

Advanced mathematics & numerical methods

Use numerical methods without trusting the computer blindly

Starting question — How do we know a numerical answer is close enough to the physical answer?

Intuition. Numerical work is controlled approximation: the step size, conditioning, reference values and convergence evidence matter as much as the final digits.

- Explain the governing idea before calculating.
- Name the units, evidence source and operational boundary of every key quantity.
- Check the result with an inverse, bound, refinement or independent physical estimate.

This module marks a decisive step: mathematics is no longer a collection of formulas to plug numbers into, but a language for building, testing and criticizing models. A trajectory, navigation filter, thermal budget or reliability estimate rapidly becomes a multi-variable problem whose quantities evolve with time. The objective is not to train a pure mathematician, but to develop the habits needed to understand what engineering software is computing and to recognize a physically absurd result.

Before you start — Prerequisites: modules 01 to 09 recommended. Every important symbol is defined again at first use.

Mastery objectives

- explain concepts with units and assumptions
- redo a simple calculation by hand before using a tool
- identify at least one failure mode or model limitation
- connect the discipline to a complete Mars architecture

Zero-prerequisite concepts

These four concept cards establish the vocabulary needed before the calculations and mission decisions in this module.

numerical error

Definition. Numerical error is the discrepancy introduced by finite precision, discretisation or an approximate algorithm.

Example. A trajectory propagated with a coarse time step can diverge from a finer reference even when the physics model is unchanged.

Pitfall. A small displayed residual does not prove the numerical method is globally accurate.

Mental check. Change step size or precision and check whether the result converges toward a stable value.

Guided exercise — numerical error

In a Mars mission scenario, identify one situation in which “numerical error” changes an engineering or operational decision. State the evidence you would inspect, the mistake you must avoid, and one independent check you would perform before accepting the decision.

Detailed correction — numerical error

Core meaning. Numerical error is the discrepancy introduced by finite precision, discretisation or an approximate algorithm.

Mission example. A trajectory propagated with a coarse time step can diverge from a finer reference even when the physics model is unchanged.

Error to reject. A small displayed residual does not prove the numerical method is globally accurate.

Independent check. Change step size or precision and check whether the result converges toward a stable value.

Quantification

Use the physical unit that belongs to numerical error when it is quantitative; if it is qualitative, do not invent a numerical unit.

Verification

Compare the conclusion with the mission example, the stated pitfall and the mental check before using numerical error operationally.

convergence

Definition. Convergence means successive refinements of a method approach a stable solution or reduce a defined residual.

Example. A Newton solver may require several iterations before the residual falls below a chosen tolerance.

Pitfall. Stopping because two printed numbers look similar can hide a poorly scaled residual.

Mental check. Track both the change in the solution and a physically meaningful residual.

Guided exercise — convergence

In a Mars mission scenario, identify one situation in which “convergence” changes an engineering or operational decision. State the evidence you would inspect, the mistake you must avoid, and one independent check you would perform before accepting the decision.

Detailed correction — convergence

Core meaning. Convergence means successive refinements of a method approach a stable solution or reduce a defined residual.

Mission example. A Newton solver may require several iterations before the residual falls below a chosen tolerance.

Error to reject. Stopping because two printed numbers look similar can hide a poorly scaled residual.

Independent check. Track both the change in the solution and a physically meaningful residual.

Quantification

Use the physical unit that belongs to convergence when it is quantitative; if it is qualitative, do not invent a numerical unit.

Verification

Compare the conclusion with the mission example, the stated pitfall and the mental check before using convergence operationally.

time step

Definition. The time step is the interval advanced by one numerical integration update.

Example. An orbit propagator may be compared at 60 s, 20 s and 10 s steps to check sensitivity.

Pitfall. A smaller step is not automatically “correct”; it also increases computation and can expose other numerical issues.

Mental check. Halve the step and compare a mission-relevant output, not only the number of iterations.

Guided exercise — time step

In a Mars mission scenario, identify one situation in which “time step” changes an engineering or operational decision. State the evidence you would inspect, the mistake you must avoid, and one independent check you would perform before accepting the decision.

Detailed correction — time step

Core meaning. The time step is the interval advanced by one numerical integration update.

Mission example. An orbit propagator may be compared at 60 s, 20 s and 10 s steps to check sensitivity.

Error to reject. A smaller step is not automatically “correct”; it also increases computation and can expose other numerical issues.

Independent check. Halve the step and compare a mission-relevant output, not only the number of iterations.

Quantification

Use the physical unit that belongs to time step when it is quantitative; if it is qualitative, do not invent a numerical unit.

Verification

Compare the conclusion with the mission example, the stated pitfall and the mental check before using time step operationally.

conditioning

Definition. Conditioning describes how strongly a problem’s output changes when its inputs are perturbed slightly.

Example. Nearly singular navigation geometry can amplify tiny measurement errors into large position uncertainty.

Pitfall. A stable algorithm cannot repair a fundamentally ill-conditioned physical geometry.

Mental check. Perturb inputs within their realistic uncertainty and observe whether the output changes disproportionately.

Guided exercise — conditioning

In a Mars mission scenario, identify one situation in which “conditioning” changes an engineering or operational decision. State the evidence you would inspect, the mistake you must avoid, and one independent check you would perform before accepting the decision.

Detailed correction — conditioning

Core meaning. Conditioning describes how strongly a problem’s output changes when its inputs are perturbed slightly.

Mission example. Nearly singular navigation geometry can amplify tiny measurement errors into large position uncertainty.

Error to reject. A stable algorithm cannot repair a fundamentally ill-conditioned physical geometry.

Independent check. Perturb inputs within their realistic uncertainty and observe whether the output changes disproportionately.

Quantification

Use the physical unit that belongs to conditioning when it is quantitative; if it is qualitative, do not invent a numerical unit.

Verification

Compare the conclusion with the mission example, the stated pitfall and the mental check before using conditioning operationally.

Calculation laboratory — formula, units, inverse check and limits

Each relation is taught from zero prerequisites: what it asks, what every symbol means, why the operation is valid, how to calculate it and how to catch a wrong result independently.

Canonical quantitative mini-lessons

This block is the module quantitative reference; exercise values are pedagogical and must not be confused with mission data.

Relative error

$$e_{\text{rel}} = |x_{\text{calc}} - x_{\text{ref}}| / |x_{\text{ref}}|$$

1 — Concrete question

What does " $e_{rel} = |x_{calc} - x_{ref}| / |x_{ref}|$ " compute in "Relative error"?

2 — Intuition without symbols

Relative error scales the discrepancy by the reference value so problems of different sizes can be compared.

3 — Quantities

e_{rel} : relative error; x_{calc} : calculated value; x_{ref} : reference value

4 — Formula

$$e_{rel} = |x_{calc} - x_{ref}| / |x_{ref}|$$

5 — Read aloud

Read " $e_{rel} = |x_{calc} - x_{ref}| / |x_{ref}|$ " by naming every operation, subscript and grouping explicitly.

6 — Symbols and meaning

e_{rel} : relative error; x_{calc} : calculated value; x_{ref} : reference value

7 — Pronunciation

Say subscripts, powers, Greek letters and parentheses explicitly so the relation can be reconstructed without seeing the page.

8 — Units

e_{rel} dimensionless; x_{calc} and x_{ref} in the same unit

9 — Convention

Keep the same frame, time basis, system boundary and sign convention throughout the substitution.

10 — Why this operation

Division compares one quantity with a homogeneous reference and produces a proportion indicator.

11 — Assumptions

Both quantities must belong to the same scope and time period.

12 — Unit check

e_{rel} dimensionless; x_{calc} and x_{ref} in the same unit Verify that dimensional reduction reaches the unit of the requested output.

13 — Numerical case

With $x_{calc} = 316.3$ and $x_{ref} = 316.0$, $e_{rel} = 0.3/316.0 \approx 0.000949 = 0.0949\%$.

14 — Why the calculation works

Division compares one quantity with a homogeneous reference and produces a proportion indicator.

15 — Independent check

Multiplying the result by the denominator must recover the numerator.

16 — Mental estimate

Before calculating, compare numerator and denominator mentally to predict whether the result should be below, near or above one.

17 — Interpretation

A small relative error does not guarantee that the reference value is exact or decision-relevant.

18 — What the result does not prove

The calculation proves neither that the inputs are correct nor that the model remains valid outside the declared scenario; those questions require separate evidence.

19 — Sensitivity

The result rises with the numerator and falls with the denominator, all else equal.

20 — Guided and autonomous exercises

Guided exercise. $x_{calc} = 99.2$ and $x_{ref} = 100.0$.

Detailed guided correction — open after trying

$x_{calc} = 99.2$ and $x_{ref} = 100.0$. $e_{rel} = 0.8/100 = 0.008 = 0.8\%$.

Autonomous exercise. $x_{calc} = 48.5$ and $x_{ref} = 50.0$.

Autonomous correction — open after trying

$x_{calc} = 48.5$ and $x_{ref} = 50.0$. $e_{rel} = 1.5/50 = 0.03 = 3\%$.

21 — Mission decision

Compare error with the engineering tolerance that is actually acceptable, not an arbitrary threshold.

Finite-difference derivative

$$dx_dt \approx (x_next - x_now) / dt$$

1 — Concrete question

What does “ $dx_dt \approx (x_next - x_now) / dt$ ” compute in “Finite-difference derivative”?

2 — Intuition without symbols

A numerical derivative estimates rate of change from a measured difference over a short interval.

3 — Quantities

dx_dt : estimated rate of change; x_next : next-step value; x_now : current value; dt : time interval

4 — Formula

$$dx_dt \approx (x_next - x_now) / dt$$

5 — Read aloud

Read “ $dx_dt \approx (x_next - x_now) / dt$ ” by naming every operation, subscript and grouping explicitly.

6 — Symbols and meaning

dx_dt : estimated rate of change; x_next : next-step value; x_now : current value; dt : time interval

7 — Pronunciation

Say subscripts, powers, Greek letters and parentheses explicitly so the relation can be reconstructed without seeing the page.

