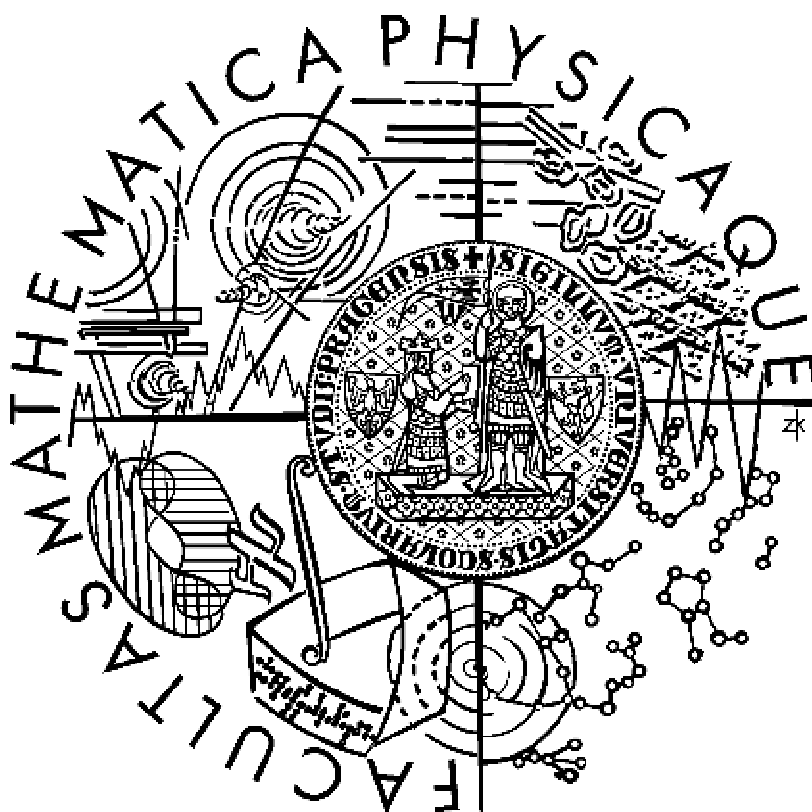


Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

DIPLOMOVÁ PRÁCE



Tomáš Hradecký

HARMONICKÁ ANALÝZA SKLADBY

Katedra softwarového inženýrství

Vedoucí diplomové práce: Doc. RNDr. Jan Obdržálek, CSc.

Studijní program: Informatika, databázové systémy

Poděkování: Děkuji všem, kteří mi byli oporou při vytváření této práce, především svému vedoucímu Doc. RNDr. Janu Obdržálkovi, CSc., který mi poskytl veškeré podklady k teorii akustiky a ladění, a také za obětavý přístup při konzultacích. Dále děkuji Prof. Vladimíru Tichému, CSc., díky jehož aktivnímu přístupu mohla být tato práce prezentována na HAMU. Vznesené připomínky mi pomohly k vylepšení software, který je součástí této práce. V neposlední řadě děkuji vedoucímu katedry softwarového inženýrství Prof. RNDr. Jaroslavu Pokornému, CSc. za to, že podpořil zadání této mezioborové práce. Její specifické zaměření na pomezí informatiky, akustiky a hudební teorie je nezvyklé, ale věřím, že tato práce bude přínosem pro vzájemnou spolupráci vědy a umění.

Nedílnou součástí práce je přiložený CD-ROM se software Přelad' a jeho zdrojovým kódem.

Prohlašuji, že jsem svou diplomovou práci napsal samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

Ve Praze dne 12.12.2006

Tomáš Hradecký

Obsah

1.	Úvod	6
2.	Hudební ladění.....	7
2.1.	Základní pojmy.....	7
2.1.1.	Interval.....	7
2.1.2.	Centová míra	7
2.1.3.	Přirozené intervaly.....	8
2.2.	Druhy ladění	8
2.2.1.	Rovnoměrné temperované ladění	8
2.2.2.	Pythagorejské ladění.....	9
2.2.3.	Přirozená ladění	10
2.2.4.	Částečně temperovaná ladění	12
3.	Harmonické a melodické tóny.....	13
3.1.	Použitá kritéria melodičnosti tónu.....	13
3.2.	Použitá kritéria harmoničnosti tónu.....	15
3.3.	Klasifikace melodických a harmonických tónů.....	16
3.3.1.	Příklad použití klasifikace	16
3.3.2.	Použití více kritérií a princip učení	17
4.	Použití programu – implementované funkce ladění.....	18
4.1.	Způsoby nastavení ladění	18
4.2.	Manuální nastavení ladění.....	19
4.2.1.	Výběr předem nastaveného ladění.....	19
4.2.2.	Vlastní výběr tónů z dlaždicového prostoru	21
4.2.3.	Vlastní nastavení centového posunu tónů	22
4.2.4.	Vložení zvoleného přeladění do skladby	22
4.3.	Automatické nastavení ladění.....	23
4.3.1.	Vlastní klasifikace harmonických a melodických tónů.....	23
4.3.2.	Automatická klasifikace harmonických a melodických tónů.....	23
4.3.3.	Kombinace automatické a ruční klasifikace melodických tónů	24
4.3.4.	Automatická harmonická analýza skladby	24
4.4.	Postup při učení parametrů harmonických a melodických tónů.....	25
5.	Algoritmus automatické harmonické analýzy skladby.....	27
5.1.	Krok A: Rozdělení tónů na melodické a harmonické.....	28
5.1.1.	Aktuálně znějící harmonie	28
5.2.	Krok B: Rozpoznání typu akordu a umístění tónů v dlaždicovém prostoru	29
5.2.1.	Pravidla pro rozpoznání typu akordu	29
5.2.2.	Porovnání aktuálně znějící harmonie s pravidly.....	31
5.3.	Krok C: Změna ladění	34
5.3.1.	Pořadí pokusů změn ladění.....	35
5.3.2.	Příklad změn ladění pro konkrétní skladbu	35
6.	Algoritmus učení pro stanovení vah parametrů analýzy melodických tónů	38
6.1.	Obecný popis práce genetických algoritmů.....	38
6.2.	Genetický algoritmus pro stanovení vah parametrů analýzy melodických tónů.....	39
6.2.1.	Genetická reprezentace – kódování parametrů.....	39
6.2.2.	Vytvoření počáteční populace	41
6.2.3.	Výpočet cílové funkce	41
6.2.4.	Operátor selekce	41

