

# Technická zpráva

Způsob práce, forma zprávy a obsah

**Pavel Krsek**

březen 2024

- ◆ Způsob práce, postup řešení a hodnocení
- ◆ Struktura technické zprávy
- ◆ Faktický obsah zprávy
- ◆ Příklad odevzdaných textů (jak zpráva vypadat nemá)



## Základní/ střední škola

- ◆ Přesné zadání s detailním seznamem kroků
- ◆ Hodnotí se dodržení jednotlivých kroků a postupů
- ◆ Je nepřipustný jiný než požadovaný postup

## Vysoká škola (budoucí inženýr)

- ◆ Obecně zadaná úloha bez přesného návodu
- ◆ Jednotlivé kroky a jejich realizaci navrhuje student
- ◆ Hodnotí se výsledek a správnost navrženého postupu (metody)
- ◆ Očekává se a hodnotí aktivní přístup k řešení
- ◆ K řešení je potřeba přistupovat komplexně a “předvídat” problémy

## Postup řešení

1. **Přečíst pečlivě zadání a splnit všechny body.**
2. Rozmyslet způsob řešení.
3. Sestavit potřebné rovnice a základní algoritmy.
4. Průběžně připravovat testovací příklady.  
Je potřeba projít všechny zákoutí úlohy.
5. Implementovat vztahy a algoritmy.
6. Ověřit funkčnost na příkladech (jeden příklad nic neznamena).  
Dle potřeby implementovat kód pro testování.
7. Předvést, předat či odevzdat funkční kód.
8. Pokud je požadováno, dokumentovat jednotlivé kroky.

## Způsob hodnocení práce

- ◆ Znamku A mohou dostat jen studenti, kteří splní úlohu s maximální mírou kvality.
- ◆ Je třeba mít snahu úlohu vyřešit a zprávu napsat co možná nejlépe.
- ◆ Pouhé minimální splnění zadání neodpovídá přístupu studenta s hodnocením A. (např. místo výsledků a grafů pouze tvrzení “funguje to”)

## Proč se učíme psát technické zprávy?

- ◆ Technické zprávy, návody, normy a směrnice jsou typickým výstupem práce inženýra.
- ◆ Slouží k dokumentaci výsledků, ke komunikaci s ostatními a k řízení lidí.
- ◆ Dle těchto výstupů bývají techničtí pracovníci hodnoceni.
- ◆ Pozor na chyby (pravopis, jednoduché vzorce).  
Zejména neschopní podřízení takové chyby využijí.

## Formální obsah zprávy

## Faktický obsah zprávy

# Formální obsah zprávy

1. Titulní stránka (název, autor, datum, ...).
2. Úvod (stručný popis řešeného problému).
3. Rozbor problému, návrh řešení, očekávaná funkčnost (specifikace, formulace úlohy, související práce).  
Pokud z rozboru vyplývá je možné uvést změny specifikace.
4. Řešení problému, matematický popis, algoritmy.
5. Implementace (nezacházet do detailů).
6. Popis experimentů a experimentální výsledky.
7. Kritická diskuse výsledků (závěry).
8. Závěr (stručné shrnutí výsledků).
9. Literatura (citace použitých zdrojů).
10. Přílohy (jsou-li nutné).

*Poznámka:*

*Zpráva nemusí obsahovat všechny uvedené kapitoly (body).  
Je na úvaze autora, co je pro konkrétní text relevantní.*

# Obecné požadavky na technický text

- ◆ Text neobsahuje věcné ani gramatické chyby.
- ◆ Každý dokument musí být opatřen jménem autora či autorů a datem vypracování.
- ◆ Číslují se jednotlivé prvky v textu (stránky, rovnice, obrázky, tabulky), aby bylo možné na ně odkazovat.
- ◆ Každý obrázek a tabulka mají legendu, která popisuje jejich obsah.
- ◆ Rovnice a matematické symboly jsou zapsány ve formě používané v matematických knihách, ne ve formě pseudokódu.
- ◆ Použité formulace by měly být jasné a jednoznačné.
- ◆ Pro jednu věc používáme vždy jedno označení, které při prvním použití vysvětlíme.
- ◆ **Grafy:** popsané osy, uvedené jednotky, legenda k průběhům.
- ◆ **Tabulky:** srozumitelné a jednoznačné texty, nezapomínat na jednotky.
- ◆ Čísla jsou uváděna s odpovídající přesností (záleží na jejich původu).
- ◆ Text musí navazovat (formálně i fakticky) i pokud ho píše více autorů.

# Faktický obsah zprávy

## Otázky, které si musíme zodpovědět’:

- ◆ Pro koho je zpráva určena?
- ◆ Co chceme zprávou čtenáři sdělit?

## Různý pohled na obsah zprávy

1. Text a údaje ve zprávě musí umožnit úplnou implementaci navržených metod a algoritmů. Člověk se stejným vzděláním jako autor (například spolužák) by měl být schopen podle technické zprávy algoritmus znovu implementovat se stejným chováním a výsledky.
2. Popis a uvedené hodnoty musí být dostačující pro posouzení správnosti a funkčnosti navržených metod a algoritmů jiným technikem s podobným vzděláním.
3. Obsah zprávy by měl přesvědčit čtenáře o funkčnosti implementovaného řešení (pečlivě popsané experimenty, výsledky, grafy). Musí umožnit opakování experimentů.



# Co napsat do zprávy - úvod/ závěr

## Úvod

- ◆ Seznamuje čtenáře obecněji s problematikou.
- ◆ Uvádí motivaci a cíle práce (stručné zadání).
- ◆ Definuje základní předpoklady a omezení.
- ◆ Může naznačit způsob řešení (zejména pokud je voleno).

## Závěr

- ◆ Stručně shrnuje provedenou práci (obsah zprávy).
- ◆ Uvádí nejdůležitější výsledky a závěry (informace, které autor sděluje).
- ◆ Může stručně popsat možnosti pokračování práce.

## Abstrakt

- ◆ Je stručným shrnutím úvodu a závěru.
- ◆ Nemusí být vždy přítomen.

## *Poznámka:*

*Čtenář by měl být po přečtení úvodu a závěru obeznámen s problematikou i všemi podstatnými výsledky a závěry, které mu chceme sdělit. Zbylý text zprávy rozvíjí, upřesňuje a dokládá tvrzení uvedená v těchto částech.*

# Co napsat do zprávy - stať

**Rozbor problému** – cílem je popsat postup řešení, rozdělit na dílčí úlohy.

- ◆ Popis jednotlivých dílčích úloh a jejich rozhraní.
- ◆ Očekávaná funkčnost řešení a podmínky (specifikace).
- ◆ Součástí bývá i rozbor souvisejících prací ostatních (literatura).

**Řešení problému** – představuje matematický popis a algoritmy použité pro řešení.

- ◆ Matematický popis jednotlivých kroků a jejich řešení (výpočty).
- ◆ Popis algoritmů a postupů (slovní, pseudokód, odkazy na literaturu).
- ◆ Musí být uvedeny detaily nutné pro implementaci (vč. konkrétních hodnot).
- ◆ Součástí může být i upřesnění či úprava specifikace.

**Implementace** – popisuje způsob realizace algoritmů

- ◆ Použitý hardware a programovací jazyk.
- ◆ Detaily implementace (např. datová reprezentace, členění programu, optimalizace/ zjednodušení výpočtu, nové algoritmy).

# Co napsat do zprávy - obrázky

Obrázek bývá srozumitelnější než dlouhý výklad.

- ◆ Blokové schema, Vývojový diagram, Stavový diagram ...

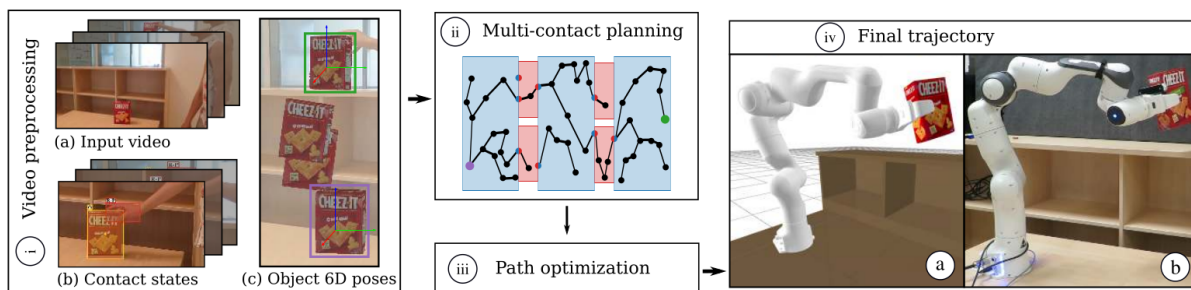


Fig. 2. **Approach overview.** (i) First, we extract contact states and 6D object poses from the input instructional video as described in Sec. III-A. (ii) Next, we grow multiple trees in the admissible configuration space until we find a path between the start and goal configurations. More details on the state space are in Sec. III-B and more details on planning the path in Sec. III-C. (iii) This path is then further shortened by an optimization module, and (iv) executed either in simulation (iv-a) or on a real-world robot (iv-b).

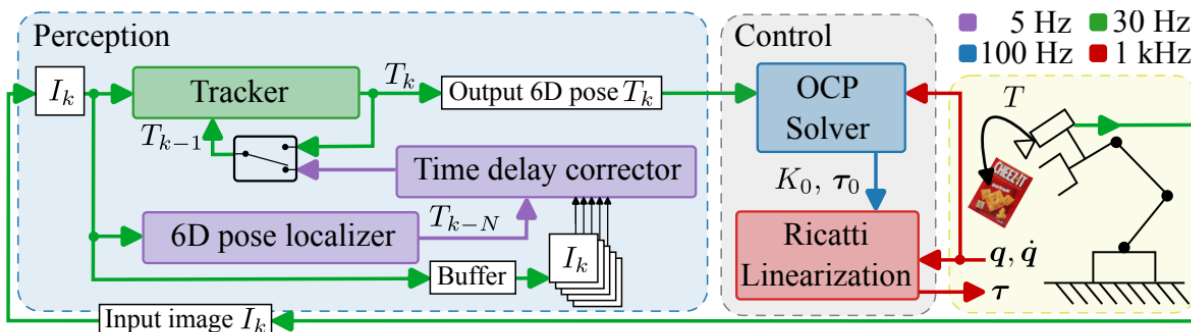


Fig. 2: **Overview of the perception-control cycle.** The objective of the feedback control is to track 6D pose of an object seen by a camera, as illustrated on the right by a robot and red cheez-it box. To achieve that, we designed a perception module that runs a fast local Tracker on an input image  $I_k$  with the initial pose  $T_{k-1}$  selected either from the previous run of the tracker or from the 6D pose localizer & Time delay corrector modules, if that information is available. The 6D pose localizer is slow and the objective of the Time delay corrector is to catch-up in time by quickly tracking through images stored in the buffer while the 6D pose localizer was computing. The output of the tracker, the pose  $T_k$ , is used by the OCP solver to compute Ricatti gains  $K_0$  and torques  $\tau_0$  that are used by the Ricatti Linearization module to provide fast feedback for real-time robot control. Typical processing frequencies of individual modules are 5 Hz for the 6D pose localizer and the time delay corrector, 30 Hz for the camera and tracker, 100 Hz for the OCP solver, and 1 kHz for real-time robot control.