8 — Units

dx_dt in x-unit per time; dt in time

9 — Convention

Keep the same frame, time basis, system boundary and sign convention throughout the substitution.

10 — Why this operation

Subtraction isolates a margin or change between two quantities expressed on the same basis.

11 — Assumptions

Both terms must use the same frame, unit and system boundary.

12 — Unit check

dx_dt in x-unit per time; dt in time. Verify that dimensional reduction reaches the unit of the requested output.

13 — Numerical case

With $x_now = 1,200$ kg, $x_next = 1,110$ kg and $dt = 9$ h, $dx_dt = -90/9 = -10$ kg/h.

14 — Why the calculation works

Subtraction isolates a margin or change between two quantities expressed on the same basis.

15 — Independent check

Adding the difference back to the subtracted term must recover the initial term.

16 — Mental estimate

Comparing the two terms is enough to predict the sign and scale of the difference.

17 — Interpretation

The result depends on step size and can hide fast dynamics between samples.

18 — What the result does not prove

The calculation proves neither that the inputs are correct nor that the model remains valid outside the declared scenario; those questions require separate evidence.

19 — Sensitivity

Increasing the first term raises the result; increasing the second lowers it with unit slope.

20 — Guided and autonomous exercises

Guided exercise. $x_now = 80$, $x_next = 92$, $dt = 4$ s.

Detailed guided correction — open after trying

$x_now = 80$, $x_next = 92$, $dt = 4$ s. $dx_dt = 12/4 = 3$ per second.

Autonomous exercise. $x_now = 50$, $x_next = 44$, $dt = 3$ min.

Autonomous correction — open after trying

$x_now = 50$, $x_next = 44$, $dt = 3$ min. $dx_dt = -6/3 = -2$ per minute.

21 — Mission decision

Reduce the step or compare multiple step sizes when the decision is sensitive to the estimated derivative.

Explicit Euler step

$$x_{\text{next}} = x_{\text{now}} + dt \times f_{\text{now}}$$

1 — Concrete question

What does “ $x_{\text{next}} = x_{\text{now}} + dt \times f_{\text{now}}$ ” compute in “Explicit Euler step”?

2 — Intuition without symbols

Euler advances the state by adding the estimated change over a small step using the current slope.

3 — Quantities

x_{next} : estimated next state; x_{now} : current state; dt : time step; f_{now} : derivative evaluated at current state

4 — Formula

$$x_{\text{next}} = x_{\text{now}} + dt \times f_{\text{now}}$$

5 — Read aloud

Read “ $x_{\text{next}} = x_{\text{now}} + dt \times f_{\text{now}}$ ” by naming every operation, subscript and grouping explicitly.

6 — Symbols and meaning

x_{next} : estimated next state; x_{now} : current state; dt : time step; f_{now} : derivative evaluated at current state

7 — Pronunciation

Say subscripts, powers, Greek letters and parentheses explicitly so the relation can be reconstructed without seeing the page.

8 — Units

x_{next} and x_{now} in state units; dt in time; f_{now} in state-unit per time

9 — Convention

Keep the same frame, time basis, system boundary and sign convention throughout the substitution.

10 — Why this operation

Addition combines contributions expressed on the same basis into one coherent total.

11 — Assumptions

All contributions must use a common unit and scope.

12 — Unit check

x_{next} and x_{now} in state units; dt in time; f_{now} in state-unit per time. Verify that dimensional reduction reaches the unit of the requested output.

13 — Numerical case

With $x_{\text{now}} = 100$, $dt = 2$ s and $f_{\text{now}} = -3/\text{s}$, $x_{\text{next}} = 100 + 2 \times (-3) = 94$.

14 — Why the calculation works

Addition combines contributions expressed on the same basis into one coherent total.

15 — Independent check

Removing one contribution from the total must recover the sum of the others.

16 — Mental estimate

Adding the dominant terms first quickly gives the order of magnitude.

17 — Interpretation

Euler is simple but can become unstable or inaccurate when the step is too large for the dynamics.

18 — What the result does not prove

The calculation proves neither that the inputs are correct nor that the model remains valid outside the declared scenario; those questions require separate evidence.

19 — Sensitivity

The total changes linearly with each contribution when the others stay fixed.

20 — Guided and autonomous exercises

Guided exercise. $x_{\text{now}} = 20$, $dt = 0.5$ s, $f_{\text{now}} = 4/\text{s}$.

Detailed guided correction — open after trying

$$x_{\text{now}} = 20, dt = 0.5 \text{ s}, f_{\text{now}} = 4/\text{s}. \quad x_{\text{next}} = 20 + 0.5 \times 4 = 22.$$

Autonomous exercise. $x_{\text{now}} = 60$, $dt = 1.5$ s, $f_{\text{now}} = -2/\text{s}$.

Autonomous correction — open after trying

$x_{\text{now}} = 60$, $dt = 1.5$ s, $f_{\text{now}} = -2/\text{s}$. $x_{\text{next}} = 60 - 3 = 57$.

21 — Mission decision

Test convergence by reducing the step before using the result for a critical decision.

