



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ**

**ÚSTAV AUTOMATIZACE A MĚŘICÍ TECHNIKY**

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION  
DEPARTMENT OF CONTROL AND INSTRUMENTATION

## **PROGRAMY PRO VÝPOČET NEJISTOTY MĚŘENÍ METODOU MONTE CARLO**

PROGRAMS FOR CALCULATING MEASUREMENT UNCERTAINTY USING MONTE CARLO  
METHOD

**DIPLOMOVÁ PRÁCE**

MASTER'S THESIS

**AUTOR PRÁCE**

AUTHOR

**Bc. MAREK NOVOTNÝ**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**Ing. SOŇA ŠEDIVÁ, Ph.D.**

BRNO 2015



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav automatizace a měřicí techniky

# Diplomová práce

magisterský navazující studijní obor  
Kybernetika, automatizace a měření

**Student:** Bc. Marek Novotný

**ID:** 136565

**Ročník:** 2

**Akademický rok:** 2014/2015

## NÁZEV TÉMATU:

**Programy pro výpočet nejistoty měření metodou Monte Carlo**

## POKYNY PRO VYPRACOVÁNÍ:

Cílem diplomové práce je vytvoření programů pro výpočet nejistoty měření metodou Monte Carlo v různých softwarových prostředích.

1. Provedte literární rešerši v oblasti stanovení nejistot měření klasickým postupem podle metodiky GUM a metodou Monte Carlo.
2. Provedte posouzení generátorů náhodných čísel v softwarech, ve kterých je možné naprogramovat výpočet nejistoty měření metodou Monte Carlo - Excel, LabVIEW, Matlab, GNU Octave, atd.
3. Navrhněte metodiku výpočtu nejistoty nepřímého měření pomocí metody Monte Carlo. Přesné zadání ukázkového příkladu nepřímého měření zadá vedoucí práce.
4. Realizujte program pro výpočet nejistoty měření metodou Monte Carlo ve vybraných softwarech (minimálně tři).
5. Ověřte funkci programů pro konkrétní příklad výpočtu nejistoty nepřímého měření.
6. Zhodnoťte dosažené výsledky.

## DOPORUČENÁ LITERATURA:

PALEŇČÁR, R. - VDOLEČEK, F. - HALAJ, M.: Nejistoty v měření I. až V., soubor článků v časopise AUTOMA, č. 7-8/2001, č. 10/2001, č. 12/2001, č. 4/2002 a č. 5/2002

**Termín zadání:** 9.2.2015

**Termín odevzdání:** 18.5.2015

**Vedoucí práce:** Ing. Soňa Šedivá, Ph.D.

**Konzultanti diplomové práce:**

**doc. Ing. Václav Jirsík, CSc.**

*Předseda oborové rady*

## UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

# Abstrakt

Diplomová práce se zabývá stanovováním nejistot nepřímých měření. Zaměřuje se především na generátory náhodných čísel v softwarech umožňujících výpočet nejistot měření metodou Monte Carlo. Dále se zaměřuje na výpočet nejistot nepřímého měření, jak metodou Monte Carlo, tak i klasickou numerickou metodou. Praktická část práce pojednává o ověření náhodnosti generátorů čísel obsažených v různých softwarech. Dále se zabývá stanovením nejistot nepřímého měření proudu oběma výše zmiňovanými metodami a následným porovnáním a zhodnocením dosažených výsledků.

## Klíčová slova

Nepřímé měření, metoda Monte Carlo, nejistota měření, generátor náhodných čísel, generátor pseudonáhodných čísel, Matlab, LabVIEW, GNU Octave.

## Abstract

The thesis deals with establishing uncertainties of indirect measurements. It focuses primarily on random number generators in software enabling the calculation of measurement uncertainties using Monte Carlo. Then it focuses on the uncertainty calculation indirect measurement as the Monte Carlo method and the classical numerical method. The practical part deals with the verification of randomness generators numbers contained in various softwares. It also deals with the determination of uncertainties indirect current measurements by both above-mentioned methods and then comparing and evaluating the values achieved.

## Keywords

Indirect measurement, Monte Carlo method, uncertainty measurement, random number generator, pseudo-random generator, Matlab, LabVIEW, GNU Octave.

## **Bibliografická citace:**

NOVOTNÝ, M. *Programy pro výpočet nejistoty měření metodou Monte Carlo*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2015. 79 s. Vedoucí práce Ing. Soňa Šedivá, Ph.D..

## Prohlášení

„Prohlašuji, že svoji diplomovou práci na téma Programy pro výpočet nejistoty měření metodou Monte Carlo jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nesáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících porušení autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. Díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: .....

.....

podpis autora

# Poděkování

Děkuji vedoucí diplomové práce Ing. Soni Šedivé, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování diplomové práce.

V Brně dne: .....

.....

podpis autora

# Obsah

1	Úvod.....	11
2	Nejistoty měření .....	12
2.1	Základní pojmy ze statistiky .....	12
2.1.1	Základní termíny .....	12
2.1.2	Rozdělení pravděpodobnosti .....	14
2.2	Typy nejistot .....	17
2.3	Zdroje nejistot měření.....	18
2.4	Nejistoty přímých měření .....	19
2.4.1	Standardní nejistota typu A .....	19
2.4.2	Standardní nejistota typu B .....	20
2.4.3	Kombinovaná standardní nejistota měření .....	21
2.4.4	Rozšířená standardní nejistota.....	22
2.5	Nejistoty nepřímých měření .....	22
2.6	Zásady pro vyjadřování nejistot.....	23
2.6.1	Vyjádření výsledku s použitím kombinované nejistoty .....	23
2.6.2	Vyjádření výsledku s použitím rozšířené nejistoty .....	23
2.6.3	Bilanční tabulka.....	24
3	Metoda Monte Carlo .....	25
3.1	Oblasti využití metody Monte Carlo .....	25
3.2	Výpočet nejistot pomocí metody Monte Carlo.....	25
3.3	Výhody a nevýhody metody Monte Carlo.....	26
3.4	Software vhodný pro metodu Monte Carlo .....	26
4	Generátory náhodných čísel .....	28
4.1	Generátory náhodných čísel .....	28
4.2	Generátory pseudonáhodných čísel .....	28
4.2.1	Generátory pseudonáhodných čísel s uniformním rozdělením pravděpodobnosti .....	29
4.2.2	Generátory pseudonáhodných čísel s obecným rozdělením pravděpodobnosti .....	32
5	Testování generátorů pseudonáhodných čísel.....	34
5.1	Statistické testy pseudonáhodných čísel.....	34
5.1.1	Frekvenční test .....	35
5.1.2	Blokový frekvenční test.....	35
5.1.3	Test hodnosti binární matice .....	35
5.1.4	Test sérií .....	35

5.1.5	Test shody nepřekrývajících se řetězců.....	36
5.1.6	Test shody překrývajících se řetězců.....	36
5.1.7	Test lineární složitosti .....	37
5.1.8	Spektrální test.....	37
6	Testování Vybraných generátorů čísel .....	39
6.1	MATLAB R2013b.....	40
6.2	Microsoft Excel 2013 .....	42
6.3	Maple 17.....	44
6.4	LabVIEW 2013.....	45
6.5	GNU Octave .....	46
6.6	Zhodnocení testů.....	48
7	Výpočet nejistoty nepřímého měření proudu numerickou metodou .....	49
7.1	Výpočet hodnoty měřeného proudu.....	50
7.2	Výpočet dílčích nejistot měření .....	50
7.3	Výsledek měření .....	54
8	Výpočet nejistoty nepřímého měřeného proudu metodou Monte Carlo .....	55
8.1	Výpočet nejistot měření metodou Monte Carlo v programu Matlab 2010b.....	57
8.2	Výpočet nejistot měření metodou Monte Carlo v programu LabVIEW 2013.....	60
8.3	Výpočet nejistot měření metodou Monte Carlo v programu GNU Octave .....	62
9	Porovnání dosažených výsledků výpočtů nejistot.....	64
9.1	Porovnání výpočtů proudu protékajícího měřicím rezistorem.....	64
9.2	Porovnání výpočtů směrodatné odchylky proudu .....	64
9.3	Porovnání výpočtů intervalů pokrytí u jednotlivých metod výpočtu.....	65
9.4	Porovnání výsledků z hlediska náročnosti jednotlivých metod a programů.....	66
10	Závěr.....	67
11	Literatura .....	69
12	Seznam příloh.....	72

# Seznam obrázků

Obrázek 1 Rozdělení pravděpodobností a jejich příslušné koeficienty [6].....	21
Obrázek 2 Grafické schéma metody Monte Carlo [3] [10].....	27
Obrázek 3 Rozdělení čísel při vygenerování $10^6$ náhodných čísel. ....	40
Obrázek 4 Rozdělení čísel při vygenerování $10^3$ náhodných čísel. ....	41
Obrázek 5 Schéma navrženého matematického modelu se zdroji nejistot. ....	56
Obrázek 6 Výsledný histogram proudu $I$ vypočtený programem Matlab 2010b. ....	59
Obrázek 7 Generátor náhodných čísel s normálním rozdělením v LabVIEW. ....	60
Obrázek 8 Generátor náhodných čísel s rovnoměrným rozdělením v LabVIEW. ....	60
Obrázek 9 Ukázka čelního panelu (uživatelská část).....	61
Obrázek 10 Ukázka čelního panelu (výsledková část).....	61
Obrázek 11 Výsledný histogram proudu $I$ vypočtený programem LabVIEW. ....	62
Obrázek 12 Výsledný histogram proudu $I$ vypočtený programem GNU Octave. ....	63

# Seznam tabulek

Tabulka 2-1 Základní charakteristiky rovnoměrného rozdělení pravděpodobnosti.....	15
Tabulka 2-2 Základní charakteristiky normálního rozdělení pravděpodobnosti.....	16
Tabulka 2-3 Základní charakteristiky binomického rozdělení pravděpodobnosti.....	16
Tabulka 2-4 Základní charakteristiky trojúhelníkového rozdělení pravděpodobnosti.....	17
Tabulka 2-5 Hodnoty korekčních koeficientů pro různé počty opakovaných měření [7].....	19
Tabulka 2-6 Možný způsob uspořádání bilanční tabulky.....	24
Tabulka 6-1 Výsledky testů MATLAB MT 3.4 pro $10^3$ generovaných hodnot.....	41
Tabulka 6-2 Výsledky testů MATLAB MT 3.4 pro $10^5$ generovaných hodnot.....	42
Tabulka 6-3 Výsledky testů pro generátor AS 183 využívaný programem Excel pro $10^3$ čísel.	43
Tabulka 6-4 Výsledky testů pro generátor AS 183 využívaný programem Excel pro $10^5$ čísel.	43
Tabulka 6-5 Výsledky testů pro generátor v programem Maple 17 pro $10^3$ vzorků.....	44
Tabulka 6-6 Výsledky testů pro generátor v programem Maple 17 pro $10^5$ vzorků.....	44
Tabulka 6-7 Výsledky testů pro generátor čísel v programu LabVIEW pro $10^3$ vzorků.....	46
Tabulka 6-8 Výsledky testů pro generátor čísel v programu LabVIEW pro $10^5$ vzorků.....	46
Tabulka 6-9 Výsledky testů gen. čísel, využívaného programem Octave pro $10^3$ hodnot.....	47
Tabulka 6-10 Výsledky testů gen. čísel, využívaného programem Octave pro $10^5$ hodnot.....	47
Tabulka 7-1 Přesnost měřicího rezistoru.....	49
Tabulka 7-2 Přesnost měření stejnosměrného napětí multimetrem Agilent 34401A [25].	49
Tabulka 7-3 Naměřené hodnoty napětí na měřicím rezistoru.....	50
Tabulka 7-4 Tabulka naměřených a vypočtených hodnot napětí.....	51
Tabulka 7-5 Souhrn všech nejistot vyskytujících se ve výpočtu proudu tekoucího rezistorem..	54
Tabulka 8-1 Soupis zdrojů nejistot působící na matematický model.....	56
Tabulka 8-2 Statistické parametry získané simulací metodou MC v Matlabu.....	59
Tabulka 8-3 Statistické parametry získané metodou MC v LabVIEW.....	62
Tabulka 8-4 Statistické parametry získané metodou MC v GNU Octave.....	63
Tabulka 9-1 Vypočtené hodnoty proudu protékajícího měřicím rezistorem.....	64
Tabulka 9-2 Vypočtené hodnoty směrodatných odchylek proudu.....	65
Tabulka 9-3 95% intervaly pokrytí vypočteny oběma metodami.....	65

# 1 ÚVOD

V dnešní době je při vyhodnocování různých měření teorie chyb nahrazována pojmem nejistota měření. V oborech jako jsou kalibrace a metrologie je již pojem nejistota měření nedílnou a nepostradatelnou součástí. Z důvodu rozšíření oboru stanovování nejistot měření se začali hledat i nové metody jak je co nejefektivněji a nejsnadněji stanovit. Jednou z takových metod je i simulační metoda Monte Carlo.

V teoretické části diplomové práce bude popsána metodika stanovení nejistot. Bude zde popsána jak klasická numerická metoda pro stanovení nejistot měření, tak i simulační metoda Monte Carlo. Metoda Monte Carlo je založena na výpočetním softwaru, který je schopen generovat velké množství náhodných čísel. Ke generování těchto čísel je třeba mít kvalitní generátory pseudonáhodných (náhodných) čísel obsažených ve výpočetním softwaru. Z tohoto důvodu zde budou popsány základní charakteristiky a vlastnosti jednotlivých generátorů. Dále budou v teoretické části popsány testovací baterie, se zaměřením na baterii NIST, které slouží k ověření těchto pseudonáhodných generátorů z hlediska náhodnosti generovaných čísel.

Praktická část diplomové práce se bude nejprve věnovat ověření a zhodnocení generátorů pseudonáhodných čísel obsažených ve vybraných softwarech, sloužících k realizaci stanovení nejistot měření pomocí metody Monte Carlo. Následně se bude zabývat stanovením nejistoty nepřímého měření proudu. Tato nejistota bude stanovena klasickou numerickou metodou a následně simulační metodou Monte Carlo. U metody Monte Carlo bude výpočet proveden ve třech předem vybraných dostupných softwarech. V poslední části práce bude provedeno zhodnocení dosažených výpočtů jednotlivými metodami a porovnání získaných výsledků.

## 2 NEJISTOTY MĚŘENÍ

V dřívějších dobách se přesnost měření definovala pomocí chyb měření. Tato metodika vycházela ze stanovení pravé hodnoty a chyby měření. Jelikož pravou hodnotu nelze z fyzikálních důvodů absolutně zjistit, zavedl se pojem konvenčně pravá hodnota. Tu lze zjistit buď pomocí přesnějších měření, teoretických výpočtů nebo aritmetickým průměrem z většího počtu měření. I zde ovšem nastává problém, jelikož výrobci přístrojů udávají různé parametry. Proto se od konce minulého století začala tato metodika nahrazovat metodikou stanovení nejistot měření [2].

V současnosti se velmi často v oblasti měření a jeho vyhodnocování setkáváme s pojmem nejistoty měření. Především v oblastech kalibrace a vrcholné metrologie jsou tyto pojmy nedílnou součástí výsledné zprávy, zatímco v oblasti průmyslových měření se pojem nejistota měření začíná pozvolna prosazovat. Za základní dokument v oblasti nejistot měření můžeme považovat směrnici *Guide to Expression of the Uncertainty of Measurement* [4], vydanou mezinárodními metrologickými institucemi v roce 1993, která byla následně přijata i jako norma. V České republice tento způsob vyhodnocování výsledků měření reprezentují především Technické předpisy metrologické TPM řady 055x.

Nejistota měření může být definována jako odhad, který je přidán k výsledku měření charakterizující určitý interval hodnot, o němž můžeme s určitou pravděpodobností tvrdit, že uvnitř leží skutečná hodnota. Výsledná nejistota měření je složena z mnoha dílčích složek. Některé složky lze charakterizovat směrodatnými odchylkami na základě statistického rozdělení výsledků měření. Jiné složky lze odhadnout pouze na základě zkušeností z předchozích měření nebo dokonalou znalostí dané měřicí metody [1] [2].

### 2.1 Základní pojmy ze statistiky

V této kapitole jsou popsány základní pojmy ze statistiky a teorie pravděpodobnosti. V první podkapitole jsou popsány základní pojmy z oboru statistiky. V druhé podkapitole jsou popsány nejčastěji se vyskytující rozdělení pravděpodobnosti v oblasti nejistot měření. Text vychází především ze zdrojů [2], [6] a [8].

#### 2.1.1 Základní termíny

##### **Náhodná veličina**

Tato veličina je libovolná veličina, kterou je možné opakovaně měřit v libovolném čase v různých místech. Její hodnoty je možné zpracovat metodami teoretické pravděpodobnosti a statistiky. Nejčastěji je označována  $X$ .

##### **Hustota pravděpodobnosti**

Hustota pravděpodobnosti  $f(x)$  spojitě náhodné veličiny  $X$  je nezáporná reálná funkce taková, že pro všechna reálná  $x$  se dá nalézt distribuční funkce  $F(x)$  náhodné veličiny

$X$ , pomocí vztahu (1). V některých případech se pro hustotu pravděpodobnosti využívá označení  $g(\xi_i)$ .

$$f(x) = \int_{-\infty}^x f(t)dt, \quad \text{pro } X \in (-\infty; +\infty). \quad (1)$$

Pro diskrétní veličinu je obdobou hustoty pravděpodobnosti pravděpodobnostní funkce značená  $P$ .

### Směrodatná odchylka

Směrodatná odchylka vyjadřuje rozptyl hodnot kolem střední hodnoty. Informuje nás tedy o tom, jak hustě jsou náhodné veličiny rozmístěny kolem střední hodnoty. Označuje se  $\sigma$ . Vztah pro výpočet je:

$$\sigma = \sqrt{\frac{\sum_{i=1}^N (x_i - E(X))^2}{N}}, \quad (2)$$

kde  $N$  je počet pozorování,  $x_i$  jsou hodnoty, kterých nabývá náhodná veličina  $X$  při jednotlivých pozorováních a  $E(X)$  je střední hodnota všech pozorování.

### Výběrová směrodatná odchylka

Výběrová směrodatná odchylka se užívá v případě výpočtu směrodatné odchylky na empiricky zjištěné řadě čísel. Označuje se  $s$ . Vztah pro výpočet výběrové směrodatné odchylky:

$$s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2}, \quad (3)$$

kde  $N$  je počet pozorování,  $x_i$  jsou jednotlivá reálná čísla a  $\bar{x}$  je průměrná hodnota z daného výběru čísel.

### Distribuční funkce

Distribuční funkce je funkce udávající pravděpodobnost, že hodnota náhodné veličiny  $X$  je menší než zadaná hodnota. Funkce je definována vztahem (4). V případě diskrétní distribuční funkce se užívá označení  $G(X)$ .

$$F(X) = P(X \leq x) \quad (4)$$

### Rozptyl

Rozptyl vyjadřuje variabilitu rozdělení souboru náhodných hodnot kolem její střední hodnoty. Nejčastěji je označován  $\sigma^2(X)$ ,  $D(X)$ . V případě diskrétní náhodné veličiny se rozptyl vypočte dle vztahu (5).

$$D(X) = \sum_{i=1}^n [x_i - E(X)]^2 \cdot p(x_i), \quad (5)$$

kde  $x_i$  jsou hodnoty, kterých nabývá náhodná veličina  $X$ ,  $p_i$  značí s jakou pravděpodobností jich nabývá a  $E(X)$  je střední hodnota náhodné veličiny  $X$ . V případě spojité náhodné veličiny je rozptyl dán vztahem (6).

$$D(X) = \int_{-\infty}^{+\infty} (x - E(X))^2 \cdot f(x) dx \quad (6)$$

### Střední hodnota

Střední hodnota je nejznámější míra ve statistice. Nejčastěji je označována jako *populační průměr*. Většinou je označována  $E(X)$ ,  $E$ ,  $\mu$  nebo  $\langle X \rangle$ . Existují dva typy střední hodnoty a to střední hodnota pro spojitou náhodnou veličinu (7) a střední hodnota pro diskrétní náhodnou veličinu (8).

$$E(X) = \int_{-\infty}^{\infty} x \cdot f(x) dx \quad (7)$$

$$E(X) = \sum_{(i)} x_i \cdot p(x_i) \quad (8)$$

kde  $x_i$  jsou hodnoty, kterých nabývá náhodná veličina  $X$  a  $p_i$  značí s jakou pravděpodobností jich nabývá.

## 2.1.2 Rozdělení pravděpodobnosti

Rozdělení pravděpodobnosti náhodné veličiny je pravidlo vyjadřující pravděpodobnost, která je přiřazena každému jevu popisovanému náhodnou veličinou. Existuje celá řada typů rozdělení. Základní dělení těchto rozdělení je na diskrétní rozdělení pravděpodobnosti a spojité rozdělení pravděpodobnosti. Mezi diskrétní rozdělení pravděpodobnosti patří například alternativní rozdělení, rovnoměrné rozdělení, binomické rozdělení, Poissonovo rozdělení, geometrické rozdělení a negativně binomické rozdělení. Spojitá roz-

dělení jsou například rovnoměrné rozdělení, exponenciální rozdělení, trojúhelníkové rozdělení, normální rozdělení. Dále je u spojitých rozdělení možno uvést i chí-kvadrát rozdělení, studentovo rozdělení, Fischerovo-Snedecorovo rozdělení, jelikož tyto rozdělení vycházejí z normálního rozdělení. Nejčastěji se vyskytující typy rozdělení pravděpodobnosti v oblasti nejistot měření jsou detailněji popsány v této podkapitole. Více informací ohledně jednotlivých typů rozdělení je možné nalézt ve zdrojích [2], [8].

