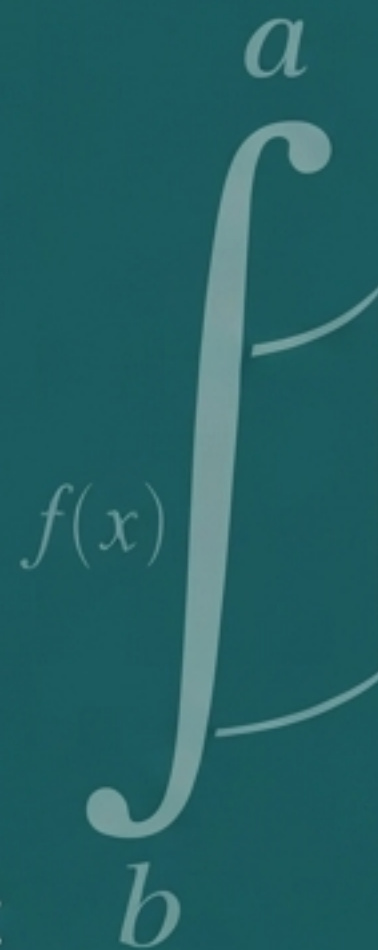
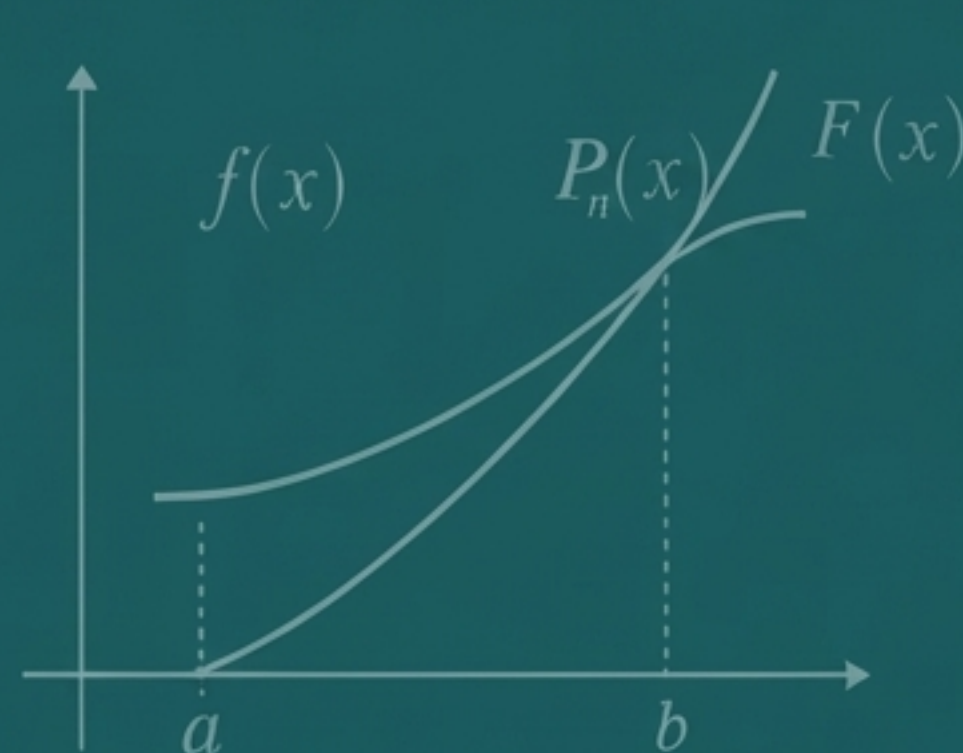


Numerické Metódy: Od Teórie k Letu

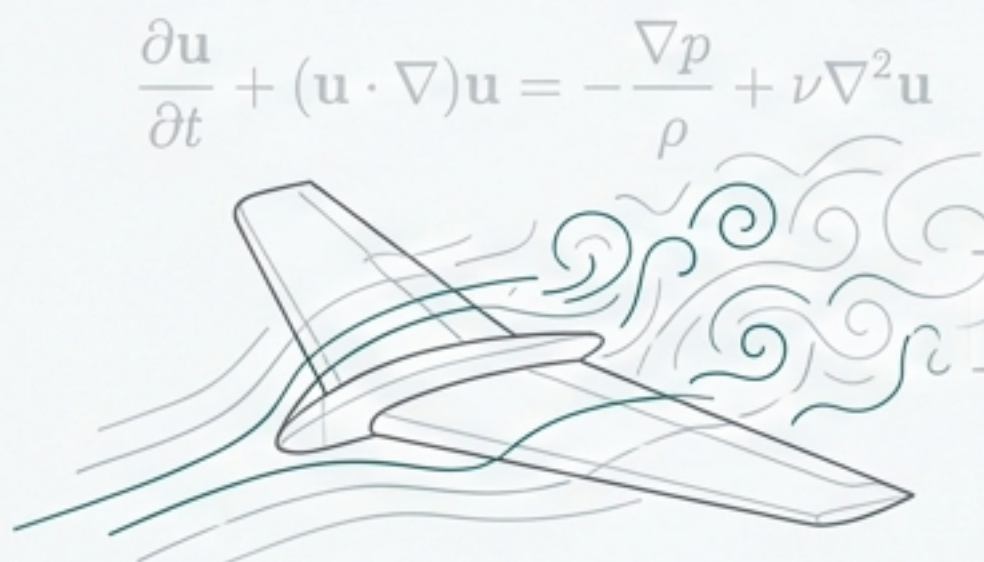
Prehľad kľúčových konceptov a aplikácií v letectve



doc. RNDr. Peter Szabó, PhD. | Author type:et typeset
Kurz: Numerická matematika v letectve matematika v letectve

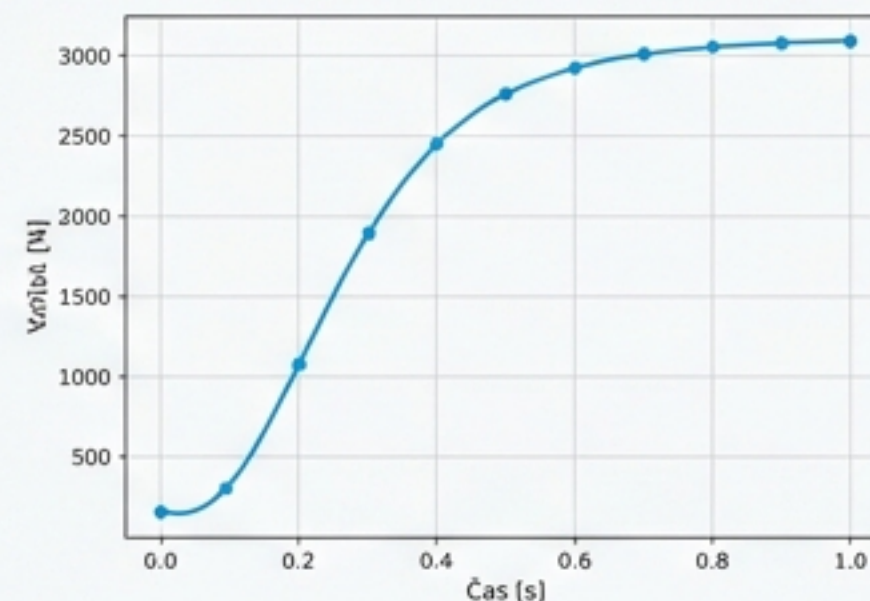
Prečo nestačí analytické riešenie? Inžinierska realita je komplexná.

Komplexný Fyzikálny Systém



Numerické Metódy

Aproximácia a Simulácia



Mnohé fyzikálne systémy v leteectve – od aerodynamiky po riadenie letu – sú opísané rovnicami, ktoré nemajú jednoduché, „ceruzkou na papieri“ riešiteľné vzorce.

Analytické riešenia sú ideálne, ale zriedkavé. Numerické metódy nám umožňujú nájsť vysoko presné **aproximácie** a **simulovať** správanie systémov, ktoré by sme inak nedokázali analyzovať.

