

FUNDAMENTALS OF AI-AUGMENTED ENGINEERING: A CONVERSATIONAL APPROACH

Taner Onsay, Ph.D. & Sebnem Onsay
Authors
August 2026 · Revision 1.0

© 2026 AI First Engineering

Problem



Model



Compute



Analyze



Validate



Improve

The engineering workflow you will use throughout this book

by Taner & Sebnem Onsay,

CHAPTER - 4

PART 1 · ≈ 35 MIN

Written a New Way — $f(v) = 0$

Chapter 3 ended with a promise: the same problem returns one more time, written a new way. A thousand other engineering questions are waiting behind it.

Problem

Model

Compute

Analyze

Validate

Improve



CONTENTS OF CHAPTER 4:

Chapter 3 asked: how much should we trust the number the computer hands back? **Chapter 4 asks:** can the computer earn that number intelligently — with a guarantee instead of a march?

1

Written a new way — $f(v) = 0$

The same 50 m of road becomes an equation · one shape behind four problems you already solved



2

Trap it — the bracket & the sign change

Chapter 2's last-safe / first-unsafe pair gets its name · a guarantee before any hunting begins



3

Hunt it — bisection

Halve, and halve again · by hand, then in Python · accuracy on a schedule · 2,037 evaluations vs. about a dozen



4

Trust it — guarantees have conditions

When bisection lies with a straight face · BREAK THE AI audits a guarantee · validate & decide



Prediction before computation is used throughout — as always. Every Part returns to problems you have already solved; what is new is how intelligently they can be solved.



A promise made at the end of Chapter 3

THE TOOLKIT YOU BUILT

error · resolution · convergence · tolerance

You can measure an answer's accuracy — and say what accuracy the decision actually requires.

THE COST WALL

2,037 evaluations

for two decimal places at step 0.01. Each $\times 10$ refinement cost $\times 10$ the work. The march was honest — and expensive.

THE PROMISE

**“The same problem returns one more time, written a new way:
 $f(v) = 0$.”**

Chapter 3's closing sentence. Today is that return.

STILL STANDING

RUNS $\neq$ CORRECT

CONVERGED $\neq$ CORRECT

Chapters 2 and 3, undiminished. Nothing in this chapter retires them — one new rung is coming.

Chapter 3 taught you what accuracy costs. This chapter asks whether that price was ever necessary.

Problem



Model



Compute



Analyze



Validate



Improve



DISCUSS

After 41 coarse evaluations, you already knew: the answer lives between 20.0 and 20.5.

The fine march then spent 2,037 evaluations to earn two decimals. What knowledge did the march ignore — and what would a smarter engineer do with “it’s between 20.0 and 20.5”?

Say it in plain words. Whatever you propose, you are about to meet its formal name.

Problem



Model



Compute



Analyze



Validate



Improve



From “how fast may I go?” to $f(v) = 0$

THE CHAPTER 2 QUESTION

“The largest v with $d_{\text{stop}}(v) \leq 50$ m.”

You answered it by walking v upward until the model said “too far” — and reporting the last safe step.

THE SAME QUESTION, AS AN EQUATION

$$f(v) = d_{\text{stop}}(v) - d_{\text{avail}}$$

The answer is the v where $f(v) = 0$ — the ROOT. Crossing zero and crossing the limit are the same instant.

STOPS SHORT — SAFE

$$f(v) < 0$$

$f(20.0) = -1.43$ m: stops with 1.43 m of road to spare.

THE ROOT — THE EDGE

$$f(v) = 0$$

$v_{\text{ref}} = 20.368$ m/s: the stopping distance exactly uses the road.