Trapezoidal integration over one step

$$I_{\text{step}} = 0.5 \times (y_{\text{now}} + y_{\text{next}}) \times dt$$

1 — Concrete question

What does " $I_{\text{step}} = 0.5 \times (y_{\text{now}} + y_{\text{next}}) \times dt$ " compute in "Trapezoidal integration over one step"?

2 — Intuition without symbols

The trapezoidal rule uses the average of the two endpoints to approximate area under the curve.

3 — Quantities

I_{step} : integrated contribution; y_{now} : starting value; y_{next} : ending value; dt : step duration

4 — Formula

$$I_{\text{step}} = 0.5 \times (y_{\text{now}} + y_{\text{next}}) \times dt$$

5 — Read aloud

Read " $I_{\text{step}} = 0.5 \times (y_{\text{now}} + y_{\text{next}}) \times dt$ " by naming every operation, subscript and grouping explicitly.

6 — Symbols and meaning

I_{step} : integrated contribution; y_{now} : starting value; y_{next} : ending value; dt : step duration

7 — Pronunciation

Say subscripts, powers, Greek letters and parentheses explicitly so the relation can be reconstructed without seeing the page.

8 — Units

I_{step} in y-unit times time; y in its unit; dt in time

9 — Convention

Keep the same frame, time basis, system boundary and sign convention throughout the substitution.

10 — Why this operation

Multiplication converts a rate or specific quantity into a total over the chosen interval or population.

11 — Assumptions

Factors must describe the same scenario and their units must be compatible.

12 — Unit check

I_{step} in y-unit times time; y in its unit; dt in time. Verify that dimensional reduction reaches the unit of the requested output.

13 — Numerical case

With $y_{\text{now}} = 10$ kW, $y_{\text{next}} = 14$ kW and $dt = 2$ h, $I_{\text{step}} = 0.5 \times 24 \times 2 = 24$ kWh.

14 — Why the calculation works

Multiplication converts a rate or specific quantity into a total over the chosen interval or population.

15 — Independent check

Dividing the result by any non-zero factor must recover the other factor.

16 — Mental estimate

Estimating each factor separately predicts the order of magnitude of the product.

17 — Interpretation

Accuracy depends on actual curvature between points; two endpoints do not prove a linear profile.

18 — What the result does not prove

The calculation proves neither that the inputs are correct nor that the model remains valid outside the declared scenario; those questions require separate evidence.

19 — Sensitivity

With independent factors, a relative change in one factor transfers directly to the product.

20 — Guided and autonomous exercises

Guided exercise. $y_{\text{now}} = 5$, $y_{\text{next}} = 9$, $dt = 3$ s.

Detailed guided correction — open after trying

$y_{\text{now}} = 5$, $y_{\text{next}} = 9$, $dt = 3$ s. $I_{\text{step}} = 0.5 \times 14 \times 3 = 21$ unit·s.

Autonomous exercise. $y_{\text{now}} = 20$, $y_{\text{next}} = 16$, $dt = 4$ min.

Autonomous correction — open after trying

$y_{\text{now}} = 20$, $y_{\text{next}} = 16$, $dt = 4$ min. $I_{\text{step}} = 0.5 \times 36 \times 4 = 72$ unit·min.

21 — Mission decision

Refine the mesh when the quantity changes rapidly or contains discontinuities.

Linear interpolation

$$y = y_1 + (x - x_1) \times (y_2 - y_1) / (x_2 - x_1)$$

1 — Concrete question

What does “ $y = y_1 + (x - x_1) \times (y_2 - y_1) / (x_2 - x_1)$ ” compute in “Linear interpolation”?

2 — Intuition without symbols

Between two points, linear interpolation allocates the change in proportion to the target position.

3 — Quantities

y : interpolated value; x : target position; x_1 : first abscissa; x_2 : second abscissa; y_1 : first value; y_2 : second value

4 — Formula

$$y = y_1 + (x - x_1) \times (y_2 - y_1) / (x_2 - x_1)$$

5 — Read aloud

Read “ $y = y_1 + (x - x_1) \times (y_2 - y_1) / (x_2 - x_1)$ ” by naming every operation, subscript and grouping explicitly.

6 — Symbols and meaning

y : interpolated value; x : target position; x_1 : first abscissa; x_2 : second abscissa; y_1 : first value; y_2 : second value

7 — Pronunciation

Say subscripts, powers, Greek letters and parentheses explicitly so the relation can be reconstructed without seeing the page.

8 — Units

x , x_1 and x_2 in one unit; y , y_1 and y_2 in one unit

9 — Convention

Keep the same frame, time basis, system boundary and sign convention throughout the substitution.

10 — Why this operation

The relation turns the mini-lesson mechanism into a quantitative statement that can be checked by numerical substitution.

11 — Assumptions

Inputs must come from the same scenario with units, signs and conventions explicitly aligned.

12 — Unit check

x , x_1 and x_2 in one unit; y , y_1 and y_2 in one unit. Verify that dimensional reduction reaches the unit of the requested output.

13 — Numerical case

Between $(x_1=0, y_1=10)$ and $(x_2=100, y_2=30)$, at $x=25$, $y = 10 + 25 \times 20 / 100 = 15$.