### 2.1.2.1 Rovnoměrné rozdělení

Rovnoměrné rozdělení pravděpodobnosti je takové rozdělení, při kterém je všem hodnotám náhodné veličiny přiřazena stejná hodnota pravděpodobnosti. Toto rozdělení má svoji diskrétní i spojitou podobu.

U spojitě podoby rovnoměrného rozdělení nabývá náhodná veličina  $X$  s rovnoměrným rozdělením pro všechna  $x \in \langle \alpha, \beta \rangle$  konstantní hustotu pravděpodobnosti. Pro hodnoty mimo tento interval nabývá hodnoty nulové. Základní charakteristiky spojitě podoby rovnoměrného rozdělení jsou uvedeny v tabulce 2-1.

U diskrétní podoby rovnoměrného rozdělení může náhodná veličina nabývat  $n$  hodnot se stejnou pravděpodobností, která je rovna  $1/n$ , přičemž vzdálenosti mezi jednotlivými hodnotami náhodné veličiny jsou stejné.

**Tabulka 2-1 Základní charakteristiky rovnoměrného rozdělení pravděpodobnosti.**

Hustota pravděpodobnosti:	
$f(x) = \begin{cases} \frac{1}{\beta - \alpha}, & \text{pro } x \in \langle \alpha, \beta \rangle, \\ 0, & \text{pro } x \in (-\infty, \alpha) \cup \langle \beta, +\infty \rangle. \end{cases}$	
Distribuční funkce:	
$F(x) = \begin{cases} 0, & \text{pro } x \in (-\infty; \alpha), \\ \int_{\alpha}^x \frac{1}{\beta - \alpha} dx = \frac{x - \alpha}{\beta - \alpha}, & \text{pro } x \in \langle \alpha; \beta \rangle, \\ 1, & \text{pro } x \in (\beta; +\infty). \end{cases}$	
Číselné charakteristiky	
Střední hodnota:	Rozptyl:
$E(X) = \mu = \int_{-\infty}^{+\infty} xf(x)dx = \int_{\alpha}^{\beta} x \frac{1}{\beta - \alpha} dx = \frac{\beta + \alpha}{2}$	$D(X) = \sigma^2 = \int_{-\infty}^{+\infty} (x - \mu)^2 f(x)dx = \frac{\beta - \alpha}{12}$

V problematice nejistot měření se rovnoměrné rozdělení nejčastěji využívá u měřících přístrojů.

### 2.1.2.2 Normální rozdělení

Normální (Gaussovo) rozdělení pravděpodobnosti je jedno z nejdůležitějších a nejčastěji využívaných rozdělení. Nejčastěji využívané je především ze dvou důvodů. Prvním důvodem je schopnost tímto rozdělením nahradit řadu jiných rozdělení pravděpodobnosti. Druhý důvod je takový, že celá řada rozdělení pravděpodobnosti se blíží právě tomuto rozdělení. Zápis se provádí ve tvaru  $X \rightarrow N(\mu, \sigma^2)$ , kde  $X$  je náhodná veličina,  $\mu$  je střední hodnota a  $\sigma^2$  je rozptyl. Základní charakteristiky normálního rozdělení pravděpodobnosti jsou uvedeny v tabulce 2-2.

**Tabulka 2-2 Základní charakteristiky normálního rozdělení pravděpodobnosti.**

Hustota pravděpodobnosti:	
$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-E(X))^2}{2D(x)}} \quad \text{pro } E(X) \in (-\infty; +\infty), \quad D(X) > 0, \quad x \in (-\infty; +\infty)$	
Distribuční funkce:	
$F(x) = \int_{-\infty}^x \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(t-\mu)^2}{2\sigma^2}} dt$	
Číselné charakteristiky	
Střední hodnota: $E(X) = \mu = \int_{-\infty}^{+\infty} x \cdot f(x) dx$	Rozptyl: $D(x) = \sigma^2 = \int_{-\infty}^{+\infty} (x - \mu)^2 \cdot f(x) dx$

### 2.1.2.3 Binomické rozdělení

Binomické rozdělení pravděpodobnosti udává četnost výskytu náhodného jevu v  $n$  nezávislých pokusech, přičemž náhodný jev má stále stejnou pravděpodobnost. K zápisu tohoto rozdělení slouží matematický výraz  $X \rightarrow Bi(n, p)$ , kde  $n$  je počet pokusů a  $p$  udává pravděpodobnost daného náhodného jevu. Základní charakteristiky tohoto rozdělení pravděpodobnosti jsou uvedeny v tabulce 2-3.

**Tabulka 2-3 Základní charakteristiky binomického rozdělení pravděpodobnosti.**

Hustota pravděpodobnosti:	
$p(x) = \begin{cases} \binom{n}{x} p^x (1-p)^{n-x}, & x \in \{0, 1, \dots, n\} \\ 0, & x \notin \{0, 1, \dots, n\} \end{cases}$	
Číselné charakteristiky:	
Střední hodnota: $E(x) = \mu = np$	Rozptyl: $D(x) = \sigma^2 = np(p-1)$

### 2.1.2.4 Trojúhelníkové (Simpsonovo) rozdělení pravděpodobnosti

Trojúhelníkové rozdělení pravděpodobnosti je spojité rozdělení, jehož hustota pravděpodobnosti vyjádřena v grafu je rovnoramenný trojúhelník. Jeho symbolický matematický zápis je  $X \rightarrow Tri(a, b, c)$ , kde parametry  $a, b$  vymezují podstavu a  $c$  polohu vrcholu hustoty pravděpodobnosti. Základní charakteristiky jsou uvedeny v tabulce 2-4. Toto rozdělení má u nejistot využití při stanovování nejistoty u hodnot získaných z posuvných měřítok.

Tabulka 2-4 Základní charakteristiky trojúhelníkového rozdělení pravděpodobnosti.

Hustota pravděpodobnosti:	
$f(x) = \begin{cases} 0, & \text{pro } x \leq a, \\ \frac{2(x-a)}{(b-a)(c-a)}, & \text{pro } a < x \leq c, \\ \frac{2(b-x)}{(b-a)(b-c)}, & \text{pro } c < x \leq b, \\ 0, & \text{pro } x > b. \end{cases}$	
Distribuční funkce:	
$F(x) = \begin{cases} 0, & \text{pro } x \leq a, \\ \frac{(x-a)^2}{(b-a)(c-a)}, & \text{pro } a < x \leq c, \\ 1 - \frac{(x-b)^2}{(b-a)(b-c)}, & \text{pro } c < x \leq b, \\ 1, & \text{pro } x > b. \end{cases}$	
Číselné charakteristiky:	
Střední hodnota: $E(X) = \mu = \frac{a+b+c}{3}$	Rozptyl: $D(X) = \sigma^2 = \frac{a^2 + b^2 + c^2 - ab - ac - bc}{18}$

## 2.2 Typy nejistot

Směrnice *Guide to Expression of the Uncertainty of Measurement* [4] definuje několik základních typů nejistot. Tyto nejistoty jsou:

- **standardní nejistoty typu A** - Tyto nejistoty jsou stanoveny z výsledků opakovaných měření statistickou analýzou série naměřených hodnot. Příčiny těchto nejistot se považují za neznámé. Jejich hodnota klesá s počtem měření.

- **standardní nejistoty typu B** - Tyto nejistoty jsou stanovovány jinou metodou než statistickým zpracováním výsledků z opakovaných měření. Jejich hodnoty nejsou závislé na počtu opakovaných měření, nýbrž jsou vyhodnoceny pro jednotlivé zdroje nejistoty stanovené pro konkrétní měření. Společné působení těchto nejistot od různých zdrojů vyjadřuje výsledná standardní nejistota typu B.
- **kombinovaná nejistota** - Nejistota se získá sloučením standardní nejistoty typu A s výslednou standardní nejistotou typu B.
- **rozšířená nejistota** – Tato nejistota svojí hodnotou vymezuje šířku pásu v němž leží správná hodnota s určitou pravděpodobností.

Konkrétní vztahy pro výpočty těchto typů nejistot jsou uvedeny v podkapitolách 2.4 a 2.5.

## 2.3 Zdroje nejistot měření

Zdroje nejistot jsou všechny vlivy, které ovlivňují výsledek měření a vzdalují naměřenou hodnotu od hodnoty skutečné. Velkou roli sehrává i skutečnost zda se jedná o přímou nebo nepřímou měřicí metodu. Dalšími důležitými faktory, které ovlivňují nejistoty jsou výběr měřicího přístroje, výběr vzorkovačů, výběr filtrů a způsoby úprav měřicího signálu. Velký podíl u nejistot mají také rušivé vlivy prostředí. Nejčastěji vyskytující se vlivy jsou [1]:

- nedokonalá či neplná definice měřené veličiny,
- nevhodný výběr přístroje,
- nevhodný postup měření,
- zaokrouhlování, linearizace, aproximace, interpolace a extrapolace,
- neznámé nebo nekompenzované vlivy prostředí,
- nedodržování shodných podmínek při opakovaném měření,
- nepřesnosti etalonů nebo referenčních materiálů.

## 2.4 Nejistoty přímých měření

Tato podkapitola vychází zejména ze zdrojů [2], [4], [6], [7]. Přímá měření jsou taková měření, při kterých je výsledná hodnota měření přímo indikována měřicím přístrojem.

### 2.4.1 Standardní nejistota typu A

Výpočet standardní nejistoty typu A je založen na statistické analýze naměřených hodnot, které jsou získány za neměnných podmínek. Počet hodnot  $n$  získaných touto metodou by mělo být co nejvíce, pokud možno alespoň deset. Mírou nejistoty typu A je výběrová směrodatná odchylka výběrového průměru  $u_A$ .

$$u_A(y) = s_{\bar{y}} = \sqrt{\frac{\sum_{i=1}^n (y_i - \bar{y})^2}{n(n-1)}}, \quad (9)$$

$$\bar{y} = \frac{\sum_{i=1}^n y_i}{n}, \quad (10)$$

kde:

$u_A(y)$  je standardní nejistota typu A,

$s_{\bar{y}}$  je výběrová směrodatná odchylka výběrového průměru,

$n$  je počet opakování daného měření,

$\bar{y}$  je aritmetický průměr naměřených hodnot,

$y_i$  reprezentuje jednotlivé naměřené hodnoty.

V některých případech, kdy není možné dosáhnout počtu opakování měření alespoň deset a získat tedy kvalifikovaný odhad, určí se korigovaná nejistota  $u_{Ak}(y)$  ze vztahu:

$$u_{Ak}(y) = k \cdot s(\bar{y}), \quad (11)$$

kde  $k$  je korekční koeficient, který kompenzuje věrohodnost nejistoty. Korekční koeficient je závislý na počtu opakování měření. Tato závislost je uvedena v tabulce 2-5.

Tabulka 2-5 Hodnoty korekčních koeficientů pro různé počty opakování měření [7].

$n$	9	8	7	6	5	4	3	2
$k$	1,2	1,2	1,3	1,3	1,4	1,7	2,3	7,0

## 2.4.2 Standardní nejistota typu B

Nejistoty stanovené metodou typu B jsou na rozdíl od nejistot stanovené metodou typu A vázány na známé, identifikovatelné a kvantifikované zdroje. Výpočet touto metodou je založen na kvalifikovaném úsudku ze všech dostupných informací o měřené veličině a jejich možných změnách. Jako základní zdroje informací ke stanovení nejistoty typu B mohou složit:

- předchozí měření a jejich známé výsledky,
- zkušenosti a všeobecné znalosti o chování materiálu a měřicí techniky,
- nejistoty referenčních údajů převzatých z různých materiálů,
- údaje z certifikátů a kalibračních listů.

Při výpočtu se vychází z dílčích nejistot jednotlivých zdrojů, kde hodnota  $z_{j\max}$  je maximální odchylka zdroje nejistot.

$$u_B(z_j) = \frac{z_{j\max}}{\chi}, \quad (12)$$

kde koeficient  $\chi$  je hodnota příslušná k rozdělení pravděpodobnosti jednotlivých zdrojů nejistot. Přičemž základní typy rozdělení pravděpodobnosti jsou popsány v podkapitole 2.1.2. Na obrázku 1 je zobrazen přehled nejčastěji užívaných rozdělení pravděpodobnosti zdrojů nejistot a k nim příslušná hodnota koeficientu  $\chi$ . Výslednou nejistotu typu B je poté možno vypočíst dle vztahu (13).

$$u_B(y) = \sqrt{\sum_{j=1}^m u_B(z_j)^2}. \quad (13)$$

V případě, že nastane mezi některými zdroji nejistot vzájemná korelace, je třeba určit výběrový korelační koeficient  $r_B(y_z, y_i)$ , který nabývá hodnot v intervalu  $\langle -1; 1 \rangle$  podle vzájemné závislosti odhadů  $y_z$  a  $y_i$ . Pokud nastane případ, že odhady  $y_z$  a  $y_i$  jsou nezávislé, je korelační koeficient  $r_B = 0$ . Dále je třeba stanovit citlivostní koeficienty  $A_z$  a  $A_i$  dle vztahu (20). Poté můžeme výslednou nejistotu typu B vypočíst dle vztahu (14).

$$u_B(y) = \sqrt{\sum_{z=1}^n A_z^2 u_B(y_z)^2 + 2 \sum_{z=1}^{n-1} \sum_{i>z}^n A_z A_i u_B(y_z) r_B(y_z, y_i)}. \quad (14)$$

Rozdělení	Z <sub>max</sub>	χ	Rozdělení	Z <sub>max</sub>	χ
rovnoměrné 	a	√3	lichoběžníkové 	a	2,32
trojúhelníkové 	a	√6	bimodální 	a	-2,19
normální 	a	3	trojúhelníkové 	a	√2
	b	2	bimodální 	a	1
			Diracovo 		

Obrázek 1 Rozdělení pravděpodobností a jejich příslušné koeficienty [6].

### 2.4.3 Kombinovaná standardní nejistota měření

Kombinovaná standardní nejistota je dána geometrickou sumací nejistot typu A a B. Tato nejistota udává interval, ve kterém se s 68% pravděpodobností vyskytuje skutečná hodnota měření.

$$u_c(y) = \sqrt{u_A^2 + u_B^2}. \quad (15)$$

V případě, že je požadována větší pravděpodobnost výskytu skutečné hodnoty měřené veličiny, je třeba zavést rozšířenou standardní nejistotu.

## 2.4.4 Rozšířená standardní nejistota

Rozšířená standardní nejistota se zavádí v případě, že je třeba zajistit, co největší spolehlivost výsledku blížíci se 100%. Tuto nejistotu získáme tak, že zvětšíme interval možných hodnot. Nejčastěji se koeficient rozšíření  $k_r$  volí roven dvěma pro rozšíření na 95%, pokud bude požadována větší spolehlivost výsledku zvolí se koeficient rozšíření  $k_r$  roven třem, který rozšíří interval na 99,7%.

$$U(y) = k_r \cdot u_c(y). \quad (16)$$

## 2.5 Nejistoty nepřímých měření

Nepřímé měření jsou měření, ve kterých je dána výsledná hodnota funkčním vztahem vstupních hodnot odečítaných přímo z měřicích přístrojů.

Výstupní veličina  $Y$  u nepřímých měření se vypočte z dílčích přímých měření. Platí zde vztah:

$$Y = f(X_1, X_2, \dots, X_n), \quad (17)$$

kde  $f$  je známa funkce. Odhad  $y$  výstupní veličiny  $Y$  je dán vztahem:

$$y = f(x_1, x_2, \dots, x_n). \quad (18)$$

Kde  $x_1, x_2, \dots, x_n$  jsou odhady vstupních veličin  $X_1, X_2, \dots, X_n$ . V případě, že vstupní veličiny jsou nekorelované (tzn. není mezi vstupními veličinami žádná souvislost), nejistota odhadu se určí dle vztahu (19).

$$u_y^2 = \sum_{i=1}^m A_i^2 u^2(x_i), \quad (19)$$

kde  $A_i$  je koeficient citlivosti, který se vypočte dle vztahu (20).

$$A_i = \frac{\partial f(X_1, X_2, \dots, X_n)}{\partial X_i} \Big|_{X_1 = x_1, \dots, X_n = x_n} \quad (20)$$

V případě, že odhady vstupních veličin  $x_1, x_2, \dots, x_n$  jsou korelované, je třeba také zvažovat jednotlivé kovariance mezi jednotlivými odhady. Výpočet nejistoty je poté dán vztahem (21).

$$u_y^2 = \sum_{i=1}^m A_i^2 u^2(x_i) + 2 \sum_{i=2}^m \sum_{j<i}^{m-1} A_i A_j u(x_i, x_j), \quad (21)$$

kde  $u(x_i, x_j)$  je kovariance mezi dvěma navzájem korelovanými odhady  $x_i$  a  $x_j$ .

Kovariance mezi odhady vlivů jednotlivých zdrojů určují, jak jsou tyto odhady vzájemně ovlivněny společnými zdroji nejistot. Navzájem závislé zdroje nejistot přispívají k výsledné nejistotě, buď více nebo méně dle toho, jak se nejistoty slučují. Kovariance mohou výslednou nejistotu zvětšit i zmenšit. Závisí to především na tvaru funkce, kterou jsou odhady vázány na výstupní veličinu a na tom zda zdroje působí souhlasně nebo protichůdně na dva uvažované odhady. Podrobnější popis ohledně kovariancí je možné nalézt v dokumentu [5].

## 2.6 Zásady pro vyjadřování nejistot

Výpočet nejistot je nedílnou součástí zpracování naměřených dat. Z tohoto důvodu musejí výsledné nejistoty být součástí zprávy o výsledku měření. Důležité je, aby u každého údaje nejistoty bylo přesně uvedeno o jakou nejistotu se jedná. K uvádění výsledků měření se doporučuje využít zápis výsledku buď s použitím kombinované nejistoty  $u_c$  nebo zápis výsledku s užitím rozšířené nejistoty  $U$ . Pokud je nutné zpřehlednit dosažené výsledky je možné využít i tzv. bilanční tabulku [4],[5].

### 2.6.1 Vyjádření výsledku s použitím kombinované nejistoty

Při zápisu výsledku s použitím standardní kombinované nejistoty  $u_c$  je třeba uvést podrobnou definici měřené veličiny. Dále je nutné uvést odhad  $y$  měřené veličiny  $Y$  společně s kombinovanou standardní nejistotou  $u_c(y)$  a její jednotkou. V některých případech je vhodné uvést jak absolutní tak i relativní standardní kombinovanou nejistotu. V případě potřeby je možné uvést i bilanční tabulku.

Takto uváděná standardní kombinovaná nejistota se zásadně zaokrouhlí na maximálně dvě platná čísla směrem nahoru. Větší počet míst je možné ponechat pouze v případě, že nejistota podléhá dalšímu zpracování.

### 2.6.2 Vyjádření výsledku s použitím rozšířené nejistoty

Při zápisu výsledku s rozšířenou nejistotou  $U$  je třeba uvést definici měřené veličiny  $Y$ . Dále uvést výsledek měření v podobě  $Y = (y \pm U)$ , včetně jednotek dané měřené veličiny. Nutné je uvést i koeficient rozšíření  $k_r$ , který byl použit k výpočtu dané rozšířené nejistoty. Opět je možné uvést jak absolutní tak relativní rozšířenou nejistotu jelikož jsou ekvivalentní.

Rozšířená nejistota se opět zaokrouhlí na maximálně dvě platná čísla přednostně směrem nahoru.

Příkladem správného zápisu může být například odpor  $R = 17,8 \Omega$  a jeho rozšířená nejistota  $U(R) = 0,3 \Omega$ . Správný zápis poté je  $R = (17,8 \pm 0,3) \Omega$ .

Příkladem nesprávného zápisu je například  $f = (12,53 \pm 0,854) \text{ Hz}$ . Pro správný zápis je třeba zaokrouhlit obě čísla na desetiny s tím, že nejistota musí být zaokrouhlena směrem nahoru. Správný zápis by tedy byl  $f = (12,5 \pm 0,9) \text{ Hz}$ .

### 2.6.3 Bilanční tabulka

Bilanční tabulka se užívá pokud chceme zpřehlednit výpočet jednotlivých nejistot, nebo zjednodušit porovnání nejistot při analýze dosažených výsledků. Tabulka by měla obsahovat přinejmenším měřenou veličinu, odhad měřené veličiny, standardní nejistotu měřené veličiny, citlivostní koeficient a příspěvek ke standardní nejistotě. Dále může obsahovat rozdělení pravděpodobnosti nebo koeficient krytí.

Tabulka 2-6 Možný způsob uspořádání bilanční tabulky.

Veličina	Odhad	Standardní nejistota	Rozdělení pravděpodobnosti	Citlivostní koeficient	Příspěvek k standardní nejistotě
$X_i$	$x_i$	$u(x_i)$		$A_i$	$u_i(y)$
$X_1$	$x_1$	$u(x_1)$	rovnoměrné	$A_1$	$u_1(y)$
$X_2$	$x_2$	$u(x_2)$	normální	$A_2$	$u_2(y)$
.	.	.	.	.	.
.	.	.	.	.	.
.	.	.	.	.	.
$X_a$	$x_a$	$u(x_n)$	normální	$A_n$	$u_n(y)$
$Y$	$y$				$u(y)$

## 3 METODA MONTE CARLO

Metoda Monte Carlo je definována jako třída algoritmů pro simulaci systémů. Jedná se o stochastické metody používající k výpočtu pseudonáhodná nebo náhodná čísla. Tato metoda byla formulována ve 40. letech 20. století. Objeviteli byli vědci Stanislaw Marcin Ulam a John von Neumann. Tito dva vědci zkoumali v Národní laboratoři v Los Alamos chování neutronů a využili přitom poprvé tuto metodu [9].

