Informační Bulletin

České Statistické Společnosti číslo 4, ročník 14.

Zajímavý generátor náhodných čísel

Petr Máša [1]

1. Úvod

Tento příspěvek se zabývá otázkou generování a zejména testování náhodných čísel. Standardní testové balíky (sady testů) se snaží odhalit některé systematické chyby ve výstupní posloupnosti daného zdroje neurčitosti. Pokud se u uznávané sady testů stane, že generovaná posloupnost „projde“ všemi testy (tedy jeví se, jako by opravdu byla náhodná), považujeme daný generátor za dobrý. My si ukážeme, že existuje velmi primitivní deterministický generátor, který těmito testy „hladce projde“, a přesto obsahuje řadu systematických chyb.

2. Náhodná a pseudonáhodná posloupnost

Přistupme nejprve k základním definicím. Posloupnosti nezávislých náhodných veličin $\{X_i\}_{i=1}^n$, které mají alternativní rozdělení s parametrem $p = 0,5$, budeme říkat náhodná posloupnost nul a jedniček, či krátce jen náhodná posloupnost. Úsek této posloupnosti, který má pevnou délku, budeme chápat jako náhodné číslo (v binární podobě). Generátor, který tuto posloupnost vytváří, budeme nazývat generátorem náhodných čísel (RNG – random number generator).

Všimněme si, že definice náhodné posloupnosti říká (stejně jako základy pravděpodobnosti) spíše něco o tom, jak náhodná posloupnost vzniká. Už se moc nemluví o tom, jakým způsobem tuto posloupnost vygenerovat v praxi (jak zajistit například nezávislost daných veličin při generování) a už vůbec ne o tom, jak poznat, zda je tato posloupnost náhodná.

[1] petrmasa@yahoo.com

Podívejme se nyní na generování „náhodných“ posloupností v praxi. Jednou z možností je házení mincí. To je však zdlouhavé a nepraktické, potřebujeme-li náhodných čísel větší počet. S rozvojem výpočetní techniky se začaly rozvíjet generátory *pseudonáhodných* čísel. To jsou generátory, které generují čísla deterministicky (nějakou funkcí), obvykle na základě několika přechozích hodnot.

Podívejme se na možnost definování pseudonáhodných čísel ([4]). Generátor pseudonáhodných čísel je deterministické zařízení, které pro daný (skutečně náhodný) vstup délky k vrátí posloupnost délky $l \gg k$, která „vypadá“ náhodně. Vstup pseudonáhodného generátoru se nazývá semínko (hnízdo, seed) a výstup se nazývá pseudonáhodná posloupnost.

Pro účel testování, zdali je daná posloupnost „náhodná“, existují statistické testy (ve smyslu statistické indukce). My potřebujeme otestovat dva základní požadavky - parametr alternativního rozdělení a nezávislost. Právě na nezávislost neexistují pro praxi univerzálně použitelné testy. Existující testy náhodných posloupností si všímají vždy jen určité charakteristiky. Například můžeme zjišťovat, zdali počet nul a jedniček je přibližně stejný (ve statistickém smyslu), zdali se dvojice symbolů vyskytují se stejnou pravděpodobností, zdali v dané posloupnosti neexistují tzv. dlouhé běhy (dlouhé úseky složené jen z nul či jen z jedniček) apod.

Při reálném použití se často stává, že mezi posloupností generovanou pseudonáhodným generátorem a skutečně náhodným generátorem nejsme často schopni v rozumném čase[2] rozlišovat ([4] definuje takovou třídu (vlastnost) generátorů).

Obecně tedy platí, že nejsme schopni rozpoznat, zdali je daná posloupnost bitů (čísel) generována nějakou (deterministickou) funkcí. Dá se dokonce dokázat, že tento problém (určení funkce, kterou byla posloupnost generována) je daleko obtížnější než například rozluštění současných šifrovacích algoritmů (tyto algoritmy se též používají pro generování náhodných posloupností, dokonce tzv. kryptograficky bezpečných náhodných posloupností). Tedy prakticky jedinou možností, jak testovat, zdali je daná posloupnost náhodná, je vzít (existující) sadu testů a ty aplikovat na zkoumanou posloupnost. Každý z těchto testů zkoumá určitý druh systematické chyby generátoru (např. zda generuje nějakou speciální posloupnost častěji než je její očekávaná relativní četnost apod.). Podle výsledků testů této sady se usuzuje na kvalitu generátoru. Naznačený princip je ovšem „důkaz nenalezením protipříkladu“, navíc se výsledky ověřují pomocí statistické indukce (testování statistických hypotéz).

Ukážeme si, že existuje generátor pseudonáhodných čísel (je to generátor implementovaný v Borland Pascalu 7), který má velmi jednoduchou strukturu a při použití „slavné“ sady testů DIEHARD [1] se jeví jako velmi kvalitní.

[2] ve smyslu teorie složitosti v polynomiálním čase

3. Struktury generátorů

Mezi nejjednodušší generátory pseudonáhodných čísel patří lineární kongruenční generátory. Ty jsou založeny na myšlence, že máme nějakou inicializační hodnotu X_0 (zvanou též *seed*, *hnízdo*, *semínko*) a další hodnoty jsou generovány funkcí tvaru

$$X_{n+1} = (a \cdot X_n + b) \mathrm{mod}\ c. \tag{1}$$

Často je pro jednoduchost implementace bráno $c = 2^d$ pro nějaké přirozené číslo d. Důvodem je snadná a velice rychlá implementace (většinou implicitně díky omezení datového typu, např. pro 16-bitové číslo jsou operace automaticky „modulo 2^{16}“). Algoritmicky bychom mohli psát

1. Seed $\leftarrow$ f(Seed)
2. Output(Seed).

Algoritmus 1 : Základní schéma lineárního kongruenčního generátoru

Dá se ale dokázat, že pro $c = 2^d$ (kromě triviálních případů) platí, že pokud daný generátor dosahuje plné periody, pak se v dané posloupnosti střídají sudá a lichá čísla. Na tento problém též myslela sada testů DIEHARD, a tudíž obsahuje některé testy aplikované na specifický bit čísel generované posloupnosti. Dá se ukázat, že ještě několik dalších bitů (od nejméně významných) vykazuje statisticky špatné vlastnosti co se týče náhodnosti.

Příkladem klasického lineárního kongruenčního generátoru je *generátor VBA* – generátor pseudonáhodných čísel implementovaný v jazyce VBA (Visual Basic for Applications).

Struktura generátoru implementovaného v Borland Pascalu 7 (dále jen generátor BP) obsahuje mírně upravené základní schéma lineárního generátoru. Také odstraňuje vlastnost, že každé následující číslo je funkcí předchozího (ale je zachován determinismus generátoru, a tedy i opakovatelnost). Struktura je následující:

1. Seed $\leftarrow$ f(Seed)
2. Output(g(Seed)).

Algoritmus 2 : Rozšířené schéma lineárního kongruenčního generátoru

Funkce g je zobrazení, které není prosté a které zobrazuje jednu množinu na jinou, podstatně menší. Tedy na výstup se nedostává celá informace ze semínka generátoru a právě tím je zajištěno, že následující číslo není funkcí předchozího (jen následující semínko je funkcí předchozího semínka, ale ta se na výstup celá nedostávají).

Budeme-li konkrétní, semínko generátoru BP je 32-bitové číslo a výstupem generátoru je 16-bitové číslo. Funkce g vybírá z daného semínka horních 16 bitů (tedy těch „náhodnějších“ 16 bitů). Tímto postupem je také snížena možnost predikovatelnosti dalšího čísla (pokud známe jen jedno aktuální).

Schéma Algoritmu 2 je použito i jinde, než u generátoru BP. Jiným příkladem generátoru, který je založen na schématu Algoritmu 2, je generátor pseudonáhodných čísel implementovaný v Jazyce Java (http://java.sun.com). Ten má semínko o velikosti 48 bitů a jeho výstupem jsou 32-bitová čísla. My se však budeme dále zabývat generátorem BP.

Výše uvedené algoritmy generují celá čísla z intervalu od 0 do `MAXRAND`-1 (případně do $c-1$ při použití značení z (1)). Pro generování čísel z rovnoměrného rozložení na intervalu od 0 do 1 se používá původní algoritmus, který generuje čísla od 0 do `MAXRAND`-1, přičemž se výsledek vydělí číslem `MAXRAND`.

Jak již bylo řečeno, po vygenerování posloupnosti generátorem BP a otestování sadou testů DIEHARD bylo dosaženo překvapivých výsledků. Všechny testy dopadly tak, jako by byla testována opravdu náhodná posloupnost. Tedy některé testy byly sice na hladině významnosti 5% zamítnuty, ale byla jich přibližně dvacetina. Žádný test nebyl zamítnut na hladině významnosti např. 0,001% (což se někdy označuje jako selhání generátoru či jeho systematická chyba), ani po opakovaných testech na další vygenerované posloupnosti.

Sada DIEHARD byla vybrána proto, že obsahuje mnoho zajímavých a užitečných testů, dále proto, že testuje poměrně dlouhý úsek generované posloupnosti a v neposlední řadě proto, že některé nové sady z ní testy přebírají (např. sada testů NIST [2]), což svědčí o kvalitách sady DIEHARD.

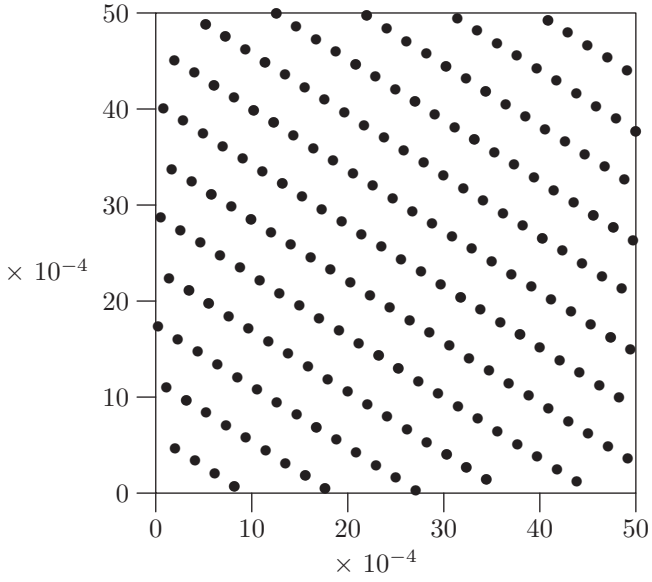
4. Generátory s lineární strukturou

Klasické generátory s lineární strukturou (Algoritmus 1) mají při generování dvojic čísel $(X_{2i-1}, X_{2i}), i = 1, \ldots$ a při dosažení periody (ekvivalentně vygenerování nekonečné posloupnosti[3]) tu vlastnost, že jsou generované dvojice uspořádány do nadrovin (viz obrázek 1). Vezmeme-li si generátor BP, je tato nepříjemná vlastnost (částečně) potlačena. Generované dvojice jsou na obrázku 2. Uveďme ještě periodu a nejmenší možný rozdíl mezi dvěma generovanými čísly pro oba generátory[4].

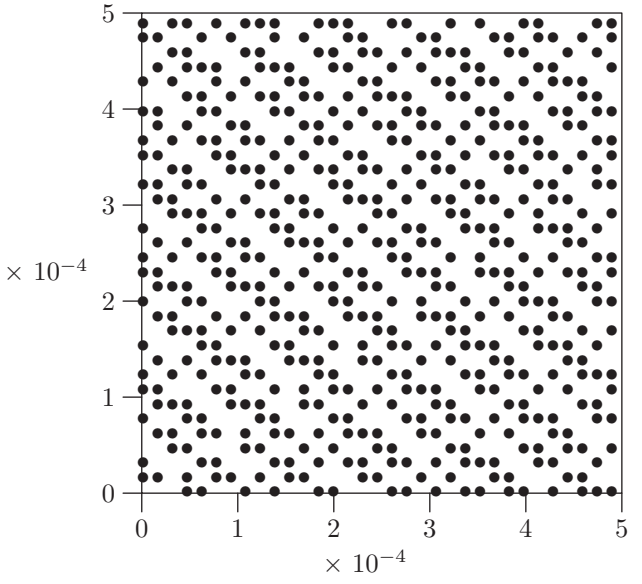
	perioda	nejmenší rozdíl dvou čísel
generátor VBA	2^{24}	2^{-24}
generátor BP	2^{32}	2^{-16}

