

Rozhledy matematicko-fyzikální

Stanislav Trávníček
Generátor hudby

Rozhledy matematicko-fyzikální, Vol. 88 (2013), No. 4, 7–12

Persistent URL: <http://dml.cz/dmlcz/146544>

Terms of use:

© Jednota českých matematiků a fyziků, 2013

Institute of Mathematics of the Czech Academy of Sciences provides access to digitized documents strictly for personal use. Each copy of any part of this document must contain these *Terms of use*.



This document has been digitized, optimized for electronic delivery and stamped with digital signature within the project *DML-CZ: The Czech Digital Mathematics Library* <http://dml.cz>

Generátor hudby

Stanislav Trávníček, PřF UP, Olomouc

Abstract. Musical composers find inspiration from various sources; one of them has been even mathematics, its numbers and processes. This article deals with an idea of making a computer program which could provide the needed inspiration. The idea is analysed and it is shown that a computer may generate a certain kind of “technical music”. This way a composer can choose selected motives for his further art work.

Úvod

Matematika a hudba k sobě mají blízko, v každém případě dobře spolupracují, viz např. [1, 2, 3]. Připomeňme, že vůbec první člověk, o němž víme, že zkoumal vztahy mezi matematikou a hudbou, byl Pythagoras. Dnes neznámější uplatnění matematických metod v hudební analýze, samozřejmě s mohutným využitím výpočetní techniky, je patrně hledání charakteristik jednotlivých skladatelů a hudebních forem, kdy např. přehrajete „počítači“ nějakou skladbu a on vám řekne, kdo ji složil. Jde o rozsáhlé a hluboké analýzy, ale tímto směrem spolupráce se zde zabývat nebudeme.

My se věnujme kompozičnímu procesu, tj. skládání hudby a tomu, jak je matematika může ovlivnit. Zejména někteří dnešní hudební skladatelé se při tvorbě svých skladeb uchylují k novým a kreativním způsobům kompozice, kdy pramenem zajímavých nápadů a inspirací může být i matematika. Ale tomu se nebránili ani někteří dřívější skladatelé a využívali matematické principy třeba při tvorbě rozmanitých variací. Například o Mozartovi bylo známo, že některé tónové postupy určoval pomocí hracích kostek, i když toto zrovna matematika moc není. Avšak i když skladatel skládá svou hudbu a na matematiku vůbec nemyslí, tak v jeho skladbě ta matematika prostě je, např. výšky tónů – jsou vlastně čísla, stejně i délky tónů a intervaly mezi tóny, melodie je pak posloupnost čísel, jinou posloupností čísel je rytmus atd.

Hudební teoretici říkají, že soudobá hudba není často ani melodická ani harmonická, souzvuky se začínají rozvolňovat a to si vynucuje nové, moderní chápání hudby.

S tímto pohledem jsem v roce 2007 začal jistý pokus, o němž zde píšu, a další část článku bude proto i poněkud osobní. Vytvořil jsem algoritmus pro skládání „technické“ hudby a realizoval jej počítačovým programem; nazval jsem si jej *HudGe* (hudební generátor). Chápal jsem to v dalších letech jako občasnou zábavu, ale snad to může být inspirací pro ty čtenáře, kteří se aktivně zabývají nebo chtějí zabývat hudbou. Uvedu zde odůvodnění použité myšlenky a její uplatnění. Předpokládám, že čtenář zná základní hudební terminologii v rozsahu, jako se učí ve škole v předmětu hudební výchova.

Budeme jednat o tónech; vlastnosti tónu jsou výška, barva, síla a délka. Pro začátek jsem vyšel z toho, že si výsledné skladbičky budu přehrávat na sopránové zobcové flétně, čímž odpadly barva tónu a dynamika, zůstaly jen výška a délka. Zapsaný tón se nazývá nota, takže nám samozřejmě půjde i o noty a podle toho budeme dále oba pojmy používat.

Základní myšlenka

Podíváme se na řádek jedné národní písně (obr. 1):

Na tom našem dvoře

Mírně *z Bydžovska*

Na tom na - šem dvo-ře vše - chno to krá - ko-ře, i ten ko - hout:

Obr. 1

Tuto píseň zapíšeme takto: Každý tón (vlastně notu) zakódujeme jako dvojici (v, d) , kde v znamená relativní výšku tónu a d znamená jeho délku. Píseň je zapsána v F dur, takže relativní výška 1. tónu je 1 (prima); 2. tón má výšku 3 (tercie), 4. tón 4 (kvarta) atd.

Kód délky tónu zvolíme 1 pro tón osminový; pak čtvrtý tón má délku 2 a tříosminový délku 3. Záznamem písně je pak posloupnost (pro názornost vkládáme mezi takty mezery)

$$(1,2) (3,2) (4,2), \quad (5,3) (8,1) (5,2), \quad (5,1) (4,1) (4,2) (4,2), \\ (3,3) (5,1) (3,2), \quad (2,2) (2,3) (3,1), \quad (1,4) (0,2)$$

(pauzu jsme zatím zakódovali nulou).

