

WASH-OUT ALGORITMUS PRO ŘIDIČSKÉ A LETECKÉ SIMULÁTORY

E. Thöndel

Katedra elektrických pohonů a trakce, FEL ČVUT v Praze

Abstrakt

Příspěvek popisuje vývoj simulačního modelu algoritmu simulace pohybových vjemů (wash-out algoritmus), který je vyvíjen v rámci projektu, jehož cílem je vytvořit a prakticky ověřit systém pro simulaci pohybových vjemů určený pro výcvik, výzkum a vývoj v oblasti rozhraní člověk-stroj. V článku je podrobně rozebrána implementace rozšířeného wash-out algoritmu, která se snaží o maximální využití celého rozsahu pohybu šestistupňové pohyblivé základny.

1 Základní principy a význam simulace pohybových vjemů

Simulační techniku a její význam v oblasti výcviku řidičů či pilotů není třeba zdůrazňovat. Simulace pohybových vjemů však bývá často opomíjena, a to z důvodu relativně vysokých nákladů při ne zcela prokazatelném přínosu u některých aplikací. Cílem této simulace je reprodukovat síly a momenty působící na pilota/řidiče během letu/jízdy a přiblížit tak simulátor více realitě. Tyto pohybové vjemy jsou zvláště významné při nácvičení zvládnutí krizových situací, protože obecně předcházejí vjemům vizuálním a slouží tedy jako včasná detekce chybového chování dopravního prostředku, přestože pohybové vjemy nemusí být v tomto čase právě v centru pozornosti operátora.

Simulace pohybových vjemů vychází ze základního principu mechaniky: pohybuje-li se těleso zrychleným pohybem, působí na něj setrvačná síla, která má opačný směr než je směr jeho pohybu. Velikost této síly je podle Newtonova zákona úměrná hmotnosti tělesa a jeho zrychlení. Cílem algoritmů simulace pohybových vjemů je reprodukovat tyto síly vzniklé zrychleným pohybem a polohou vůči gravitačnímu poli.

Zařízení používaná pro reprodukci pohybových vjemů mají omezený pracovní prostor, proto není možné reprodukovat všechny síly a momenty v plném rozsahu, ale jen v omezené frekvenční oblasti, či je nutné uchýlit se k určitým trikům. Tato zařízení jsou obecně konstruována jako paralelní robotické manipulátory s různým stupněm volnosti. Nejčastějším typem je tzv. Stewartova plošina se šesti stupni volnosti, která byla užita i v tomto projektu.

2 Šestistupňová pohyblivá plošina

Pohyblivá plošina se šesti stupni volnosti (neboli Stewartova plošina) je synergický systém s paralelní robotickou strukturou, která vyniká vysokým poměrem výstupní síly k hmotnosti zařízení. Pohybová plošina se skládá z horního (pohyblivého) a dolního (nepohyblivého) rámu, které jsou oba spojeny šesti identickými aktuátory.

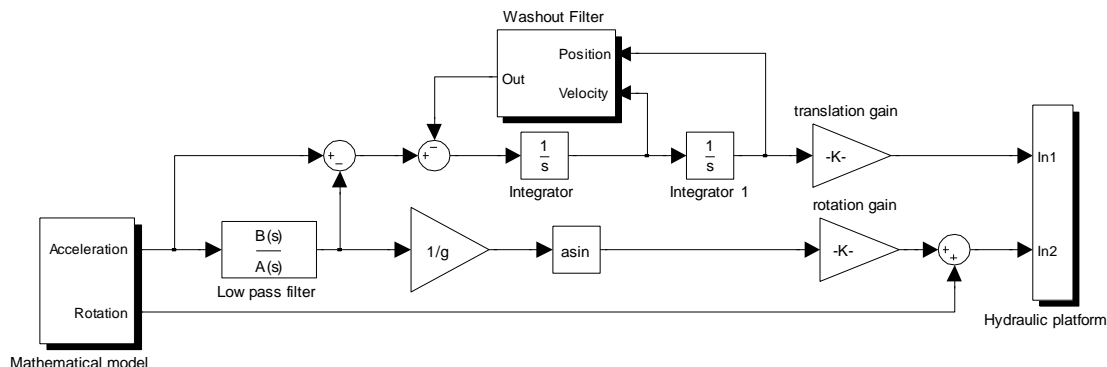
Pohyblivá plošina je schopna konat pohyb do šesti základních směrů. Tyto směry jsou rotace kolem os X, Y a Z (označíme je po řadě Φ , Θ a Ψ) a translace podél těchto os (označíme je po řadě ΔX , ΔY a ΔZ).