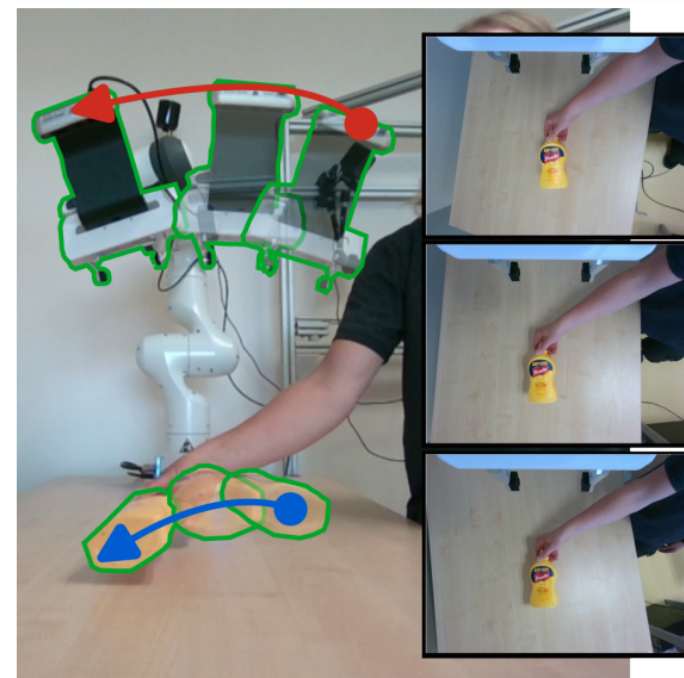


Fig. 1: **Robot arm control by 6D pose of the object.** The objective is to control the robot arm with a mounted camera (red arrow) by commanding joint torques such that the object 6D pose (blue arrow) w.r.t. the camera remains constant. This is illustrated by three frames (see insets) captured by the robot camera corresponding to the robot/object poses shown by green contours in the main image. Please note (see the insets) how the object pose remains stable while the background changes in the captured frames. **More results and experimental analysis in the companion video.**

# Co napsat do zprávy - výsledky

## Experimenty a výsledky

- ◆ Definice cíle experimentů a návrh provedení.
- ◆ Způsob provedení a specifikace vyhodnocení (co měříme).
- ◆ Samotné výsledky (grafy, tabulky).

**Diskuse výsledků a závěry** – výsledky experimentů vedou k závěrům.

**Literatura** – citace použitých zdrojů.

## Přílohy

- ◆ Používají se pro zvýšení přehlednosti textu (redukce rozsahu).
- ◆ Patří sem informace, které chceme sdělit (uchovat), ale pro výklad nejsou nezbytné.
- ◆ Často jsou zde delší důkazy, detaily implementace (výpisy kódu).
- ◆ Umisťují se zde podpůrné a doplňující výsledky (data), rozšířené experimenty.  
V textu se uvádí nezbytná část výsledků a závěry s odkazem na výsledky v příloze.

## Kinematická kalibrace manipulátoru.

Provedte kinematickou kalibraci paralelního planárního manipulátoru.

1. Sestavte soustavu rovnic vazbových podmínek.
2. Navrhněte kalibrační polohy manipulátoru.
3. Navržené kalibrační polohy definované kloubovými souřadnicemi uložte do souboru `data.mat` ve formátu MAT, který bude součástí odevzdávaného archivu.
4. Implementujte v MATLABu iterativní algoritmus kinematické kalibrace manipulátoru.
5. V průběhu kalibrace nezapomínejte kontrolovat podmíněnost.
6. Připravte si pomocí řešení DKT data pro testování a ověřte funkčnost kalibrace.
7. Funkčnost kalibrace dokumentujte ve zprávě. Zajímavá je například chyba odhadu parametrů manipulátoru v závislosti na přesnosti referenčního měření.
8. Pokud je k Vašemu řešení potřeba Jacobiho matice kritériální funkce, uveďte ji do zprávy (pokud není automaticky generovaná např. v Maplu). Pokud použijete standardní optimalizační funkci, uveďte parametry, které jste nastavili.
9. Zkontrolujte syntaxi Vašeho kódu příkazem `mlint`. Váš kód by neměl generovat žádné chyby ani varovná hlášení.
10. Průvodní nákresy, schemata či odvození případně zprávu připravte ve formátu PDF. Zprávu nahrajte jako výsledek úlohy označené stejným pořadovým číslem jako zadaná úloha s dovětkem "report".