6.2.5.	Operátory křížení a mutace.....	43
6.3.	Dosažené výsledky	43
7.	Příloha A: popis standardu MIDI	45
7.1.	Formát SMF – Standard MIDI File (*.mid)	45
7.1.1.	Hlavička SMF.....	45
7.1.2.	Hlavička stopy	46
7.1.3.	Data stopy	46
7.1.4.	Kódování deltačasu.....	46
7.1.5.	Kódování typu a dat události	47
7.1.6.	Popis formátu událostí	48
7.1.7.	Trvající stavový byte	49
7.1.8.	Formát metaudálostí	50
7.1.9.	Nejdůležitější typy metaudálostí SMF	50
7.1.10.	Zvláštní systémová data	51
7.2.	Zápis mikroladění do MIDI.....	52
7.2.1.	Mikroladění pomocí ohýbání tónu	52
7.2.2.	Instrukce mikroladění – standard MTS	53
7.3.	Dodatek – tabulky MIDI not	57
7.3.1.	Tabulka číslování MIDI not	57
7.3.2.	Tabulka frekvencí MIDI not.....	58
8.	Příloha B: Uživatelská dokumentace.....	59
8.1.	Okno Hlavní nabídka.....	59
8.2.	Okno Notová osnova	60
8.3.	Okno Hledání melodických tónů	61
8.4.	Okno Ladítko	62
8.5.	Okno Dlaždice	63
9.	Příloha C: Programátorská dokumentace	65
9.1.	Datové struktury	65
9.1.1.	Datové struktury pro definici skladby	65
9.1.2.	Datové struktury pro nastavení ladění	69
9.1.3.	Datové struktury pro klasifikaci melodických tónů	70
9.1.4.	Datové struktury genetického algoritmu	71
9.2.	Algoritmy	73
9.2.1.	Algoritmus načítání MIDI	73
9.2.2.	Algoritmus uložení MIDI	74
9.2.3.	Algoritmus automatické harmonické analýzy skladby.....	75
9.2.4.	Genetický algoritmus pro stanovení vah kritérií	76
9.2.5.	Některé pomocné výpočty	78
9.3.	Dodatek: Vlastní formáty datových souborů.....	81
9.3.1.	Formát pravidel pro změnu ladění – Pravidla.dat.....	81
9.3.2.	Formát pořadí pokusů změn ladění – Zmenaladeni.dat.....	81
9.3.3.	Formát *.lad - uložené ladění	81
9.3.4.	Formát *.vah – Parametry klasifikace melodických tónů	82
9.3.5.	Formát *.pga – Parametry genetického algoritmu.....	82
9.3.6.	Formát *.skl – Vlastní formát pro uložení skladby	82
10.	Seznam použité literatury	85

Abstrakt / Abstract

Název práce: Harmonická analýza skladby

Autor: Tomáš Hradecký

Katedra: Katedra softwarového inženýrství

Vedoucí diplomové práce: Doc. RNDr. Jan Obdržálek, CSc., katedra teoretické fyziky

E-mail vedoucího: jan.obdrzalek@mff.cuni.cz

Abstrakt: Práce popisuje známé poznatky teorie ladění a na jejich základě navrhuje algoritmus pro dynamické přirozené ladění. Nedílnou součástí práce je software Přelad', který obsahuje následující funkce:

- načtení hudební skladby v podobě MIDI souboru
- vkládání instrukcí mikroladění a export výsledku opět jako MIDI soubor
- rozřídění tónů na melodické a harmonické na základě uživatelsky nastavitelných kritérií
- automatický návrh změn ladění v průběhu skladby s cílem dosažení přirozených intervalů v akordech
- možnost uživatelské korekce výsledného návrhu automatického přeladění
- genetický algoritmus pro hledání suboptimálního nastavení vah kritérií a prahu pro analýzu melodických a harmonických tónů

Text práce obsahuje popis známých ladění, vysvětlení pojmů harmonický a melodický tón, popis formátu MIDI souborů a popis funkcí programu Přelad'. Popsán je také algoritmus automatického doplnění přirozeného ladění a genetický algoritmus pro hledání suboptimálního nastavení vah kritérií pro analýzu melodických a harmonických tónů. Součástí textu práce je také uživatelská a programátorská dokumentace k programu Přelad'.

Klíčová slova: ladění, přirozené ladění, mikroladění, MIDI, genetický algoritmus

Title: Harmonic analysis of a musical composition

Author: Tomáš Hradecký

Department: Department of software engineering

Supervisor: Doc. RNDr. Jan Obdržálek, CSc., Department of theoretical physics

Supervisor's e-mail address: jan.obdrzalek@mff.cuni.cz

Abstract: This work describes what is known about music theory and it suggests an algorithm for dynamic just-intonation tuning. Software Přelad' (Retune) is an integral part of this work. This software contains the following functions:

- loading a MIDI file containing a musical composition
- inserting microtuning instructions and exporting to a MIDI file
- classifying melodic and harmonic notes
- the automatic suggestion of a tuning in the musical composition with a view to achieving just-intonation distances between notes in chords
- the possibility of user corrections of the automatic suggestion
- genetic algorithm to find suboptimal settings for the weighting of criteria for classifying melodic and harmonic notes

The text of this work contains the description of well-known tunings, the explanation of melodic and harmonic notes, the description of the MIDI-file format and the description of the software Prelad (Retune). It also describes the algorithm for the automatic suggestion of a tuning and the genetic algorithm for finding suboptimal settings for weighting of criteria for classifying melodic and harmonic notes. This text also contains the user's manual and the program documentation.

Keywords: tuning, just-intonation tuning, microtuning, MIDI, genetic algorithm

1. Úvod

Již od první chvíle, kdy člověk začal používat hudební nástroje, se důležitým hudebním pojmem stalo ladění. Tuto problematiku můžeme chápat komplexně jako otázku výšky tónu, barvy zvuku nebo také souznění jednotlivých nástrojů. Tato práce podrobně rozebírá jednu z těchto oblastí, a to ladění ve smyslu stanovení relativní výšky jednotlivých tónů.

Slovo ladění vychází ze slova lad, neboli soulad – tedy něco, co společně zní dobře. Otázkou je, co zde míníme pojmem „dobře“ – protože hudba je oborem značně subjektivním, a názory na to, co je dobré se mohou značně lišit. Rozdíly v subjektivním chápání souladu se mohou u různých hudebníků lišit, ladění také prošlo dlouhým historickým vývojem. Během staletí vznikly různé systémy ladění, které více či méně plnily požadavky na ně kladené.

Tato práce není porovnáním, který z nich je lepší, ani návrhem nových ladění. Naopak, cílem bylo vytvoření praktického software, umožňujícího nastavení změny ladění v existujícím MIDI záznamu. Tím je umožněn poslech stejné skladby zahrané stejným přednesem, ale s jinými mikrointervalově nastavenými výškami tónů.

Ladění je možno nastavovat dynamicky neboli proměnlivě, čímž se zde myslí možnost nastavit změnu ladění kdykoliv v průběhu skladby. Tím uživatel získává široké možnosti úprav mikrointervalového ladění pro různé části skladby dle vlastního uvážení.

Pro usnadnění práce obsahuje software také algoritmy pro rozlišení melodických a harmonických tónů a automatické doplnění vhodného přirozeného ladění dle aktuálně znějící harmonie. Zde je ale nutno podotknout, že vzhledem ke složitosti a nutnému subjektivnímu vnímání významu hudebních pojmů nemůže být strojový postup nikdy dokonalý. Proto má poslední slovo vždy uživatel – ten může algoritmicky vytvořený návrh rozlišení tónů a automaticky navržené ladění upravovat dle svého hudebního cítění.