[3]je-li délka posloupnosti sudá

[4]nejmenší rozdíl mezi dvěma libovolnými generovanými čísly (bez uspořádání), nikoliv mezi dvěma po sobě generovanými čísly



Obrázek 1: Dvojice $(X_{2i-1}, X_{2i}), i = 1, \ldots$ generované algoritmem VBA



Obrázek 2: Dvojice $(X_{2i-1}, X_{2i}), i = 1, \ldots$ generované algoritmem BP

5. Generátor jako Markovský proces

Fakt, že výstup generátoru BP „prošel“ sadou DIEHARD, je sice zajímavý, ale člověku hloubavému nedá spát. Je totiž pozoruhodné, že u takto primitivního generátoru s krátkou periodou (2^{32}, které se dosahuje) testy, které mají k dispozici 1/1000 této periody, nenašly žádné problémy. Proto byl navržen test, který by ukázal chyby tohoto generátoru a zejména předvedl některé vlastnosti generované posloupnosti. Tento test byl postaven na již zmíněné otázce predikovatelnosti (jednoho čísla jen z čísla předchozího). Bylo zkoumáno, jaká je pravděpodobnost jednotlivých čísel (16-bitových) za předpokladu, že víme, že přišlo dané (konkrétní) číslo. Generátor si můžeme představit jako Markovský proces v diskrétním čase, kde jednotlivé hodnoty generátoru (16-bitová čísla) určují stavy. Bylo-li tedy poslední vygenerované číslo a_i, jsme ve stavu i a ptáme se, jaká je pro každé j pravděpodobnost, že přejdeme do stavu j – tedy jaká je podmíněná pravděpodobnost, že další náhodné číslo bude a_j pokud víme, že poslední generované číslo bylo a_i.

Matici přechodových pravděpodobností P výše uvedeného Markovského procesu nemusíme jen odhadovat, dá se „snadno“ zjistit, neboť perioda generátoru je „jen“ 2^{32}. Lze tedy vygenerovat celou periodu a zapsat do frekvenční tabulky M počet přechodů z i do j. Matici M lze pak podrobit testům a je již předem jasné, že tímto generátor prohrál. Očekávaná hodnota v každém políčku tabulky je totiž 1. Navíc se po průchodu více period generátoru matice M pouze násobí konstantou.

Nyní to vypadá, že by se daný test vůbec nemusel provádět. Ale když už jsme došli až sem, bylo by vhodné podívat se na matici M a tím i na vlastnosti generované posloupnosti.

Při pokusu spočítat matici M můžeme narazit na některé „drobné problémy“, jako například

- matice se nevejde celá do paměti (bylo by potřeba 2^{32} 33-bitových čítačů, tj asi 17GB paměti)[5],
- použití disku pro počítání matice M či disku jako virtuální paměti nepřipadá v úvahu, neboť (za předpokladu náhodnosti posloupnosti) by pravděpodobnost, že čítač je ve fyzické paměti byla rovna podílu *velikost vyhrazené paměti / celková požadovaná paměť* a přístup na disk je velmi pomalý oproti paměti (pomalejší v několika řádech).

Ale jsou tu i jistá pozitiva, jako například to, že projít celou periodu trvá jen cca 30 minut na počítači 333MHz, tedy několik málo minut na dnes běžném kancelářském 2GHz Celeronu. Další výhodou je fakt, že generování posloupnosti

[5]stačí 32-bitových (tj. 4-bytových) čítačů plus servisní informace malé velikosti, jedna pro celou matici; z toho $2^{32} \times 4$ bytů $\approx$ 17GB paměti

je opakovatelné. Z praktického hlediska byl ještě učiněn předpoklad (který se potvrdil), že pro čísla v matici M nebudou potřeba celé 4 byty. Byly použity 1 bytové (8-bitové) čítače s tím, že pokud hodnota „přetekla“ (tj. rozmezí čítače nestačilo), provedl se zápis do speciálního souboru (log-souboru) a pak bylo nutné výsledek upravit. Celý test byl prováděn tak, že se spustil generátor na celou periodu a zápisy do paměti (matice M) byly omezeny jen na malý výsek této matice, tj. $i_1 \leq i \leq i_2$ a $j_1 \leq j \leq j_2$. Zápisy do matice M totiž znamenají jen zvýšení čítače o jedničku, tedy akci, která je závislá jen na jednom prvku této matice. Proto můžeme použít postup, kdy do paměti ve skutečnosti zapisujeme jen výřez matice M (tj. její podmatice) a zbytek ignorujeme. Spojením těchto výřezů nám vznikne celá matice M. Po prvním zkušebním průchodu (kdy byl zapisován do paměti jen levý horní roh M) se ukázalo, že výsledné čítače nabývají pouze hodnot 0,1 a 2. Zde by stálo za zvážení, zdali nemá smysl ještě „půlit byty“, tzn. do jednoho bytu ukládat dvě 4-bitová čísla, ale vzhledem k malé době dosažení celé periody a očekávané velké režii v důsledku „vylepšení“ bylo od tohoto nápadu upuštěno. Následně byla napočítána celá matice M a i ta obsahuje pouze hodnoty 0, 1 a 2.

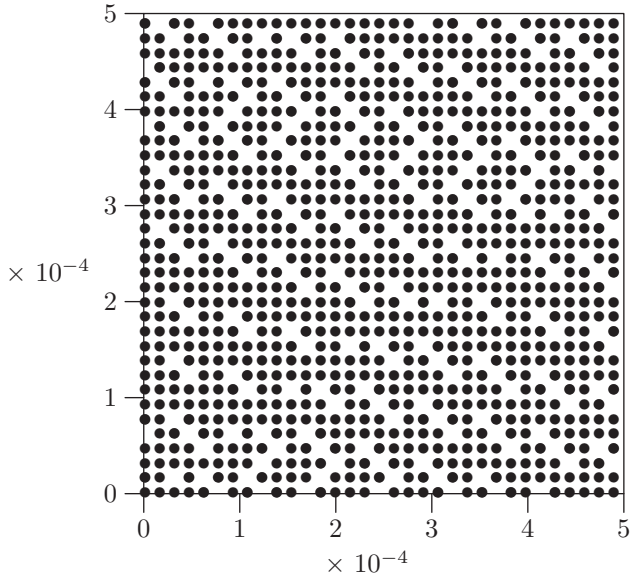
Je zajímavé, že studovaný („geniální“) generátor, který projde sadou testů, obsahuje v matici M pouze tři hodnoty. Na druhé straně musíme uznat, že právě tyto tři hodnoty mohou některé testy zmýlit, neboť pokud si vezmeme úsek generované posloupnosti, pak se nám přechod ze stavu i do stavu j jeví opravdu jako náhodný. V obrázku 3 je uveden levý horní roh matice M, tedy hodnoty i a j od nuly výše.