Cieľom je premeniť zložité diferenciálne rovnice na sériu aritmetických operácií, ktoré dokáže vykonať počítač.

Naše výpočtové laboratórium: Dva nástroje, dva prístupy.

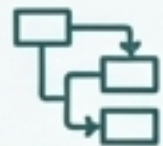


Komerčný štandard v priemysle. Optimalizovaný pre numerické výpočty, maticové operácie a simulácie.

Silné stránky:



Vysoký výkon pre numerické úlohy (2-4x rýchlejší v testoch sčítania radu).



Simulink: Grafické prostredie pre modelovanie dynamických systémov.



Rozsiahle, profesionálne toolboxy (Aerospace, Control Systems, atď.).



Open-source systém postavený na Pythone. Zameraný na symbolickú matematiku, algebru a teóriu čísel.

Silné stránky:



Výnimočné schopnosti v symbolických výpočtoch (derivácie, integrály).



Flexibilita vďaka ekosystému Pythonu.



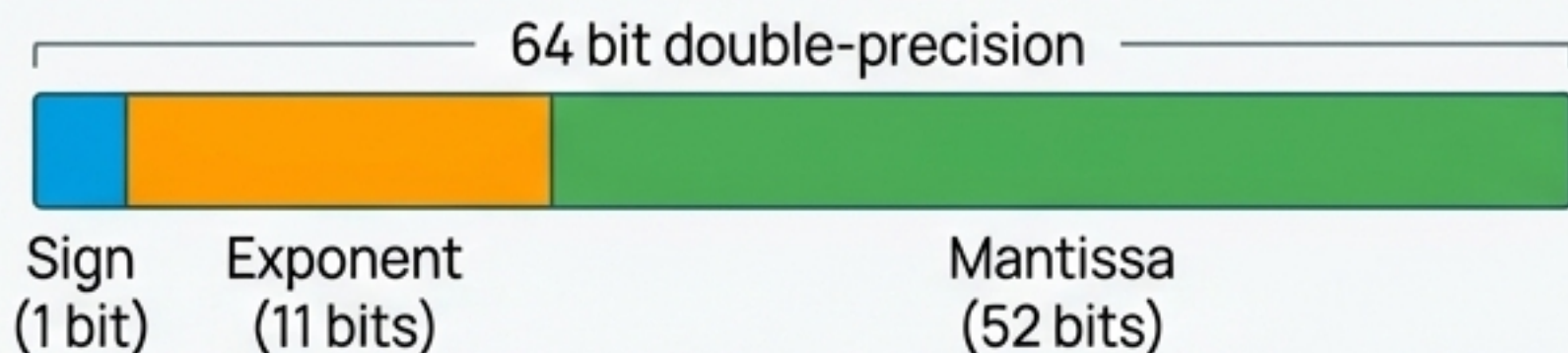
Zadarmo, ideálny pre akademickú prácu a výučbu.

Oba nástroje sú dostupné vo virtuálnom laboratóriu TUKE (Ubuntu 24.04 v rámci WSL na Windows Server 2025).

Základ výpočtov: Konečná presnosť a nevyhnutné chyby.

Reprezentácia čísel (IEEE 754)

Počítače neukladajú reálne čísla presne. Používajú formát s plávajúcou desatinnou čiarkou (mantisa + exponent).



Dôsledok*: $0.1 + 0.1 + 0.1 \neq 0.3$.

V skutočnosti sa výsledok môže rovnať
`0.30000000000000004`.

Strojová presnosť (Epsilon)

Najmenšie číslo ϵ , pre ktoré platí $1 + \epsilon > 1$.

Pre 64-bitovú presnosť je $\epsilon \approx 2.22 \cdot 10^{-16}$.

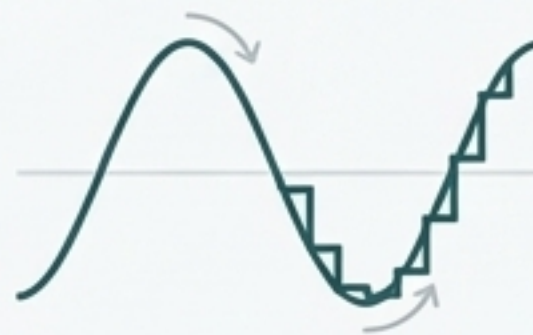
Akkoľvek menšia zmena sa stratí.



Typy chýb

Zaokrúhľovacie chyby

Vznikajú z konečnej reprezentácie čísel.



Aproximačné (Truncation) chyby

Vznikajú nahradením presného matematického vzťahu jeho aproximáciou (napr. použitím konečného počtu členov Taylorovho radu).

$$\sum_{\infty} \dots \text{✂} \dots \sum_n$$

Univerzálny nástroj aproximácie: Taylorov polynóm.

Každú dostatočne hladkú funkciu $f(x)$ môžeme v okolí bodu a aproximovať polynómom. Čím viac členov (derivácií) použijeme, tým je aproximácia presnejšia.

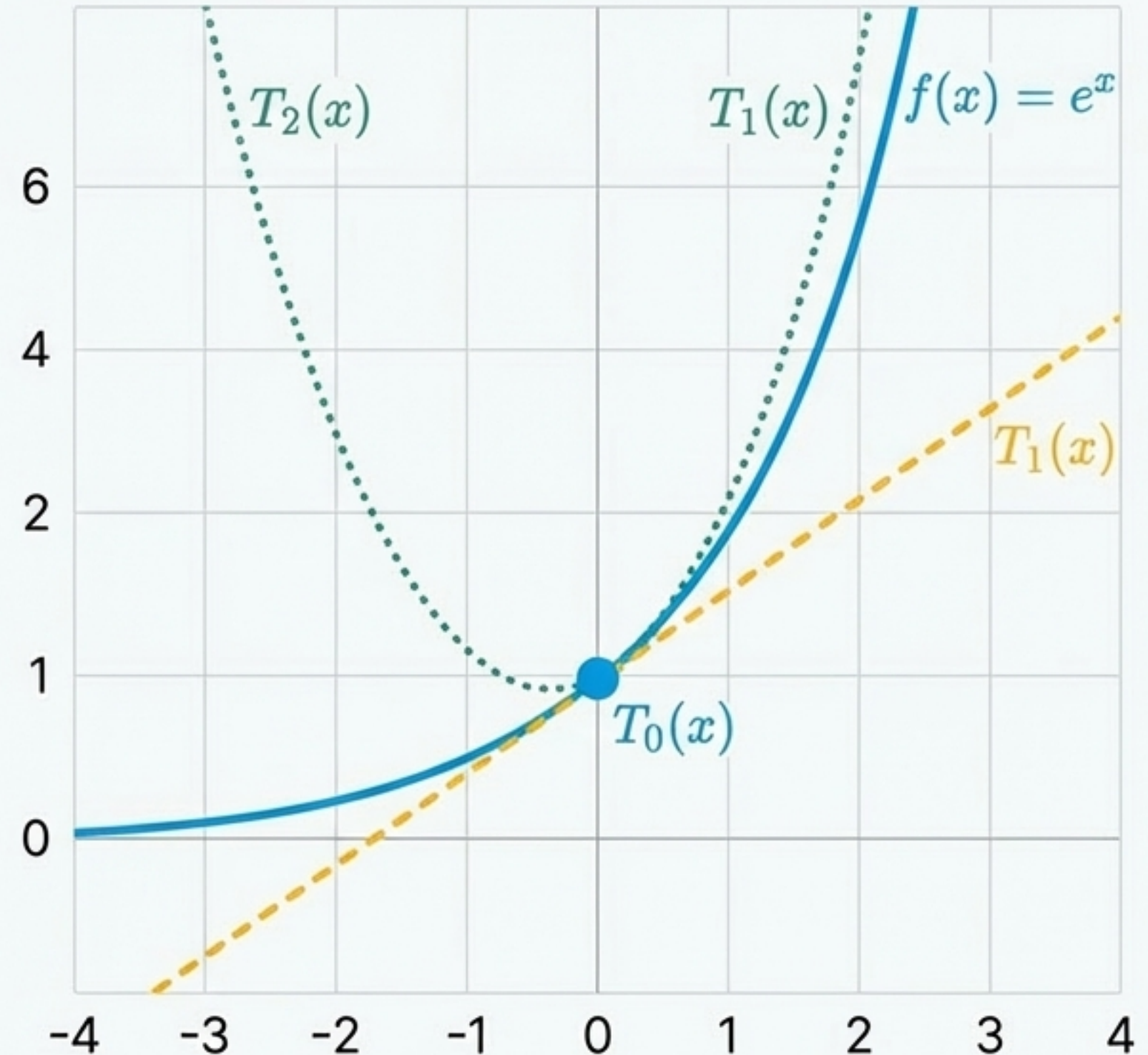
The Formula

$$T_n(x) = f(a) + f'(a)(x - a) + \frac{f''(a)}{2!}(x - a)^2 + \dots + \frac{f^{(n)}(a)}{n!}(x - a)^n$$