Aktuátory jsou nejčastěji realizovány pomocí hydraulických válců, i když v poslední době roste tlak uživatelů na náhradu hydraulického mechanismu elektromechanickým. Oba zmiňované systémy mají své výhody a nevýhody a závisí především na typu aplikace, který ze systémů bude využit.

Tento článek se nezabývá do detailu technickými prostředky pro simulaci pohybových vjemů, zájemce je odkázán např. na publikace [1] nebo [2], kde je podrobně rozebrána struktura uvedeného zařízení s pohledu kinematiky, dynamiky a řízení.

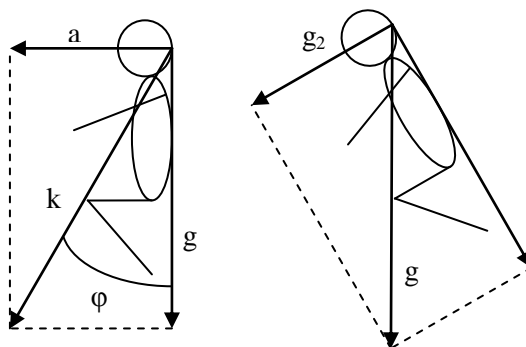
3 Rozšířený wash-out algoritmus a jeho implementace v prostředí Matlab-Simulink

Jak již bylo řečeno, je pracovní prostor pohyblivé plošiny omezený, proto je hlavním úkolem dále popsaného algoritmu, věrohodně reprodukovat síly a momenty působící na řidiče/pilota a zároveň respektovat kinematické a dynamické limity pohybového zařízení.



Obr. 1: Principiální schéma wash-out algoritmu.

Tato omezení se projeví zejména při simulaci translačního zrychlení, neboť rozsah lineárního pohybu základny je omezen řádově na cca 30 – 50 cm. V tomto rozsahu je možné simulovat jen krátkodobá zrychlení. Dlouhotrvající zrychlení by způsobila „vyjetí“ základny mimo omezený rozsah, proto je v tomto případě využito malého triku. Vestibulární systém člověka totiž nedokáže rozlišit zrychlený pohyb od situace, kdy je kabina simulátoru otočena o vhodný úhel (toto platí za předpokladu, že člověk nemá vizuální kontakt s okolím simulátoru – je uzavřen v kabině). Tato situace je znázorněna na obrázku 2.



Obr. 2: Simulace translačního zrychlení náklonem základny.

Na řidiče působí gravitační síla charakterizovaná gravitačním zrychlením \mathbf{g} . Dále na řidiče působí setrvačná síla daná zrychlením \mathbf{a} (obrázek popisuje situaci, kdy vozidlo brzdí). Řidič pociťuje zrychlení \mathbf{k} . Za předpokladu, že $|\mathbf{a}| \ll |\mathbf{g}|$ platí $|\mathbf{k}| \approx |\mathbf{g}|$. To znamená, že výsledek bude stejný, pokud kabinu pootočíme o úhel φ . Poslední tvrzení můžeme vyjádřit matematicky:

$$\varphi = \arcsin\left(\frac{|\mathbf{a}|}{|\mathbf{g}|}\right) \quad (1)$$

Předchozí podmínka přibližně platí až do zrychlení 5 m/s². Odpovídající úhel je v tomto případě $\varphi = 30^\circ$. Rotace kabiny však musí probíhat dostatečně pomalu (nejlépe pod prahem citlivosti člověka), aby člověk uvnitř kabiny jen minimálně cítil pohyb základny.

