

I. IDENTIFIKAČNÍ ÚDAJE

Název práce:	Plánování pohybu robotu s 9 stupni volnosti pro svařování plastových nádrží
Jméno autora:	Ján Pravda
Typ práce:	diplomová
Fakulta/ústav:	Fakulta strojní (FS)
Katedra/ústav:	Ústav mechaniky, biomechaniky a mechatroniky
Oponent práce:	Ing. Lukáš Malý
Pracoviště oponenta práce:	ALAD CZ s.r.o.

II. HODNOCENÍ JEDNOTLIVÝCH KRITÉRIÍ

Zadání	mimořádně náročné
<i>Hodnocení náročnosti zadání závěrečné práce.</i>	
<p>Automatické plánování pohybů reálného 9-osého stroje po dané trajektorii v prostoru s danou orientací nástroje je velmi náročný proces. Kromě nalezení vhodné volby 3 redundantních os pomocí optimalizace je třeba v algoritmu správně rozhodnout o volbě konfigurace polohy angulárního 6-osého ramene (4 až 8 základních konfigurací) krát varianty otočení kloubů o $\pm 360^\circ$. S ohledem na koncové limity os a značná omezení rychlostí a zrychlení některých přídavných os je řešení o to složitější, že pro fyzickou proveditelnost je třeba správně naplánovat plynulý průběh pohybů, tak aby nedocházelo ke změně předepsané rychlosti posuvu nástroje při svařování například v blízkosti rohu, kde je nutno změnit orientaci nástroje o 90°. Řešení problému navíc skrývalo přípravu částí softwaru jako G-CODE parser nebo dopočítávání třetí souřadnice orientace nástroje.</p>	

Splnění zadání	splněno
<i>Posuďte, zda předložená závěrečná práce splňuje zadání. V komentáři případně uveďte body zadání, které nebyly zcela splněny, nebo zda je práce oproti zadání rozšířena. Nebylo-li zadání zcela splněno, pokuste se posoudit závažnost, dopady a případně i příčiny jednotlivých nedostatků.</i>	
<p>Bod 1 splněn – prokázáno vytvořením řídicího softwaru. Bod 2 splněn – kolizní model a vizualizační model byl předveden v příloze. Bod 3 splněn – optimalizační algoritmus navržen včetně vysvětlení kritérií váhové funkce. Bod 4 splněn – algoritmus byl do ROS naprogramován. Bod 5 splněn – experimenty v simulátoru předvedeny a byla dle mého názoru velmi objektivně zhodnocena využitelnost v praxi. Práce se v testech funkčnosti nad rámec zadání okrajově zabývá také časovými průběhy souřadnic os a jejich rychlostmi, protože tyto vlastnosti jsou důležité pro samotné svařovací zařízení.</p>	

Zvolený postup řešení	správný
<i>Posuďte, zda student zvolil správný postup nebo metody řešení.</i>	
<p>Optimalizační kritérium v podobě funkce (4.7) se soft limity kloubů angulárního manipulátoru je velmi elegantní řešení, kterým se dá vyhnout nežádoucím změnám konfigurace manipulátoru v průběhu svaru v optimalizačním algoritmu. Optimalizační kritéria se zdají být zvolena logicky, vyvstaly však otázky doporučené níže. Způsob řešení g-code parseru a jeho algoritmů se zdá být vhodný, jen se v budoucnu může narazit na potíž s dopočítáváním třetí souřadnice orientace nástroje v rohu hranaté nádrže. Neměl by to však být velký problém.</p>	

Odborná úroveň	B - velmi dobře
<i>Posuďte úroveň odbornosti závěrečné práce, využití znalostí získaných studiem a z odborné literatury, využití podkladů a dat získaných z praxe.</i>	
<p>V optimalizačním kritériu vystupuje matice možných kolizí, ve které figuruje také botka. Botka je svařovací hrot nástroje, který je při svařování v kolizním kontaktu s výrobkem vždy. Hrot nástroje tedy určitě není zahrnut v optimalizačním kritériu</p>	

kolizní vzdálenosti, resp. v jeho kolizním modelu, jinak by algoritmus nenalezl žádné řešení. Tuto vlastnost bych očekával alespoň zmínit v části optimalizačních kritérií 4.3.1. V samotném softwaru je správně nastavena vypnutá kontrola kolizí mezi botkou a výrobkem.

V (4.2) by mělo být popsáno, že jednotkový vektor k_1 pochází přímo z instrukcí G-CODE vygenerovaného CAM softwarem.

Formální a jazyková úroveň, rozsah práce

B - velmi dobře

Posudte správnost používání formálních zápisů obsažených v práci. Posudte typografickou a jazykovou stránku.

Zvolení anglického jazyka je v tomto případě velmi vhodné vzhledem k rozdílnému rodnému jazyku autora diplomové práce a oponenta.

Algoritmy, transformace souřadnic i optimalizační kritéria jsou popsána srozumitelně.

Obrázky jsou názorné a zřetelně vystihují podstatu.

V algoritmu 1 RRT na straně 6 by bylo dobré alespoň zmínit význam písmen G, V a E.

Občas nějaké chyby, např. str. 9, algoritmus 2, řádek 2: neuzavřená závorka.

Sem tam chybí člen nebo je člen v nesprávném tvaru (např. „a optimization“ místo „an optimization“).