Chyba aproximácie

Zvyškový člen $R_n(x)$ nám hovorí, akej chyby sa dopúšťame. Jeho veľkosť je úmerná prvej zanedbanej derivácii. Zapisujeme to pomocou Landauovej notácie:

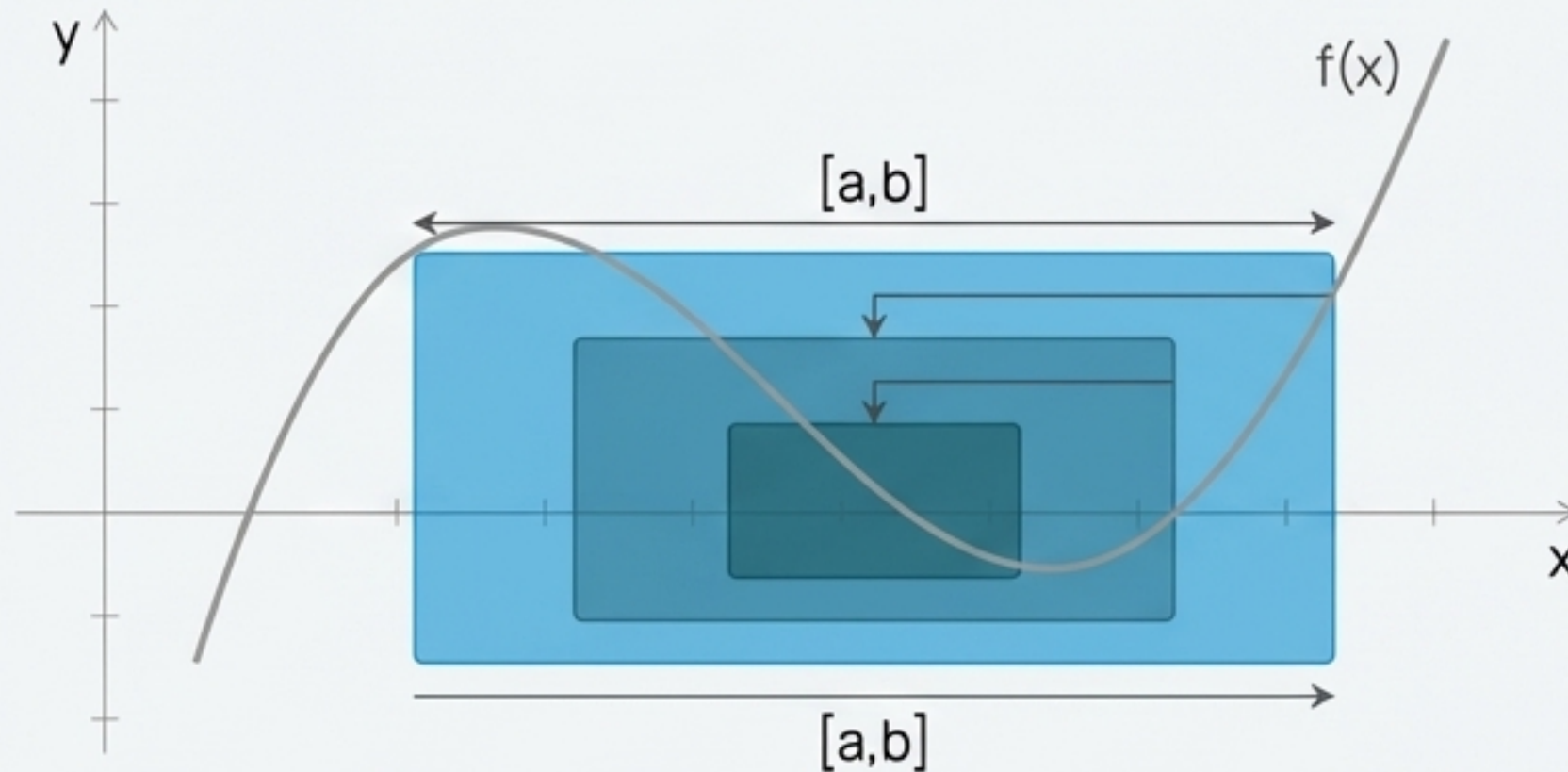
$$R_n(x) = O((x - a)^{n+1}).$$



Hľadanie rovnovážnych stavov: Riešenie nelineárnych rovníc $f(x) = 0$.

Problem Statement: Kedy je systém v rovnováhe? Kedy sa sila rovná nule? Tieto otázky vedú k hľadaniu koreňov funkcií.

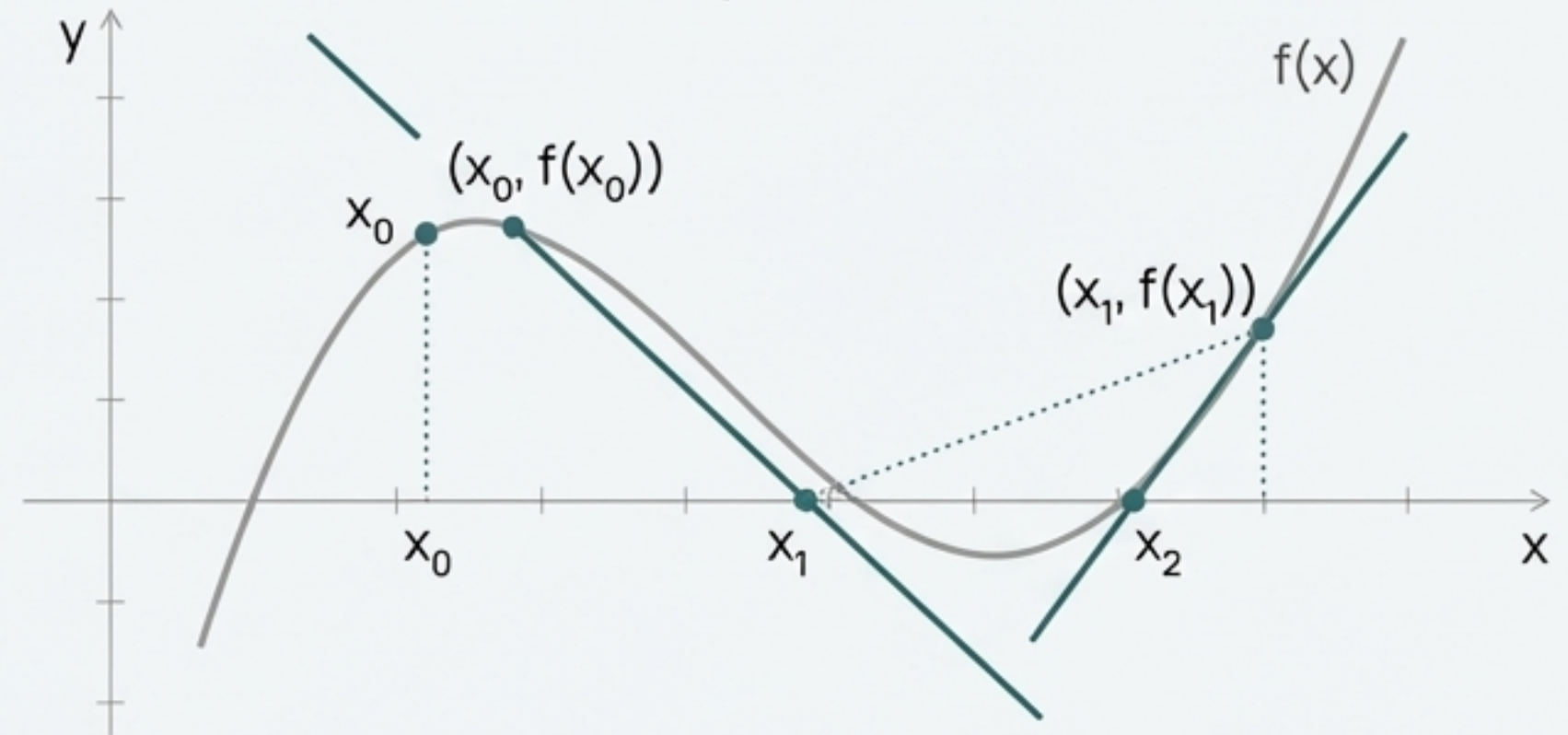
Bisekčná metóda



Bisekčná metóda (Spoločlivá, ale pomalá)

