

VŠB – Technická univerzita Ostrava

Fakulta strojní

Katedra robotiky

**SW nástroje pro tvorbu komplexních  
simulačních modelů robotizovaných  
pracovišť a jejich základní  
charakteristiky**

SW Tools for Creation of Complex  
Simulation Models of Robotic  
Workstations and their Basic  
Characteristics

Student:

Radim Bednárík

Osobní číslo

BED0118

Vedoucí bakalářské práce:

Ing. Václav Kryš, Ph.D.

Ostrava 2020

VŠB - Technická univerzita Ostrava  
Fakulta strojní  
Katedra robotiky

## Zadání bakalářské práce

Student: **Radim Bednárík**

Studijní program: B2341 Strojírenství

Studijní obor: 2301R013 Robotika

Téma: SW nástroje pro tvorbu komplexních simulačních modelů robotizovaných pracovišť a jejich základní charakteristiky  
SW Tools for Creation of Complex Simulation Models of Robotic Workstations and their Basic Characteristics

Jazyk vypracování: čeština

Zásady pro vypracování:

1. Zpracujte přehled existujících SW nástrojů pro tvorbu simulačních modelů robotizovaných pracovišť, jejich hlavních funkcionalit a oblastí použití.
2. Analyzujte četnost výskytu odborných publikací k jednotlivým simulačním systémům v publikačních databázích dle pokynů vedoucího.
3. Pro jednotlivé simulační systémy zpracujte seznamy alespoň 3 nejvíce citovaných publikací v analyzovaných publikačních databázích. Seznamy zpracujte formou citací dohledaných publikací.
4. Pro vedoucím určené simulační systémy uveďte příklady jejich použití.
5. Vytvořte soupis potřebných podkladů a parametrů, které je zapotřebí pro vypracování komplexního simulačního modelu robotického systému.
6. Práci též doložte v elektronické podobě ve formátu MS WORD dle pokynů vedoucího.

Seznam doporučené odborné literatury:

ČSN 01 6910 *Úprava písemností psaných strojem nebo zpracovaných textovými editory*. Praha: Český normalizační institut, srpen 2007. 48 s.

ČSN ISO 690 *Informace a dokumentace - Pravidla pro bibliografické odkazy a citace informačních zdrojů*. Praha: Český normalizační institut, 2010.

KREJČÍ, T. *Vytvoření robotizovaného pracoviště v simulačním prostředí RobotExpert*. Diplomová práce (Ing.). VŠB - TU Ostrava, Fakulta strojní, Katedra robotiky. 2019, 122 s.

Web of Science. Dostupné z: [https://apps.webofknowledge.com/WOS\\_GeneralSearch\\_input.do?product=WOS&search\\_mode=GeneralSearch&SID=F4YaFrkU1oZZRbewprc&preferencesSaved=](https://apps.webofknowledge.com/WOS_GeneralSearch_input.do?product=WOS&search_mode=GeneralSearch&SID=F4YaFrkU1oZZRbewprc&preferencesSaved=)

Scopus - Document search. Dostupné z: <https://www.scopus.com/search/form.uri?display=basic>

IEEE Xplore Digital Library. Dostupné z: <https://ieeexplore.ieee.org/Xplore/home.jsp>

ScienceDirect. Dostupné z: <https://www.sciencedirect.com/> <https://www.sciencedirect.com/>

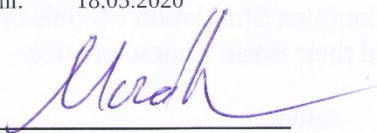
Home – Springer. Dostupné z: <https://link.springer.com/>

Formální náležitosti a rozsah bakalářské práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí bakalářské práce: **Ing. Václav Kryš, Ph.D.**

Datum zadání: 20.12.2019

Datum odevzdání: 18.05.2020



prof. Dr. Ing. Petr Novák  
vedoucí katedry



prof. Ing. Ivo Hlavatý, Ph.D.  
děkan fakulty



Místopřísežné prohlášení studenta

Prohlašuji, že jsem celou bakalářskou práci včetně příloh vypracoval samostatně pod vedením vedoucího bakalářské práce a uvedl jsem všechny použité podklady a literaturu.

V Ostravě dne 18. května 2020.



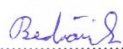
.....

Podpis studenta

Prohlašuji, že:

- jsem si vědom, že na tuto moji závěrečnou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. Zákon o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (dále jen Autorský zákon), zejména § 35 (Užití díla v rámci občanských či náboženských obřadů nebo v rámci úředních akcí pořádaných orgány veřejné správy, v rámci školních představení a užití díla školního) a § 60 (Školní dílo),
- beru na vědomí, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo užít tuto závěrečnou bakalářskou práci nekomerčně ke své vnitřní potřebě (§ 35 odst. 3 Autorského zákona),
- bude-li požadováno, jeden výtisk této bakalářské práce bude uložen u vedoucího práce,
- s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 Autorského zákona,
- užít toto své dílo, nebo poskytnout licenci k jejímu využití, mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše),
- beru na vědomí, že podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů – že tato bakalářská práce bude před obhajobou zveřejněna na pracovišti vedoucího práce a v elektronické podobě uložena a po obhajobě zveřejněna v Ústřední knihovně VŠB-TUO, a to bez ohledu na výsledek její obhajoby.

V Ostravě dne 18. května 2020.



.....

Podpis studenta

### ANOTACE BAKALÁŘSKÉ PRÁCE

BEDNÁRIK, R. SW nástroje pro tvorbu komplexních simulačních modelů robotizovaných pracovišť a jejich základní charakteristiky: bakalářská práce. Ostrava: VŠB – Technická univerzita Ostrava, Fakulta strojní, Katedra robotiky, 2020, 80 s. Vedoucí práce: Krys, V.

Bakalářská práce se zabývá softwarovými nástroji pro tvorbu simulačních modelů robotizovaných pracovišť. V úvodu je vytvořen přehled existujících softwarových nástrojů, které jsou dále porovnávány. Na základě informací z odborných publikací jsou popsány hlavní funkcionality těchto softwarů. Je provedena analýza četnosti výskytu jednotlivých softwarů v publikačních databázích. Dále jsou popsány nejvíce citované publikace a příklady použití jednotlivých softwarů. Poslední kapitola obsahuje přehled potřebných parametrů k vytvoření simulačního modelu robotického systému.

### ANNOTATION OF BACHELOR THESIS

BEDNÁRIK, R. SW Tools for Creation of Complex Simulation Models of Robotic Workstations and their Basic Characteristics: Bachelor Thesis. Ostrava: VŠB – Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Robotics, 2020, 80 p. Thesis head: Krys, V.

Bachelor thesis is dealing with software tools for creation of simulation models of robotics workstations. The introduction provides an overview of existing software tools, which are further compared. Based on information from professional publications, the main functionalities of these software are described. An analysis of the frequency of occurrence of individual software in publication databases is performed. The most cited publications and examples of the use of individual software are then described. The last chapter contains an overview of the necessary parameters to create a simulation model of a robotic system.

## Obsah

	strana
<b>Seznam použitých termínů a zkratk.....</b>	<b>9</b>
<b>Seznam obrázků.....</b>	<b>10</b>
<b>Seznam tabulek.....</b>	<b>12</b>
<b>Úvod.....</b>	<b>13</b>
<b>1 Přehled existujících softwarových nástrojů.....</b>	<b>14</b>
1.1 Základní rozdělení simulačních SW.....	14
1.2 Abecední seznam.....	15
1.3 Podrobná analýza vybraných simulačních SW.....	16
1.4 Přehled hlavních funkcionalit.....	25
1.5 Aplikační oblasti simulačních softwarů.....	31
1.6 Porovnání V-Rep, Gazebo a ARGoS.....	32
1.7 Simulační systémy robotizovaných pracovišť – obecné.....	33
1.7.1 Tecnomatix.....	33
1.7.2 RoboDK.....	35
1.7.3 Visual Components.....	36
1.7.4 3DEXPERIENCE DELMIA Robotics.....	37
1.8 Další doporučené zdroje informací.....	39
<b>2 Analýza četnosti výskytu.....</b>	<b>40</b>
2.1 Průběh analýzy.....	40
2.2 Obecné simulační systémy.....	41
2.3 Simulační systémy robotizovaných pracovišť – obecné.....	44
2.4 Simulační systémy robotizovaných pracovišť – specializované.....	47
<b>3 Nejvíce citované publikace a příklady použití.....</b>	<b>50</b>
3.1 Gazebo.....	50
3.2 V-rep.....	51
3.3 Webots.....	52
3.4 Actin.....	53
3.5 SimSpark.....	53
3.6 MORSE.....	55
3.7 OpenRAVE.....	56
3.8 OpenHRP.....	58
3.9 RobotExpert.....	59

---

3.10	Process Simulate.....	59
3.11	3DEXPERIENCE Delmia Robotics.....	60
3.12	ABB RobotStudio.....	62
3.13	Shrnutí kapitoly .....	64
<b>4</b>	<b>Vypracování komplexního simulačního modelu .....</b>	<b>65</b>
4.1	Analýza prostředí.....	65
4.2	Tvorba 3D modelu.....	66
4.3	Určení momentů setrvačnosti k těžišti .....	66
4.4	Parametry motorů a senzorů .....	68
4.5	Program robotu.....	69
4.6	Shrnutí potřebných parametrů .....	70
4.7	Vliv nastavení simulace na výsledek a rychlost simulace.....	71
	<b>Závěr .....</b>	<b>74</b>
	<b>Zdroje.....</b>	<b>76</b>
	<b>Přílohy.....</b>	<b>80</b>



---

## Seznam použitých termínů a zkratk

- API** (anglická zkratka pro Application Programming Interface) je rozhraní pro programování aplikací.
- AUV** (anglická zkratka Autonomous Underwater Vehicle) je podvodní autonomní vozidlo.
- CAD** (anglická zkratka pro Computer Aided Design) je počítačem podporované navrhování.
- Deep reinforced learning** je učení s využitím principů hlubokého učení a zpětnovazebného učení.
- GPS** (anglická zkratka pro Global Positioning System) je globální polohový systém.
- Emulátor** je software, který umožňuje na počítači spustit software nebo připojit zařízení původně určené pro jiný systém.
- Engine** je označení pro ústřední či kritickou část softwaru.
- IEEE** je zkratka pro Institute of Electrical and Electronics Engineers.
- IMU** (anglická zkratka pro Inertial Measurement Unit) je zařízení pro měření síly, úhlové rychlosti a orientace.
- Line Follower** je robot sledující čáru.
- MAV** (Micro Aerial Vehicle) je třída miniaturních bezpilotních letadel.
- Middleware** je specializovaný software, který usnadňuje vývoj komunikace a vstupů nebo výstupů, je to software spojující softwarové komponenty.
- MM-UAV** (anglická zkratka pro Mobile Manipulating Unmanned Aerial Vehicle) je bezpilotní letadlo určené k manipulaci.
- Open-source** („otevřený software“) je software u kterého byl zveřejněn kód a také licence určující dovolené zacházení s otevřeným kódem.
- Plug-in** je doplňkový modul k jinému softwaru, rozšiřuje jeho funkcionality.
- PR** je zkratka pro průmyslový robot.
- ROS** (Robot Operating System) je soubor softwarových knihoven a nástrojů pro tvorbu robotických systémů.
- ROV** (anglická zkratka Remotely Operated Underwater Vehicle) je podvodní dálkově ovládané vozidlo.
- SW** je zkratka pro software.
- UAV** (Unmanned Aerial Vehicle) je bezpilotní letadlo.
- UGV** (anglická zkratka pro Unmanned Ground Vehicle) je bezpilotní pozemní vozidlo.
- VR** je zkratka pro virtuální realitu.
- WoS** je zkratka pro Web of Science.

## Seznam obrázků

Obr. 1 RobotExpert [10] .....	34
Obr. 2 Pracovní prostředí softwarů RoboDK [13].....	35
Obr. 3 Někteří výrobci podporovaní softwarem RoboDK [12].....	35
Obr. 4 Software Visual Components [15] .....	36
Obr. 5 Software Robotics Arc Engineer [16] .....	37
Obr. 6 Software Robotics Surcafe Engineer [16] .....	38
Obr. 7 Software Robotic Workcell Simulation Engineer [16].....	38
Obr. 8 Graf četnosti výskytu obecných simulačních systémů na webu WoS.....	41
Obr. 9 Graf četnosti výskytu obecných simulačních systémů na webu Scopus .....	42
Obr. 10 Graf četnosti výskytu obecných simulačních systémů na webu IEEE .....	42
Obr. 11 Graf četnosti výskytu obecných simulačních systémů na webu Science Direct 43	
Obr. 12 Graf četnosti výskytu obecných simulačních systémů na webu Springer.....	43
Obr. 13 Graf četnosti výskytu průmyslových simulačních systémů na webu WoS.....	44
Obr. 14 Graf četnosti výskytu průmyslových simulačních systémů na webu Scopus ...	45
Obr. 15 Graf četnosti výskytu průmyslových simulačních systémů na webu IEEE .....	45
Obr. 16 Graf četnosti výskytu průmyslových simulačních systémů na webu Science Direct .....	46
Obr. 17 Graf četnosti výskytu průmyslových simulačních systémů na webu Springer .	46
Obr. 18 Graf četnosti výskytu speciálních simulačních systémů na webu WoS.....	47
Obr. 19 Graf četnosti výskytu speciálních simulačních systémů na webu Scopus .....	48
Obr. 20 Graf četnosti výskytu speciálních simulačních systémů na webu IEEE .....	48
Obr. 21 Graf četnosti výskytu speciálních simulačních systémů na webu Science Direct .....	49
Obr. 22 Graf četnosti výskytu speciálních simulačních systémů na webu Springer .....	49
Obr. 23 Obsáhlá simulace čtyřmotorového UAV pomocí ROS a Gazebo [17] .....	50
Obr. 24 Detekce a mapování rostlin pro zemědělské roboty pomocí 3D LIDAR senzoru [18] .....	51
Obr. 25 Adaptivní kontrola vizuální trajektorie mobilních robotů [20] .....	52
Obr. 26 Vizuálně založené autonomní přistání UAV [21] .....	52
Obr. 27 Experiment s vytahováním tyčinek [22].....	53

---

Obr. 28 Optimalizační přístup pro vývoj chování humanoidních robotů [25] .....	54
Obr. 29 Učení jednoduchých fotbalových dovedností [27] .....	55
Obr. 30 Přehled AUV a robotických simulátorů pro více vozidlové operace [28] .....	56
Obr. 31 Robot využívající cloud a uchopení pomocí „enginu“ pro rozpoznávání objektů [31] .....	57
Obr. 32 Ovládání MM-UAV [32] .....	57
Obr. 33 Vnímání pro mobilní manipulaci [33] .....	58
Obr. 34 Napodobování lidských pohybů humanoidním robotem [35] .....	59
Obr. 35 Spolupráce člověka s robotem [38] .....	60
Obr. 36 Offline programování pro komplexní svařovací systém [36] .....	61
Obr. 37 Energeticky optimální návrh rozvržení robotických pracovních buněk [42] ....	61
Obr. 38 Efekt kinematiky robotu na rovnoměrnost tloušťky povlaku [45] .....	62
Obr. 39 Flexibilní robotická buňka pro dávkování tmelu [46] .....	63
Obr. 40 „Line follower“ .....	65
Obr. 41 Závodní trať pro line-follower .....	66
Obr. 42 Určení polohy těžiště a velikosti momentů setrvačnosti robotu .....	67
Obr. 43 Vložení dynamických parametrů opěrné kuličky .....	67
Obr. 44 Vložení parametrů motoru .....	68
Obr. 45 Vložení parametrů senzorů .....	68
Obr. 46 Vizuální model robotu a hierarchie scény .....	69
Obr. 47 Dynamický model robotu .....	69
Obr. 48 Parametry programu robotu .....	69
Obr. 49 Nastavitelné parametry simulace .....	71
Obr. 50 Enginy používané v softwaru V-Rep (CoppeliaSim) [47] .....	71

---

## Seznam tabulek

Tab. 1 Abecední seznam simulačních softwarů.....	15
Tab. 2 Vývojáři .....	18
Tab. 3 Podporované platformy .....	18
Tab. 4 „Engine“ fyziky .....	19
Tab. 5 „Engine“ 3D renderování .....	20
Tab. 6 3D modelování .....	20
Tab. 7 Formáty 3D modelů.....	21
Tab. 8 Hlavní programovací jazyk .....	22
Tab. 9 Rozšiřitelnost („Pluginy“) .....	23
Tab. 10 Možnost API.....	23
Tab. 11 Možnost off-line programování robotů .....	24
Tab. 12 Uživatelská podpora – část 1 .....	25
Tab. 13 Uživatelská podpora – část 2 .....	26
Tab. 14 Uživatelská podpora – část 3 .....	26
Tab. 15 Podporované roboty – část 1 .....	27
Tab. 16 Podporované roboty – část 2 .....	28
Tab. 17 Podporované roboty – část 3 .....	28
Tab. 18 Podporované senzory – část 1 .....	29
Tab. 19 Podporované senzory – část 2 .....	30
Tab. 20 Podporované senzory – část 3 .....	30
Tab. 21 Nejčastěji využívaný software podle typu robotů .....	31
Tab. 22 Srovnání softwaru pro simulaci podvodních robotů [28].....	55
Tab. 23 Porovnání výsledku a rychlosti simulace .....	72

## Úvod

V mé bakalářské práci se zabývám softwary pro simulace robotů a jejich pracovišť ve virtuálním prostředí. Jedná se o nástroje, které se snaží co nejpřesněji simulovat, jak by se daný robot choval ve skutečnosti. Aplikace těchto softwarů přináší mnoho výhod. Hrají důležitou roli při návrhu, testování, optimalizaci a posuzování bezpečnosti robotů a robotizovaných pracovišť. Umožňují ověření robotizovaného procesu před zahájením výroby a urychlení návrhu systémů, kdy programátor již pracuje v simulačním prostředí, zatímco se konstrukční řešení teprve realizuje. Pomáhají při optimalizaci a zvýšení produktivity, je díky nim možné rychlejší přizpůsobení měnícím se podmínkám [1].

Obsahem práce bude analýza těchto simulačních softwarů (dále SW). V současné době neexistuje mnoho zdrojů informací, které se těmto SW věnují v širším měřítku a porovnávají existující simulační SW mezi sebou.

Na začátku práce se budu věnovat přehledu existujících SW a vybranými informacemi o těchto SW. Vyhledám co největší počet SW, následně dle dostupných informací určím jejich hlavní funkcionality a ty detailněji popíšu. Dále bude následovat analýza četnosti výskytu informací o jednotlivých SW v odborných publikacích. Jejím výstupem budou grafy, které ukážou, zda je daný software dnes používán a relevantní. Poté bude uveden seznam nejvíce citovaných publikací týkajících se simulačních SW a příklady použití jednotlivých simulačních SW, které ukážou nejčastější způsob využití těchto SW. V závěru práce bude uveden soupis podkladů potřebných pro vytvoření simulace robotického systému pro sledování čáry v softwaru V-Rep (CoppeliaSim).

# 1 Přehled existujících softwarových nástrojů

Dnes existuje velké množství těchto nástrojů, popsání každého z nich není v možnostech této bakalářské práce. Proto jsem podle množství dohledatelných informací vybral několik softwarů, kterým se budu v podkapitolách 1.3 a 1.4 věnovat podrobněji. Nicméně i tak v této kapitole uvádím seznam všech, které jsem během psaní této práce dohledal. Pro začátek bych však chtěl uvést rozdělení simulačního softwaru, které používám v pozdějších kapitolách.

## 1.1 Základní rozdělení simulačních SW

Na základě analyzovaných informačních zdrojů jsem zjistil, že simulační SW je možno rozdělit do tří základních skupin:

### Obecné simulační systémy

Tyto softwary umožňují simulace velkého množství typů robotů a prostředí. Oproti ostatním dvěma skupinám zvládají simulaci servisních robotů a jsou nejvíce univerzální. Asi nejnámějšími příklady těchto systémů jsou V-Rep a Gazebo. Jelikož se také používají k simulaci robotizovaných pracovišť, bude v této práci věnována pozornost i této skupině softwarů.

### Simulační systémy robotizovaných pracovišť – obecné

Tato skupina se zaměřuje pouze na simulaci průmyslových robotů a robotizovaných pracovišť. Jedná se o software, který funguje pro většinu průmyslových robotů (dále PR) nebo přímo všechny PR, nezávisle na výrobci. Jmenovitě např. Visual Components, Process Simulate, RoboDK, Robotmaster, nebo 3DEXPERIENCE DELMIA Robotics.

### Simulační systémy robotizovaných pracovišť – specializované

Některé firmy dodávající na trh PR jsou tak velké, že si mohou dovolit vývoj vlastního simulačního softwaru určeného pouze pro roboty jejich výroby. Tento simulační software je lépe přizpůsoben pro dané roboty, ale zároveň je na ně omezen. Jedná se o ABB RobotStudio, Fanuc Roboguide, Kuka.sim apod.

## 1.2 Abecední seznam

Níže uvádím seznam softwarů, které byly v blíže určeném rozsahu použity pro simulace týkající se robotiky. Seznam není kompletní. Nové softwary se neustále vyvíjí a existují i takové, na které jsem při průzkumu nenarazil anebo jsou soukromého typu (narazil jsem alespoň na jeden příklad univerzity, která si vyvinula svůj vlastní).

Tab. 1 Abecední seznam simulačních softwarů

1.	3DEXPERIENCE	30.	Fanuc Roboguide	61.	RobotExpert
2.	3DRealize R	31.	Gazebo	62.	Robotics Toolbox for MATLAB
3.	4DV-Sim	32.	graspIT!	63.	RoboticsLab
4.	ABB RobotStudio	33.	Icub_SIM	64.	Robotmaster
5.	Acrome	34.	Khepera	65.	Robotran
6.	Actin	35.	Klamp't	66.	RobotSim
7.	ADAMS	36.	Kuka.sim	67.	RoboWorks
8.	AirSim	37.	LGSVL	68.	Sim
9.	ANVEL	38.	LpzRobots	69.	Simbad
10.	Anycode	39.	MapleSim	70.	SIMPACK
11.	AnyLogic	40.	miniBloq	71.	SimplyCube
12.	Arboris-Python	41.	Moby	72.	SimRobot
13.	ARGoS	43.	MORSE	73.	SimSpark
14.	AristoSim	44.	MotoSim	74.	Simulink/MATLAB
15.	ARS	45.	MRDS	75.	SL
16.	ARTE	46.	MuJoCo	76.	Stage
17.	Autodesk Inventor	47.	ODE	77.	STDR Simulator
18.	Autolev	49.	OLP Kawasaki	78.	trep
19.	AUTOMAPPS	50.	OpenHRP	79.	UCHILSIM
20.	AX On Desk	51.	OpenRAVE	80.	UWSim
21.	Blender	52.	OpenSIM	81.	Visual Components
22.	breve	53.	ORCA-Sim	82.	Vortex
23.	Bullet	54.	Process Simulate	83.	VoxCad
24.	Carla	55.	RCIS	84.	V-Rep
25.	crrcsim	56.	RoboCell	85.	Webots
26.	Drake	57.	RoboDK	86.	WorkCellSimulator
27.	EASY-ROB	58.	ROBOGUIDE	87.	Workspace
28.	Eurek	59.	RoboLogix	88.	XDE
29.	EZPhysics	60.	roborobo		

### 1.3 Podrobná analýza vybraných simulačních SW

V dotazníku pro „Tools for simulating humanoid robot dynamics: A survey based on user feedback“ [2] dotazovaní uvedli, že je pro ně nejdůležitějším kritériem při volbě simulačního softwaru vhodnost pro daný úkol. Je proto důležité dobře znát podstatu úkolu, ze které pak vyplynou požadavky na vhodný simulační SW. Tato bakalářská práce by měla přispět k pochopení problematiky simulačních SW a poskytnout náhled na současný stav těchto systémů.

Věnovat se každému z uvedených simulačních SW není v možnostech této bakalářské práce. Proto je v následujících tabulkách uveden přehled vybraných systémů. V některých případech existují velké rozdíly mezi jednotlivými SW. Proto některé uvedené v Tab. 1 mohou již být například zastaralé, překonané nebo nemusí být kompatibilní s dnešními počítači. Z tohoto důvodu jsem se zaměřil na menší počet SW, které jsou v dnešní době relevantní.

Během získávání informací o simulačních SW se často opakovaly zmínky o určitých vlastnostech a funkcích. Proto jsem pro vybrané simulační SW vytvořil tabulkový přehled. Pro tuto podkapitolu jsem vybral zástupce z každé skupiny simulačních SW. Níže uvádím krátký seznam vlastností a funkcí simulačních SW, kterým byla v nalezených publikacích věnována pozornost. Seznam není seřazen dle významnosti daného kritéria.

- **Vývojáři** jsou zodpovědní za vývoj SW, rozšiřování funkcionality a podporu SW do budoucna. Definují cenu licence SW, případně ji poskytují zdarma. Některé SW nabízí několik verzí, např. vzdělávací verzi poskytovanou zdarma a komerční placenou verzi. U některých simulačních SW již vývojáři mohou být neaktivní, tím se zastavil další vývoj a podpora daného SW.
- **Uživatelská podpora.** Jedná se o návody, výukové materiály a pomoc s konkrétními problémy. Další částí podpory nezávislou na vývojářích je i komunita a množství informací k danému SW volně dostupných na internetu.
- **Podporované operační systémy a hardwarová náročnost** jsou dalšími znaky, kterými se od sebe jednotlivé SW odlišují. Můžou mít vliv na složitost instalace a další aspekty. V závislosti na hardwarové náročnosti je třeba mít dostatečně výkonný počítač pro dosažení požadované rychlosti simulace.
- **Použité „enginy“** mají vliv na zmiňovanou hardwarovou náročnost, rychlost simulace, přesnost simulace a kvalitu vykreslování.
- **3D modelování a podpora formátů 3D modelů** jsou důležité z hlediska kompatibility s ostatními SW, které se zabývají tvorbou modelů robotů pro simulaci. 3D modelování zabudované přímo v simulačních SW může šetřit čas potřebný k vytvoření simulace. Navíc odpadá závislost na externím SW pro vytvoření 3D modelů.
- **Programovací jazyk.** Simulační SW může podporovat více programovacích jazyků.
- **Rozšiřitelnost** je důležitá, pokud daný SW neobsahuje požadovanou funkci, ale vyhovuje v ostatních vlastnostech. Podpora rozšiřitelnosti umožňuje tuto funkci přidat.