### 3.1 Oblasti využití metody Monte Carlo

Metodu Monte Carlo lze využít na celou škálu aplikací. Z obecného pohledu lze říci, že metodu lze využít v případech, kdy je možné nalézt řešení pomocí mnohokrát opakovaných náhodných pokusů. Podmínkou je znalost pravděpodobnostního rozdělení sledovaných veličin, jelikož se jedná o stochastickou metodu [3], [9]. Nejčastěji je metoda Monte Carlo využívána v oblastech [3], [9], [10]:

- Matematika – metoda je využívána k řešení jednoduchých i vícerozměrných integrálů, systémů lineárních rovnic, hledání kořenů rovnic i k řešení parciálních diferenciálních rovnic.
- Fyzika – zde se metody využívá k výpočtům v kvantové chromodynamice, aerodynamice, fyzice polymerů, statické fyzice. Další využití nachází metoda v oblasti předpovědi počasí.
- Počítačová grafika – metoda nachází uplatnění především u počítačových her, grafiky, různých animací a filmových efektů.
- Hazardní hry – zde metoda umožňuje mechanizovat hazardní hry, jelikož základem metody jsou generátory pseudonáhodných nebo náhodných čísel, které umožňují nastavit všem alternativám stejnou pravděpodobnost.
- Využití v elektrotechnice – **výpočet nejistot.**

### 3.2 Výpočet nejistot pomocí metody Monte Carlo

Metoda Monte Carlo je realizována podle algoritmu, který je popsán v následujících bodech [9], [10]:

- Vytvoření matematického modelu  $Y = f(X)$ , kde  $Y$  reprezentuje skalární výstupní veličinu a  $X$  reprezentuje  $n$  vstupních veličin. Každá veličina  $X_i$  je náhodná veličina

s hustotou pravděpodobnosti  $g(\xi_i)$ , kde  $\xi_i$  reprezentuje hodnotu dané veličiny.  $Y$  je náhodná veličina s možnou hodnotou  $\eta$  a hustotou pravděpodobnosti  $g(\eta)$ .

- Stanovení počtu opakování  $M$  a pravděpodobnost pokrytí  $p$  pro metodu Monte Carlo.
- Vygenerování  $M$  náhodných vektorů  $x_s$ ,  $s = 1, \dots, M$  dle hustoty pravděpodobnosti.
- Dosazení vygenerovaných čísel do modelu  $y_s = f(x_s)$ .
- Seřadit hodnoty  $y_s$  do neklesajícího pořadí. Z řady těchto hodnot se poté určí diskrétní distribuční funkce  $D$ .
- Výpočet aritmetického průměru  $\bar{y}$  a směrodatné odchylky  $u(\bar{y})$  dle vztahů (22) a (23).

$$\bar{y} = \frac{1}{M} \sum_{r=1}^M y_r \quad (22)$$

$$u(\bar{y}) = \sqrt{\frac{1}{M-1} \sum_{r=1}^M (y_s - \bar{y})^2} \quad (23)$$

- Určení intervalu pokrytí pro  $Y$  stanoveného z diskrétní podoby  $G$ . Interval stanovíme tak, že nejprve vypočteme  $q = pM$ , poté  $[y_{\text{spodni}} \ y_{\text{horni}}]$  je  $100.p \%$  interval pokrytí pro  $Y$  kde  $y_{\text{spodni}} = y_s$  a  $y_{\text{horni}} = y_{(s+q)}$ . Pravděpodobnostně symetrický  $100.p \%$  interval vypočteme jako  $s = (M - q)/2$ .

### 3.3 Výhody a nevýhody metody Monte Carlo

Výhodou metody MC je především schopnost výpočtu nejistot s komplikovaným rozdělením vstupních veličin. Další výhodou je možnost výpočtu s komplexními čísly. Také v případě, že model obsahuje nelinearity je vhodné využít metody MC.

Nevýhodou této metody je náročnost na výpočetní kapacitu PC a simulační program. Nevýhodou je také, že metodu není možné řešit ručně na papír [12].

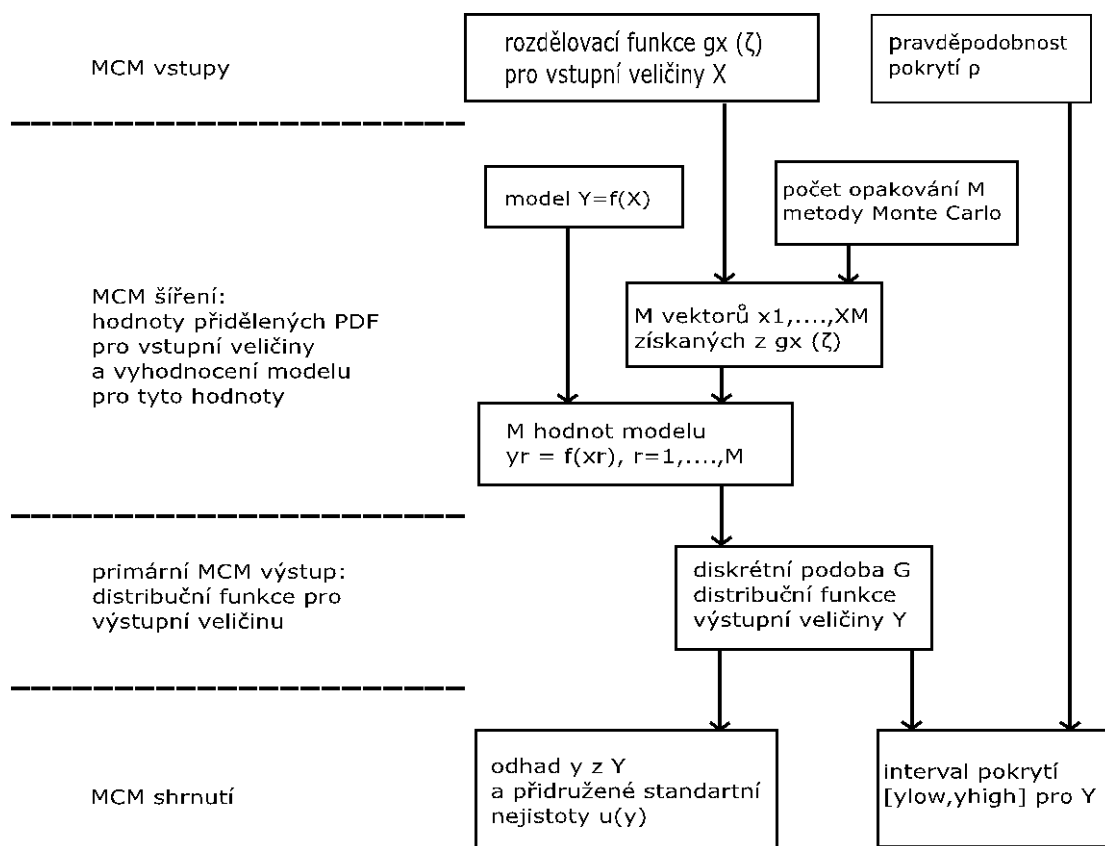
### 3.4 Software vhodný pro metodu Monte Carlo

Programů, v kterých lze realizovat simulace a výpočty pomocí metody Monte Carlo je celá řada. Základním předpokladem pro tyto programy musí být schopnost zpracovávat velké množství dat. Dalším velice důležitým předpokladem je, aby program obsahoval zabudovaný kvalitní generátor pseudonáhodných čísel, jelikož takový generátor je pro úspěšné provedení simulace zásadní. Často užívanými programy jsou MATLAB [29] od společnosti MathWorks a LabVIEW od společnosti National Instruments [23]. Dalším

velmi často využívaným program je GNU Octave [24]. U tohoto programu je výhodou částečná kompatibilita s programem MATLAB a také jeho volně šiřitelná licence.

Další skupinou jsou programy, které jsou zaměřeny přímo na simulace pomocí metody Monte Carlo, jako například OpenBUGS stažitelný z [27]. I u tohoto programu je nespornou výhodou volně dostupná licence. Do této skupiny by se daly zařadit i programy, které jsou určeny přímo pro výpočet nejistot. Jedním z takových programů je GUM Workbench Pro [28], u něhož je možnost výpočet nejistot jak klasickou metodou tak ve verzi 2.4 i metodou MC.

Poslední skupinou jsou tabulkové editory u kancelářských balíků. Příkladem může být Microsoft Excel 2003 a nižší verze. Tyto softwary se však pro výpočet nedoporučují, jelikož generátor čísel v nich obsažený není dostatečně kvalitní. Dalším důvodem jsou problémy se zpracováním velkého obsahu dat. Od verze Microsoft Excel 2007 je software vybaven dostatečně kvalitním generátorem náhodných čísel, ale problémy se zpracováním velkého množství dat nadále přetrvávají. Text v této kapitole vychází především ze zdrojů [12], [9].



Obrázek 2 Grafické schéma metody Monte Carlo [3] [10].

## 4 GENERÁTORY NÁHODNÝCH ČÍSEL

Důležitou podmínkou pro správnou funkčnost metody Monte Carlo jsou náhodná čísla. Tyto náhodná čísla se získávají pomocí generátorů čísel. Nejčastěji se využívají generátory náhodných a pseudonáhodných čísel [12], [13].

### 4.1 Generátory náhodných čísel

Generátor náhodných čísel je zařízení, které na základně výpočtu nebo fyzikálním procesu generuje řady náhodných čísel, které postrádají jakýkoliv vzor. Nejčastěji užívané metody pro generování těchto čísel jsou [13]:

- Fyzikální a Hardwarové metody – Metody mohou být realizovány velkým množstvím principů. Jedna z metod je založena na nepředvídatelnosti chování atomárních a subatomárních jevů. Tyto jevy je možné sledovat za pomoci kvantové mechaniky. Zjišťuje se časový interval, ve kterém se radioaktivní zdroj rozpadá. Tento časový interval je zcela náhodný. Fyzikální a Hardwarové metody se užívají především v hazardních hrách. Pro užití v kryptografii a statistice jsou pomalé, tím pádem nepoužitelné.
- Metody podle rozdělení pravděpodobnosti – Tyto metody provádějí transformaci náhodného čísla vybraného z rovnoměrného rozložení. Například metoda „přijetí-zamítnutí“ pracuje na principu výběru hodnot  $x$  a  $y$  a testuje, zda funkce v bodě  $x$  je větší než v bodě  $y$ . V případě, že funkce je větší, bude hodnota přijata, v opačném případě algoritmus vybere nové hodnoty  $x$  a  $y$  a opakuje proces.

### 4.2 Generátory pseudonáhodných čísel

Generátory pseudonáhodných čísel jsou deterministické programy, generující posloupnost čísel. Tato posloupnost by měla být pokud možno statistickými testy nerozlišitelná od náhodné. Pseudonáhodné generátory jsou klíčovým prostředkem kryptografie a různých simulačních metod. Jednou z těchto simulačních metod je i metoda Monte Carlo [13].

Generátory pseudonáhodných čísel lze rozdělit do dvou základních skupin. První skupinou jsou generátory pseudonáhodných čísel s uniformním rozdělením. Výstupem těchto generátorů je soubor vygenerovaných čísel, které mají rovnoměrné rozdělení pravděpodobnosti viz. podkapitola 2.1.2. Druhou skupinou jsou generátory pseudonáhodných čísel s obecným rozdělením pravděpodobnosti. Tyto rozdělení pravděpodobnosti jsou také popsány v podkapitole 2.1.2.

## 4.2.1 Generátory pseudonáhodných čísel s uniformním rozdělením pravděpodobnosti

Generování náhodných čísel s daným rozdělením pravděpodobnosti je, jak již bylo zmíněno, zcela zásadní pro různé simulace a statistické výpočty. V praxi se z důvodu rychlosti a velkého množství dat využívají především deterministické algoritmy. V této podkapitole budou popsány deterministické algoritmy pro generování vzorků s rovnoměrným rozdělením pravděpodobnosti. Algoritmus tohoto generování vychází z počáteční hodnoty  $x_0 \in \Omega$  z prostoru všech možných hodnot  $\Omega$ , na kterou se poté aplikuje zobrazení  $D: \Omega \rightarrow \Omega$ , čímž vznikne nová hodnota  $x_1 = D(x_0)$ . Další hodnoty jsou poté generovány deterministickým způsobem dle vztahu (24).

$$x_n = D(x_{n-1}) = D^n(x_0). \quad (24)$$

Ze vztahu (24) plyne označení vzorků jako pseudonáhodné. V případě vhodně zvoleného zobrazení  $D$  a počáteční hodnoty  $x_0$  je možné získat posloupnost, která je pomocí běžných testů téměř nerozlišitelná od posloupnosti náhodných vzorků. Speciální jev nastane v případě  $\Omega = (0,1)$ . V tomto případě splňuje vygenerovaná posloupnost několik základních podmínek, jak projít testy na uniformitu rozdělení, mít velkou periodu opakování a nesouvztažnost.

Základní dělení těchto generátorů je na lineární generátory pseudonáhodných čísel a nelineární generátory pseudonáhodných čísel. Z důvodu existence velkého množství jednotlivých typů generátorů, jsou v následujících podkapitolách uvedeny a stručně popsány pouze základní typy lineárních i nelineárních generátorů pseudonáhodných čísel s uniformním rozdělením pravděpodobnosti. Důvodem je vytvoření obrazu o principu generování těchto náhodných čísel. Tato kapitola vychází především ze zdrojů [13], [14], [15].

### 4.2.1.1 Kongruentní generátory

Kongruentní generátory jsou nejpoužívanějším a nejrozšířenějším pseudonáhodné generátory. Vytvářejí pseudonáhodnou posloupnost čísel z intervalu  $\langle 0; m \rangle$  pomocí vztahu (25).

$$x_n = (a_1 x_{n-1} + \dots + a_k x_{n-k} + b) \bmod m; n \geq 0, \quad (25)$$

kde  $x_0, \dots, x_k$  jsou počáteční hodnoty,  $a_1, \dots, a_k$  jsou multiplikatívni členy,  $b$  je aditivní člen a  $m$  je celočíselný modul. Generovaná čísla vznikají jako zbytek po dělení tak, že číslo  $x_{n+1}$  vznikne vydělením čísla  $(a_1 x_n + \dots + a_k x_{n-k} + b)$  modulem  $m$ . Vlastnosti i perioda tohoto generátoru poté závisí na zvolených parametrech počátečních hodnot,

multiplikačních členů, aditivního členu a modulu. Detailnější popis je možné nalézt ve zdroji [14]. Vybrané typy lineárních a nelineárních generátorů založených na tomto principu jsou:

- **Inverzní kongruentní generátor**

Inverzní kongruentní generátor je typ nelineárního generátoru vytvářející posloupnost generovaných čísel  $x_n \in \{0, 1, \dots, p-1\}$ . Ke generování čísel využívá modulární převrácenou hodnotu dle vztahu (26).

$$x_n = \overline{ax_{n-1}} + b \pmod{m} \quad n \geq 0, \quad (26)$$

kde  $m$  je modul,  $b$  aditivní člen,  $a$  multiplikační člen. V případě, že je třeba generovat čísla v intervalu  $(0;1)$  vydělíme jednotlivá generovaná čísla  $x_n$  modulem  $m$ .

- **Kvadratický kongruentní generátor**

Je dalším typem nelineárního generátoru. Pro celá čísla  $a, b, c, x_0 \in \{0, 1, \dots, m-1\}$ , kde  $a \neq 0$  je posloupnost  $x_n$  generována kvadratickým kongruentním generátorem podle vztahu (27).

$$x_n = (ax_{n-1}^2 + bx_{n-1} + c) \pmod{m}, \quad n \geq 0, \quad (27)$$

kde  $a$  je multiplikační člen,  $b$  aditivní člen a  $c$  je celé reálné číslo. Pro generování čísel v intervalu  $(0;1)$  se provede totožná matematická operace jako u inverzního kongruentního generátoru.

- **Lineární kongruentní generátor**

Jedná se o jeden z nejstarších a nejrozšířenějších lineárních generátorů pseudonáhodných čísel. Tento generátor pracuje na principu vytvoření sekvence náhodných celých čísel s uniformním rozdělením pomocí vztahu (28).

$$x_n = (ax_{n-1} + b) \pmod{m}, \quad n \geq 0, \quad (28)$$

kde  $a$  je multiplikační člen,  $b$  aditivní člen,  $m$  je modul v podobě kladného celého čísla. K dosažení, co nejlepších vlastností generovaných čísel, je nutné dodržet následující pravidla:

- aditivní člen  $b$  a modul  $m$  jsou navzájem nesoudělná čísla,
- $a-1$  je hodnota dělitelná prvočíselnými faktory  $m$ ,

- $a-1$  je hodnota dělitelná čtyřmi, jestliže  $m$  je dělitelné čtyřmi.

#### 4.2.1.2 Zpožděný Fibonacciho generátor

Zpožděný Fibonacciho generátor je jeden z nejčastěji využívaných generátorů. Důvodem je jeho jednoduchost a poměrně velká rychlost. Generátor získal název podle Fibonacciho posloupnosti, která je definována vztahem (29).

$$S_n = S_{n-1} + S_{n-2}, \quad S_0 = 1, S_1 = 1, \quad (29)$$

kde  $S_n$  je aktuální číslo a  $S_{n-1}$ ,  $S_{n-2}$  jsou dvě předešlá čísla. Ze vzorce vyplývá, že každé číslo je dáno součtem dvou předchozích čísel. Po zobecnění vztahu (29) dostáváme vztah s následujícím přepisem:

$$x_i = (x_{i-p} \times x_{i-q}) \bmod m, \quad (30)$$

kde  $p$  a  $q$  jsou skoky, přičemž  $p > q$  a znak  $\times$  reprezentuje libovolnou binární aritmetickou operaci. Příkladem může být sčítání, odčítání nebo násobení. Fibonacciho generátor využívá operace sčítání nebo odečítání. Výpočty se provádějí v pohyblivé řádové čárce, čímž je algoritmus zjednodušen o převod z celých čísel. Důležitou podmínkou u tohoto typu generátoru je správná volba skoků  $p$  a  $q$ . V případě, že  $p$  bude velmi malé číslo, tak generátor neprojde s velkou pravděpodobností ani základními testy.

#### 4.2.1.3 Blum Blum Shub

Generátor Blum Blum Shub má velmi dobré kryptografické vlastnosti. Od předešlých generátorů se liší tím, že vytváří sekvenci pseudonáhodných bitů. Je definován na základě rekurentního vztahu:

$$x_n = x_{n-1}^2 \bmod M, \quad (31)$$

kde  $\bmod M$  je dán součinem dvou velkých prvočísel. Výstup je poté jeden nebo více méně významných bitů  $x_n$  nebo paritní bit  $x_n$ .

## 4.2.2 Generátory pseudonáhodných čísel s obecným rozdělením pravděpodobnosti

V případě, že je třeba generovat pseudonáhodná čísla, která mají mít rozdělení jiné než uniformní, vznikly speciální postupy určené právě k tomuto případu. Tyto postupy budou popsány v následujících podkapitolách [14], [15].

### 4.2.2.1 Metoda přijetí zamítnutí vzorku

Předpokládejme generátor náhodné veličiny  $U$  s rovnoměrným rozdělením v intervalu  $\langle 0,1 \rangle$ . V případě, že jsme schopni generovat vzorky náhodné veličiny  $Y$  s rovnoměrným pravděpodobnostním rozdělením s hustotou  $g$  a s distribuční funkcí  $G$  můžeme definovat vztah:

$$G(y) = \int_{-\infty}^y g(s) ds. \quad (32)$$

Cílem je nalézt řešení, které pro nezápornou funkci  $h$  vybere ze vzorků  $Y$  jen takové, které budou odpovídat realizaci náhodné veličiny  $X$  s hustotou pravděpodobnosti  $h$ . Důležité předpoklady pro realizaci jsou takové, aby veličiny  $U$  a  $Y$  byly nezávislé a aby byl splněn požadavek dle vztahu (33).

$$\sup_{s \in R} \frac{h(x)}{g(x)} \leq c \in R^+. \quad (33)$$

Samotný algoritmus je poté realizován následujícím postupem:

- Vygenerování čísla  $g$  náhodné veličiny  $Y$  s distribuční funkcí  $G$ .
- Vygenerování čísla  $u$  náhodné veličiny  $U$ , nezávislé na  $Y$ .
- V případě, že je splněna podmínka  $u \leq \frac{h(y)}{c \cdot g(y)}$ , je číslo  $y$  přijato jako vzorek náhodné veličiny  $X$ . V opačném případě je číslo zahozeno a celý postup se opakuje.

### 4.2.2.2 Metoda inverzní transformace

Metoda inverzní transformace vychází z tvrzení, že pokud existuje distribuční funkce  $F$  náhodné veličiny a inverze této distribuční funkce  $F^{-1}$ . Přičemž  $U$  je náhodná veličina s uniformním rozdělením  $\langle 0,1 \rangle$ . Náhodná veličina je poté definována jako  $X = F^{-1}(U)$

a  $X$  má rozdělení pravděpodobnosti s distribuční funkcí  $F$ . Vztah, který z předešlého tvrzení vyplývá:

$$P(X \leq x) = P(F^{-1}(U) \leq x) = P(U \leq F(x)) = F(x) . \quad (34)$$

Ze vztahu (34) vyplývá, že ke každému vygenerovanému číslu  $u$  náhodné veličiny  $U$  lze přiřadit vzorek  $F^{-1}(u)$ . Tento vzorek je poté realizací náhodné veličiny  $X$  s distribuční funkcí  $F$ .

## 5 TESTOVÁNÍ GENERÁTORŮ PSEUDONÁHODNÝCH ČÍSEL

Tato kapitola se zabývá testováním generátorů náhodných čísel. Cílem těchto testů není určit zda je generátor kvalitní či nekvalitní ale vyhledat vady vzorů ve vygenerovaných sekvencích a na jejich základě generátor ohodnotit. Jelikož jsou generátory pseudonáhodných čísel důležitou a nedílnou součástí novodobé statistiky a kryptografie jsou na ně kladeny vysoké nároky z hlediska kvality. Z tohoto důvodu byla vytvořena celá sada baterií k testování těchto generátorů.

Jednou z těchto baterií je sada DIEHARD. Sada je volně dostupná z [16] a sestává z několika statistických testů, detekujících odchylky prokazující nenáhodnost testované sekvence. Nejčastěji využívanou testovací baterií je baterie NIST volně dostupná z [17] od společnosti National Institute of Standards and Technology. V této sadě je obsaženo šestnáct statistických testů. Tyto statistické testy pracují na principu testování sekvence náhodných bitů o velikosti  $n$ , ve kterých detekují statistické odchylky poukazující na nenáhodnost testované sekvence.

### 5.1 Statistické testy pseudonáhodných čísel

Následující kapitola vychází se zdroje [18], který je manuálem k testovací baterii NIST. Jelikož výše zmíněné testovací baterie testují náhodnost generovaných posloupností čísel a náhodnost je pravděpodobnostní vlastností, je možné tyto testy charakterizovat pomocí pojmů z oblasti pravděpodobnosti. Statistické testy jsou vymezeny testováním *nulové hypotézy*. Tato hypotéza říká, že testovanou sekvenci lze považovat za náhodnou. Z tohoto důvodu je cílem testů minimalizovat pravděpodobnost přijetí sekvence, kterou nelze považovat za náhodnou.

Každý statistický test je založen na počítání statistické hodnoty. K vyjádření slouží *p-hodnota*, která je definována jako pravděpodobnost, že za předpokladu pravdivosti nulové hypotézy bude statistická hodnota alespoň tak velká jako zjištěná hodnota. Podle této hodnoty se následně statistické testy vyhodnocují. Pokud je *p-hodnota* rovna 1, je možné tvrdit, že testovaná sekvence je skutečně náhodná. V opačném případě pokud je hodnota rovna 0, testovaná sekvence není náhodná. Pokud *p-hodnota* je větší než 0,05 pak nelze o testované sekvenci tvrdit, že není náhodná. V případě, že *p-hodnota* nabývá hodnot v intervalu definovaném v testovací baterii NIST jako (0.01;0.05), tak test není spolehlivý a měl by se opakovat.

V následujících podkapitolách jsou stručně popsány některé testy, které jsou obsaženy v testovací baterii NIST a jsou dále využity k testování generátorů pseudonáhodných čísel. Detailní popis jednotlivých testů je obsažen v manuálu k testovací baterii NIST ve zdroji [18].

### 5.1.1 Frekvenční test

Frekvenční test je určen k ověření rovnoměrnosti rozdělení generovaných čísel. Pravděpodobnost, že náhodná veličina s rovnoměrným rozdělením v intervalu  $\langle 0,1 \rangle$ , nabude hodnoty z intervalu  $\langle \alpha, \beta \rangle$ , kde  $0 \leq \alpha \leq \beta \leq 1$ , je rovna  $\beta - \alpha$ . V případě rozdělení jednotkového intervalu na  $k$  úseků a vygenerování  $n$  náhodných čísel, je možné testovat nulovou hypotézu o shodě očekávaných a skutečných četností výskytů čísel v jednotlivých úsecích. Počet jednotlivých úseků je dán vztahem (35).

$$k = 4.5 \sqrt{\frac{2(n-1)^2}{c^2}}, \quad (35)$$

kde  $c$  je voleno tak, aby platilo:

$$\alpha = \frac{1}{\sqrt{2\pi}} \int_c^{\infty} e^{-\frac{\pi^2}{2}}. \quad (36)$$

### 5.1.2 Blokový frekvenční test

Blokový frekvenční test testuje zastoupení jedniček v bloku o délce  $M$ . Cílem blokového testu je ověřit, zda absolutní četnosti jsou přibližně  $M/2$ , jako u skutečně náhodné sekvence. Neúspěch testu značí velkou odchylku počtu jedniček alespoň v jednom bloku. Velikost bloků by měla být volena alespoň  $M \geq 20$  a délka testované sekvence alespoň 100 bitů.

### 5.1.3 Test hodnosti binární matice

Tento test je zaměřen na testování hodností podmatic z testované sekvence. Cílem je zjistit, zda existuje lineární závislost mezi podřetězci původní sekvence. Sekvence se rozdělí do  $N$  disjunktivních matic o rozměrech  $M \times Q$  a určí se její hodnost. Nevýhodou testu je počet vytvořených matic, který musí být alespoň 38. Velikost takto vytvořených matic musí být minimálně  $32 \times 32$ , z čehož plyne délka sekvence alespoň 38 912 bitů.

### 5.1.4 Test sérií

Test sérií spočívá v celkovém počtu jedniček a nul v testované sérii. Za sérii je považován nepřerušovaný řetězec stejných bitů. Série délky  $k$  určuje řetězec délky  $k$  stejných bitů ohraničených bity s jinou hodnotou. Výsledkem testu je ověřit zda počet sérií nul

a jedniček různých délek je stejný jako u náhodné sekvence. Výpočet je realizován dle vztahu (37).

$$\lim_{n \rightarrow \infty} P\left(\frac{V_n - 2n\pi(1-\pi)}{2\sqrt{n\pi(1-\pi)}} \leq z\right) = \phi(z), \quad (37)$$

kde  $\phi(z)$  je distribuční funkce normovaného normálního rozdělení.

### 5.1.5 Test shody nepřekrývajících se řetězců

Test je zaměřen na výskyt předem specifikovaných řetězců v testované sekvenci. Cílem je zjistit, zda generátor čísel nevytváří příliš mnoho neperiodických vzorů, předem definovaných uživatelem. Test pracuje na vytvoření  $m$ -bitového okna, pomocí něhož vyhledává  $m$ -bitové vzory. Pokud test hledaný bit nenalezne, posune okno o jeden bit dál v testované sekvenci. V případě nalezení vzoru se okno posune na bit ležící za nalezeným řetězcem.

K provedení testu se testovací sekvence rozdělí do  $N$  bloků velikosti  $M$ . Pro vzor délky  $m$  se zjistí četnost výskytu  $W_j$ . V případě skutečné náhodnosti platí, že střední hodnota je dána vztahem (39) a rozptyl vztahem (40). Testovací statistika je poté dána předpisem:

$$\chi^2(obs) = \sum_{j=1}^N \frac{(W_j - \mu)^2}{\sigma^2}, \quad (38)$$

kde  $W_j$  jsou četnosti výskytu vzoru v daných blocích,  $\mu$  je střední hodnota definována vztahem:

$$\mu = (M - m + 1) / 2^m, \quad (39)$$

$\sigma^2$  je rozptyl vyjádřen vztahem:

$$\sigma^2 = M \left( \frac{1}{2^m} - \frac{2m-1}{2^{2m}} \right). \quad (40)$$

### 5.1.6 Test shody překrývajících se řetězců

Test překrývajících se řetězců je zaměřen stejně jako předešlý test na výskyt předem specifikovaných řetězců uživatelem v testované sekvenci. Test využívá  $m$ -bitové číslo (okno) pro vyhledávání  $m$ -bitových vzorů. V případě, že hledaný vzor není nalezen,

okno se posune o jeden bit dál v prohledávané sekvenci. Pokud je vzor nalezen okno se posune na následující bit, čímž je zajištěno nalezení i vzájemně se překrývajících řetězců. Vlastní test se provede tak, že testovací sekvence je rozdělena do  $N$  bloků velikosti  $M$ . Následně se vypočte četnost výskytu vzoru  $B$  délky  $m$  v jednotlivých blocích. Výskyt hledaného vzoru se zaznamená inkrementací buňky v poli  $v_i$ , takovým způsobem, že při nulovém výskytu vzoru  $B$  se inkrementuje buňka  $v_0$ , při jednom výskytu vzoru se inkrementuje buňka  $v_1$  atd. Při pěti nebo více výskytech vzoru  $B$  se inkrementuje buňka  $v_5$ . Vztah pro testovací statistiku je dán předpisem:

$$\chi^2(obs) = \sum_{i=0}^5 \frac{(v_i - N\pi_i)^2}{N\pi_i}, \quad (41)$$

kde  $v_i$  jsou počty bloků a  $\pi_i$  jsou teoretické pravděpodobnosti.

### 5.1.7 Test lineární složitosti

Tento test je zaměřen na délku lineárního zpětnovazebního posuvného registru. Úkolem je rozhodnout, zda je možné považovat testovanou sekvenci za náhodnou z hlediska komplexnosti. Testovaná sekvence se rozdělí na  $N$  bloků velikosti  $M$ . Dále se určí lineární složitost  $L_i$  všech bloků pomocí Masseyho algoritmu. V případě náhodnosti se vypočítá střední hodnota  $\mu = \frac{M}{2} + \frac{9 + (-1)^{M+1}}{36} - \frac{(M/3) + (2/9)}{2^M}$ . Poté je nutné pro jednotlivé bloky určit hodnoty  $T_i = (-1)^M \cdot (L_i - \mu) + (2/9)$ . Následná testovací statistika je dána předpisem:

$$\chi^2(obs) = \sum_{i=0}^6 \frac{(v_i - N\pi_i)^2}{N\pi_i}, \quad (42)$$

kde  $\pi_i$  jsou teoretické pravděpodobnosti.

### 5.1.8 Spektrální test

Spektrální test pomocí Fourierovy transformace ověřuje statické rozdělení  $n$ -tic náhodných čísel, které následují za sebou. Cílem jsou výšky vrcholů v diskretní Fourierově transformaci. Test spočívá v detekci opakujících se vzorů, které jsou ve vzájemné blízkosti v dané posloupnosti. Tyto opakující se vzory by mohly indikovat odchylku z hypotézy náhodnosti. Výsledkem testu je, zda počet vrcholů, které jsou vyšší než 95% se výrazně liší od vrcholů, které jsou vyšší než 5%. K získání vztahu pro výpočet je nutné nejprve stanovit hranici  $T = \sqrt{(\ln(\frac{1}{0,05})).n}$ , kterou by za předpokladu náhodnosti

nemělo překročit 95% hodnot. Testovací statistika je poté dána předpisem podle vztahu (43).

$$d = \frac{N_1 - N_0}{\sqrt{0,95 \cdot 0,05 \cdot \frac{n}{4}}}, \quad (43)$$

kde  $N_1$  je počet vrcholů, které nepřesáhly hodnotu  $T$ .  $N_0$  je očekávaný počet vrcholů, které nepřesáhly hodnotu  $T$ .

## 6 TESTOVÁNÍ VYBRANÝCH GENERÁTORŮ ČÍSEL

Tato kapitola se zabývá testováním generátorů náhodných čísel implementovaných v softwarech, které jsou nejčastěji využívány pro výpočet nejistot pomocí metody Monte Carlo. Testované aplikace jsou Matlab ve verzi R2013b, Microsoft Excel 2013, LabVIEW, GNU Octave, Maple 17. K testování je využita testovací baterie NIST [17].

Vstupními daty pro tuto testovací baterii jsou sekvence bitů o délce  $n$ . Jelikož generátory náhodných čísel, které využívají testované softwary generují čísla v desítkové soustavě, je nutné generovaná čísla převést do dvojkové soustavy. Z důvodu zachování vlastností vygenerovaných čísel a zachování rozdělení nul a jedniček jsou všechna čísla v baterii zapsána pomocí deseti bitů. Převod z desítkové soustavy do dvojkové soustavy a následný zápis pomocí deseti bitů byl pro všechny generátory proveden v programu Matlab. Převod spočíval vždy v načtení souboru vygenerovaných dat od jednotlivých generátorů. Takto načtená data se poté upravila na vybraný interval testování, což byl v tomto případě interval  $\langle 0;1023 \rangle$ . Poté byla takto upravená data převedena do dvojkové soustavy tak, aby každé číslo bylo reprezentováno deseti bity.

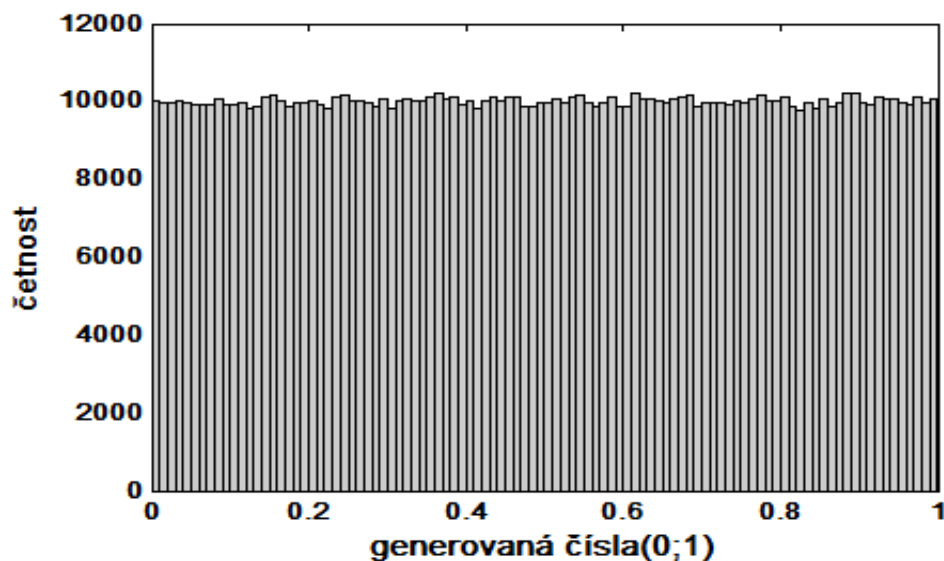
Dle předešlých parametrů byly vytvořeny dvě varianty sekvencí generovaných hodnot od jednotlivých generátorů z výše uvedených softwarů. V první variantě byly generovány čtyři na sobě nezávislé sekvence bitů z  $10^3$  generovaných hodnot od každého generátoru. Délka těchto sekvencí tedy byla  $n = 10^4$ . V druhé variantě byly opět od každého generátoru generovány čtyři na sobě nezávislé sekvence bitů, ale z  $10^5$  generovaných hodnot. Tím pádem délka těchto sekvencí byla  $n = 10^6$ . Čtyři sekvence a dvě varianty počtů generovaných čísel byly zvoleny z důvodu potvrzení výsledků, jelikož v případě otestování pouze jedné sekvence by mohlo dojít k nespolehlivým výsledkům. Poté byly všechny tyto vygenerované sekvence otestovány sérií testů obsažených v testovací baterii NIST. Tato série obsahovala frekvenční test, blokový frekvenční test, test hodnosti binární matice, sériový test, test lineární složitosti a spektrální test. Výstupem těchto testů byla  $p$ -hodnota, která byla následně zapsána do tabulek podle příslušnosti testovaného generátoru k danému softwaru.  $P$ -hodnota vyjadřuje jaká je minimální hodnota hladiny významnosti, na které je možné zamítnout nulovou hypotézu, jak již bylo popsáno v kapitole 5.1. U prováděných testů je hladina významnosti nastavena na hodnotu 0,01 viz. [18]. V případě, že  $p$ -hodnota bude větší než nastavená hladina významnosti, je test úspěšný. Následně byly u obou variant vypočteny pro každý test průměrné  $p$ -hodnoty vždy ze čtyř testovaných sekvencí. Tyto průměrné hodnoty budou sloužit k názornějšímu zhodnocení jednotlivých generátorů, jelikož může nastat situace, že u některé testované sekvence získáme výsledek testu menší než 0,01. V tomto případě by měla být sekvence považována za nenáhodnou, což nemusí značit nenáhodnost celého generátoru čísel. Proto je průměrná  $p$ -hodnota vytvářena ze čtyř nezávislých sekvencí.

## 6.1 MATLAB R2013b

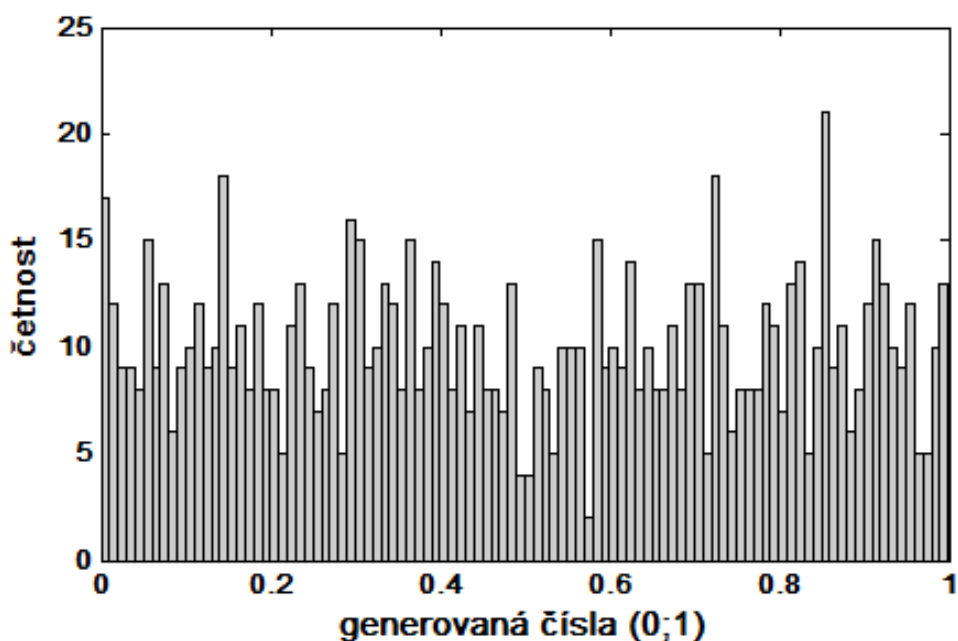
Matlab [29] je interaktivní programové prostředí a skriptovací jazyk čtvrté generace. Za vývoj odpovídá společnost MathWorks. Matlab umožňuje využívat celou škálu matematických a simulačních operací. Například vykreslování 2D i 3D grafů, práce s maticemi, implementace algoritmů, analýza dat, prezentace dat, počítačové simulace, vytváření uživatelského rozhraní atd. Hlavní využití programu Matlab je v oblasti ekonomie a technických oborů. Ke generování náhodných čísel program využívá Matlab MT 3.4 generátor.

Důležitými částmi programu jsou knihovny funkcí (toolboxy), simulink, guide a rozhraní pro jiné programovací jazyky. Knihovny funkcí obsahují uceleným způsobem, včetně dokumentace a příkladů, zpracovaný určitý obor numerické matematiky, statistiky, analytické matematiky, systémového přístupu k regulaci. Momentálně je k dispozici 35 těchto knihoven. Simulink je program, který využívá MATLAB k simulaci dynamických systémů. Guide je prostředí umožňující vytvářet aplikace s grafickým rozhraním. Rozhraní pro jiné programovací jazyky umožňuje vytvářet Java komponenty z matlabovských programů [19], [20].

Na následujících dvou obrázcích je možné vidět rovnoměrné rozložení generovaných náhodných čísel v programu Matlab R2013b na intervalu  $\langle 0;1 \rangle$  v případě, že je generováno  $10^6$  náhodných čísel nebo  $10^3$  náhodných čísel. Toto rozložení mají generátory čísel ve všech softwarech podobné, proto je třeba pro výpočty nejistot pomocí simulačních metod, jako je například metoda Monte Carlo, generovat takový počet hodnot, aby výsledné rozložení co nejvíce odpovídalo skutečnému. Z tohoto důvodu je třeba generovat více než  $\frac{10^4}{1-p}$  hodnot [11], kde  $p$  je pravděpodobnost pokrytí. Pokud je k dispozici výkonný software, je doporučeno generovat  $10^6$  náhodných (pseudonáhodných) hodnot [11].



Obrázek 3 Rozdělení čísel při vygenerování  $10^6$  náhodných čísel.



Obrázek 4 Rozdělení čísel při vygenerování  $10^3$  náhodných čísel.

V tabulce 6-1 jsou uvedeny výsledky vybraných testů provedených na generátoru Matlab MT 3.4 s testovanými sekvencemi vytvořenými z  $10^3$  čísel. Výsledky jsou reprezentovány *p-hodnotami*. Pro zpřehlednění bude v tabulkách výsledků jednotlivých testů generátorů uváděna i průměrná *p-hodnota*, která bude vypočtená ze čtyř testovaných sekvencí. Důvodem vytváření průměrné hodnoty u jednotlivých testů z více testovaných sekvencí je zpřehlednění jejich výsledků. Z tabulky 6-1 je patrné, že testovaný generátor obsažený v programu Matlab R2013b na testovaných sekvencích vytvořených z  $10^3$  hodnot všemi testy prošel. Nejlépe dopadl test lineární složitosti, jehož průměrná *p-hodnota* je nejvyšší. Nejhuře naopak dopadl test sérií, protože jeho průměrná *p-hodnota* je nejnižší. Detailnější zhodnocení jednotlivých testů bude obsaženo v závěrečné podkapitole 6.6.

Tabulka 6-1 Výsledky testů MATLAB MT 3.4 pro  $10^3$  generovaných hodnot.

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,952155	0,825871	0,872881	0,548506	0,581633
Blokový frekvenční test	0,533320	0,785450	0,621197	0,189375	0,532335
Test hodnosti binární matice	0,862457	0,648387	0,441305	0,862457	0,703651
Sériový test	0,574998	0,557892	0,448790	0,137610	0,2903495
Test lineární složitosti	0,971599	0,703577	0,663002	0,945036	0,820803
Spektrální test	0,713570	0,217814	0,664574	0,646355	0,506124

V tabulce 6-2 jsou uvedeny výsledky vybraných testů provedených na generátoru Matlab MT 3.4 s testovanými sekvencemi vytvořenými z  $10^5$  čísel. Výsledky jsou opět reprezentovány *p-hodnotami*. Testovaný generátor obsažený v programu Matlab R2013b opět všemi testy prošel. Nejlépe se dle výsledků testů jeví blokový frekvenční test společně s testem lineární složitosti. Nejhůře naopak test sérií a test hodnosti binární matice.

**Tabulka 6-2 Výsledky testů MATLAB MT 3.4 pro  $10^5$  generovaných hodnot.**

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,462948	0,169441	0,690630	0,797326	0,530086
Blokový frekvenční test	0,941387	0,188475	0,484513	0,797326	0,602925
Test hodnosti binární matice	0,238095	0,909851	0,424899	0,182263	0,438777
Sériový test	0,453133	0,134404	0,510904	0,736943	0,458846
Test lineární složitosti	0,593494	0,660944	0,232402	0,784532	0,567843
Spektrální test	0,693141	0,383328	0,919595	0,154916	0,537745

Díky dosaženým výsledkům z jednotlivých testů pro obě varianty generovaných čísel, obsažených v tabulkách 6-1, 6-2, můžeme tvrdit, že na základě teorie obsažené v kapitole 5.1, je generátor v programu Matlab R2013b dostatečně kvalitní. Na základě toho není možné vyvrátit hypotézu, že generovaná čísla nejsou náhodná.

## 6.2 Microsoft Excel 2013

Microsoft Excel je tabulkový editor od firmy Microsoft. Od roku 1993 má dominantní postavením na trhu. Dnes se prodává jako součást kancelářského balíku Microsoft Office. Obsahuje přibližně 300 funkcí, výpočetní nástroje, grafické nástroje, kontingenční tabulky. Program také umožňuje programování pomocí maker. Ke generování čísel tento program využívá generátor pseudonáhodných čísel nazvaný jako AS 183.

Program pracuje s buňkami, které jsou uspořádány do tabulky. Vztahy mezi těmito buňkami definuje uživatel, přičemž změna obsahu jedné buňky může vyvolat změnu jedné nebo více buněk. Do buněk může uživatel definovat data a vzorce pracující s těmito daty. Soubor, který je vytvořen v tomto editoru se nazývá tabulkový sešit, který se sestává z tabulkových listů [21].

V tabulce 6-3 jsou uvedeny výsledky vybraných testů. Tyto testy byly provedeny na generátoru čísel obsaženém v programu Microsoft Excel 2013. Testované sekvence byly vytvořeny z  $10^3$  hodnot. Testované sekvence všemi testy úspěšně prošly.

Z průměrných *p-hodnot* je možné říci, že nejlépe dopadl test lineární složitosti společně se spektrálním testem. Nejhůře dopadl frekvenční test a blokový frekvenční test.

**Tabulka 6-3 Výsledky testů pro generátor AS 183 využívaný programem Excel pro  $10^3$  čísel.**

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,423710	0,180245	0,115520	0,703945	0,355855
Blokový frekvenční test	0,406472	0,642806	0,341214	0,192782	0,395819
Test hodnoti binární matice	0,374305	0,713851	0,374305	0,499513	0,490494
Sériový test	0,498335	0,438644	0,376722	0,821863	0,533891
Test lineární složitosti	0,676532	0,484288	0,757083	0,717050	0,658738
Spektrální test	0,118753	0,926883	0,783086	0,813223	0,660486

V tabulce 6-4 jsou uvedeny výsledky vybraných testů pro generátor obsažený v programu Microsoft Excel 2013. Rozdílem od tabulky 6-3 je délka testované sekvence. V tomto případě byla každá sekvence vytvořena z  $10^5$  čísel. Všechny testy proběhly opět úspěšně.

**Tabulka 6-4 Výsledky testů pro generátor AS 183 využívaný programem Excel pro  $10^5$  čísel.**

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,604458	0,931466	0,709892	0,665741	0,727889
Blokový frekvenční test	0,439210	0,668842	0,698871	0,917630	0,681138
Test hodnoti binární matice	0,714807	0,552949	0,875616	0,592103	0,683869
Sériový test	0,139445	0,896872	0,205807	0,810133	0,513064
Test lineární složitosti	0,868223	0,077880	0,172259	0,407729	0,381523
Spektrální test	0,861585	0,243844	0,569389	0,358795	0,508403

Vzhledem k výsledkům dosažených v jednotlivých testech u obou variant délky sekvencí, můžeme z hlediska teorie uvedené v kapitole 5.1 tvrdit, že generátor obsažený v programu Microsoft Excel 2013 je dostatečně kvalitní. Čísla, která tento generátor generuje, nelze považovat za nenáhodná.

## 6.3 Maple 17

Maple je algebraický počítačový systém vyvíjený společností Maplesoft. Výhodou tohoto systému je, že uživatelé mohou zadávat matematické operace v tradičním matematickém zápisu nebo si mohou vytvořit vlastní uživatelské rozhraní. Maple je napsán pomocí dynamického programovacího jazyka, který obsahuje předdefinované funkce a proměnné. Předdefinované funkce pokrývají velkou část matematického odvětví. Například diferenciální a integrální počet, lineární algebru, řešení diferenciálních rovnic a mnohé další [22].

V tabulce 6-5 jsou uvedeny výsledky vybraných testů pro generátor čísel obsažený v programu Maple 17. Vytvořené sekvence, které byly následně podrobeny jednotlivým testům, byly vytvořeny z  $10^3$  čísel. Všechny tyto sekvence všemi testy prošly úspěšně. Z hlediska průměrných *p-hodnot* nejlépe dopadl test lineární složitosti.

**Tabulka 6-5** Výsledky testů pro generátor v programem Maple 17 pro  $10^3$  vzorků.

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,496504	0,180245	0,167586	0,674485	0,379705
Blokový frekvenční test	0,359099	0,195052	0,589307	0,376467	0,379981
Test hodnoty binární matice	0,587006	0,157493	0,499513	0,137209	0,345305
Sériový test	0,471849	0,359587	0,109937	0,292534	0,308477
Test lineární složitosti	0,900309	0,808713	0,910169	0,521364	0,785139
Spektrální test	0,168668	0,312768	0,232884	0,198887	0,228302

V tabulce 6-5 jsou uvedeny výsledky vybraných testů opět pro generátor čísel obsažený v programu Maple 17. Rozdíl je v délce testované sekvence. Testované sekvence v této části měly délku  $10^5$  čísel. Opět všechny testované sekvence všemi testy prošly úspěšně. Z hlediska průměrných *p-hodnot* nejlépe dopadl test lineární složitosti.

**Tabulka 6-6** Výsledky testů pro generátor v programem Maple 17 pro  $10^5$  vzorků.

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,523473	0,771816	0,394214	0,770286	0,614947
Blokový frekvenční test	0,728551	0,844340	0,243197	0,936676	0,688191
Test hodnoty binární matice	0,450108	0,939267	0,103962	0,397584	0,472730
Sériový test	0,477935	0,167697	0,912318	0,771151	0,582275
Test lineární složitosti	0,925674	0,853386	0,126552	0,361927	0,766885
Spektrální test	0,613759	0,854380	0,727305	0,926883	0,780582

Jelikož všechny *p-hodnoty* u všech testů u obou testovaných variant odpovídají nezamítnutí nulové hypotézy. Podle kapitoly 5.1 můžeme tvrdit, že generátor čísel v programu Maple 17 je dostatečně kvalitní a čísla, která generuje, nelze považovat za nenáhodná.

## 6.4 LabVIEW 2013

LabVIEW [23] je grafický programovací jazyk vytvořený společností National Instruments. Program je obecným vývojovým prostředím s velkým množstvím knihoven pro vytváření aplikací zaměřených na oblasti sběru, analýzy a prezentaci naměřených výsledků. Poskytuje uživateli plnohodnotný programovací jazyk se všemi potřebnými datovými a programovými strukturami v grafické podobě. Výsledný produkt tohoto vývojového prostředí se nazývá Virtual Instrument, jelikož svým chováním a projevem připomíná klasický přístroj ve fyzické podobě. Základní jednotka aplikace vytvořená v tomto prostředí obsahuje:

- **Interaktivní grafické rozhraní** (čelní panel) je simulující čelní panel fyzického přístroje. Obsahuje prvky pro ovládání a indikaci. Čelní panel je ovládán pomocí klávesnice nebo myši.
- **Blokové schéma** (Block Diagram) je tvořeno ikonami reprezentujícími ovládací a indikační prvky čelního panelu. Dále ve svých uzlových blocích reprezentuje blokové schéma bloky zpracovávající procházející data. Blokový diagram je zdrojovou podobou každé aplikace.
- **Podřízené virtuální přístroje** (Sub-VI). Jelikož má virtuální přístroj hierarchickou a modulární strukturu, lze je využívat jako celý program nebo jednotlivé podprogramy (Sub-VI).
- **Vývojové prostředí** LabVIEW je k dispozici pro standardně používané platformy. Zdrojové kódy aplikace vytvořené v tomto prostředí jsou mezi jednotlivými platformami vzájemně přenositelné.

V tabulce 6-7 jsou uvedeny výsledky vybraných testů pro generátor čísel obsažený v programu LabVIEW. Vytvořené sekvence, které byly následně podrobeny jednotlivým testům, byly vytvořeny z  $10^3$  čísel. Všechny tyto sekvence všemi testy prošly úspěšně. Z hlediska průměrných *p-hodnot* nejlépe dopadl test lineární složitosti a nejhůře frekvenční test.

Tabulka 6-7 Výsledky testů pro generátor čísel v programu LabVIEW pro  $10^3$  vzorků.

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,123560	0,155607	0,825871	0,904483	0,502380
Blokový frekvenční test	0,784185	0,890247	0,369117	0,974611	0,754540
Test hodnoty binární matice	0,862457	0,949535	0,374305	0,499513	0,671453
Sériový test	0,920876	0,911127	0,867856	0,345880	0,761435
Test lineární složitosti	0,971599	0,505819	0,985608	0,904483	0,841877
Spektrální test	0,713570	0,217814	0,646355	0,664574	0,560578

V tabulce 6-8 jsou uvedeny výsledky vybraných testů opět pro generátor čísel obsažený v programu LabVIEW. Rozdíl je v délce testované sekvence. Testované sekvence v této části měly délku  $10^5$  čísel. Opět všechny testované sekvence všemi testy prošly úspěšně.

Tabulka 6-8 Výsledky testů pro generátor čísel v programu LabVIEW pro  $10^5$  vzorků.

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,312419	0,273954	0,222464	0,362321	0,292790
Blokový frekvenční test	0,744335	0,965088	0,472568	0,111399	0,573348
Test hodnoty binární matice	0,303857	0,406966	0,539864	0,266638	0,379331
Sériový test	0,682494	0,517232	0,290115	0,670358	0,540050
Test lineární složitosti	0,311320	0,161424	0,441584	0,211911	0,281560
Spektrální test	0,963403	0,120937	0,532620	0,255161	0,468030

Na základě výsledků jednotlivých testů můžeme tvrdit, že generátor čísel obsažený v programu LabVIEW je dostatečně kvalitní.

## 6.5 GNU Octave

GNU Octave je volně dostupný software pro provádění různých číselných výpočtů. Software je šířen pod licencí GPL (General Public Licence). Octave obsahuje rozsáhlý soubor nástrojů pro řešení numerických či algebraických problémů. Dále se pomocí něho dají řešit nelineární rovnice, integrace a derivace funkcí, integrování diferenciálních rovnic. Základní strukturou v programu Octave jsou matice. Je možné použít i řídké matice [24].

V tabulce 6-9 jsou uvedeny výsledky vybraných testů pro generátor čísel obsažený v programu GNU Octave. Vytvořené sekvence, které byly následně podrobeny jednotlivým testům, byly vytvořeny z  $10^3$  čísel. Všechny tyto sekvence všemi testy prošly úspěšně. Z hlediska průměrných *p-hodnot* nejlépe dopadl test lineární složitosti a nejhůře spektrální test.

**Tabulka 6-9** Výsledky testů generátoru čísel, využívaného programem Octave pro  $10^3$  hodnot.

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,289144	0,157433	0,841480	0,857152	0,536302
Blokový frekvenční test	0,865970	0,773623	0,250535	0,817296	0,676856
Test hodnosti binární matice	0,066919	0,439867	0,157493	0,166919	0,207800
Sériový test	0,510242	0,062441	0,388835	0,720342	0,420465
Test lineární složitosti	0,821158	0,460373	0,757082	0,770221	0,702209
Spektrální test	0,462869	0,118753	0,232884	0,358795	0,293325

V tabulce 6-10 jsou uvedeny výsledky vybraných testů pro generátor obsažený v programu GNU Octave. Rozdílem od tabulky 6-9 je délka testované sekvence. V tomto případě byla každá sekvence vytvořena z  $10^5$  čísel. Všechny testy proběhly opět úspěšně.

**Tabulka 6-10** Výsledky testů generátoru čísel, využívaného programem Octave pro  $10^5$  hodnot.

Název testu	p-hodnota	p-hodnota	p-hodnota	p-hodnota	průměrná p-hodnota
Frekvenční test	0,281033	0,625549	0,526080	0,765703	0,226646
Blokový frekvenční test	0,265969	0,847877	0,442559	0,490991	0,278462
Test hodnosti binární matice	0,638298	0,460701	0,705360	0,751583	0,274750
Sériový test	0,972413	0,574466	0,227724	0,317986	0,243103
Test lineární složitosti	0,142998	0,591671	0,586829	0,588668	0,183667
Spektrální test	0,783086	0,713570	0,12767	0,152834	0,374164

Vzhledem k výsledkům dosažených v jednotlivých testech u obou variant délky sekvencí můžeme tvrdit, že generátor obsažený v programu GNU Octave je dostatečně kvalitní. Sekvence čísel, které tento generátor generuje nelze považovat za nenáhodné.

## 6.6 Zhodnocení testů

U generátorů čísel obsažených v programech Matlab R2013b, LabVIEW 2013, GNU Octave, Maple 17, Microsoft Excel 2013 byly provedeny vybrané statistické testy z testovací baterie NIST [17]. Vybranými testy byl frekvenční test, blokový frekvenční test, test hodnosti binární matice, sériový test, test lineární složitosti, spektrální test. Cílem těchto testů bylo ověřit, zda testované sekvence vygenerované v jednotlivých generátorech jsou co nejvíce podobné náhodným sekvencím. Jako vstupní data do těchto testů sloužily vygenerovaná čísla s rovnoměrným rozdělením na intervalu  $\langle 0;1023 \rangle$ , která byla reprezentována deseti bity v dvojkové soustavě. Pro každý generátor byly testovány čtyři sekvence vytvořené z  $10^3$  generovaných hodnot a čtyři sekvence vytvořené z  $10^5$  generovaných hodnot. Výstupem testů byly vždy *p-hodnoty*, podle kterých bylo možné posoudit náhodnost dané sekvence.

Z výstupních *p - hodnot* jednotlivých testů vyplývá, že u žádného testovaného generátoru nelze vyvrátit nulovou hypotézu. Důvodem je, že u všech *p-hodnot* jednotlivých testů je hodnota vyšší než hladina významnosti, která má hodnotu 0,01. V tabulkách 6-1 až 6-10 jsou zaznamenány příslušné *p-hodnoty* pro jednotlivé testy a jednotlivé sekvence. Je možné si všimnout lišících se hodnot u stejného testu pro různé sekvence. Tato odlišnost je způsobena tím, že každá sekvence byla generována nezávisle na ostatních. U každého testu je také uvedena průměrná *p-hodnota*. Tato hodnota je uváděna z důvodu, kdyby některá ze čtyř sekvencí nesplnila kritéria daného testu. V takovém případě pokud ostatní sekvence budou hodnoceny kladně, vyjde průměrná hodnota splňující kritéria testu a test může být hodnocen kladně. Na základě získaných *p - hodnot* a výše uvedených důvodů je možné říci, že generátory implementované v testovaných softwarech jsou dostatečně kvalitní, abychom je mohli považovat za náhodné a bylo možno v nich provádět simulace pomocí metody Monte Carlo.

Při porovnání průměrných *p-hodnot* u jednotlivých testů se nejlépe jeví generátor obsažený v programu Matlab R2013b. Důvodem je velikost průměrných hodnot u všech provedených testů, které se nejvíce blíží hodnotě jedna. Přičemž hodnota jedna značí náhodnou sekvenci. Nejhoršího výsledku z hlediska testů dosáhl generátor čísel obsažený v programu GNU Octave. Nicméně i tento generátor z hlediska nulové hypotézy odpovídá předpokladu náhodnosti generovaných čísel.

## 7 VÝPOČET NEJISTOTY NEPŘÍMÉHO MĚŘENÍ PROUDU NUMERICKOU METODOU

Následující kapitola obsahuje příklad demonstrující postup stanovení nejistoty měřeného stejnosměrného proudu. Jedná se o ukázkový příklad, na kterém bude demonstrováno, že je možné provést stanovení nejistoty měření několika způsoby. Prvním způsobem bude výpočet nejistoty proudu klasickou numerickou metodou. Druhým způsobem bude výpočet nejistoty proudu pomocí metody Monte Carlo ve třech různých softwarech. Cílem bude dokázat, zda pomocí simulací v jednotlivých softwarech dosáhneme stejného výsledku, který bude navíc srovnatelný s výpočtem pomocí klasické numerické metody. Příklad není složitý na výpočet, ale jeho značnou výhodou je, že se v něm vyskytuje více typů rozdělení nejistot typu B, na něž je při výpočtu brán potaz. Výpočty tohoto příkladu jsou založeny na dokumentu nacházejícím se ve zdrojích [4] a [2].

Úkolem tohoto příkladu bude stanovit přesnou hodnotu stejnosměrného elektrického proudu tekoucího rezistorem, přičemž budeme vycházet z Ohmova zákona. Stejnosměrný proud bude stanovován pomocí napětí na měřicím rezistoru. Jako zdroj měřeného proudu byl použit STATRON 2223 [26] s výrobním číslem 0306006. Jako měřicí rezistor byla využita odporová dekáda R1-100 nastavena na nominální hodnotu  $R = 3 \Omega$ . Úbytek napětí byl měřen pomocí digitálního multimetru Agilent 34401A s výrobním číslem MY41002147 od firmy Agilent s nastavením vysoké vstupní impedance [25]. Pro účely měření byla pokojová teplota udržována v rozmezí  $(23 \pm 1) ^\circ\text{C}$ . Všechny vstupní veličiny byly považovány za nekorelované. Soubor vstupních veličin je uveden v následujících tabulkách.

**Tabulka 7-1 Přesnost měřicího rezistoru.**

Hodnota odporu	3 $\Omega$
Teplota místnosti v době měření	23 $^\circ\text{C}$
Hodnota relativního teplotního koeficientu	50ppm
Přesnost	1%

**Tabulka 7-2 Přesnost měření stejnosměrného napětí multimetrem Agilent 34401A [25].**

Rozsah	Přesnost $\pm(\% \text{ rdg} + \% \text{ rozsahu})$
100 mV	0,0050+0,0035
1 V	0,0040+0,0007

## 7.1 Výpočet hodnoty měřeného proudu

Výsledná hodnota měřeného stejnosměrného elektrického proudu byla vypočtena z Ohmova zákona ( $I = U / R$ ), přičemž za napětí  $U$  je dosazen výběrový aritmetický průměr z naměřených hodnot v tabulce 7-3, za hodnotu odporu  $R$  měřicího rezistoru je dosazena hodnota z tabulky 7-1.

Výběrový aritmetický průměr z naměřených hodnot napětí viz. tabulka 7-3:

$$\bar{U} = \frac{1}{n} \sum_{i=1}^n U_i = \frac{1}{10} (640,69 + 640,66 + \dots + 640,58) \cdot 10^{-3} = 640,63 \cdot 10^{-3} \text{ V} \quad (44)$$

Stejnoseměrný elektrický proud tekoucí obvodem je poté vypočten ze vztahu:

$$I = \frac{\bar{U}}{R} = \frac{640,628 \cdot 10^{-3}}{3} = 213,5 \cdot 10^{-3} \text{ A} \quad (45)$$

Tabulka 7-3 Naměřené hodnoty napětí na měřicím rezistoru.

Číslo měření	1	2	3	4	5	6	7	8	9	10
Napětí [mV]	640,69	640,66	640,67	640,53	640,58	640,59	640,69	640,58	640,64	640,65

## 7.2 Výpočet dílčích nejistot měření

Na výsledné nejistotě měření se podílí celá řada složek. Výčet těchto složek je následující:

- Náhodné chyby měření, které mají vliv na naměřenou hodnotu napětí.
- Nejistota měřicího přístroje.
- Nejistota odporu použitého rezistoru.
- Nejistota odporu spojená s vlivem teploty místnosti a okolí.