Hloubavý čtenář si možná všiml následujícího faktu. Obrázky 2 a 3 si neodpovídají (body na obrázku 2 mají odpovídat číslům různým od nuly na obrázku 3). Je to způsobeno faktem, že v obrázku 2 jsou generovány dvojice (vždy spotřebována dvě čísla a vykreslen z nich jeden bod), zatímco v matici M na obrázku 3 je generováno vždy jedno číslo a zvýšen čítač (předchozí číslo, aktuální číslo). Díky sudé periodě to není ekvivalentní.

Pojďme si tento rozdíl ukázat na příkladu generování čísel s periodou 4 a nechť vygenerovaná čísla jsou a,b,c a d. Pak pokud generujeme dvojice, získáme dvojice (a, b) a (c, d), dále se již opakují, zatímco pokud daný proces chápeme jako Markovský v diskrétním čase, získáváme přechody (a, b), (b, c), (c, d) a (d, a). Kdybychom při generování dvojic po dosažení periody vygenerovali jedno číslo a zahodili ho, dosáhneme obdobných výsledků, jaké jsou uvedeny na obrázku 3. Výsek matice M může být zavádějící též z toho důvodu, že se jedná o obdélníkový výsek (neproporcionální písmena jsou sázena do obdélníků s poměrem stran přibližně 2:1). Všechny možné dvojice po sobě jdoucích čísel vygenerovaných algoritmem BP jsou znázorněny na obrázku 4.

```
12110211111111111111111111011201120112011
11111111111111111201120112110211021102110211
10112011201120112011201120102110211111111
12110211021102110211021102111111111111111
20102111111111111111111111111111011201120112
02111111111111112011201120112110211021102110
11110112011201120112011202020102111111111
20112110211021102110211021102110211111111112
02020111111111111111111111111111111011201120
21102111111111120112011201120112011211021102
11111110112011201120112020202110111111111
01120112110211021102110211021102110211111120
20202110111111111111111111111111111011201
11021102111111201120112011201120112011211021
11111111110112011201120202021102110111111
11201120112110211021102110211021102110211201
02021102110111111111111111111111111111011
10211021102112011201120112011201120112011211
11111111111110112011202020211021102110211011
12011201120112110211021102110211021102111111
```

Obrázek 3: Část matice M; levý horní roh odpovídá pozici (0,0)



Obrázek 4: Všechny možné dvojice – algoritmus BP

6. Závěr

Pro generování „náhodných“ posloupností se často používají deterministické algoritmy. K testování generátorů se používají sady testů. Ty si všímají jen (pečlivě vybrané) skupiny systematických chyb.

Pokud potřebujeme opravdu kvalitní náhodná čísla, neměli bychom spoléhat na jednoduché generátory. Pokud daný generátor projde sadou testů, ještě to neznamená, že neobsahuje některé systematické chyby. Též by bylo vhodné se vyvarovat „geniálních zaručeně dobrých“ generátorů, pokud jsou typu černá skříňka (například nepopsané generátory náhodných čísel v softwarových balících). Může se stát, že právě v naší aplikaci se daná chyba projeví.

Dobrým námětem ke zvážení je fakt, zdali nepoužít ke generování nějaký fyzikální zdroj entropie, případně nezkombinovat několik těchto (nezávislých!) zdrojů neurčitosti. Též je otázkou, zdali používat málo známé a neprozkoumané generátory, nebo zdali používat jednoduché generátory (například lineární kongruenční generátory), které byly podrobeny mnoha testům a jsou známé i jejich slabé stránky.

Generátor implementovaný v Borland Pascalu 7 (generátor BP), má mnoho dobrých vlastností. Dokonce i predikovatelnost dalšího čísla při znalosti předchozího čísla se blíží ideálnímu generátoru. Tím ale neříkáme nic o situaci, kdy máme k dispozici dvě po sobě jdoucí čísla. Tam už může být situace jiná. Struktura generátoru BP je upravená struktura lineárního kongruenčního generátoru. Stejnou úpravu používá např. i generátor zabudovaný v Jazyce Java.

Literatura

[**1**] Marsaglia, G : DIEHARD, battery of tests of randomness, `http://stat.fsu.edu/~geo/diehard.html`

[**2**] National Institute of Standards and Technology: A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST Special publication 800-22, NIST 2001.

[**3**] Borland International: Borland Pascal 7 Resource Tools (RTL).

[**4**] Meneyes, A., van Oorschot, P., Vanstone, S: Handbook of Applied Cryptography. CRC Press, 1996.

Setkání TEXistů v Brně

Pavel Stříž [6]

Motto *Otázka: Proč jsou studovány obory jako jsou matematika a statistika?*
Odpověď: Aby mohlo býti psáno v TEXu.

Úvod

V sobotu 6. prosince 2003 se v Brně sešel Výbor CSTUGu (5i), proběhla přednáška Petra Olšáka na téma *K čemu a jak je možné použít TEX?* a uskutečnilo se Valné shromáždění Československého sdružení uživatelů TEXu. Diskuse se následně přesunuly mimo akademickou půdu.

Ani jako člen sdružení jsem neměl přístup k první části dne. O to více jsem si vychutnal zbylé dvě části. Taktéž poslední část tohoto TEXového dne jsem si nechal ujít. Téma bylo zajímavé, a to *Budoucnost TEXu a CSTUGu*, ale k tomu bych neměl mnoho co říci (vyjma vtipu na obr. 3).[7]

Obrázek 1.
Moravská metropole – Brno.

Setkání se zúčastnilo přibližně 60 účastníků. Konalo se v moravské metropoli Brně na půdě Fakulty informatiky Masarykovy univerzity (6i). Kdo by se rád něco dověděl o moravské metropoli, nechť navštíví například oficiální internetové stránky města Brna (4i).

Zvláště v sekci Významné osobnosti – Slavné osobnosti si doufám každý najde svoji oblíbenou osobnost. Určitě zaujme i sekce Historie – Brněnské pověsti a určitě také řada dalších sekcí.

O sdružení CSTUG

Československé sdružení uživatelů TEXu řeší dosti podobné problémy jako například i ČStS. Zvláště se jedná o příjmy, výdaje a o uspokojování náročných potřeb svých členů.

[6] ÚIS FaME UTB, Zlín, `http://uis.fame.utb.cz`,
e-mail `striz@fame.utb.cz, pavel.striz@email.cz`

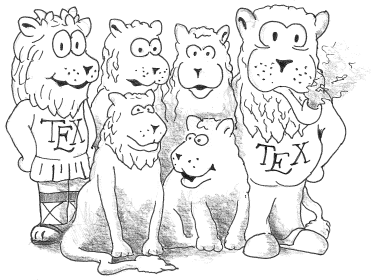
[7] Poděkování patří panu Antochovi za to, že mě „donutil“ psát v TEXu.
Rád bych také poděkoval panu Novozámskému za bezchybnou přípravu obrázků „na zakázku“.

Sdružuje uživatele a odborníky využívající programové vybavení pro stolní tisk. Pro zajímavost $\mathcal{CS}$TUG má 45 kolektivních a 307 řádných členů.

$\mathcal{CS}$TUG vydává pro své členy Zpravodaj, který je s dvanáctiměsíčním zpožděním přístupný na síti Internet (1i).

Pořádá a spolupořádá konference, semináře a přednášky. Spolupracuje s různými institucemi, produkuje programového vybavení pro oblast využívání TEXu a spravuje archiv software.

Navíc pečuje o to nejcennější a některým dosti známé: o TEX (8i).



Obrázek 2.
Spokojená rodinka u TEXů.

O přednášejícím: pan Olšák

Těm, kteří trochu znají TEX a jeho českou implementaci, není potřeba pana Olšáka příliš představovat.

Je to mimo jiné správce $\mathcal{CS}$TEXu a předseda $\mathcal{CS}$TUGu, autor knih *Typografický systém TEX* a *TEXbook naruby*, řady článků a softwarových doplňků pro TEX (3i).

Namátkou lze jmenovat makro `OFS` (Olšákův fontový systém) a makro pro tisk čárových kódů EAN, programy `vlna` (přidává vlnky (místo původních mezer) za neslabičné předložky), `a2ac` (pomůcka pro vytváření českých `PostScript`ových fontů) či konfigurovatelné menu `MNU` pro DOS. Na stránkách $\mathcal{CS}$TUGu mimo jiné informuje o Střešovické písmolijně (5i $\oplus$ stormtype/). Za zmínku stojí *i psané písmo se slabikáři* a další (3i).

O Marečkově knihkupectví

Na místě bylo možné přikoupit si starší čísla Zpravodaje a další publikace a této možnosti také mnozí využili.

Zájemci si mohli přikoupit i další knihy nabízené v Marečkově knihkupectví (2i). Přestože je mi známo, že kniha *TEXbook naruby* je přístupná na síti Internet (3i $\oplus$ tbn.html), neodolal jsem a knihu jsem si stejně koupil.

O problémech řešených na přednášce

Přednáška byla rozhodně z Vysoké školy TEXařské a nebyla pro žádné začátečníky. Diskutovaná makra byla přístupná již o dva týdny dříve (7i).

Bylo by opravdu prostorově náročné tlumočit nebo ještě lépe blíže seznámit s problémy a řešenými makry.

Za dostatečně provokující však považuji zmínit se o většině problémů a zajímavých situacích, které pan Olšák pomocí TEXu vyřešil (a na této přednášce zmínil).

Tisk písemek pro algebru a matematiku

Jeden z problémů je, jak z databáze zadání příkladů a jejich řešení připravit písemky (vyučující si vybírá příklady) tak, aby byla vhodná forma pro studenty (menší písmo pro tisk nebo větší písmo na fólie k promítnutí) s pomocnými čarami k nařezání papíru. Zvlášť se má vytisknout verze pro opravujícího (vyučujícího) s výsledky.

Verbatim tisk

Pokud je tištěn `PostScript`ový soubor, není na `PostScript`ové tiskárně většinou problém. Pokud však přijde na tisk textový soubor, pak je vhodné nechat tento soubor proběhnout TEXem a pak teprve „kontrolovaně“ tisknout.

Problémem je, jak nastavit filtr na proběhnutí TEXem, jak automaticky zjistit kódování češtiny a co vše se má kontrolovat před vlastním tiskem. Lahůdkou je posun tabulátorů na nejbližší celistvý násobek osmi znaků.

Pan Olšák navíc upozornil na volbu písma tak, aby šetřil toner a zároveň aby byl tisk ještě dosti čitelný. Zhuštěné písmo je pak vhodné k úspoře papíru.

Makro pro layout požadovaný Verlag Dashöfer

Zvláštností layoutu je, že se jedná o samostatné listy do kroužkové vazby a že musí být zachována velikost čar a fontů na mikrometr – při převodu autorova rukopisu.

Makro na takový přesný layout se pak hodí autorovi i nadále, pokud v rukopise pokračuje a chce doma na svém počítači vidět přibližný výstup požadovaný firmou Verlag Dashöfer.

Perličkou tohoto makra je jednoduché přepínání mezi různými fonty.

Inzertní příloha časopisu Dotek

V makru je celá série drobných vychytávek. Práce a vstupy ze souborů. Otáčení stran, dvousloupcová sazba a zajímavostí je plovoucí záhlaví po levé i pravé straně publikace.

V makru je zrealizováno i několik druhů výstupů, včetně náhledu s dekorací a pasovacími značkami[8].

Zajímavé bylo, že pan Olšák si ořezával obrázky ručně. Avšak takový ořezaný obrázek v textu vypadá velmi nerušeně. Druhou takovou zajímavostí byl poloprůhledný obrázek na celém listě.