- **Specializace a univerzálnost.** Některé simulační SW se snaží pokrýt co nejširší spektrum podporovaných robotů, prostředí a senzorů, jiné se naopak specializují na konkrétní robot nebo činnost, kterou simulují více do detailů. Podobně tomu je i u možného počtu simulovaných robotů. Některé simulátory se soustředí na jeden detailní model robotu, jiné na simulaci velkého množství zjednodušených modelů.
- **Podporované roboty a senzory.** Různé simulační SW podporují různý rozsah typů robotů a senzorů. Výhodou některých je, že mají vestavěné knihovny předem připravených robotů anebo jsou knihovny dostupné na internetu.

### Značení pro tab. 2 až 11 [3]

Modrá – obecný simulační systém

Oranžová – simulační systém robotizovaných pracovišť – obecný

Červená – simulační systém robotizovaných pracovišť – specializovaný

Zelená – osobně ověřená informace, aktuální v době psaní této bakalářské práce

Bílá – dohledaná informace, ale osobně neověřená, v době vzniku této práce již nemusí být aktuální

Další tabulky jsou uvedeny v příloze Příloha A. Informace, které považuji za ověřené jsem získával z oficiálních webových stránek jednotlivých softwarů nebo z uživatelských fór. Jsou tedy platné v době psaní této práce.

Způsobů, jakými lze porovnávat jednotlivé softwary je celá řada. Jedním z nich může být např. porovnání podle vývojáře v Tab. 2. Některé systémy vyvíjí uzavřená společnost a jsou často placené, některé jsou však tzv. open-source a zdarma. Druhým nejdůležitějším kritériem pro výběr simulátoru bylo podle dotazovaných [2] to, zda se jednalo o open-source software. Jako open-source software označili tvůrci publikace Gazebo, ARGoS, ODE, Bullet, V-Rep, OpenRave, OpenSIM.

Tab. 2 Vývojáři

Software	Vývojáři
V-Rep	Coppelia Robotics
Gazebo	Open Source Robotics Foundation
Webots	Cyberbotics Ltd.
Actin	Energid Technologies
SimSpark	O. Obst, J. Boedecker, J. Murray, M. Rollmann
OpenRAVE	OpenRAVE Community
MORSE	Academic community
OpenHRP	AIST
Process Simulate	Siemens
RobotExpert	Siemens
RoboDK	RoboDK Software S.L., RoboDK Inc.
Robotmaster	Hypertherm
3DEXPERIENCE DELMIA Robotics	Dassault Systemes
Visual Components	Visual Components
ABB RobotStudio	ABB
Fanuc Roboguide	Fanuc
Kuka.sim	KUKA

Další důležitou vlastností simulačních SW je kompatibilita s operačními systémy v Tab. 3. Ačkoliv většina podporuje několik těchto systémů a nejedná se proto o příliš velkou překážku, najdou se takové výjimky jako často zmiňované Gazebo, které podporuje pouze operační systém Linux. V dnešní době však existují tzv. „emulátory“, které dokážou spustit software i na jiných operačních systémech, než pro které byly určeny.

Tab. 3 Podporované platformy

Software	Podporované platformy
V-Rep	Linux, macOS, Windows
Gazebo	Linux (Ubuntu)
Webots	Linux (Ubuntu), macOS, Windows
Actin	Windows, macOS, Linux, RTLinux, VxWorks, RTOS32, RTX64
SimSpark	Windows, macOS, Linux
OpenRAVE	Windows, macOS, Linux
MORSE	Linux, macOS (neoficiálně)
OpenHRP	Windows, Linux
Process Simulate	Windows 7, 8 a 10 64-bit
RobotExpert	Windows 7, 8 a 10 64-bit
RoboDK	Linux, macOS, Windows, Android, iOS, Debian
Robotmaster	Windows 10 64-bit

Software	Podporované platformy
3DEXPERIENCE DELMIA Robotics	Windows 64-bit
Visual Components	Windows 7, 8 a 10 64-bit
ABB RobotStudio	Windows 7 32-bit, Windows 7 a 10 64-bit
Fanuc Roboguide	Windows 32-bit a 64-bit
Kuka.sim	Windows 32-bit a 64-bit

Tzv. „engine“ fyziky je důležitou součástí simulačních SW. Právě tato část softwaru má na starost samotnou simulaci. Použití odlišných „enginů“ může mít za následek jiné hardwarové nároky na simulaci, kratší nebo naopak delší čas simulace a v některých případech i jiné konečné výsledky simulace [1]. Výhodou některých simulačních SW je, že dokážou snadno přepínat mezi jednotlivými „enginy“ a přizpůsobit se tak nejlépe potřebám konkrétní simulace. Používané „enginy“ jsou uvedeny v Tab. 4.

Tab. 4 „Engine“ fyziky

Software	Engine fyziky
V-Rep	Bullet, ODE, Vortex, Newton
Gazebo	ODE, Bullet, Simbody, DART
Webots	Vlastní založený na ODE
Actin	Vlastní
SimSpark	ODE, Bullet, Simbody, DART
OpenRAVE	ODE, Bullet, Simbody, DART
MORSE	Bullet
OpenHRP	ODE, Vlastní
Process Simulate	IPSI
RobotExpert	
RoboDK	Gravity plugin
Robotmaster	
3DEXPERIENCE DELMIA Robotics	IPSI
Visual Components	NVIDIA PhysX
ABB RobotStudio	
Fanuc Roboguide	
Kuka.sim	

Pojem „engine“ se používá i ve spojení s 3D renderováním. Jedná se o další část simulačního SW, která je tentokrát odpovědná za vykreslení simulované scény, stará se tedy o grafický výstup (vykreslování barev, osvětlení, přiřazení textur atd) [4]. Volbou „enginu“ 3D renderování lze ovlivnit kvalitu a rychlost vykreslování, ale také hardwarové nároky.

Tab. 5 „Engine“ 3D renderování

Software	Engine 3D renderování
V-Rep	Vlastní
Gazebo	OGRE (OpenGL)
Webots	OGRE (OpenGL)
Actin	OpenGL
SimSpark	Vlastní
OpenRAVE	Coin3D, OpenSceneGraph
MORSE	Blender
OpenHRP	Java3D
Process Simulate	
RobotExpert	
RoboDK	OpenGL
Robotmaster	
3DEXPERIENCE DELMIA Robotics	
Visual Components	Vlastní
ABB RobotStudio	
Fanuc Roboguide	
Kuka.sim	

Užitečným nástrojem, který může usnadnit a urychlit tvorbu simulace je možnost 3D modelování přímo v simulačním softwaru. Díky tomu odpadá potřeba spoléhat se v tomto ohledu na externí softwary, protože se všechny potřebné 3D modely mohou vytvořit přímo v simulačním SW.

Tab. 6 3D modelování

Software	3D modelování
V-Rep	Integrované
Gazebo	Integrované
Webots	Vlastní
Actin	Vlastní
SimSpark	Žádné
OpenRAVE	Integrovaný
MORSE	Blender
OpenHRP	Integrovaný
Process Simulate	Integrované
RobotExpert	Integrované
RoboDK	Integrované
Robotmaster	Integrované
3DEXPERIENCE DELMIA Robotics	
Visual Components	Ano (ve verzích Professional a Premium)
ABB RobotStudio	

Software	3D modelování
Fanuc Roboguide	
Kuka.sim	

V případě, že se uživatel rozhodne importovat do simulace vlastní předem vytvořené 3D modely, může tak učinit pouze ve formátech, který daný simulační SW podporuje. Mezi nejčastější podporované formáty se podle Tab. 7 řadí hlavně formáty STL a OBJ.

Tab. 7 Formáty 3D modelů

Software	Formáty 3D modelů
V-Rep	OBJ, DXF, STL, COLLADA, URDF
Gazebo	STL, Collada, OBJ
Webots	VRML97 a další
Actin	CAD formáty a plugin pro SolidWorks
SimSpark	
OpenRAVE	collada, obj, iv, vrml, wrl, stl, blend, 3ds, ase,ply, dxf, lwo, lxo, ac, ms3d, x, mesh.xml, irrmesh, irr, nff, off, raw a další
MORSE	
OpenHRP	VRML2.0
Process Simulate	JT modely, CAD formáty, nepodporuje .stp
RobotExpert	JT modely, CAD (NX, Solid Enge, SolidWorks, CATIA, Pro/E, IGES, STEP, Parasolid
RoboDK	STL, STP, IGS, WRML, 3DS, OBJ
Robotmaster	IGES, Parasolid, SAT (ACIS), AutoCAD (DXF, DWG, Inventor TM), SolidWorks, Solid Edge, STEP, EPS, CADL, STL, VDA, ASCII, CATIA, Pro/E, a další
3DEXPERIENCE DELMIA Robotics	
Visual Components	Autodesk, CATIA, Creo, IGES, SolidWorks atd.
ABB RobotStudio	STEP, SAT, Parasolid, SolidWorks, IGES, VDA atd.
Fanuc Roboguide	csb, igs, iges, stl, obj, 3ds, dae, wrl, vrml, rwx, dxf
Kuka.sim	STEP, IGES, CATIA, Creo, Inventor, Siemens Nx atd

Tab. 8 Hlavní programovací jazyk

Software	Hlavní programovací jazyk
V-Rep	LUA
Gazebo	C++
Webots	C++
Actin	C++
SimSpark	C++, Python
OpenRAVE	C++, Python
MORSE	Python
OpenHRP	C++
Process Simulate	
RobotExpert	
RoboDK	Python, C++
Robotmaster	
3DEXPERIENCE DELMIA Robotics	
Visual Components	Python
ABB RobotStudio	RAPID
Fanuc Roboguide	TP, KAREL
Kuka.sim	KRL

Po vložení 3D modelů robotů je třeba naprogramovat jejich chování. K tomu slouží programovací jazyky uvedené v Tab. 8.

V průběhu zjišťování informací o jednotlivých simulačních SW jsem zjistil, že neexistuje univerzální SW, který by zcela vyhovoval ve všech potřebných oblastech. Z toho důvodu může být důležitá vlastnost rozšířit daný SW o další funkce či vlastnosti dle konkrétní potřeby uživatele. Toto řešení bylo využito v několika publikacích v kapitole 3.

Tab. 9 Rozšiřitelnost („Pluginy“)

Software	Rozšiřitelnost (Pluginy)
V-Rep	API, rozšíření, pluginy
Gazebo	Pluginy (C++)
Webots	API, PROTOs, Pluginy (C/C++)
Actin	Pluginy (C++), API
SimSpark	Mods (C++)
OpenRAVE	Pluginy (C++), API
MORSE	Python
OpenHRP	Pluginy (C++), API
Process Simulate	Technomatix .NET API
RobotExpert	
RoboDK	API, Plug-In Interface
Robotmaster	
3DEXPERIENCE DELMIA Robotics	
Visual Components	.NET, Python
ABB RobotStudio	RobotStudio SDK
Fanuc Roboguide	Fanuc PCDK
Kuka.sim	

Tab. 10 Možnost API

Software	Možnosti API
V-Rep	C/C++, Python, Java, Urbi, Matlab/Octave
Gazebo	C++
Webots	C, C++, Python, Java, Matlab, ROS
Actin	C++
SimSpark	Network (sexpr)
OpenRAVE	C/C++, Python, Matlab
MORSE	Python, C/C++, C#, Matlab, a další
OpenHRP	C/C++, Python, Java
Process Simulate	Technomatix .NET API, Java, C#, Tcl
RobotExpert	Technomatix .NET API
RoboDK	Python, C/C++, C#, Matlab, a další
Robotmaster	
3DEXPERIENCE DELMIA Robotics	Vlastní
Visual Components	Python, .NET
ABB RobotStudio	
Fanuc Roboguide	Fanuc PCDK
Kuka.sim	Python, VBA, C++

V závislosti na předmětu a účelu simulace může mít velkou váhu i schopnost off-line programování robotů, Tab. 11. To je výhodou například u PR, kdy se pomocí simulace ověří průběh procesu (bezpečnost, případné kolize atd.) a následně se kód jednoduše nahraje do skutečného robotu.

Tab. 11 Možnost off-line programování robotů

Software	Možnost off-line programování robotů
V-Rep	Ne v základu, pouze pomocí RCS
Gazebo	Ano
Webots	Ano
Actin	Ano
SimSpark	ne
OpenRAVE	ne
MORSE	ne
OpenHRP	ne
Process Simulate	Ano
RobotExpert	ano
RoboDK	ano
Robotmaster	ano
3DEXPERIENCE DELMIA Robotics	Ano, s využitím Robotics Offline Programmer
Visual Components	ano
ABB RobotStudio	ano
Fanuc Roboguide	ano
Kuka.sim	ano



## 1.4 Přehled hlavních funkcionalit

Součástí prvního bodu této bakalářské práce je i přehled hlavních funkcionalit simulačních systémů. V této kapitole porovnávám vybrané simulační SW z Tab. 1. Přestože se jedná, oproti původní tabulce, o relativně malý vzorek, lze z porovnání vyčíst užitečné informace. Příkladem může být v dnešní době rozsáhlá podpora různých typů senzorů anebo omezené možnosti při simulaci podvodních robotů.

### Značení pro tab. 12 až 25 [3]

Modrá – obecný simulační systém

Oranžová – simulační systém robotizovaných pracovišť – obecný

Červená – simulační systém robotizovaných pracovišť – specializovaný

Zelená – obsahuje danou funkci

Červená (sloupce funkcionalit) – neobsahuje danou funkci

Bílá – není známo

Z důvodů odlišného přístupu vývojářů mohou existovat různé rozdíly mezi uživatelskou podporou různých softwarů. Dnes už však všechny mají nějakou formu uživatelské příručky a ty větší z nich uživatelská fóra a detailnější návody i s jednoduchými příklady pro usnadnění výuky.

Tab. 12 Uživatelská podpora – část 1

Software	Seznam kontaktů na vývojáře	API dokumentace
4DV-Sim	ano	-
Actin	-	ano
Gazebo	ano	ano
MORSE	ano	ano
OpenHRP	ano	ano
OpenRAVE	ano	ano
Sim Spark	ano	ano
V-Rep	-	ano
Webots	-	ano
AUTOMAPPPS	ano	ano
Process Simulate	ano	ano
RoboDK	ano	ano
RobotExpert	ano	ano
Visual Components	ano	ano
ABB RobotStudio		ano
Fanuc Roboguide		ano
Kuka.sim		

Tab. 13 Uživatelská podpora – část 2

Software	Fórum nebo jiná forma nápovědy	Uživatelská příručka
4DV-Sim	ano	ano
Actin	-	ano
Gazebo	ano	ano
MORSE	-	ano
OpenHRP	-	ano
OpenRAVE	ano	ano
Sim Spark	-	ano
V-Rep	ano	ano
Webots	ano	ano
AUTOMAPPPS	-	ano
Process Simulate	ano	ano
RoboDK	ano	ano
RobotExpert	ano	ano
Visual Components	ano	ano
ABB RobotStudio	ano	ano
Fanuc Roboguide	ano	ano
Kuka.sim	ano	ano

Tab. 14 Uživatelská podpora – část 3

Software	Sledování problémů	Wikipedie	Chat
4DV-Sim	ano	-	
Actin	ano	-	
Gazebo	ano	ano	
MORSE	ano	-	
OpenHRP	ano	-	
OpenRAVE	ano	ano	
Sim Spark	ano	ano	
V-Rep	ano	-	
Webots	ano	ano	ano
AUTOMAPPPS	ano	-	
Process Simulate	ano	-	
RoboDK	ano	-	
RobotExpert	ano	-	
Visual Components	ano	ano	
ABB RobotStudio		-	
Fanuc Roboguide		-	
Kuka.sim		-	

Jak bylo zmíněno v kapitole 1.1, simulační systémy robotizovaných pracovišť, jako je v Tab. 15, Tab. 16 a Tab. 17 RoboDK, umožňují pouze simulaci robotických ramen. Zbytek softwarů (modře) v těchto tabulkách spadá do kategorie obecných simulačních SW, ale i tak jsou mezi nimi rozdíly v podporovaných typech robotů. Možnost simulace více typů robotů je výhodou, neboť to přispívá k univerzálnosti daného softwaru. Někdy je požadováno, aby se v jedné simulaci vyskytovalo více typů robotů a nebyla omezena pouze na robotizované pracoviště. Konkrétním případem může být simulace v softwaru V-Rep v kapitole 3.2.

Tab. 15 Podporované roboty – část 1

Software	UGV (pozemní mobilní)	UAV (letecké)
4DV-Sim	ano	ano
Actin	ano	ano
Gazebo	ano	ano
MORSE	ano	ano
OpenHRP	ano	-
OpenRAVE	ano	
Sim Spark	ano	-
V-Rep	ano	ano
Webots	ano	ano
AUTOMAPPPS	-	-
Process Simulate	-	-
RoboDK	-	-
RobotExpert	-	-
Visual Components	-	-
ABB RobotStudio	-	-
Fanuc Roboguide	-	-
Kuka.sim	-	-

Zatímco v podstatě všechny obecné simulační SW podporují pozemní mobilní roboty, ne každý dokáže simulovat UAV a co se týče podvodních robotů, existuje ještě méně softwarů schopných se této problematice věnovat. Při průzkumu publikačních databází v podkapitole 3.6 vyšlo najevo, že všechny tři nejvíce citované publikace týkající se SW MORSE pojednávají o podvodních robotech.

Tab. 16 Podporované roboty – část 2

Software	AUV (podvodní)	Robotická ramena
4DV-Sim	-	ano
Actin	ano	ano
Gazebo	ano	ano
MORSE	ano	ano
OpenHRP	-	ano
OpenRAVE		ano
Sim Spark	-	
V-Rep	ano	ano
Webots	ano	ano
AUTOMAPPPS	-	ano
Process Simulate	-	ano
RoboDK	-	ano
RobotExpert	-	ano
Visual Components	-	ano
ABB RobotStudio	-	ano
Fanuc Roboguide	-	ano
Kuka.sim	-	ano

Ačkoliv se simulační systémy robotizovaných pracovišť specializují na robotická ramena, tento typ robotů dokážou simulovat i obecné simulační SW. Tuto možnost nabízely zatím všechny obecné SW, u kterých se mi podařilo tuto informaci dohledat. V některých případech může být k simulaci robotizovaných pracovišť dokonce vhodnější použití obecných simulačních SW. Například pokud se do výrobního procesu zapojuje člověk. SW RoboDK simulaci lidí neumožňuje, zatím co obecné SW často ano. U těchto obecných SW rovněž nebyl větší problém s podporou humanoidních robotů.

Další specializovanou funkcí je detailní simulace uchopování objektů. V průběhu průzkumu v publikačních databázích jsem několikrát narazil na specializovaný SW zabývající se touto problematikou GraspIt! [5]. Využívá se k vyhodnocení uchopení a výpočtu optimálních uchopovacích sil. Příklad použití tohoto SW je uveden v kapitole 3.7.

Tab. 17 Podporované roboty – část 3

Software	Ruce (uchopování)	Humanoidní	Lidé
4DV-Sim		-	ano
Actin	ano	ano	ano
Gazebo	ano	ano	ano
MORSE	-	-	ano
OpenHRP	ano	ano	ano
OpenRAVE	ano	ano	ano
Sim Spark		ano	-
V-Rep	ano	ano	ano
Webots	ano	ano	ano

Software	Ruce (uchopování)	Humanoidní	Lidé
AUTOMAPPPS		-	-
Process Simulate	ano	-	ano
RoboDK	-	-	-
RobotExpert		-	-
Visual Components	ano	-	ano
ABB RobotStudio	ano	-	
Fanuc Roboguide	ano	-	ano
Kuka.sim	ano	-	

V robotice se používá velké množství různých typů senzorů. Z tohoto důvodu vzniká i potřeba simulace těchto senzorů. Z dohledaných informací se ukázalo, že až na výjimky je podpora různých senzorů obecnými simulačními SW téměř bez omezení. Existuje proto velká šance, že daný SW bude obsahovat vestavěnou podporu potřebných senzorů. Podporovaným senzorům se věnují Tab. 18, Tab. 19 a Tab. 20. U simulačních SW robotizovaných pracovišť se mi nepodařilo dohledat typy senzorů dle rozdělení použitého v tabulkách pro obecné SW. Z toho důvodu jsem uvedl typy senzorů, které se mi podařilo vyhledat. Je možné, že tyto SW podporují více senzorů.

Tab. 18 Podporované senzory – část 1

Software	Odometry	IMU	Kolizní	GPS
4DV-Sim	ano	ano	ano	ano
Actin	ano	ano	ano	
Gazebo	ano	ano	ano	ano
MORSE	ano	ano	ano	ano
OpenHRP		ano	ano	
OpenRAVE	ano	ano	ano	ano
Sim Spark	ano	ano	ano	částečně
V-Rep		ano	ano	ano
Webots	ano	ano	ano	ano
AUTOMAPPPS				
Process Simulate	Senzory fotoelektrické, přiblížení, pozice kloubu, vlastností objektu			
RoboDK				
RobotExpert	-	-	-	-
Visual Components	Detekce součástí, Raycast senzor			
ABB RobotStudio	Detekce součástí, Force senzor			
Fanuc Roboguide				
Kuka.sim				

Tab. 19 Podporované senzory – část 2

Software	Monokulární kamery	Stereo kamery	Hlubkové kamery	Všesměrové kamery
4DV-Sim	ano	ano	ano	ano
Actin	ano	ano	ano	ano
Gazebo	ano	ano	ano	ano
MORSE	ano		ano	
OpenHRP	ano		ano	
OpenRAVE	ano	ano	ano	
Sim Spark	ano	částečně		
V-Rep	ano	ano	ano	
Webots	ano	ano	ano	ano
AUTOMAPPPS	ano	ano	ano	
Process Simulate	Senzory fotoelektrické, přiblížení, pozice kloubu, vlastností objektu			
RoboDK		ano	ano	ano
RobotExpert	-	-	-	-
Visual Components	Detekce součástí, Raycast senzor			
ABB RobotStudio	Detekce součástí, Force senzor			
Fanuc Roboguide				
Kuka.sim				

Tab. 20 Podporované senzory – část 3

Software	2D laserové skenery	3D laserové skenery
4DV-Sim	ano	ano
Actin	ano	ano
Gazebo	ano	ano
MORSE	ano	ano
OpenHRP	ano	ano
OpenRAVE	ano	ano
Sim Spark	-	-
V-Rep	ano	ano
Webots	ano	ano
AUTOMAPPPS	ano	ano
Process Simulate		
RoboDK	ano	ano
RobotExpert	-	-
Visual Components		
ABB RobotStudio		
Fanuc Roboguide		
Kuka.sim		

## 1.5 Aplikační oblasti simulačních softwarů

Tato podkapitola se věnuje výsledkům průzkumu obsaženého v publikaci z roku 2014, „Tools for simulating humanoid robot dynamics: A survey based on user feedback“[2]. Týkala se 119 lidí z různých zemí, pracujících v oboru robotiky, 70% z nich na univerzitách. Hlavním předmětem byly dotazy týkající se humanoidních systémů, nicméně obsahuje dostatek informací i o dalších typech robotů a její přečtení doporučuji všem, kteří si chtějí udělat počítačový přehled o různých simulačních systémech.

Následující tabulka obsahuje nejčastěji simulované typy robotů a nejčastěji používaný (seřazený sestupně) software pro daný typ robotu. Sloupec nepoužívanější software obsahuje systémy, které použil více než jeden dotazovaný, sloupec další použitý software pak systémy, které použil alespoň jeden dotazovaný.

Tab. 21 Nejčastěji využívaný software podle typu robotů

Typ robotů	Nejpoužívanější software	Další použitý software
<b>Humanoidní</b>	ODE, Gazebo, Robotran, OpenRave, Arboris-Python, XDE, Icub_SIM	Drake, MapleSim, MuJoCo, OpenSIM, RoboticsLab, SL, Vortex, V-Rep, Webots, vlastní
<b>Mobilní</b>	Gazebo, ARGoS, Webots, V-Rep, Vortex	ADAMS, Autodesk Inventor, Bullet, ODE, Morse, roborobo, Sim, vlastní
<b>Vícenohý</b>	Webots, ODE	Gazebo, ADAMS, Autolev, Bullet, Moby, RoboticsLab, SIMPACK, VxCad
<b>Servisní</b>	Gazebo, OpenRave	OpenSIM, V-Rep, Morse, RCIS, SL
<b>Létající</b>	ARGoS	Robotran, crrcsim, Gazebo, Simulink/Matlab
<b>Roj</b>	ARGoS	roborobo
<b>Kolové</b>	Gazebo, V-Rep	ARGoS, Morse, Webots, Vortex, Autodesk, Matlab/Simulink, Adams, trep, XDE, SIMPACK, Autolev, RCIS, Bullet, RoboticLab, vlastní

## 1.6 Porovnání V-Rep, Gazebo a ARGoS

Jako další užitečný zdroj informací se ukázala publikace „Feature and Performance Comparison of the V-REP, Gazebo and ARGoS Robot Simulators“ [6]. Porovnává tři zmíněné simulační systémy hned z několika hledisek, jako jsou např. hardwarové nároky a jak dobře softwary zvládají simulovat vyšší počet robotů nebo náročnější prostředí. Podává tak přehled různých vlastností a funkcionalit, ve kterých se simulační softwary mohou lišit.

Jedním z nejvíce používaných softwarů je V-Rep od společnosti Coppelia Robotics. V porovnání s ostatními simulátory je nejsložitější a nejnáročnější na výpočetní výkon počítače. Nabízí však několik „enginů“ fyziky, velkou knihovnu modelů robotů, schopnost uživatele interagovat s prostředím během simulace (přemísťování objektů, robotů atd.), úpravu síťových modelů a jejich optimalizaci. Další dobrou vlastností je využívání více jader procesoru počítače. Je vhodný pro přesnou simulaci dějů jako přeprava objektu, sledování oblasti, ale také průmyslové aplikace, kde je třeba, aby několik robotů pracovalo současně. Jednou z nevýhod je, že používá vlastní formát ukládání simulací a lze je proto upravovat pouze ve V-Repu. Oproti ostatním simulátorům se jeví jako nejvíce univerzální. Jeho nevýhodu v potřebě většího výkonu počítače lze částečně vyrušit použitím vhodných parametrů simulace a optimalizací 3D modelů. Je možné simulovat více robotů najednou, avšak při složitých modelech a příliš vysokém počtu robotů se simulace značně zpomaluje. To platí obecně pro všechny systémy, ovšem některé jsou na tento problém lépe optimalizovány. Pokud je předmětem simulace chování velkého počtu robotů, může být výhodnější použít jiný systém, než V-Rep.

