

Simulační studie

Implementace abstraktního modelu samohybné kanónové houfnice
ShKH vz. 77 „Dana“

Balistika ve vojenství

4. prosince 2022

Martin Pech (xpechm00)
David Konečný (xkonec83)

1 Úvod

Tématem tohoto projektu je implementace abstraktního modelu [3, snímek 10] samohybné kanónové houfnice ShKH vz. 77 „Dana“. Projekt vznikl jako zápočtová práce do předmětu IMS, která má prakticky prověřit poznatky osvojené v tomto kurzu. Na základě modelu bude demonstrováno chování této houfnice v případě bojové pohotovosti. Pro daný simulační model [3, snímek 44] uvažujeme, že toto samohybné dělo lze ovládat pouze v režimu manuální střelby. Jelikož se ve vojenství vyskytuje mnoho náhodných a neovlivnitelných jevů, které není možné v tomto modelu zachytit, budeme pro jednoduchost uvažovat pouze některé (popsáno v kapitole 3.1).

1.1 Autoři a původ faktů

Na implementaci projektu a tvorbě technické zprávy spolupracovali studenti třetího ročníku David Konečný a Martin Pech. Odborná fakta a údaje o manipulaci, údržbě, jízdních vlastnostech a střelbě daného typu houfnice jsou oficiální údaje převzaté z webových stránek Armády České republiky a Ministerstva obrany České republiky.

1.2 Validace výstupů

Jelikož je tato houfnice dostupná pouze jako inventář Ministerstva obrany, není možné pro účely tohoto simulačního modelu validovat přesně to, co tento model zkoumá. Jelikož ale Armáda ČR používá toto vozidlo více než 40 let (a to v mnoha různých verzích), lze za validní data považovat technické parametry vozidla, uvedené na webu Armády ČR. Zkombinujeme-li tato data pro účely tohoto simulačního modelu, dá se předpokládat, že jsou výstupy tohoto modelu validní.

2 Rozbor tématu a použitých metod/technologií

Samohybná kanónová houfnice Dana (dále jen „houfnice Dana“) je dělostřelecká zbraň ve verzích s automatickým, nebo manuálním nabíjením. Pro účely tohoto modelu uvažujeme pouze verzi s manuálním nabíjením. „Vyznačuje se velkým dostřelem, přesností a rychlostí střelby. Patří do kategorie vojenská bojová vozidla.“ [1]. Dále se vyznačuje rychlostí jízdy až 80 km/h po zpevněném povrchu (a to až do vzdálenosti 800 km) a 25 km/h při jízdě v terénu (do vzdálenosti 400 km). Osádku tvoří pětičlenná skupina vojáků, která tvoří jednotku. Při palbě s manuálním nabíjením dosahuje rychlosti střelby až dvou výstřelů za minutu. Zajímavostí je, že v případě verze s automatickým nabíjením se rychlost střelby více než zdvojnásobí až na 5 výstřelů za minutu. [4]

V tomto simulačním modelu uvažujeme situaci, kdy je v základním táboře odstaven uživatelem zadaný počet houfnic Dana. Jakmile je vyhlášena bojová pohotovost a je vytvořen požadavek na palbu, přesouvá se jedna dostupná houfnice na bojiště, kde po určitou dobu vykonává činnost spojenou se svým úkolem. Na bojiště se takto může přesunout 1 – N houfnic podle toho, kolik dostane základní tábor požadavků na palbu. V případě, že houfnice dokončí svou činnost na bojišti (uvažujeme situaci, kdy byly zničeny všechny nepřátelské cíle, nebo houfnice vystřílela veškerou munici), vrací se houfnice zpět do základního tábora, kde probíhá její příprava na další akci. Houfnici lze znovu použít až ve chvíli, kdy je plně přezbrojena. U houfnice, která se nachází na bojišti, může dojít k poruše děla. V takovém případě je houfnici potřeba dostat do základního tábora, kde podstoupí opravu. Její místo musí nahradit jiná dostupná houfnice.

Houfnice nemůže fungovat bez obsluhy, tudíž je vždy potřeba příslušný počet vojáků pro její obsluhu a případnou opravu. Pro obsluhu houfnice je potřeba jedna jednotka. Jednotky vojáků mohou z války dezertovat (zběhnout) s výchozí pravděpodobností 0, 9⁵%, nebo s mírou, kterou zadá uživatel simulace. [2]

2.1 Popis použitých postupů

Pro tvorbu základní koncepce tohoto projektu bylo využito deklarativních modelů [3, snímek 49], konkrétně Petriho sítě [3, snímek 123] vyobrazené níže. Pomocí Petriho sítě byla vytvořena obslužná linka simulující chování samohybné houfnice Dana a obsluhujících vojáků. Pro následnou tvorbu simulačního modelu (abstraktního modelu zapsaného formou programu) bylo využito jazyku C++ a knihovny SIMLIB. Zde bylo nutné důkladně nastudovat vlastnosti a chování knihovny SIMLIB, zejména pro správnou práci s modelováním náhodných procesů [3, snímek 72].

2.2 Popis původu použitých metod/technologií

Jak již bylo zmíněno, pro implementaci byl použit jazyk C++ v kombinaci s knihovnou SIMLIB ve verzi 3.09¹. S pomocí tohoto jazyka bylo přistupováno k rozhraní knihovny SIMLIB, pomocí kterého byly vytvořeny předpisy pro všechny komponenty potřebné k simulaci. Pro sestavení programu byl použit GNU Make².

¹<https://www.fit.vutbr.cz/~peringer/SIMLIB/>

²<https://www.gnu.org/software/make/>

3 Koncepce modelu

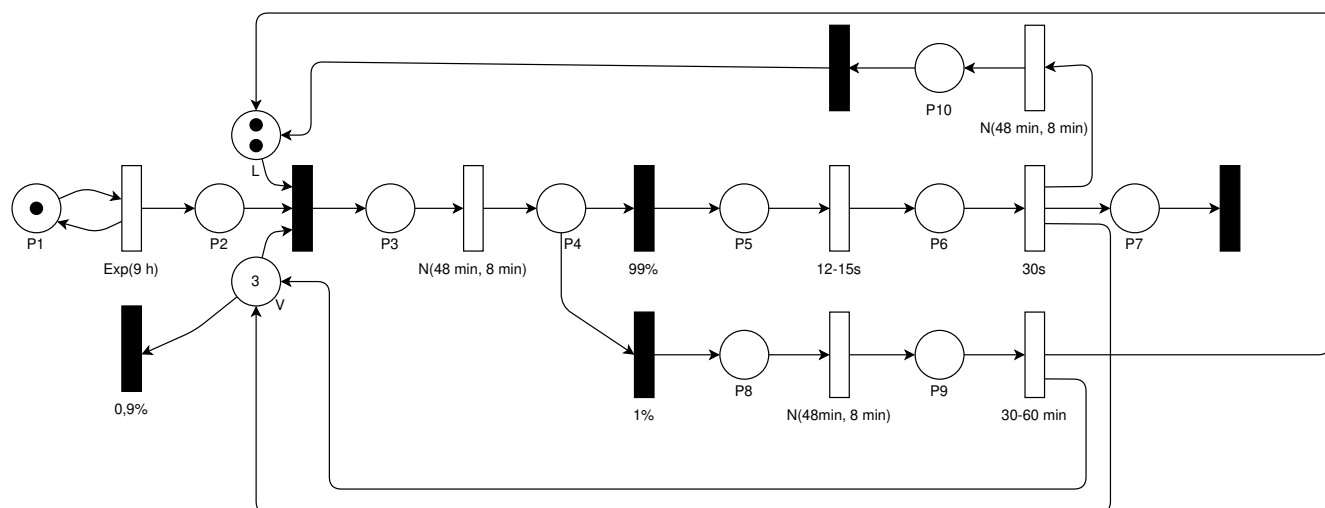
Konceptuální model [3, snímek 48] je koncipován jako systém hromadné obsluhy [3, snímek 136]. Naše dva zdroje – jednotky vojáků a houfnice jsou vyjádřeny obslužnými linkami. Houfnice představuje obslužná linka s kapacitou 2, jednotky vojáků obslužná linka s kapacitou 3. Do modelu vstupuje ještě jeden činitel – dezerce, míra dezerce jednoho vojáka je stanovena na 0,9%. Dezertovat může pouze celá jednotka.

3.1 Popis konceptuálního modelu

Do modelu vstupují požadavky na palbu, které se hromadí ve frontě, pokud je k dispozici volná houfnice a jednotka vojáků, je možné houfnici dopravit ze základního tábora na bojiště, nabít ho a vypálit, houfnice i s jednotkou vojáků se poté vrací zpět do základního tábora, kde je opět připraven, stejně jako jednotka vojáků, obsloužit další požadavek na palbu. Při nabíjení však může nastat porucha, v tom případě je houfnice dopravena do základního tábora, kde proběhně její oprava, po dokončení opravy je houfnice i jednotka vojáků opět připravena obsloužit další požadavek na palbu.

3.2 Forma konceptuálního modelu

Model je reprezentován Petriho sítí, která je zobrazena níže.