OVERRUNS — UNSAFE

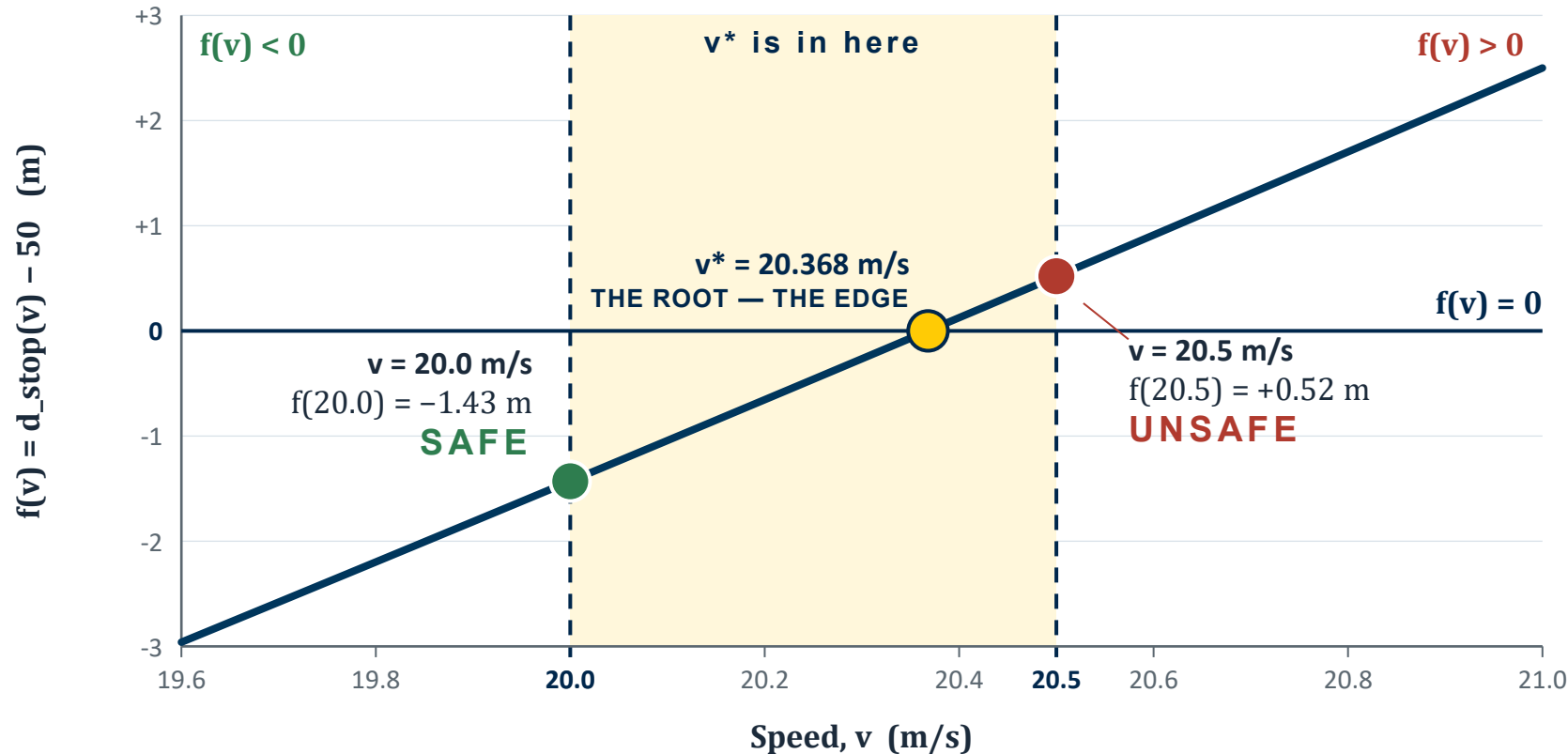
$$f(v) > 0$$

$f(20.5) = +0.52$ m: passes the hazard by half a meter.

The sign of f is not algebraic decoration — it carries the engineering meaning.
Negative: road to spare. Positive: overrun. Zero: the edge the decision cares about.



See the root — don't search the whole road



USE WHAT YOU KNOW

THE ROOT IS TRAPPED.

$$20.0 < v^* < 20.5$$

Negative on the left, positive on the right. A **smooth** f cannot change sign without crossing zero — so the crossing is inside.

Chapter 2 arithmetic, unchanged:

$$d_{\text{stop}}(20.0) = 48.57 \text{ m}$$

$$d_{\text{stop}}(20.5) = 50.52 \text{ m}$$

“Why keep searching everywhere else?” If the answer already lives between 20.0 and 20.5, every evaluation outside that interval is wasted.

Problem



Model



Compute



Analyze



Validate



Improve



One shape, four problems you already own

THE ROAD · CHAPTERS 2–3

$$f(v) = d_{\text{stop}}(v) - 50 \text{ m}$$

Where does stopping distance cross the available road?
You walked to this root at six different step sizes — and paid 2,037 evaluations for two decimals.

THE MUG · LAB 3

$$f(t) = T(t) - 40 \text{ }^{\circ}\text{C}$$

When does the cooling curve cross drinkable? Your thermometer trapped the crossing between two readings — a bracket, measured by hand.

THE BASIN · HOMEWORK 3

$$f(t) = h(t)$$

When does the water depth reach zero? Your simulation crossed it between two time steps — and the PASS / FAIL verdict depended on where.