ARGoS je simulátor primárně navržený pro velké skupiny robotů (robotické roje). Je rozšiřitelný pomocí různých „plug-inů“. Oproti V-Repu má zjednodušené roboty, prostředí i složitost fyziky. Díky tomu však dosahuje lepšího výkonu. Další výhodou oproti V-Repu je, že nastavení simulace se ukládá do souboru XML, který sdílí i některé další softwary a lze s ním pracovat odděleně. Mezi nevýhody patří chybějící možnost vložení 3D síťových modelů. Software je vhodný pro velké počty robotů a malou velikost prostředí, pro velké prostředí je vhodnější použít Gazebo.

Gazebo je nejbližší alternativou k V-Repu. Je velmi podobný, co se týče funkcí a rozhraní. Jeho knihovna robotů je však menší a modely jsou jednodušší, srovnatelné se softwarem ARGoS. V provedených simulacích při 50 a více robotech a velkém prostředí dosahoval lepších výsledků (rychlejší simulace při nižším potřebném výkonu), než ARGoS. Je zde sice možnost vložení 3D síťových modelů, ale nelze je nijak upravit. Nevýhodou je špatné rozhraní a problémová rozšiřitelnost.



## 1.7 Simulační systémy robotizovaných pracovišť – obecné

V této kapitole jsou dále popsány obecné simulační systémy robotizovaných pracovišť, kterým se věnuji v tabulkových přehledech a jsou součástí analýzy četnosti výskytu v odborných publikacích.

### 1.7.1 Tecnomatix

Tato podkapitola se věnuje skupině softwarů Tecnomatix. Jedná se o softwary určené k digitalizaci výroby, tvorbě digitálních dvojčat robotických buněk atd. Jmenovitě jsou to tyto softwary:

- Plant Simulation
- Process Designer
- RobotExpert
- Process Simulate Robotics
- Robcad
- Process Simulate Human
- Jack

Z pohledu robotiky jsou nejvíce zajímavé softwary RobotExpert a Process Simulate Robotics. Oba softwary slouží k simulacím robotizovaných pracovišť. Těmito nástroji se zabývají dvě diplomové práce vypracované dříve na Katedře robotiky [10][11]. Z tohoto důvodu zde uvedu stručný popis těchto softwarů.

RobotExpert je kompaktní nástroj určený k off-line programování průmyslových robotů v 3D prostředí. Podporuje řídicí jednotky pro roboty ABB, Fanuc, Kuka atd. Další roboty je možno programovat pomocí obecného kontroléru. Software podporuje definování libovolného počtu externích os, buňky s více roboty, režim master-slave, možnost výměny nástrojů během programu. Umožňuje připravit robotické buňky na technologické operace, manipulaci, montáž, lakování, leštění, nanášení lepicí pasty, svařování, frézování atd. Mezi základní funkce tohoto softwaru patří:

- 3D modelování kinematiky nástrojů, periférií a robotů,
- podpora robotů ze široké škály výrobců,
- tvorba a export robotických procesů a drah, včetně logiky
- detekce kolize,
- výpis operací pomocí Ganttova grafu,
- interakce s kontrolérem robotu, off-line programování robotů,
- výpočet přesného času cyklu pomocí realistických simulací robotů
- přizpůsobivé rozhraní a funkčnost pro obsluhu
- možnost doplnění knihoven robotů



Obr. 1 RobotExpert [10]

Process Simulate je komplexním prostředím pro simulaci a optimalizaci výrobních procesů. Je možné simulovat funkce jednotlivých pracovišť, ale i celých výrobních linek. Umožňuje návrh a simulaci složitých zón, kdy v blízkosti pracuje několik robotů, návrh pracovních cyklů bez kolizí a optimalizaci cyklických časů, simulaci práce člověka a optimalizaci z hlediska ergonomie. Nabízí možnost současné spolupráce více uživatelů na tvorbě simulace. Software obsahuje tyto funkcionality:

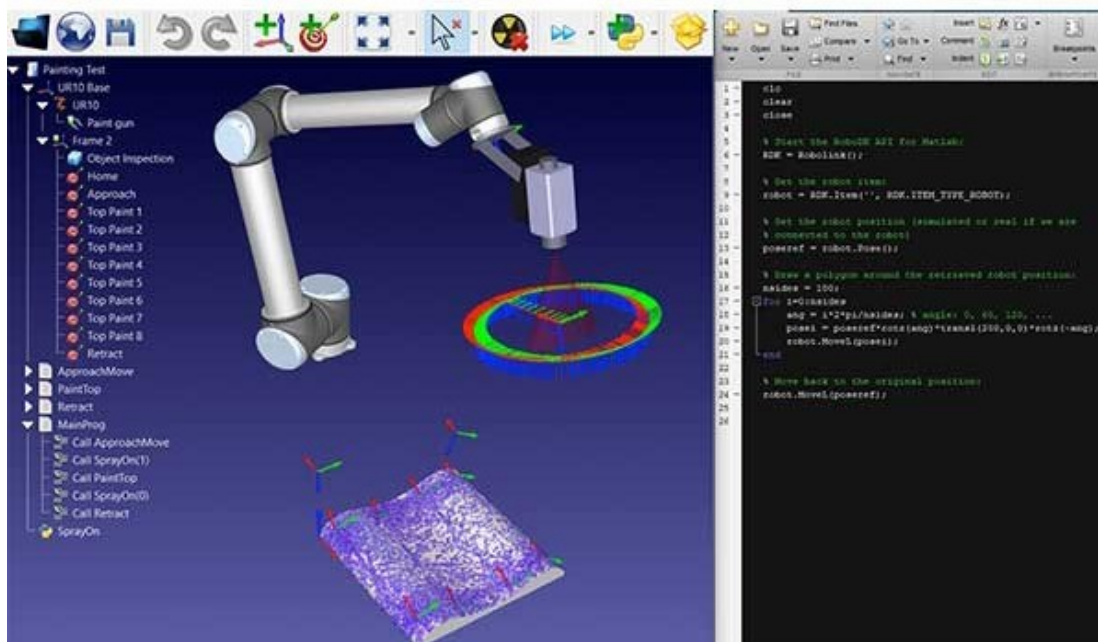
- 3D simulace,
- statická a dynamická detekce kolizí,
- 3D měření,
- montáž a plánování robotických cest,
- návrh linek a pracovních stanic,
- dokumentační nástroje,
- simulace lidských úkolů,
- diskrétní a kontinuální simulaci,
- virtuální uvedení do provozu.

Při porovnání těchto dvou softwarů se dá říct, že Process Simulate je RobotExpert rozšířený o další funkce. RobotExpert postačí na robotickou buňku, zatímco pomocí Process Simulate lze simulovat celé výrobní haly. Studie vytvořené v těchto softwarech nejsou vzájemně kompatibilní. Přehled nejdůležitějších funkcí, které RobotExpert oproti Process Simulate nepodporuje:

- bodové svařování
- senzory a logické bloky
- propojení s PLC zařízením
- simulace řízená událostmi
- načtení mračen bodů
- simulace člověka, ergonomie
- virtuální realita

## 1.7.2 RoboDK

Dalším zástupcem obecných simulačních systémů robotizovaných pracovišť je RoboDK. Tento software je dostupný ve třech verzích, „Trial“, „Educational“ a „Professional“. [12]



Obr. 2 Pracovní prostředí softwarů RoboDK [13]

Výhodou tohoto SW je knihovna s více než 500 průmyslovými roboty, otočnými stoly, lineárním vedením a dalšími systémy. Seznam podporovaných zařízení je uveden na webových stránkách softwaru [12].



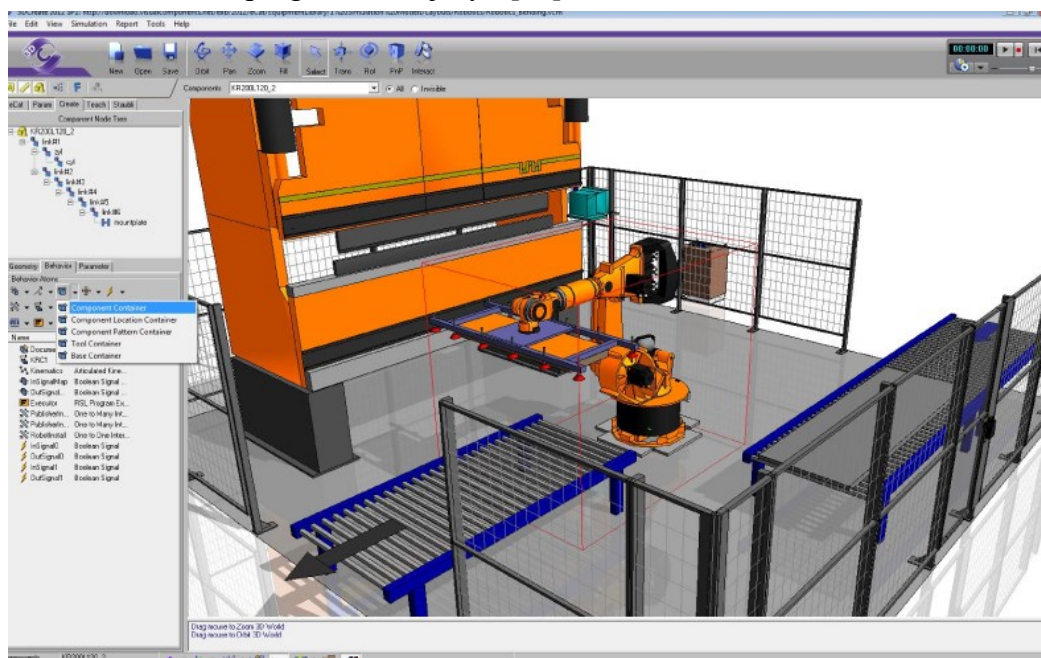
Obr. 3 Někteří výrobci podporovaní softwarem RoboDK [12]

V následujícím odstavci stručně popíšu vytvoření programu pro robot pomocí tohoto SW. Po vložení robotu z knihovny SW umožňuje vložit vlastní nástroj pomocí vložení 3D modelu a následným definováním parametrů nástroje. Mezi podporované formáty 3D modelů patří STL, STEP a IGES. Podobným způsobem lze vložit i model vyráběné součásti. Následně se provádí programování dráhy nástroje, k tomu lze využít „plug-inů“ pro SolidWorks, Mastercam, Fusion 360 a další CAD a CAM softwary (tzv. „CAD to path“). SW dokáže vytvořit dráhu nástroje vyhýbající se bodům singularity, limitům jednotlivých

os a kolizím. Po vytvoření simulace lze vygenerovat program pro roboty více než 40 výrobců. Pro off-line programování robotu tedy uživatel nemusí mít teoreticky žádné znalosti týkající se psaní kódu. SW kromě tohoto postupu umožňuje i programování robotu pomocí API v preferovaném programovacím jazyku: Python, C#, Visual Basic, C++ a Matlab. SW nabízí možnost simulace obrábění, svařování, řezání, odjehlování, malování, manipulace s objekty, 3D tisku atd. SW lze využít pro kalibraci robotu pro zvýšení přesnosti. Podobně jako softwary Tecnomatix umožňuje spočítat a optimalizovat čas pracovního cyklu.

### 1.7.3 Visual Components

Obdobným softwarem je i Visual Components, rovněž dostupný ve třech verzích, „Essentials“, „Professional“ a „Premium“. Rozdíly jsou v podpoře 3D modelování komponent a v pokročilých funkcích. Poslední verze podporuje VR technologii. Software pro simulaci fyzikálních vlastností a zákonů využívá NVIDIA PhysX engine [14]. SW je modulární a postavený na platformě .NET, je k dispozici také Python API. Podobně jako ostatní SW pro simulaci robotizovaných pracovišť má vestavěnou knihovnu modelů různých PR, momentálně jsou dostupné modely PR od 33 výrobců. SW umožňuje i tvorbu vlastního modelu robotu. Roboty se programují pomocí programovacího jazyku RSL (Robot Sequence Language), je zde možnost použití i nativního programovacího jazyku nebo převodu RSL na nativní programovací jazyk [15].

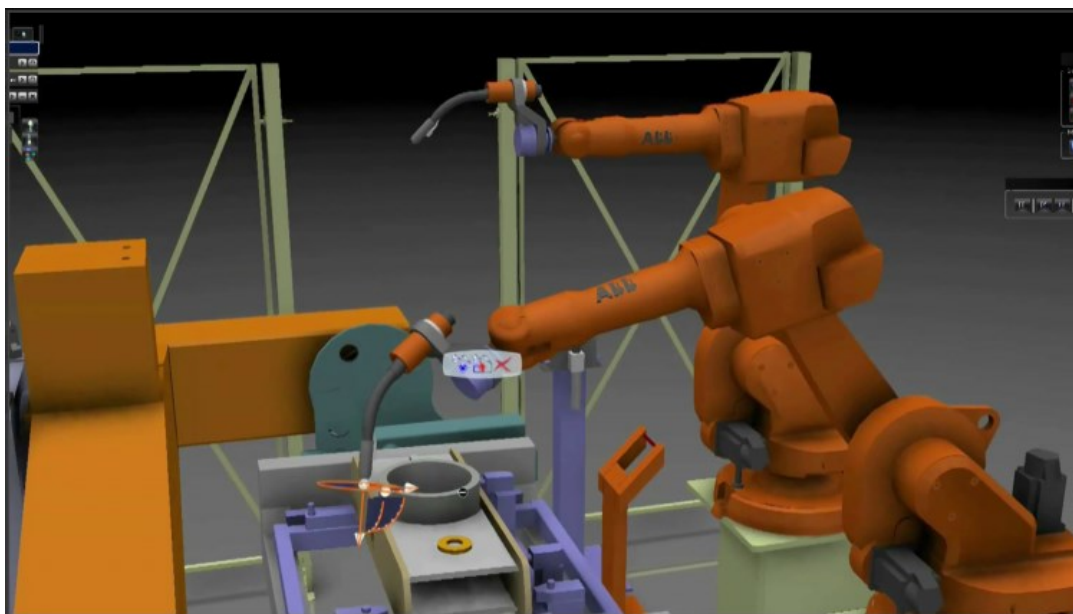


Obr. 4 Software Visual Components [15]

### 1.7.4 3DEXPERIENCE DELMIA Robotics

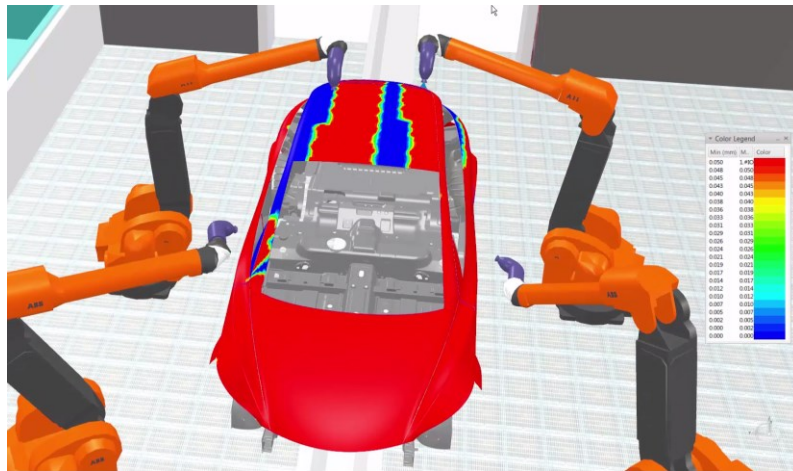
3DEXPERIENCE DELMIA Robotics je skupina SW, podobně jako Tecnomatix, rozdělená dle různých specializací [16]. Těchto SW je dohromady 15:

- **Robotics Optimization Expert** – zabývá se výpočtem a optimalizací dráhy PR. Lze vypočítat nejrychlejší a nejekonomičtější dráhu.
- **Robotics Arc Engineer** – simuluje obloukové svařování s využitím PR. Automaticky navrhuje optimální pozice kloubů z hlediska svařovacího procesu.



Obr. 5 Software Robotics Arc Engineer [16]

- **Robotic Engineer** – simuluje PR při manipulačních úlohách. Řeší možné kolize, pokud je v buňce více PR. Obsahuje vestavěnou knihovnu PR. Optimalizuje pohyb PR.
- **Robotics Surface Engineer** – simulace povrchových procesů s využitím PR (např. nanášení barvy).
- **Robotics Offline Programmer** – řeší import a export programu PR. Podporuje PR výrobců Fanuc, ABB, Kuka atd.
- **Robotics Spot Engineer** – zabývá se návrhem a simulací robotických pracovišť určených k bodovému svařování, vrtání a nýtování. Umožňuje simulaci koordinace několika PR.
- **Virtual Commissioning Expert** – simulace chování robotu s využitím nativního softwaru controlleru.
- **Robotic Workcell Simulation Engineer** – umožňuje návrh robotických buněk, software funguje skrze technologii cloud.



Obr. 6 Software Robotics Surcafe Engineer [16]

- **Assembly Riveting System Programmer for Aerospace** – programování, simulace a ověřování komplexního procesu vrtání a nýtování s využitím PR v kontextu „3D Virtual Factory“ („Virtual Factory“ je další skupina SW vyvíjených tímto vývojářem).
- **Assembly Robot Programmer** – programování, simulace a ověřování montážních systémů s využitím PR v kontextu „3D Virtual Factory“.
- **Body-in-White Robot Programmer** – Definování, ověřování a off-line programování „body-in-white“ systémů ve „Virtual Factory“.
- **Body-in-White Robotics Engineer** – ověřování a optimalizace „body-in-white“ robotických procesů v kontextu „Virtual Factory“.
- **Fabrication Robot Programmer** – Definování, ověřování a off-line programování PR pro svařování a nanášení tmelu ve „Virtual Factory“.
- **Fabrication Robot Workcell Programmer for Industrial Equipment** – Ověřování a optimalizace obloukového svařování a nanášení tmelu s využitím PR.
- **Surface Robot Programmer** – definování, ověřování a off-line programování nanášení barvy s využitím PR ve „Virtual Factory“.



Obr. 7 Software Robotic Workcell Simulation Engineer [16]

## 1.8 Další doporučené zdroje informací

V této podkapitole uvedu některé další publikace a zdroje informací, které dále ve větších detailech popisují simulační softwary. Některé z nich jsem použil při tvorbě této bakalářské práce, na jiné jsem narazil v průběhu vypracování a obsahují detailnější informace o vybraných tématech zmíněných v této bakalářské práci. Jelikož zde nebyl prostor věnovat se této problematice hlouběji, doporučuji tyto publikace a zdroje k případnému dalšímu doplnění informací o simulačních softwarech.

- **Feature and Performance Comparison of the V-REP, Gazebo and ARGoS Robot Simulators“ [6].** Publikace porovnávající V-Rep, Gazebo a ARGoS.
- **Tools for simulating humanoid robot dynamics: A survey based on user feedback [2].** Publikace porovnávající simulační softwary z mnoha hledisek, softwary jsou v ní seřazeny podle hodnocení uživatelů.
- **Simulation tools for model-based robotics: Comparison of Bullet, Havok, MuJoCo, ODE and PhysX [8].** Publikace porovnávající jednotlivé „enginy“ simulačních softwarů.
- **Survey of Popular Robotics Simulators, Frameworks, and Toolkits [7].** Publikace porovnávající nástroje pro simulace robotických systémů.
- **V-REP: A versatile and scalable robot simulation framework [1].** Publikace zabývající se detailněji simulačním softwarem V-Rep.
- **Design and use paradigms for Gazebo, an open-source multi-robot simulator [9].** Nejvíce citovaná publikace týkající se softwaru Gazebo, vzhledem k jejímu stáří však už některé informace nemusí být aktuální.
- **Vytvoření robotizovaného pracoviště v simulačním prostředí RobotExpert [10].** Tato diplomová práce se detailněji zabývá softwarem RobotExpert.
- **Tvorba digitálního dvojčete robotické buňky v simulačním prostředí Tecnomatix [11].** Diplomová práce zabývající se podrobněji nástroji Tecnomatix.

## 2 Analýza četnosti výskytu

Druhým cílem této práce je analýza četnosti výskytu odborných publikací k jednotlivým simulačním systémům v publikačních databázích. Pro analýzu byly vybrány publikační databáze WoS, Scopus, IEEE, Science Direct a Springer. Pro větší přehlednost jsem výsledné hodnoty zpracoval do grafů a dodržel původní rozdělení na skupiny systémů. Tyto skupiny jsou podrobněji popsány v kapitole 1.1.

V roce 2011 byl vytvořen přehled populárních robotických simulátorů [7]. Výsledkem tohoto přehledu byl závěr, že většina simulátorů v průzkumu byla zaměřena na specifické roboty a senzory, které byly příliš drahé a nepříliš užitečné pro levnější systémy. Tato složitost a vysoká cena simulačních SW jim bránila v masovém rozšíření mezi menší projekty. Kód vyvinutý v těchto simulátorech často vyžadoval drahý hardware pro přenos do reálných robotických systémů. Chyběly levné simulátory, které by dokázaly simulovat jednodušší senzory.

Ve výstupu z mnou provedeného průzkumu publikačních databází lze vidět, že právě okolo roku 2011 začalo využití některých simulačních SW prudce růst. Nejvíce je to patrné na SW Gazebo a V-Rep, které odpovídají popisu chybějících simulátorů ve zmíněné publikaci [7]. Dalo by se tedy říct, že tyto simulátory naplnily potřebu levných, jednoduchých a univerzálních simulačních SW.

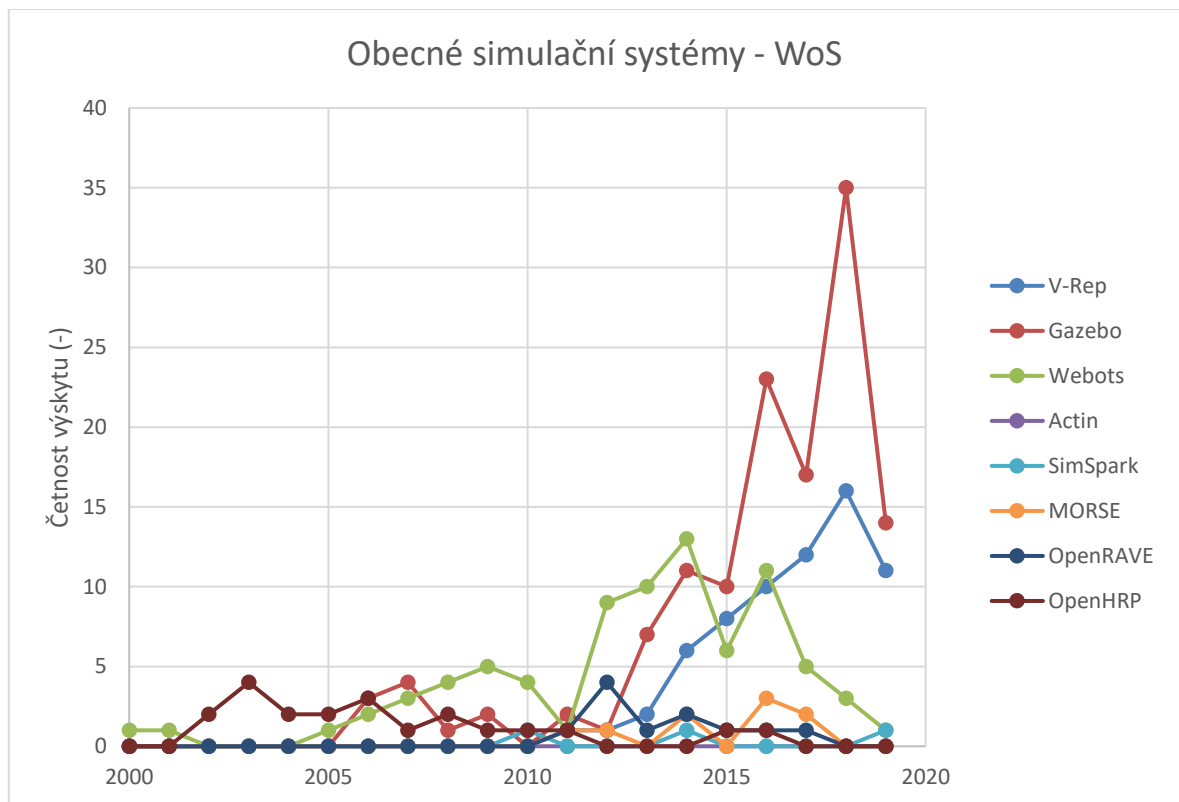
### 2.1 Průběh analýzy

Pro vyhledání informací jsem použil názvy jednotlivých softwarů, v některých případech bylo nutné použít k vyhledání jiná klíčová slova, neboť název softwaru odkazoval na jiné, s tématem nesouvisející publikace. Pro nalezení výsledků byla použita kategorie „ROBOTICS“ na webu WoS (a odpovídající kategorie v dalších databázích), která byla v tomto případě dostatečně rozsáhlá, aby poskytla dobrý přehled o vybraných simulačních systémech. Jednotlivé četnosti jsem zaznamenal do tabulek v Excelu. Tyto tabulky jsou uvedeny v příloze Příloha A. Je třeba upozornit na použití rozdílného měřítka na svislé ose u jednotlivých grafů. Z tohoto důvodu jsem vytvořil i grafy porovnávající četnost výskytu skupin softwarů v odborných publikacích, tyto grafy jsou uvedené v příloze.