### Výpočet standardní nejistoty typu A:

Bylo provedeno deset měření napětí na rozsahu 1 V. Naměřené hodnoty odpovídají normálnímu (Gaussovu) rozložení pravděpodobnosti. Mírou nejistoty, která vyjadřuje nejistotu náhodných chyb měření napětí je tedy odhad výběrové směrodatné odchylky výběrového aritmetického průměru. V tabulce 7-4 jsou pro názornost znovu uvedeny naměřené hodnoty napětí a vyjádřeny některé dílčí výrazy ze vztahu (46).

Tabulka 7-4 Tabulka naměřených a vypočtených hodnot napětí.

Číslo měření	$U_i$ [mV]	$U_i - U$ [mV]	$(U_i - U)^2 \cdot 10^{-3}$ [mV <sup>2</sup> ]
1	640,69	0,062	3,844
2	640,66	0,032	1,024
3	640,67	0,042	1,764
4	640,53	-0,098	9,604
5	640,58	-0,048	2,304
6	640,59	-0,038	1,444
7	640,69	0,062	3,844
8	640,58	-0,048	2,304
9	640,64	0,012	0,144
10	640,65	0,022	0,484
$\Sigma$	640,63	0,000	26,760

Mírou nejistoty jednotlivého měření je výběrová směrodatná odchylka  $s(U_i)$ . Tento odhad získáme dle vztahu:

$$s(U_i) = \sqrt{\frac{1}{n-1} \sum_{i=1}^n (U_i - \bar{U})^2} = \sqrt{\frac{1}{9} (26,760 \cdot 10^{-3})} = 0,055 \cdot 10^{-3} \text{ V} \quad (46)$$

Mírou nejistoty výsledku měření ze souboru naměřených hodnot je výběrová směrodatná odchylka průměru  $s(\bar{U})$ . Její odhad je dán vztahem:

$$s(\bar{U}) = \frac{s(U_i)}{\sqrt{n}} = \frac{0,055 \cdot 10^{-3}}{\sqrt{10}} = 0,017 \cdot 10^{-3} \text{ V} \quad (47)$$

Odhad výběrové směrodatné odchylky průměru je mírou nejistoty charakterizující náhodné chyby měření. Lze tedy říci, že standardní nejistota typu A je dána vztahem:

$$u_A(U) = s(\bar{U}) = 0,017 \cdot 10^{-3} V \quad (48)$$

**Výpočet standardní nejistoty typu B multimetru Agilent 34401A pro měření napětí:**

Průměrná hodnota naměřená na multimetru při měření napětí na měřicím rezistoru byla 640,63 mV. V katalogovém listu přístroje udává výrobce pro rozsah 1 V přesnost přístroje (0,0040% čtené hodnoty + 0,0007% rozsahu). Dále předpokládáme rovnoměrné (obdélníkové) rozložení pravděpodobnosti. Při znalosti těchto údajů můžeme nejistotu multimetru vypočítat dle vztahu:

$$\begin{aligned} u_B(U) &= \frac{p\% \text{ z čten. hodnoty} + p\% \text{ z rozsahu}}{\chi} = \\ &= \frac{0,0040 \cdot 640,628 \cdot 10^{-3} + 0,0007 \cdot 1}{\sqrt{3}} = 1,884 \cdot 10^{-3} V \end{aligned} \quad (49)$$

**Výpočet standardních nejistot typu B spojených s hodnotou odporu používaného měřicího rezistoru:**

Jedna z nejistot odporu měřeného rezistoru je spojena s vlivem teploty v místnosti. Víme, že teplota místnosti při měření elektrického proudu byla  $t = (23 \pm 1 \text{ } ^\circ\text{C})$ . Z technické dokumentace od výrobce víme, že teplotní koeficient daného rezistoru je 50 ppm. Vliv teploty na rezistor tedy vyjádříme pomocí rovnoměrného rozdělení pravděpodobnosti s pološířší  $a$ , kterou stanovíme jako součin teplotního koeficientu, maximální možné odchylky teploty v místnosti od referenční teploty a hodnoty elektrického odporu použitého měřicího rezistoru.

$$a = \alpha \cdot \Delta t \cdot R = 5 \cdot 10^{-5} \cdot 1,3 = 1,5 \cdot 10^{-4} \Omega \quad (50)$$

Odhad směrodatné odchylky rovnoměrného rozdělení pravděpodobnosti potom stanovíme dle vztahu:

$$s(R_T) = \frac{a}{\sqrt{3}} = \frac{1,5 \cdot 10^{-4}}{\sqrt{3}} = 8,660 \cdot 10^{-5} \Omega \quad (51)$$

Odhad této směrodatné odchylky bude vstupovat do vztahu pro výpočet kombinované standardní nejistoty a můžeme tedy psát:

$$u_B(R_T) = s(R_T) = 8,66 \cdot 10^{-5} \Omega \quad (52)$$

Další nejistotu týkající se odporu měřicího rezistoru určíme z přesnosti udávané výrobcem viz. tabulka 7-1. Tuto nejistotu stanovíme vynásobením přesnosti s hodnotou odporu a poté vydělením koeficientem pokrytí, který je roven 2, jelikož uvažujeme normální rozložení.

$$u_B(R) = \frac{\text{přesnost} \cdot R}{\chi} = \frac{0,013}{2} = 0,0065 \Omega \quad (53)$$

#### Výpočet kombinované standardní nejistoty:

K výpočtu kombinované standardní nejistoty vyjdeme z matematického modelu daného měření. V případě našeho měření je matematický model reprezentován vztahem:

$$I = f(x_1, x_2) = \frac{U}{R} \quad (54)$$

Pro výpočet kombinované nejistoty v případě nekorelovaných vstupních veličin využijeme vztahu (19). Pro námi řešený případ je výstupní veličinou proud  $I$ . Vstupními veličinami jsou napětí  $U$ , odpor  $R$  a funkcí je Ohmův zákon. Po dosazení těchto hodnot do vztahu (19) získáme vztah:

$$u_c(I)^2 = \left(\frac{dI}{dU}\right)^2 (u_A(U)^2 + u_B(U)^2) + \left(\frac{dI}{dR}\right)^2 (u_B(R)^2 + u_B(R_T)^2), \quad (55)$$

kde:

$$\frac{dI}{dU} = \frac{1}{R} = 0,333 \Omega^{-1}, \quad (56)$$

$$\frac{dI}{dR} = -\frac{\bar{U}}{R^2} = -0,0712 V\Omega^{-2}. \quad (57)$$

Po dosazení vztahů (56), (57) do vztahu (55) získáme vztah:

$$\begin{aligned}
 u_c(I)^2 &= \left(\frac{1}{R}\right)^2 \left(u_A(U)^2 + u_B(U)^2\right) + \left(-\frac{\bar{U}}{R^2}\right)^2 \left(u_B(R)^2 + u_B(R_T)^2\right) = \\
 &= (0,333)^2 \left[(0,017 \cdot 10^{-3})^2 + (1,884 \cdot 10^{-3})^2\right] + \\
 &+ (-0,071)^2 \left[(0,015)^2 + (8,660 \cdot 10^{-5})^2\right] = 1,5 \cdot 10^{-6} \text{ A}
 \end{aligned}
 \tag{58}$$

Výsledná kombinovaná standardní nejistota je poté:

$$u_c(I) = \sqrt{1,5 \cdot 10^{-6}} = 1,3 \cdot 10^{-3} \text{ A}
 \tag{59}$$

#### Výpočet rozšířené nejistoty měření:

Jelikož požadujeme interval, kde se žádaná hodnota nachází s 95% pravděpodobností výskytu musí se celková nejistota  $u_c(I)$  rozšířit pomocí koeficientu  $k$ . Pokud zachováme normální rozložení koeficient  $k$  bude nabývat hodnoty 2. Rozšířená nejistota se poté vypočte dle vztahu:

$$U(I) = k \cdot u_c(I) = 2 \cdot 1,3 \cdot 10^{-3} = 2,6 \cdot 10^{-3} \text{ A}
 \tag{60}$$

### 7.3 Výsledek měření

Výsledná hodnota stejnosměrného elektrického proudu protékajícího měřicím rezistorem byla  $I = (213,5 \pm 2,6) \text{ mA}$ .

Tabulka 7-5 Souhrn všech nejistot vyskytujících se ve výpočtu proudu tekoucího rezistorem.

Zdroj nejistoty	Typ	Hodnota nejistoty	Koef. citlivosti	Rozdělení	Koef. krytí	Standardní nejistota
Náhodné chyby	A	$0,017 \cdot 10^{-3} \text{ V}$	0,333	Normální	1,00	$5,7 \cdot 10^{-6}$
Voltmetr	B	$3,3 \cdot 10^{-3} \text{ V}$	0,333	Rovnoměrné	$\sqrt{3}$	$6,3 \cdot 10^{-4}$
Rezistor odpor	B	$0,03 \Omega$	0,0712	Normální	2,00	$1,1 \cdot 10^{-3}$
Rezistor Vliv teploty	B	$1,5 \cdot 10^{-4} \Omega$	0,0712	Rovnoměrné	$\sqrt{3}$	$6,2 \cdot 10^{-6}$
Proud	Kombinovaná	-	-	Normální	-	$1,3 \cdot 10^{-3}$
Proud	Rozšířená	-	-	Normální	2,00	$2,6 \cdot 10^{-3}$

## 8 VÝPOČET NEJISTOTY NEPŘÍMÉHO MĚŘENÉHO PROUDU METODOU MONTE CARLO

Pro simulace metody Monte Carlo je využito programů Matlab 2010b, LabVIEW 2013, GNU Octave. V těchto programech budou generátory pseudonáhodných čísel generovat jednotlivé hodnoty nejistot působících na stejnosměrný proud protékající měřicím rezistorem. Pro každou nejistotu bude v každém softwaru generováno  $10^6$  hodnot. Důvodem je, jak již bylo zmíněno v podkapitole 6.1, co největší podobnost rozdělení pravděpodobnosti u jednotlivých nejistot ke skutečnému tvaru. Matematický model vychází z Ohmova zákona a je uveden ve vztahu (61) a (62).

$$Y = f(X) = \frac{X_1}{X_2}, \quad (61)$$

kde  $X_1$  a  $X_2$  jsou vstupní hodnoty matematického modelu a  $Y$  je výstup matematického modelu. Po dosazení konkrétních veličin vstupujících do modelu je získán vztah:

$$I = \frac{U}{R}, \quad (62)$$

kde  $I$  [A] je stejnosměrný proud protékající měřicím rezistorem,  $U$  [V] je napětí na měřicím rezistoru a  $R$  je odpor měřicího rezistoru [ $\Omega$ ]. Na vztah (62) působí následující zdroje nejistot:

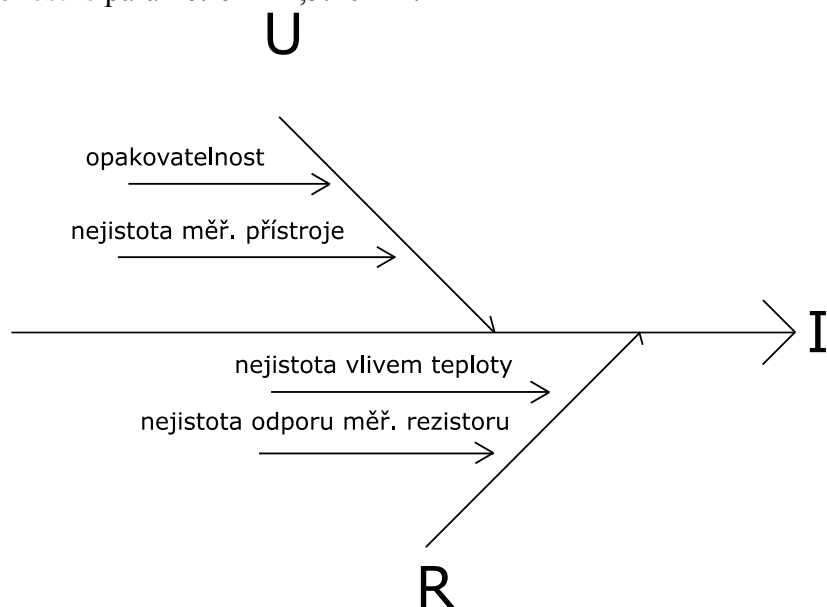
**Napětí ( $U$ ).** V realizovaném experimentu bylo napětí  $U$  opakovaně měřeno viz. tabulka 7-3. Průměrná hodnota byla vypočtena dle vztahu (44) a byla  $640,63 \cdot 10^{-3}$  V s výběrovou směrodatnou odchylkou měření  $0,017 \cdot 10^{-3}$  V. Tento zdroj nejistoty je čistě statistický a je klasifikován jako zdroj nejistoty typu A. Rozdělení pravděpodobnosti, které nejlépe specifikuje tento případ je normální rozdělení pravděpodobnosti s průměrem  $640,63 \cdot 10^{-3}$  V a výběrovou směrodatnou odchylkou průměru vypočtenou dle vztahu (47), která je rovna  $0,017 \cdot 10^{-3}$  V.

Dalším zdrojem nejistoty působící na měřené napětí je digitální měřicí multimetr. Zdroj nejistoty lze klasifikovat jako zdroj typu B. Pravděpodobnostní rozdělení zdroje je rovnoměrné s mezní chybou přístroje  $\pm 3,263 \cdot 10^{-3}$  V.

**Odpor ( $R$ ).** Na odpor měřicího rezistoru působí dva zdroje nejistot. Prvním zdrojem je nejistota spojená s hodnotou odporu použitého měřicího rezistoru. Jedná se o zdroj

typu B s normálním rozdělením pravděpodobnosti, s hodnotou odporu  $3 \Omega$  a standardní odchylkou  $0,015 \Omega$ .

Druhým zdrojem nejistoty je nejistota spojená s vlivem teploty místnosti na měření. Zdroj lze klasifikovat jako zdroj nejistoty typu B s normálním rozdělením pravděpodobnosti s parametrem  $\pm 1,5 \cdot 10^{-4} \Omega$ .



Obrázek 5 Schéma navrženého matematického modelu se zdroji nejistot.

Tabulka 8-1 Soupis zdrojů nejistot působící na matematický model.

Vstupní zdroje	Typ	Rozdělení	Parametry rozdělení
Napětí (U)			
- opakovatelnost měření	A	Normální	Průměrná hodnota: $640,63 \cdot 10^{-3} \text{ V}$ Standardní odchylka: $0,017 \cdot 10^{-3} \text{ V}$
- nejistota měř. přístroje	B	Rovnoměrné	Průměrná hodnota: $0 \text{ V}$ Min: $-3,263 \cdot 10^{-3} \text{ V}$ , Max: $+3,263 \cdot 10^{-3} \text{ V}$
Odpor (R)			
- nejistota vlivem teploty	B	Rovnoměrné	Průměrná hodnota: $0 \Omega$ Min: $-1,5 \cdot 10^{-4} \Omega$ , Max: $+1,5 \cdot 10^{-4} \Omega$
- nejistota odporu měř. rezistoru	B	Normální	Průměrná hodnota: $3 \Omega$ Standardní odchylka: $0,015 \Omega$

U dvou zdrojů nejistot uvedených v tabulce 8-1 je uvedeno, že průměrná hodnota je nula. Jedná se o matematický trik. Důvodem je působení více zdrojů nejistot na stejnou vstupní veličinu. Z tohoto důvodu dojde v průběhu simulace ke sloučení zdrojů nejistot působících na totožnou vstupní veličinu, a proto stačí, aby byla průměrná hodnota uvedena u jednoho zdroje nejistoty. Více informací je možné nalézt ve zdroji [11].

V následujících podkapitolách jsou popsány simulace výpočtu nejistot měření pomocí metody Monte Carlo v jednotlivých programech s tím, že výsledný histogram je vždy vytvořen v programu Matlab pro lepší porovnání výsledných dat.

## 8.1 Výpočet nejistot měření metodou Monte Carlo v programu Matlab 2010b

V programu Matlab je třeba nejprve vygenerovat náhodné hodnoty napětí a proudu v daném rozdělení, jejichž hodnoty jsou uvedeny v tabulce 8-1. Toto generování se provede podle následujících funkcí:

Vygenerování náhodných hodnot napětí se zdroji nejistot uvedených v tabulce 8-1 je provedeno dle následujícího kódu:

```
%parametry pro generování náhodné hodnoty napětí
M = 1000000;
U = 640.63e-3;
deltaUA = 0.017e-3;
deltaUB = 3.263e-3;

%vygenerování náhodných hodnot pro zdroj nejistoty A
randUA = U + deltaUA .* randn(1,M);
%vygenerování náhodných hodnot pro zdroj nejistoty B
randUB = 0 - deltaUB + (2 * deltaUB) .* rand(1,M);
%vytvoření výsledné náhodné hodnoty napětí
randUV = randUA + randUB;
```

Vygenerování náhodných hodnot pro odpor měřicího rezistoru včetně zdrojů nejistot je provedeno obdobně jako u náhodných hodnot pro napětí:

```
%parametry pro generování náhodné hodnoty odporu
M = 1000000;
R = 3;
deltaR = 0.015;
deltaRt = 1.5e-4;
```

```

%vygenerování náhodných hodnot pro nejistotu odporu re-
zistoru
randR = R + deltaR .* randn(1,M);
%vygenerování náhodných hodnot pro nejistotu teploty od-
poru
randRt = 0 + deltaRt .* randn(1,M);
%vytvoření výsledné náhodné hodnoty odporu
RandRV = randR + randRt;

```

Takto vygenerované hodnoty jsou poté dosazeny do matematického modelu (63) a seřazeny do neklesajícího pořadí z důvodu získání diskrétní distribuční funkce  $G$  viz. kapitola 3.2.:

```

%matematický model
randI = randUV / randRV;
%seřazení do neklesajícího pořadí
randI = sort(randI);

```

Dále je třeba vypočítat průměrnou hodnotu proudu, přidruženou standardní nejistotu a interval pokrytí [ $I_{\text{spodní}}$   $I_{\text{horní}}$ ]:

```

%průměrná hodnota proudu
prumI = mean(randI);

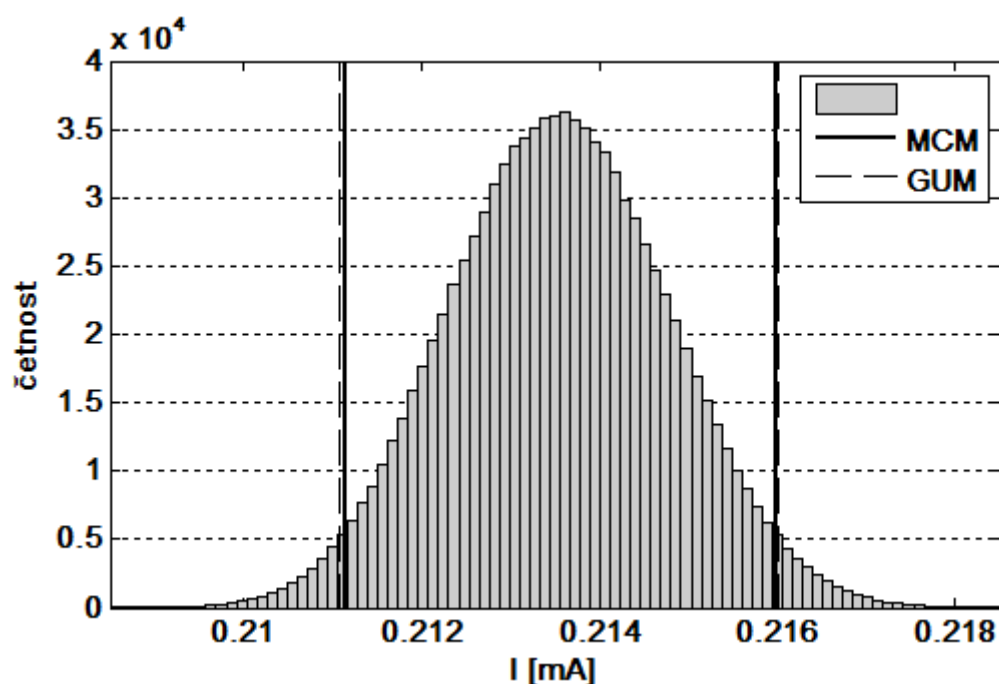
%výpočet směrodatné odchylky
sumaI = 0;
for i = 1:M
sumaI = sumaI + (randI(i) - prumI)^2;
end;
uR = sqrt((1/(M-1) * sumaI));

%výpočet intervalu pokrytí
P = 0.95;
q = p * M;
r = (M - q) / 2;
I_spodni = randI(r);
I_horni = randI(r+q);

```

Celý zdrojový kód pro simulaci je uveden v příloze č. 1. V následující tabulce 8-2 jsou uvedeny výsledky simulace. Je zde uvedena průměrná hodnota proudu, Směrodatná odchylka a vyčíslen horní a spodní interval v kterém by měřená veličina měla ležet

s pravděpodobnosti 95 %. Pokud provedeme srovnání získaných hodnot simulací metodou Monte Carlo v programu Matlab R2010b a výpočtu hodnot klasickou numerickou metodou uvedených v tabulce 7-5, dojdeme k závěru, že hodnoty jsou totožné. Jediný rozdíl nastává u intervalů pokrytí. Interval pokrytí vypočtený pomocí simulace vychází užší, než interval pokrytí u klasické numerické metody. Rozdíl je patrný na obrázku 6, kde je uvedený výsledný histogram proudu získaný pomocí simulace. V tomto histogramu jsou pro porovnání vyneseny intervaly pokrytí získané jak z výpočtu numerickou metodou (čerchovaně), tak i z výpočtu pomocí simulace metodou Monte Carlo (plná čára).



Obrázek 6 Výsledný histogram proudu  $I$  vypočtený programem Matlab 2010b.