Tuto píseň bychom mohli někomu poslat a nepotřebujeme ani notový papír. Můžeme též zjednodušeně zapsat ekvivalentním způsobem

$$1 \ 2 \ 3 \ 2 \ 4 \ 2 \ 5 \ 3 \ 8 \ 1 \ 5 \ 2 \ 5 \ 1 \ 4 \ 1 \ 4 \ 2 \ 4 \ 2 \ 3 \ 3 \ 5 \ 1 \ 3 \ 2 \ 2 \ 2 \ 2 \ 6 \ 6 \ 1 \ 1 \ 4 \ 0 \ 2,$$

nebo jen

123242538152514142423351322226611402.

Kdybyste si tento řádek chtěli zahrát na svůj hudební nástroj „z listu“, sotva by se vám to povedlo, ale pro počítač jsou takové vstupy docela příjemné.

Nyní se nabídla jednoduchá inverzní myšlenka: Vezmeme nějaký řetězec čísel a budeme jej považovat za zakódovanou jednohlasou skladbu. Co z toho vyjde? Právě o tom je program *HudGe*. Abych mohl vstupní podmínky parametrizovat a tím možnosti generování obohatit, trochu jsem tu původní jednoduchou myšlenku pozměnil a rozvinul.

Vstupy

Na rozdíl od předchozího příkladu se jako vstupy zpracovávají dvojice číslic, tedy čísla od 0 do 99, nazvěme si je *VK* (vstupní kódy). Lze je získat několika způsoby; vyzkoušel jsem tři. První tak, že do datového souboru zapíšeme dostatečně dlouhou posloupnost číslic, které se čtou po dvou a přetvářejí se na *VK*. Druhou možností je použití implicitního generátoru náhodných čísel a jako třetí (a snad nejzajímavější) alternativu jsem použil desetinného rozvoje při dělení dvou čísel, např. 54/71. Tento rozvoj je periodický, což má v generované hudbě polidšťující důsledky.

Délka tónu

Na počátku se zadá délka nejkratší noty, např. že to bude nota osminová, ta pak má kód 1, a délka taktu, např. C; do něj se pak vejdou noty s celkovou délkou 8. Za jednotku skladby jsem volil 16 taktů. I pro generování délek tónů je možno program mírně polidštit, např. tak, že pro těch 100 různých možných *VK* si předepíšeme, kterým délkám tónů dáváme jakou přednost. Ve výše uvedené situaci můžeme např. zadat, že délky tónu 1, 2, 3, . . . , 8 dostaneme, když *VK* bude

0–10, 11–35, 36–40, 41–65, 66–70, 71–90, 91–95, 96–99.

I tyto rozsahy můžeme postupně „odladit“. Přitom takt je respektován jako uzavřený celek, tedy vytvoří-li se délka větší, než je zbylé místo v rozdělaném taktu, délka se patřičně zkrátí. Návaznost tónů se realizuje např. tak, že když *VK* končí nulou, je právě vytvářená nota připojena k předešlé legátem.

Výška tónu

Tvorbou výšky tónu tím způsobem, který jsme použili u písni „Na tom našem dvoře“, bychom dostali něco, co by bylo celkem k neposlechu. Proto bylo lépe vytvářet výšku tónů užitím intervalů a generování velikosti intervalů z VK ovlivnit stejnou metodou, jako jsme ukázali v předchozím odstavci u délky. Rozdíl je ten, že intervaly mohou jít jak nahoru (kladné kroky), tak dolů (záporné kroky), a to lze zařídit jednoduše tak, že uvážíme rozdíl $VK - 50$, a v rozsahu od 0 do 50 přidělíme jednotlivým hodnotám $|VK - 50|$ velikost kroku od 0 až po hodnotu, pro kterou se rozhodneme. Přitom nastavíme předpokládanou frekvenci jednotlivých kroků. Při zadávání vstupních dat se zadá i tónový rozsah vytváření skladby (např. jak jsme se zmínili o zobcové flétně), takže musíme vhodně ošetřit případy, kdy by nám následující tón měl vyjít mimo zadaný rozsah. Některé z hodnot VK rezervujeme pro pauzu. Opět pro polidštění skladby je vhodné připravit zvlášť výšku prvního i posledního tónu skladby, jak je to v dané tónině obvyklé.

Notace

Zdrojový text programu v Pascalu má dnes asi 2 200 řádků. Začal vznikat za jednoduchých předpokladů a po ověření funkčnosti a kvality výsledku se jeho možnosti postupně rozšiřovaly. Protože byl od počátku koncipován ve shodě se strukturovaným programováním, nemusel se přepracovávat, ale jen se doplňoval o nové funkce a možnosti. Také šlo i o to, *poslechnout si* skladbu, kterou program vytvořil. (Dále jako příklad uvádíme šestnáctitaktovou skladbičku vytvořenou ze vstupních dat podílu 15/53). Nejprve byla zvolena jednoduchá notace na obrazovce, viz ukázkou v tab. 1:

```
!01C.2 !01A.2 !02E.2 !02H.1 !01G.2 !01E.2 !02H.1 -02E.1 !04H.1 !01H.1
*03=== !04D.2 !04D.2 !01H.1 !01G.2 !02D.2 !02A.2 !01F.2 !01D.2 !02A.1
-02D.1 !04A.1 !01A.1 *03=== !04C.2 !04C.2 !01A.1 !01F.2 !02C.2 !02G.2
!01E.2 !01C.2 !02G.1 -02C.1 !04G.1 !01G.1 *03=== !04H.1 !04H.1 !01G.1
!01E.2 !02H.1 !02F.2 !01D.2 !01H.1 !02F.1 -02H.1 !04F.2 !01F.2 *03===
!04A.2 !04A.2 !01F.2 !01D.2 !02A.1 !02E.2 !01C.2 !01A.2 !02E.2 -02A.1
!04E.2
```