Dekorace a přechody z barvy na barvu se také nevidí příliš často.

Čtení databázových údajů

Jakmile z `MySQL` databáze, často používané ve spojení s `PHP`, dostaneme textový soubor (např. přes parametr \t), můžeme chtít vytisknout některé údaje z této

[8] Pasovací značky nám zobrazují postupné zvětšování mezery mezi stránkami pro vnější archy svazečku tak, aby po ořezání stránky lícovaly.

databáze, například kartičky na slevy se jmény, adresy na obálky nebo štítky na konferenci. Takové makro pan Olšák připravil a stačí jen říci, jak má takový vzor vypadat, doplnit dělící čáry a může se tisknout.

Testy Kalibro

Jedná se o zpracování a vyhodnocování srovnávacích testů žáků základních a středních škol a o zpracování databází dat a porovnání odpovědí se správnými odpověďmi. Tisknou se následně jen závěry a shrnutí.

Slovíčka

Lahůdkou na závěr byla příprava kartiček k samostudiu anglického jazyka. Na jedné straně mají být anglické tvary (plus výslovnost, tak jak ji známe ze slovníků) a na druhé straně mají být české ekvivalenty.

Jde tedy o načítání dat, vhodné rozmístění na stránce, doplnění pomocných čar k nařezání a závěrečný tisk.

Pro změnu závěrem trochu TEXařiny

Předchozí problémy byly vyřešeny pomocí maker a je možné jimi procházet dosti dlouho. Na závěr článku uvedu jeden řešený a jeden neřešený problém, se kterými jsem se velice nedávno setkal.

Spolupřipravuji skripta a cvičebnici do stejnojmenného kurzu *Rozhodování při riziku a nejistotě* vyučovaného na naší alma mater[9].

V textu, který je určen pro zopakování látky minulých let, jsem narazil na Gaussovu eliminační metodu při řešení soustavy rovnic. Jedná se jen o úpravy matic plus komentáře, interpretace a závěry k výpočtům. Typograficky podobně můžeme psát i výpočet determinantů, inverzních matic atd.

Víte však, jak takové maticové úpravy zapsat, aby byly prvky matic zarovnané podle desetinné čárky?

Řešený problém

Nepovažuji se za příliš zkušeného TEXistu[10] a nechci nikoho urazit jednoduchým problémem, ale snad problém potěší. Potřebujete mít standardní balíky `dcolumn`, `color` a `gauss`.

[9] V době odeslání rukopisu článku je již známo, že název a náplň tohoto předmětu se budou významně měnit a zmíněné partie (prezentované jako zajímavé problémy pro čtenáře) vůbec použity nebudou.

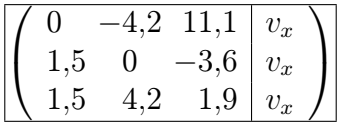
[10] TEXista – člověk, který píše nebo se zabývá typografií v TEXu.

Zkušený TEXista – člověk, který pracuje s TEXem a dosti mu rozumí.

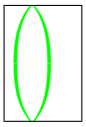
Vášnivý TEXista – člověk, který svým tělem brání svůj počítač, aby do něj někdo nenainstaloval jiný typografický systém.

Použiji balík `gauss`, aniž bych udělal úpravu zdrojového kódu. Použiji i běžné pole (`array`), které umím zarovnat dle desetinné tečky/čárky (`dcolumn`). Vše to spojím dohromady a bílou barvou odstraním věci přebytečné. Podobně smýšlíme i u příkazu `\phantom{}`. Pro přesné spojení použiji příkazů TEXu (zvláště příkaz `\kern`).

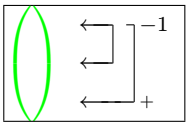
Při bližším zkoumání lze zjistit, že k vysázení stačí jen tyto tři základní části s jejich délkami:



Původní matice (`array`, `dcolumn`) šířka: 124.32336pt



Prázdná matice (`gauss`) šířka: 28.87497pt



Úpravy matice (`gauss`) šířka: 67.73627pt

Nyní stačí trochu programování (práce s příkazem `\kern ±<délka>`) a jsem schopen vysázet maticové úpravy beze změny zdrojového kódu.

```
1  \def\pravaa{{\color{green} \begin{gmatrix}[p] \\ \\ \rowops
2       {\color{black} \swap01 \add[-1]02}\end{gmatrix}}}
3  \def\stredd{{\color{green}\begin{gmatrix}[p]\\ \\
4       {\color{black}}  \end{gmatrix}}}
5  \def\levaa{\left( \begin{array}{D{,}{,}{1}D{,}{,}{1}D{,}{,}{1}|r}
6       0&-4,2&11,1&v_{x}\\1,5&0&-3,6&v_{x}\\1,5&4,2&1,9&v_{x}
7       \end{array} \right)}
8  \def\obvod #1{\vbox{\hrule \hbox{\vrule #1\vrule}\hrule}}
9  \def\mes {\message{výška: \the\ht0, hloubka:
10      \the\dp0, šířka: \the\wd0}}
11 \def\mess {\message{výška: \the\ht1, hloubka:
12      \the\dp1, šířka: \the\wd1}}
13 \def\messs {\message{výška: \the\ht2, hloubka:
14      \the\dp2, šířka: \the\wd2}}
15 \setbox0=\hbox{$\levaa$} \mes \setbox1=\hbox{$\stredd$} \mess
16 \setbox2=\hbox{$\pravaa$} \messs
```

Řádky 1-7 definují, co chceme zobrazit a v jaké barvě. Za *green* stačí dát *white* a na první pohled máme čistou typografickou úpravu. Řádky 8-16 si zjišťujeme potřebné délky s výstupem na obrazovku. Místo balíku `dcolumn` lze použít dva sloupce (první zarovnaný doprava a druhý doleva) – cifry před desetinnou čárkou s čárkou a cifry po desetinné čárce.