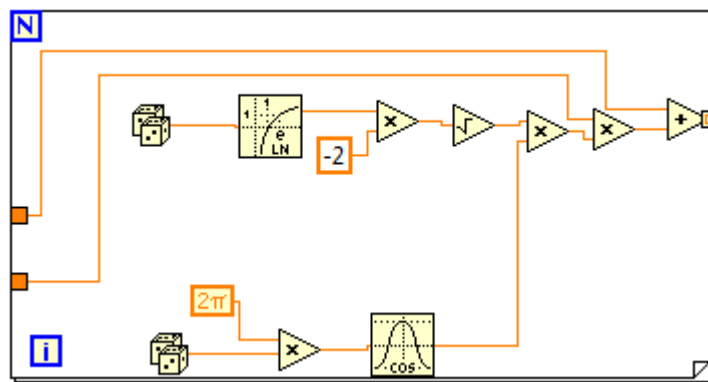
Tabulka 8-2 Statistické parametry získané simulací metodou MC v Matlabu.

Parametr	Hodnota
Průměrná hodnota	$213,5 \cdot 10^{-3}$ A
Směrodatná odchylka	$1,3 \cdot 10^{-3}$ A
Spodní koncový bod pro 95%	$211,14 \cdot 10^{-3}$ A
Horní koncový bod pro 95%	$215,98 \cdot 10^{-3}$ A

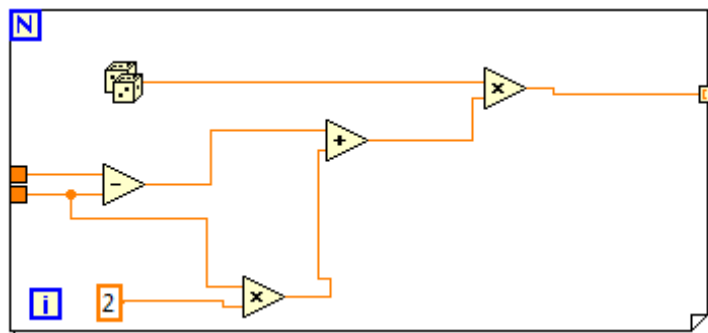
## 8.2 Výpočet nejistot měření metodou Monte Carlo v programu LabVIEW 2013

Výpočet nejistot programem LabVIEW je obdobný jako programem Matlab. Hlavním rozdílem je psaní zdrojového kódu. Zatímco v Matlabu se kód skládá s klasických psaných funkcí a příkazů v LabVIEW je tento kód „kreslen“. Simulace se obdobně jako u ostatních programů skládá z několika hlavních kroků.

Prvním krokem je vygenerování  $10^6$  náhodných hodnot pro měřené napětí a odpor rezistoru. Generování je provedeno za použití funkcí uvedených na obrázcích 7, 8.



Obrázek 7 Generátor náhodných čísel s normálním rozdělením v LabVIEW.



Obrázek 8 Generátor náhodných čísel s rovnoměrným rozdělením v LabVIEW.

Dalšími kroky je dosazení vygenerovaných hodnot do matematického modelu, seřazení vypočtených hodnot do neklesajícího pořadí, výpočet průměrné hodnoty, výpočet směrodatné odchylky a určení intervalů  $[I_{\text{spodní}} I_{\text{horní}}]$  v kterých s pravděpodobností 95% leží správná hodnota proudu. Posledním krokem je uložení výsledných dat do textového souboru pro vykreslení výsledného histogramu v programu Matlab. Důvodem je, jak již bylo zmíněno, standardizace histogramu pro porovnání dosažených výsledků v jednotlivých programech.

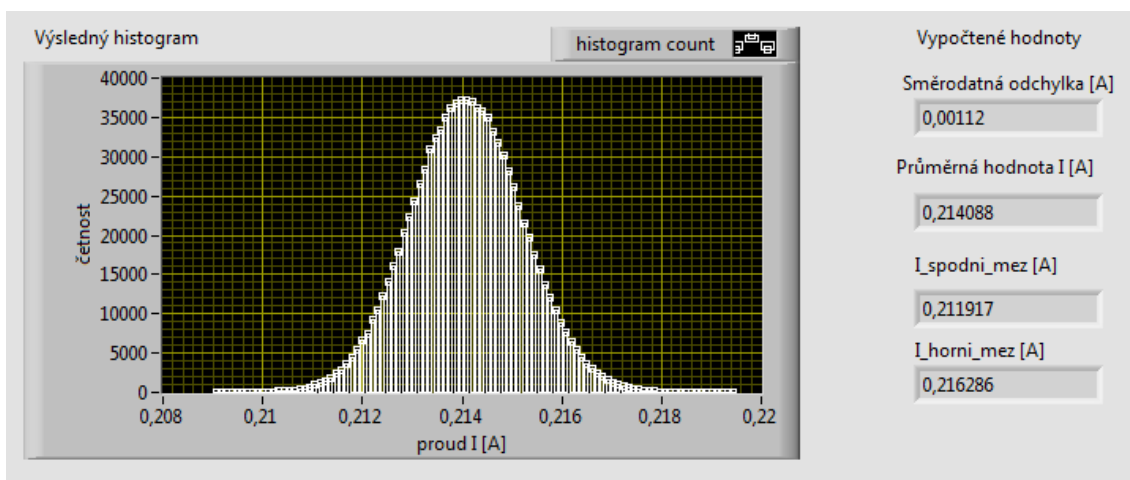
Čelní panel je uzpůsoben tak, že v horní části panelu uživatel zadá hodnoty jednotlivých veličin důležitých pro vygenerování náhodných hodnot. Dále musí zadat počet opakování, které stanoví kolik hodnot bude pro každou veličinu generováno. Posledním důležitým parametrem pro zadání je interval pokrytí. Do něho uživatel zadá jaký interval pokrytí požaduje. V případě výsledků této simulace byl použit interval pokrytí 95%. Ukázka uživatelské části panelu je uvedena na obrázku 9.

The screenshot shows two main sections: 'Generování náhodných hodnot odporu rezistoru' and 'Generování náhodných hodnot měřeného napětí'. Each section contains several input fields with numerical values and up/down arrows for adjustment.

Generování náhodných hodnot odporu rezistoru		Generování náhodných hodnot měřeného napětí	
odpor R [ohm]	3	Počet opakování	1000000
delta R [ohm]	0,015	Interval pokrytí [%]	95
delta Rt [ohm]	0,00015	Průměrná hodnota napětí U [V]	0,64063
		delta UA [V]	1,7E-5
		delta UB [V]	0,003263

Obrázek 9 Ukázka čelního panelu (uživatelská část)

Ve spodní části čelního panelu je situován orientační graf. Na tomto grafu bude po provedení simulace zobrazen výsledný histogram proudu. Vedle tohoto histogramu jsou poté zobrazeny vypočtené hodnoty jako jsou: průměrná hodnota proudu, směrodatná odchylka, spodní koncový interval a horní koncový interval. Ty hodnoty jsou třeba následně zaokrouhlit podle kapitoly 2.6.



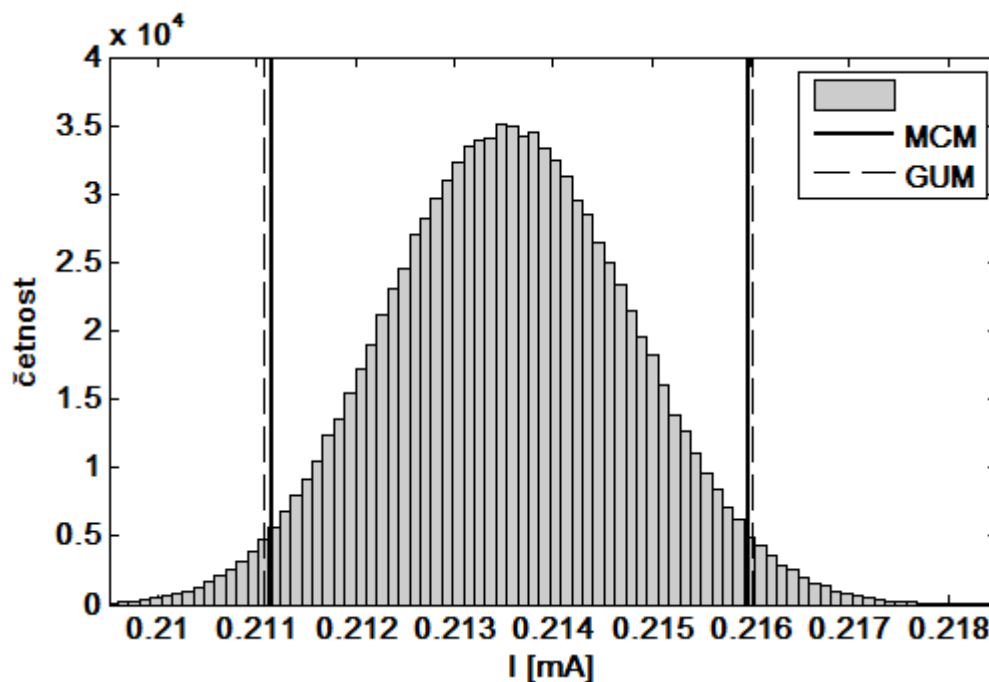
Obrázek 10 Ukázka čelního panelu (výsledková část)

Vypočtené hodnoty pomocí simulace jako jsou: průměrná hodnota proudu, směrodatná odchylka, spodní a horní koncový interval jsou uvedeny v tabulce 8-3. Z výsledků je možné pozorovat, že hodnoty jsou téměř totožné z výsledky dosaženými pomocí výpočtu klasickou numerickou metodou. Jediný rozdíl opět nastává u intervalu pokrytí, který v tomto případě vyšel užší. Výsledný histogram simulace je zobrazen na

obrázku 11. V histogramu jsou opět vyneseny meze nejistoty vypočtené jak klasickou numerickou metodou (čerkovaně), tak i pomocí simulace v programu LabVIEW (plná čára). Detailnější porovnání výsledků simulace bude provedeno v kapitole 9.

Tabulka 8-3 Statistické parametry získané metodou MC v LabVIEW.

Parametr	Hodnota
Průměrná hodnota	$214,1 \cdot 10^{-3} \text{ A}$
Směrodatná odchylka	$1,2 \cdot 10^{-3} \text{ A}$
Spodní koncový bod pro 95%	$211,91 \cdot 10^{-3} \text{ A}$
Horní koncový bod pro 95%	$216,28 \cdot 10^{-3} \text{ A}$



Obrázek 11 Výsledný histogram proudu  $I$  vypočtený programem LabVIEW.

Celé schéma zapojení v programu LabVIEW je uvedeno v příloze na CD. V příloze č.2 je ukázka celého čelního panelu.

### 8.3 Výpočet nejistot měření metodou Monte Carlo v programu GNU Octave

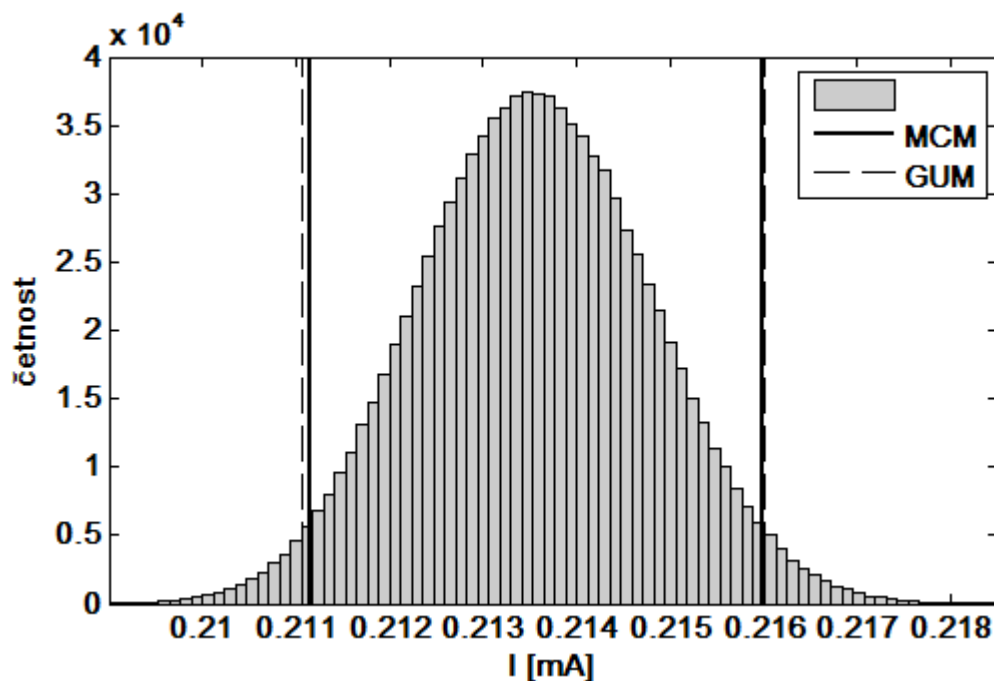
V programu GNU Octave je simulace provedena stejným způsobem jako u programu Matlab. Jelikož je software primárně určen pro operační systémy Linux je práce v něm

v operačním systému Windows poměrně obtížná. Základní vývojové prostředí je řešeno klasickým příkazovým řádkem. V tomto prostředí je následně možné si vytvořit m-file podobně jako u programu Matlab. V tomto m-file je poté napsán celý simulační skript, který je uveden v příloze č.3.

Výsledky simulace jsou uvedeny v tabulce 8-4. Při porovnání těchto výsledků s výsledky získanými pomocí klasické numerické metody je patrné, že výsledky jsou podle předpokladu kompatibility téměř totožné. Na obrázku 12 je uveden výsledný histogram proudu. Na histogramu jsou opět vyznačeny intervaly pokrytí pro klasickou numerickou metodu (čerchovaně) a pro výpočet metodou Monte Carlo (plná čára).

Tabulka 8-4 Statistické parametry získané metodou MC v GNU Octave.

Parametr	Hodnota
Průměrná hodnota	$213,5 \cdot 10^{-3}$ A
Směrodatná odchylka	$1,3 \cdot 10^{-3}$ A
Spodní koncový bod pro 95%	$211,15 \cdot 10^{-3}$ A
Horní koncový bod pro 95%	$215,98 \cdot 10^{-3}$ A



Obrázek 12 Výsledný histogram proudu  $I$  vypočtený programem GNU Octave.

## 9 POROVNÁNÍ DOSAŽENÝCH VÝSLEDKŮ VÝPOČTŮ NEJISTOT

V této kapitole budou popsány a zhodnoceny dosažené výsledky při stanovování nejistoty měření proudu protékajícího měřicím rezistorem. Budou zde porovnány jak výsledky výpočtů dosažených klasickou numerickou metodou stanovování nejistot měření, tak i výsledky dosažené pomocí metody Monte Carlo v různých programech.

### 9.1 Porovnání výpočtů proudu protékajícího měřicím rezistorem

Výpočet proudu protékajícího měřicím rezistorem byl proveden dvěma způsoby. První způsob byl u výpočtu nejistoty měření numerickým způsobem. Tento výpočet vycházel ze vztahů (44) a (45). Druhým způsobem výpočtu byl výpočet pomocí simulace metodou Monte Carlo v jednotlivých programech. U těchto simulací výpočet vycházel z matematického modelu (62). Dosažené výsledky jsou pro porovnání uvedeny v tabulce 9-1.

Tabulka 9-1 Vypočtené hodnoty proudu protékajícího měřicím rezistorem.

Proud $I$ vypočtený numerickou metodou	$213,5 \cdot 10^{-3}$ A
Proud $I$ vypočtený simulací MC v programu Matlab	$213,5 \cdot 10^{-3}$ A
Proud $I$ vypočtený simulací MC v programu LabVIEW	$214,1 \cdot 10^{-3}$ A
Proud $I$ vypočtený simulací MC v programu GNU Ocvate	$213,5 \cdot 10^{-3}$ A

Z tabulky 9-1 je patrné, že hodnoty proudu jsou téměř totožné. Jediný rozdíl nastal při výpočtu průměrné hodnoty proudu v programu LabVIEW. Důvodem rozdílné hodnoty je navržený generátor čísel s normálním rozdělením viz. obrázek 7, který způsobuje minimální odchylku.

### 9.2 Porovnání výpočtů směrodatné odchylky proudu

Hodnoty směrodatných odchylek proudu vypočtených pomocí metody Monte Carlo v jednotlivých programech jsou uvedeny v tabulce 9-2. V této tabulce je pro porovnání uvedena i hodnota výsledné kombinované standardní nejistoty, která byla vypočtena pomocí klasické numerické metody ze vztahu (59).

**Tabulka 9-2** Vypočtené hodnoty směrodatných odchylek proudu.

Výsledná kombinovaná nejistota $u_c(I)$	$1,3 \cdot 10^{-3}$ A
Směrodatná odchylka proudu vypočtená metodou MC v programu Matlab	$1,3 \cdot 10^{-3}$ A
Směrodatná odchylka proudu vypočtená metodou MC v programu LabVIEW	$1,2 \cdot 10^{-3}$ A
Směrodatná odchylka proudu vypočtená metodou MC v programu Octave	$1,3 \cdot 10^{-3}$ A

Z tabulky 9-2 je patrné, že hodnoty směrodatných odchylek vypočtených pomocí metody Monte Carlo v jednotlivých softwarech jsou téměř totožné s výslednou kombinovanou nejistotou měření, která byla vypočtena klasickou numerickou metodou stanovení nejistot měření.

### 9.3 Porovnání výpočtů intervalů pokrytí u jednotlivých metod výpočtu

V tabulce 9-3 jsou porovnány 95% intervaly pokrytí, jež byly vypočteny jak klasickou metodou, tak i metodou Monte Carlo v jednotlivých simulačních programech.

**Tabulka 9-3** 95% intervaly pokrytí vypočteny oběma metodami.

Metoda výpočtu	Spodní interval $I_{spodní}$	Horní interval $I_{horní}$
Numerická metoda	$211,07 \cdot 10^{-3}$ A	$216,02 \cdot 10^{-3}$ A
Metoda MC v Matlabu	$211,14 \cdot 10^{-3}$ A	$215,98 \cdot 10^{-3}$ A
Metoda MC v LabVIEW	$211,93 \cdot 10^{-3}$ A	$216,26 \cdot 10^{-3}$ A
Metoda MC v GNU Octave	$211,15 \cdot 10^{-3}$ A	$215,98 \cdot 10^{-3}$ A

Intervaly pokrytí vypočtené metodou Monte Carlo vyšly užší než intervaly vypočtené numerickou metodou. Tyto intervaly se vždy překrývají, z čehož vyplývá, že metoda Monte Carlo i numerická metoda jsou kompatibilní.

## 9.4 Porovnání výsledků z hlediska náročnosti jednotlivých metod a programů

Z hlediska náročnosti výpočtů se jeví jako nejobtížnější klasická numerická metoda. Hlavním důvodem je výpočet parciálních derivací, jednotlivých citlivostních koeficientů, které při výpočtu metodou Monte Carlo nejsou třeba. Další nevýhodou oproti metodě Monte Carlo je časová náročnost. Ta je i u jednodušších výpočtů několikanásobně vyšší. Naproti tomu značnou výhodou numerické metody je možnost výpočtu bez použití výpočetní techniky.

Výpočet nejistot metodou Monte Carlo je vhodné použít v případech, kdy měřené veličiny mají různá pravděpodobnostní rozložení. Výhodou této metody je především rychlost a jednoduchost výpočtů, které provádí výpočetní program. Důležitou podmínkou pro úspěšné provedení simulace je mít dostatečně velkou výpočetní kapacitu, jelikož metoda pracuje s velkým počtem generovaných dat. V této práci bylo k výpočtu nejistot pomocí metody Monte Carlo využito programů Matlab, LabVIEW, GNU Octave. Z hlediska výsledků simulací se všechny programy jeví totožně. Při programování simulace méně složitých výpočtů je nejnadnější využít programů Matlab a GNU Octave. V těchto softwarech se celý zdrojový kód zapíše do souboru *m-file*. Nevýhodou jsou ovšem případné modifikace zdrojového kódu u složitějších měření, jelikož v případě špatného stanovení simulačních parametrů je třeba upravovat zdrojový kód, který může být velmi rozsáhlý. Velkou výhodou je u těchto dvou softwarů jejich vzájemná kompatibilita. Zdrojový kód naprogramovaný v prostředí Matlab je možné přenést do prostředí GNU Octave a naopak. Této kompatibility je vhodné využít pokud uživatel nemá zakoupenou licenci programu Matlab, ale vlastní pouze GNU Octave, který má volně přístupnou licenci. Výpočet nejistot pomocí metody Monte Carlo v programu LabVIEW je vůči dvěma výše zmiňovaným programům časově náročnější. Důvodem je především způsob programování. Zde není zdrojový kód primárně psán, ale kreslen, což může být v začátcích problematické. Oproti tomu je zde výhodou snadná změna parametrů simulace. Tyto parametry má uživatel možnost měnit na čelním panelu, aniž by zasahoval do primárního zdrojového kódu.

## 10 ZÁVĚR

V této diplomové práci jsem se zabýval stanovováním nejistoty nepřímého měření proudu pomocí metody Monte Carlo. Cílem práce bylo nejprve ověřit generátory pseudonáhodných čísel obsažených ve vybraných softwarech, ve kterých je možné provádět simulace pomocí metody Monte Carlo. Dále navrhnout metodiku výpočtu nejistoty nepřímého měření proudu pomocí klasické numerické metody a metody Monte Carlo a následně tento výpočet realizovat ve vybraných softwarech. Dalším cílem bylo porovnat takto dosažené výsledky a zhodnotit výhody a nevýhody jednotlivých metod i jednotlivých softwarů.