- **Princíp:** Opakované delenie intervalu, v ktorom je zaručený koreň, na polovicu.
- **Konvergencia:** Lineárna, $O(2^k)$. Pomalá, ale istá.

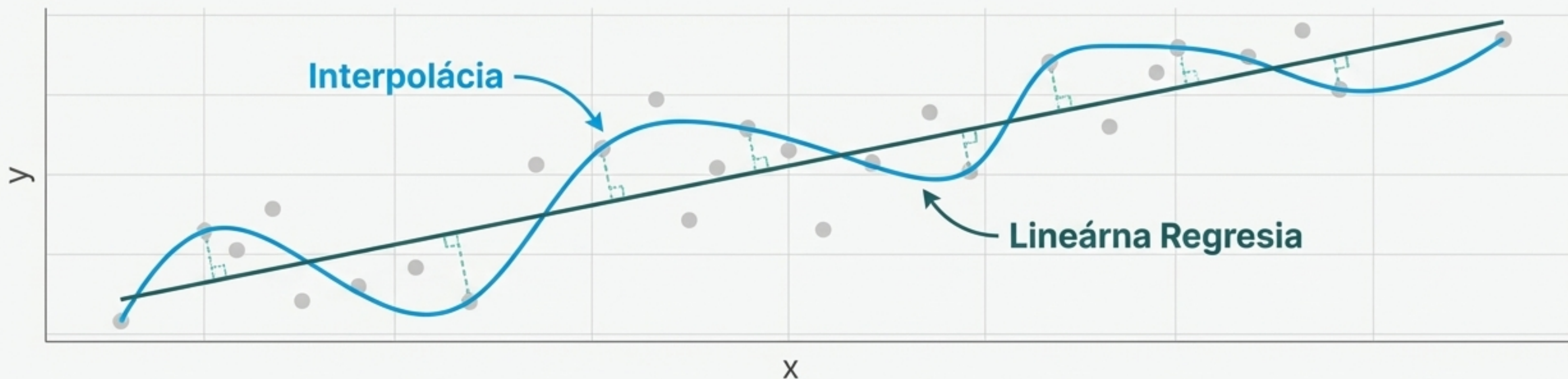
Newton-Raphsonova metóda



Newton-Raphsonova metóda (Rýchla, ale citlivá)

- **Princíp:** Iteračné spresňovanie odhadu pomocou dotýčnice ku grafu funkcie.
- **Vzorec:** $x_{k+1} = x_k - f(x_k) / f'(x_k)$
- **Konvergencia:** Kvadratická, $O(e_k^2)$. Extrémne rýchla, ak je počiatočný odhad dobrý. Môže zlyhať, ak $f'(x_k) \approx 0$ alebo je odhad ďaleko.

Práca s dátami: Prekladanie bodov vs. hľadanie trendu.



Interpolácia (Presne cez body)

- 🔍 **Otázka:** Ak poznáme hodnoty v niekoľkých bodoch, aká je hodnota medzi nimi?
- 🎯 **Cieľ:** Nájsť funkciu (typicky polynóm), ktorá prechádza **všetkými** zadanými bodmi.
- ⚙️ **Metóda:** Lagrangeov interpolačný polynóm.
- 📈 **Chyba:** $O(h^{n+1})$. Presnosť rastie s počtom bodov (do istej miery).

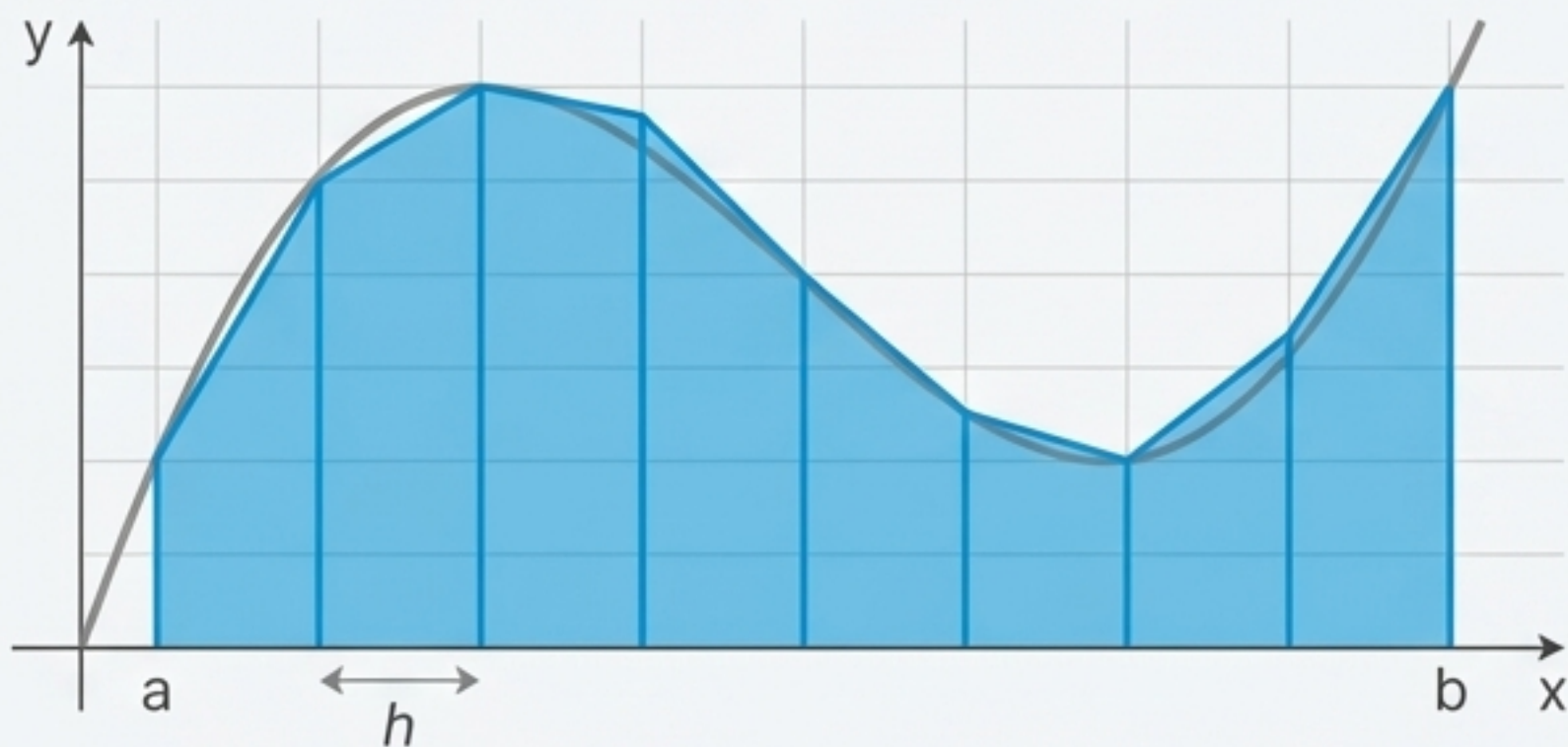
Regresia (Aproximácia trendu)

- 🔍 **Otázka:** Aké sú dáta zaťažené šumom, aký je základný trend, ktorý ich opisuje?
- 🎯 **Cieľ:** Nájsť funkciu (priamku, parabolu), ktorá **najlepšie vystihuje** dáta, ale nemusí prechádzať žiadnym bodom.
- ⚙️ **Metóda:** Metóda najmenších štvorcov – minimalizuje sumu štvorcov odchýlok $\sum (y_i - f(x_i))^2$.