Výběr zdrojů, korektnost citací

A - výborně

Vyjádřete se k aktivitě studenta při získávání a využívání studijních materiálů k řešení závěrečné práce. Charakterizujte výběr pramenů. Posudte, zda student využil všechny relevantní zdroje. Ověřte, zda jsou všechny převzaté prvky řádně odlišeny od vlastních výsledků a úvah, zda nedošlo k porušení citační etiky a zda jsou bibliografické citace úplné a v souladu s citačními zvyklostmi a normami.

U citací by bylo dobré sjednotit označení rozsahu stran. U některých je uvedeno „pages“, u některých pouze čísla stran. Nicméně lze pochopit, že někteří autoři si ve své literatuře citace přímo v tomto znění vyžádali.

Vzhledem k softwarovému charakteru zadání práce je pochopitelné značné množství internetových citací a jsou opodstatněné.

Další komentáře a hodnocení

Vyjádřete se k úrovni dosažených hlavních výsledků závěrečné práce, např. k úrovni teoretických výsledků, nebo k úrovni a funkčnosti technického nebo programového vytvořeného řešení, publikačním výstupům, experimentální zručnosti apod.

Systém ROS je vyvíjen zejména pro systémy Linux. Samotné operační systémy existují v mnoha verzích, také ROS existuje v mnoha verzích. Ani tyto shodné verze však nejsou zárukou, že program v ROS bude napoprvé bezproblémově funkční. I jednotlivé softwarové balíky, jako například open source algoritmus *Ceres Solver* a jeho další závislé balíky (*Eigen*) existují v mnoha různých verzích, které nejsou vzájemně kompatibilní. Po překonání těchto svízelných se nakonec podařilo projekt zkompileovat úspěšně.

Při prvním pokusu o spuštění „*debug.launch*“ došlo k chybě: *terminate called after throwing an instance of 'std::out_of_range' ... what(): vector::_M_range_check: __n (which is 18446744073709551615) >= this->size() (which is 1)*. K této chybě dojde, když se projekt spustí v té verzi a nastavení, jak je vystaven na GitLabu. Příčinou je neošetřená výjimka G-code parseru ve zdrojovém kódu „*libgcode_parser.cpp*“ na řádce 68: „*new_poses->at(i - 1).orientation*“, kde s výchozím nastavením došlo k volání s hodnotou $i=0$, neboť zdrojový G-code „*opt_tester_full_4.nc*“ nebyl očekávaného nového typu. K nesouladu výchozího nastavení mohlo dojít při aktualizaci projektu z lokálního stroje na GitLab.

Ve zdrojovém souboru „*gcode_parser.cpp*“ na řádce 43: „*node_name=argv[1];*“ by měl být nejprve ošetřen počet zadaných argumentů v *argc*, jinak bychom se mohli dočkat nepřekonatelného zhavarování procesu *gcode_parseru* při zavolání omylem bez argumentů.

V závěru 7.2 autor upřímně přiznává problémy s prudkými výchyly rychlostí některých os stroje pocházejícími z plánovacího optimalizačního algoritmu. Správně vyvozuje, že takové pohyby by narušovaly kontinuitu a kvalitu svaru. Rychlostní omezení některé osy by způsobilo zpomalení předepsaného posuvu extrudéru po svaru. Existuje ovšem také možnost, že při některých tvarech svarů nebude možné trajektorii provést jinak, než v rozích svaru zastavit, pootočit výrobek a potom pokračovat dále – tento případ byl zjištěn z praxe.

III. CELKOVÉ HODNOCENÍ, OTÁZKY K OBHAJOBĚ, NÁVRH KLASIFIKACE

Shrňte aspekty závěrečné práce, které nejvíce ovlivnily Vaše celkové hodnocení. Uveďte případné otázky, které by měl student zodpovědět při obhajobě závěrečné práce před komisí.

Diplomová práce je přehledná a srozumitelná. Zabývá se praktickými problémy úlohy, i těmi, které nebyly výslovně vyjmenovány v zadání práce. Vzhledem k aktuální výzkumné fázi vývoje řídicího softwaru svařovacího zařízení je pochopitelné, že se v softwaru ještě sem tam vyskytují neošetřené výjimky vstupních dat a jejich ošetření bude spadat až do následujících etap projektu. Pro tuto část vývoje software svou funkci plní a je možné jej dále rozvíjet.

Navržené doplňující otázky:

1. K bodu 4.3.4 Region I criterion: Dividing line přední a zadní konfigurace odpovídá singulární ose. Takže pokud je správně implementováno dexterity, neměl by přes ni algoritmus nikdy přejít už z tohoto kritéria. Proč by optimalizace nefungovala bez kritéria 4.3.4?
2. K bodu 4.3.6 Previous point distance criterion: Proč nestačilo kritérium rychlostí z 4.3.5?
3. K bodu 4.3.8 No inverse kinematics solution criterion: Jak je zajištěno, aby *wno ikt* učinila váhovou funkci dražší než řešení, které má vysokou cenu, ale je jediné proveditelné? (Mohlo by nastat například při velmi úzkých vnitřních prostorech s malou vzdáleností od kolize.)

Předloženou závěrečnou práci hodnotím klasifikačním stupněm **B - velmi dobře**.

Datum: 31.8.2023

Podpis: