

## Posudek vedoucího bakalářské práce

**Název práce:** Automatická analýza vlivu konfigurace Linuxu na výkonnost systému

**Jméno autora:** Karel Kočí

**Vedoucí práce:** Ing. Michal Sojka, Ph.D.  
Katedra řídicí techniky, FEL, ČVUT

Linux je jedním z největších open-source projektů a najdeme ho v miliónech zařízení po celém světě. Přestože se na jeho vývoji podílí stále větší počet lidí a mnozí z nich se starají o zvyšování jeho kvality, stále existují místa, kde má vývojový model Linuxu rezervy. Jak připouštějí sami Linuxoví vývojáři, jedním z takových míst je sledování výkonnosti a zejména pak výkonnostních regresí, tj. situací, kdy dojde k nechtěnému snížení výkonu v důsledku změny nebo přidání nějakého kódu. Práce pana Kočího směřuje ke zmírnění tohoto nedostatku. Cílem je automaticky měřit výkon a pomocí sofistikovaného zpracování výsledků získávat detailnější informace o výkonnosti Linuxu.

Pan Kočí pracoval velmi samostatně. Na začátku byl seznámen s cílem práce a byl mu nastíněn možný způsob řešení. Poté se student pustil do práce a začal přicházet i s vlastními nápady. Konzultace byly vždy konstruktivní. První částí práce bylo vytvoření převodníku z jazyku Kconfig na SAT problém. Zde bych měl ke studentovu postupu drbnou výtku a to, že začal vyvíjet vlastní převodník, místo aby použil nebo upravil existující převodník v nástroji Undertaker. Výsledkem bylo, že se implementací převodníku strávila většina času a na vlastní experimenty a vyhodnocení výsledků už ho moc nezbylo.

Dalším menším недостатkem práce je kvalita autorova písemného projevu. Myslím, že konečná verze textu práce nakonec dopadla docela dobře, ale do budoucna bych chtěl studentovi doporučit, aby na psaní kladl větší důraz. Byla by totiž škoda, kdyby kvalitní technické dílo vypracované bystrým studentem bylo špatně hodnoceno jenom kvůli nekvalitní dokumentaci.

Výsledkem práce pana Kočího je plně funkční nástroj u kterého počítám se s jeho dalším vývojem. Ačkoli autor v závěru zmiňuje, že výsledky nejsou ideální, já jsem něco podobného očekával a věřím, že při jiných experimentech, nebo při drobné změně vyhodnocovacího algoritmu získáme navrženým postupem zajímavé a užitečné výsledky.

S prací jsem tedy spokojen a navrhuji ji hodnotit stupněm B – **velmi dobře**.

V Praze dne 3. června 2015

Ing. Michal Sojka, Ph.D.

# Hodnocení bakalářské práce oponentem

Student: Karel Kočí  
Vedoucí: Ing. Zdeněk Hurák, Ph.D.  
Datum: 19.06.2015

---

Cílem bakalářské práce Karla Kočího bylo prozkoumat možnosti automatického testování vlivu konfiguračních voleb na výkon kompilovaného jádra operačního systému Linux. Navržené a implementované řešení mělo být otestováno na experimentální platformě.

Samotný nápad generovat automaticky různé (funkční) konfigurace pro Linuxové jádro, konfigurované jádro zkompilovat, a na základě změřených kvantitativních charakteristik zkompilovaného jádra usuzovat na vliv té které volby, zní poměrně zajímavě. Mnohé čtenáře sice v první chvíli možná napadne, zda je opodstatněné přistupovat k člověkem navrženému systému (Linuxovému jádru) jako k černé skřínce, o jejímž vnitřku nic nevíme, a snažit se mu porozumět pouze na základě odezvy na externí podněty (konfigurační volby). Ale hned ve druhé chvíli si čtenář jistě uvědomí, že u systému takové složitosti, jako je jádro Linuxu, už to porozumění souvislostem mezi jednotlivými konfiguračními volbami (při uvažování mnoha stovek či tisícovek konfiguračních voleb) jde nad možnosti jednoho člověka. Proto shledávám téma práce velmi zajímavé. Koneckonců, i k analýze numerických algoritmů v matematice takto často přistupujeme.

Jakkoliv je studovaný problém velmi zajímavý, samotné provedení práce je pak o něco rozpačitější, a to z následujících důvodů:

1. **Textová zpráva je velmi těžce čitelná**, a přitom je práce poměrně krátká a téma práce není až tak moc náročné na vysvětlení. Má nespokojenost s textem je dána zejména **častým mícháním koncepčních úvah s popisem implementačních detailů**. K tomu dochází ne pouze v rámci kapitol, ale velmi často i rámci odstavců a někdy i v rámci jednotlivých vět. Kritické je to zejména v prvních kapitolách práce, kde se čtenář má dozvědět motivaci, zvolenou strategii, analýzu úskalí, a jistě nepotřebuje hned dostat instrukce pro psaní komentářů v tom či onom jazyce/nástroji. Vskutku se obávám, v tomto textu **nebyla prokázána schopnost vystihnout to důležité a čtenář je neustále rozptylován technikalitami**, které by si sám mohl najít během pár sekund hledání na internetu.
2. V návaznosti na předchozí výhradu jsem toho názoru, že práce měla začínat až kapitolou 4 (Architektura řešení), ve které je vysvětlena celá filozofie řešení. Tato, společně s kapitolou 6 (Testování a vyhodnocení), tvoří jádro textové části práce a tyto kapitoly měly být zdůrazněny a i více rozvinuty. Ostatní kapitoly popisující volbu konkrétních nástrojů, způsob jejich používání či adresářovou strukturu vlastní implementace jsou sice užitečné, ale mohly se v práci objevit až v pozdějších kapitolách či dokonce v příloze.
3. Jestliže se v úvodu práce píše o analýze vlivu konfiguračních voleb na **výkon systému**, tak tento ale **není pořádně nikde definován**. Je možné jej chápat jako závislý na aplikaci? Tedy že jednou je to doba provádění jednorázové úlohy, jindy třeba paměťové či komunikační nároky? Diskuzi tohoto v práci nenacházím.
4. Poměrně **nesrozumitelnou shledávám část o analýze dat**, která má být přitom tím vyvrcholením práce. Jestli tomu rozumím dobře, pro praktický experiment je vybráno několik binárních konfiguračních parametrů, celkem 10. To dává celkem  $2^{10}=1024$  možných kombinací, pokud nejsou zahrnuty nějaká omezení. V práci v sekci 6.3 se píše o celkem 384 konfiguracích, ale není komentováno, jak konkrétně byly tyto vybrány. Pravděpodobně byly vyloučeny nějaké konfliktní konfigurace. I pro těch 384 konfigurací však následně není z textu v této kapitole jasné, jak vlastně byla změřená data zpracována a jak je tedy možné interpretovat grafy na stránkách 27 a 28. Relevantní pro porozumění grafům je nepochybně

stručná podkapitola 5.3.7, která je však „ztracená“ v sekci popisující kroky jako jsou generování konfigurace či bootování jádra. I ta je však až příliš stručná, celá úvaha vedoucí k řešení přeurené soustavy rovnic měla být detailněji a pečlivěji zdokumentována. V této podobě je dokonce i matoucí. Vždyť jestliže je vysoká matice A (definující lineární soustavu rovnic na straně 21) obsahující pouze nuly a jedničky sestavována tak, že její řádky dávají možné kombinace binárních voleb, pak její řádky musí být z principu závislé, zatímco v textu se píše, že „lineární nezávislost řádků je dána vzhledem k formátu vstupních dat“? Podobně je vysvětlováno, že „musíme zajistit, aby sloupce byly nezávislé“, ale přitom řešiče *nejmenších čtverců* dokáže pracovat i s vysokou maticí se závislými sloupci, není nutné dělat tu redukci sám ručně. Že je vše nakonec spočítáno pomocí pythonovského `numpy.linalg.lstsq` je sice chvályhodné (z hlediska propagace open source software), ale z hlediska čtenáře zvědavého na tu samotnou analýzu opět naprosto irelevantní a rozptylující.

5. Nakonec, výsledky provedené analýzy jsou vyhodnoceny jako neprůkazné. Jednou větou je v závěru naznačena **existence možnost vylepšení analýzy dat**, avšak byt' i **jen stručný náčrtek možného řešení chybí**. Zpochybňuje tedy výsledek práce platnost toho lineárního modelu, nebo jaký je tedy vlastně závěr?

Další drobné avšak stále platné komentáře k formátu či obsahu (**není vhodné číst u obhajoby**)

- Odkazování na kousky kódu coby na *obrázky* shledávám nevhodným a v běžné literatuře o programování se to nepoužívá (viz například slavné knihy [1] či [2]).
- Před prvním použitím vždy definujte význam zkratek. Takto vlastně v celé práci není vysvětlen význam zkratky SAT či CNF (i když u té druhé si to lze domyslet z českého textu).
- U víceřádkových rovnic jako na straně 15 nevkládejte symbol rovnítko (=) na začátek i konec řádku. Zvyklostí je vkládat ho pouze na začátek nového řádku.
- Absence popisu vertikální osy u obrázků na stranách 27 a 28.

Domnívám se, že i tak však jde o práci dobrou (nikoliv však výbornou či velmi dobrou), a tedy ji hodnotím **DOBŘE (C)**.

#### Literatura

- [1] D. E. Knuth, *The Art of Computer Programming*, 1 edition., 1–4A vols. Amsterdam: Addison-Wesley Professional, 2011.
- [2] B. W. Kernighan and D. M. Ritchie, *The C Programming Language*, 2 edition. Englewood Cliffs, N.J: Prentice Hall, 1988.

.....  
Ing. Zdeněk Hurák, Ph.D.