

V1.0.0

Actual Demand of a Computational Substrate: A Reconciled Demand-Side and Allocation Framework for Physical Computing Platforms

Rowan Brad Quni-Gudzin

QNFO-ULA-2.1

ABSTRACT

Debates about novel computational substrates—reservoir systems, post-qubit quantum architectures, thermodynamically bounded devices—are usually conducted from the supply side: what can a given physical system compute? This paper asks the complementary question posed by the QNFO programme: what is the actual demand placed on a computational substrate, and how should capacity against that demand be provisioned? We reconcile two complementary formalizations. First, a demand-side accounting framework describes a workload by a demand vector over capability axes and a substrate by a supply vector, yielding a match score M and an overprovisioning waste fraction W . Second, an allocation-theoretic framework models the substrate as a capacity-constrained server facing partially observed, non-stationary demand, yielding waste, shortfall, utilization, and a censoring bias. Both are coupled to physical floor costs derived from the Landauer bound and the quantum error-correction overhead of 10^2 – 10^3 physical operations per logical operation reported in the QNFO limits literature. [...]

CITE AS Quni-Gudzin, R. B. (n.d.). Actual Demand of a Computational Substrate: A Reconciled Demand-Side and Allocation Framework for Physical Computing Platforms. QNFO Preprint v1.0.0.

Abstract

Debates about novel computational substrates—reservoir systems, post-qubit quantum architectures, thermodynamically bounded devices—are usually conducted from the supply side: what can a given physical system compute? This paper asks the complementary question posed by the QNFO programme: what is the *actual demand* placed on a computational substrate, and how should capacity against that demand be provisioned? We reconcile two complementary formalizations. First, a demand-side accounting framework describes a workload by a demand vector over capability axes and a substrate by a supply vector, yielding a match score M and an overprovisioning waste fraction W . Second, an allocation-theoretic framework models the substrate as a capacity-constrained server facing partially observed, non-stationary demand, yielding waste, shortfall, utilization, and a censoring bias. Both are coupled to physical floor costs derived from the Landauer bound and the quantum error-correction overhead of 10^2 – 10^3 physical operations per logical operation reported in the QNFO limits literature. With full arithmetic we compute: a Landauer floor of 2.871×10^{-21} J per operation at $T = 300$ K; a per-logical-operation floor of 2.871×10^{-19} to 2.871×10^{-18} J; an illustrative match score $M = 0.800$ with waste $W \approx 0.0732$; a two-state provisioning model with waste = shortfall = 0.168 at mean provisioning; and a censoring bias of 38.2% in naively measured demand. The framework converts substrate advocacy into an auditable ledger of demand, supply, and cost.

1. Introduction

Every proposed computational substrate arrives with a claim about what it will compute. Far fewer arrive with a claim about who will demand that computation, at what rate, with what variance, and at what physical cost floor. This asymmetry is not benign. A substrate whose capacity is provisioned against an imagined demand profile wastes resources when demand is low and fails its users when demand spikes; a substrate whose floor cost per operation exceeds what any realistic workload can pay is stillborn regardless of its theoretical power.

The present paper takes its cue from a question posed in the QNFO corpus: after the critique of qubit-centred ontology, what is the *actually demanded* role of a computational substrate [9], [11]? The phrase "actual demand" is deliberately double-edged. It means both (i) the genuine computational requirements of workloads, as opposed to requirements inherited from an ontological picture, and (ii) the economic sense of demand—whether anyone actually needs the substrate at all.

The demand-side framing is not exotic; it is implicit in how several neighbouring fields already operate. In power systems, demand response is defined by the arrival of elastic or inelastic demands that must be allocated without violating network constraints [2]; the demand is the primitive and the grid is the substrate. In demand forecasting, a foundation model is justified by the specific statistical pathologies of demand series—short histories, frequent zeros, censoring by stock-outs [3]. In navigation, an agent is evaluated on whether the user-specified object is actually present in the scene [6]. In inventory management, demand is partially observed and non-stationary, and the manager must infer the state of the world rather than read it off [8]. Each of these fields treats demand as the object of study and the resource as the constraint. Substrate research in computing has largely inverted this.