Pokud si to uživatel bude přát, může nastavovat parametry těchto algoritmů. Pro nastavení vah kritérií klasifikace melodických a harmonických tónů je v programu implementován genetický algoritmus umožňující hledání vhodných parametrů na základě vzorového řešení předloženého uživatelem. Program tak dokáže nastavovat své parametry tak, aby při dalším použití lépe splňoval požadavky uživatele.

2. Hudební ladění

V této kapitole stručně shrneme známé poznatky z teorie ladění. Podrobněji tuto problematiku popisuje například Obdržálekova Teorie ladění [6], Geistova Akustika [2] nebo Tvrzského Systémy ladění klávesových nástrojů [9].

Zabývat se budeme pouze nástroji, které umožňují přesné nastavení výšky jednotlivých tónů, fyzikálně vyjádřitelné hodnotou frekvence tónu. To nám umožní počítat vztahy výšek jednotlivých tónů.

Poznámka: Ne u všech nástrojů je mikrointervalové ladění jednotlivých tónů pevně nastavitelné. Například u dechových nástrojů je možné výslednou výšku tónu značně ovlivnit silou dechu a postavením úst, u smyčcových nástrojů je možné výšku tónu spojitě měnit. Pevné nastavení výšky tónu pochopitelně nelze očekávat ani v případě zpěvu.

Našemu požadavku tedy vyhovují klávesové nástroje, kde je možné mikrointervalově naladit tóny jednotlivých kláves, nebo nástroje elektrofonické a zvukové karty počítačů, které umožňují reprodukovat tón požadované frekvence.

2.1. Základní pojmy

2.1.1. Interval

Interval (hudební) znamená vzdálenost dvou tónů. Lze jej vyjádřit poměrem frekvencí tónů, které interval tvoří.

Hudební intervaly mají svá historická jména (prima, sekunda, ...), která budeme dále v textu používat. Podrobný výklad intervalů a hudební teorie je obsažen např. v Kofroňově Učebnici harmonie [4].

2.1.2. Centová míra

Pro diskrétní rozdělení oktávy na dílky používáme tzv. centové míry. Oktáva je rozdělena na 1200 stejně velkých intervalů zvaných cent. Výraz *stejně velké intervaly* znamená, že poměr frekvencí sousedních tónů je vždy stejný.

Poměr frekvencí odpovídající intervalu jednoho centu je tedy roven $\sqrt[1200]{2}$. Označíme-li frekvence sousedních stupňů centové míry jako f_i a f_{i+1} , pak platí:

$$f_{i+1} = \sqrt[1200]{2} \cdot f_i$$

což lze zobecnit pro interval velikosti k centů jako

$$f_{i+k} = 2^{\frac{k}{1200}} \cdot f_i$$

Pokud naopak budeme znát poměr frekvencí tónů a budeme chtít znát velikost intervalu v centech, vyjádříme si ze vzorce proměnnou k

$$k = 1200 \cdot \log_2 \frac{f_{i+k}}{f_i}$$

Poznámka: Název cent je odvozen od skutečnosti, že jeden temperovaný půltón se skládá ze 100 centů.

Poznámka: Jelikož cent je logaritmem poměru frekvencí, odpovídá násobení frekvence tónu sčítání v centové míře. Dělení frekvence pak znamená odčítání v centové míře.

2.1.3. Přirozené intervaly

Přirozeným intervalem nazýváme takový, jehož poměr frekvencí lze vyjádřit jako poměr malých celých čísel. Protože s přirozenými intervaly budeme dále v textu pracovat, vyjmenujme nejčastěji používané přirozené intervaly:

- Čistá oktáva: poměr frekvencí $2/1$ (velikost intervalu v centech: 1200 centů)
- Čistá kvinta: poměr frekvencí $3/2$ (velikost intervalu v centech: 702 centů)
- Čistá kvarta: poměr frekvencí $4/3$ (velikost intervalu v centech: 498 centů)
- Velká tercie: poměr frekvencí $5/4$ (velikost intervalu v centech: 386 centů)
- Malá tercie: poměr frekvencí $6/5$ (velikost intervalu v centech: 316 centů)

Poznámka: Čistá kvarta je doplněním čisté kvinty do oktávy, podobně malá tercie je doplněním velké tercie do kvinty. Podobným způsobem vznikají přirozené intervaly malé a velké sexty.

2.2. Druhy ladění

Jednotlivé teorie ladění nepředepisují pevně frekvence jednotlivých tónů, ale na základě frekvence základního tónu odvozují podle svých pravidel frekvence ostatních tónů.

Existují různé standardy základního tónu (viz Geistova Akustika [2]). Ve většině případů se sice přidržíme použití standardu frekvence 440 Hz pro jednočárkované a , obecně je ale v každém druhu ladění možné zvolit frekvenci základního tónu jakoukoliv.

Všechna používaná ladění dodržují oktávy – interval oktávy pro každý tón vždy odpovídá dvojnásobku frekvence. Z toho plyne, že výšky jednotlivých tónů stačí stanovit pouze v rámci jedné oktávy. V ostatních oktávách lze frekvenci dopočítat pomocí násobení nebo dělení dvěma.

Jednotlivé systémy ladění se tedy odlišují různými vztahy mezi frekvencemi jednotlivých tónů.

2.2.1. Rovnoměrné temperované ladění

Rovnoměrné temperované ladění je nejčastěji používaným druhem ladění. Rozděluje oktávu na 12 stejně velkých intervalů – temperovaných půltónů, které tvoří chromatickou stupnici.

Protože poměr frekvencí sousedních chromatických stupňů musí být vždy stejný a při postupu o 12 půltónů musíme dostat dvojnásobnou frekvenci (interval oktávy), je interval jednoho temperovaného půltónu roven $\sqrt[12]{2}$.

Výhodou rovnoměrného temperovaného ladění je rovnocennost tónin – všechny stupnice obsahují stejné intervaly, skladby znějí ve všech tóninách obdobně a je bez problémů možné tóninu změnit nebo z ní vybočit.

Nevýhodou rovnoměrného temperovaného ladění je nepřesnost přirozených intervalů; rovnoměrné temperované ladění zachovává jediný přirozený interval – oktávu. Ostatní intervaly se od ideálního poměru frekvencí odlišují a dosahují jej pouze přibližně – je ale až s podivem, že se výsledné hodnoty ideálu velmi blíží.

Intervaly v rovnoměrně temperovaném ladění:

- Temperovaná čistá kvinta: $\left(\sqrt[12]{2}\right)^7 = 1,4983\dots$ (700 centů; rozdíl -2 centy od poměru 3/2)
- Temperovaná čistá kvarta: $\left(\sqrt[12]{2}\right)^5 = 1,3348\dots$ (500 centů; rozdíl +2 centy od poměru 4/3)
- Temperovaná velká tercie: $\left(\sqrt[12]{2}\right)^4 = 1,2599\dots$ (400 centů; rozdíl +14 centů od poměru 5/4)
- Temperovaná malá tercie: $\left(\sqrt[12]{2}\right)^3 = 1,1892\dots$ (300 centů; rozdíl -16 centů od poměru 6/5)

2.2.2. Pythagorejské ladění

Pythagorejské ladění si stanovuje za základní interval čistou kvintu a všechny tóny stupnice odvozuje právě za pomoci tohoto intervalu. Složením 12 kvint nad sebe získáváme tzv. pythagorejský kruh, kde výsledné tóny převedené oktavovými posuny do jedné oktávy a seřazený dle výšky vytvářejí tzv. pythagorejskou stupnici.