THE PUMP · SELF-ASSESSMENT 3

$$f(Q) = \Delta p(Q) - \Delta p_{\text{required}}$$

Where does delivered pressure cross the requirement? Your incremental search reported the last safe flow — one step below the crossing.

Solve this one shape well, and you have solved all four — and thousands more. That is why numerical methods are a discipline, not a recipe per problem.

Problem

**Model**

Compute



Analyze



Validate



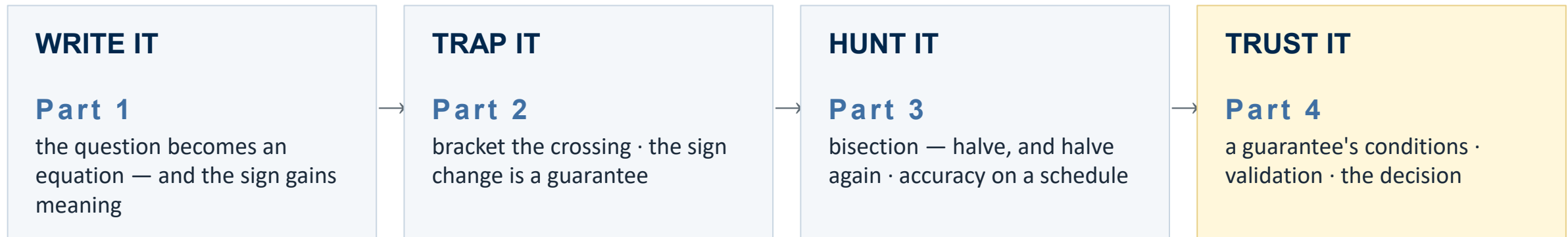
Improve



How do we find the crossing without walking the whole road?

CHAPTER 4 · PRINCIPLE No. 1

A THOUSAND ENGINEERING QUESTIONS, ONE MATHEMATICAL SHAPE: $f(x) = 0$.



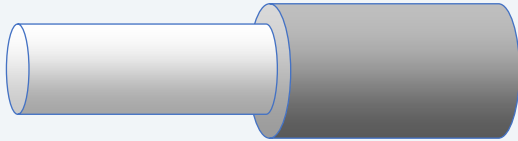
Every Part returns to problems you have already solved. No new physics enters this chapter — again.



A machinist already knows something about intelligent search

Internship story: machining hydraulic-lift rods from stock material — fast when far away, careful when the drawing material (finer cuts) starts to matter.