We model a substrate along two reconciled axes:

1. **Capability accounting:** what level of each required capability does the substrate supply, and where does it fall short of or exceed demand?
2. **Allocation dynamics:** how do waste, shortfall, utilization, and measurement bias behave when demand is stochastic, partially observed, and censored by capacity?

Both are coupled to a thermodynamic cost floor built from the Landauer bound and the error-correction overhead range stated in [12]. Every quantity is defined by an equation and evaluated with explicit arithmetic in Section 4. No empirical measurements are reported; the numbers in Section 5 are either arithmetic consequences of stated inputs or clearly labelled projections with stated assumptions.

The paper proceeds as follows. Section 2 reviews the related work. Section 3 defines the framework. Section 4 carries out all derivations. Section 5 reports results. Section 6 discusses limitations and failure modes. Section 7 concludes.

2. Background and Related Work

We draw on two literatures that rarely meet: the physics-of-computation and substrate-characterization literature, and the demand-forecasting-and-allocation literature.

Reservoir computing and substrate independence. The substrate-independent framework for reservoir computers [1] states that any nonlinear, input-driven dynamical system exhibiting fading memory and input separability can be trained to perform computational tasks, and that this broad inclusion has led to many new physical substrates for reservoir computing, with essential properties tuned through reconfiguration of the substrate. This is the closest existing analogue to a demand-side vocabulary in physical computing: the framework deliberately abstracts away *what* the substrate is and names only the *properties* a workload-relevant computation requires. Our capability axes in Section 3 generalize exactly this move—fading memory and input separability become two axes among others—and our match score is a scalarization of the property checklist that [1] implies.

Demand as the primitive in power systems. The online demand-response problem [2] allocates incoming elastic or inelastic demands without violating operating constraints of electric networks, in an online fashion, and identifies the presence of non-linear constraints as the distinctive challenge of power systems relative to traditional online decision problems. The structural analogy to substrate evaluation is exact: the workload (demand) arrives exogenously, the substrate (network) has hard constraints, and the allocation algorithm must respect the constraints without knowing the future. In our framework, the apparent power constraint of [2] plays the role of a supply ceiling on a capability axis, and inelastic demand plays the role of a demand component that cannot be reduced by re-choosing the workload.

Demand-specific data pathologies. The EXAONE Demand 1.0 time-series foundation model [3] is motivated by the observation that demand series make up only a small fraction of general time-series corpora, and that demand data has properties such corpora rarely contain: short histories, frequent zeros, censoring by stock-outs, and exogenous events that the series does not record. Each of these pathologies has an exact analogue for a novel computational substrate: deployment histories are short, utilization can be zero for long stretches, observed demand is censored by capacity stock-outs, and exogenous events are not recorded in the usage series. We adopt this taxonomy as the specification of what a substrate-demand forecaster must handle; the supplied entry does not report forecast accuracy numbers, so we make no quantitative claim from it.

Differentiating the substrate's own cost model. The adjoint-computation work [4] differentiates a computational fluid dynamics code using algorithmic differentiation in tangent and adjoint modes, with the novelties that the adjoint code is obtained by letting the AD tool Tapenade invert the complete layer of MPI communications, and that the adjoint code integrates time-dependent, nonlinear and dissipative behaviour. Its relevance here is as a model of *accounting for what the substrate actually does*: the adjoint code is an exact ledger of the solver's operations, inverted. Our overhead multiplier plays the same bookkeeping role at the hardware level, and [4] demonstrates that such ratios can be made exact rather than rhetorical when the computation is instrumented.

Topological invariants of data. The persistent Stiefel–Whitney class construction [5] defines persistent Stiefel–Whitney classes of vector bundle filtrations by seeing vector bundles as subsets of Euclidean spaces, endowing the usual Čech filtration with a vector bundle structure, and showing the construction is stable and consistent, including for finite samples of line bundles. We cite it as a methodological exemplar: it shows how a mathematically delicate property can be made computable and stable on finite data. A demand-side substrate ledger aspires to the same standard—that its quantities survive finite, noisy measurement of the substrate.

Demand-conditioned perception. The demand-driven navigation work [6] studies Visual Object Navigation, in which an agent must locate a particular object in a scene, and identifies two essential conditions: the user must know the name of the desired object, and the user-specified object must actually be present within the scene; a simulator can incorporate these conditions. The second condition—*actual presence*—is precisely the "actual demand" distinction: a demand that cannot be met by the environment is not a planning failure but a supply failure, and conflating the two produces misleading evaluations. Our match score separates them by construction.

Declarative paradigms. The proceedings of the eleventh Workshop on Answer Set Programming and Other Computing Paradigms (ASPOCP 2018), held in Oxford, UK, on July 18th, 2018 [7], document the declarative-programming community's continued interest in computing paradigms beyond the mainstream. The supplied entry is a proceedings record and gives no further detail on individual contributions; we therefore relate it to our argument only through its title and stated scope: answer-set programming is one of the established "other computing paradigms" against which any claim that a new substrate constitutes a *paradigm* shift must be measured.

Partially observed demand. The inventory-management model [8] considers continuous-time inventory management with Markov-modulated non-stationary demands, introduces active learning by assuming the state of the world is unobserved and must be inferred by the manager, assumes demands are observed only when completely met, derives explicit filtering equations, and passes to an equivalent fully observed impulse-control formulation. The "observed only when completely met" assumption is precisely the stock-out censoring that [3] identifies in demand data; for substrates it means that refused jobs are invisible, so the demand process must be filtered, not merely measured. We use the Markov-modulated structure of [8] as the skeleton of our illustrative demand model in Section 3.

The QNFO programme. *Beyond the Qubit* [9] is the companion paper to *The Qubit Delusion* [11] and poses the question this paper formalizes: if the qubit-gate-circuit model is an epistemic failure—a projection of particle ontology onto relational, field-theoretic reality—what comes next? It states that it strips quantum mechanics to its minimal ontological commitments and surveys constructive paradigms for post-particle computation. *No Thing There* [10] is a pedagogical bridge between the qubit ontology critique and working physics, explaining what control pulses actually manipulate and what readout measures, across superconducting transmon, trapped-ion, and spin-qubit platforms, including an electrical seesaw analogy and a self-referential metrology formalization. It supplies the operational vocabulary any demand-side ledger must respect: a capability axis is only meaningful if the substrate's control and readout layer can actually address it. *The Qubit Delusion* [11] is the root critique of the series; the supplied entry is a revision note (v1.1, fixing PDF rendering of Unicode dashes and special characters) and gives no further substantive detail, so we rely on it only as the identified source of the ontological critique that [9] and [10] develop. Finally, *The Physics of Computation* [12] examines fundamental physical limits on computation—the Landauer bound, the Margolus–Levitin theorem, the Bremermann limit, the Bekenstein bound—and states that quantum error-correction overhead of 10^2 – 10^3 physical operations per logical operation multiplies thermodynamic cost. That overhead range is the single quantitative input we take from the literature, and it enters our energy derivation in Section 4 directly.

3. Methods

3.1 Capability axes and vectors

Fix K capability axes, indexed by $k \in \{1, \dots, K\}$, each normalized to $[0, 1]$. A workload is summarized by a demand vector

$$d = (d_1, \dots, d_K), \quad d_k \in [0, 1],$$

where d_k is the minimum level of capability k the workload requires. A substrate is summarized by a supply vector

$$s = (s_1, \dots, s_K), \quad s_k \in [0, 1],$$

where s_k is the level of capability k the substrate delivers at its best documented operating point. Axis levels are *assessed*, not measured here; Section 6 discusses how they would be evidenced, drawing on the control/readout vocabulary of [10] and the property checklist of [1]. Each axis carries a weight $w_k \geq 0$ with $\sum_{k=1}^K w_k = 1$.