Tuto stupnici tvoříme od základního tónu například šesti kvartovými kroky dolů a šesti kvintovými kroky nahoru. Pokud zvolíme jako základní tón *c* a výšky všech tónů budeme zapisovat v rámci jedné oktávy, získáváme následující pythagorejský kruh:

Tab. 1: Pythagorejský kruh

Název tónu	Rozdíl od zákl.tónu v centech	Rozdíl od zákl.tónu v centech (v rámci jedné oktávy)
ges	-4212 centů	588 centů
des	-3510 centů	90 centů
as	-2808 centů	792 centů
es	-2106 centů	294 centů
b	-1404 centů	996 centů
f	-702 centů	498 centů
c	0 centů	0 centů
g	702 centů	702 centů
d	1404 centů	204 centů
a	2106 centů	906 centů
e	2808 centů	408 centů
h	3510 centů	1110 centů
fis	4212 centů	612 centů

Tím získáváme celou stupnici – pro úplnost zde uvádíme tóny fis i ges, ve dvanáctitónové stupnici si vybereme jeden z nich. Vidíme, že tón ges se zhruba o 24 centů (přibližně čtvrtina půltónu) liší od tónu fis. Důvodem je, že dvanáct kvint se oproti sedmi oktavám liší právě o těchto 24 centů. Tomuto rozdílu se říká pythagorejské komma.

Pythagorejské ladění vzniklo v době, kdy se v hudbě neužívalo akordů, ale především melodie. Vzhledem k tomu, že všechny intervaly odvozujeme jen pomocí kvinty a oktávy,

vycházejí ostatní přirozené intervaly nepřesně, přičemž už u obou tercií je výsledek horší než v případě temperovaného ladění.

Jediným správně znějícím přirozeným intervalem je tedy kvinta (a kvarta jako její obrat). To ovšem platí pouze pro tóny, které spolu sousedí v pythagorejském kruhu.

Jiné „kvinty“ jsou o pythagorejské komma menší v případě, že v pythagorejském kruhu postupujeme opačně – například fis-des nebo h-ges, nebo naopak o pythagorejské komma větší, pokud překračují celý pythagorejský kruh, například ges-cis nebo des-gis.

Kvinta takto zvětšená nebo zmenšená o pythagorejské komma se nazývá vlčí kvinta (quinte de loupe) a zní nelibozvučně.

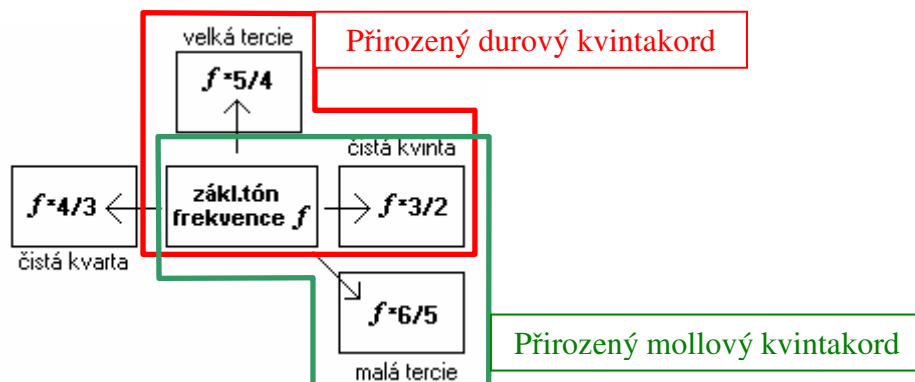
2.2.3. Přirozená ladění

Přirozenými laděními se dále v textu budeme zabývat nejvíce. Základním stavebním kamenem přirozených ladění je durový kvintakord. Jeho součástí jsou intervaly velké a malé tercie a čisté kvinty. V obrazech získáme interval kvarty a velkou a malou sextu.

Vytvořme si, podobně jako v Obdržádkově Teorii ladění [6], diskrétní dvojrozměrný prostor, kde postup po ose x o jedno políčko doprava bude znamenat interval kvinty, a postup po ose y o jedno políčko nahoru bude znamenat interval velké tercie. Vztahy mezi sousedními prvky tohoto prostoru graficky znázorňuje následující obrázek.

Poznámka: Pro zjednodušení dalšího textu budeme tento prostor dále nazývat „Dlaždicovým prostorem přirozených ladění“ nebo zkráceně „Dlaždicovým prostorem“.

Obr. 1: Dlaždicový prostor – schematické znázornění přirozených intervalů



Poznámka: Všechny tóny udržujeme pomocí oktavových kroků v rámci jedné oktávy. Tóny v jiných oktavách lze odvodit násobením nebo dělením dvěma. Poměry frekvencí v tabulce tedy budou 5/4 nebo 5/8, 4/3 nebo 2/3, 3/2 nebo 3/4 a 6/5 nebo 3/5. Kvintakordy v tabulce tedy mohou být v obrazech, ale vždy z nich lze pomocí násobení a dělení frekvencí dvěma získat základní kvintakord. V textu budeme používat rozmezí frekvencí v rozmezí (220 Hz; 440 Hz), což odpovídá rozsahu od malého a po jednočárkované a. Pro přehlednost dále nebudeme rozlišovat, ve které oktávě (malá, jednočárkovaná) se tón konkrétní výšky nachází, čárku (např. a') tedy psát nebudeme.

Poznámka: Zde popsané zobrazení dlaždicového prostoru se mírně odlišuje od zobrazení použitého v Obdržádkově Teorii ladění [6]. Tam jsou prostřední tóny kvintakordů rozmístěny podobně pro durový i mollový kvintakord, tak, aby svislý průmět odpovídal běžnému pořadí tónů

Příklad: durový a mollový kvintakord c-dur a c-moll jsou zobrazeny takto: $c \frac{e}{es} g$

V dlaždicovém prostoru vytvoříme přirozenou diatonickou stupnici od základního tónu tak, aby obsahovala kvintakordy tónický, dominantní a subdominantní. Uvedme jako příklad přirozenou diatonickou stupnici C-dur a c-moll. Základní tón stupnice je vyznačen červeně, tóny přirozené diatonické stupnice jsou vyznačeny zeleně.

