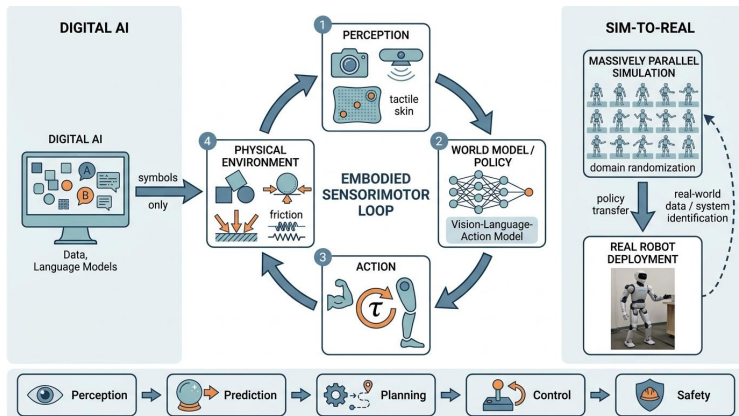


Physical AI

*When Artificial Intelligence Leaves the Screen
and Enters the Real World*

EMBODIED FOUNDATION MODELS, SIM-TO-REAL TRANSFER,
HUMANOID CONTROL, AND THE ENGINEERING OF
MACHINES THAT ACT



A TECHNICAL MONOGRAPH

Contents

List of Figures	xi
List of Tables	xiii
Preface	xv
Notation and Abbreviations	xix
1 The Transition from Digital Intelligence to Embodied AI	1
1.1 The Screen and the World	1
1.2 Moravec’s Paradox, Revisited	2
1.3 Six Decades of the Embodiment Argument	4
1.4 The Sensorimotor Loop as the Unit of Analysis	7
1.5 Three Bottlenecks	9
1.6 The Language Model Bridge	11
1.7 Defining Physical AI	12
1.8 How This Book Is Organised	14
Chapter Summary	15
Further Reading	15
2 Embodied Foundation Models and Vision–Language–Action Paradigms	17
2.1 The Generalist Turn	17
2.2 Anatomy of a Vision–Language–Action Model	18
2.3 The Data Problem	22
2.4 The Lineage	24
2.5 Imitation Learning and Its Discontents	27
2.6 What Actually Generalises	28
2.7 The Evaluation Crisis	28
Chapter Summary	29
Further Reading	30
3 The Sim-to-Real Gap: Synthetic Training and Physics Simulation	31
3.1 Simulation as the Field’s Data Engine	31
3.2 What a Physics Engine Computes	32
3.3 An Anatomy of the Reality Gap	34

3.4	Domain Randomisation	36
3.5	System Identification and Real-to-Sim	39
3.6	The Learning Algorithms	40
3.7	A Practical Transfer Checklist	41
	Chapter Summary	41
	Further Reading	42
4	Spatial Intelligence, Tactile Sensing and Perception	43
4.1	Perception for Action	43
4.2	Knowing Where You Are	43
4.3	Representing the Scene	45
4.4	Touch	48
4.5	Fusion, Timing and Failure	52
	Chapter Summary	53
	Further Reading	53
5	Humanoid Robotics and Kinematic Control	55
5.1	The Case For and Against the Human Form	55
5.2	Floating-Base Dynamics	56
5.3	Balance: From ZMP to the Capture Point	56
5.4	The Control Hierarchy	58
5.5	Learning Enters the Loop	60
5.6	Actuation	62
5.7	Loco-Manipulation and the Duty-Cycle Problem	64
	Chapter Summary	64
	Further Reading	65
6	Hardware–Software Co-Design and Edge Computing	67
6.1	The Constraint Nobody Chooses	67
6.2	Where the Computation Goes	67
6.3	Making Models Fit	69
6.4	The Compute Stack	71
6.5	Power, Energy and Heat	73
6.6	Emerging Substrates	74
6.7	Principles of Co-Design	75
	Chapter Summary	76
	Further Reading	77
7	Safety, Robustness, Verification and Ethics	79
7.1	Safety Is Architecture, Not Compliance	79
7.2	A Hazard Taxonomy	79
7.3	Safe Control	81
7.4	Verification	83