Obrázek 1: Petriho síť

P_1	Generování požadavků
P_2	Fronta požadavků
P_3	Přesun na bojiště
P_4	Příprava na nabíjení
P_5	Nabíjení
P_6	Palba
P_7	Odchod požadavků ze systému
P_8	Přesun zpět do tábora (z důvodu poruchy)
P_9	Oprava houfnice
P_{10}	Přesun zpět do tábora
L	Linka houfnic
V	Linka jednotek vojáků

Tabulka 1: Legenda k Petriho síti

4 Architektura simulačního modelu

Spuštěním simulačního modelu [3, snímek 44] se spustí simulační experiment se zadanými parametry, jejich zadávání je popsáno v kapitole 4.2. Během simulačního experimentu jsou vypisovány všechny změny stavu, výpis každé změny stavu je doprovázen modelovým časem [3, snímek 21]. Příkladem změny stavu může být přesun houfnice s jednotkou

na bojiště, palba, oprava houfnice a další. Stavby jsou znázorněny v Petriho síti, která reprezentuje konceptuální model v kapitole 3.2.

Pro účely simulace byla vzdálenost základního tábora od bojiště na 20 km, tomu odpovídají i hodnoty doby přesunu, které vycházejí z reálných hodnot popsaných v kapitole 2.

Simulační experiment ve výchozím stavu běží tak, aby doba trvání experimentu odpovídala jednomu dni v reálném čase [3, snímek 21]. Modelový čas [3, snímek 21] lze nastavit způsobem popsaným v kapitole 4.2. Jednotka modelového času je chápána jako minuta v reálném čase. Dále zde pracujeme s poruchovostí 1%, která je také nastavitelná, jelikož se může měnit na základě různých vnějších vlivů, které byly v simulaci zanedbány.

4.1 Mapování konceptuálního modelu do simulačního modelu

Generování požadavků palby je reprezentováno třídou *Generátor*, která je implementována jako událost [3, snímek 169]. Proces přesunu na bojiště, nabíjení, palby a návratu zpět, je implementována třídou *Palba*. Dalším samostatným procesem je porucha (včetně přesunu zpět do tábora a následné opravy), ta je reprezentována třídou *Porucha*. Proces dezerce jednotek vojáků je reprezentován třídou *Dezerce*.

4.2 Spouštění simulačního modelu

Simulační experiment je možné spouštět s volitelnými parametry, pokud není parametr zadán, použije se výchozí hodnota.

- Parametr `-t` umožňuje nastavit modelový čas, jeho výchozí hodnota je 1440, tedy 1 den.
- Parametr `-d` umožňuje nastavit procento dezerce (pro jednoho vojáka), výchozí hodnota je 0,95%.
- Parametr `-f` umožňuje nastavit intenzitu příchodu požadavků na palbu, výchozí hodnota je 540, tedy 9 hodin.
- Parametr `-p` umožňuje nastavit poruchovost (v procentech), výchozí hodnota je 1%.

U všech parametrů je vyžadováno kladné číslo.

Počet houfnic a počet jednotek vojáků je možné měnit změnou konstanty `POCET_RAKETOMETU` či `POCET_VOJAKU` v souboru `ballistics.cpp`, poté je ještě nutné program znovu sestavit pomocí příkazu `make`.

5 Podstata simulačních experimentů a jejich průběh

Cílem experimentů bylo pozorování chování modelu. Účelem bylo zjistit, na základě chování modelu, optimální vytížení obslužných linek.

5.1 Postup experimentování

Samotný experiment byl vždy spuštěn se zadanými parametry, následně byly výsledky každého experimentu zaneseny do tabulky. Z výstupů experimentu byl poté vyvozen závěr.

5.2 Experimenty

Každý experiment má svůj cíl, závěr a nachází se u něj výpis jeho výsledků.

5.2.1 Experiment 1

Cílem tohoto experimentu je ověření funkčnosti modelu, jsou použity výchozí hodnoty, tedy intenzita požadavků s časem 9 hodin (540 minut) a doba trvání experimentu 1 den (1440 minut).

Intenzita požadavků [min]	Celkový čas [min]	Bez čekání	S čekáním	Celkem
540	1440	5	0	5

Tabulka 2: Výsledky experimentu 1

Z výsledků experimentu vidíme, že všechny požadavky byly obslouženy, dokonce bez čekání. Nabízí se zde možnost existence rezervy, tzn. že systém by mohl zpracovávat více požadavků bez čekání.

5.2.2 Experiment 2

V tomto experimentu vyzkoušíme, jak se chová systém při vyšším vytížení, intenzita požadavků byla nastavena na 30 minut.

Intenzita požadavků [min]	Celkový čas [min]	Bez čekání	S čekáním	Celkem
30	1440	2	52	54

Tabulka 3: Výsledky experimentu 2

Z tabulky 3 vidíme, že drtivá většina požadavků musela na svoji obsluhu čekat, systém byl tedy vytížen poměrně hodně.

5.2.3 Experiment 3

V tomto experimentu si klademe za cíl najít optimální hodnotu intenzity požadavků, při které systém ještě zpracovává požadavky bez čekání. Postupným experimentováním byla získána hodnota intenzity požadavků 130 minut.