Obr. 2: Přirozená diatonická stupnice C-dur v dlaždicovém prostoru

244,44 h	366,67 fis	275,0 cis	412,50 gis	309,37 dis	232,3 ais	348,5 eis	261,3 his
391,11 g	293,33 d	440,0 a	330,0 e	247,50 h	371,25 fis	278,43 cis	417,65 gis
312,89 es	234,67 hes	352,0 f	264,0 c	396,0 g	297,0 d	222,75 a	334,12 e
250,31 ces	375,47 ges	281,60 des	422,40 as	316,80 es	237,60 hes	356,40 f	267,30 c
400,50 asas	300,37 eses	225,28 heses	337,92 fes	253,44 ces	380,16 ges	285,12 des	427,68 as

Obr. 3: Přirozená diatonická stupnice c-moll (aiolská) v dlaždicovém prostoru

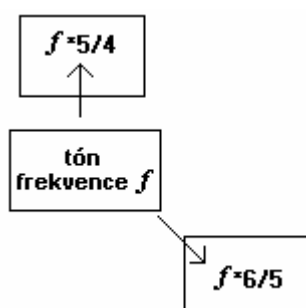
244,44 h	366,67 fis	275,0 cis	412,50 gis	309,37 dis	232,3 ais	348,5 eis	261,3 his
391,11 g	293,33 d	440,0 a	330,0 e	247,50 h	371,25 fis	278,43 cis	417,65 gis
312,89 es	234,67 hes	352,0 f	264,0 c	396,0 g	297,0 d	222,75 a	334,12 e
250,31 ces	375,47 ges	281,60 des	422,40 as	316,80 es	237,60 hes	356,40 f	267,30 c
400,50 asas	300,37 eses	225,28 heses	337,92 fes	253,44 ces	380,16 ges	285,12 des	427,68 as

Poznámka: Vzhledem k tomu, že používáme rozmezí frekvencí v intervalu jedné oktávy v hodnotách (220 Hz; 440 Hz), je výška tónu h vypočtena jako 247,5 Hz, což odpovídá dvojnásobné frekvenci (495 Hz) v jednočárkované oktávě. Obdobně u tónu b (hes) je odpovídající frekvenci v jednočárkované oktávě hodnota 475,2 Hz.

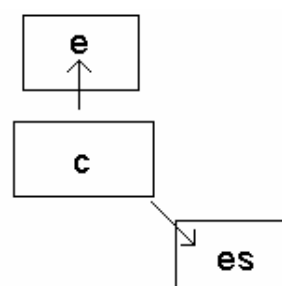
Otázkou zůstává, jak máme doplnit tóny chromatické. Hudební teorie zná několik postupů:

- Aristoxenovo ladění tvoří chromatické tóny zvětšením primy, tj. rozdílem velké tercie (c-e) a malé tercie (c-es). V našem schématu to tedy znamená, že můžeme pomocí známého tónu vytvořit nové chromatické tóny doplněním do durového nebo mollového kvintakordu.

Obr. 4: Chromatické tóny Aristoxenes

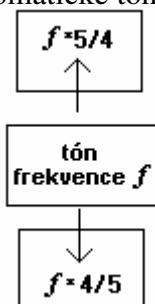


Příklad:

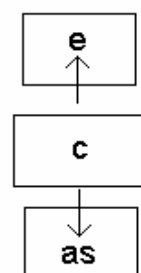


- Ptolemaiovo ladění tvoří chromatické tóny přidáním velké tercie nebo jejího obratu – malé sexty.

Obr. 5: Chromatické tóny Ptolemaios

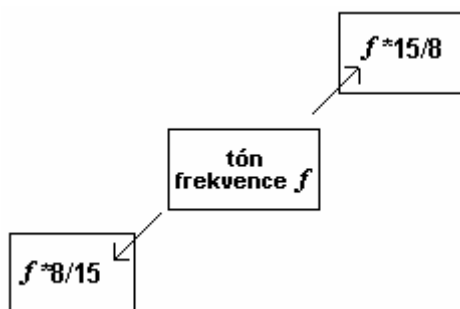


Příklad:

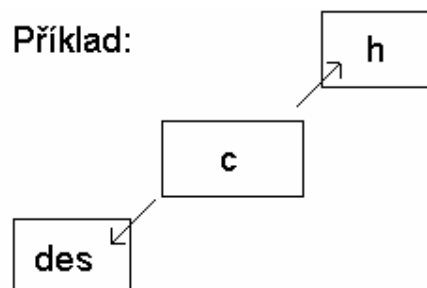


- Delezenovo ladění využívá půltónového rozdílu mezi tóny v diatonické stupnici.

Obr. 6: Chromatické tóny Delezenne



Příklad:



- Pokud si jako kritérium při určování chromatických tónů stanovíme pokrytí co největšího množství přirozených durových i mollových kvintakordů, přičemž z enharmonických tónů (např. cis-des, dis-es) si vybereme vždy pouze jeden z nich, získáme následující systém, který je symetrický pro tónorod dur i moll.

Obr. 7: Chromatické tóny symetrické v dur a moll – příklad pro základní tón c

244,44 h	366,67 fis	275,0 cis	412,50 gis	309,37 dis	232,3 ais	348,5 eis	261,3 his
391,11 g	293,33 d	440,0 a	330,0 e	247,50 h	371,25 fis	278,43 cis	417,65 gis
312,89 es	234,67 hes	352,0 f	264,0 c	396,0 g	297,0 d	222,75 a	334,12 e
250,31 ces	375,47 ges	281,60 des	422,40 as	316,80 es	237,60 hes	356,40 f	267,30 c
400,50 asas	300,37 eses	225,28 heses	337,92 fes	253,44 ces	380,16 ges	285,12 des	427,68 as

2.2.4. Částečně temperovaná ladění

V historii se používaly také systémy, které se nazývají částečně temperované. Tyto systémy vycházejí většinou z přirozených ladění, ale z důvodů potřeby použitelnosti více tónin byly některé intervaly částečně temperovány. O existenci těchto ladění se zde zmiňujeme pro úplnost a dále se jimi v textu nebudeme zabývat. Podrobněji se těmito laděními zabývají Tvrzského Systémy ladění klávesových nástrojů [9]

3. Harmonické a melodické tóny

Tóny klasické vícehlasé skladby mají buď melodický nebo harmonický charakter. Melodické tóny tvoří melodii, zatímco harmonické tóny tvoří harmonický doprovod, který může být vyjádřen buď souzvukem nebo rozloženým akordem.

Z definice vidíme, že hranice mezi harmonickými a melodickými tóny není ostrá – například posloupnost tónů *c-e-g* můžeme chápat jako melodickou (začátek písně Ovčáci čtveráci) nebo i jako harmonickou (rozložený kvintakord c-dur).

Pokud si ale stanovíme, že harmonickou analýzu chceme provádět pouze na základě harmonických tónů, je nutné od sebe melodické a harmonické tóny jednoznačně oddělit. To může provést zkušený hudebník, nebo je možné stanovit si pro určitou skladbu pevná kritéria pro rozlišení harmonických a melodických tónů.

Rozpoznání harmonických a melodických tónů podle určitých kritérií vyžaduje stanovení *vah* jednotlivých kritérií a *práh*, neboli mezní hodnotu, jež stanovuje hranici mezi melodickými a harmonickými tóny.

V programu se používá celkem dvanáct kritérií „harmoničnosti“ a „melodičnosti“ tónu, přičemž některé z nich mají ještě další parametry. Používaná kritéria byla zvolena na základě pozorování a konzultací s vedoucím diplomové práce. Jejich důležitost ale není explicitně dána, ale závisí na nastavení jejich vah.