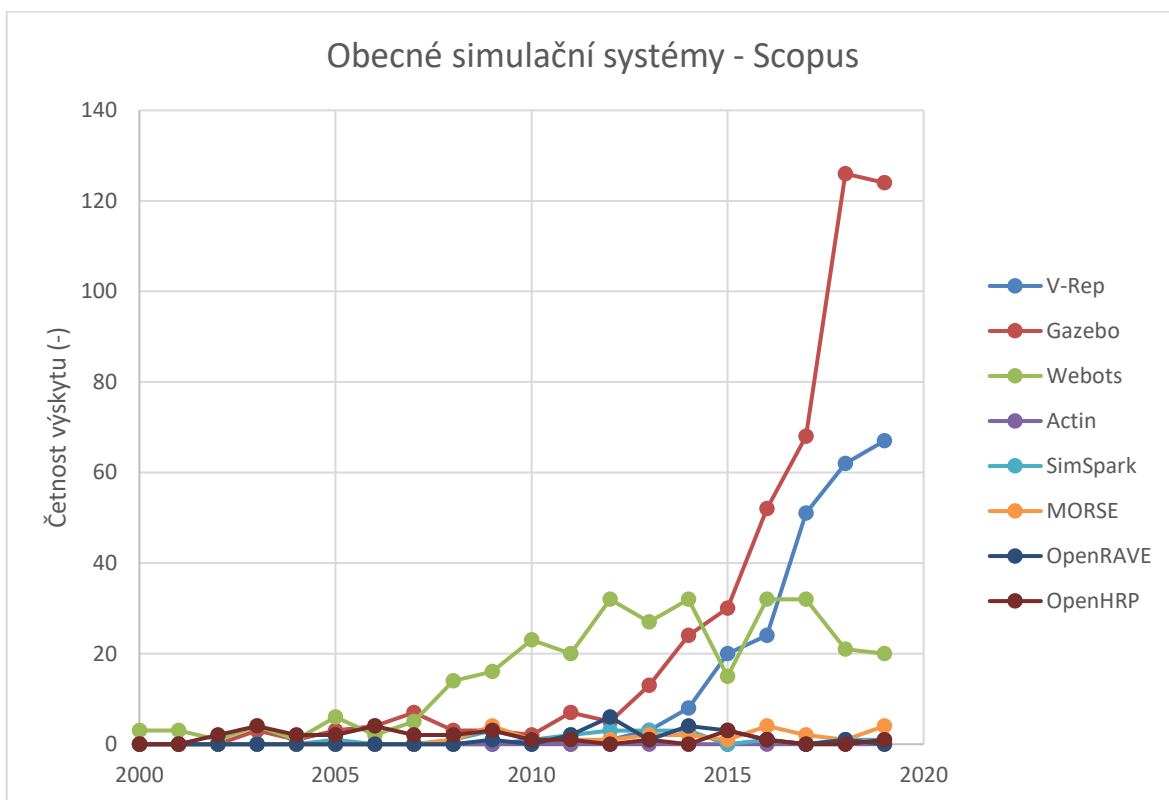


## 2.2 Obecné simulační systémy

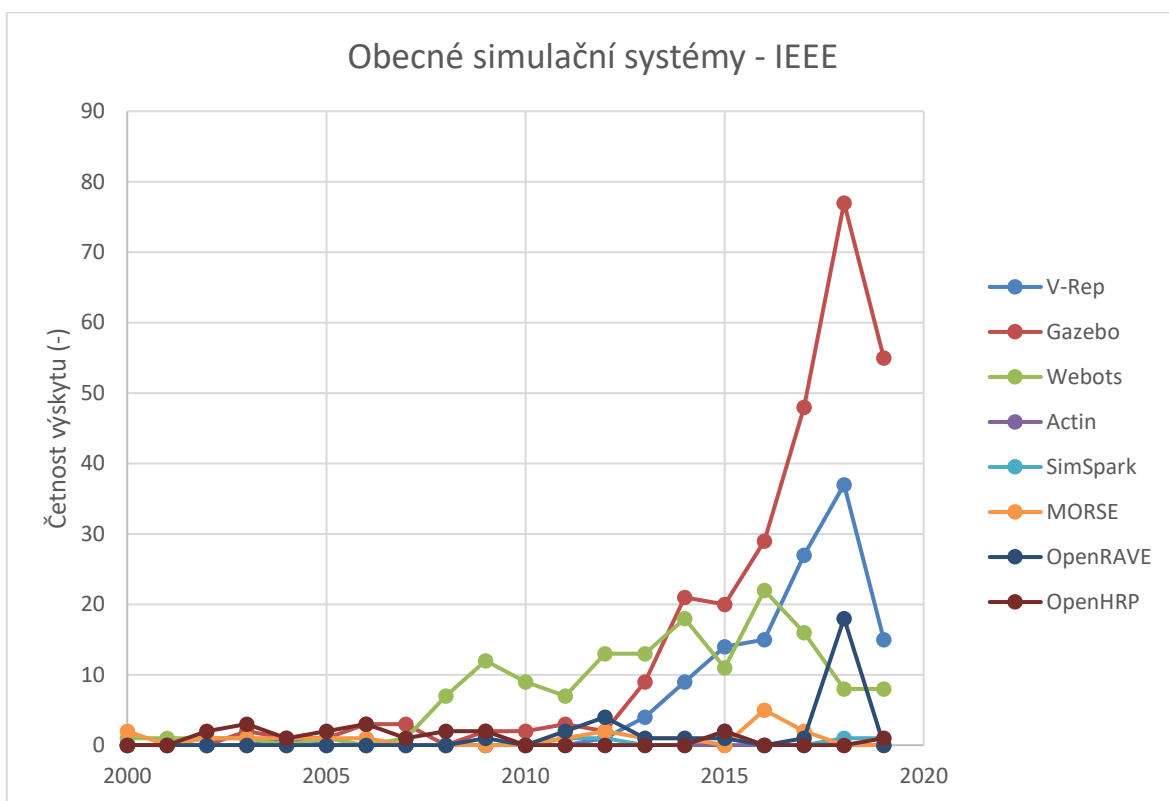
Tato podkapitola se věnuje výskytu obecných simulačních systémů v zadaných publikačních databázích. Z výsledný grafů lze vyčíst několik informací. Tou první je, že výsledky z jednotlivých databází jsou si velmi podobné. Ve všech případech s nárůstem dominuje Gazebo a jako druhý se vždy umístil V-Rep. Třetí byl, kromě jednoho případu, Webots. Zbytek systémů za výše zmíněnými značně zaostává. Rovněž lze dobře vidět, že trend využívání simulačních systémů v posledních letech exponenciálně roste a že se mezi všemi systémy začínají objevovat favoriti.



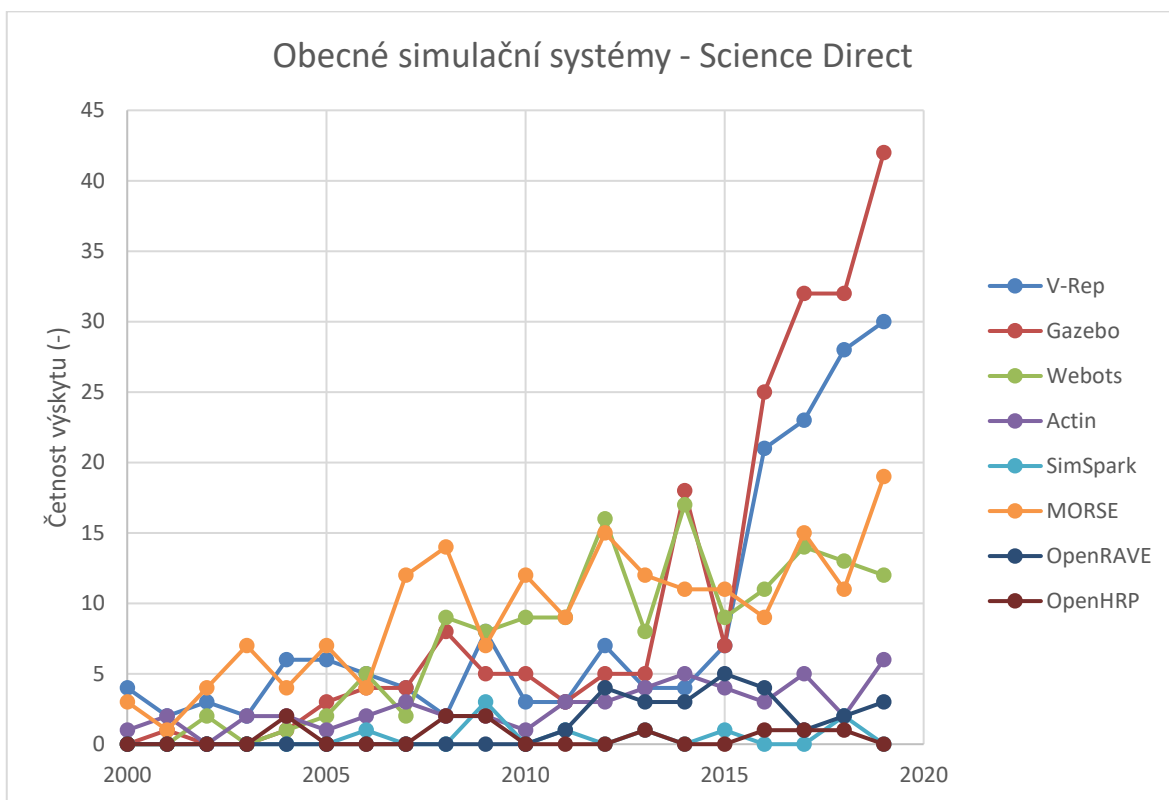
Obr. 8 Graf četnosti výskytu obecných simulačních systémů na webu WoS



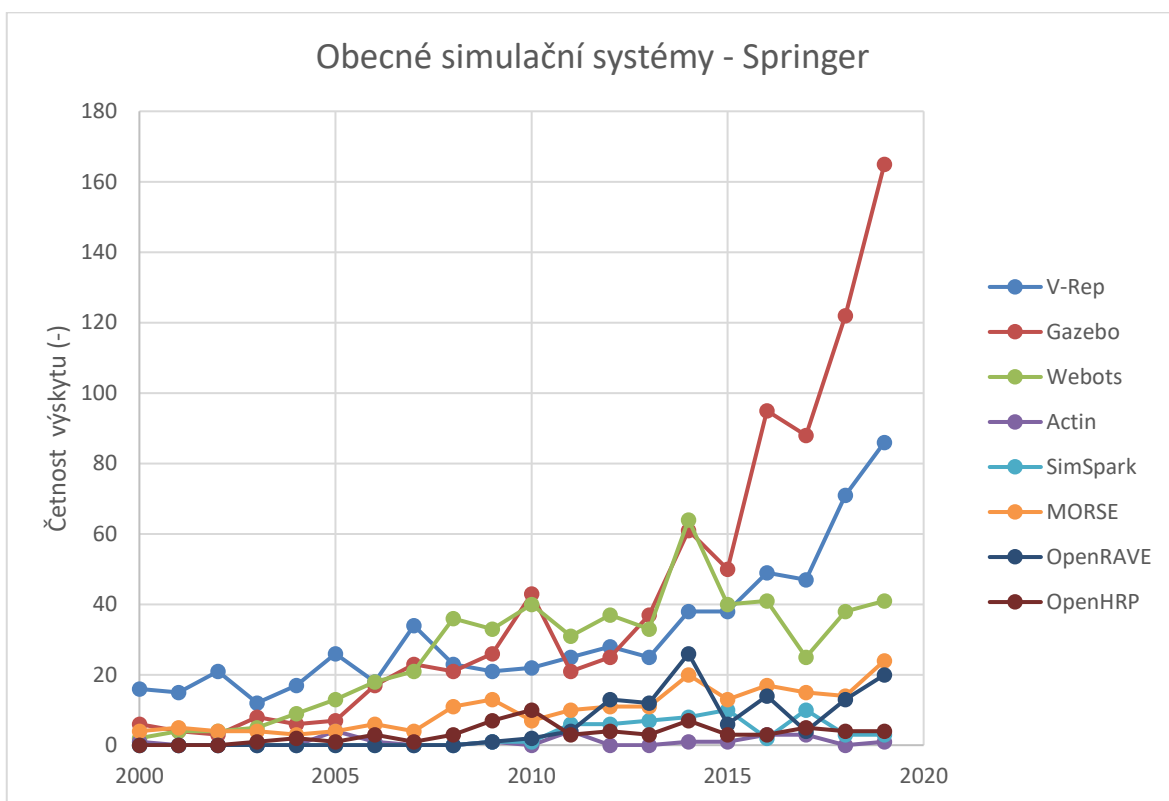
Obr. 9 Graf četnosti výskytu obecných simulačních systémů na webu Scopus



Obr. 10 Graf četnosti výskytu obecných simulačních systémů na webu IEEE



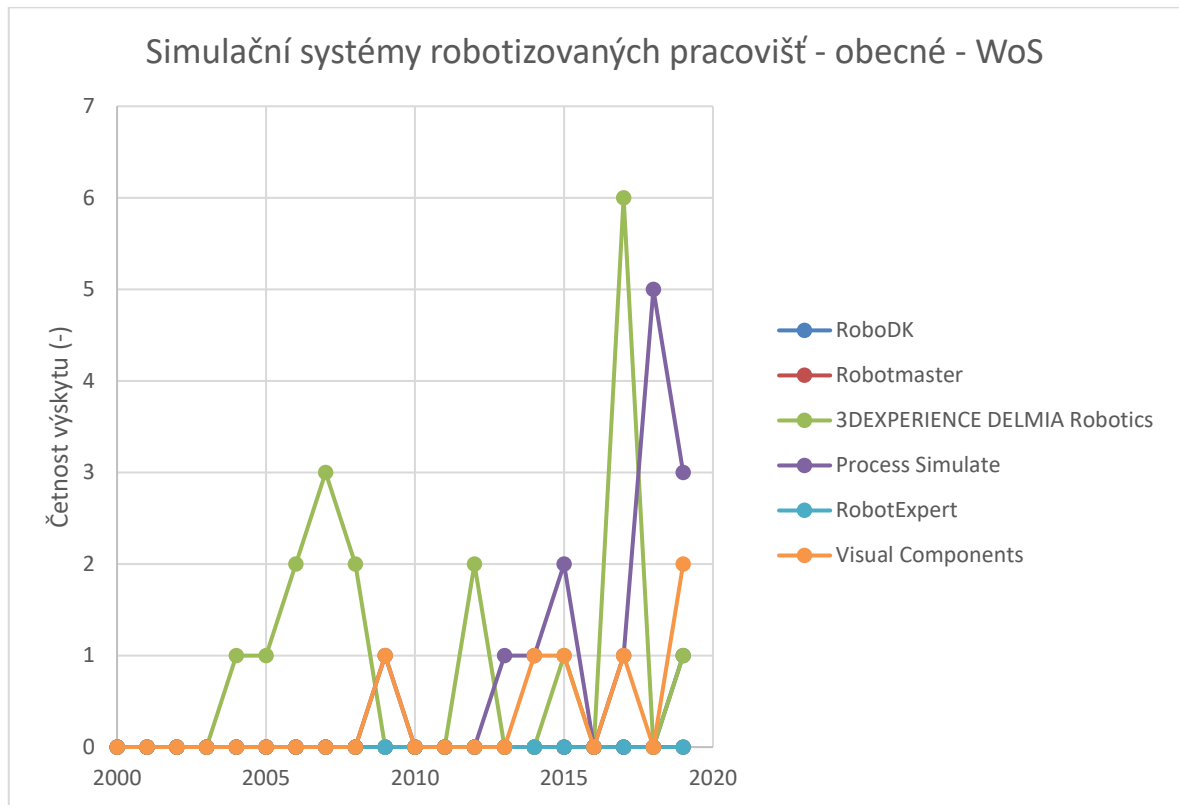
Obr. 11 Graf četnosti výskytu obecných simulačních systémů na webu Science Direct



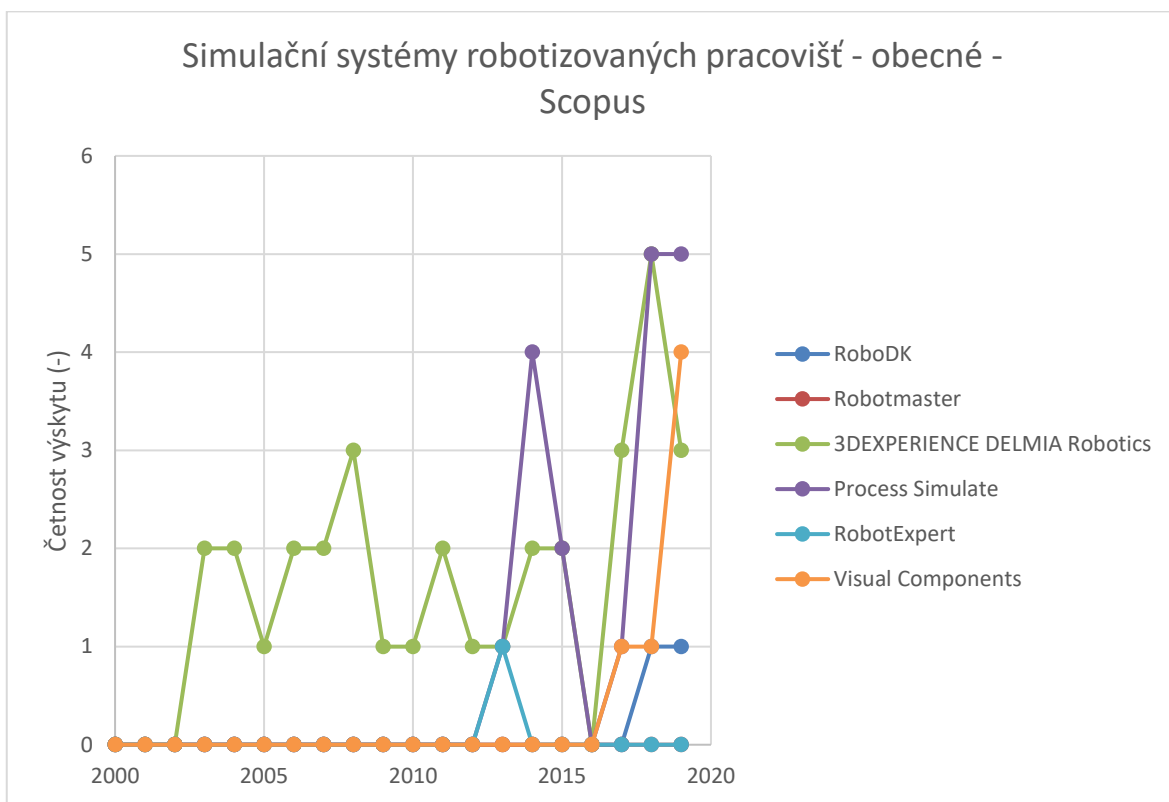
Obr. 12 Graf četnosti výskytu obecných simulačních systémů na webu Springer

### 2.3 Simulační systémy robotizovaných pracovišť – obecné

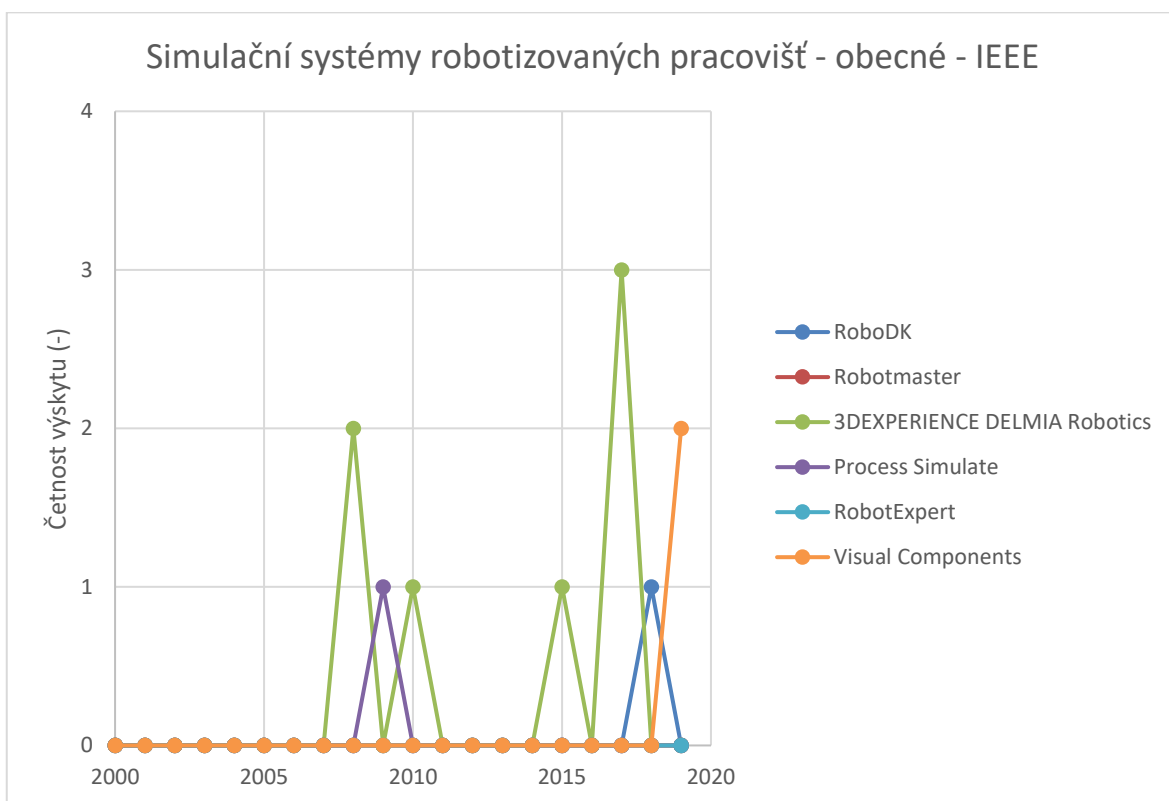
Oproti obecným SW se tyto SW pro simulaci robotizovaných pracovišť ve zkoumaných zdrojích vyskytují s daleko menší četností, což je dle mého názoru dáno jejich specializací, která omezuje oblast jejich použití. Během průzkumu v databázích jsem zjistil, že většina publikací se týkala mobilních robotů. Jejich simulaci průmyslové systémy nenabízí. Dalším rozdílem oproti obecným SW je to, že nelze jednoznačně určit nejvíce citovaný SW.



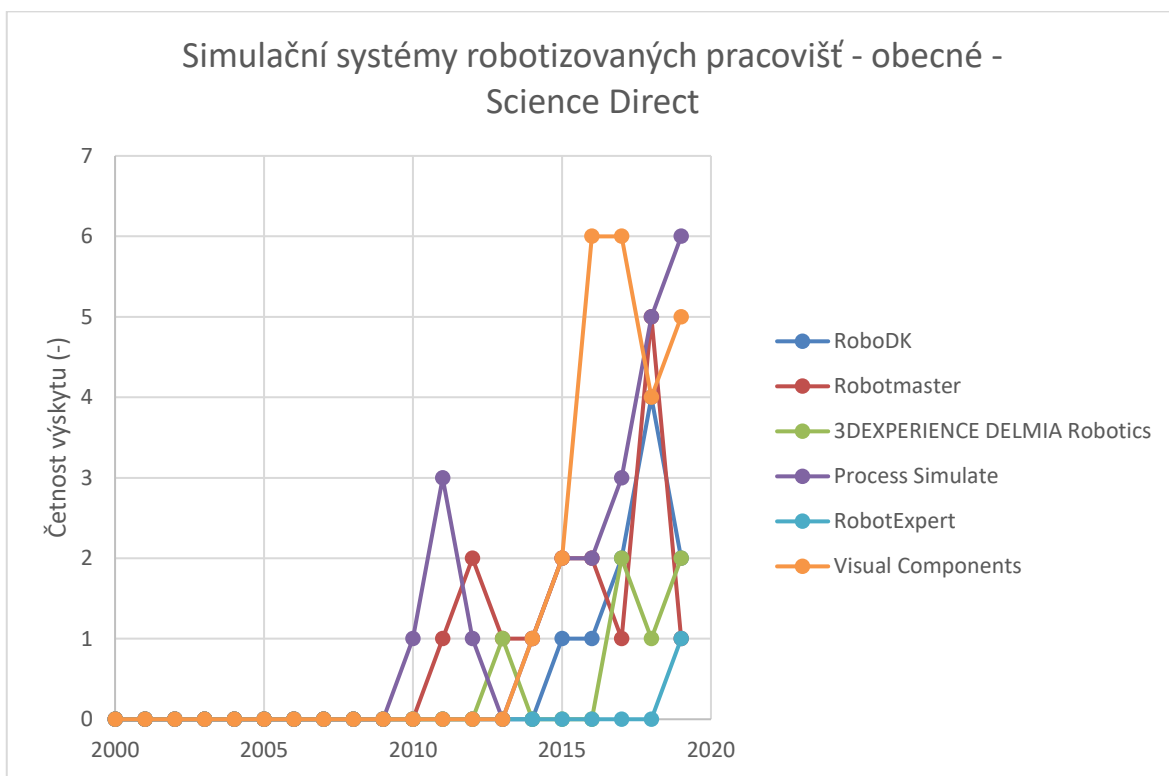
Obr. 13 Graf četnosti výskytu průmyslových simulačních systémů na webu WoS



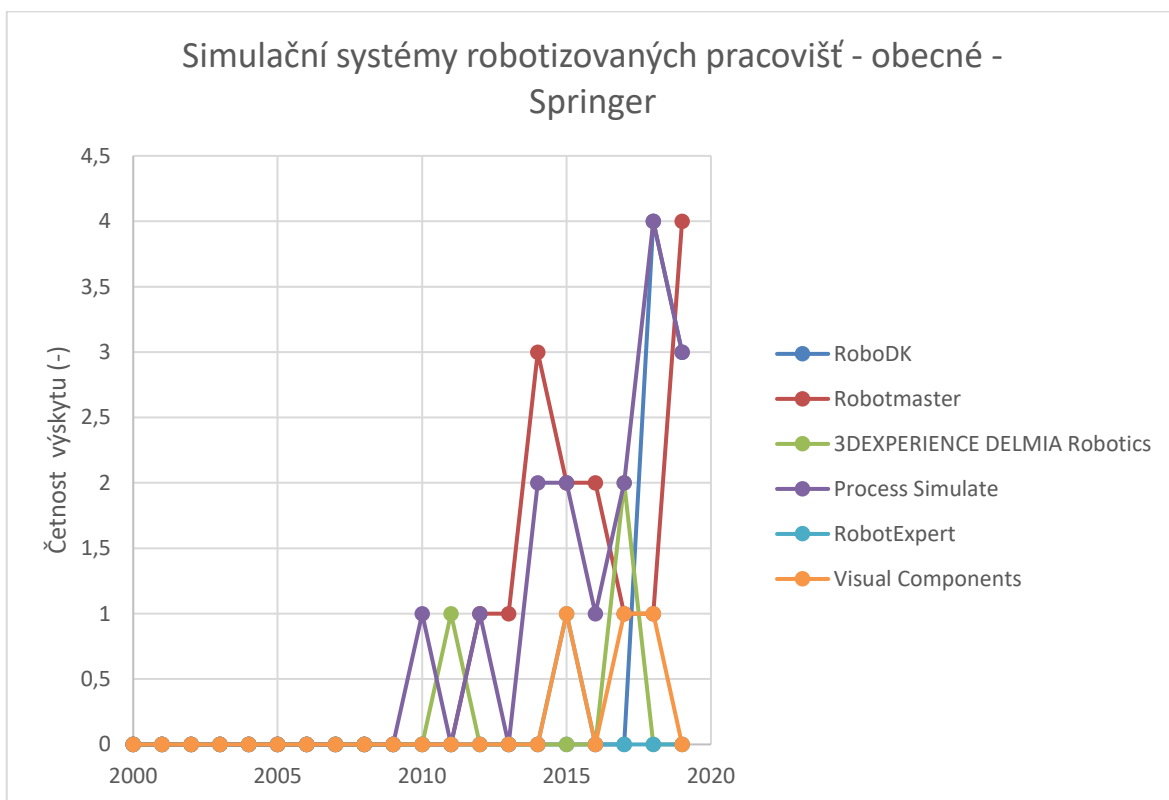
Obr. 14 Graf četnosti výskytu průmyslových simulačních systémů na webu Scopus



Obr. 15 Graf četnosti výskytu průmyslových simulačních systémů na webu IEEE



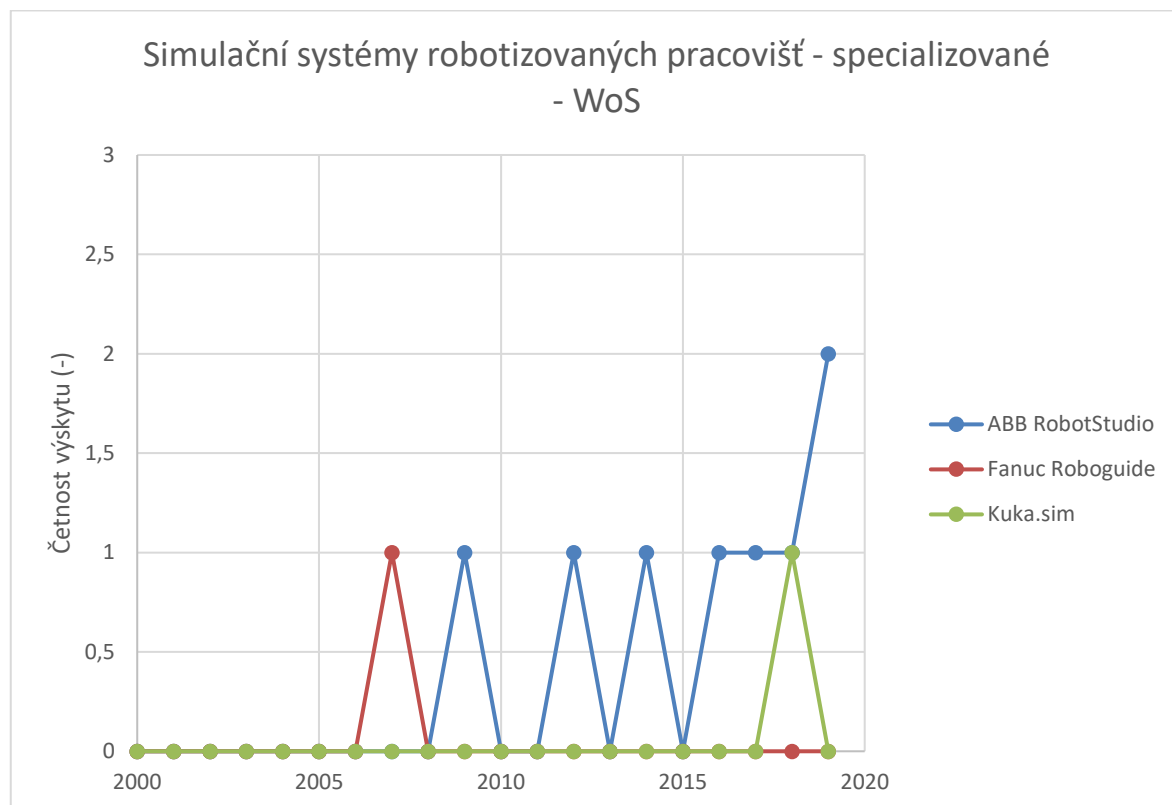
Obr. 16 Graf četnosti výskytu průmyslových simulačních systémů na webu Science Direct



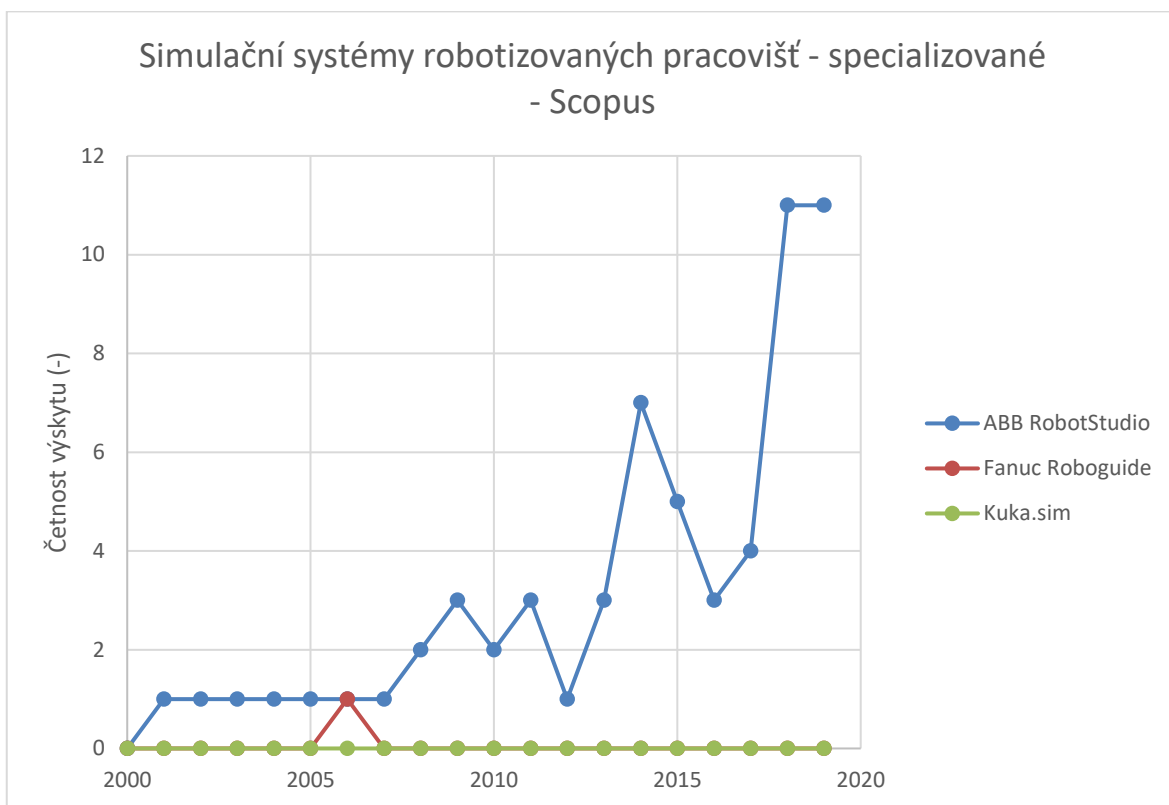
Obr. 17 Graf četnosti výskytu průmyslových simulačních systémů na webu Springer

## 2.4 Simulační systémy robotizovaných pracovišť – specializované

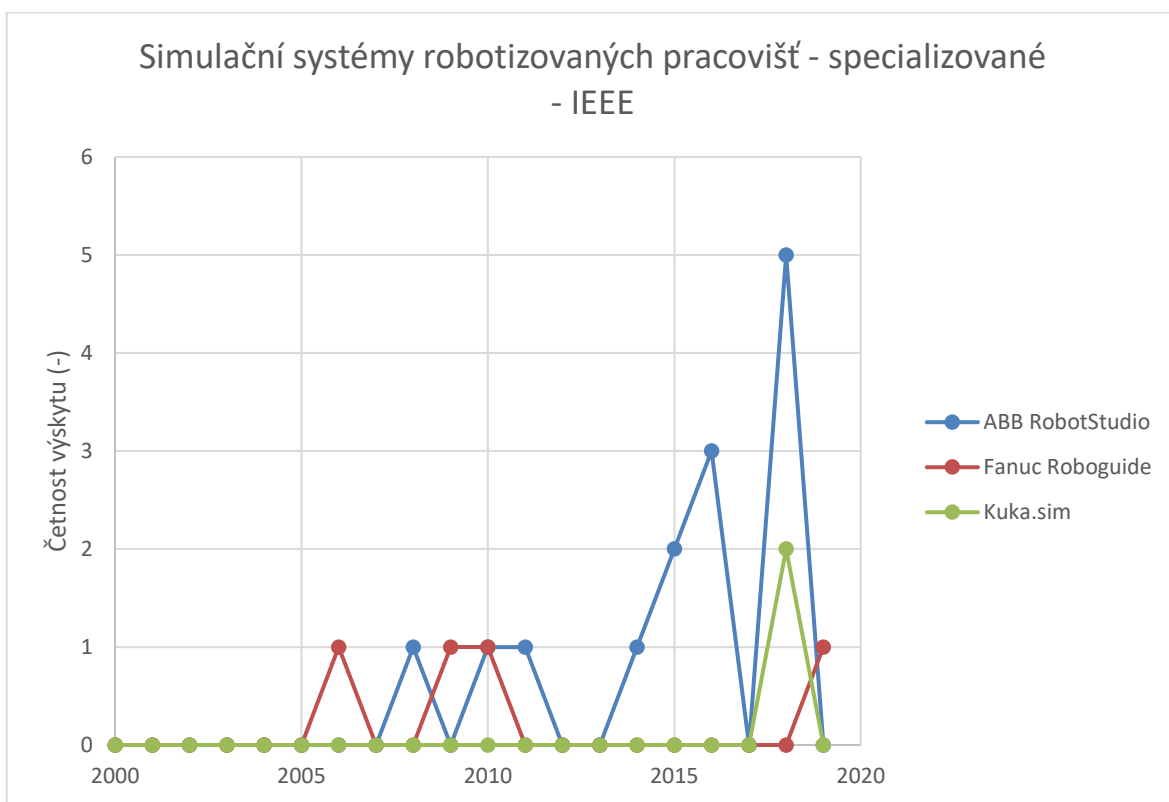
Podobně jako u průmyslových, tak i u speciálních simulačních SW je jejich četnost výskytu relativně nízká. Dle mého názoru je to opět specializací, která je v tomto případě ještě užší. Nejvíce zmiňovaný software je v tomto případě ABB RobotStudio.



Obr. 18 Graf četnosti výskytu speciálních simulačních systémů na webu WoS

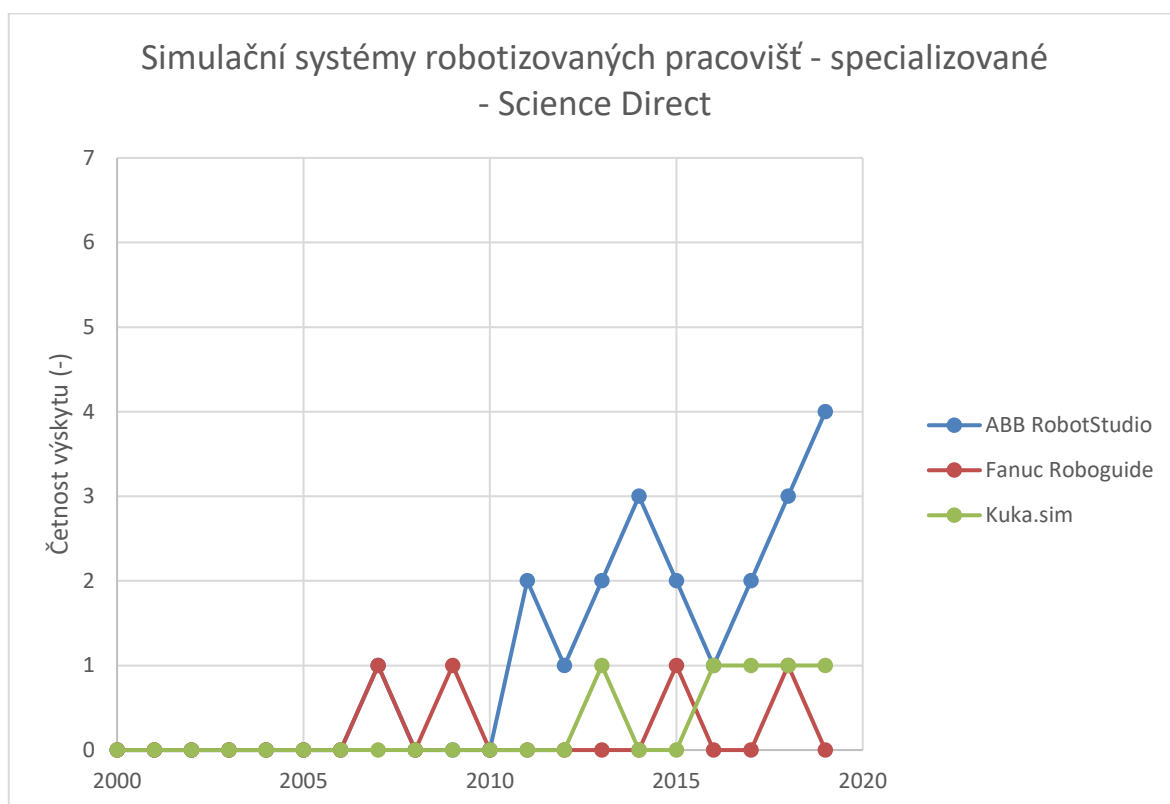


Obr. 19 Graf četnosti výskytu speciálních simulačních systémů na webu Scopus

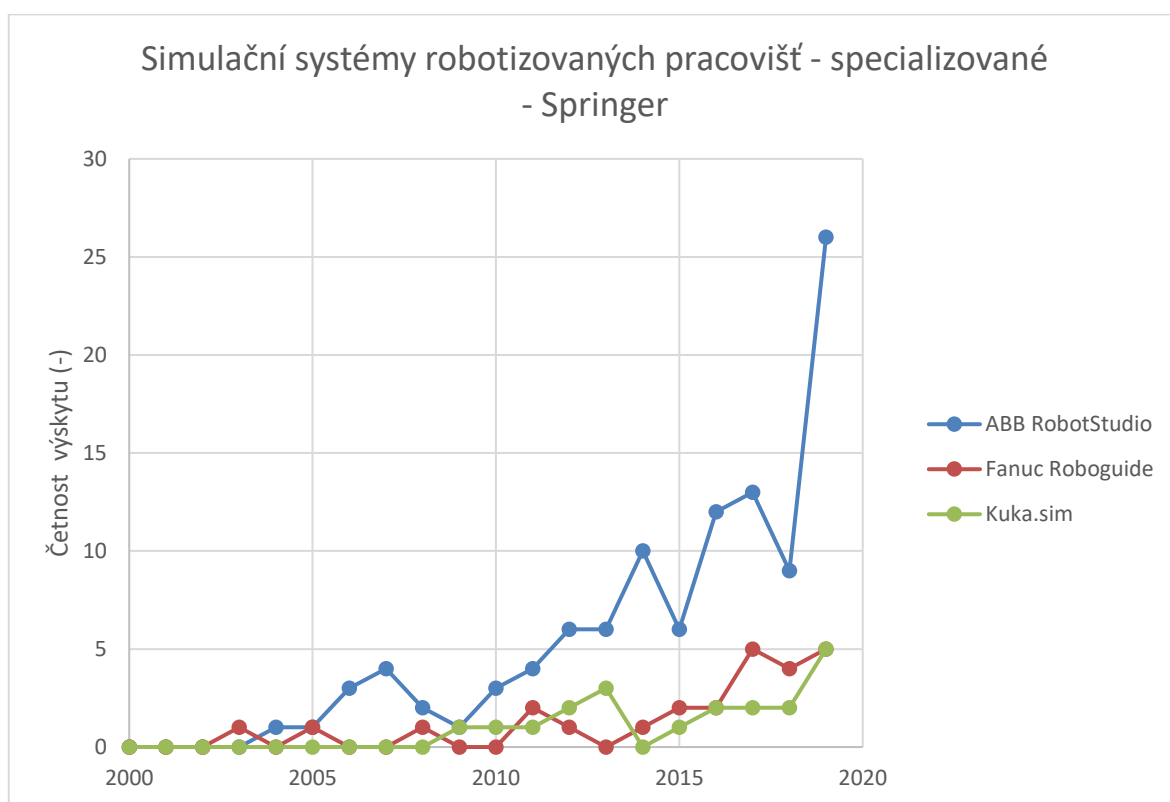


Obr. 20 Graf četnosti výskytu speciálních simulačních systémů na webu IEEE





Obr. 21 Graf četnosti výskytu speciálních simulačních systémů na webu Science Direct



Obr. 22 Graf četnosti výskytu speciálních simulačních systémů na webu Springer

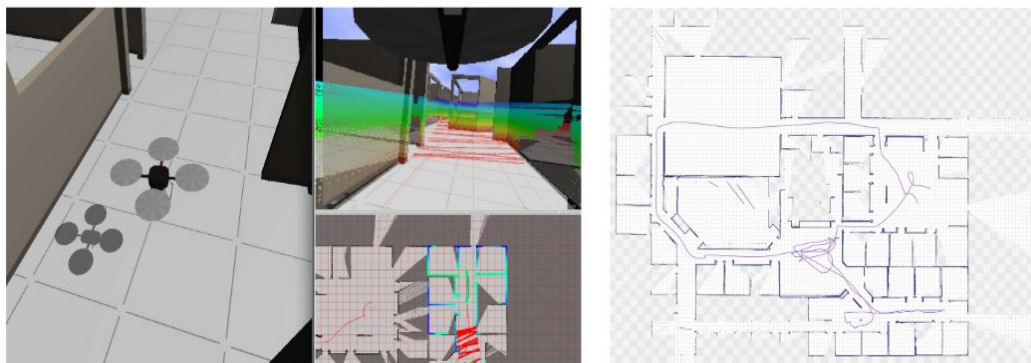
### 3 Nejvíce citované publikace a příklady použití

Seznam jsem tvořil z databází WoS a Scopus, které nabízejí vhodné nástroje a zároveň velký počet citací. Ostatní databáze buď neumožňovaly setřídít výsledky podle počtu citací (Springer, Science Direct), nebo obsahovaly citací příliš málo (IEEE). Do seznamu jsem nezahrnoval dokumenty, které se věnovaly přímo představení daného softwaru jako celku, a ne konkrétním případům jeho využití. Nejvíce citované články se v různých databázích často opakovaly. Například články nacházející se zároveň v databázích WoS a Scopus měly vyšší četnost citování v databázi Scopus, proto je u nich zmíněn počet citací v databázi Scopus. Další publikace a zdroje informací, které se nehodily jako příklady použití, ale vyskytovaly se mezi nejvíce citovanými anebo z nich lze získat další informace o simulačních softwarech jsou uvedeny v kapitole 1.8.

#### 3.1 Gazebo

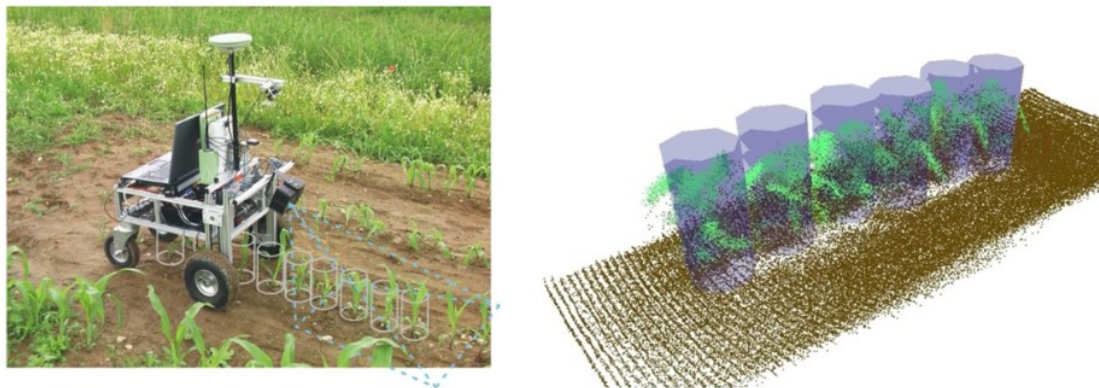
1. **Comprehensive simulation of quadrotor UAVs using ROS and Gazebo** (Obsáhlá simulace čtyřmotorového UAV pomocí ROS a Gazebo). Scopus: 101 citací. [17]
2. **Plant detection and mapping for agricultural robots using a 3D LIDAR sensor** (Detekce a mapování rostlin pro zemědělské roboty pomocí 3D LIDAR senzoru). Scopus: 100 citací. [18]
3. **RotorS-A Modular Gazebo MAV Simulator Framework** (RotorS – modulární Gazebo MAV simulátor). WoS: 91 citací. [19]

V první publikaci se jedná o komplexní simulaci čtyřmotorového UAV. Prezentovaný systém je integrován s ROS a softwarem Gazebo. Umožňuje simulaci dynamiky letu, palubních senzorů, např. IMU a komplexního prostředí. Dynamický model UAV byl parametrizován pomocí aerodynamického tunelu a ověřen porovnáním simulovaných a skutečných letových údajů.



Obr. 23 Obsáhlá simulace čtyřmotorového UAV pomocí ROS a Gazebo [17]

V publikaci „Plant detection and mapping for agricultural robots using a 3D LIDAR sensor“ se pomocí Gazeba porovnává 3D LIDAR senzor (laserový scanner) oproti typickým 3D senzorům používaných na mobilních robotech. Aplikace se zabývá detekcí a členěním rostlin autonomními zemědělskými roboty. Senzor se nakonec ukázal jako vhodné řešení.



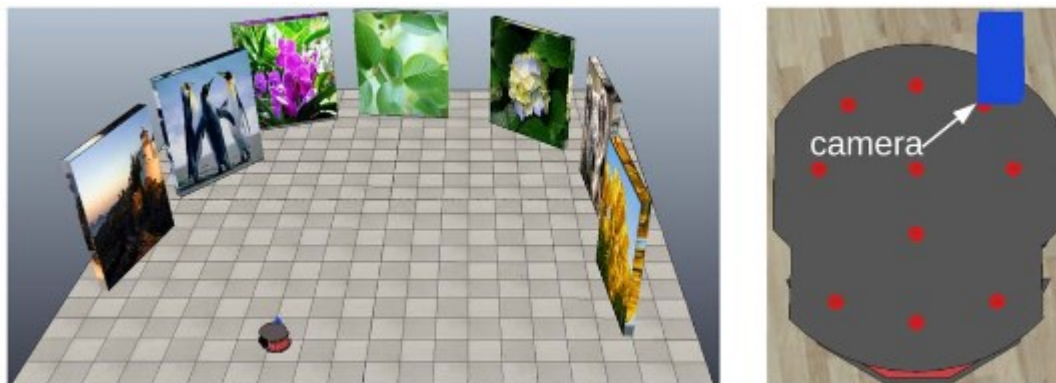
Obr. 24 Detekce a mapování rostlin pro zemědělské roboty pomocí 3D LIDAR senzoru [18]

V posledním případě se jedná také o létajícího robota, ovšem tentokrát o MAV. Cílem této publikace je představit modulární simulátor MAV, který má sloužit jako základ pro řešení složitějších problémů, jako jsou vizuální problémy, plánování cesty a vyhýbání se kolizím.

### 3.2 V-rep

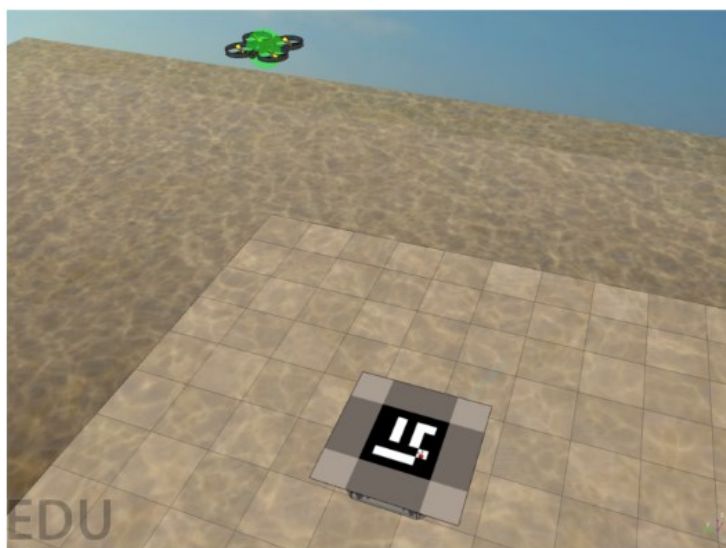
1. **Trifocal Tensor-Based Adaptive Visual Trajectory Tracking Control of Mobile Robots** (Adaptivní kontrola vizuální trajektorie mobilních robotů založená na trifokálním tenzoru). Scopus: 25 citací. [20]
2. **Tools for simulating humanoid robot dynamics: A survey based on user feedback** (Nástroje pro simulaci dynamiky humanoidních robotů: Průzkum založený na zpětné vazbě od uživatelů). Scopus: 19 citací. [2]
3. **Vision based fuzzy control autonomous landing with UAVs: From V-REP to real experiments** (Vizuálně založené ovládání autonomního přistání s UAV: Od V-REPU ke skutečným experimentům) Scopus: 18 citací. [21]

První publikace se věnuje trifokálnímu tenzoru a úloze sledování vizuální trajektorie neholonomického mobilního robota vybaveného monokulární kamerou. Požadovaná trajektorie je vyjádřena sadou předem zaznamenaných obrazů a robot je regulován tak, aby sledoval požadovanou trajektorii pomocí vizuální zpětné vazby. Trifokální tenzor se využívá k získání informací o orientaci a měřítku polohy použitých v řídicím systému a pracuje pro obecné scény.



Obr. 25 Adaptivní kontrola vizuální trajektorie mobilních robotů [20]

V publikaci „Vision Based Fuzzy Control Autonomous Landing with UAVs: From V-REP to Real Experiments“ byl V-Rep byl použit pro návrh a vyladění řízení UAV na základě vizuální informace. Cílem UAV bylo sledovat pohybující se mobilní robot a následně na něm přistát. Jako UAV byl použit dron, jako mobilní přistávací plošina robot se všesměrovými koly. Je zde prezentováno simulační prostředí vytvořené ve V-Repu a propojené s ROS a jsou popsány i reálné experimenty.



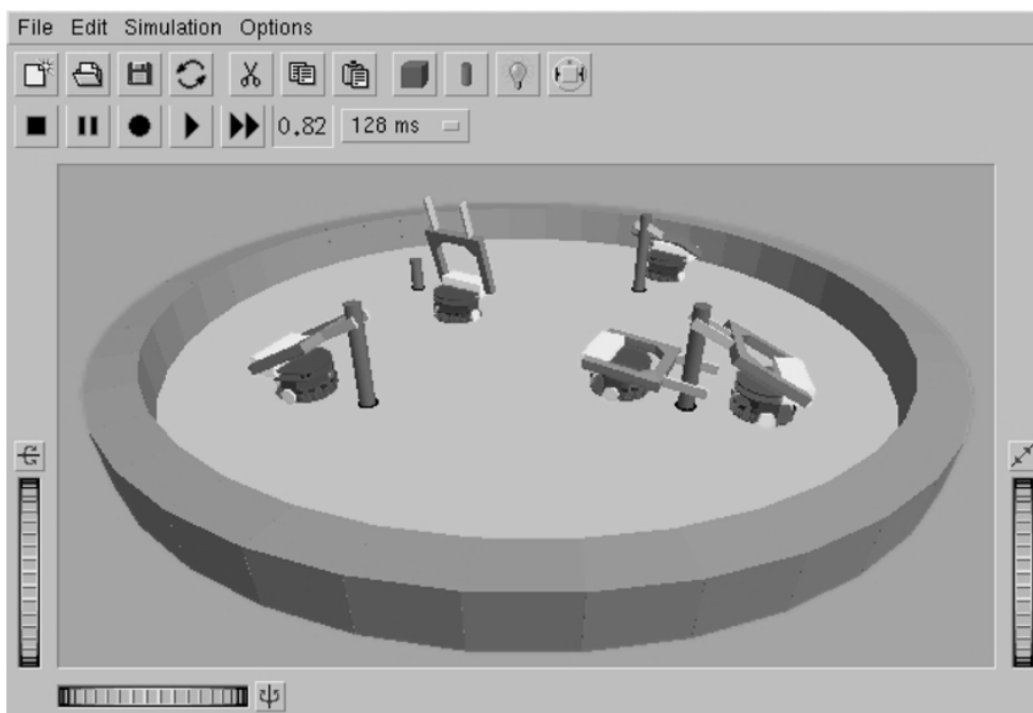
Obr. 26 Vizuálně založené autonomní přistání UAV [21]

### 3.3 Webots

1. **Collaboration through the exploitation of local interactions in autonomous collective robotics: The stick pulling experiment** (Spolupráce prostřednictvím využití místních interakcí v autonomní kolektivní robotice: Experiment s vytahováním tyčinek). Scopus: 137 citací. [22]
2. **Swarm robotic odor localization** (Lokalizace zápachu robotickým rojem). Scopus: 85 citací. [23]

3. **Webots: Symbiosis between virtual and real mobile robots** (Webots: Symbióza mezi virtuálními a reálnými mobilními roboty) Scopus: 71 citací. [24]

„Collaboration through the exploitation of local interactions in autonomous collective robotics: The stick pulling experiment“ využívá Webots pro simulaci úkolu vytahování tyčinek. Pro úspěšné vytáhnutí je potřeba spolupráce dvou robotů.



Obr. 27 Experiment s vytahováním tyčinek [22]

„Swarm robotic odor localization“ se zabývá výzkumem lokalizace zápachu skupinami autonomních mobilních robotů pomocí „Swarm Intelligence“. V publikaci je ověřeno, že simulátor dokáže reprodukovat experimenty skutečných robotů a může být užitečným nástrojem pro off-line studium. Poslední zmiňovaná publikace týkající se systému Webots popisuje kompatibilitu rozhraní mezi simulátor a skutečným robotem. Reálné i simulované roboty jsou zde programovány v programovacím jazyce C a rozhraní Khepera API.

### 3.4 Actin

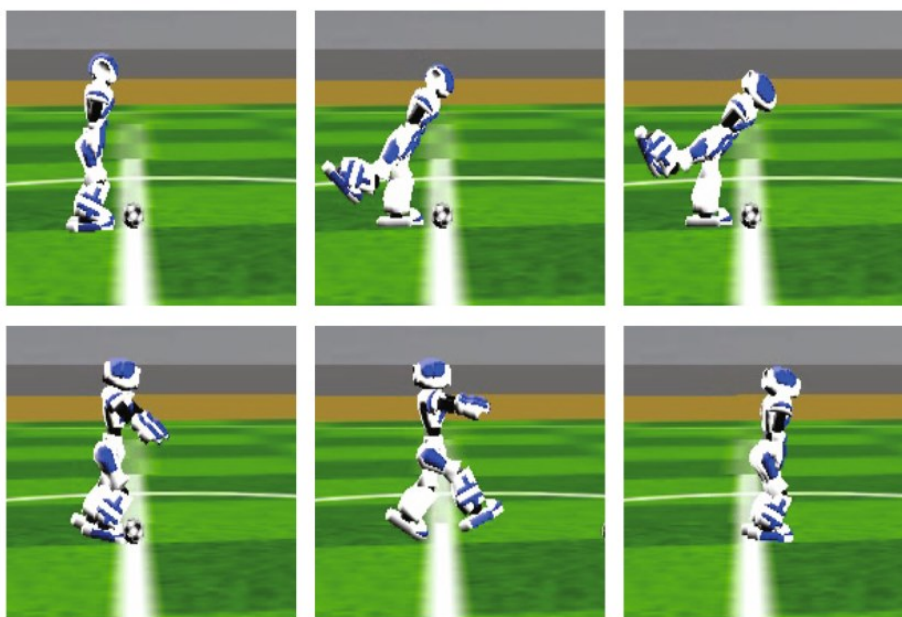
U tohoto softwaru nebyly v databázích žádné publikace týkající se využití v robotice. To však neznamená, že v praxi není používán.

### 3.5 SimSpark

1. **Optimization Approach for the Development of Humanoid Robots' Behaviors** (Optimalizační přístup pro vývoj chování humanoidních robotů). WoS: 4 citace. [25]

2. **Two humanoid simulators: Comparison and synthesis** (Dva humanoidní simulátory: srovnání a syntéza). Scopus: 4 citace. [26]
3. **Learning low level skills from scratch for humanoid robot soccer using deep reinforcement learning** (Učení jednoduchých fotbalových dovedností od počátku pro humanoidní roboty pomocí „deep reinforced learning“). Scopus: 2 citace. [27]

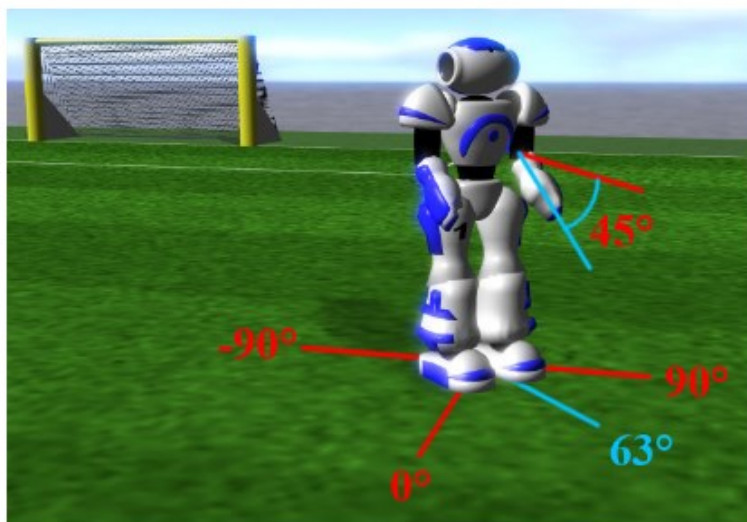
V nejvíce citované publikaci týkající se softwaru SimSpark se řeší automatický přístup k rozvoji humanoidního chování založený na vývoji obecného optimalizačního systému. Je popsána optimalizace „bočního a dopředného kopu“. V obou případech optimalizace umožnila vytvoření rychlejšího a stabilnějšího chování robotů.



Obr. 28 Optimalizační přístup pro vývoj chování humanoidních robotů [25]

Publikace „Two humanoid simulators: Comparison and synthesis“ srovnává softwaru SimSpark a SimTwo. Předmětem porovnání bylo základní nízko úrovněvé humanoidní chování. Oba simulátory sdílejí stejný model humanoidního robotu Albaderan NAO robot. Výsledky ukázaly, že simulace provedené v softwaru SimSpark mají blíže k realitě.

Ve třetí publikaci se rovněž vyskytuje humanoidní robot NAO. Jak název napovídá, jedná se o učení jednoduchých fotbalových dovedností pomocí „deep reinforced learning“. Je navržen nepřímý přístup k řízení a podrobné nastavení parametrů, aby bylo dosaženo přirozeného chování při běhu a driblování. Získaný výkon je v některých případech srovnatelný nebo lepší než nejlepší týmy RoboCup.



Obr. 29 Učení jednoduchých fotbalových dovedností [27]

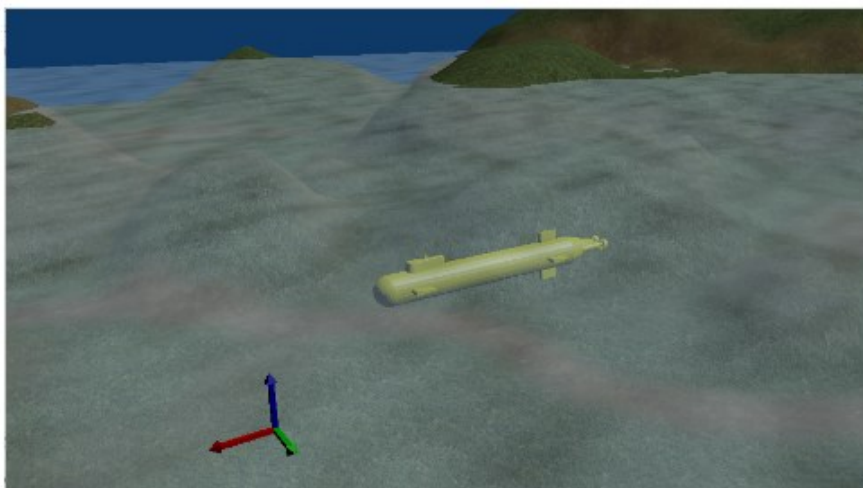
### 3.6 MORSE

1. **A survey of AUV and robot simulators for multi-vehicle operations** (Přehled AUV a robotických simulátorů pro více vozidlové operace). Scopus: 23 citací. [28]
2. **UW MORSE The UnderWater Modular Open Robot Simulation Engine** (UW MORSE, Podvodní modulární otevřený robotický simulační „engine“). WoS: 5 citací. [29]
3. **A simulator for Underwater Human-Robot Interaction scenarios** (Simulátor pro podvodní scénáře interakcí člověk-robot). Scopus: 5 citací. [30]

„A survey of AUV and robot simulators for multi-vehicle operations“ porovnává simulátory jako MORSE, UWSim, Gazebo atd. z hlediska jejich vhodnosti pro simulace podvodních robotů (a skupin podvodních robotů). Porovnávaná kritéria jsou například: dostatečná fyzická věrnost realitě, manipulátory a efekty, programové prostředí, dokumentace, senzory atd. Podle srovnávací tabulky Tab. 22 se MORSE současně se softwarem Gazebo ukázal jako nejlepší software pro simulaci podvodních robotů.

Tab. 22 Srovnání softwaru pro simulaci podvodních robotů [28]

	Hlavní kritéria				Vedlejší kritéria	
	Přesnost fyziky	Programové rozhraní	Dostatečná dokumentace	Předchozí použití ve výzkumu	Vizuální přesnost	Rozšiřitelnost
<b>UWSim</b>	Vysoká	Jednoduché	Ne	Ano	Vysoká	Nízká
<b>MORSE</b>	Vysoká	Násobné	Ano	Ano	Střední	Vysoká
<b>Gazebo</b>	Vysoká	Jednoduché	Ano	Ano	Střední	Vysoká



Obr. 30 Přehled AUV a robotických simulátorů pro více vozidlové operace [28]

„UW MORSE The UnderWater Modular Open Robot Simulation Engine“ dále popisuje vlastnosti simulátoru. Lze jej použít pro vývoj řídicích systémů, plánování cest a testování ve virtuálním prostředí. Modulová povaha MORSE umožňuje uživateli snadno konfigurovat simulace, vytvářet nová prostředí a roboty a také přidávat senzory a řídicí rozhraní. Toto rozšíření zahrnuje moduly pro hydrodynamickou simulaci, trysky a podvodní senzory. Tyto moduly umožňují uživateli vytvořit virtuální repliku konkrétního podvodního robotického systému. Simulační prostředí umožňuje interakci s velkou sadou robotických „middlewarů“.

„A simulator for Underwater Human-Robot Interaction scenarios“ pojednává o nástavbě MOOS pro ROS a MORSE. Tato nástavba byla vyvinuta za účelem testování algoritmů pro podvodní roboty, které mají spolupracovat s člověkem. MOOS dále demonstuje základní scénář, kdy musí ROV následovat lidského potápěče.

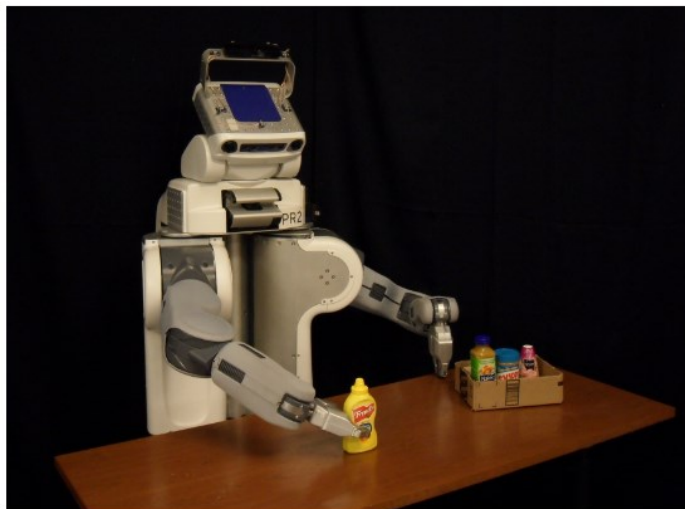
### 3.7 OpenRAVE

1. **Cloud-based robot grasping with the google object recognition engine** (Robot využívající cloud a uchopení pomocí „engine“ pro rozpoznávání objektů Google). Scopus: 119 citací. [31]
2. **Modeling and control of MM-UAV: Mobile manipulating unmanned aerial vehicle** (Modelování a ovládání MM-UAV: Mobilního bezpilotního vzdušného vozidla pro manipulaci). Scopus: 72 citací. [32]
3. **Perception for mobile manipulation and grasping using active stereo** (Vnímání pro mobilní manipulaci a uchopení pomocí aktivního sterea). Scopus: 35 citací. [33]

Software OpenRAVE byl využit v publikaci „Cloud-based robot grasping with the google object recognition engine“. Publikace popisuje systém vzniklý propojením robotu Willow Garage PR2, modulu pro rozpoznání objektů společnosti Google, Point Cloud Library, GraspIT! a OpenRAVE. Kamery robotu pošlou informace na servery Google, které

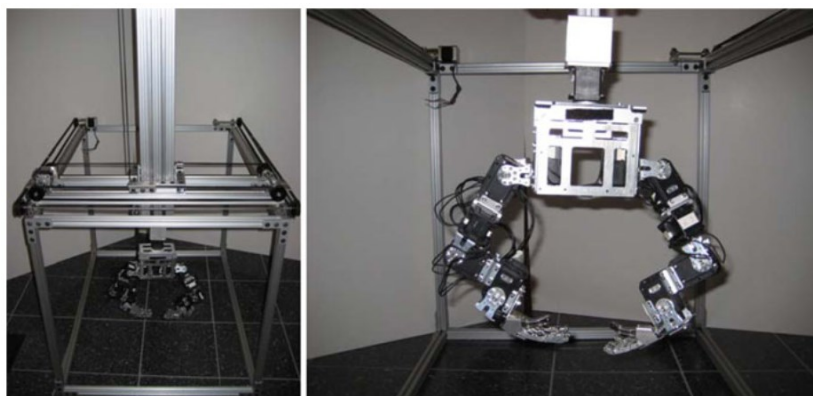


následně vrátí seznam rozpoznaných předmětů a instrukcí k uchopení daných předmětů. Následně je pomocí změřených bodů určena poloha objektu a určení celkového provedení úchopu. Po provedení úkonu robot vyhodnotí výsledek a odešle ho na server pro pozdější reference.



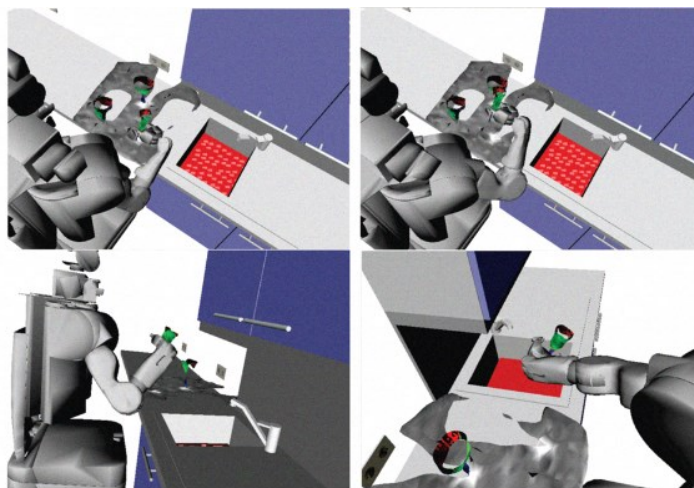
Obr. 31 Robot využívající cloud a uchopení pomocí „engine“ pro rozpoznávání objektů [31]

Publikace „MM-UAV: Mobile manipulating unmanned aerial vehicle“ se věnuje možným využitím létajících robotů z hlediska manipulace s objekty. Je zde uveden popis systému, který má napodobit létající obratný mobilní manipulátor. K otestování a simulaci tohoto systému byl použit OpenRAVE a ROS.



Obr. 32 Ovládání MM-UAV [32]

Poslední publikace v tomto seznamu zmiňující OpenRAVE se opět zabývá uchopováním objektů a robotem Willow Garage PR2. Ke vnímání objektů je použito mračno bodů, ze kterého jsou následně objekty rekonstruovány a jejich použitelnost je ověřena v softwaru OpenRAVE.



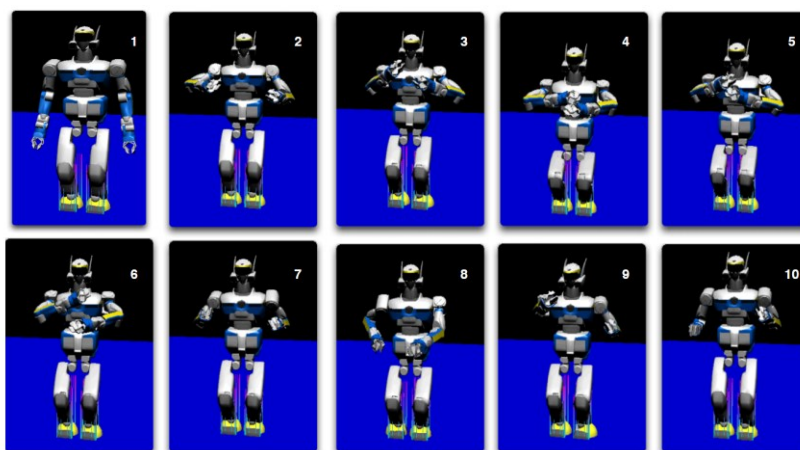
Obr. 33 Vnímání pro mobilní manipulaci [33]

### 3.8 OpenHRP

1. **Generating whole body motions for a biped humanoid robot from captured human dances** (Generování pohybů celého těla pro dvounohý humanoidní robot ze zachycených lidských tanců). Scopus: 166 citací. [34]
2. **On human motion imitation by humanoid robot** (Napodobování lidských pohybů humanoidním robotem). Scopus: 83 citací. [35]
3. **The first humanoid robot that has the same size as a human and that can lie down and get up** (První humanoidní robot stejné velikosti jako člověk, který sám dokáže lehnout a vstát). Scopus: 48. 78[36]

„Generating whole body motions for a biped humanoid robot from captured human dances“ popisuje postup snímání tanečních pohybů pomocí technologie „motion capture“ a následné pokusy robotu HRP-1S o napodobení těchto pohybů. Generované pohyby byly testovány v softwaru OpenHRP.

Stejným tématem napodobování pohybů zachycených technologií „motion capture“ se zabývá i druhá nejvíce citovaná publikace týkající se tohoto softwaru. Pomocí OpenHRP se ověřovalo, zda pohyby robotu odpovídají původním zachyceným pohybům.



Obr. 34 Napodobování lidských pohybů humanoidním robotem [35]

Software OpenHRP byl také použit u simulace dalších lidských pohybů, konkrétně lehnutí a vstání. Všechny tyto tři nejvíce citované publikace ukazují na možnost využití OpenHRP jako simulátoru humanoidních robotů.

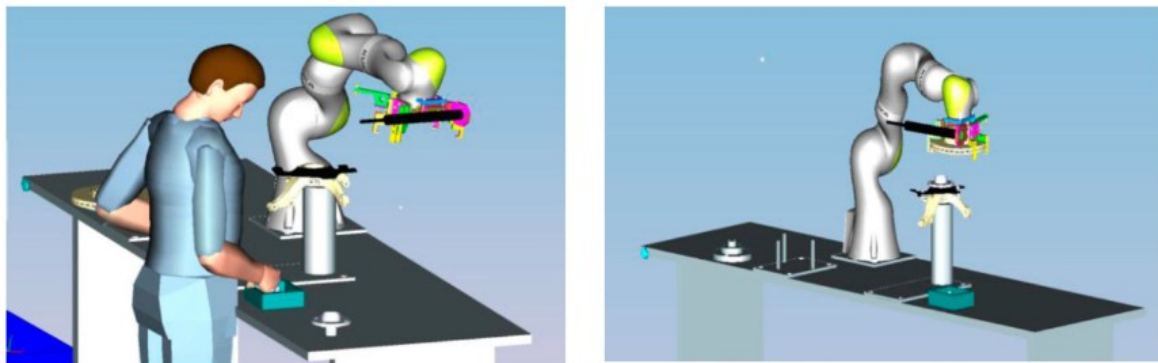
### 3.9 RobotExpert

1. **Offline programming for robotic deburring process of aluminium wheels** (Off-line programování pro proces robotického odjehlování hliníkových kol). Scopus: 3 citace. [37]

Jediná citovaná publikace týkající se SW RobotExpert se zabývá využitím tohoto SW k off-line programování. Je popsán a vyvinut proces odjehlování hliníkových kol. SW byl použit společně s roboty ABB. Finální dráha robotického nástroje se kontroluje pomocí prohlížeče kolizí, sledování statusu kloubů a rychlosti středového bodu nástroje.

### 3.10 Process Simulate

1. **Safety design and development of a human-robot collaboration assembly process in the automotive industry** (Návrh zabezpečení a vývoj procesu montáže spolupráce člověk-robot v automobilovém průmyslu). Scopus: 13 citací. [38]
2. **Virtual Commissioning with Process Simulation (Tecnomatix)** (Virtuální uvedení do provozu se simulací procesu (Tecnomatix)). Scopus: 12 citací. [39]
3. **Simulation techniques for ergonomic performance evaluation of manual workplaces during preliminary design phase** (Simulační techniky pro hodnocení ergonomického výkonu manuálních pracovišť během přípravné fáze návrhu). Scopus: 3 citace. [40]



Obr. 35 Spolupráce člověka s robotem [38]

V nejvíce citované publikaci týkající se SW Process Simulate jsou prezentovány výsledky výzkumu spolupráce mezi člověkem a robotem v případě sestavení brzdového kotouče pro automobily. K modelování pracovní stanice se používá zmíněný software, což poskytuje příležitost k vyhodnocení proveditelnosti procesu pomocí různých scénářů. Nakonec je proveden experimentální test. Výsledkem bylo, že ačkoliv spolupráce robota s člověkem mírně zvyšuje celkovou dobu zpracování, výrazně vylepšuje ergonomii člověka a snižuje riziko zranění.

V další publikaci [39] byla vytvořena přesná reprodukce výrobní buňky. Virtuální buňka spolupracuje s programovatelným logickým kontrolérem (dále PLC) Siemens S7-300, který operátorovi umožňuje vytvářet a ověřovat kód PLC uvnitř virtuálního modelu. Na konci této publikace je navržena aplikace těchto konceptů v učebně vysoké školy.