Výpočet plochy pod krivkou: Numerická integrácia.

Problém: Ako vypočítať určitý integrál $\int f(x) dx$, keď nepoznáme analytické riešenie? Rozdelíme plochu na malé, jednoduché tvary.

Lichobežníková metóda



Idea: Aproximujeme funkciu na malých intervaloch úsečkami.

Presnosť: Globálna chyba je $O(h^2)$. Zmenšenie kroku h na polovicu zníži chybu štvornásobne.

Simpsonova metóda

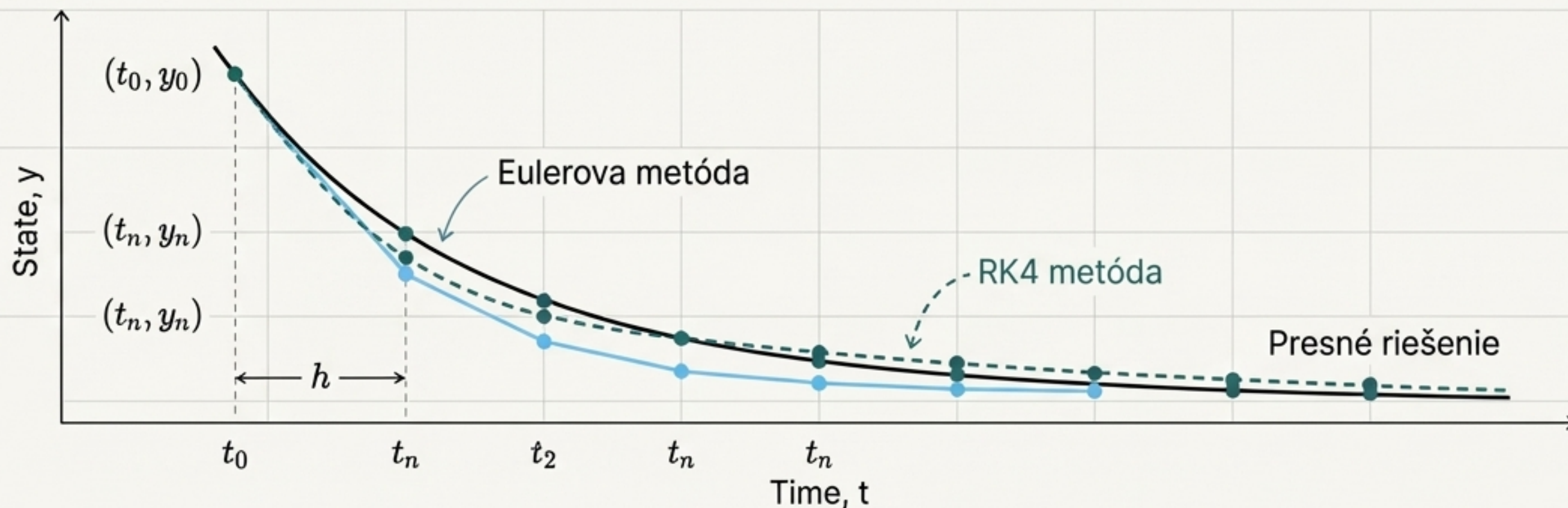


Idea: Aproximujeme funkciu na malých intervaloch parabolami (kvadratická interpolácia).

Presnosť: Globálna chyba je $O(h^4)$. Zmenšenie kroku h na polovicu zníži chybu až **šestnásobne**.

Modelovanie dynamiky v čase: Riešenie diferenciálnych rovníc.

Problém: Ako predpovedať budúci stav systému (napr. polohu lietadla), ak poznáme jeho súčasný stav a zákony jeho pohybu? Riešime rovnicu $y' = f(t, y)$.



Eulerova metóda (Jednoduchá, ale nestabilná)

☰ **Princíp:** Robíme malé kroky vpred v smere dotyčnice (derivácie).

✎ **Vzorec:** $y_{n+1} = y_n + h \cdot f(t_n, y_n)$

⌞ **Presnosť:** Nízka, globálna chyba je $O(h)$. Vyžaduje extrémne malé kroky.

Runge-Kutta 4. rádu (RK4 – Zlatý štandard)

☰ **Princíp:** V každom kroku vypočíta štyri pomocné odhady smeru a ich váženým priemerom dosiahne oveľa vyššiu presnosť.

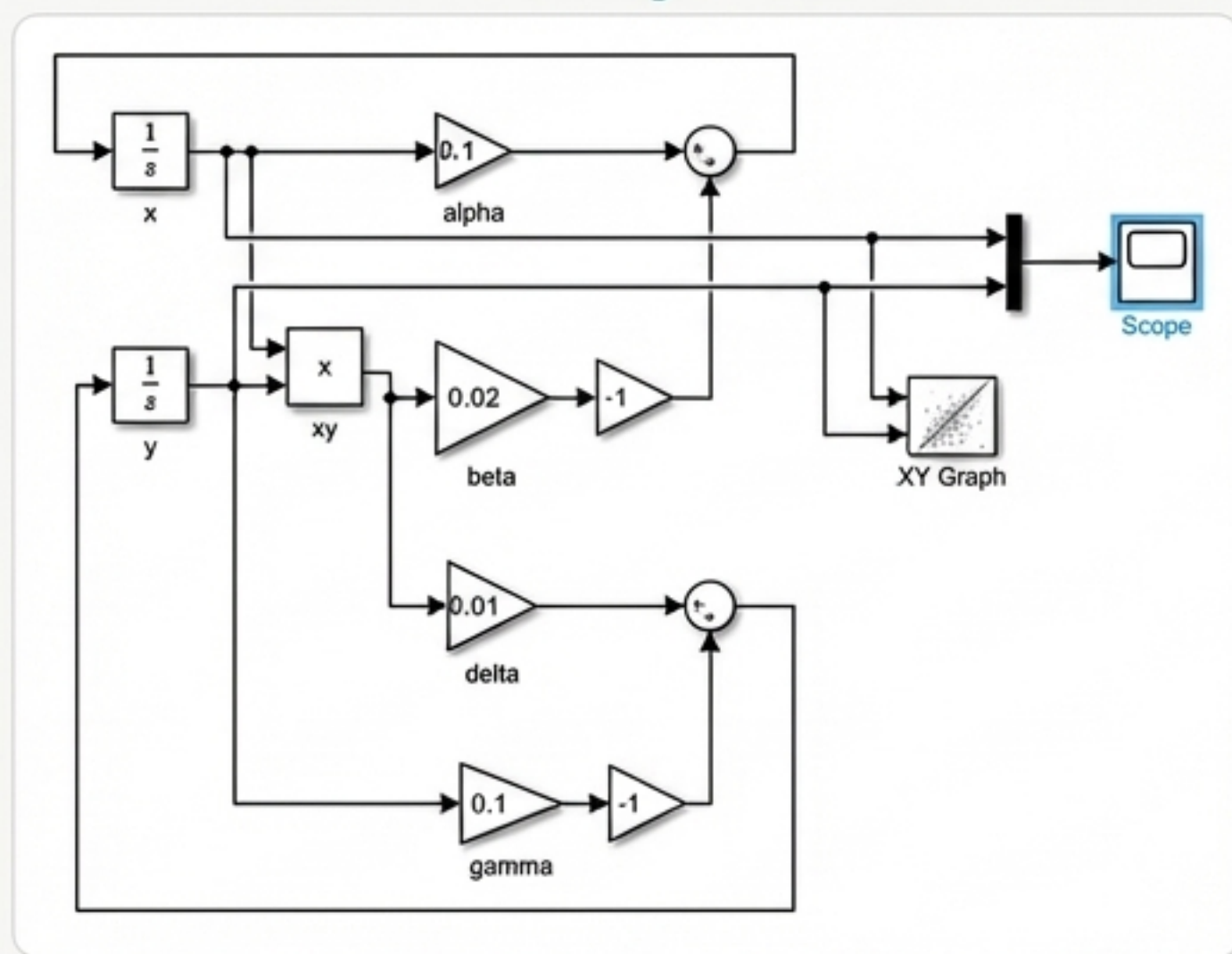
⌞ **Presnosť:** Vysoká, globálna chyba je $O(h^4)$. Štandardná metóda v Simulinku (ode45) a vedeckých výpočtoch.