Hodnoty vah kritérií melodičnosti nabývají kladných hodnot, kritéria harmoničnosti nabývají záporných hodnot. Každý tón skladby je ohodnocen podle všech kritérií, přičemž jejich váhy se v případě splnění kritéria sčítají. Tón, jehož součet vah je vyšší nebo roven hodnotě prahu, je označen jako melodický, ostatní tóny jsou označeny jako harmonické.

Následující seznamy vyjmenovávají jednotlivě použitá kritéria.

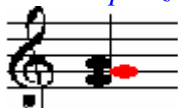
Poznámka: Při posuzování kritérií, u kterých závisí na intervalu mezi tóny, jsou jednotlivé tóny chápány v rámci jedné oktávy, tj. pracujeme s 12 tóny uspořádanými do kruhu.

3.1. Použitá kritéria melodičnosti tónu

- tvorba sekundy v harmonii – toto kritérium je splněno, pokud tón tvoří v souzvuku s dalšími tóny interval malé nebo velké sekundy (v rámci kruhového uspořádání tónů v rámci jedné oktávy). Tón musí být zároveň tvůrcem sekundového intervalu, tj. pokud sečteme počet sekund zkoumaného tónu a okolních tónů, je počet sekund zkoumaného tónu nejvyšší.

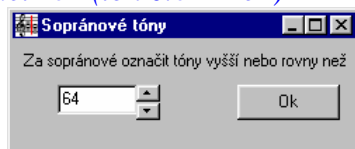
Toto kritérium splňují tóny, které jsou cizí harmonickému podkladu durového nebo mollového kvintakordu.

Příklad: V následujícím souzvuku tón g splňuje kritérium tvorby sekundy v harmonii, protože tvoří s okolními tóny dvě sekundy (s tóny f a a), zatímco tóny f a a vytvářejí sekundu pouze jednu. Tóny f a a kritérium ze stejného důvodu nesplňují.



- kritérium sopránu (tón vyšší než stanovený parametr) – toto kritérium je splněno, pokud je tón ostře vyšší než stanovený parametr. Toto kritérium by mělo být používáno jako doplňkové kritérium zdůrazňující kritérium sopránu jako nejvyššího znějícího tónu.

Příklad: V následujícím souzvuku tóny f, a, d splňují kritérium sopránu jako tónů vyšších než parametr 64 (tón č.64 = e1)



- kritérium sopránu (nejvyšší tón) – toto kritérium je splněno, pokud je analyzovaný tón nejvyšším aktuálně znějícím tónem

Příklad: V následujícím souzvuku tón d splňuje kritérium sopránu jako nejvyššího aktuálně znějícího tónu



- kritérium samostatnosti – toto kritérium je splněno, pokud je tón stisknut samostatně, neboli že ve stejném okamžiku nebyly stisknuty jiné tóny.

Příklad: označený tón c splňuje kritérium samostatnosti



- kritérium tvorby sekundy v melodii – toto kritérium je splněno, pokud tón tvoří interval malé nebo velké sekundy s okolními tóny z obou stran. Toto kritérium splňuje např. stupnicový postup, nebo střídavý tón (viz Šínova Nauka o harmonii [8]).

Příklad: označený tón splňuje kritérium tvorby sekundy v melodii



- kritérium kanálů melodických tónů – toto kritérium má význam pouze v případě, že hlasy jsou umístěny do jednotlivých kanálů a některý hlas obsahuje pouze melodické tóny. Kritérium je splněno pro všechny tóny označených kanálů.

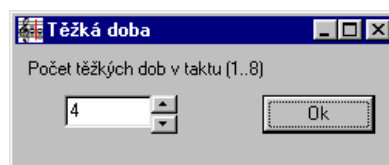
Příklad: maximální váha kritéria a označení kanálů melodických tónů



3.2. Použitá kritéria harmoničnosti tónu

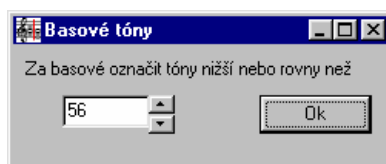
- těžká doba – toto kritérium je splněno, pokud je tón stisknut na těžkou dobu. To, zda je doba těžká nebo lehká, je specifikováno parametrem *počet těžkých dob v taktu*, který může nabývat hodnot 1 až 8.

Příklad: černě označené tóny splňují kritérium těžké doby pro parametr počtu 4 těžkých dob v taktu. Červeně označené tóny toto kritérium nesplňují



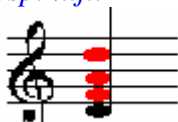
- kritérium basu (tón nižší než stanovený parametr) – toto kritérium je splněno, pokud je tón ostře nižší než stanovený parametr. Toto kritérium by mělo být používáno jako doplňkové kritérium zdůrazňující kritérium basu jako nejnižšího aktuálně znějícího tónu.

Příklad: V následujícím souzvuku tóny g splňují kritérium basu jako tónů nižších než tón č.56 (malé gis) a jsou označeny černě jako harmonické. Červeně označené tóny toto kritérium nesplňují.



- kritérium basu (nejnižší tón) – toto kritérium je splněno, pokud je analyzovaný tón nejnižším aktuálně znějícím tónem

Příklad: V následujícím souzvuku je černě označen tón d, který splňuje kritérium basu jako nejnižšího aktuálně znějícího tónu. Červeně označené tóny toto kritérium nesplňují.



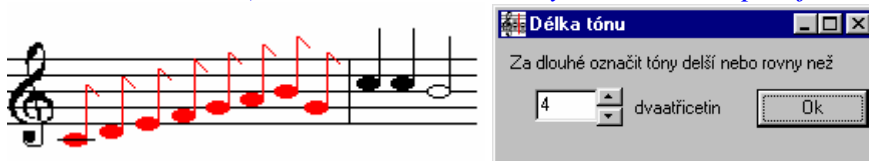
- kritérium nesamostatnosti – toto kritérium je splněno, pokud tón není stisknut samostatně, neboli že ve stejném okamžiku došlo ke stisknutí jiných tónů.

Příklad: černě označené tóny splňují kritérium nesamostatnosti (současně došlo ke stisknutí jiných tónů), červeně označený tón kritérium nesplňuje.



- kritérium dlouhého tónu – toto kritérium je splněno, pokud tón je delší než parametricky stanovený počet dvaatřicetin.

Příklad: černě označené tóny splňují kritérium dlouhého tónu pro parametr 4 (tj. tóny delší než osminová), červeně označené tóny kritérium nesplňují.



- kritérium kanálů harmonických tónů – toto kritérium má význam pouze v případě, že hlasy jsou umístěny do jednotlivých kanálů a některý hlas obsahuje pouze harmonické tóny. Kritérium je splněno pro všechny tóny označených kanálů.

