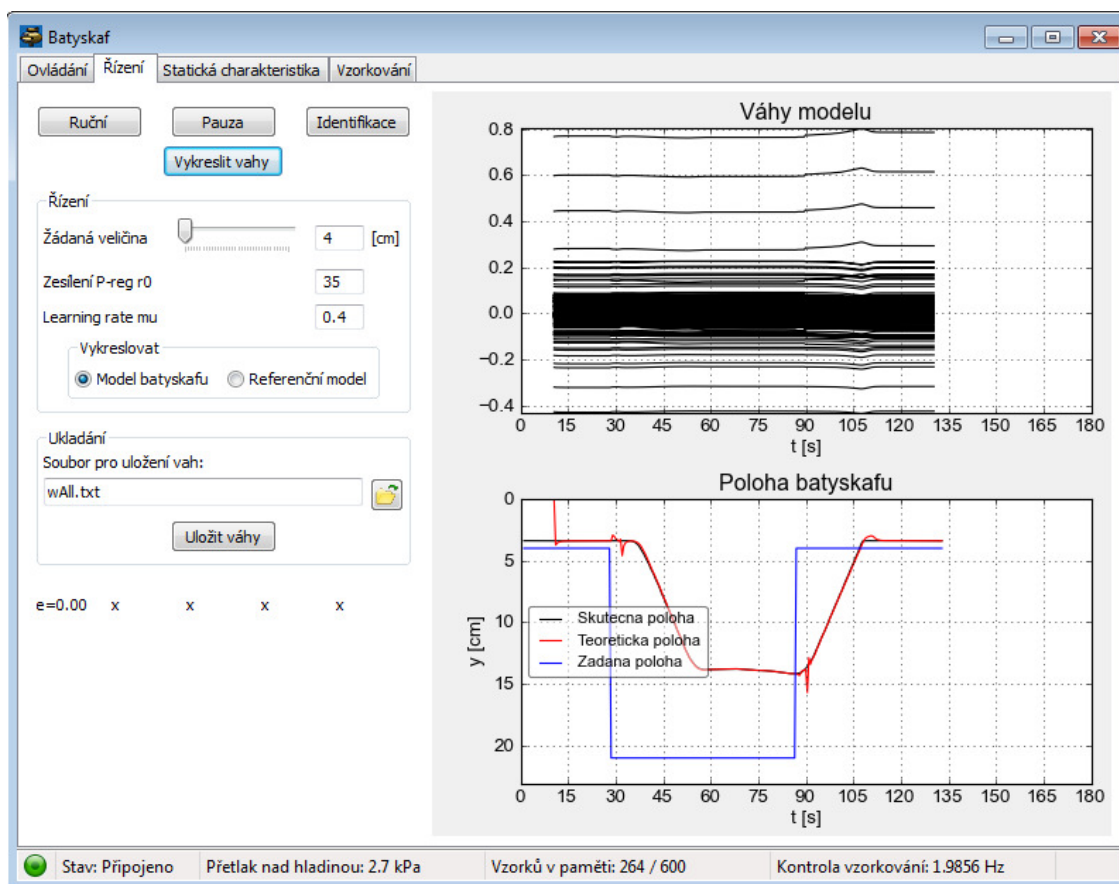


# **Ovládání laboratorní úlohy v prostředí Python**

V této práci popíši vývoj ovládacího programu pro laboratorní úlohu Batyskaf, od verze, která vyšla a byla obhájena v rámci mé bakalářské práce. Další vývoj programu probíhal v rámci předmětu Python pro vědecké výpočty a řízení s cennými radami a připomínkami Doc. Ing. Iva Bukovského, Ph.D.

Hlavní body:

- zvýšení stability programu
- zpřesnění vzorkovací frekvence
- přidání záložky řízení obsahující:
  - tlačítko „Automatika“ pro spuštění regulace
  - Identifikace soustavy
  - P – regulátor s nastavitelným zesílením  $r_0$
  - nastavování žádané veličiny
  - QNU Batyskafu
  - referenční soustavu
  - možnost vykreslování a ukládání průběhu vah modelu
  - nastavování „Learning rate“ modelu