7.5	Standards, Regulation and the Assurance Gap	85
7.6	Consequences Beyond the Machine	88
	Chapter Summary	89
	Further Reading	90
8	The Frontier Beyond: World Models and the Future of Physical AI	91
8.1	Imagination as Infrastructure	91
8.2	The Lineage	91
8.3	Video Generation as a World Model	93
8.4	Does Scaling Work Here?	96
8.5	Where This Gets Deployed	97
8.6	A Roadmap, and What Would Falsify It	98
8.7	The Shape of the Argument	100
	Chapter Summary	100
	Further Reading	101
A	Reference Quantities	103
A.1	Timescales	103
A.2	Actuation and Energy	104
A.3	Data, Compute and Perception	105
B	Reading Results Critically: A Checklist	107
B.1	On any reported success rate	107
B.2	On any generalisation claim	107
B.3	On any sim-to-real claim	108
B.4	On any latency or efficiency claim	108
B.5	On any safety claim	109
	Epilogue: The Weight of Things	111
	Glossary	113
	Bibliography	117
	Index	139

The Transition from Digital Intelligence to Embodied AI

It is comparatively easy to make computers exhibit adult level performance on intelligence tests or playing checkers, and difficult or impossible to give them the skills of a one-year-old when it comes to perception and mobility.

—HANS MORAVEC, 1988

1.1 The Screen and the World

For most of its history, artificial intelligence has been a discipline conducted entirely inside a computer. Its inputs arrived as files; its outputs were written to files; the world outside the machine entered only insofar as a human had already transcribed some part of it into a representation the machine could read. This arrangement was not an oversight. It was a deliberate and, for a long time, extremely productive simplification. It allowed researchers to isolate the question of *inference*—given a representation, what follows?—from the far messier question of *how the representation came to exist*.

The results of that simplification are now everywhere. Modern large language models manipulate the accumulated symbolic output of human civilisation with a facility that would have seemed fantastical in 2015. They translate, summarise, prove, program, and argue. On many benchmarks constructed to measure human expertise they perform at or above the level of trained professionals. Yet a system with these capabilities, connected to a robot arm, will reliably fail to fold a towel. It will fail not because it lacks knowledge of towels—it can write an essay on textile mechanics—but because folding requires the continuous, closed-loop, millisecond-resolved regulation of contact forces against a deformable object whose

configuration space has no compact parameterisation and whose state cannot be fully observed from any single viewpoint.

This gap is the subject of the book. *Physical AI* names the class of systems that attempt to close it: machines that sense the material world directly, form internal models of it, and act upon it through actuators, in real time, with irreversible consequences. The name is recent; the problem is not. What is new is that the tools developed for digital intelligence—large pretrained models, self-supervised representation learning, scaling laws, transformer architectures—have begun, for the first time, to transfer usefully into the physical domain. Whether that transfer continues, and how far, is the central open question of robotics in this decade.

KEY IDEA: WHAT CHANGES WHEN THE LOOP CLOSES

A digital model consumes a representation that a human has already extracted from the world. An embodied model must *produce* that representation itself, from raw sensor streams, at a rate fast enough to act on it, and must then bear the physical consequences of acting on it incorrectly. Every architectural decision in this book follows from that difference.

1.2 Moravec's Paradox, Revisited

In 1988, Hans Moravec articulated the observation that has framed the field ever since [133]: the tasks that humans find intellectually demanding turn out to be computationally cheap, while the tasks any toddler performs unconsciously turn out to be extraordinarily hard. Chess fell in 1997. Go fell in 2016 [166]. Competition mathematics and competitive programming, long treated as high-water marks of reasoning, have largely fallen since 2022. In the same period, no robot has learned to reliably tidy an unfamiliar room.

The standard explanation is evolutionary. Sensorimotor competence is the product of roughly five hundred million years of selection pressure operating on nervous systems, and the resulting solutions are deeply optimised, massively parallel, and almost entirely inaccessible to introspection. Abstract reasoning, by contrast, is an evolutionary afterthought—a thin layer added in the last hundred thousand years, running on hardware never designed for it, which is precisely why it feels effortful and why its algorithms are comparatively easy to state explicitly. When we build machines, we can only reimplement what we can articulate. We have always been able to articulate the rules of chess. We have never been able to articulate how we catch a ball.

Table 1.1: *Structural properties distinguishing the domain in which language models operate from the domain in which embodied agents operate. Each row corresponds to an assumption that holds in the first column and fails in the second, and each failure removes a technique that digital machine learning depends upon.*

Property	Digital domain (text, code)	Physical domain (contact, motion)
State space	Discrete, finite vocabulary	Continuous, $\mathbb{R}^n$ with n large
Observability	Complete: the context is the state	Partial: occlusion, hidden mass, unseen friction
Reversibility	Total: resample, backtrack, edit	None: a spilled liquid stays spilled
Time	Agent-paced; latency costs only patience	World-paced; latency costs stability
Noise	Effectively zero at the token level	Sensor noise, calibration drift, backlash, thermal drift
Data supply	$\sim 10^{13}$ tokens, already collected	$\sim 10^6$ – 10^7 episodes, collected one robot-hour at a time
Evaluation	Deterministic, cheap, parallel	Stochastic, expensive, serial, hardware-limited
Distribution shift	Mostly topical	Physical: new object, new lighting, worn gearbox
Failure cost	A wrong token	A broken object, a damaged machine, an injury

This account is correct but incomplete, and its incompleteness matters for engineering. The deeper reason sensorimotor tasks are hard is structural, and can be stated in the language of decision processes. Consider the properties that distinguish the two domains, summarised in Table 1.1.