Od rovníc k simulácii: Model predátor-korist' v Simulinku.

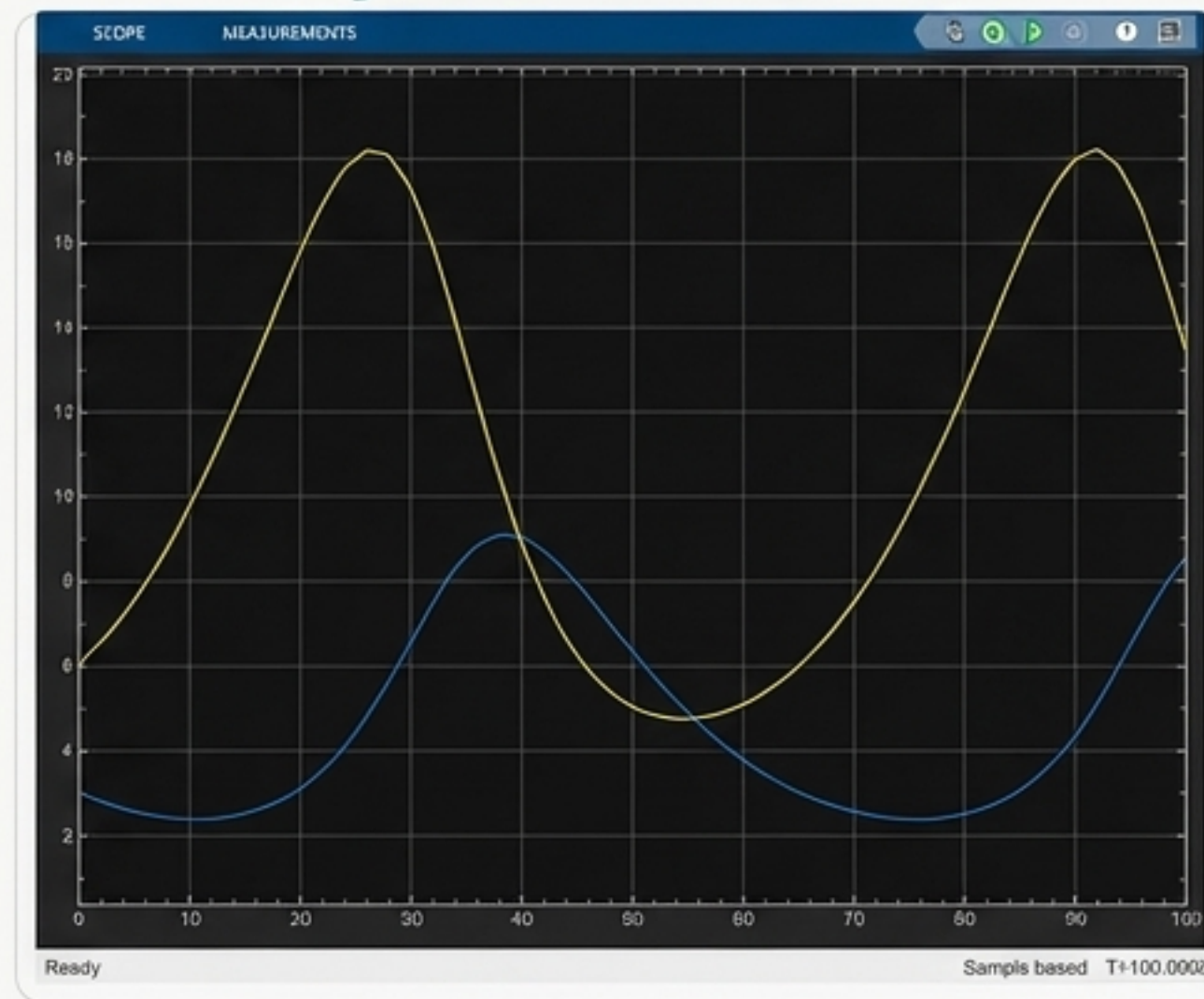
Kontext: Diferenciálne rovnice sú jazykom modelovania. Simulink nám umožňuje tento jazyk 'nakresliť' a spustiť.

Model (Lotka-Volterra): $\frac{dx}{dt} = \alpha x - \beta xy$ (Rast koristi, brzdený predátormi),
 $\frac{dy}{dt} = \delta xy - \gamma y$ (Rast predátorov, závislý od koristi).

Simulačný model



Výsledok simulácie



Pointa: Komplexné dynamické správanie vyplýva priamo z jednoduchých pravidiel (rovníc), ktoré sme numericky vyriešili.

Zjednodušenie dynamiky: Sila Laplaceovej transformácie



Kľúčová myšlienka

Laplaceova transformácia premieňa operácie z časovej oblasti (derivácie, integrály) na jednoduché algebraické operácie v "s-oblasti" (násobenie, delenie).



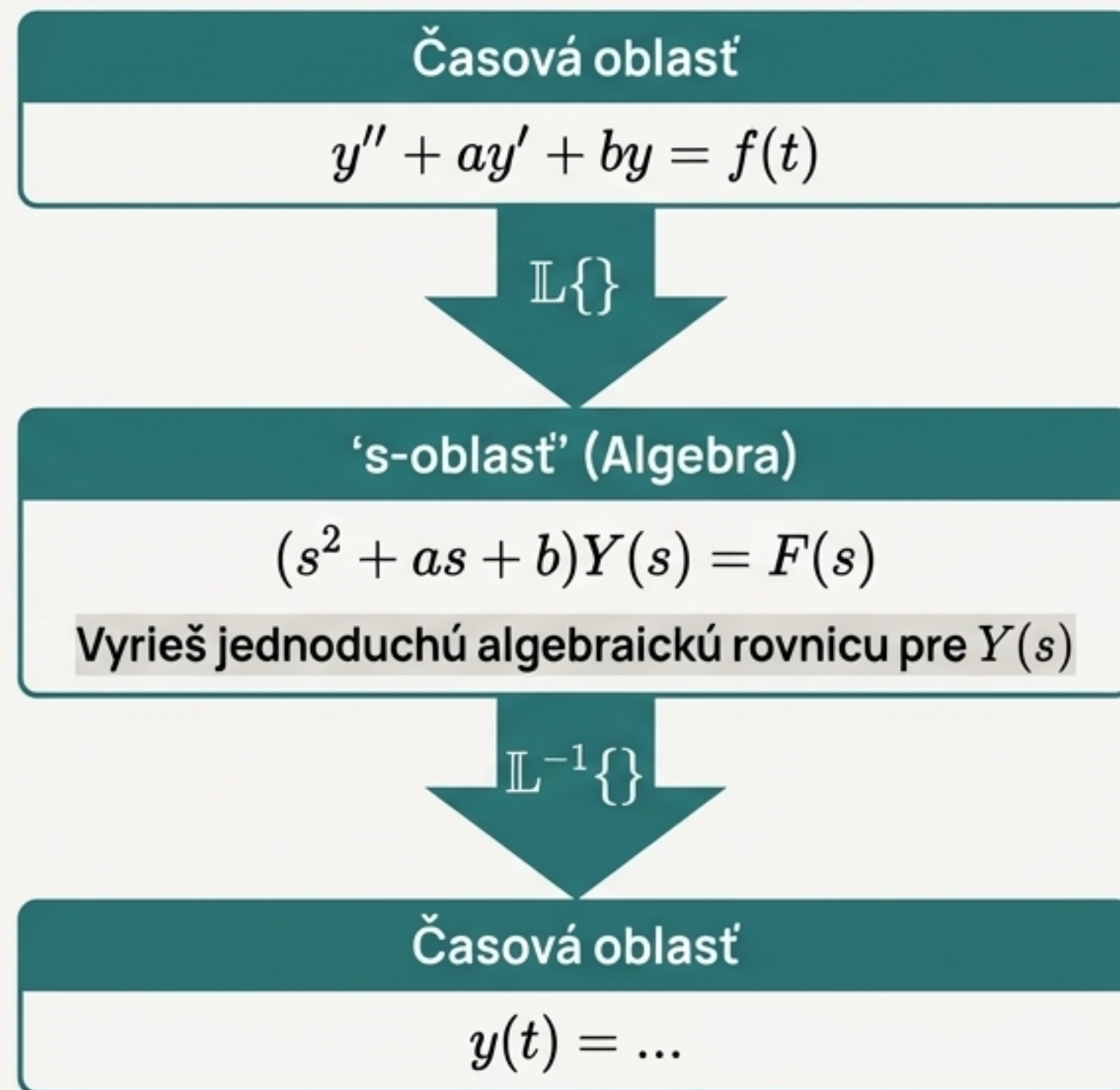
Hlavná vlastnosť (The Magic Trick)