V teoretické části práce se nachází stručný úvod do metod stanovování nejistot měření. Jsou zde popsány základní vztahy pro výpočty a základní dělení nejistot. Následující kapitola je zaměřena na popis metody Monte Carlo. V této kapitole je metoda stručně popsána, dále je zde nastíněn postup výpočtu nejistot za využití této metody. V závěru kapitoly jsou popsány výhody a nevýhody využití této metody. V poslední teoretické kapitole jsou popsány generátory náhodných a pseudonáhodných čísel. Je zde uveden stručný popis jak základních náhodných generátorů, tak i pseudonáhodných generátorů. Dále jsou zde uvedeny způsoby testování náhodnosti těchto generátorů se zaměřením na testovací baterii NIST od společnosti National Institute of Standards and Technology. U této baterie jsou následně popsány i vybrané testy, které tato baterie obsahuje.

V praktické části jsem se nejprve zabýval otestováním generátorů pseudonáhodných čísel obsažených ve vybraných softwarech umožňujících provedení simulací metodou Monte Carlo. Testoval jsem generátory v softwarech Matlab 2010b, LabVIEW 2013, GNU Octave, Microsoft Excel 2013 a Maple 17. Pro každý generátor jsem provedl osm nezávislých testů viz. kapitola 6. Po provedení těchto testů jsem došel k závěru, že všechny generátory jsou vyhovující jak z hlediska schopnosti generovat dostatečný počet čísel, tak i z hlediska náhodnosti, tudíž jsou vhodné pro provedení simulací metodou Monte Carlo. V druhé části se zabývám stanovením nejistoty nepřímého měření proudu protékajícího měřicím rezistorem. Tuto nejistotu nejprve stanovuji pomocí klasické numerické metody viz. kapitola 7, a poté pomocí simulací metodou Monte Carlo v softwarech Matlab R2010b, LabVIEW 2013, GNU Octave. Při výpočtu klasickou metodou vyšla tato nejistota  $I = 213,5 \pm 2,6A$ . Při výpočtu nejistoty pomocí simulace metodou Monte Carlo jsem dosáhl obdobných výsledků viz. srovnávací tabulky v kapitole 9.

Z dosažených výsledků je patrné, že intervaly pokrytí vypočtené metodou Monte Carlo jsou užší než intervaly pokrytí vypočtené numerickou metodou viz. tabulka 9-3. Z toho vyplývá, že výsledky dosažené simulací jsou přesnější než výsledky dosažené výpočtem pomocí numerické metody. Dále mohu říci, že výsledky vypočtené oběma metodami jsou kompatibilní, jelikož intervaly pokrytí se překrývají.

Po provedení všech výpočtů mohu konstatovat, že výpočet nejistot pomocí metody Monte Carlo je z hlediska rychlosti a jednoduchosti výhodnější. Důležitou podmínkou je však zvolení vhodného softwaru pro simulace. Z využitých softwarů se mi nejlépe

jevily Matlab a GNU Octave, díky jednoduchosti programování. U LabVIEW je nespornou výhodou snadná modifikace na různé modely a uživatelská přehlednost. Nevýhodou je však složitější způsob programování.

# 11 LITERATURA

- [1] PALENČÁR, R., VDOLEČEK, F., HALAJ, M.: *Nejistoty v měření I: Vyjadřování nejistot. AUTOMA*. 2001, 7, s. 50-54.
- [2] HRUŠKA, K., BRADÍK, J. *Stanovení nejistot při měření parametrů jakosti*. Brno: Vysoké učení technické, 2001. ISBN 80-214-1656-1.
- [3] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP and OIML, Evaluation of measurement data-Supplement 1 to the “*Guide to the expression of Uncertainty in Measurement*“ Propagation of distributions using a Monte Carlo method. 1. vyd. 2008.
- [4] BIPM, IEC, IFCC, ILAC, ISO, IUPAC, IUPAP and OIML, Evaluation of measurement data – *Guide to the expression of Uncertainty in Measurement*
- [5] PALENČÁR, R., VDOLEČEK, F., HALAJ, M.: *Nejistoty v měření III: nejistoty nepřímých měření. AUTOMA*. 2001, 12, s. 28-33.
- [6] PALENČÁR, R., VDOLEČEK, F., HALAJ, M.: *Nejistoty v měření II: nejistoty přímých měření. AUTOMA*. 2001, 10, s. 52-56.
- [7] ROSSLER, T., *Nejistoty měření*. [online]. [cit. 2015-04-21]. Dostupné z: <http://fyzika.upol.cz/cs/system/fines/download/vujtek/rcptm/nejistoty.pdf>.
- [8] SADÍLEK, V., DOLŽAL J., VOŘECHOVSKÝ M., *Řešené úlohy z oblasti spolehlivosti stavebních konstrukcí*. [online]. [cit. 2015-04-21]. Dostupné z: <http://www.fce.vutbr.cz/stm/sadilek.v/Frvs/Frvs.pdf>.
- [9] KLVAŇA, J. *Principy a aplikace metody Monte Carlo*. Praha: České vysoké učení technické, 2006. ISBN 80-01-03587-5.
- [10] FABIAN, F. *Metoda Monte Carlo a možnosti jejího uplatnění*. Praha: Prospektum, 1998. ISBN 80-7175-058-1.
- [11] GUIMARAES COUTO P., CARRETEIRO J., DE OLIVER S., Monte Carlo Simulations Applied to Uncertainty in Measurement. *Theory and Applications of Monte Carlo Simulations* [online]. InTech, 2013-03-06 [cit. 2015-04-21]. DOI: 10.5772/53014. Dostupné z: <http://www.intechopen.com/books/theory-and-applications-of-monte-carlo-simulations/monte-carlo-simulations-applied-to-uncertainty-in-measurement>.

- [12] ŠÍRA, M., NOVÁKOVÁ ZACHOVALOVÁ, V., ZŮDA, J. *Vybrané problémy metrologie fyzikálních a elektrických veličin* [online]. Beneš 2012-06-11 [cit. 2014-12-12]. Dostupné z: <http://crr.vutbr.cz/kurzy-seminare/merici-systemy/seminar-vybrane-problemy-metrologie-fyzikalnich-elektrickych-velicin>.
- [13] HRACH, R. *Počítačová fyzika I*. Ústí nad Labem: PF UJEP, 2003.
- [14] Random Number Generators [online] 2001-05-30 [cit. 2015-01-15]. Dostupné z: <https://www.cs.indiana.edu/~kapadia/project2/node6.html>.
- [15] XIANG, T., KHALED, B. *Mersenne Twister Random Number Generation on FPGA, CPU and GPU* [online]. 2009 [cit. 2015-01-15]. Dostupné z: [http://www.see.ed.ac.uk/~SLIg/papers/tian\\_AHS09.pdf](http://www.see.ed.ac.uk/~SLIg/papers/tian_AHS09.pdf).
- [16] The Marsaglia Random CDROM including the Diehard Battery of Tests of Randomness [online]. 1995 [cit. 2015-01-16]. Dostupné z: <http://www.stat.fsu.edu/pub/diehard>.
- [17] Computer Security Division [online]. 2014-06-16 [cit. 2015-01-16]. Dostupné z: [http://www.csrc.nist.gov/groups/ST/toolkit/rng/documentation\\_software.html](http://www.csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html).
- [18] RUKHIN, A., SOTO, J., NECHVATAL, J., SMID, M., BARKER, E., LEIGHT, S., LEVENSON, M., VANGEL, M., BANKS, D., HECKERT, A., DRAY, J., VO, S. *A statistical test suite for random and pseudorandom number generators for cryptographic applications* [online]. [cit. 2015-01-25]. Dostupné z: <http://www.csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22b.pdf>.
- [19] HERINGOVÁ, B., HORA, P.. *MATLAB: Díl I. – práce s programem* [online]. Plzeň, 1995 [cit. 2015-02-22]. Dostupné z: <http://www.cdm.cas.cz/czech/hora/vyuka/mvs/tutorial.pdf>.
- [20] HERINGOVÁ, B., HORA, P.. *MATLAB: Díl II. – funkce* [online]. Plzeň 1995 [cit. 2015-02-15]. Dostupné z: <http://www.cmd.cz/czech/hora/vyuka/mvs/reference.pdf>.
- [21] DOSTÁL, J. *MS EXCEL 2007 PRO UČITELE* [online]. Olomouc, 2009 [cit. 2015-02-22]. Dostupné z: <http://www.kteiv.upol.cz/frvs/ict-kubricky/inc/office/excel.pdf>.

- [22] KOLÁŘOVÁ, E. *Maple* [online]. Brno [cit. 2015-02-22]. Dostupné z: <http://www.umat.feec.vutbr.cz/~kolara/Maple.pdf>
- [23] NATIONAL INSTRUMENTS. *LabVIEW: User manual* [online]. 1992 [cit. 2015-02-24]. April 2003 Edition, 320999E-01. Dostupné z: <http://www.ni.com/pdf/manuals/320999e.pdf>.
- [24] EATON, J. GNU Octave. [online]. [cit. 2015-02-24]. Dostupné z: <https://www.gnu.org/software/octave/doc/interpreter/>.
- [25] AGILENT TECHNOLOGIES. *Agilent 34401A Multimeter: Data Sheet* [online]. [cit. 2015-04-22]. Dostupné z: <http://cp.literature.agilent.com/litweb/pdf/5968-0162EN.pdf>
- [26] STATRON A.W.V: *Standardní laboratorní zdroj* [online]. [cit. 2015-04-22]. Dostupné z: <http://storage.merici-opticke-pristroje.cz/contentresources/Technicka-data-zdroju-Statron-804-125.pdf>
- [27] *Openbugs* [online]. [cit. 2015-04-22]. Dostupné z: <http://www.openbugs.net/w/Downloads>
- [28] *Metrodata: Download* [online]. [cit. 2015-04-28]. Dostupné z: [http://www.metrodata.de/download\\_en.html](http://www.metrodata.de/download_en.html)
- [29] Matlab. 1994. *MathWorks* [online]. [cit. 2015-02-24]. Dostupné z: <http://www.mathworks.com/producta/matlab/>

## 12 SEZNAM PŘÍLOH

Příloha 1 - Simulační skript pro výpočet nejistot nepřímého měření proudu pomocí metody Monte Carlo v programu Matlab.....	73
Příloha 2 – Čelní panel pro výpočet nejistoty nepřímého měření proudu metodou Monte Carlo v programu LabVIEW .....	75
Příloha 3 – Simulační skript pro výpočet nejistot nepřímého měření proudu pomocí metody Monte Carlo v programu GNU Octave.....	76
Příloha 4 - Ukázka prostředí testovací baterie NIST.....	78
Příloha 5 - Příložené CD .....	79

## Příloha 1 - Simulační skript pro výpočet nejistot nepřímého měření proudu pomocí metody Monte Carlo v programu Matlab.

```
clear all;
clc;
format long;

%pocet opakovani
M=1000000;

%vygenerovani nahodnych hodnot pro odpor
R=3; %odpor rezistoru
deltaR = 0.015; %nejistota odporu rezistoru
deltaRt=1.5e-4; %nejistota odporu rezistoru spojená s teplotou

randR=R+deltaR.*randn(1,M); %vygenerovani nah. hodnot s normalnim rozdelenim
randRt=0-deltaRt+(2*deltaRt).*rand(1,M); %vygenerovani nah. hodnot s rovnomernym rozdelenim
randRV=randR+randRt; %vypocet vyslednych nah. hodnot pro odpor rezistoru

%vygenerovani nahodnych hodnot pro napeti
U=640.63e-3; %prumerna hodnota napeti
deltaUA = 0.017e-3;
deltaUB = 3.263e-3;
randUA=U+deltaUA.*randn(1,M); %vygenerovani nah. hodnot s normalnim rozdelenim
randUB = 0 - deltaUB + (2 * deltaUB) .* rand(1,M); %vygenerovani nah.hodnot s rov. Rozdělením
randUV=randUA+randUB; %vypocet vyslednych nah. hodnot pro napeti

%vypocet proudu
randI=randUV./randRV;
```

```

%prumerna hodnota prudu
prumI = mean(randI)

%vykrelsleni histogramu pravdepodobnosti
hist(randI,100)
grid on
hold on

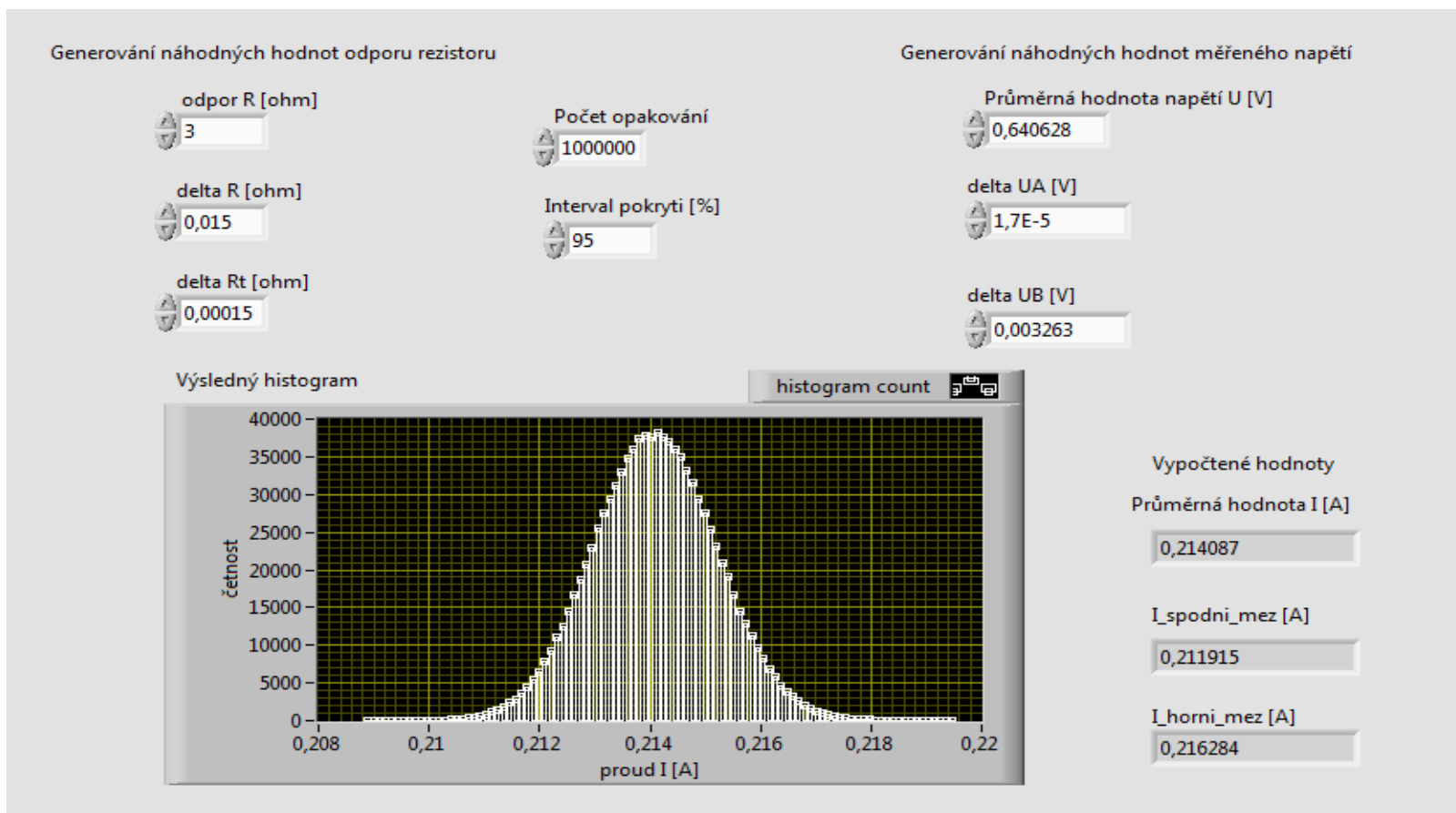
% serazeni hodnot do neklesajiciho poradi
randI = sort(randI);

%vypocet smerodatne odchyvky
sumaI = 0;
for i=1:M
sumaI = sumaI + (randI(i)-prumI)^2;
end;
uI = sqrt((1/(M-1)*sumaI))

%vypocet intervalu pokryti 95% pro I
p = 0.95;
q = p*M;
r = (M-q)/2;
f=r+q
Ilow = randI(r)
Ihigh = randI(r+q)
%vykresleni svislych car pro porovnaní metod
line([Ilow Ilow],[0 4e4],'Color','k','LineWidth',2)
hold on
line([213.543e-3+2.47e-3 213.543e-3+2.47e-3],[0 4e4],'Color','k','LineStyle','--','LineWidth',1)
legend(' ','MCM','GUM')
hold on
line([Ihigh Ihigh],[0 4e4],'Color','k','LineWidth',2)
hold on
line([213.543e-3-2.47e-3 213.543e-3-2.47e-3],[0 4e4],'Color','k','LineStyle','--','LineWidth',1)

```

## Příloha 2 – Čelní panel pro výpočet nejistoty nepřímého měření proudu metodou Monte Carlo v programu LabVIEW



## Příloha 3 – Simulační skript pro výpočet nejistot nepřímého měření proudu pomocí metody Monte Carlo v programu GNU Octave

```
## Copyright (C) 2015 Marek
## Author: Marek Novotny<Marek@B05-429>
## Created: 2015-04-27
%pocet opakovani
M=1000000;
%vygenerovani nahodnych hodnot pro odpor
R=3; %odpor rezistoru
deltaR= 0.015; %nejistota odporu rezistoru
deltaRt = 1.5e-4; %nejistota odporu rezistoru spojená s teplotou
randR=R+deltaR.*randn(1,M); %vygenerovani nah. hodnot s normalnim rozdelenim
randRt = 0 - deltaRt+(2*deltaRt).*rand(1,M); %vygenerovani nah. hodnot s rov. rozdelenim
randRV = randR+randRt; %vypocet vyslednych nah. hodnot pro odpor rezistoru

%vygenerovani nahodnych hodnot pro napeti
U=640.63e-3; %prumerna hodnota napeti
deltaUA = 0.017e-3; %nejistota typu A u mereneho napeti
deltaUM = 3.263e-3; % nejistota spojená s mer. pristrojem
randUA = U+deltaUA.*randn(1,M); %vygenerovani nah. hodnot s norm. rozdelenim
randUM = 0-deltaUM+(2*deltaUM).*rand(1,M); %vygenerovani nah. hodnot s rov. rozdelenim
```

```
randUV=randUA+randUM;
```

```
%vypocet vyslednych nah. hodnot pro merene napeti
```

```
%vypocet proudu
```

```
randI=randUV./randRV;
```

```
%prumerna hodnota proudu
```

```
prumI = mean(randI)
```

```
%serazeni do neklesajícího pořadí
```

```
randI = sort(randI);
```

```
%vypocet smerodatne odchylky
```

```
sumaI = 0;
```

```
for i=1:M
```

```
sumaI=sumaI+(randI(i)-prumI)^2;
```

```
end;
```

```
uI = sqrt((1/(M-1))*sumaI)
```

```
%vypocet intervalu pokryti pro I
```

```
p=0.95;
```

```
q=p*M;
```

```
r=(M-q)/2;
```

```
Ilow=randI(r)
```

```
Ihigh=randI(r+q)
```

```
%vykresleni vysledneho histogramu
```

```
hist(randI,100)
```

## Příloha 4 - Ukázka prostředí testovací baterie NIST

The image shows a screenshot of the NIST test battery configuration interface. The main window has a file path at the top: `C:\Users\Marek\Desktop\diplomka\testovani generatoru\data_matlab_1000\data_Matlab_1.bt`. Below this, there is a list of tests with checkboxes and numerical parameters. The tests and their parameters are:

Test Name	Checked	n	M	m	Other
Frequency Monobit	<input checked="" type="checkbox"/>	10000			
Frequency Test Within a Block	<input checked="" type="checkbox"/>	10000	128		
Runs	<input type="checkbox"/>	1000000			
Longest Run of Ones in a Block	<input type="checkbox"/>	1000000			
Binary Matrix Rank	<input checked="" type="checkbox"/>	10000			
Discrete Fourier Transform (Spectral)	<input checked="" type="checkbox"/>	10000			
Non Overlapping Template Matching	<input type="checkbox"/>	1000000			Single B 111111111
Overlapping Template Matching	<input type="checkbox"/>	1000000			B 111111111
Universal	<input type="checkbox"/>	1000000			Custom
Linear Complexity	<input checked="" type="checkbox"/>	10000	500		
Serial	<input checked="" type="checkbox"/>	10000		5	
Approximate Entropy	<input type="checkbox"/>	1000000		10	
Cumulative Sums	<input type="checkbox"/>	1000000			Forward
Random Excursions	<input type="checkbox"/>	1000000			
Random Excursions Variant	<input type="checkbox"/>	1000000			

An inset window titled "Reports Form" displays the results of the tests. The report content is as follows:

```
FREQUENCY TEST
-----
COMPUTATIONAL INFORMATION:
-----
(a) The nth partial sum = 6
(b) S_n/n              = 0,0006
-----
SUCCESS                                     p_value = 0,952155634691786
```

## **Příloha 5 - Příložené CD**

Na přiloženém CD se nachází kompletní text diplomové práce ve formátu .pdf. Dále se na CD nachází zdrojové kódy ke všem simulacím provedeným v této práci a ukázkový soubor vytvořených dat pro testování generátorů baterií NIST.