Tab. 1

Každý tón je zaznamenán šesticí znaků. První je charakteristika tónu, a to ! pro samostatný tón, - je znak pro legato, tedy připoj k předchozímu

tónu, a hvězdička * je znakem pro pauzu. Pak následuje dvojmístná délka tónu, jak jsme ji výše popsali, a na konci je jméno tónu i s příslušnou oktávou. Po kratším cviku jsem si dokonce podle této notace dovedl skladbu přehrát. Ale pak mě na nějakou dobu zaujala práce na běžné, daleko srozumitelnější notaci; stejná skladba je pak (programem *HudGe*) zobrazena takto (obr. 2):



Obr. 2

Poznámka: Pro zajímavost podotýkám, že díky mému vnukovi Tomášovi, profesionálnímu informatikovi, jsem mohl doplnit pascalovské unity tak, že jsem program *HudGe* rozšířil o možnost zapsat vytvářenou skladbu souběžně ve formátu MID (může se zvolit i hudební nástroj, standardně jsem zadal klarinet), tedy po jejím složení si ji lze na počítači hned přehrát.

Možnosti programu

Pro čtenáře, které tyto otázky zajímají tak, že si budou chtít takový generátor hudby také pořídit, doplňuji další možnosti programu, který jsem v *HudGe* už realizoval, a které se volí na počátku chodu programu:

- tónina – lze volit všech 12 možností, a to v dur i moll (harmonické, melodické nebo aiolské)
- délka skladby – lze volit 1 až 8 bloků po 16 taktech
- druh taktu – 2/4, 3/4 nebo C
- nejkratší nota – 1/16, 1/8 nebo 1/4
- nejnižší tón, nejvyšší tón – volí se rozsah; možnosti od c kontra po h3
- generátor vstupu – již bylo uvedeno
- výstup výsledků – volba možností, co zobrazit na displeji a co uložit do souboru (noty lze jen na displeji a soubor MID jen uložit)

Jak už bylo zmíněno, další možnosti lze uplatnit přímo změnami ve zdrojovém textu programu.

Závislost generované hudby na vstupech

Všimněme si naší ukázky, u níž VK byla desetinná místa podílu 15/53.

Tato desetinná místa začínají 2830 1886 7924 5 a pak se tato skupina třinácti číslic periodicky opakuje. Vytvořily se tak 3 tóny a vzhledem k přesahu jedné číslice jsou následující čtveřice jiné než ty počáteční, v přesahu jsou teď 2 číslice, přidáním další periody dostaneme přesah 3 číslic a po přidání další periody je počet číslic $4 \times 13 = 52$; vytvořilo se 13 tónů a tuto posloupnost tónů můžeme nazvat hudební motiv. Z tab. 1 a obr. 2 je vidět, že se tu stejný motiv (ale pokaždé v jiném provedení) vyskytuje pětkrát, přitom délky tónů (něco jako rytmus) se opakují, ale výšky vytvářejí různé variace. Ověřte si tentýž motiv z 15/53 a variace ještě u skladby v tónině d moll aiolské (obr. 3).



Obr. 3

Pokud u „své“ skladby tolik variací motivu nechceme, zvolíme za jmenovatele větší číslo (stovky nebo tisíce). Čítatel způsobuje jen jakýsi posun motivů. Jestliže jsou *VK* náhodná čísla, pak se ve skladbě nic neopakuje (a zní trochu nelidsky), pokud se *VK* berou z dato-vého souboru, je přečtená *n*-tice chápána jako perioda ve výše uvedeném smyslu.

Závěr

Vraťme se k úvodním odstavcům. Potenciální hudební skladatel si může na počítači takové motivy vygenerovat, upravit (dát do nich duši) a použít při své tvorbě. Já nejsem hudební skladatel, ale udělal jsem jinou věc – užitím vhodného SW jsem si k některým vytvořeným melodiím pořídil druhý hlas a doprovod (basa, kytara, klarinety, někde i další hudební nástroje).

Literatura

- [1] Piknerová, J.: *Matematika a hudba*. DP, MU, Brno, 2009.
- [2] Trávníček, S.: Rytmičké etudy. *Rozhledy matematicko-fyzikální* **60**, 3 (1981/82), s. 105–117.
- [3] Trávníček, S.: Matematické podklady k rytmickým etudám. *MFI* **11**, 3 (2001/02), s. 168–177.