Match score. Define per-axis fulfilment

$$m_k = \frac{\min(d_k, s_k)}{d_k}, \quad d_k > 0,$$

so $m_k = 1$ when supply meets or exceeds demand, and $m_k = s_k/d_k < 1$ otherwise. The match score is the weighted mean

$$M(d, s) = \sum_{k=1}^K w_k m_k, \quad 0 \leq M \leq 1.$$

Overprovisioning waste. With $(x)^+ = \max(x, 0)$, the waste fraction is the weighted surplus relative to weighted total supply:

$$W(d, s) = \frac{\sum_{k=1}^K w_k (s_k - d_k)^+}{\sum_{k=1}^K w_k s_k}.$$

W measures capability supplied but not demanded. In the demand-response setting of [2], surplus capacity that cannot be reallocated is precisely the inefficiency online algorithms are designed to avoid; W imports that intuition into substrate evaluation.

3.2 Allocation model

Following the Markov-modulated structure of [8], we model substrate demand as a process $d(t)$ modulated by a hidden world-state $s(t) \in \{h, l\}$ with stationary occupancy probabilities $p_h = p$ and $p_l = 1 - p$. Within regime s , demand takes the value d_s , with $d_h > d_l \geq 0$. Following [8] and the censoring pathology of [3], demand is observed only when completely met, so measured demand $\hat{d}(t) = \min(d(t), C)$ is right-censored by capacity C .

Following [2], demand is partitioned into elastic (deferrable) and inelastic (must be served on arrival) components. Over a horizon T_{horizon} :

$$W_{\text{alloc}}(C) = \int_0^{T_{\text{horizon}}} (C - d(t))^+ dt \quad (\text{waste}), \quad S(C) = \int_0^{T_{\text{horizon}}} (d(t) - C)^+ dt \quad (\text{shortfall}).$$

Shortfall is demand refused, which under censoring is invisible in $\hat{d}$ and must be inferred by a filter in the sense of [8]. Realized utilization is

$$\rho = \frac{D_{\text{act}}}{C T_{\text{horizon}}}, \quad D_{\text{act}} = \int_0^{T_{\text{horizon}}} d(t) dt.$$

Following the separation in [6] between the demanded object and its actual presence, each unit of demand may carry a task-attribute vector; utilization should ultimately be evaluated per attribute class, though no attribute-resolved data is supplied here.

3.3 Cost floor

Following [12], each operation carries a thermodynamic floor. The Landauer bound gives the minimum energy to erase one bit:

$$E_{\min}(T) = k_B T \ln 2, \quad k_B = 1.380649 \times 10^{-23} \text{ J/K}.$$

If error correction requires an overhead factor r physical operations per logical operation, with $r \in [10^2, 10^3]$ as stated in [12], the per-logical-operation floor is

$$E_{\text{logical}}(r) = r E_{\min}(T),$$

and for N_L logical operations the workload floor is $E_{\text{work}} = N_L E_{\text{logical}}$. This is a *floor*, not a prediction: it assumes all physical operations are irreversible bit operations at the Landauer limit, which is optimistic. Its role is to make the multiplier concrete rather than rhetorical, in the spirit of the exact operation ledgers exemplified by [4].

3.4 Procedure

The framework is applied in four steps: (i) elicit d and w_k from the workload; (ii) assess s from documented substrate operation; (iii) compute M , W_{alloc} , S , and ρ ; (iv) compute the energy floor with a stated r and T . Steps (i)–(ii) are judgemental and are where the framework is falsifiable; steps (iii)–(iv) are pure arithmetic, shown in full in Section 4.

4. Analysis

Every input number is stated with its source; every arithmetic step is shown.

4.1 Landauer floor

Inputs. $k_B = 1.380649 \times 10^{-23}$ J/K (SI defined value); $T = 300$ K (assumption: room temperature; [12] states the Landauer bound but not a temperature, so $T = 300$ K is an explicit assumption of this paper); $\ln 2 = 0.693147$.

Step 1. $k_B T = 1.380649 \times 10^{-23} \times 300 = 4.141947 \times 10^{-21}$ J.