Například pomocí prostředí `alignat` si zajistíme možnost použití zarážek. Vysázíme prvně bílý text (v článku vytištěno šedou barvou) a přes něj pak černý text, námi požadovaný.

```
\begin{small}
\begin{alignat*}1
\setbox0=\hbox{$\levaa$} \setbox1=\hbox{$\stredd$}
\setbox2=\hbox{$\pravaa$} \kern \the\wd0 \kern -\the\wd1 \pravaa
     \kern -\the\wd2 \kern \the\wd1 \kern -\the\wd0
     \levaa \kern -\the\wd1 \kern \the\wd2
&\sim \def\pravaa{{\color{green} \begin{gmatrix}[p]
     \\ \\ \rowops {\color{black} \add12}\end{gmatrix}}}
\def\levaa{\left( \begin{array}{D{,}{,}{1}D{,}{,}{1}
     D{,}{,}{1}|r} 1,5&0&-3,6&v_{x}\\0&-4,2&11,1&v_{x}
     \\0&4,2&5,5&0\end{array}\right)}
\setbox0=\hbox{$\levaa$} \setbox1=\hbox{$\stredd$}
\setbox2=\hbox{$\pravaa$} \kern \the\wd0 \kern -\the\wd1 \pravaa
\kern -\the\wd2
     \kern \the\wd1 \kern -\the\wd0 \levaa
     \kern -\the\wd1 \kern \the\wd2 \sim\\ &\sim
\def\levaa{\left( \begin{array}{D{,}{,}{1}D{,}{,}{1}
     D{,}{,}{1}|r} 1,5&0&-3,6&v_{x}\\0&-4,2&11,1&v_{x}
     \\0&0&16,6&v_{x}\end{array} \right)}
\def\pravaa{{\color{green} \begin{gmatrix}[p] \\ \\
     \rowops \swap01 \add[-1]02 \end{gmatrix}}}
\setbox0=\hbox{$\levaa$} \setbox1=\hbox{$\stredd$}
\setbox2=\hbox{$\pravaa$} \kern \the\wd0 \kern -\the\wd1 \pravaa
     \kern -\the\wd2 \kern \the\wd1 \kern -\the\wd0
     \levaa \kern -\the\wd1 \kern \the\wd2
\end{alignat*}
\end{small}
```

Výstup, zarovnaný od zarážek zleva, může pak vypadat takto:

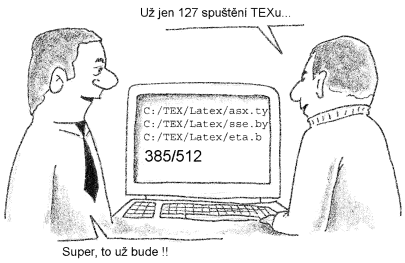
$$\left(\begin{array}{ccc|c} 0 & -4{,}2 & 11{,}1 & v_x \\ 1{,}5 & 0 & -3{,}6 & v_x \\ 1{,}5 & 4{,}2 & 1{,}9 & v_x \end{array}\right)\begin{smallmatrix}\leftarrow\\ \leftarrow\\ \leftarrow\end{smallmatrix}\!\!\begin{bmatrix}-1\\ \\ +\end{bmatrix} \sim \left(\begin{array}{ccc|c} 1{,}5 & 0 & -3{,}6 & v_x \\ 0 & -4{,}2 & 11{,}1 & v_x \\ 0 & 4{,}2 & 5{,}5 & 0 \end{array}\right)\!\!\begin{bmatrix}\\ \\ \leftarrow+\end{bmatrix} \sim$$

$$\sim \left(\begin{array}{ccc|c} 1{,}5 & 0 & -3{,}6 & v_x \\ 0 & -4{,}2 & 11{,}1 & v_x \\ 0 & 0 & 16{,}6 & v_x \end{array}\right)\begin{smallmatrix}\leftarrow\\ \leftarrow\\ \leftarrow\end{smallmatrix}\!\!\begin{bmatrix}-1\\ \\ +\end{bmatrix}$$

Neřešený problém

S touto zajímavou situací jsem se setkal při přípravě prvního vydání zmíněných skript psaných v MS Wordu®. Autoři chtěli mít na každé straně jeden citát nebo přísloví. To zrealizovat jde. Problém může nastat, pokud se žádají rozsáhlejší změny již ve vysázeném záhlaví a zápatí. MS Word® nás nutí udělat změnu na každé stránce.

Pokud si uvědomíme, že máme dokument o 150 stránkách a chceme udělat změny na každé stránce, tak je to práce dosti nezáživná.



Obrázek 3.
O budoucnosti TEXu?

Zkuste si vyřešit proměnné záhlaví a zápatí na každé straně dokumentu v TEXu, když víte, že změn bude celá řada (přidání bloku citátů, odebrání bloku citátů, přemísťované a různě řazené citáty a životní moudra).

V záhlaví můžete chtít citát a v zápatí informaci o jeho autorovi.

Jedno z možných řešení bude popsáno v dalším článku, který bude také zaměřen na typografický systém TEX, zvláště však bude o novém TEXLive 8.0 (TEXLive 2003) a CTANu. Můžete se těšit.

Zdroje Internet: Odkazy byly funkční k 14. únoru 2004.

(1i) `http://bulletin.cstug.cz/`
(2i) `http://marecek.kup.to/`
(3i) `http://petr.olsak.net/`
(4i) `http://www.brno.cz/`
(5i) `http://www.cstug.cz/`
(6i) `http://www.fi.muni.cz/`
(7i) `http://www.olsak.net/texpraxe.html`
(8i) `http://www.tug.org/`

Poznámka redakce Bulletinu ČStS. *Příspěvek pana Stříže o TEXu do tohoto bulletinu nepatří pouze zdánlivě. Jak dobře vědí především dosavadní přispěvatelé, redakce preferuje příspěvky připravené v tomto typografickém systému. Uživatelé MS Wordu to někdy nazývají diskriminací svého oblíbeného editoru. Neuvědomují si přitom, kolik úsilí stojí redakci převedení „wordovské“ předlohy do kulturní a čitelné podoby. MS Word prostě na vzorce není! Žádáme proto naše přispěvatele, aby i nadále posílali své příspěvky v TEXu a MS Wordu se na hony vyhnuli. Vaše příspěvky budou mnohem hezčí, čitelnější a navíc UŠETŘÍTE !!! TEXje zadarmo.*

Oznámení o semináři

Pracovní skupina pro statistické modelování při ENBIS (European Network for Business and Industrial Statistics) organizuje v úterý 11. května 2004 v Praze půldenní mezinárodní seminář na téma

Statistické modely a metody s aplikací v průmyslu.