Příklad: maximální váha kritéria a označení kanálů harmonických tónů



3.3. Klasifikace melodických a harmonických tónů

Program umožňuje uživateli určení melodických tónů buď ručně, nebo automaticky na základě uživatelsky nastavitelné váhy jednotlivých kritérií. Tím jsou nastaveny dostatečně široké meze pro různé výklady pojmů harmonický a melodický tón, a záleží na uživateli, jaké váhy kritérií zvolí a jaké tóny určí ručně.

V praxi se předpokládá, že uživatel nastaví váhy kritérií, poté spustí automatické rozpoznání a nakonec opraví klasifikaci tónů, které nebyly stanoveny podle jeho přání – může označit další melodické tóny, nebo automaticky rozpoznané melodické tóny označit zpět jako harmonické.

Nejproblematictější úkolem tedy zůstává, jak zvolit váhy kritérií a prahovou hodnotu. Pro různé typy skladeb se tyto parametry mohou lišit, různý může být také výklad definice harmonického a melodického tónu.

3.3.1. Příklad použití klasifikace

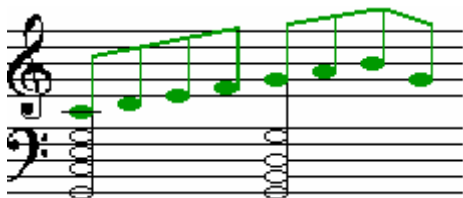
Aplikaci různých vah kritérií si ukážeme na příkladu národní písně Vyletěla holubička. Pokud jako jediné kritérium stanovíme například tvorbu sekundy v harmonii, budou jako melodické označeny pouze tóny, které vytvářejí sekundu.

Obr. 8: Rozpoznání melodických tónů – tvorba sekundy v harmonii (tóny klasifikované jako melodické jsou označeny zeleně)



Pokud bychom zvolili jako jediné například kritérium sopránů jako nejvyššího tónu, bude výsledek určení melodických tónů jiný.

Obr. 9: Rozpoznání melodických tónů – kritérium sopránů (tóny klasifikované jako melodické jsou označeny zeleně)



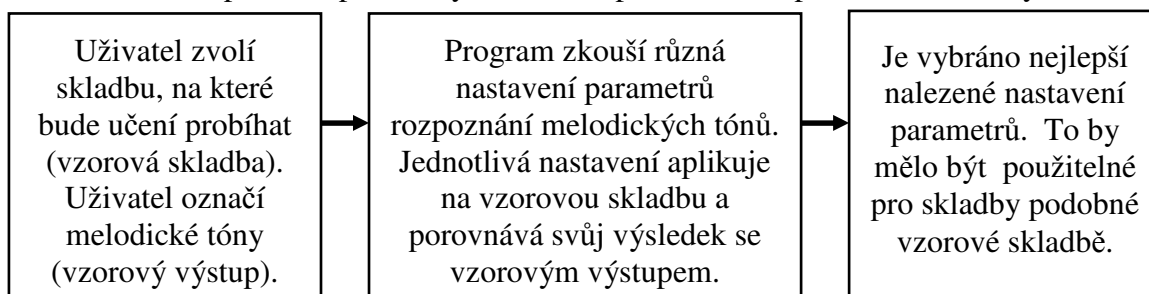
3.3.2. Použití více kritérií a princip učení

Ne vždy ale bude jediné kritérium stačit – potřebných výsledků bude v různých typech skladeb možno dosáhnout pouze kombinací více kritérií a nastavením jejich vah a prahové hodnoty.

Jak má ale uživatel vůbec tušit, jaké váhy budou pro jeho potřeby optimální? Metodou pokus-omyl se může dobrat k rozumnému nastavení, ale není zaručeno, že na správné řešení přijde u všech typů skladeb.

Proto byl jako vhodná pomůcka pro stanovení kritérií použit princip učení. Program sám o sobě sice neví, jak má nastavit váhy kritérií a prahovou hodnotu, ale na základě předloženého správného označení melodických tónů může vyzkoušet různé možnosti nastavení parametrů a zvolit nejlépe vyhovující řešení.

Obr. 10: Princip učení aplikovaný na hledání parametrů rozpoznání melodických tónů



Nejjednodušší algoritmus pro nalezení optimálního řešení je vyzkoušení všech možností nastavení vah a prahu. To je však v praxi neproveditelné, protože možností je příliš mnoho – nastavujeme 12 vah, přičemž některé z nich mají další parametry. Celkem do výpočtu vstupuje 20 proměnných. Každá z nich může nabývat minimálně 128 různých hodnot. Z toho plyne, že počet možností je minimálně 128^{20} . I kdyby použití jednoho nastavení na vzorovou skladbu trvalo pouhou mikrosekundu (ve skutečnosti to bude větší množství času), vyzkoušení všech možností bychom se dočkali za více než 10^{28} let.

Budeme tedy muset použít heuristickou metodu – budeme zkoušet jen některá řešení a z nich vybereme to nejlepší. Nebude sice zaručeno, že to bude řešení optimální, ale pokud bude zvolená dostatečně „chytrá“ metoda, můžeme i tak dosáhnout dobrého výsledku. Pro praktické použití tedy plně postačí nalezené suboptimální řešení.

V programu je prakticky použita metoda tzv. genetického algoritmu – na začátku jsou stanoveny náhodné váhy, které se postupně v několika krocích (generacích) zlepšují. Podrobně je tento postup a dosažené výsledky popsán v kapitole 6 (Algoritmus učení pro stanovení vah melodických tónů).

4. Použití programu – implementované funkce ladění

Program „*Přelad*“, který je součástí této práce, je praktickou aplikací umožňující skutečnou práci s různými laděními. Program umožňuje načíst hudební skladbu v podobě MIDI souboru, vkládat do tohoto souboru instrukce pro změnu ladění a výsledek opět exportovat jako MIDI soubor.

Ladění je možné v průběhu skladby dynamicky měnit. Program „*Přelad*“ nabízí několik způsobů, jak může uživatel ladění nastavit. Tyto způsoby si nyní popíšeme.

4.1. *Způsoby nastavení ladění*

Program obsahuje následující funkce:

A. Manuální nastavení ladění

V první řadě má uživatel možnost vybrat si předem připravené ladění (např. pythagorejské, Aristoxenovo, apod., (viz kapitola 2: Hudební ladění) a stanovit jeho základní tón. Toto ladění se názorně zobrazuje v dlaždicovém prostoru.

Další možností je, že si uživatel může přímo vybrat tóny z dlaždicového prostoru, které chce použít.

Nejobecnější, ale také nejpracnější, je možnost nastavení vlastního ladění v rámci mikrointervalového posunu jednotlivých tónů oktávy.

Nastavená ladění je možné ukládat na disk a později znovu používat.