14 — Why the calculation works

The relation turns the mini-lesson mechanism into a quantitative statement that can be checked by numerical substitution.

15 — Independent check

Recomputing with the inverse relation or a second method should recover the same order of magnitude.

16 — Mental estimate

Estimating the scale before using a calculator helps catch factor, sign or unit errors.

17 — Interpretation

Interpolation is reliable only when the function is sufficiently regular between the two points.

18 — What the result does not prove

The calculation proves neither that the inputs are correct nor that the model remains valid outside the declared scenario; those questions require separate evidence.

19 — Sensitivity

Varying one input at a time reveals which variable drives the decision most strongly.

20 — Guided and autonomous exercises

Guided exercise. $x_1=0$, $x_2=10$, $y_1=50$, $y_2=70$, $x=4$.

Detailed guided correction — open after trying

$x_1=0$, $x_2=10$, $y_1=50$, $y_2=70$, $x=4$. $y = 50 + 4 \times 20/10 = 58$.

Autonomous exercise. $x_1=20$, $x_2=40$, $y_1=100$, $y_2=160$, $x=35$.

Autonomous correction — open after trying

$x_1=20$, $x_2=40$, $y_1=100$, $y_2=160$, $x=35$. $y = 100 + 15 \times 60/20 = 145$.

21 — Mission decision

Avoid unjustified extrapolation beyond the data bounds.

RSS combination of independent uncertainties

$u_{\text{total}} = \sqrt{u_1^2 + u_2^2 + u_3^2}$

1 — Concrete question

What does " $u_{\text{total}} = \sqrt{u_1^2 + u_2^2 + u_3^2}$ " compute in "RSS combination of independent uncertainties"?

2 — Intuition without symbols

Independent contributions do not systematically add in the same direction; root-sum-square reflects that structure.

3 — Quantities

u_{total} : combined uncertainty; u_1 : first contribution; u_2 : second; u_3 : third

4 — Formula

$u_{\text{total}} = \sqrt{u_1^2 + u_2^2 + u_3^2}$

5 — Read aloud

Read " $u_{\text{total}} = \sqrt{u_1^2 + u_2^2 + u_3^2}$ " by naming every operation, subscript and grouping explicitly.

6 — Symbols and meaning

u_{total} : combined uncertainty; u_1 : first contribution; u_2 : second; u_3 : third

7 — Pronunciation

Say subscripts, powers, Greek letters and parentheses explicitly so the relation can be reconstructed without seeing the page.

8 — Units

all uncertainty terms in the same unit

9 — Convention

Keep the same frame, time basis, system boundary and sign convention throughout the substitution.

10 — Why this operation

Root-sum-square combines independent contributions without assuming they all add in the same direction.

11 — Assumptions

Contributions must be independent or sufficiently uncorrelated for this combination to be justified.

12 — Unit check

all uncertainty terms in the same unit Verify that dimensional reduction reaches the unit of the requested output.

13 — Numerical case

With $u_1 = 0.20$ mm, $u_2 = 0.15$ mm and $u_3 = 0.10$ mm, $u_{\text{total}} = \sqrt{0.04+0.0225+0.01} \approx 0.2693$ mm.

14 — Why the calculation works

Root-sum-square combines independent contributions without assuming they all add in the same direction.

15 — Independent check

The square of the result must equal the sum of the squared contributions.

16 — Mental estimate

The result must be at least as large as the dominant contribution and below their arithmetic sum.

17 — Interpretation

If contributions are correlated, covariance terms must be included instead of this simple form.

18 — What the result does not prove

The calculation proves neither that the inputs are correct nor that the model remains valid outside the declared scenario; those questions require separate evidence.

19 — Sensitivity

The largest contribution influences the result more strongly because it is squared.

20 — Guided and autonomous exercises

Guided exercise. $u_1=3$, $u_2=4$, $u_3=0$.

Detailed guided correction — open after trying

$u_1=3$, $u_2=4$, $u_3=0$. $u_{\text{total}} = \sqrt{9+16} = 5$.

Autonomous exercise. $u_1=1$, $u_2=2$, $u_3=2$.

Autonomous correction — open after trying

$u_1=1$, $u_2=2$, $u_3=2$. $u_{\text{total}} = \sqrt{1+4+4} = 3$.

21 — Mission decision

Document independence and correlations before crediting RSS reduction.

1. Vectors and matrices: organize several quantities at once

A vector groups components that belong to one state: position x , y , z ; velocity v_x , v_y , v_z ; or sensor errors. A matrix describes how several quantities combine. In space problems it can change reference frames, propagate covariance or linearize a system. Dimensional compatibility comes first: a 3×3 matrix can only multiply a compatible object. Writing dimensions before computing prevents many mistakes.

Engineering habit. For “vectors and matrices: organize several quantities at once”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

2. Derivatives and gradients: measure the rate of change

A derivative answers the question ‘how much does the output change when the input changes slightly?’ Velocity is the derivative of position, acceleration the derivative of velocity, and a gradient collects several partial derivatives. In design these ideas become sensitivity analysis: if mass rises by one percent, which mission variable reacts most strongly? A gradient helps rank parameters instead of changing everything blindly.