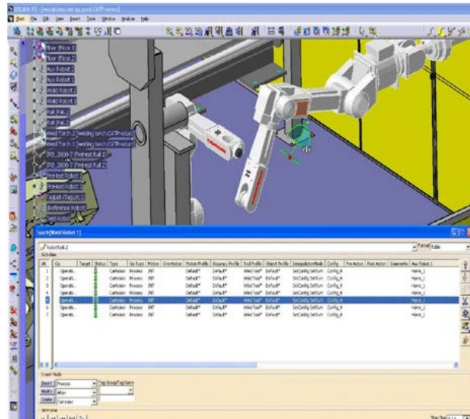
Poslední zmíněná publikace týkající se tohoto softwaru měla za cíl navrhnout inovativní postup na podporu návrhu pracovišť založený na simulačních technikách, které umožňují nastavení virtuálního scénáře, ve kterém je digitální lidský model schopen provádět úkoly spojené s montáží.

### 3.11 3DEXPERIENCE Delmia Robotics

1. **Offline programming for a complex welding system using DELMIA automation** (Offline programování pro komplexní svařovací systém pomocí DELMIA automatizace) Scopus: 10 citací. [41]
2. **Energy-optimal layout design of robotic work cells: Potential assessment on an industrial case study** (Energeticky optimální návrh rozvržení robotických pracovních buněk: Posouzení potenciálu na průmyslové případové studii). Scopus: 7 citací. [42]
3. **On the implementation of a robotic welding process using 3D simulation environment** (Implementace robotického svařovacího procesu pomocí 3D simulačního prostředí). Scopus: 4 citace. [43]

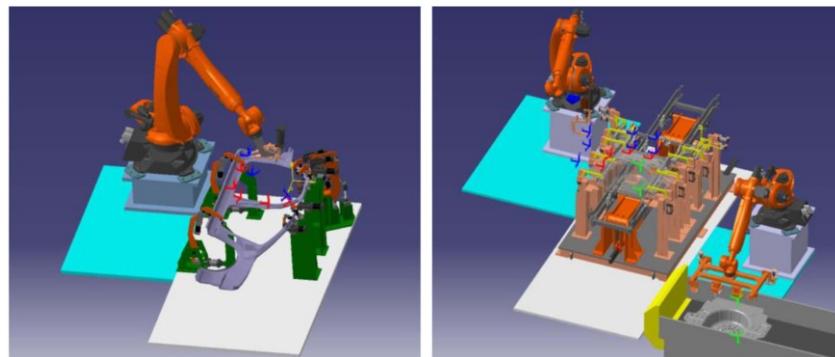
Publikace „Offline programming for a complex welding system using DELMIA automation“ ukazuje, že je možné upravit simulační software tak, aby automatizoval některé

kroky v procesu off-line programování. To snižuje čas potřebný k vytvoření kódu pro robotizovanou svařovací buňku.



Obr. 36 Offline programování pro komplexní svařovací systém [36]

Druhá publikace představuje metodu rozmístění několika PR do výrobní buňky tak, aby byla spotřebovaná energie při procesu výroby minimální. Nejdříve je nutné jednotlivým PR přiřadit úkoly, následně se vypočte optimální rozložení PR v buňce. Výstupem je i barevná nebo obrysová mapa zobrazující energetickou náročnost. Metoda byla ověřena v simulačním prostředí Delmia. Numerické výsledky potvrdily možnou úsporu až 20% energie.



Obr. 37 Energeticky optimální návrh rozvržení robotických pracovních buněk [42]

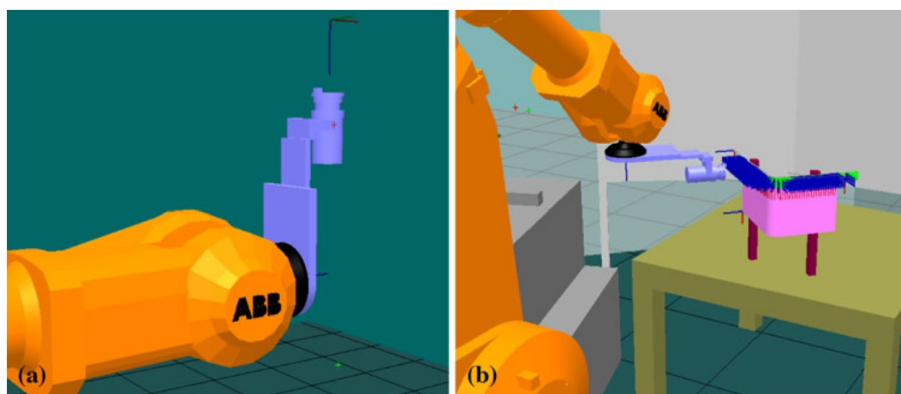
Poslední příklad použití softwaru 3DEXPERIENCE zahrnuje simulaci automatizovaného svařovacího procesu. Simulovaným robotem byl robot KUKA KR16. Simulace měla pomoci odhalit různé chyby, které by mohly vzniknout při následné konstrukci pracoviště. Jmenovitě rozměrová kontrola, kontrola kolizí, plánování dráhy atd.

### 3.12 ABB RobotStudio

1. **Technology and applications of ABB RobotStudio** (Technologie a využití ABB RobotStudio). Scopus: 32 citací. [44]
2. **The effect of robot kinematics on the coating thickness uniformity** (Efekt kinematiky robotu na rovnoměrnost tloušťky povlaku). Scopus: 18 citací. [45]
3. **Flexible robot sealant dispensing cell using RGB-D sensor and off-line programming** (Flexibilní robotická buňka pro dávkování tmelu využívající RGB-D sensor a offline programování). Scopus: 13 citací [46]

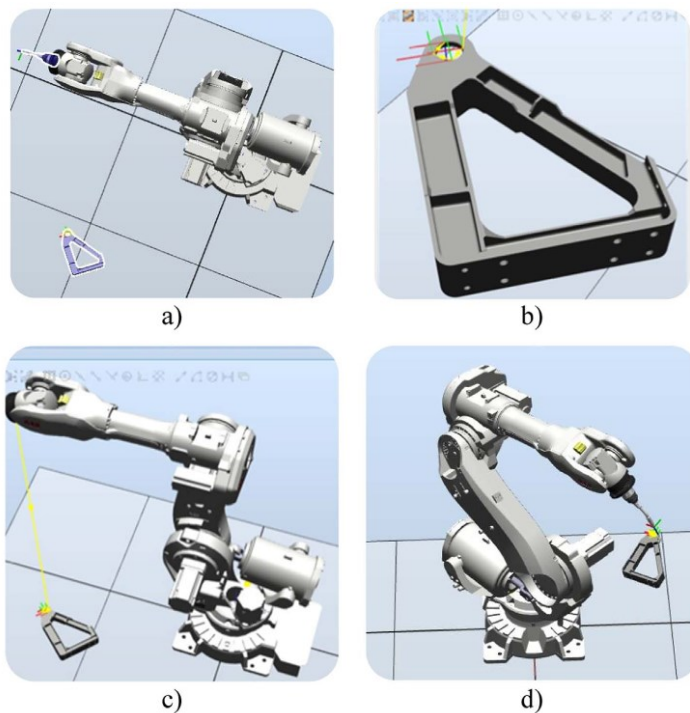
První publikace se věnuje možnostem využití softwaru. Jako výhody popisuje zvýšení efektivity a přesnosti programování robotu a zkrácení prostojů. Software také hraje roli při údržbě, odstraňování problémů, snižování rizika nehody a při vzdělávání.

„The effect of robot kinematics on the coating thickness uniformity“ řeší téma nanášení rovnoměrné vrstvy povlaku pomocí trysky připevněné na efektor robotu. K tomu je potřebná optimalizace kinematiky robotu, aby byl jeho pohyb co nejvíce konstantní. Za tímto účelem byl vyvinut přídatný program k softwaru ABB RobotStudio, který tuto problematiku řeší.



Obr. 38 Efekt kinematiky robotu na rovnoměrnost tloušťky povlaku [45]

Poslední publikace se rovněž zabývá rozšířením funkcionality RobotStudia. Předmětem simulace je nanášení tmelu na součásti při malosériové výrobě, kde se často mění tvar těchto součástí. Programovat pracoviště na každý tvar součásti je tedy pracné a nasazení automatizace je ekonomicky nevýhodné. Úkolem nástavby je pomocí RGB-D senzoru rozeznat součást a její pozici a dynamicky vygenerovat potřebnou část programu pro robot.



Obr. 39 Flexibilní robotická buňka pro dávkování tmelu [46]

### **3.13 Shrnutí kapitoly**

Díky informacím obsaženým v této kapitole je možné určit vhodné použití zmíněných simulačních SW. Nejvíce citované publikace obecných simulačních SW se zabývaly převážně mobilními roboty. Lze vidět, že u některých SW se nejvíce citované publikace týkají jednoho tématu. Simulace robotu Albaderan NAO pomocí SW SimSpark, podvodní aplikace MORSE a simulace robotizovaného svařovacího procesu pomocí 3DEXPERIENCE Delmia Robotics jsou toho příkladem. V případě SW SimSpark je to z důvodu specializace a omezení simulace na daného robotu. V dalších dvou příkladech je to dle mého názoru dáno částečně omezením SW, částečně vhodností simulačního SW pro popsanou simulaci. Také nejvíce citované publikace u ostatních simulačních SW často vykazují podobnosti. Všechny publikace zmiňující V-Rep a Gazebo zahrnují nejvýše dva simulované roboty, jedná se o složitější simulace, často s využitím vizuálních senzorů. Oproti tomu publikace týkající se Webots se věnují velkému počtu jednoduchých robotů.

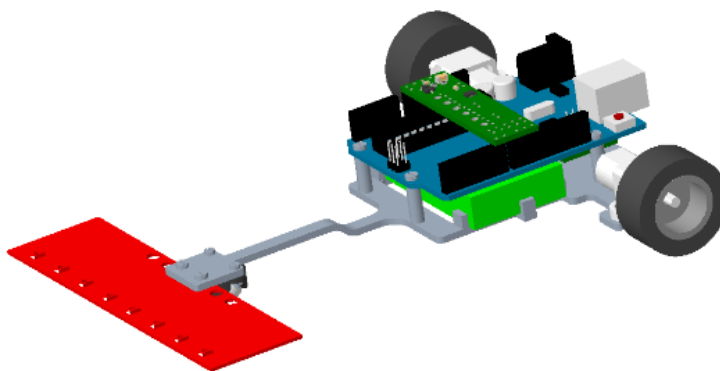
Příklady nejvíce citovaných publikací dále podporují důležitost funkcí a vlastností zmíněných v kapitole 1.4. Z uvedených příkladů je také zřejmá důležitost rozšiřitelnosti daného softwaru. SW byl v některých případech rozšířen o specializované funkce, díky kterým se následně stal vhodným nástrojem pro simulaci dané problematiky. Další důležitou vlastností je kompatibilita s ostatními SW nebo reálnými roboty. Důkazem je kombinace SW u příkladu použití OpenRAVE.

Nejedná se zde o kompletní výčet vhodného použití zmíněných simulačních SW. Pro dohledání dalších informací o schopnostech SW doporučuji navštívit internetové stránky daného SW nebo vývojářů, případně další zdroje. To, že je předmět simulace obsažený v nejvíce citovaných publikacích uvedeného SW nemusí nutně znamenat, že je uvedený SW nejvhodnějším simulačním nástrojem pro danou problematiku. Navíc zde existuje možnost, že některé nejvíce citované publikace nemusí dobře reprezentovat nejčastější způsob použití zmíněných simulačních SW. I tak je tato kapitola podle mého názoru dostatečným ukazatelem rozlišností zmíněných SW a poukazuje na vhodné příklady použití, ze kterých lze odvodit použitelnost SW na podobné případy.



## 4 Vypracování komplexního simulačního modelu

V této kapitole je vytvořen soupis potřebných parametrů a podkladů pro vypracování komplexního simulačního modelu robotického systému. Pro lepší názornost jsem vytvořil soupis týkající se konkrétního příkladu, mobilního robota sledujícího čáru – „line follower“. Se simulací tohoto robota se setkávají studenti Katedry robotiky v předmětu Servisní roboty. Úkolem studentů je navrhnout robot, který projede trať vyznačenou černou čarou v co nejkratším čase. V této kapitole není popsán konkrétní návod „krok po kroku“, obsah kapitoly je zaměřen na parametry a podklady potřebné k vytvoření simulace a na to, jak je získat.

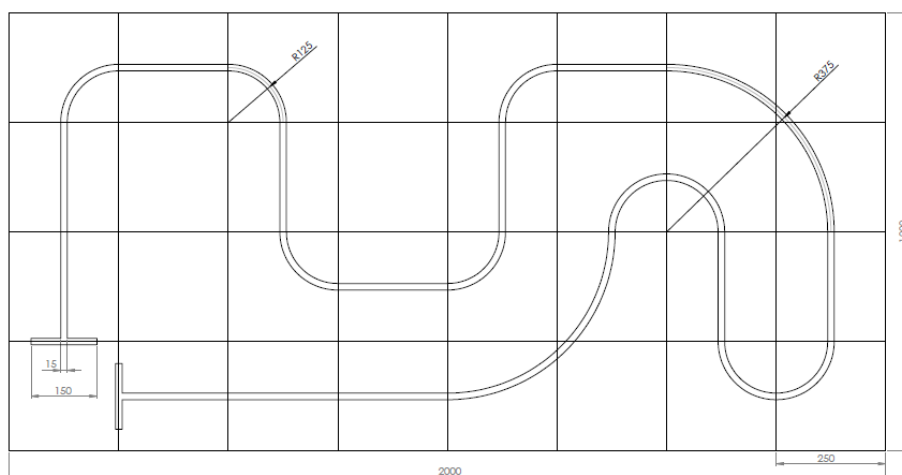


Obr. 40 „Line follower“

Jelikož se v předmětu používá SW V-Rep, vytvořím soupis parametrů a podkladů právě pro tento SW, protože s ním mám osobní zkušenost. Lze předpokládat, že pro ostatní simulační systémy bude soupis obdobný.

### 4.1 Analýza prostředí

Prvním krokem při tvorbě simulačního modelu je analýza daného robotického systému a jeho okolí. Robot se v tomto případě nachází v uzavřené místnosti. Je uvažována stálá teplota a další ideální vlastnosti prostředí. V místnosti je absence jakýchkoliv překážek, lidí a dalších robotů. V závislosti na zvolených senzorech robota na jeho funkci může mít vliv osvětlení, to je v uvažované místnosti dostatečné a konstantní. Prostředí tedy na robot nebude mít žádný rušivý vliv. Je však nutné získat rozměry tratě, po které se bude robot pohybovat.



Obr. 41 Závodní trať pro line-follower

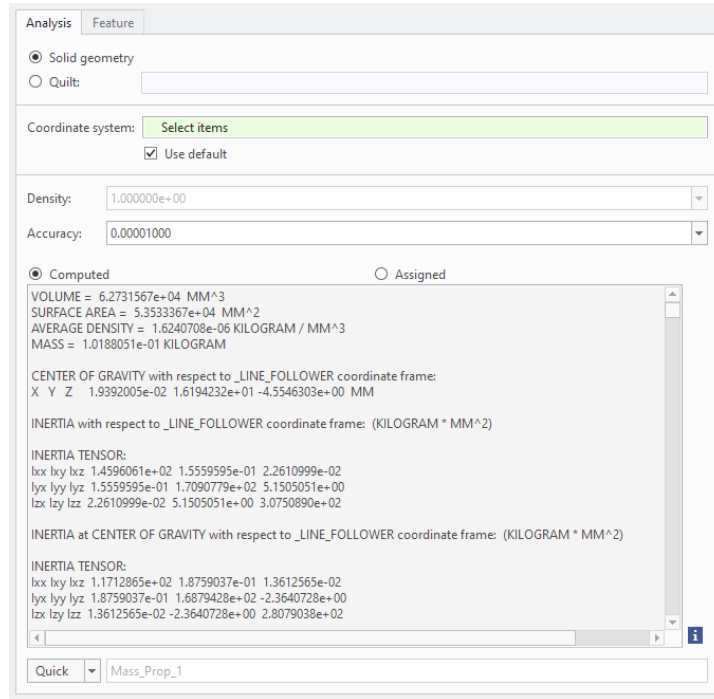
## 4.2 Tvorba 3D modelu

Následně je třeba věnovat pozornost samotnému robotu. Robot se skládá z několika součástí od různých výrobců. Pro zjištění celkových parametrů robotu, potřebných pro simulaci, je v tomto případě nejjednodušší vytvořit 3D model pomocí CAD softwaru, např. Creo Parametric. K tomu je potřeba zjistit parametry těchto součástí, které lze získat z katalogů výrobců nebo z internetu, v nejlepším případě je k dispozici přímo 3D model od výrobce. Pomocí rozměrů z katalogů nebo přímo vložených 3D modelů výrobce lze vytvořit 3D model robotu.

Tento model slouží v simulačním SW pouze pro vizuální reprezentaci robotu, není důležité, aby byl na modelu zaznamenán každý detail. Je často nežádoucí, aby byl vizuální model složitý, zpomaluje pak práci v prostředí V-Repu. Vytvořený model je zobrazen na Obr. 40. Na druhou stranu tento model bude před přenosem do simulačního SW sloužit k určení dalších parametrů, proto je vhodné, aby se co nejvíce blížil skutečnému robotu. Je nutné volit kompromis mezi jednoduchostí modelu z důvodu přílišného zatížení simulačního SW a jeho přesností z důvodu co nejpřesnějšího určení hmotnosti součástí a momentů setrvačnosti k těžišti. Řešením může být i vytvoření dvou modelů, jednodušší model se použije pro vizualizaci v prostředí simulace a přesnější model pro získání parametrů.

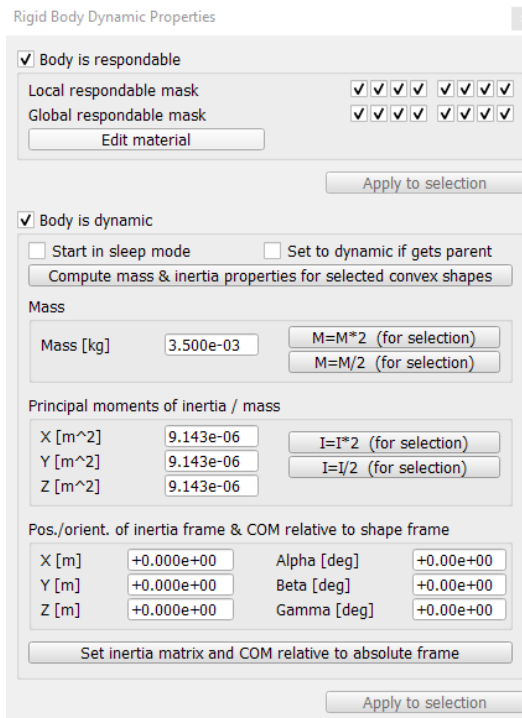
## 4.3 Určení momentů setrvačnosti k těžišti

Po vytvoření modelu je nutné ke všem součástem přiřadit správné hmotnosti. Ty jsou získány buď od výrobce anebo výpočtem pomocí rozměrů a hustoty dané součásti. Díky těmto hmotnostem lze pomocí CAD softwaru určit polohu těžiště robotu a momenty setrvačnosti jednotlivých os k tomuto těžišti. To je nutné provést pro robot jako celek a dále pro všechny součásti, které se v simulaci budou chovat jako dynamické. V tomto případě bylo tedy pomocí CAD softwaru určeno těžiště robotu, dále těžiště kola a těžiště opěrné kuličky. Následně momenty setrvačnosti k těžišti jednotlivě pro robot, kolo a opěrnou kuličku.



Obr. 42 Určení polohy těžiště a velikosti momentů setrvačnosti robotu

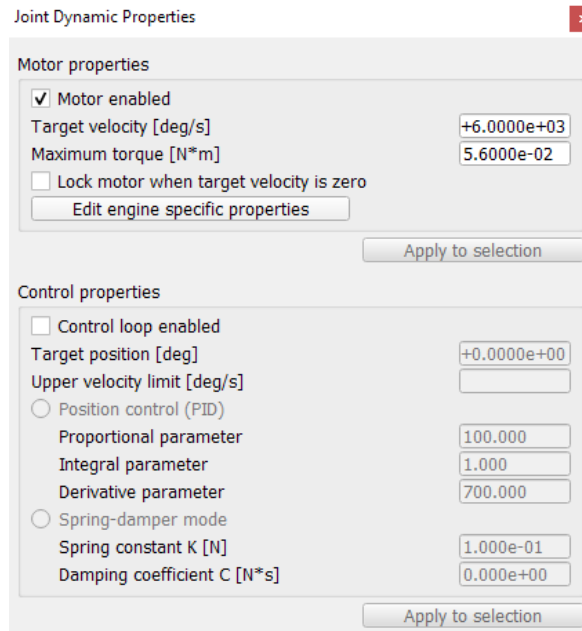
V této fázi je hotový vizuální model robotu, je známá celková hmotnost, hmotnost kola a hmotnost opěrné kuličky. Dále jsou známé momenty setrvačnosti k těžišti pro robot, kolo a opěrnou kuličku. Takto vytvořený model se získanými parametry lze přenést do simulačního SW.



Obr. 43 Vložení dynamických parametrů opěrné kuličky

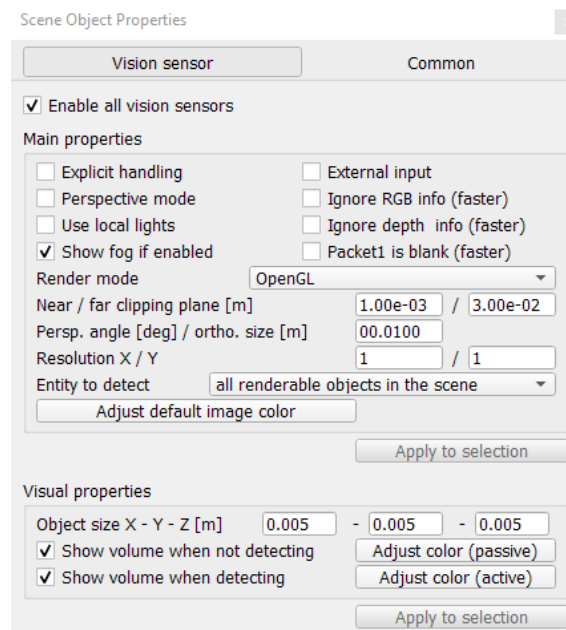
#### 4.4 Parametry motorů a senzorů

Dalšími potřebnými parametry jsou parametry týkající se pohonu robotu. V tomto případě se jedná o dva stejné elektromotory se zabudovanou převodovkou. Z dokumentace výrobce těchto motorů je nutné získat hodnoty maximální úhlové rychlosti a kroutícího momentu, který tento typ motoru dokáže dlouhodobě vyvinout.



Obr. 44 Vložení parametrů motoru

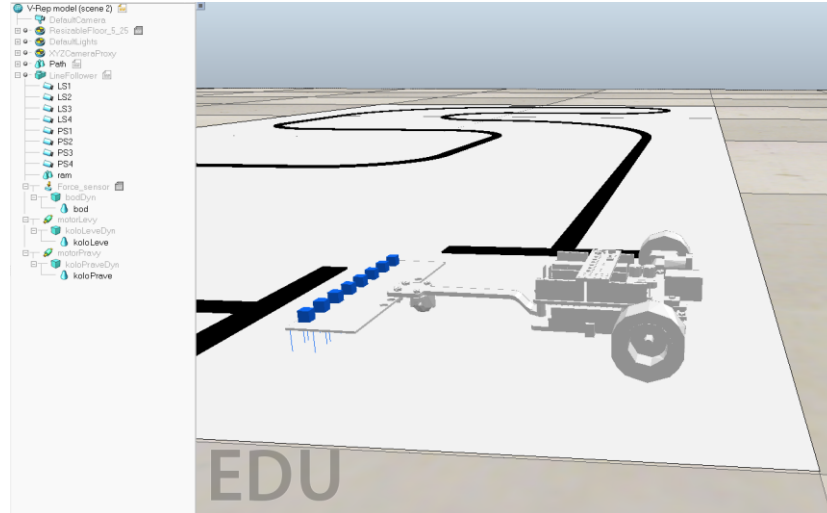
Robot používá aktivní infračervenou senzorovou desku s 8 senzory. Pro umístění senzorů v simulačním prostředí V-Rep je nutné znát polohu senzorů na desce. Dalším parametrem senzorů je rozsah jejich snímací vzdálenosti.



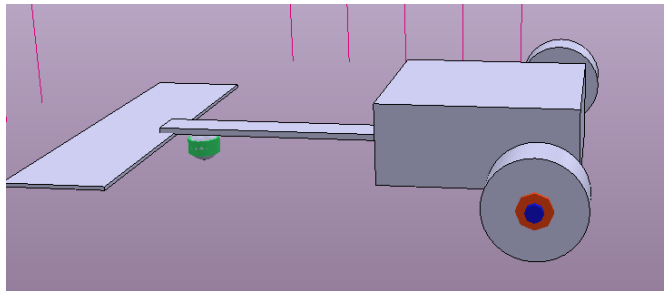
Obr. 45 Vložení parametrů senzorů

## 4.5 Program robotu

V tomto bodě je v simulačním SW vizuální model robotu vložený z CAD softwaru i dynamický model, vytvořený pomocí vložení dynamických parametrů součástí a parametrů motorů a senzorů.



Obr. 46 Vizuální model robotu a hierarchie scény



Obr. 47 Dynamický model robotu

Posledním krokem pro dokončení tvorby simulačního modelu je vytvoření programu robotu. V tomto kroku se může soupis parametrů lišit i pro stejný simulační SW, protože záleží na zvoleném přístupu. To, aby robot sledoval čáru a dojel tak až do cíle, lze naprogramovat různými způsoby, které využívají různé parametry. V tomto případě je to maximální rychlost robotu, na kterou byl omezen z důvodu snazšího řízení, a koeficienty pro různé zpomalení jednotlivých motorů.

```

7 | -- Definovani rychlosti a koeficientu
8 | speed = 29
9 | k1 = 2
10 | k2 = 200
11 | k3 = 10000

```

Obr. 48 Parametry programu robotu

Program řídí robot tak, že pokud snímají čáru prostřední senzory na sensorové desce, jede maximální dovolenou rychlostí. Pokud začnou snímat čáru ostatní senzory, znamená to, že je třeba upravit směr jízdy. To se provede snížením rychlosti na jednom z motorů. Toto snížení závisí na zvolených koeficientech, čím vzdálenější senzor detekuje čáru, tím více se zpomalí motor na stejné straně robotu, čímž se docílí požadované rychlosti zatáčení. Všechny koeficienty obsažené v programu robotu byly zvoleny experimentálně.