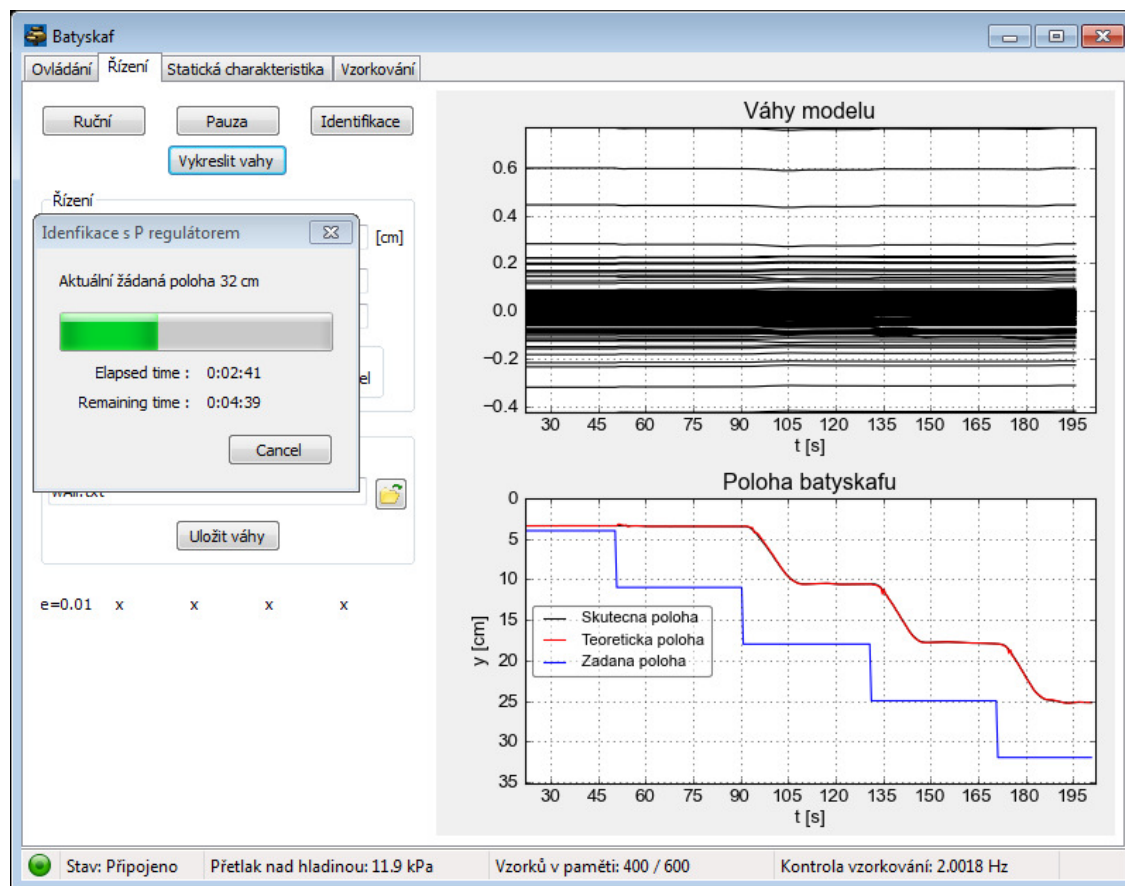
Obr 1. – Nová záložka „Řízení“ s možností vykreslování vah modelu

Ke zvýšení stability programu přispělo hlavně odstranění a opravení menších nedostatků v kódu programu a zlepšení komunikace s úlohou. Největší chybou v komunikaci s Batyskafem bylo špatné pořadí ukončování spojení po sériové lince a tím docházelo ke zmatení řídicí jednotky Batyskafu.

Zpřesnění vzorkovací frekvence bylo dosaženo nahrazením složitějšího dvoustupňového časovacího algoritmu funkcí `time.sleep()`.

Hlavní motivací pro další vývoj programu je implementace adaptivního regulátoru a za tímto účelem byla přidána záložka Řízení, které bude obsahovat veškeré nastavování parametrů regulace úlohy. V současné době program obsahuje pouze P regulátor, který reguluje polohu plováčku s regulační odchylkou asi 5 cm. Také obsahuje referenční model, což je vlastně ideální chování soustavy, kterému se adaptivní regulátor bude snažit co nejvíce přiblížit. Dále je v programu zahrnuta neuronová jednotka modelující chování plováčku s P regulátorem. Její váhy byly vypočteny mimo program, který je při spuštění načítá a dle nastavené „Learning rate“ dále adaptuje. Na záložce Řízení lze tedy momentálně nastavovat zesílení P regulátoru  $r_0$ , learning rate modelu  $\mu$ , žádanou polohu plováčku. Dále lze pomocí tlačítka kdykoliv vykreslit průběh adaptace vah a ten si také uložit do txt souboru. Pomocí výběrových políček lze také volit zda chceme vykreslovat vypočtenou polohu plováčku pomocí QNU nebo polohu dle referenčního modelu.