Popišme dále celý algoritmu na implementaci pomocí nástroje MATLAB-Simulink (viz obrázek 3). Translační vektor zrychlení vstupující do algoritmu je rozdělen na vysokofrekvenční část, která je simulována translačním pohybem základny, a nízkofrekvenční část, která je simulována

náklonem základny. V klasickém pojetí algoritmu se používá lineárního filtru druhého i vyššího řádu. Zde je použito jiného přístupu, jak je vidět na obrázku 3, kde je využito pro rozdělení vstupního vektoru zrychlení nelineárního filtru s jedním integrátorem, jehož matematický popis je uveden níže.

$$\begin{aligned} \frac{d\Theta}{dt} &= K_{a \rightarrow \omega} \underbrace{\left[a_x - \underbrace{g \sin \Theta}_{a_{x,low}} \right]}_{a_{x,high}} \\ \frac{d\Phi}{dt} &= K_{a \rightarrow \omega} \underbrace{\left[a_y - \underbrace{g \cos \Theta \sin \Phi}_{a_{y,low}} \right]}_{a_{y,high}} \end{aligned} \quad (2)$$

V této rovnici určuje konstanta K rychlost náklonu kabiny v závislosti na simulovaném (vstupním) zrychlení. Tuto konstantu lze rovněž interpretovat jako proporcionální složku klasického PID regulátoru. Ve schématu (obrázek 3) je uveden navíc saturační blok, který udržuje rychlost náklonu pod prahem citlivosti člověka.

Tento nelineární filtr vznikl kombinací rovnice 1 a lineárního filtru prvního řádu, což má v praxi řadu výhod:

1. Řád filtru je velmi malý, tedy i dopravní zpoždění je velmi malé.
2. Odstranění neperiodické funkce arkussinus, která v praktické implementaci způsobovala problémy se stabilitou algoritmu.
3. Snadnější interpretace a ladění konstant filtru; jediná konstanta K a mezní hodnoty saturačního bloku umožňují snadné naladění algoritmu s ohledem na kinematické a dynamické vlastnosti pohybové základny.

Nízkofrekvenční část zrychlení je tedy tímto postupem převedena na náklon základny, vysokofrekvenční část je po dvojitě integraci převedena na translační pohyb základny. V této části algoritmu je nutné do schématu zavést zpětnou vazbu, která po odeznění signálu navrátí základnu do její výchozí polohy a připraví ji tak pro další manévr. Tento pohyb se opět musí odehrávat pod prahem citlivosti člověka. Podle této funkce dostal také tento algoritmus své jméno (wash-out = vymývání). Tento filtr je možné realizovat různým způsobem. Popíšme zde způsob vycházející z teorie dynamických systémů.

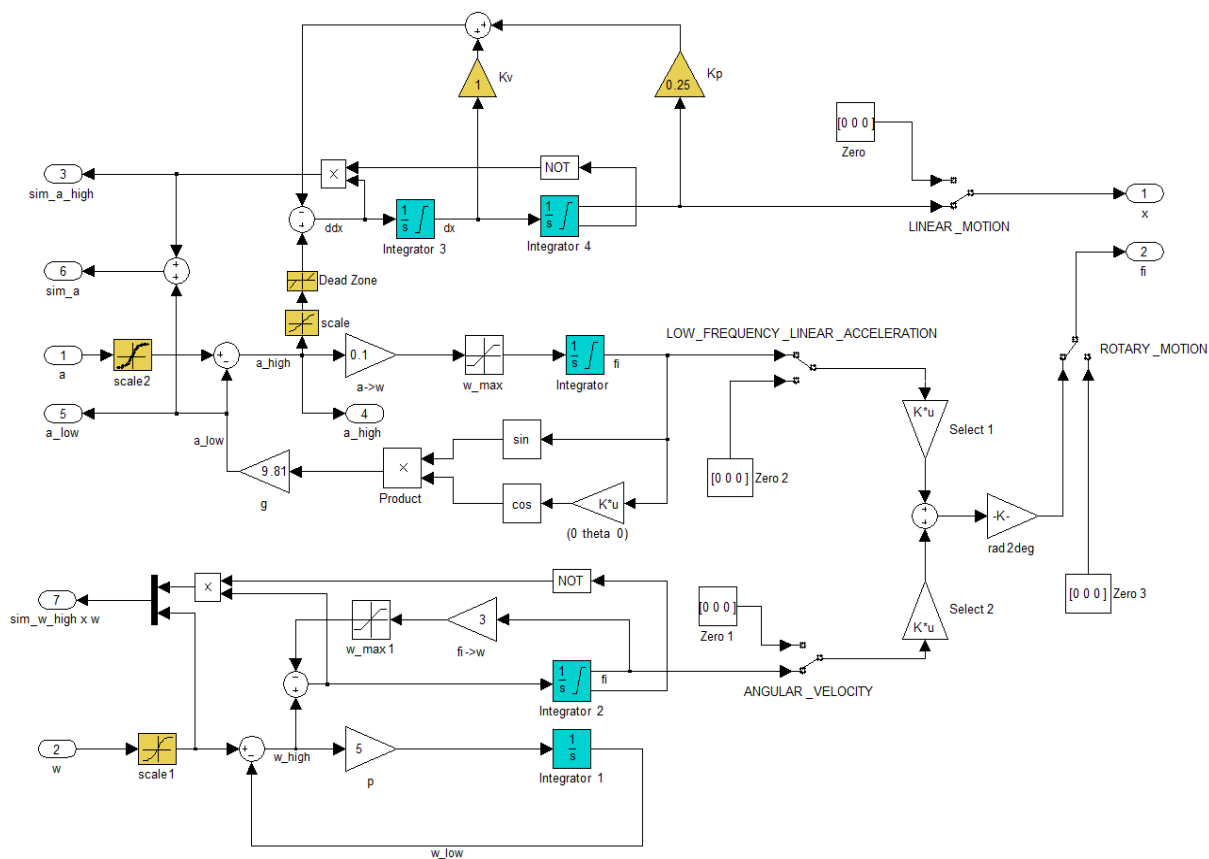
Laplaceův obraz přenosové funkce vysokofrekvenční části tvořené dvojicí integrátorů má tvar:

$$G_h(s) = \frac{1}{s^2} \quad (3)$$

Rozšířme uvedený systém o stavovou zpětnou vazbu, aby výsledný systém byl stabilní. Dalším požadavkem je, aby přechodová charakteristika byla bez překmitu a návrat do stabilního nulového bodu se odehrával pod prahem citlivosti člověka. Nastavme proto stavovou zpětnou vazbu tak, aby výsledný přenos měl jeden dvojnásobný reálný kořen ležící v záporné polorovině komplexní roviny:

$$G_h(s) = \frac{1}{(s - \lambda)^2} = \frac{1}{s^2 - 2s\lambda + \lambda^2} \quad (4)$$

Poloha pólu λ na reálné ose určuje rychlost filtrace a je stanovena empiricky při realizaci algoritmu. Uveďme, že čím menší bude daný pól, tím rychlejší bude filtrace.



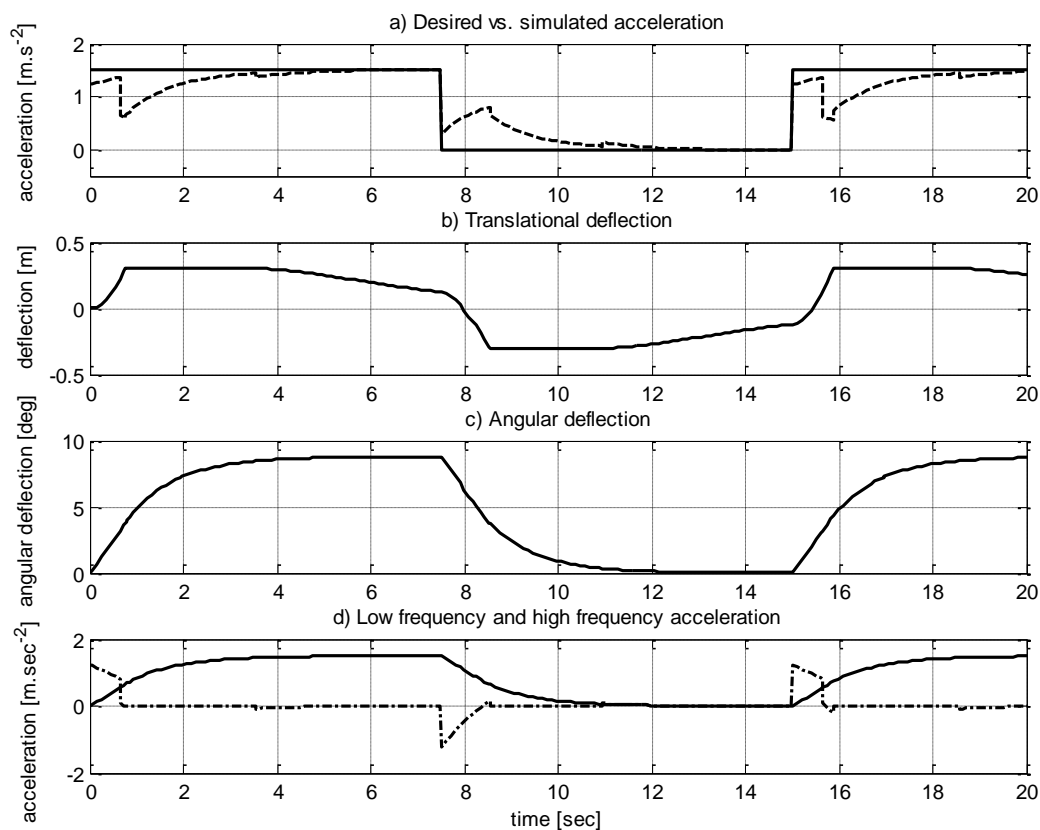
Obr. 3: Implementace algoritmu wash-out v prostředí MATLAB-Simulink.

Vedle pohybových efektů vzniklých translačním zrychleným pohybem, jsou zde další efekty vznikající v důsledku rotačního pohybu (tečné zrychlení, odstředivé zrychlení). Tyto efekty jsou v algoritmu zahrnuty simulací vysokofrekvenční složky úhlové rychlosti, která je po integraci přičtena k úhlové poloze pohybové základny. Obdobně jako v předchozím případě je zde nutné zavést zpětnou vazbu, která po odeznění signálu navrací polohu do výchozího stavu.

Výsledky simulací shrnuje obrázek 4, na kterém jsou zobrazeny průběhy v jednotlivých místech algoritmu při skokové (periodické) změně vstupního zrychlení. Je jasné, že z důvodu omezeného pracovního prostoru základny není možné simulovat vstupní zrychlení v plném rozsahu, avšak praktické testy ukázaly dobré výsledky a vhodnost algoritmu pro použití v simulátorech dopravních prostředků.



Obr. 4: Příklady simulátorů: a) řídicí simulátor bojového vozidla pěchoty, b) simulátor lehkého sportovního letadla.



Obr. 4: a) Porovnání skutečného a simulované zrychlení, b) časový průběh translační výchylky, c) časový průběh rotační výchylky, d) rozdělení vstupního zrychlení na vysokofrekvenční a nízkofrekvenční část.

4 Závěr

Uvedený článek představil jednu z možných implementací algoritmu simulace pohybových vjemů. Demonstrační implementace byla provedena v prostředí MATLAB-Simulink. Velká výhoda uvedeného algoritmu je snadná interpretace jednotlivých částí a tím i snadné nastavení s ohledem na kinematické omezení konkrétního pohybového systému.

Reference

- [1] E. Thöndel. *Simulating motion effects using a hydraulic platform with six degrees of freedom*. proceeding of the Second IASTED Africa Conference on Modelling and Simulation, Gaborone, Botswana, 2008.
- [2] D. Stewart. *A Platform with Six Degrees of Freedom*. UK Institution of Mechanical Engineers Proceedings, 1965-66, Vol 180.
- [3] Z. Šika, P. Beneš, M. Valášek. *New washout algorithms for driving simulator*. International Journal of Engineering Mechanics, 2005.
- [4] T. Alfred Lee. *Flight simulation*. Aldershot, Ashgate, 2005.