#### **4.6 Shrnutí potřebných parametrů**

Níže je uveden seznam potřebných parametrů pro vytvoření simulačního modelu „line follower“. Pro přehlednost jsem parametry rozdělil na vstupní a odvozené. Vstupní parametry bylo potřeba získat ještě před započítím tvorby simulace. Odvozené parametry vyplynou v průběhu tvorby simulačního modelu ze vstupních parametrů.

Pro tvorbu komplexního simulačního modelu „line follower“ jsou tedy potřeba tyto vstupní parametry:

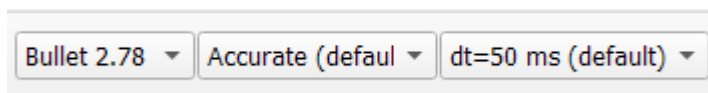
- **Rozměry nebo model závodní tratě** pro přenesení závodní tratě do simulačního SW.
- **3D modely nebo rozměry všech součástí robotu** pro vytvoření vizuálního modelu robotu.
- **Hmotnost nebo hustota všech součástí robotu** pro získání těžiště a momentů setrvačnosti.
- **Úhlová rychlost a kroutící moment motorů robotu** pro definování parametrů motorů v prostředí simulačního SW.
- **Poloha a snímací vzdálenost senzorů** pro rozmístění a definování parametrů senzorů v prostředí simulačního SW.

Pomocí těchto parametrů jsem následně získal další podklady k vytvoření simulace. Odvozené parametry:

- **Poloha těžiště robotu a všech součástí určených pro dynamický model robotu** pro určení momentů setrvačnosti.
- **Momenty setrvačnosti jednotlivých os** pro definování dynamických parametrů modelu robotu v prostředí simulačního SW.
- **Koeficienty pro řízení robotu** potřebné k vytvoření programu robotu.

## 4.7 Vliv nastavení simulace na výsledek a rychlost simulace

V této kapitole popíšu vliv změny nastavení simulace na výsledek simulace – na čas, za který robot projede dráhu a na reálný čas, který je potřeba k provedení simulace. Výsledek lze totiž dále ovlivnit nastavením parametrů simulace, „engine“ fyziky, přesnosti simulace a parametru dt (tzv. „time step“, časový krok) (Obr. 49).



Obr. 49 Nastavitelné parametry simulace

V softwaru CoppeliaSim jsou dostupné čtyři „enginy“ [47]. Bullet je open-source „engine“, který umožňuje 3D detekci kolizí, dynamiku tuhých těles a dynamiku poddajných těles (poslední funkce není momentálně v softwaru CoppeliaSim podporována).

Open Dynamics Engine (ODE) je open-source „engine“ se dvěma hlavními součástmi: dynamikou tuhých těles a detekcí kolizí.

Vortex není open-source, je to komerční „engine“ nabízející reálné parametry (odpovídající fyzikálním jednotkám) pro velké množství fyzikálních vlastností. Jeho využití v softwaru CoppeliaSim požaduje licenci, kterou v době psaní této práce nevlastním, a proto nemůžu Vortex osobně vyzkoušet.

Posledním „enginem“ dostupným v softwaru CoppeliaSim je Newton Dynamics. Momentálně je jeho implementace v beta verzi. Výsledky z tohoto „engine“ také nebudou uvedeny, protože při simulaci se pravidelně stávalo, že robot na začátku vyjel z tratě a převrátil se.

Parametr dt určuje, o kolik se posune čas simulace po vykonání kódu simulace. Má tedy vliv například na to, jak často bude robot kontrolovat data ze senzorů.



Obr. 50 Enginy používané v softwaru V-Rep (CoppeliaSim) [47]

V následující tabulce (Tab. 23) je uvedeno porovnání závislosti času projetí tratě (výsledný čas) a reálného času potřebného k provedení simulace (délka simulace) na zvolené přesnosti simulace, parametru dt a „engine“. Je třeba zdůraznit, že kód robotu je relativně jednoduchý a špatně se přizpůsobuje změnám. Například při malé změně hodnoty parametrů k1 až k3 už nemusí dorazit do cíle. Tyto parametry byly laděny pro základní nastavení

simulace (Obr. 49, toto nastavení je také v tabulce označeno červenou barvou) a jsou uvedené na obrázku Obr. 48. V některých případech robot nedojel až do cíle, ale vyjel z dráhy předčasně (v tabulkách je v závorkách uvedeno, jakou část tratě robot projel). Úspěšného projetí by se docílilo úpravou výše zmíněných parametrů v kódu robotu, ale pro dodržení stejných výchozích podmínek jsem je ponechal při každém pokusu stejné. Neúspěch projetí celé tratě tedy připisuji spíše nepřizpůsobivému kódu robotu, než tomu že by toto nastavení simulace bylo obecně nevhodné.

Tab. 23 Porovnání výsledku a rychlosti simulace

Parametry		Výsledný čas (s)			Délka simulace (s)		
Přesnost (-)	dt (ms)	Bullet 2.78	Bullet 2.83	ODE	Bullet 2.78	Bullet 2.83	ODE
Very fast	10	15,690	15,770	(0/4)	33	40	-
Very fast	25	15,725	15,725	(3/4)	16	16	-
Very fast	50	15,750	(3/4)	(3/4)	8,5	-	-
Very fast	100	(3/4)	(3/4)	(3/4)	-	-	-
Very fast	200	(1/4)	(1/4)	(1/4)	-	-	-
Fast	10	15,690	15,760	(0/4)	39	44	-
Fast	25	15,700	15,750	15,450	16	16	15,5
Fast	50	15,750	(3/4)	15,500	9	-	7
Fast	100	(3/4)	(1/4)	(3/4)	-	-	-
Fast	200	(1/4)	(1/4)	(1/4)	-	-	-
Accurate	10	15,700	15,740	(0/4)	40	39,5	-
Accurate	25	15,725	15,775	15,425	16	16	16
Accurate	50	15,750	15,850	15,450	8,5	8,5	8,5
Accurate	100	(3/4)	(3/4)	(3/4)	-	-	-
Accurate	200	(1/4)	(1/4)	(1/4)	-	-	-
Very accurate	10	15,770	15,760	(0/4)	38,5	38	-
Very accurate	25	15,725	(3/4)	15,500	16	-	16,5
Very accurate	50	15,750	15,750	15,600	8,5	8,5	8,9
Very accurate	100	(3/4)	(1/4)	(3/4)	-	-	-
Very accurate	200	(1/4)	(1/4)	(1/4)	-	-	-

Z uvedené tabulky lze vidět, že při použití jednotlivých nastavení existují rozdíly ve výsledcích. Mezi verzemi „engine“ Bullet je rozdíl patrný často až na druhém desetinném místě. Při použití ODE byl rozptýl hodnot výsledků vyšší. Nejnižší hodnota pak se vyskytla při použití ODE, 15,425 s. Nejvyšší pak při použití Bullet 2.83, 15,85 s.



### **Závěr vyvozený z vlivu nastavení simulace na výsledek**

Vliv zvoleného „engine“ je různý. Často se stávalo, že důsledkem této změny robot nedošel do cíle, což by mohlo být vyřešeno úpravou kódu robotu. Jednotlivé verze „engine“ Bullet se od sebe výrazně neliší. Při použití ODE dosáhl robot kratších časů projetí dráhy. Při použití „engine“ Newton Dynamics se robot choval zcela odlišně.

Vliv nastavení přesnosti na výsledek simulace je pro Bullet minimální, rozdíly lze vidět až na druhém desetinném místě, pokud vůbec. Pro ODE je vliv přesnosti větší, se vzrůstající přesností roste i čas potřebný k projetí dráhy.

Velký vliv na simulaci měla změna parametru dt. Délka simulace se prodlužovala v poměru se snižováním hodnoty dt (5x menší parametr dt měl za následek zhruba 5x delší čas simulace atd). Rovněž jeho vliv na výsledný čas byl relativně velký. Tento parametr měl společně se zvoleným „engine“ velký vliv na úspěšné projetí celé dráhy, což je dle mého názoru provázáno s tím, jak často se aktualizují data ze senzorů.

Zjištění pravděpodobně nelze aplikovat na všechny simulace prováděné v CoppeliaSim, nicméně lze předpokládat, že u podobně složitých simulací bude vliv těchto parametrů obdobný. Záleží na konkrétním případě, zda je rozdíl mezi zvolenými parametry důležitý nebo ne. V této simulaci je rozdíl mezi minimálním a maximálním výsledným časem 0,425 s, což v případně závodního „line followeru“ může být problém.

Parametry počítače, na kterém byly simulace prováděny:

- Operační systém: Windows 10 Pro
- Procesor: i5-9300H 2.40 GHz
- Operační paměť: 8 GB
- Grafická karta: GTX 1650 4 GB

## Závěr

Volba správného simulačního nástroje pro práci je asi tím nejdůležitějším úkolem, kterému člověk na počátku procesu návrhu robotického systému čelí. Pokud věnuje dostatek času pro porovnání a výběr vhodného nástroje, dokáže si tím ušetřit čas následnou efektivní prací. I to je jedním z důvodů vzniku této bakalářské práce.

Rozdílů mezi jednotlivými SW je mnoho. Z toho důvodu je nutné si před výběrem vhodného SW ujasnit informace o prováděné simulaci (nebo prováděných simulacích). Je potřeba zvážit, jaké typy robotů budou předmětem simulace, jaký bude počet simulovaných robotů, zda bude potřeba detailněji simulovat konkrétní technologii atd. Následně je třeba dle těchto informací co nejlépe zvolit simulační SW.

V současné době také existuje velké množství těchto softwarů. Řádově se jejich počet pohybuje ve vyšších desítkách. To je pravděpodobně způsobeno rozdílnými potřebami institucí a jednotlivců, zabývajících se robotikou. Software pro simulaci robotizovaných pracovišť potřebuje jiné funkce a vlastnosti než software pro simulaci velkého množství létajících mobilních robotů. Nedá se říct, že by se zatím objevil dokonalý SW, který by optimálně zvládal všechny kategorie simulací, a navíc ještě zcela vyhovoval po uživatelské stránce (optimalizace, podpora, kompatibilita atd.). Přesto se objevují v různých oblastech jistí favoriti. Pokud by se měl přeci jen vybrat „dokonalý software“, vhodným kandidátem na tuto pozici je simulační software V-Rep (CoppeliaSim) [1]. V několika srovnávacích publikacích se jeví jakožto nejvíce všestranný simulátor, který v porovnání s ostatními splňuje nejvíce nejčastějších požadavků na simulace v oblasti robotiky [2]. I tak má ovšem nedostatky. Například kvůli jeho relativně složité simulaci fyziky je jeho praktické využití omezeno na simulace o malém počtu robotů. Další nevýhodou může být vlastní formát ukládání scény. V porovnání s ostatními simulátory to jsou však relativně malé nedostatky. Při simulaci konkrétního technologického procesu s využitím PR může být vhodnější použití simulačních SW robotizovaných pracovišť, které se specializují na tuto aplikaci PR.

Během psaní této práce se systém V-Rep „přejmenoval“ na CoppeliaSim. Vzhledem k tomu, že se tento název ještě nezapsal do obecného povědomí a je zvyklostí nazývat ho V-Rep, je na něj takto odkazováno i zde.

V průběhu práce se také ukázalo, jak rozsáhlé je téma simulačních SW. Problémem při vyhledávání informací o těchto SW je, že mnoho důvěryhodných informací bylo vydáno již před několika lety a vzhledem k vývoji těchto SW už uvedené informace nemusí být zcela pravdivé. Z tohoto důvodu pro další, podrobnější a aktuální informace o stavu simulačních SW doporučuji sledovat weby vývojářů.

Bakalářská práce prezentuje znalosti nabyté v rámci řešení projektu: „Digitální dvojčata robotických systémů a jejich verifikace II“, reg. č.: „SP2020/141“ financovaných ze státního rozpočtu České republiky, na kterém jsem se v průběhu bakalářského studia podílel.

## **Poděkování**

Děkuji vedoucímu práce, Ing. Václavu Krysovi, Ph.D., za vedení práce, vstřícnost a ochotu pomoci i při ztížených podmínkách, doc. Ing. Zdenku Bobovskému, PhD. za odborné rady a návrhy a rodině za podporu během studia.

## Zdroje

- [1] ROHMER, E., S.P.N. SINGH a M. FREESE. V-REP: A versatile and scalable robot simulation framework. *IEEE/RSJ International Conference on Intelligent Robots and Systems* [online]. 2013, 1321-1326 [cit. 2020-04-03]. DOI: 10.1109/IROS.2013.6696520
- [2] IVALDI, S., J. PETERS, V. PADOIS a F. NORI. Tools for simulating humanoid robot dynamics: A survey based on user feedback. *IEEE-RAS International Conference on Humanoid Robots* [online]. 2014, 842-849 [cit. 2020-03-13]. DOI: 10.1109/HUMANOIDS.2014.7041462
- [3] Robotics simulator. In: *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2020 [cit. 2019-11-03]. Dostupné z: [https://en.wikipedia.org/wiki/Robotics\\_simulator](https://en.wikipedia.org/wiki/Robotics_simulator)
- [4] CHENG, Y., Z.-J. LIU a P. GUO. Design and implementation of 3D graphics rendering engine with directx. *International Conference on Communications and Information Processing* [online]. 2012, 600-607 [cit. 2020-04-03]. DOI: 10.1007/978-3-642-31968-6\_71
- [5] ALLEN, P. K. a A. T. MILLER. Grasplit! A versatile simulator for robotic grasping. *IEEE Robotics & Automation Magazine* [online]. 2004, 110-122 [cit. 2020-03-14]. DOI: 10.1109/MRA.2004.1371616
- [6] PITONAKOVA, L., M. GIULIANI, A. PIPE a A. WINFIELD. Feature and performance comparison of the V-REP, Gazebo and ARGoS robot simulators. *Lecture Notes in Computer Science* [online]. 2018, 357-368 [cit. 2020-03-04]. DOI: 10.1007/978-3-319-96728-8\_30.
- [7] HARRIS, A. a J. M. CONRAD. Survey of popular robotics simulators, frameworks, and toolkits. *Proceedings of IEEE Southeastcon* [online]. 2004, 243-249 [cit. 2020-03-15]. Dostupné z: DOI: 10.1109/SECON.2011.5752942
- [8] EREZ, T., Y. TASSA a E. TODOROV. Simulation tools for model-based robotics: Comparison of Bullet, Havok, MuJoCo, ODE and PhysX. *IEEE International Conference on Robotics and Automation* [online]. 2015, 4397-4404 [cit. 2020-04-11]. DOI: 10.1109/ICRA.2015.7139807.
- [9] KOENIG, N. a A. HOWARD. Design and use paradigms for Gazebo, an open-source multi-robot simulator. *International Conference on Intelligent Robots and Systems* [online]. 2004, 2149-2154 [cit. 2020-04-11].
- [10] KREJČÍ, T. *Vytvoření robotizovaného pracoviště v simulačním prostředí RobotExpert*. Diplomová práce (Ing.). VŠB-TU Ostrava, Fakulta strojní, Katedra robotiky. 2019, 122 s.
- [11] DAŠKOVÁ, M. *Tvorba digitálního dvojčete robotické buňky v simulačním prostředí Tecnomatix*. Diplomová práce (Ing.). VŠB-TU Ostrava, Fakulta strojní, Katedra robotiky. 2019, 80 s.
- [12] *RoboDK* [online]. RoboDK, 2020 [cit. 2020-05-06]. Dostupné z: <https://robodk.com/>
- [13] *ROBODK SIMULATION SOFTWARE* [online]. Universal Robots, 2020 [cit. 2020-05-06]. Dostupné z: <https://www.universal-robots.com/plus/urplus-components/software/robodk-simulation-software/>
- [14] Visual Components. *Visual Components* [online]. SoliCAD, 2018 [cit. 2020-05-08]. Dostupné z: <http://solicad.com/h/visual-components>

- [15] Visual Components. *Robotizace* [online]. SoliCAD, 2018 [cit. 2020-05-08].  
Dostupné z: <http://solicad.com/c/visual-components-robotizace>
- [16] Delmia Robotics. *Disciplines* [online]. Dassault Systèmes, 2020 [cit. 2020-05-08].  
Dostupné z: <https://www.3ds.com/products-services/delmia/disciplines/industrial-engineering/tag/116-164/>
- [17] MEYER, J., A. SENDOBRY, S. KOHLBRECHER, U. KLINGAUF a O. VON STRYK. Comprehensive simulation of quadrotor UAVs using ROS and Gazebo. *3rd Lecture Notes in Computer Science* [online]. 2012, 400-411 [cit. 2020-03-09]. DOI: 10.1007/978-3-642-34327-8\_36
- [18] BIBER, P. a U. WEISS. Plant detection and mapping for agricultural robots using a 3D LIDAR sensor. *Robotics and Autonomous Systems* [online]. 2011, 265-273 [cit. 2020-03-03]. DOI: 10.1016/j.robot.2011.02.011
- [19] FURRER, F., M. BURRI, M. ACHELNIK a R. SIEGWART. RotorS—A modular gazebo MAV simulator framework. *Studies in Computational Intelligence* [online]. 2016, 595-625 [cit. 2020-04-11]. DOI: 10.1007/978-3-319-26054-9\_23.
- [20] JIA, B., K. ZHANG a J. CHEN. Trifocal Tensor-Based Adaptive Visual Trajectory Tracking Control of Mobile Robots. *IEEE Transactions on Cybernetics* [online]. 2017, (11), 3784-3798 [cit. 2020-03-09]. DOI: 10.1109/TCYB.2016.2582210
- [21] MIGUEL, A., S. KANNAN, H. VOOS a M. A. OLIVARES-MENDEZ. Vision based fuzzy control autonomous landing with UAVs: From V-REP to real experiments. *Mediterranean Conference on Control and Automation* [online]. 2015, 14-21 [cit. 2020-02-24]. DOI: 10.1109/MED.2015.7158723
- [22] IJSPEERT, A.J., A. MARTINOLI, A. BILLARD a L. M. GAMBARDELLA. Collaboration through the exploitation of local interactions in autonomous collective robotics: The stick pulling experiment. *Autonomous Robots* [online]. 2001, (2), 149-171 [cit. 2020-03-09]. DOI: 10.1023/A:1011227210047
- [23] HAYES, A.T., A. MARTINOLI a R. M. GOODMAN. Swarm robotic odor localization. *IEEE International Conference on Intelligent Robots and Systems* [online]. 2001, (2), 1073-1078 [cit. 2020-03-12]. DOI: 10.1109/IROS.2001.976311
- [24] MICHEL, O. Webots: Symbiosis between virtual and real mobile robots. *Lecture Notes in Computer Science* [online]. 1998, (1434), 254-263 [cit. 2020-03-12]. ISSN 03029743. Dostupné z: databáze Scopus
- [25] CRUZ, L., L. P. REIS, N. LAU a A. SOUSA. Optimization Approach for the Development of Humanoid Robots' Behaviors. *Lecture Notes in Artificial Intelligence* [online]. 2012, (7637), 491-500 [cit. 2020-03-08]. ISSN 0302-9743. Dostupné z: databáze Web of Science Core Collection
- [26] SHAFFI, N., L. P. REIS a R. J. F. ROSSETTI. Two humanoid simulators: Comparison and synthesis. *Sistemas e Tecnologias de Informacao* [online]. 2011, (1), 773 [cit. 2020-03-08]. Dostupné z: databáze Scopus
- [27] ABREU, M., L. P. REIS, N. LAU a A. SOUSA. Learning low level skills from scratch for humanoid robot soccer using deep reinforcement learning. *IEEE International Conference on Autonomous Robot Systems and Competitions ICARSC* [online]. 2019, 256-263 [cit. 2020-03-08]. DOI: 10.1109/ICARSC.2019.8733632

- [28] COOK, D., A. VARDY a R. LEWIS. A survey of AUV and robot simulators for multi-vehicle operations. *IEEE/OES Autonomous Underwater Vehicles (AUV)* [online]. 2014, 1-8 [cit. 2020-03-08]. DOI: 10.1109/AUV.2014.7054411.
- [29] HENRIKSEN, E. H., I. SCHJØLBERG a T. B. GJERSVIK. UW MORSE: The underwater Modular Open Robot Simulation Engine. *IEEE/OES Autonomous Underwater Vehicles (AUV)* [online]. 2016, 261-267 [cit. 2020-03-08]. DOI: 10.1109/AUV.2016.7778681.
- [30] DEMARCO, K. J., M. E. WEST a A. M. HOWARD. A simulator for Underwater Human-Robot Interaction scenarios. *OCEANS - San Diego* [online]. 2013, 1-10 [cit. 2020-03-08]. DOI: 10.23919/OCEANS.2013.6741147.
- [31] KEHOE, B., A. MATSUKAWA, S. CANDIDO, J. KUFFNER a K. GOLDBERG. Cloud-based robot grasping with the google object recognition engine. *IEEE International Conference on Robotics and Automation ICRA* [online]. 2013, 4263-4270 [cit. 2020-03-08]. DOI: 10.1109/ICRA.2013.6631180.
- [32] ORSAG, M., P. OH a C. KORPELA. Modeling and control of MM-UAV: Mobile manipulating unmanned aerial vehicle. *Journal of Intelligent & Robotic Systems* [online]. 2013, (1-4), 93-101 [cit. 2020-03-08]. DOI: 10.1007/s10846-011-9591-3.
- [33] RUSU, R. B., A. HOLZBACH, R. DIANKOV, G. BRADSKI a M. BEETZ. Perception for mobile manipulation and grasping using active stereo. *HUMANOIDS09* [online]. 2009, 632-638 [cit. 2020-03-08]. DOI: 10.1109/ICHR.2009.5379597.
- [34] NAKAOKA, S., A. NAKAZAWA, K. YOKOI, H. HIRUKAWA a K. IKEUCHI. Generating whole body motions for a biped humanoid robot from captured human dances. *IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION* [online]. 2003, (3), 3905-3910 [cit. 2020-03-08]. DOI: 10.1109/ROBOT.2003.1242196.
- [35] SULEIMAN, W., E. YOSHIDA, F. KANEHIRO, J. LAUMOND a A. MONIN. On human motion imitation by humanoid robot. *IEEE International Conference on Robotics and Automation* [online]. 2008, (3), 2697-2704 [cit. 2020-03-08]. DOI: 10.1109/ROBOT.2008.4543619.
- [36] KANEHIRO, F., et al. The first humanoid robot that has the same size as a human and that can lie down and get up. *IEEE International Conference on Robotics and Automation* [online]. 2003, 1633-1639 [cit. 2020-04-11]. DOI: 10.1109/ROBOT.2003.1241828.
- [37] BILEK, O., D. SAMEK, J. KNEDLOVA. Offline programming for robotic deburring process of aluminium wheels. *Manufacturing Technology* [online]. 2013, (3), 269-275 [cit. 2020-05-03]. ISSN 12132489.
- [38] HEYDARYAN, S., J.S. BEDOLLA a G. BELINGARDI. Safety design and development of a human-robot collaboration assembly process in the automotive industry. *Applied Sciences (Switzerland)* [online]. 2018, (3), článek 344 [cit. 2020-05-03]. DOI: 10.3390/app8030344.
- [39] GUERRERO, L.V., V.V. LÓPEZ a J.E. MEJÍA. Virtual Commissioning with Process Simulation (Tecnomatix). *Computer-Aided Design and Applications* [online]. 2014, , S11-S19 [cit. 2020-05-03]. DOI: 10.1080/16864360.2014.914400.

- [40] CAPUTO, F., A. GRECO, M. FERA, G. CAIAZZO a S. SPADA. Simulation techniques for ergonomic performance evaluation of manual workplaces during preliminary design phase. *Advances in Intelligent Systems and Computing* [online]. 2019, , 170-180 [cit. 2020-05-03]. DOI: 10.1007/978-3-319-96077-7\_18.
- [41] POLDER, J., Z. PAN, N. LARKIN, S. VAN DUIN a J. NORRISH. Offline programming for a complex welding system using DELMIA automation. *Lecture Notes in Electrical Engineering* [online]. 2003, 341-349 [cit. 2020-03-08]. DOI: 10.1007/978-3-642-19959-2\_42.
- [42] GADALETA, M., G. BERSELLI a M. PELLICCIARI. Energy-optimal layout design of robotic work cells: Potential assessment on an industrial case study. *Robotics and Computer-Integrated Manufacturing* [online]. 2017, (47), 102-111 [cit. 2020-03-08]. DOI: 10.1016/j.rcim.2016.10.002.
- [43] DÀVILA-RÌOS, I., L. M. TORRES-TREVINO a I. LÒPEZ-JUÀREZ. On the implementation of a robotic welding process using 3D simulation environment. *Robotics and Automotive Mechanics Conference, CERMA 2008* [online]. 2008, 283-287 [cit. 2020-03-08]. DOI: 10.1109/CERMA.2008.71.
- [44] C. CONNOLLY. Technology and applications of ABB RobotStudio. *Robotics and Automotive Mechanics Conference, CERMA 2008* [online]. 2009, (6), 283-287 [cit. 2020-03-08]. DOI: 10.1108/01439910910994605.
- [45] FANG, D., S. DENG a H. LIAO. The effect of robot kinematics on the coating thickness uniformity. *Journal of Thermal Spray Technology* [online]. 2010, (4), 796-804 [cit. 2020-03-08]. DOI: 10.1007/s11666-010-9470-7.
- [46] MAIOLINO, P., R. WOOLLEY, D. BRANSON, P. BENARDOS, A. POPOV a S. RATCHEV. Flexible robot sealant dispensing cell using RGB-D sensor and off-line programming. *Robotics and Computer-Integrated Manufacturing* [online]. 2010, , 188-195 [cit. 2020-03-0]. DOI: 10.1016/j.rcim.2017.04.004.
- [47] Dynamics. *Coppelia Robotics* [online]. [cit. 2020-05-01]. Dostupné z: <https://www.coppeliarobotics.com/helpFiles/en/dynamicsModule.htm>

## **Přílohy**

Příloha A: simulacni\_software.xlsx – Dodatečné grafy a tabulková data.