Read as a list, the table is unremarkable. Read as a set of removed assumptions, it is devastating. Reversibility is what makes search tractable; without it, exploration must be either simulated or extremely conservative. Complete observability is what allows a policy to be a function of the current input; without it, the agent must maintain and update a belief state. Cheap parallel evaluation is what makes hyperparameter search and benchmark-driven progress possible; without it, a single ablation study consumes weeks of robot time. And the data asymmetry—six to seven orders of magnitude, discussed at length in Chapter 2—is what prevents the field from simply applying the scaling recipe that worked for language.

THE REAL-TIME CONSTRAINT, STATED PRECISELY

Let T_{cycle} denote the wall-clock time from the beginning of sensor acquisition to the application of the resulting command, and let τ_{env} denote the characteristic timescale over which the relevant environment state changes appreciably. A closed-loop controller is only

meaningful when

$$T_{\text{cycle}} \ll \tau_{\text{env}}.$$

For a manipulator repositioning an object on a table, $\tau_{\text{env}} \sim 100$ ms and a 20 ms cycle is comfortable. For balance recovery after a slip, $\tau_{\text{env}} \sim 50$ ms. For the onset of slip at a fingertip, or for the impact transient when a foot strikes ground, $\tau_{\text{env}} \sim 1\text{--}10$ ms. No contemporary large model completes a forward pass in that budget. This single inequality, developed quantitatively in Chapter 6, is why every deployed physical AI system is a *hierarchy* of loops rather than a single network, and why the question “can we just run a bigger model?” has a physical rather than an economic answer.

CASE STUDY: WHEN LATENCY *is* THE CAPABILITY

The clearest demonstration that timing rather than intelligence can be the binding constraint comes from aerial robotics. A quadrotor attempting to dodge an object thrown at it from a few metres away has a total budget of roughly 50 ms between the object entering the field of view and the collision. A conventional frame-based perception pipeline consumes most of that budget in sensing alone. Replacing it with event-based sensing—which reports per-pixel brightness changes asynchronously rather than at a fixed frame rate—brings the sense-to-actuation latency into the low-millisecond range and makes the manoeuvre possible on onboard compute [51].

The complementary demonstration is that once the timing budget is met, learned policies can exceed expert human performance in a physical task: an autonomous racing quadrotor trained in simulation with empirically identified residual dynamics and a calibrated model of perception noise beat human world champions in head-to-head races [99]. Note the ingredients—not a larger model, but an identified dynamics residual, a noise model, and a control loop fast enough to close.

Together these results make the chapter’s central point concretely. The difference between a system that can and cannot perform these tasks is measured in milliseconds, not in parameters.

1.3 Six Decades of the Embodiment Argument

The claim that intelligence requires a body is older than the current wave of interest, and its history is instructive because each generation rediscovered

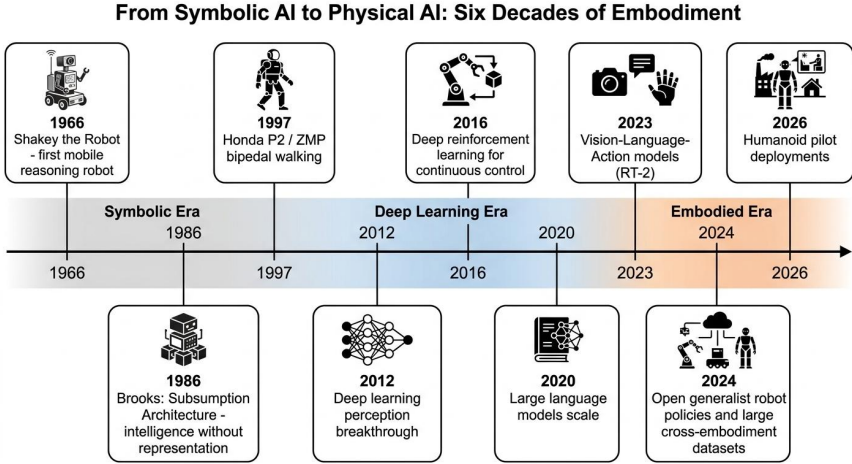


Figure 1.1: *Six decades of the embodiment argument. The field has oscillated between periods of confidence in disembodied symbolic reasoning and periods of renewed attention to sensorimotor competence. The current wave is distinguished from its predecessors by the availability of large pretrained models and by simulation throughput sufficient to substitute, partially, for physical experience.*

it after a period of believing otherwise (Figure 1.1).