# Příklad zprávy – texty

PDF soubor/y

# Příklad zprávy – co chybí

## Příklad 1:

- ◆ Chybí jméno autora a datum vypracování.
- ◆ Název zprávy má být srozumitelný a odpovídá obsahu (HW\_5 to není).
- ◆ Nejsou uvedeny vazbové rovnice ani výpočet J.
- ◆ Není popsána volba kalibračních bodů. Nejsou uvedeny body ani zdůvodnění.
- ◆ Postrádám ověření funkčnosti kalibrace (viz zadání).

## Příklad 2:

- ◆ Rovnice ve zprávě mají být uvedeny v matematickém zápisu
- ◆ Rovnice pro výpočet “new\_offset” nedává díky svému zápisu smysl (pravděpodobně chybná).
- ◆ Nejsou vysvětleny jednotlivé parametry rovnice pro výpočet “new\_offset”.
- ◆ Graf na obrázku (a) nejsou popsány osy ani význam barev.
- ◆ Tento graf postrádá smysl, protože z něj nejde nic rozumného vyčíst.
- ◆ Není popsána volba kalibračních bodů.
- ◆ Není definována chyba ani popsán způsob minimalizace.
- ◆ Postrádám ověření funkčnosti kalibrace (viz zadání).  
Například graf závislosti odhadu parametrů na chybě měření.

# Využití nástrojů UI

## Základní teze

- ◆ Využití nástrojů UI může výrazně pomoci (programování, kontrola pravopisu, reformulace, rešerše, seznámení s tématem).
- ◆ Využití nástrojů UI musí být vždy v souladu s etickými zásadami.
- ◆ Reálně hrozí použití cizího textu/ výsledků bez citace.
- ◆ Za obsah díla zodpovídá autor (včetně chyb použité UI).

## Kybernetická rizika

- ◆ UI nemá schopnost zachovávat důvěrnost informací či chránit osobní údaje.
- ◆ Předávané texty a data jsou obvykle využity pro další učení.
- ◆ Data poskytnutá UI jsou čitelná/ majetkem poskytovatele UI.
- ◆ Problémem je sdílení citlivých údajů, osobních dat, vlastních výsledků.

## Co určitě nenechat na UI

- ◆ Teze, Výsledky (nutno rozumět tématu a mít podloženo vlastní prací)
- ◆ Citace (UI má tendenci si vymýšlet, nutná velmi pečlivá kontrola)

**Poznámka:** *Nástroji UI jsou v tomto případě aplikace jako ChatGPT, Microsoft Bing, Google Bard, Github Copilot, Midjourney, Stable Diffusion, Jasper a další ?*



## Dva případy:

### 1. Obhajoba bakalářské práce

- ◆ Chybně napsaný vzorec (Bayes) v BP
- ◆ Vzorec dobře opsaný z literatury (chyba autora)
- ◆ Literatura nebyla citována



### 2. Jedna z úloh v rámci předmětu

- ◆ Student opakuje předmět
- ◆ V minulém roce odevzdal úlohu správně
- ◆ Letos nahrál do odevzdávacího systému řešení spolužáka
- ◆ Vysvětlení: sdílejí počítač na koleji - omyl při odesílání



## Použité zdroje

[Achtenova-AI2024] Gabriela Achtenová. **Rámcová pravidla používání umělé inteligence na ČVUT pro studijní a pedagogické účely v Bc a NM studiu.** Metodický pokyn ČVUT v Praze, CVUT00001676/2024, January 2024.  
<https://www.cvut.cz/legislativa-tykajici-se-studia#umela-inteligence>

[Achtenova-etic2024] Gabriela Achtenová. **Metodický pokyn O dodržování etických principů při přípravě vysokoškolských závěrečných prací.** Metodický pokyn ČVUT v Praze, CVUT00001673/2024, January 2024.  
<https://www.cvut.cz/legislativa-tykajici-se-studia#eticke-principy>