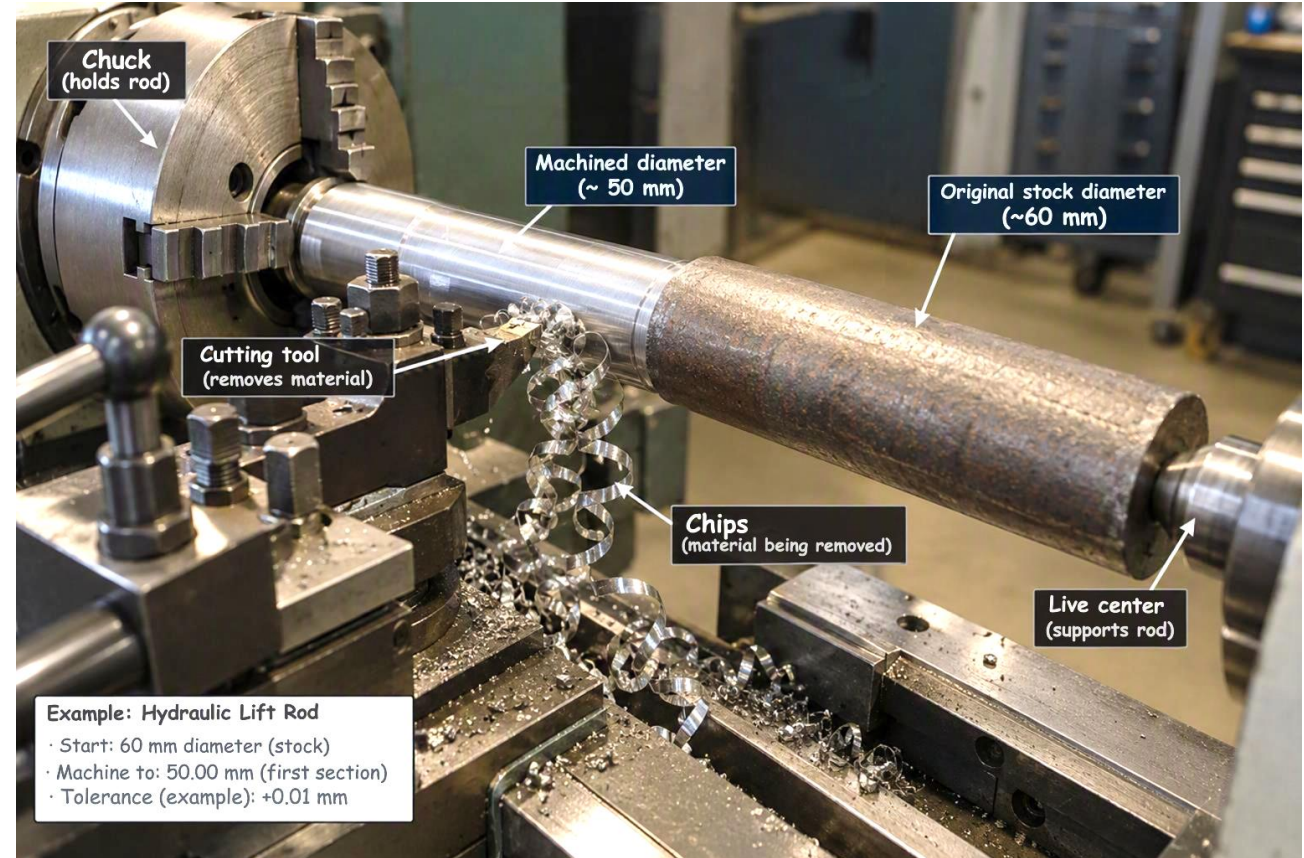
THE PART



50.00 ± 0.01 mm ← **60 mm**
FINISHED DIAMETER STOCK

$$49.99 \leq D \leq 50.01 \text{ mm}$$

Instructional tolerance for this example



PRECISION HAS A COST.

Far away, speed matters. Near the requirement, information matters more.

Problem

Model

Compute

Analyze

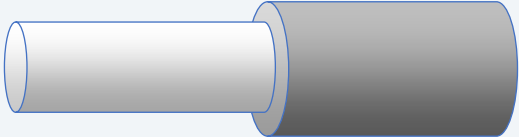
Validate

Improve

Intelligent search

Concept: Search with finer steps as you get close to your target zone.

THE PART

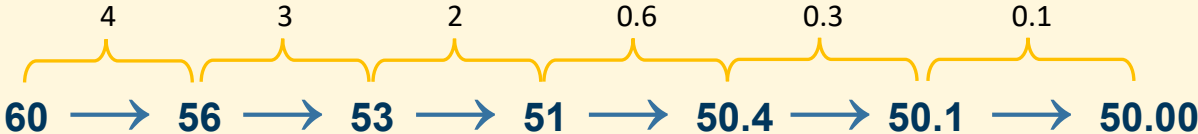


50.00 ± 0.01 mm ← **60 mm**
FINISHED DIAMETER STOCK

49.99 ≤ D ≤ 50.01 mm

Instructional tolerance for this example

WHAT EXPERIENCE CHANGES

4	3	2	0.6	0.3	0.1							
												
60	→	56	→	53	→	51	→	50.4	→	50.1	→	50.00

ROUGH MACHINING

larger cuts
measure less often

APPROACH

smaller cuts
measure more often

FINISH / VERIFY

tiny corrections
frequent checks

DISCUSS: Why not use the smallest cut and measure after every pass starting at 60 mm?

And why not keep taking large cuts all the way to the target?

COST BUILDS UP IN MASS PRODUCTION:

Machining only one part compared to 100,000 parts is the difference. Significant cost may be added to the part due to slow process with too many checks, or scrapped out of tolerance parts.

The drawing defines the target — the search should adapt

WRITE IT

Let x = machining-process setting
(time, travel, or controlled material removal)

$$f(x) = D(x) - 50.00$$

$f(x) > 0$ OVERSIZED — more material may
be removed

$f(x) = 0$ NOMINAL TARGET

$f(x) < 0$ UNDERSIZED — you went too far

THE ENGINEERING REQUIREMENT

$$49.99 \leq D(x) \leq 50.01 \text{ mm}$$

equivalently: $-0.01 \leq f(x) \leq +0.01 \text{ mm}$

THE MACHINIST'S STRATEGY

Far away → move fast.

Getting close → reduce the step.

Near tolerance → measure carefully.

Inside tolerance → STOP.