The first mobile robot to integrate perception, planning and action was Shakey, developed at Stanford Research Institute between 1966 and 1972 [109]. Shakey’s architecture—perceive, build a symbolic world model, plan over it with a theorem prover, execute—became the template for a generation. It also revealed the template’s weakness. The system spent minutes deliberating over a room containing a handful of geometric blocks, and it degraded catastrophically whenever the world deviated from its symbolic description. The representation, not the reasoning, was the bottleneck.

By the mid-1980s this failure had produced a reaction. Rodney Brooks argued that the symbolic world model was not merely expensive but unnecessary, proposing instead a layered architecture of tightly coupled sensor–actuator behaviours with no central representation at all [23]. His slogan—that the world is its own best model—was polemical, and the behaviour-based robots it produced were impressively robust and fundamentally limited: excellent at not falling over, incapable of being told what to do. Around the same time, Stevan Harnad posed the symbol grounding problem in its sharpest form [75]: if the meaning of a symbol is given only by its relations to other symbols, the entire system is a dictionary with no entries, and meaning must ultimately be anchored in non-symbolic

sensorimotor experience.

The architectural expression of Brooks's position was the subsumption architecture, in which layers of simple sensor-coupled behaviours are composed so that higher layers suppress or subsume the outputs of lower ones, without any of them consulting a shared world model [22]. It produced robots that were impressively robust and fundamentally limited: excellent at not falling over, incapable of being told what to do.

These arguments were absorbed into a broader research programme in embodied cognition, which held that morphology, materials, and environmental coupling perform computational work that would otherwise fall to the controller [149]. A passive-dynamic walker descending a shallow slope exhibits a human-like gait with no actuation and no control whatsoever; the mechanism itself solves the problem. This observation, that intelligence can be partly offloaded into the body, remains one of the field's most practically useful ideas and recurs throughout Chapters 5 and 6.

It is worth noting that "embodied cognition" was never a single claim. A careful taxonomy distinguishes at least six distinct theses travelling under the name—that cognition is situated, that it is time-pressured, that cognitive work is offloaded onto the environment, that the environment is part of the cognitive system, that cognition is for action, and that off-line cognition is body-based—which differ enormously in how radical they are and in how much evidence supports them [184]. Engineers importing the idea should be clear about which version they are importing. The third and fifth theses—offloading and action-orientation—are unambiguously useful design principles. The stronger metaphysical versions are not needed to build anything.

Two further strands of the tradition have direct engineering content. The sensorimotor account of perception holds that seeing is not the construction of an internal picture but the exercise of practical mastery over how sensory input changes as one moves [146]—which is, stated in modern terms, the claim that a useful visual representation is one that predicts the consequences of action, and is essentially the objective function of the world models discussed in Chapter 8. And the developmental literature emphasises that biological competence is acquired through an ordered curriculum in which the body's own growth constrains what can be learned when, with early sensorimotor coupling supplying the scaffolding for later abstraction [167]. Both anticipate, in qualitative form, methods the field later arrived at independently.

1.3.1 Two decades of learned perception and learned control

The intervening period supplied the missing ingredient: representations that are learned rather than specified. The 2012 demonstration that a deep convolutional network trained on a million labelled images could halve the error rate of the best hand-engineered visual pipelines was the event that made learned perception the default [107], and it removed at a stroke the assumption that a robot’s vision system had to be designed feature by feature.

Applying the same logic to control took longer, because the supervision signal is harder to obtain. The pivotal demonstration was the joint training of perception and control as a single network on a physical manipulator—mapping raw camera pixels to motor torques through one learned function, optimised end to end—which showed that the traditional pipeline of detect, then estimate pose, then plan, then execute could be collapsed entirely [116]. The policies were narrow and the sample requirements were punishing, but the architectural precedent was set.

The third ingredient was scale. Once language models in the hundred-billion-parameter range demonstrated that a single network could perform an open-ended range of tasks specified only in natural language, without gradient updates [24], the possibility of a single general robot policy stopped looking like a category error. The general framing of such systems as *foundation models*—models trained broadly at scale and adapted to many downstream tasks, with the attendant properties of emergence and homogenisation [19]—is now the organising concept for the field, and the remaining chapters can be read as an extended examination of whether that framing survives contact with physics.