- Transformácia derivácie:
 $\mathbb{L}\{f'(t)\} = sF(s) - f(0)$
- Transformácia druhej derivácie:
 $\mathbb{L}\{f''(t)\} = s^2F(s) - sf(0) - f'(0)$



Prečo komplexné čísla?

Premenná $s = \sigma + i\omega$ je komplexná. Reálna časť σ súvisí s tlmením alebo rastom, zatiaľ čo imaginárna časť ω súvisí s osciláciami.



Aplikácia v avionike: Modelovanie Roll dynamiky dronu.

Cieľ Numericky modelovať a riadiť náklon (roll) kvadrokoptéry okolo osi x .

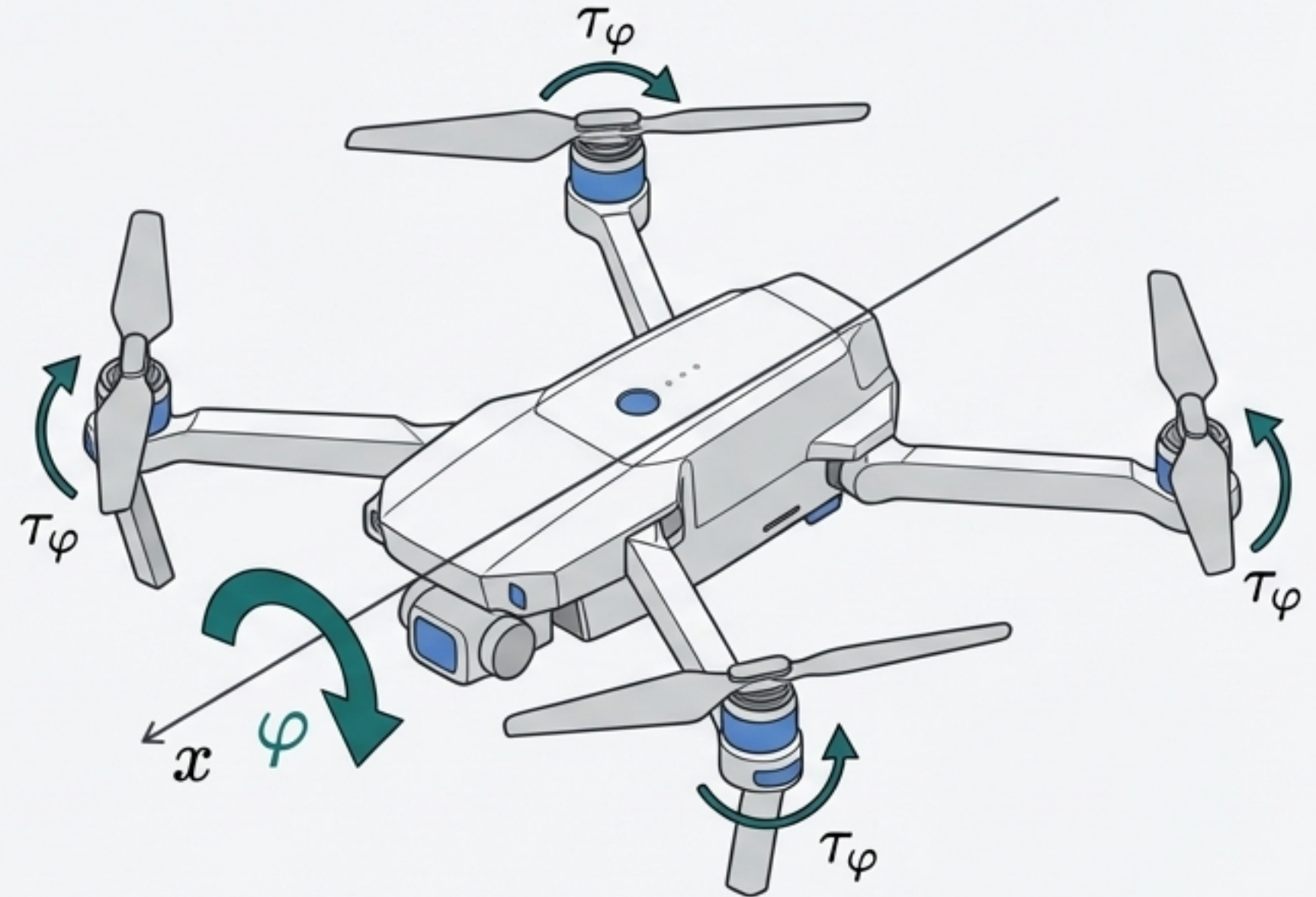
Stavový vektor $\mathbf{X} = \begin{bmatrix} \varphi \\ p \\ z \end{bmatrix}^T$

- φ : Uhol náklonu (roll angle)
- p : Uhlová rýchlosť náklonu (roll rate)
- z : Integrál regulačnej chyby

Diferenciálne rovnice pohybu (zjednodušené)