Engineering habit. For “derivatives and gradients: measure the rate of change”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

3. Differential equations: represent systems that evolve

Many physical laws connect a quantity to its derivative. The equation $dv/dt=a$ says velocity evolves according to acceleration. For a tank, dm/dt can represent an outgoing mass flow; for a battery, dE/dt connects stored energy, generation and demand. A differential equation is therefore an evolution rule, not an abstract decoration. Initial and boundary conditions are as important as the equation itself.

Engineering habit. For “differential equations: represent systems that evolve”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

4. Numerical integration: move in small steps without fooling yourself

When an exact solution does not exist or is impractical, numerical integration advances time step by step. Euler is simple but can accumulate substantial error; Runge–Kutta samples several slopes during a step and generally improves accuracy. Time step is an engineering choice: too large hides dynamics, too small wastes computation. Convergence is checked by repeating the calculation with a finer step.

Engineering habit. For “numerical integration: move in small steps without fooling yourself”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

5. Interpolation, fitting and imperfect data

Measurements do not always arrive exactly when a model needs them. Interpolation estimates a value between samples. Fitting searches for model parameters that best explain a set of observations. Interpolation must be distinguished from extrapolation: predicting outside the measured range is much riskier. A smooth curve is never proof that the model is correct.

Engineering habit. For “interpolation, fitting and imperfect data”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

6. Probability, covariance and uncertainty propagation

A mission does not have a perfectly known position or an exactly constant consumption rate. Uncertainty is therefore represented using quantities such as standard deviations and covariances. Positive covariance means two errors tend to move together; negative covariance means they tend to move in opposite directions. Propagation asks how input uncertainty transforms into output dispersion.

Engineering habit. For “probability, covariance and uncertainty propagation”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

Approfondissement — Covariance: treat uncertainty as part of the state, not a footnote

A position estimate is not merely three coordinates. It should also carry uncertainty and the correlations among variables. These are commonly organized in a covariance matrix P . The symbol P denotes the matrix; diagonal elements contain variances, which are squared standard deviations, while off-diagonal elements describe how errors in two variables vary together. If position error along x tends to grow with velocity error along x , treating them as independent creates false confidence.

Suppose the standard deviation in one position coordinate is $\sigma_x = 120$ m, where σ_x means the standard deviation of x . A naïve statement of ± 120 m still does not describe a two-dimensional landing ellipse. The ellipse also depends on the y uncertainty, x – y covariance and the chosen confidence level. During descent the covariance is propagated: sensors reduce some uncertainties, models add others and maneuvers rotate the geometry. Precision without an interval and assumptions is therefore incomplete information.

7. Monte Carlo: repeat the scenario to see a distribution

A Monte Carlo simulation randomly samples inputs from defined distributions and runs the model many times. The result is a distribution rather than one answer: median, percentiles and tails. This is powerful for nonlinear systems, but it cannot rescue a bad physical model. Unrealistic input distributions simply produce statistically precise nonsense.

Engineering habit. For “monte carlo: repeat the scenario to see a distribution”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

Approfondissement — Mission Monte Carlo: replace one nominal case with a distribution of outcomes

A mission flies once, but its model can fly thousands of times. In Monte Carlo analysis, uncertain inputs are sampled from distributions: mass, atmospheric density, wind, sensor bias, thrust, software timing and initial state. Each simulation produces an outcome such as landing point, propellant remaining, peak temperature or battery margin. After N trials, where N is the total number of simulations, the engineer sees a distribution rather than one answer and can count constraint violations.

If 18 of 20,000 landing simulations fall outside the safe region, the empirical frequency is $18 \div 20,000 = 0.0009$, or 0.09%. That does not prove the real risk is exactly 0.09%. It depends on input distributions, correlations and model fidelity. The useful step is to inspect the failure cases, identify dominant variables and rerun after mitigation. Monte Carlo is not an exercise in producing a smooth histogram; it is a method for discovering which mechanisms create rare failures and whether engineering margin can eliminate or only reduce them.

8. Numerical conditioning and plausibility checks

A calculation can be mathematically defined yet numerically fragile. Subtracting nearly equal numbers or inverting an ill-conditioned matrix can amplify rounding errors. Engineers monitor orders of magnitude, units, residuals and stability. A durable rule remains: before trusting many decimal places, estimate the answer by hand and ask whether its sign and scale make physical sense.

Engineering habit. For “numerical conditioning and plausibility checks”, write the inputs, outputs, units and validity range first. Then build one nominal and one degraded case. This two-case approach prevents a correct equation from being mistaken for an operationally robust Mars architecture.

Approfondissement — Conditioning, scale and numbers that look more precise than the data

A numerical computation can be wrong even when every arithmetic operation is correct. The danger appears when a problem is ill-conditioned: a small change in the inputs produces a large change in the answer. Imagine computing a small displacement by subtracting two nearly equal large positions. If those positions are rounded, common significant digits cancel and the relative error in the displacement can become enormous. The first habit is therefore to inspect orders of magnitude before trusting a solver: which variables are around 10^6 , which are around 10^{-3} , and which outputs are differences between almost equal values?

Sensitivity is often summarized by a condition number κ , the Greek letter kappa. Here κ measures how strongly relative input error can be amplified in the output. A κ near 1 describes a well-conditioned problem; a very large κ warns that tiny data errors may dominate the result. The number does not replace physics. It warns that carrying more digits in software may not rescue poor information. Navigation, orbit determination and parameter estimation all require the engineer to ask not only “what is the value?” but “how many digits are actually supported by the measurement process?”

Worked example step by step

The work method is always the same: state what every symbol represents, convert all units into a coherent system, perform the operation, then translate the result into a sentence. Finally perform an order-of-magnitude check. If the answer changes by a factor of one thousand because millimetres were treated as metres, the conversion must be visible in the calculation.

Progressive exercise

1. Choose a simple case and list every input with units.
2. Compute the nominal result without margin.
3. Vary the most uncertain parameter by $\pm 20\%$ and compare.
4. Inject one credible failure and explain which indicator detects it.
5. Decide whether the system continues, degrades or stops.

Reasoned solution

A good solution is not only the final number. It shows conversions, why the equation applies, sensitivity and the resulting decision. If different plausible assumptions lead to the same operational decision, the design is relatively robust to that uncertainty. If a small variation reverses the decision, the parameter becomes a priority for measurement or margin.

Validation mini-project

Build a two-to-four-page engineering note applying this course to one Mars subsystem. Include need, assumptions, functional sketch, hand calculation, second calculation or simulation, uncertainties, injected failure, decision criteria and three primary references. The goal is a chain of evidence that another reader can reproduce.

Common errors to detect

- mixing units or frames without explicit conversion;
- presenting calculated values as measured data;
- ignoring a model's validity range;
- confusing numerical precision with physical accuracy;
- sizing only the nominal case with no margin or degraded mode.

Mission reasoning laboratory — connect the calculation to a real decision

The purpose of this laboratory is not to memorise isolated definitions. It links the module to a Mars mission decision, identifies evidence, tests boundary cases and preserves an explicit recovery path.

Define the mathematical object before calculating

Many numerical failures start before the computer is involved. A vector needs a coordinate frame; a matrix needs an ordering of states; a derivative needs the variable with respect to which change is measured; a probability distribution needs a defined random quantity. Writing these meanings beside the symbols prevents correct algebra from answering the wrong physical question. In navigation, for example, subtracting two position vectors expressed in different frames can produce numbers with the correct unit of metres yet no valid geometric meaning.

Use refinement as evidence

A numerical result becomes more credible when it behaves predictably under refinement. Reduce the time step, tighten the solver tolerance or increase the number of samples and examine a mission-relevant output. The goal is not that every printed digit stays identical; it is that the result approaches a stable range at a rate consistent with the method. If halving a time step shifts predicted landing position by kilometres, the earlier result is not ready for operational use no matter how polished its plot looks.

Separate algorithm error from model error

A highly accurate numerical solution can still solve the wrong physical model. Integrating an ideal two-body orbit to machine precision does not capture atmospheric drag, third-body gravity or thrust errors if the mission needs them. Validation

therefore has two layers: first show that the algorithm solves the equations accurately enough; then show that the equations and parameters represent reality well enough for the decision. Mixing these layers can lead teams to “improve” step size while ignoring the dominant physical uncertainty.

Respect conditioning and observability

Some problems amplify tiny input errors because the geometry itself provides weak information. Navigation with nearly parallel lines of sight is a classic example. The numerical solver may be perfectly stable, yet the state estimate remains uncertain because many solutions fit the measurements almost equally well. Perturb observations inside realistic noise bounds and examine the output spread. If small measurement changes create huge state changes, adding more decimal precision will not create missing information; a new measurement geometry or sensor may be required.

Treat Monte Carlo as a distribution, not a magic number

Monte Carlo analysis samples uncertain inputs repeatedly and reveals a distribution of outcomes. Its value depends on the input distributions, correlations, model fidelity and number of samples. Reporting only the mean hides tails that may drive mission risk. Plot percentiles, threshold exceedance and failure modes. Also check whether rare but physically possible combinations were excluded by convenient assumptions. A thousand runs of an unrealistic uncertainty model can give false confidence more efficiently than one careful engineering calculation.

Use interpolation inside evidence, extrapolation with caution

Interpolation estimates between known data points and is often reasonably controlled when the underlying function is smooth and sampling is adequate. Extrapolation projects beyond the data and can fail abruptly when regime changes occur. A thermal-property table measured between 250 and 320 K should not be extended casually to 500 K. The numerical method may return a neat value while the material physics has changed. Mark extrapolated regions explicitly and seek a model or data source valid in the new regime.

Keep a numerical credibility record

For every calculation that influences a mission decision, record the governing equations, units, solver, step or tolerance, reference case, sensitivity checks and known numerical limitations. This does not need to become bureaucratic documentation for every quick estimate; it does need to exist for safety-significant results. The record allows another engineer to reproduce the result, distinguish numerical choices from physical assumptions and understand why the chosen precision was sufficient rather than merely convenient.

Progressive exercises — solve first, then open the correction

Synthesis exercise — numerical credibility

A thermal model predicts a critical component temperature of 319.0 K using a 60 s time step, 316.8 K using 20 s and 316.3 K using 10 s. A separate high-fidelity reference for the same case is 316.0 K. Calculate the relative error of the 10 s result, describe the convergence pattern, and decide whether a further step-size check is justified before accepting the model.

Detailed correction — Synthesis exercise — numerical credibility

Relative error. $|316.3 - 316.0| / 316.0 \approx 0.000949$, or about 0.095%.

Convergence. The prediction moves toward a stable value as the step is reduced: the 20 → 10 s change is only 0.5 K compared with 2.2 K for 60 → 20 s.

Decision. The trend is encouraging but one additional refinement or an error-estimation method is justified when the acceptance limit is close to the predicted temperature.

Beginner vocabulary checkpoint

Focus or tap each highlighted term for its definition. The goal is to remove hidden prerequisites before the learner reaches a safety-significant calculation or procedure.

- vector — Ordered set of components representing a quantity such as position, velocity or force in a defined coordinate system.
- matrix — Rectangular array of numbers used to represent linear transformations, coupled equations or uncertainty relationships.
- derivative — Rate at which one quantity changes with respect to another, often time.
- gradient — Vector of partial derivatives that points toward the direction of greatest increase of a scalar field.
- differential equation — Equation relating a quantity to one or more of its derivatives.
- integration — Accumulation process that reconstructs a quantity from its rate of change over an interval.
- Euler method — First-order numerical integration method that advances a state using the derivative at the start of each step.
- Runge–Kutta method — Family of numerical integration methods using multiple derivative evaluations per step for improved accuracy.
- interpolation — Estimation of a value between known data points.
- extrapolation — Estimation beyond the range of known data; usually riskier than interpolation.
- residual — Difference between observed or required behaviour and the behaviour predicted by a model or current estimate.
- tolerance — Numerical threshold used to decide whether an iterative or verification criterion is sufficiently satisfied.
- iteration — One repeated update in a numerical method.
- convergence rate — Speed at which numerical error decreases as iterations or discretisation refinements proceed.
- round-off error — Error produced by finite numeric precision in computer arithmetic.
- truncation error — Error produced when an infinite or continuous mathematical process is approximated by a finite operation.
- floating-point — Computer representation of real numbers using a finite mantissa and exponent.
- condition number — Measure of how sensitive a mathematical problem is to small changes in its input.
- singular matrix — Matrix that has no ordinary inverse because its rows or columns do not provide independent information.
- covariance — Matrix or quantity describing how uncertainties vary and co-vary among estimated variables.
- Monte Carlo — Method that samples many uncertain cases to build a distribution of possible outcomes.
- random sample — One draw from a defined probability distribution used in statistical simulation.
- reference solution — Higher-confidence solution used to evaluate the accuracy of another numerical result.
- absolute error — Magnitude of the difference between a computed value and its reference.
- relative error — Absolute error scaled by the magnitude of a reference value.
- step-size sensitivity — Change in numerical output caused by changing the integration or discretisation step.

Decision closeout — evidence before acceptance

Acceptance thresholds belong to the decision

Numerical precision should be chosen from the engineering decision, not from the number of decimals a solver can print. If a landing-site exclusion boundary is kilometres wide, sub-millimetre integration precision has no operational value. If a control law changes mode at a narrow threshold, numerical error near that threshold may matter greatly. State the acceptable error on the mission quantity first, then choose solver tolerances and step sizes that demonstrate margin relative to that requirement. This reverses the common habit of choosing a convenient algorithm setting and only afterward asking whether the result was accurate enough.

Compare independent methods when consequences are high

When a result drives a safety-significant decision, use a second method that fails differently. A trajectory can be checked with a simpler analytic approximation, a thermal integration with an energy balance, or a least-squares estimate with a geometry-based uncertainty check. Agreement between methods does not prove truth, but disagreement reveals hidden assumptions quickly. The strongest independent check is often deliberately lower fidelity yet physically transparent, because it is less likely to share the same coding or discretisation defect as the primary model.

Preserve units and scales inside software interfaces

Many numerical failures are interface failures: kilometres passed where metres were expected, degrees treated as radians, seconds mixed with milliseconds, or state components ordered differently between routines. The arithmetic can remain internally consistent while the physical answer is wrong by orders of magnitude. Use explicit unit conversion at boundaries, automated dimensional tests where practical, and small hand-computed cases whose answer is known. For Mars operations, a unit mistake in a stored parameter can propagate through navigation, power or life-support calculations long after the original code was reviewed.

Primary sources and pathways

- NASA — Guidance, Navigation & Control state of the art
- NASA Science — Basics of Space Flight: Navigation
- www.nasa.gov — institutional documentation
- Space Academy — full pathway

FR ↔ EN parity matrix

EN	FR
Relative error	Erreur relative
Finite-difference derivative	Dérivée par différence finie
Explicit Euler step	Pas d'Euler explicite
Trapezoidal integration over one step	Intégration trapézoïdale sur un pas
Linear interpolation	Interpolation linéaire
RSS combination of independent uncertainties	Combinaison RSS d'incertitudes indépendantes