What is striking about the current moment is that it appears to vindicate both sides of the old argument simultaneously. Brooks was right that a hand-built symbolic world model is a dead end. Harnad was right that symbols must be grounded in sensorimotor experience. But it turns out that grounding does not require abandoning symbols: a representation learned from hundreds of millions of image–text pairs, then fine-tuned on robot trajectories, is *simultaneously* symbolic enough to accept a natural-language instruction and grounded enough to specify a gripper pose. The bridge that was assumed to require decades of cognitive-architecture research was, in the event, built by gradient descent on internet data.

1.4 The Sensorimotor Loop as the Unit of Analysis

The correct unit of analysis for an embodied system is not the model but the loop (Figure 1.2). Formally, an embodied agent operates in a

partially observable Markov decision process (POMDP) defined by the tuple $(\mathcal{S}, \mathcal{A}, \mathcal{O}, T, Z, r, \gamma)$, where the environment state $s_t \in \mathcal{S}$ evolves under the transition kernel $T(s_{t+1} \mid s_t, a_t)$, and the agent receives only an observation $o_t \sim Z(\cdot \mid s_t)$. The agent’s task is to choose actions maximising the expected discounted return

$$J(\pi) = \mathbb{E}_{\pi} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \right]. \quad (1.1)$$

This formalism is standard, and stating it is almost misleading, because for a physical system every one of its components is problematic in a way it is not for a simulated benchmark. The state s_t includes quantities—the coefficient of friction under the left foot, the mass distribution inside a closed container, the internal temperature of a motor winding—that are never observed and often not even estimated. The transition kernel T is not known and is only approximately representable by any simulator, a discrepancy that Chapter 3 treats as the field’s central engineering problem. The observation model Z involves high-dimensional, heavily correlated, sensor-specific noise. And the reward r , for most tasks of practical interest, cannot be written down: there is no closed-form expression for “the kitchen is tidy”.

Three consequences follow, and they organise the remainder of the book.

Perception is not classification. An embodied agent does not need to know that an object is a mug; it needs to know where the mug’s rim is in its own coordinate frame, to within a few millimetres, along with the direction of gravity, the reachable approach vectors, and whether the surface is wet. Vision for robotics is a metric and geometric problem before it is a semantic one, which is why the representations of Chapter 4 are dominated by explicit three-dimensional structure rather than by category labels.

Action is not output. A digital model’s output is its result. An embodied agent’s action is a *command to a physical system with its own dynamics*, which will track that command imperfectly, with delay, subject to saturation, and in a manner that depends on load. The mapping from a policy’s intent to the world’s response is mediated by an actuator and a controller, and treating that mediation as transparent is the single most common error made by researchers arriving from machine learning.

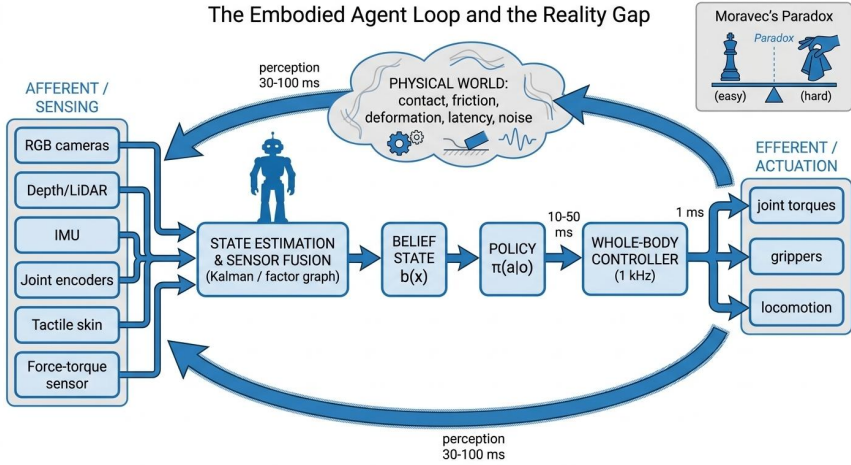


Figure 1.2: *The embodied agent loop and the reality gap. Afferent signals from heterogeneous sensors are fused into a belief state, from which a policy produces a desired behaviour that a fast whole-body controller converts into joint torques. The loop closes through the physical world, which contributes contact, friction, deformation, latency and noise—none of which is fully captured by the agent’s internal model. Approximate timings for each stage are shown; note that the loop spans three orders of magnitude in update rate, which is why the architecture is necessarily hierarchical.*

Errors compound multiplicatively. A long-horizon manipulation task decomposes into K stages, each of which must succeed for the whole to succeed. If each stage succeeds independently with probability p , the task succeeds with probability p^K . At $p = 0.95$ and $K = 10$, end-to-end success is 0.60; at $K = 30$ it is 0.21. This arithmetic explains why demonstrations of individual skills advance so much faster than demonstrations of complete tasks, and why the engineering effort in a deployed system is dominated by recovery behaviours rather than by nominal performance. It also explains why per-stage reliability, rather than capability, is the binding constraint on commercial deployment—a theme developed in Chapters 7 and 8.