Step 2. $E_{\min} = 4.141947 \times 10^{-21} \times 0.693147$. Compute 4.141947×0.693147 : $4.141947 \times 0.7 = 2.899363$; subtract $4.141947 \times 0.006853 = 0.028382$; result 2.870981. Therefore

$$E_{\min}(300 \text{ K}) = 2.870981 \times 10^{-21} \text{ J} \approx 2.871 \times 10^{-21} \text{ J}.$$

4.2 Per-logical-operation floor under error-correction overhead

Inputs. $r \in [10^2, 10^3]$ physical operations per logical operation, stated in [12]; $E_{\min} = 2.870981 \times 10^{-21}$ J from Section 4.1.

Step 1 (lower end). $E_{\text{logical}}(10^2) = 10^2 \times 2.870981 \times 10^{-21} = 2.870981 \times 10^{-19}$ J.

Step 2 (upper end). $E_{\text{logical}}(10^3) = 10^3 \times 2.870981 \times 10^{-21} = 2.870981 \times 10^{-18}$ J.

Step 3 (spread). The ratio of upper to lower floor is $10^3/10^2 = 10$. The uncertainty in the overhead factor alone spans one order of magnitude in the thermodynamic cost of a logical operation, before any implementation inefficiency is counted.

4.3 Worked capability example

Inputs (illustrative assessment, not measurement). $K = 3$ axes: (1) fading memory / state retention, (2) nonlinearity / input separability (both named as essential reservoir properties in [1]), (3) control–readout fidelity (motivated by [10]):

$$d = (0.90, 0.50, 0.80), \quad s = (0.95, 0.20, 0.90), \quad w_1 = w_2 = w_3 = \frac{1}{3}.$$

Match score. Per-axis fulfilment:

$$m_1 = \frac{\min(0.90, 0.95)}{0.90} = \frac{0.90}{0.90} = 1.000, \quad m_2 = \frac{\min(0.50, 0.20)}{0.50} = \frac{0.20}{0.50} = 0.400, \quad m_3 = \frac{\min(0.80, 0.90)}{0.80} = 1.000.$$

Weighted mean:

$$M = \frac{1}{3}(1.000 + 0.400 + 1.000) = \frac{1}{3}(2.400) = 0.800.$$

Waste fraction. Surpluses: $(s_1 - d_1)^+ = 0.05$; $(s_2 - d_2)^+ = 0$ (supply below demand); $(s_3 - d_3)^+ = 0.10$. Weighted surplus numerator: $\frac{1}{3}(0.05 + 0 + 0.10) = 0.05$. Weighted total supply: $\frac{1}{3}(0.95 + 0.20 + 0.90) = \frac{1}{3}(2.05) \approx 0.683333$. Waste fraction:

$$W = \frac{0.05}{0.683333} \approx 0.07317.$$

Interpretation. $M = 0.800$ with the entire shortfall concentrated on axis 2, where supply is 0.20 against demand 0.50. The substrate is not generically weak; it fails on one binding axis. This mirrors the actual-presence condition of [6]: the demand is legitimate, the supply is absent.

4.4 Two-state provisioning: waste, shortfall, utilization

Inputs (illustrative model parameters, not measurements). $p = 0.3$ (fraction of time in the high-demand regime), $d_h = 1$ (normalized peak demand), $d_l = 0.2$.

Case A: provision at peak, $C = d_h = 1$.

- Waste: in the low regime $(C - d_l)^+ = 1 - 0.2 = 0.8$; in the high regime 0. Per unit time, $W_{\text{alloc}} = (1 - p)(d_h - d_l) = 0.7 \times 0.8 = 0.56$.
- Shortfall: $S = 0$ by construction.
- Mean demand per unit time: $p d_h + (1 - p) d_l = 0.3 \times 1 + 0.7 \times 0.2 = 0.44$.
- Utilization: $\rho = 0.44/1 = 0.44$.

Case B: provision at mean demand, $C = \bar{d} = 0.44$.

- Waste: in the low regime $(0.44 - 0.2)^+ = 0.24$, active for fraction 0.7: $W_{\text{alloc}} = 0.7 \times 0.24 = 0.168$.
- Shortfall: in the high regime $(1 - 0.44)^+ = 0.56$, active for fraction 0.3: $S = 0.3 \times 0.56 = 0.168$.
- Utilization: $\rho = 0.44/0.44 = 1.0$ on served demand, but 30% of high-regime demand is refused. Note the exact symmetry $W_{\text{alloc}} = S = 0.168$ at mean provisioning: because demand is bimodal, the mean splits the excess evenly between the two regimes.

Case C: provision at trough, $C = d_l = 0.2$.

- Waste: $W_{\text{alloc}} = 0$.
- Shortfall: in the high regime $(1 - 0.2)^+ = 0.8$, fraction 0.3: $S = 0.3 \times 0.8 = 0.24$.
- Utilization: $\rho = 0.44/0.2 = 2.2 > 1$, i.e., infeasible: mean demand exceeds capacity by a factor 2.2, and the excess 0.24 per unit time is lost and, by the censoring assumption of [8], unmeasured.

Cross-check. In every case, served demand per unit time equals mean demand minus shortfall: Case A: $0.44 - 0 = 0.44$; Case B: $0.44 - 0.168 = 0.272$; Case C: $0.44 - 0.24 = 0.2 = C$. Case C's served demand equals capacity exactly, as it must when capacity binds at all times. The identities hold, confirming the arithmetic.

4.5 Censoring bias of measured demand

Under the observation model $\hat{d}(t) = \min(d(t), C)$, the measured mean at Case B provisioning is

$$\bar{\hat{d}} = p \min(d_h, C) + (1 - p) \min(d_l, C) = 0.3 \times 0.44 + 0.7 \times 0.2 = 0.132 + 0.14 = 0.272.$$

The true mean is 0.44, so the censored measurement understates true demand by

$$\frac{\bar{d} - \bar{\hat{d}}}{\bar{d}} = \frac{0.44 - 0.272}{0.44} = \frac{0.168}{0.44} = 0.3818 \dots,$$

i.e., by 38.2%, equal to the shortfall ratio $S/\bar{d} = 0.168/0.44$. A forecaster training on censored substrate-usage logs—exactly the stock-out censoring pathology named in [3]—would systematically underestimate demand for a capacity-constrained substrate and under-provision it in the next planning cycle. This is the quantitative core of the paper's argument: for novel substrates, naive

demand measurement is biased low by the shortfall ratio.

4.6 Projection: time-to-saturation

Projection, with stated assumptions. Assume demand grows at a compound rate $g = 0.5$ per year (50% annual growth, an assumption for illustration, not a measurement) from an initial mean demand $D_0 = 0.44$ (Section 4.4) against fixed capacity $C = 1$:

$$t_{\text{sat}} = \frac{\ln(C/D_0)}{\ln(1+g)} = \frac{\ln(1/0.44)}{\ln(1.5)} = \frac{\ln(2.2727)}{\ln(1.5)} = \frac{0.82098}{0.40546} = 2.025 \text{ years.}$$

Arithmetic: $\ln 2 = 0.693147$ and $\ln(2.2727/2) = \ln(1.13636) \approx 0.127833$, so $\ln(2.2727) = 0.820980$; $\ln 1.5 = 0.405465$; quotient $= 0.820980/0.405465 = 2.0249$. Under these assumptions the substrate saturates in approximately 2.0 years. If $g = 0.1$ instead, $t_{\text{sat}} = 0.820980/\ln(1.1) = 0.820980/0.095310 = 8.613$ years. The sensitivity is large: the ratio of the two times-to-saturation is $t_{\text{sat}}(g = 0.1)/t_{\text{sat}}(g = 0.5) = 8.613/2.025 \approx 4.25$, so a five-fold change in the growth assumption moves time-to-saturation by a factor of approximately 4.25. Because g is an assumption rather than a measurement, this sensitivity bound is itself a projection and should be read as a statement about the model, not about any substrate.

5. Results

All numbers below are computed in Section 4 with full arithmetic; no empirical measurements are reported.

1. **Landauer floor.** $E_{\min}(300 \text{ K}) = 2.871 \times 10^{-21}$ J per irreversible bit operation (Section 4.1).
2. **Per-logical-operation floor.** With the error-correction overhead $r \in [10^2, 10^3]$ stated in [12], $E_{\text{logical}} = 2.871 \times 10^{-19}$ to 2.871×10^{-18} J per logical operation; the overhead uncertainty alone spans one order of magnitude (Section 4.2).
3. **Capability match.** For the illustrative assessment of Section 4.3, $M = 0.800$ with waste fraction $W \approx 0.0732$; the entire shortfall is concentrated on a single axis (nonlinearity / input separability), where supply is 0.20 against demand 0.50.
4. **Provisioning trade-off.** In the two-state model of Section 4.4 ($p = 0.3$, $d_h = 1$, $d_l = 0.2$): provisioning at peak gives waste 0.56 and shortfall 0; provisioning at the mean demand $\bar{d} = 0.44$ gives the exact symmetry waste = shortfall = 0.168; provisioning at the trough gives waste 0 and shortfall 0.24, with utilization $\rho = 2.2 > 1$ (infeasible). Cross-check identities hold in all three cases.
5. **Censoring bias.** At mean provisioning, the censored measured mean is $\hat{d} = 0.272$ against a true mean of 0.44, an understatement of 38.2%, equal to the shortfall ratio $S/\bar{d}$ (Section 4.5).
6. **Time-to-saturation (projection).** Under the stated assumptions $D_0 = 0.44$, $C = 1$, and growth $g = 0.5$ per year, $t_{\text{sat}} \approx 2.0$ years; at $g = 0.1$, $t_{\text{sat}} \approx 8.6$ years; the sensitivity factor between the two assumptions is ≈ 4.25 (Section 4.6).

6. Discussion

Limitations. The capability assessment in Section 4.3 is illustrative, not measured: the demand and supply vectors are judgements, and the framework's falsifiability lives entirely in how they are elicited and evidenced. The energy floor is optimistic by construction—it assumes every physical operation is an irreversible bit operation at the Landauer limit, so real substrates sit strictly above it. The allocation model is a two-state Markov abstraction; real demand has more regimes, correlation structure, and attribute heterogeneity, none of which is captured here. The time-to-saturation figure is a projection resting entirely on an assumed growth rate, and its sensitivity factor of ≈ 4.25 shows how weakly identified it is.

Failure modes. Three are visible in the arithmetic. First, censoring: a planner who measures demand from censored logs understates it by the shortfall ratio (38.2% in the worked case) and under-provisions in the next cycle, compounding the error. Second, single-axis failure: a high match score ($M = 0.800$) can conceal a total failure on one binding axis; scalarization should always be reported alongside the per-axis fulfilment vector m_k . Third, floor-cost stillbirth: if a workload requires N_L logical operations, the floor $E_{\text{work}} = N_L E_{\text{logical}}$ with $E_{\text{logical}} \geq 2.871 \times 10^{-19}$ J may exceed any plausible energy budget regardless of the substrate's theoretical power.

What would falsify the claims. The framework's central quantitative claim is that naive demand measurement on a capacity-constrained substrate is biased low by exactly the shortfall ratio $S/\bar{d}$. This would be falsified if measured demand on a real substrate were unbiased—for example, if refused demand were logged rather than invisible, contradicting the censoring assumption inherited from [8] and [3]. The provisioning symmetry (waste = shortfall at mean provisioning) would be falsified by any demand distribution for which the mean does not split the excess evenly; it already fails for continuous or skewed distributions, and the two-state case is the special symmetric instance.

Open questions. How should the weights w_k be elicited in a way that is auditable rather than arbitrary? Can the filtering equations of [8] be adapted to produce unbiased demand estimates from censored substrate logs? And does the error-correction overhead range of [12] apply outside the quantum setting it was stated for, or must each substrate derive its own r ?

7. Conclusion

We have reconciled a demand-side capability-accounting framework with an allocation-theoretic model of capacity-constrained, partially observed demand, and coupled both to a thermodynamic cost floor built from the Landauer bound and the error-correction overhead of 10^2 – 10^3 physical operations per logical operation stated in [12]. With all arithmetic shown, the framework yields: a floor of 2.871×10^{-21} J per bit operation at 300 K; 2.871×10^{-19} to 2.871×10^{-18} J per logical operation; an illustrative match score of 0.800 with waste 0.0732; a provisioning trade-off in which waste and shortfall are exactly equal (0.168) at mean provisioning; a censoring bias of 38.2%; and a projected time-to-saturation of 2.0 years under a 50%-growth assumption, sensitive at a factor of ≈ 4.25 to that assumption. The contribution is not a measurement but an auditable ledger: substrate advocacy, in this framing, must state its demand vector, its supply vector, its provisioning rule, and its cost floor, each with shown arithmetic.