Intenzita požadavků [min]	Celkový čas [min]	Bez čekání	S čekáním	Celkem
130	1440	0	8	8

Tabulka 4: Výsledky experimentu 3

Zde vidíme, že systém zpracoval více požadavků než v experimentu 1 v kapitole 5.2.1, zároveň však byly stále všechny požadavky obslouženy bez čekání.

5.2.4 Experiment 4

V tomto experimentu budeme vycházet z hodnot a výsledků experimentu 3 v kapitole 5.2.3 a vyzkoušíme, zda jsou dobře využitelné i v delším časovém intervalu, v tomto případě 7 dní (10080 minut).

Intenzita požadavků [min]	Celkový čas [min]	Bez čekání	S čekáním	Celkem
130	10080	65	12	77

Tabulka 5: Výsledky experimentu 4

Z výsledků je patrné, že počet požadavků, které musely na svoji obsluhu čekat, narostl, stále však byla většina požadavků obsloužena bez čekání.

5.2.5 Experiment 5

Cílem posledního experimentu bylo nalezení optimální hodnoty intenzity požadavků, při které systém ještě zpracovává požadavky bez čekání podobně jako v experimentu 3 v kapitole 5.2.3. Nalezená hodnota se nachází níže v tabulce 6.

Intenzita požadavků [min]	Celkový čas [min]	Bez čekání	S čekáním	Celkem
260	10080	37	0	37

Tabulka 6: Výsledky experimentu 5

Vidíme, že žádný z požadavků na svoje obslužení nečekal, na rozdíl od experimentu 4 v kapitole 5.2.4 se však snížil počet všech obsloužených požadavků. Jelikož experiment probíhal v intervalu jednoho týdne, na nižší počet všech obsloužených požadavků měla vliv i poruchovost a dezerce.

5.3 Závěry experimentů

Bylo provedeno 5 experimentů s různými parametry. V prvním experimentu jsme ověřili funkčnost modelu. V dalších experimentech jsme našli časovou hodnotu intenzity požadavků, tak aby bylo vytížení obslužných linek optimální – žádný požadavek nemusel na svoje obslužení čekat. Při experimentování s větším časovým intervalem jsme zjistili, že při optimálním vytížení obslužných linek může být celkový počet obsloužených požadavků nižší než počet obsloužených požadavků bez čekání při neoptimálním vytížení obslužných linek. Může být tedy výhodné spokojit se

s určitým počtem požadavků, které budou na svoji obsluhu čekat, za účelem vyššího počtu obsloužených požadavků bez čekání, což nám ukázal experiment 4 v kapitole 5.2.4.

Experimenty lze považovat s dostatečnou věrohodností za správný zdroj informací, protože jich bylo provedeno několik a vychází z reálných dat.

6 Shrnutí simulačních experimentů a závěr

V rámci simulačních experimentů jsme ověřili funkčnost modelu a jeho vztah mezi simulací a matematickými výpočty. Studií provedenou na modelu, bylo zjištěno, že systém vyžaduje pro optimální využití obslužných linek vhodně zvolenou intenzitu požadavků palby. Vhodná intenzita se liší pro časový interval, ve kterém experiment probíhal.

V rámci projektu vznikl nástroj, který vychází z reálných dat dostupných na webových stránkách Armády České republiky a Ministerstva obrany České republiky. Nástroj byl implementovaný v jazyce C++ za použití knihovny SIMLIB a umožňuje spouštět simulační experimenty s různými parametry. Výstupem jsou hodnoty obsloužených požadavků a informace zatížení obslužných linek.

Použitá literatura

- [1] ARMÁDA ČESKÉ REPUBLIKY: 152 MM SAMOHYBNÁ KANÓNOVÁ HOUFNICE. [online], [vid. 2022-12-03]. Dostupné z: <https://acr.army.cz/technika-a-vyzbroj/delo/152-mm-samohybna-kanonova-houfnice-93164/>
- [2] Hinzman, J.: Army desertion up 80 percent since Iraq war. [online], [vid. 2022-12-03]. Dostupné z: <https://www.nbcnews.com/id/wbna21836566>
- [3] Peringer, P.; Hrubý, M.: Modelování a simulace, Text k přednáškám kursu Modelování a simulace na FIT VUT v Brně. [online], 15. září 2022, [vid. 2022-12-03]. Dostupné z: <https://www.fit.vutbr.cz/study/courses/IMS/public/prednasky/IMS.pdf>
- [4] Smíšek, M.: CZK - vz. 77 DANA (152mm samohybná kanónová houfnice). [online], [vid. 2022-12-03]. Dostupné z: <https://www.valka.cz/CZK-vz-77-DANA-152mm-samohybna-kanonova-houfnice-t12305>