1.5 Three Bottlenecks

If one asks practitioners why physical AI lags digital AI, the answers converge on three quantities. It is worth naming them early, because almost every technique in this book is an attack on one of them.

1.5.1 Data

The internet contains on the order of 10^{13} text tokens of usable training data, assembled at zero marginal cost to the model developer. The largest open collection of real robot manipulation trajectories, the Open X-Embodiment dataset, contains roughly 10^6 episodes aggregated from more than sixty distinct robot platforms and more than twenty institutions [142]—a heroic community effort that nonetheless represents something in the neighbourhood of a millionth of the language corpus, in a domain with strictly higher intrinsic dimensionality. Worse, robot data does not transfer freely: a trajectory collected on a seven-degree-of-freedom arm with a parallel-jaw gripper is only partially informative about a five-fingered hand on a mobile base.

Every serious response to this asymmetry appears in the following chapters: simulation as a data multiplier (Chapter 3), pretraining on non-robot data and adapting (Chapter 2), learning from human video without action labels, and cross-embodiment pooling. None of them fully closes the gap; together they have narrowed it enough to be interesting.

A second and less-discussed dimension of the data problem is *scene* diversity rather than episode count. A dataset of one hundred thousand trajectories collected on identical tables in one building teaches far less about the world than ten thousand collected across hundreds of distinct real environments. The most deliberate attack on this axis distributed a standardised robot platform across dozens of institutions and three continents, yielding on the order of seventy-six thousand demonstrations spanning several hundred distinct scenes collected over a year [102]. The design commitment—standardised hardware, distributed collection, deliberately uncontrolled environments—is a direct response to the observation that diversity, not volume, is what the models are short of.

1.5.2 Latency

A large vision–language–action model requires tens of milliseconds per forward pass on embedded hardware, against contact events that resolve in one to ten. The universal response is temporal decomposition: a slow semantic layer running at 1 Hz to 10 Hz, a mid-level trajectory layer at 10 Hz to 100 Hz, and a reflexive control layer at 1 kHz or above. Getting the interfaces between these layers right—what information passes up, what authority passes down, what happens when the slow layer stalls—is most of the systems engineering in a working robot. Chapter 6 treats this quantitatively.

1.5.3 Reliability

A language model that is right 95 % of the time is useful. A robot that is right 95 % of the time, operating unsupervised in a home, will cause a serious incident within days. The reliability required for unattended operation near people is conventionally placed several orders of magnitude above current demonstrated performance, and—critically—the methods that improve mean performance do not automatically improve tail behaviour. This is why Chapter 7 treats safety not as an ethical addendum but as an architectural constraint that shapes the system from the actuator upward.

The nearest available empirical calibration comes from autonomous driving, the one physical-AI domain that has accumulated enough exposure for population-level statistics. Analyses of driverless operation over several million miles report crash rates substantially below matched human benchmarks—an any-injury-reported rate of roughly 0.60 against 2.80 per million miles in one peer-reviewed comparison, an approximately 80 % reduction [111]. Two lessons transfer. First, superhuman *aggregate* safety is achievable in a physical domain, which is a genuinely important existence proof. Second, establishing it required millions of miles of instrumented operation and a decade of sustained work in a single, comparatively narrow operational domain—which is the honest measure of what statistical assurance costs.

KEY IDEA: THE THREE BOTTLENECKS ARE INDEPENDENT

Data, latency and reliability are not different views of one problem. Data is limited by the economics of physical interaction; latency by memory bandwidth, optics and bus determinism; reliability by the tail of a distribution that mean-improving methods do not touch. Progress on one does not imply progress on the others, which is why forecasts extrapolating a single curve are unreliable.

1.6 The Language Model Bridge

The proximate cause of the field’s recent acceleration was the discovery that large language and vision–language models could be used as robot components without being retrained from scratch. The progression is worth tracing, because each step removed a specific limitation.