References

- [1] A Substrate-Independent Framework to Characterise Reservoir Computers. arXiv:1810.07135v2. <https://arxiv.org/abs/1810.07135v2>
- [2] Online Algorithm for Demand Response with Inelastic Demands and Apparent Power Constraint. arXiv:1611.00559v2. <https://arxiv.org/abs/1611.00559v2>
- [3] EXAONE Demand 1.0: A Time Series Foundation Model for Demand Forecasting. arXiv:2609.30880v1. <https://arxiv.org/abs/2609.30880v1>
- [4] Adjoint computations by algorithmic differentiation of a parallel solver for time-dependent PDEs. arXiv:1912.11717v3. <https://arxiv.org/abs/1912.11717v3>
- [5] Computing persistent Stiefel-Whitney classes of line bundles. arXiv:2005.12543v3. <https://arxiv.org/abs/2005.12543v3>
- [6] Find What You Want: Learning Demand-conditioned Object Attribute Space for Demand-driven Navigation. arXiv:2309.08138v3. <https://arxiv.org/abs/2309.08138v3>
- [7] Proceedings of the eleventh Workshop on Answer Set Programming and Other Computing Paradigms 2018. arXiv:1812.03508v3. <https://arxiv.org/abs/1812.03508v3>
- [8] Inventory Management with Partially Observed Nonstationary Demand. arXiv:1206.6283v1. <https://arxiv.org/abs/1206.6283v1>
- [9] <https://papers.qnfo.org/papers/paper-beyond-the-qubit/>. QNFO: Beyond the Qubit: Constructive Paradigms for Post-Particle Computation.
- [10] <https://papers.qnfo.org/papers/no-thing-there/>. QNFO: No Thing There: Control, Readout, and Self-Referential Metrology in Engineered Quantum Systems.
- [11] <https://papers.qnfo.org/papers/paper-the-qubit-delusion/>. QNFO: The Qubit Delusion: How Particle Ontology Sabotaged Quantum Computing.
- [12] <https://papers.qnfo.org/papers/paper-physics-of-computation/>. QNFO: The Physics of Computation: Fundamental Limits and the Honest Boundaries of Post-Classical Computing.

Appendix A. Divergence report

The source drafts agreed on all substantive framework claims: the two reconciled formalizations, the Landauer floor derivation, the error-correction overhead range taken from [12], the worked capability example, the two-state provisioning cases, and the censoring-bias identity. No divergent claims between drafts were identified; the only correction applied during reconciliation was arithmetic (the time-to-saturation sensitivity factor, originally misstated, recomputed as $8.613/2.025 \approx 4.25$ in Section 4.6). No convention choice was therefore required.

Appendix B. Claim attribution

Claim	Source drafts	Agreement
C1: Demand-side capability vectors with match score M and waste W	A, B, C	CONVERGENT
C2: Allocation model with waste, shortfall, utilization under censoring	A, B, C	CONVERGENT
C3: Landauer floor 2.871×10^{-21} J at 300 K	A, B, C	CONVERGENT
C4: Per-logical-operation floor 2.871×10^{-19} – 2.871×10^{-18} J from overhead 10^2 – 10^3 in [12]	A, B, C	CONVERGENT
C5: Illustrative match $M = 0.800$, $W \approx 0.0732$	A, B	CONVERGENT
C6: Two-state provisioning, waste = shortfall = 0.168 at mean	A, B, C	CONVERGENT
C7: Censoring bias 38.2% equal to shortfall ratio	A, B, C	CONVERGENT
C8: Time-to-saturation projection and sensitivity factor ≈ 4.25	A, B (corrected)	CONVERGENT after arithmetic correction