selection or engineering, existing designs manage the trade-off between cost and performance, leading to general scaling patterns observed over several orders of magnitude in size. Moreover, power laws as a function of size are not unique to organisms and computers but are widely observed across a large variety of complex systems in nature, society and technology. The scaling of white and grey matter in the brain [**?**], of energy use and GDP in countries [**?**], and the pace of life and population in cities [**?**] are additional examples where a unifying explanation is still lacking. Because cost and performance, i.e., energy and time, impose universal constraints, we suggest that a common design principle governs the scaling of complex systems that process energy, materials and information.

Figure legends

Figure 1: Log-log plot of power consumption as a function of transistor count for 523 computer microprocessors from different vendors and technological generations. The linear regression slope is 0.495 with correlation coefficient of 0.81. A regression that removes the first seven data points and focuses on the main cloud of more modern chips produces a slope of 0.487.

Figure 2: Log-log plot of normalized throughput as a function of transistor count for 16 Intel microprocessors from different technological generations. The linear regression slope is 1.0 with correlation coefficient of 0.97.