$$\frac{d\varphi}{dt} = p \quad (\text{Zmena uhla je uhlová rýchlosť})$$

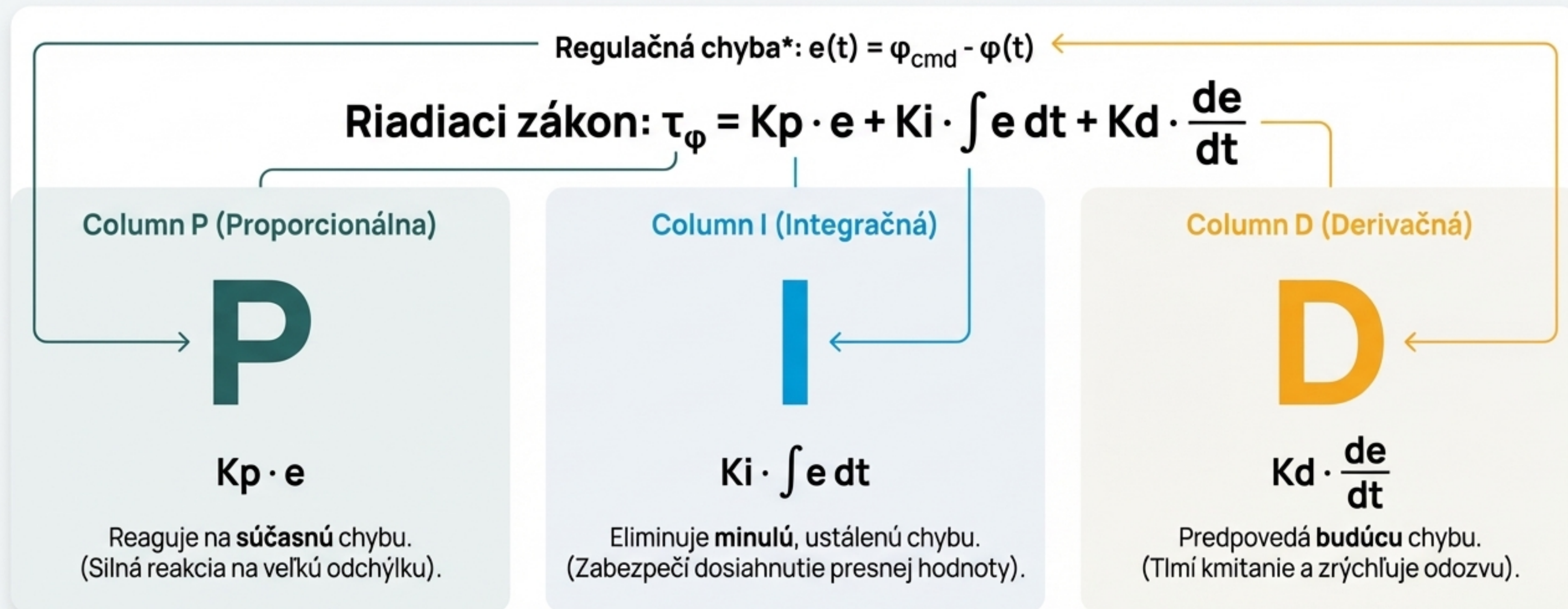
$$\frac{dp}{dt} = \frac{\tau_\varphi}{I_x} \quad (\text{Zmena uhlovej rýchlosti je daná momentom } \tau_\varphi \text{ a zotrvačnosťou } I_x)$$



Prevzatie kontroly: Stabilizácia dronu pomocou PID regulátora.

💡 **Problém:** Ako vypočítať správny riadiaci moment τ_φ , aby sa dron naklonil na požadovaný uhol φ_{cmd} ?

📊 **Riešenie:** PID regulátor – Proportional-Integral-Derivative.



****Širší kontext:** Pre zložitejšie úlohy sa používajú pokročilejšie regulátory ako **LQR** (Linear Quadratic Regulator) alebo **MPC** (Model Predictive Control).

Kompletný obraz: Dynamický model malého dronu (trieda DJI Mini).

Rotačná dynamika (Roll-Pitch-Yaw):

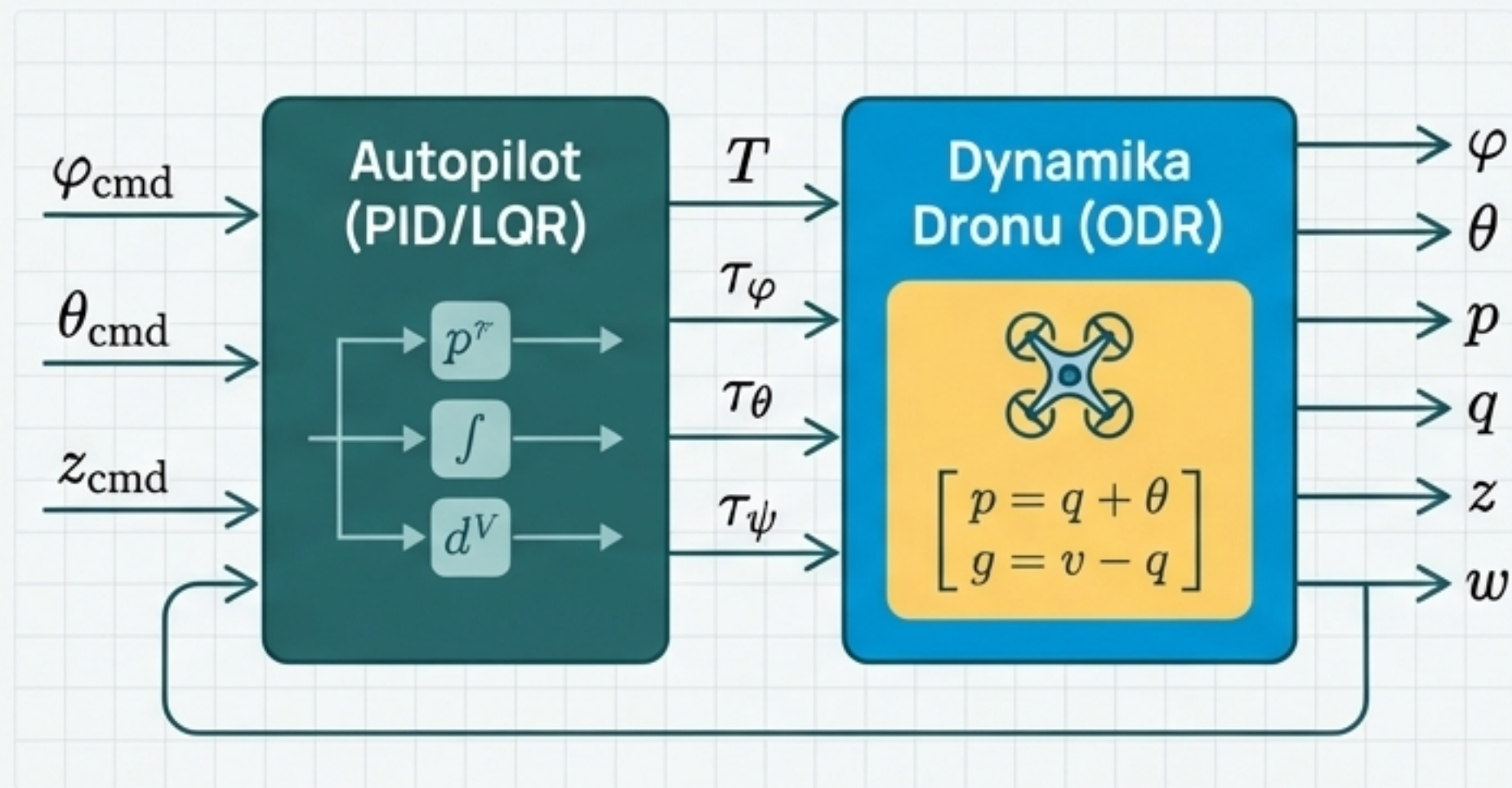
Eulerove dynamické rovnice pre pevné teleso:

$$\dot{p} \approx \frac{\tau_\varphi}{I_x}$$

$$\dot{q} \approx \frac{\tau_\theta}{I_y}$$

$$\dot{w} \approx \frac{\tau_\psi}{I_z}$$

- Kinematické rovnice pre prevod uhlových rýchlostí na zmenu uhlov.



Od riadenia k motorom:

Autopilot vypočíta potrebné $T, \tau_\varphi, \tau_\theta, \tau_\psi$.

Tieto hodnoty sa následne prepočítajú na individuálne ťahy štyroch motorov T_1, T_2, T_3, T_4 .

Vertikálny pohyb:

Rovnica pre zmenu výšky:

$$\dot{z} = w$$

$$\dot{w} \approx \frac{T}{m} - g$$

Zrýchlenie je dané celkovým ťahom T proti gravitácii g .

Váš súbor nástrojov pre moderné letectvo.

Zhrnutie cesty:



1. **Pochopili sme základy:** Uvedomili sme si limity počítačov a dôležitosť riadenia chýb.



2. **Zostavili sme si arzenál:** Osvojili sme si metódy na riešenie rovníc, prácu s dátami, integráciu a simuláciu dynamických systémov (ODR).



3. **Aplikovali sme ich v praxi:** Navrhli sme, modelovali a riadili sme dynamiku letu dronu – od základných rovníc až po komplexný model.

Finálna myšlienka: Numerické metódy nie sú len abstraktná matematika. Je to jazyk, ktorým inžinieri analyzujú, simulujú a riadia reálny svet.

Zdroje a ďalšie informácie:

Celá publikácia: <https://peter.szabo.website.tuke.sk/download/NMvp.pdf>

ISBN: 978-80-553-4929-9 (e-publikácia)