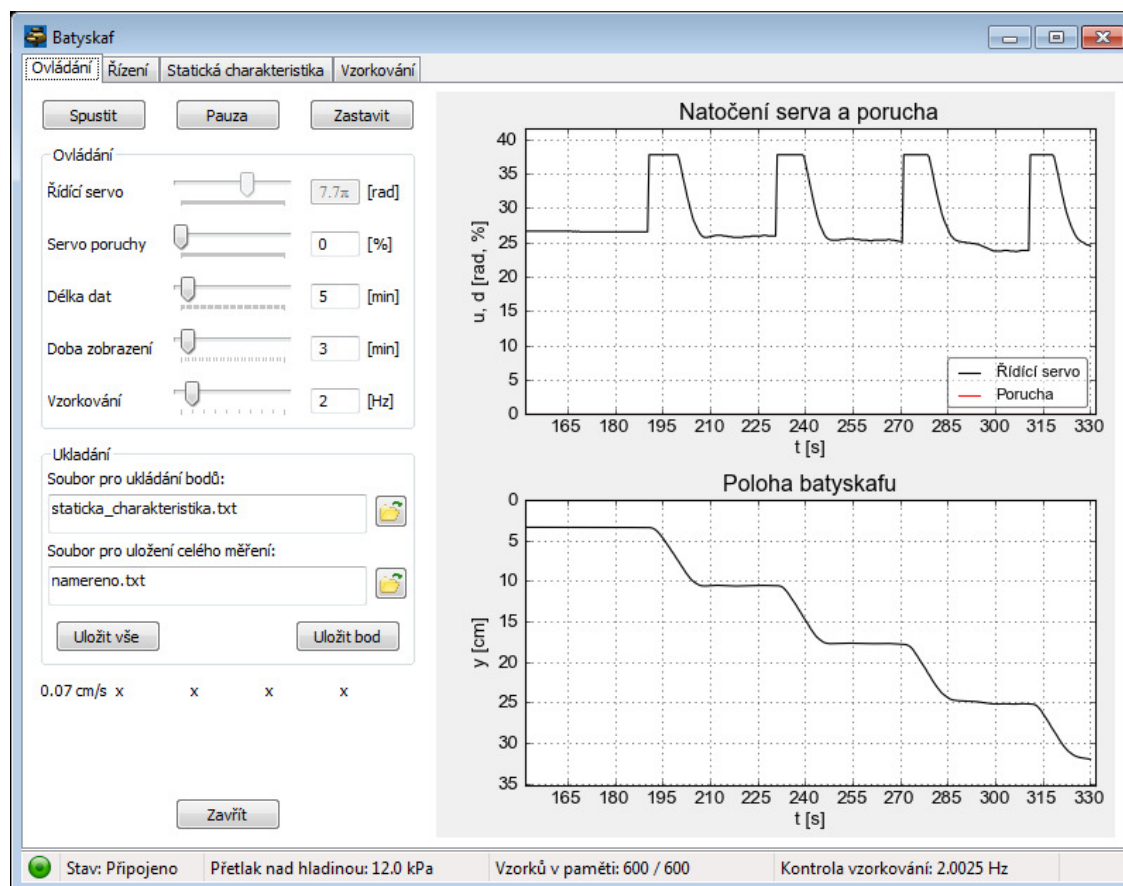
Momentálně tlačítko Identifikace pouze „projede“ v pěti schodech dolů a nahoru plováček pomocí žádané veličiny, tzn. za použití pouze P regulátoru.



Obr 2. – Probíhající identifikace – s ukazatelem průběhu

Vývoj programu bude i nadále pokračovat převážně implementací adaptivního regulátoru a jeho odladěním. Ideálním cílem je, aby byl program dále použitelný při výuce jako názorná pomůcka.

Možné je i rozšíření funkcí identifikace, aby to nebylo pouhé prohnání plováčku, ale aby program uměl ze zjištěných průběhů dále například spočítat váhy pro model soustavy.



Obr 3. – Regulační zásah P regulátoru se zesílením  $r_0=35$  (v horním grafu průběh akční veličiny)

#### Literatura:

- [1] Jakub Čepela: „Ovládání laboratorní úlohy v prostředí Python“, ČVUT FS, bakalářská práce (vedoucí práce Ivo Bukovský), Ústav přístrojové a řídicí techniky, Odbor automatického řízení a inženýrské informatiky, obhájeno 2012.