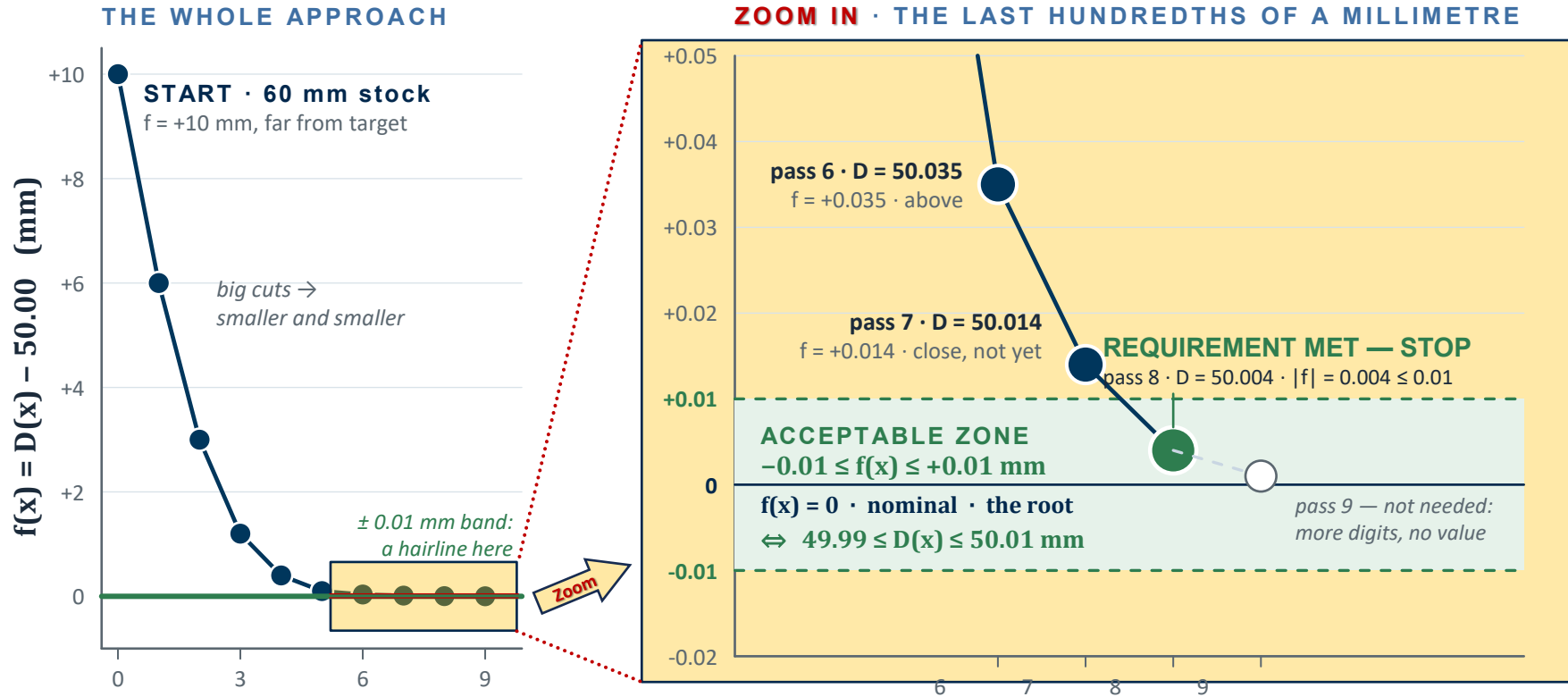
But a physical cut is irreversible: you cannot put the metal back.

The computer can explore candidate settings without destroying the part. Can it use what it learns to close in intelligently?

CHAPTER 3 → CHAPTER 4. Tolerance says how close is close enough. Now ask how to reach it without wasting evaluations.



You don't always need the exact root: $|f(x)| \leq 0.01 \text{ mm}$



Machining pass · search step

COMPARE WITH SLIDE 7

SLIDE 7 · ROOT FINDING

Trap the root: $f(v) = 0$

NOW · ENGINEERING TOLERANCE

Find any x : $|f(x)| \leq 0.01 \text{ mm}$

THE STOPPING RULE CHANGED.

The root is still $f(x) = 0$. The band is the acceptable region around it — the decision only has to land inside.

Chapter 3 told us how close is close enough.
 Chapter 4 asks how to get there intelligently.

STOP WHEN THE ENGINEERING REQUIREMENT IS MET. Inside $|f(x)| \leq 0.01 \text{ mm}$, more digits buy nothing.

Problem



Model



Compute



Analyze



Validate



Improve