Seminář se koná v Ústavu teorie informace a automatizace AV ČR, Pod vodárenskou věží 4, Praha 8, od 14 hodin, a bude sestávat z příspěvků pozvaných hostů, včetně informace o aktivitách ENBIS a zmíněné pracovní skupiny. Účast na semináři bude zdarma, pracovní jazyk angličtina. Program není dosud definitivní, předběžně přislíbili vystoupit:

David Rios Insua (Universidad Rey Juan Carlos, Madrid):
"*Bayesian discrete event simulation for a workflow line*"

Wolfgang Polasek (University of Wien):
"*Deregulation of the Austrian Telecom Market: 5 years experience*"

Simon Wilson (Trinity College, Dublin):
"*Signal processing, image analysis, material models for crack propagation and fatigue*"

Fabrizio Ruggeri (CNR-IMATI, Milano):
"*Reliability of repairable systems*"

Jaromír Antoch a Daniela Jarušková (MFF UK a FSv ČVUT, Praha):
"*Models of rare events*"

Pavel Ettler (COMPUREG Plzeň, s.r.o.):
"*Bayesian methods for industrial control systems*"

Další informace lze najít na internetové stránce
http://siprint.utia.cas.cz/public/ENBIS_WG/EWGPrague04.htm
(viz také *http://www.enbis.org*), případně je zájemcům poskytne osobně

Petr Volf, ÚTIA AV ČR, `volf@utia.cas.cz`

Druhé statistické dny v Hradci Králové

ve dnech 15. – 16. září 2004 (je to středa a čtvrtek) se uskuteční na Univerzitě Hradec Králové Druhé statistické dny (První statistické dny se v Hradci Králové konaly v roce 2001). Pořadateli jsou Česká statistická společnost a Jednota českých matematiků a fyziků spolu s katedrou matematiky Pedagogické fakulty Univerzity Hradec Králové.

Druhé statistické dny v Hradci Králové budou věnovány *aplikacím statistických metod při sledování vlivu člověka na životní prostředí*. Chceme se zaměřit na nejmodernější metody a prostředky statistického zpracování informací, které nám věda a dostupná technika poskytuje. Protože i člověk je součást přírody, organizátoři poskytnou prostor i těm, kteří se zajímají o vyhodnocování vztahu člověka k prostředí, ve kterém žije.

- ◇ *Datum konání*: 15. a 16. září 2004.
- ◇ *Místo konání*: 3. budova (nová) Univerzity Hradec Králové, Hradecká 1227/4.
- ◇ *Stravování*: Individuální, po dohodě možné hromadné objednání pro závazně přihlášené, kteří zaplatí konferenční poplatek do 15. 6. 2004.
- ◇ *Technické vybavení*: Tabule, zpětný projektor a dataprojektor s počítačem.
- ◇ *Abstrakt*: V rozsahu nejvýše jedné strany v camera ready zašlete do 30. 6.
- ◇ *Sborník* : Počítáme s jistou prezentací referátů, budeme se snažit je zveřejnit např. v Informačním Bulletinu ČStS.
- ◇ *Konferenční poplatek*:

Ubytování za jednu noc z 15. na 16. září	200 Kč
Stravování: oběd 15. září	70 Kč
večeře 15. září	50 Kč
snídaně 16. září (pro ubytované)	30 Kč
oběd 16. září	70 Kč
Náklady organizačního výboru	300 Kč
Celkem	720 Kč

  Ubytování je možné si zajistit v Palachových kolejích Univerzity.
- ◇ *Způsob úhrady*: Na účet JČMF, pobočky Hradec Králové. Při platbě je třeba použít variabilní symbol, který každý účastník obdrží spolu s potvrzením závazné přihlášky.
- ◇ *Storno poplatky*: Před 30. 6. 2004 činí 20%, při odřeknutí mezi 1.7. až 31. 8. činí 50%. Organizační výbor při odřeknutí v pozdějším termínu negarantuje vrácení ani části poplatku závazně přihlášeným účastníkům.
- ◇ *Druhé oznámení*: Závazně přihlášeným účastníkům do 5. července 2004.

Adresa pro korespondenci: Prof. RNDr. PhDr. Zdeněk Půlpán, CSc., katedra matematiky PdF Univerzity Hradec Králové, V. Nejedlého 573, 500 03 Hradec Králové 3. E-mail: `zdenek.pulpan@uhk.cz`.

Výroční zasedání České statistické společnosti

Minulé – už třinácté výroční zasedání České statistické společnosti se tentokrát konalo v Jindřichově Hradci na Fakultě managementu VŠE (6. fakulta Vysoké školy ekonomické v Praze, se sídlem v Jindřichově Hradci) ve čtvrtek 29. ledna 2004. Začátek byl ve 13.00 hodin. Na výročním zasedání byly předneseny obvyklé zprávy výboru společnosti o činnosti, hospodaření a plánech do budoucna. Ti, kdo se těšili na příspěvek Ing. Hany Šlégrové z Českého statistického úřadu na téma „Úkoly státní statistické služby ČR v souvislosti se vstupem ČR do EU“, byli zklamáni, neboť Ing. Šlégrová se kvůli velké zaneprázdněnosti nemohla zúčastnit. Nicméně program zachránil Marek Malý, který přednesl neméně zajímavou přednášku o tvorbě statistických dotazníků. Všem deseti účastníkům (z toho sedm členů společnosti) se přednáška líbila, stejně jako i příjemné prostředí a přátelská atmosféra, ve které se zasedání konalo. Po zasedání byl čas na prohlídku malebné historické části Jindřichova Hradce.

Gejza Dohnal

Informační Bulletin České statistické společnosti vychází čtyřikrát do roka v českém vydání. ISSN 1210 – 8022
Předseda společnosti: Doc. RNDr. Jaromír Antoch, CSc., KPMS MFF UK Praha, Sokolovská 83, 186 75 Praha 8, e-mail: jaromir.antoch@karlin.mff.cuni.cz
Redakce: Doc. RNDr. Gejza Dohnal, CSc., Jeronýmova 7, 130 00 Praha 3, e-mail: gejza.dohnal@fs.cvut.cz