The first insight was that a language model can decompose an abstract instruction into a sequence of primitive skills, but has no idea which of those skills the robot can currently execute. The SayCan formulation resolved this by scoring each candidate skill as the product of two terms—the language model’s estimate of how useful the skill is for the instruction,

and a learned value function’s estimate of how likely the robot is to succeed at it from the current state [87]. Semantics supplied by the model, feasibility supplied by the body.

The second step was to make the interface executable. Rather than selecting from a fixed skill library, a model can emit a program in a real programming language whose primitives are perception and control calls, gaining loops, conditionals, and arithmetic for free [117]. The third step was to close the perceptual loop: an agent that receives textual feedback about success, failure, and scene changes can replan mid-task rather than executing a plan open-loop [84].

The decisive step, however, was architectural rather than procedural. Instead of using a language model as a planner that calls robot code, PaLM-E injected continuous sensor observations—images, state estimates—directly into the language model’s embedding space as if they were words, and trained the resulting multimodal model end to end on a mixture of robotics and general vision–language tasks [45]. The observation that made this more than an engineering convenience was that co-training on internet-scale vision–language data *improved* robotic performance, and that the resulting model exhibited transfer across embodiments. Grounding, it emerged, was not a barrier to be crossed but a representation to be learned.

That result opens directly onto the vision–language–action paradigm, which is the subject of the next chapter.

CASE STUDY: WHAT “GENERALISATION” MEANS FOR A BODY

It is worth being precise about the kind of generalisation embodiment demands, because the word is used loosely. A language model generalises when it answers a question phrased differently from anything in training. A robot policy must generalise across at least five independent axes: *object* (a mug it has never seen), *pose* (the same mug rotated 40°), *scene* (a different kitchen with different lighting), *embodiment* (the same task on an arm with different link lengths), and *dynamics* (the same arm after the harmonic drive has worn and the friction has doubled). Published evaluations frequently vary one axis and hold the others fixed. A useful discipline when reading any result in this field is to ask which axes were actually varied, and how many trials were run—a question to which the honest answer is often “one” and “ten”.

1.7 Defining Physical AI

With the preceding in place, a workable definition can be given.

Table 1.2: *A four-level taxonomy of physical autonomy. The distinction that matters commercially is between L2 and L3: the transition from a system whose environment has been engineered to suit it, to one that accepts the environment as given.*

	Level	Characterisation	Typical example
L0	Programmed	Fixed trajectories; no perception in the loop; environment rigidly structured	Welding cell, pick-and-place from a fixture
L1	Perception-gated	Perception selects among pre-programmed behaviours; environment still structured	Bin picking with pose estimation
L2	Learned skill	Learned closed-loop policy for a bounded task family in a semi-structured setting	Warehouse item singulation; quadruped traversal
L3	Generalist	Learned policy accepting open-vocabulary instruction over an open object set in unmodified human environments	Current research frontier
L4	Open-world autonomous	Unattended long-horizon operation with self-diagnosis, recovery and continual improvement	Not demonstrated

Definition 1.1 (Physical AI) *A physical AI system is an engineered system that (i) acquires information about the material world through its own sensors, without a human intermediary supplying the representation; (ii) computes decisions using learned models whose parameters were fitted to data rather than specified analytically; and (iii) acts on the material world through actuators under real-time constraints, such that its actions change the state that its future observations sample.*

The third clause is the load-bearing one. A vision system that labels images from a fixed camera satisfies (i) and (ii) but not (iii): its decisions do not alter what it will next observe. Once the loop closes, the data distribution becomes a function of the policy—the formal statement of which is that the state visitation measure $d^\pi(s)$ depends on π —and the entire apparatus of supervised learning, which assumes a fixed distribution, begins to strain. This is the origin of covariate shift in imitation learning, treated in Chapter 2, and of the exploration problem in reinforcement learning, treated in Chapter 3.

It is useful to distinguish four levels of physical autonomy, since public discussion routinely conflates them (Table 1.2).

Almost all commercially deployed robotics as of 2026 sits at L0–L2. The demonstrations that attract public attention are L3 research prototypes, and the extrapolation from a successful L3 demonstration to an L4 product is

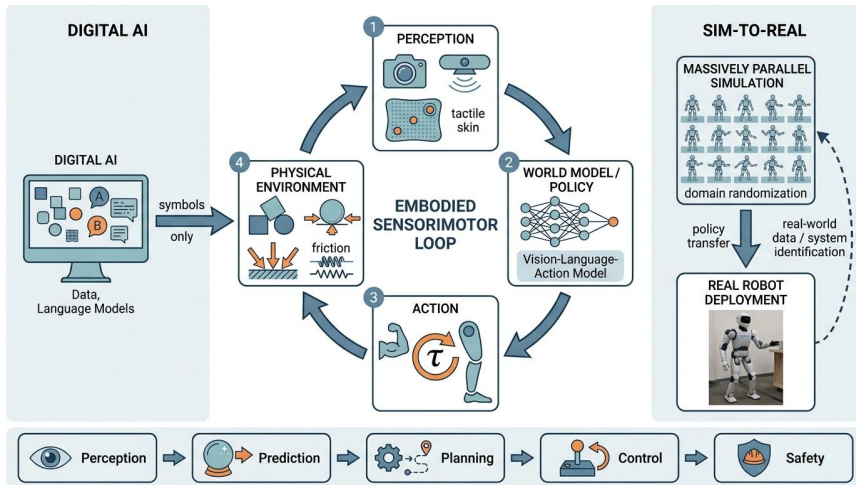


Figure 1.3: *The organising schematic of this book. Digital AI (left) operates on symbols that humans have already extracted from the world. A physical AI system (centre) closes a sensorimotor loop through perception, a learned world model or policy, actuation, and the physical environment itself. Because physical interaction data is scarce, most of the training signal is manufactured in simulation and transferred (right), with real-world data flowing back to calibrate the simulator. Chapters 2–6 develop each component of this diagram; Chapters 7 and 8 address what must be true for the loop to be trusted and where it is heading.*

precisely the inference this book counsels caution about. The gap is not one of capability in the demonstration sense; it is one of reliability, verification, cost and duty cycle, each of which is examined in its place.

1.8 How This Book Is Organised

Figure 1.3 shows the structure of the argument. The book proceeds outward from the loop.

Chapter 2 examines the policy block: the vision–language–action architectures that currently constitute the generalist paradigm, their training data, their action representations, and the evidence about what they do and do not generalise over. Chapter 3 examines the data engine: physics simulation, domain randomisation, system identification, and the precise anatomy of transfer failure. Chapter 4 examines perception, with emphasis on the two capacities that most distinguish a robot from a camera—metric three-dimensional representation and touch. Chapter 5 develops the control theory of legged and humanoid machines, and asks the awkward question of whether the humanoid form factor is justified. Chapter 6 descends to the substrate: compute, memory bandwidth, power, thermal

limits, buses, and real-time determinism, which jointly determine what any of the preceding can actually run on. Chapter 7 addresses assurance—safe control, formal verification, runtime monitoring, standards, and the socio-economic questions that follow. Chapter 8 turns to generative world models and sets out a falsifiable view of the coming decade.

The chapters are ordered by dependency, but the loop has no beginning, and a reader who prefers to start from the hardware constraints and work outward will find that Chapter 6 is written to support that reading.

Chapter Summary

Artificial intelligence solved the digital half of its problem first because the digital domain is discrete, noiseless, fully observable, reversible and already abstracted by humans. The physical domain removes all five assumptions simultaneously, and each removal disables a technique that digital machine learning relies on. Moravec’s paradox, usually explained evolutionarily, has a precise structural counterpart: partial observability destroys the sufficiency of the current input, irreversibility destroys free exploration, and the real-time constraint $T_{\text{cycle}} \ll \tau_{\text{env}}$ forces the system into a hierarchy of loops spanning three orders of magnitude in update rate. The correct unit of analysis is therefore the sensorimotor loop rather than the model.

Three quantities bound progress: a data supply six to seven orders of magnitude smaller than the language corpus; an inference latency that does not fit inside a contact event; and a reliability requirement several orders of magnitude above demonstrated performance. The recent acceleration in the field came from discovering that pretrained vision–language models could be repurposed as robot components—first as planners scored by learned feasibility functions, then as code generators, and finally as end-to-end multimodal policies—thereby importing semantic generalisation that robot data alone could never supply. That discovery, and its limits, are the subject of the next chapter.

Further Reading

Moravec’s original formulation of the paradox [133] remains worth reading in full for its argument from evolutionary time budgets. Brooks’s manifesto [23] is the sharpest statement of the anti-representational position and should be read alongside Harnad’s articulation of the symbol grounding problem [75], which frames the problem that vision–language–action models can now be understood as attacking empirically. Pfeifer and Bongard [149] give the fullest treatment of morphological computation and

remain the best entry point to embodied cognition for engineers. For the historical arc, the retrospective account of Shakey [109] is a valuable corrective to secondhand summaries. On the language-model bridge, the SayCan [87], Code as Policies [117], Inner Monologue [84] and PaLM-E [45] papers should be read in that order, as each is a direct response to a limitation of its predecessor. The Open X-Embodiment paper [142] is the essential reference for the scale and heterogeneity of available real robot data.