B. Automatické nastavení ladění

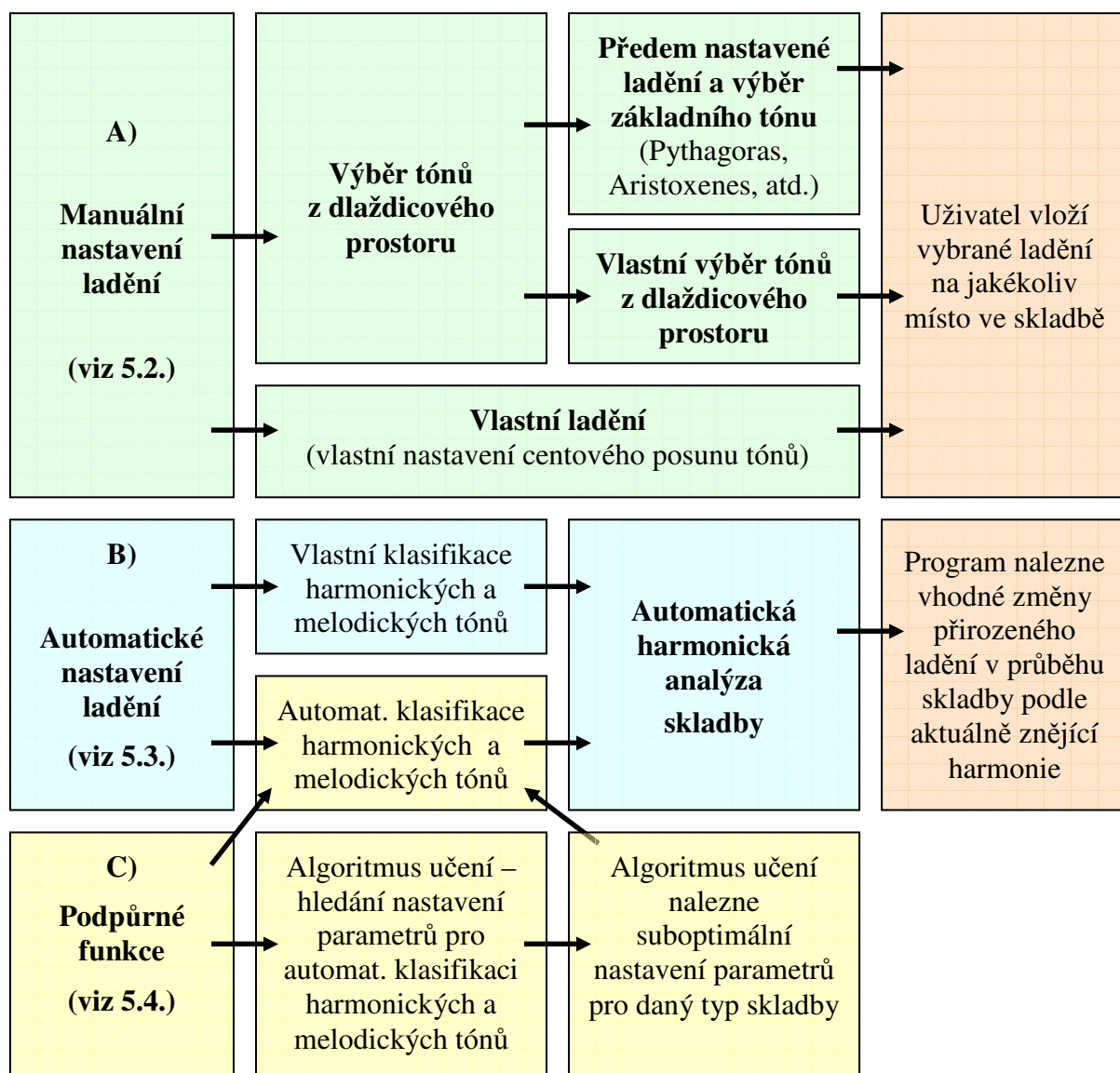
Dalším možným způsobem použití programu je využití funkce automatického návrhu doplnění přirozeného ladění podle aktuálně znějící harmonie. Této funkci by mělo předcházet rozřídění tónů na melodické a harmonické, které může proběhnout ručně nebo automaticky.

C. Podpůrné funkce – automatická klasifikace melodických tónů

Automatická klasifikace melodických tónů probíhá na základě zvolených parametrů. Tyto parametry si uživatel může stanovit dle svého uvážení, nebo může použít algoritmus učení na základě vzorku dat. Ten na základě vzorové skladby najde suboptimální hodnoty parametrů pro daný typ skladby. Tyto parametry by měly s minimálním počtem chyb klasifikovat melodické tóny ve skladbách podobného typu.

Jednotlivé způsoby použití programu znázorňuje následující obrázek.

Obr. 11: Způsoby použití programu „Přelad“



4.2. Manuální nastavení ladění

4.2.1. Výběr předem nastaveného ladění

Tento způsob umožňuje přeladění na přednastavené druhy ladění. Uživatel si vybere druh ladění a v dlaždicovém ladění vybere základní tón tóniny.

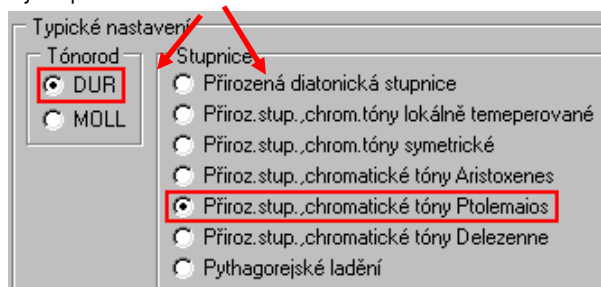
Program nabízí následující druhy ladění (podrobněji viz kapitola 2: Hudební ladění):

- Pythagorejské ladění
- Přirozené ladění – Aristoxenes
- Přirozené ladění – Ptolemaios
- Přirozené ladění – Delezenne
- Přirozené ladění – chromatické tóny symetrické pro dur i moll

- Přirozené ladění – pouze diatonická stupnice
(chromatické tóny jsou naladěny na standardní výšku MIDI tónu)
- Přirozené ladění – chromatické tóny lokálně temperované
(chromatické tóny jsou rovnoměrně temperovány mezi sousední diatonické tóny v přirozené diatonické stupnici dur nebo moll)

Obr. 12: Výběr přednastaveného ladění a základního tónu:

Výběr přednastaveného ladění



Výběr základního tónu

366,67 -16fis	275,0 -14dis	412,50 -12gis	309,37 -10dis	232,3 -8 ais	348,5 -6 eis
293,33 -2 d	440,0 0 a	330,0 2 e	247,50 4 h	371,25 6 fis	278,43 8 cis
234,67 12 hes	352,0 14 f	264,0 16 c	396,0 18 g	297,0 20 d	222,75 22 a
375,47 25 ges	281,60 27 des	422,40 29 as	316,80 31 es	237,60 33 hes	356,40 35 f

Frekvence tónů, které si vybereme, se schematicky zobrazí pomocí centového posunu jednotlivých tónů oproti rovnoměrnému temperovanému ladění. V následujícím schématu centového posunu je použita přesnost v řádu jednoho centu, proto jsou některá desetinná místa frekvencí odlišná od dlaždicového schématu.

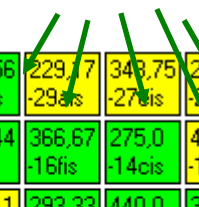
Obr. 13: Zobrazení centového posunu oproti rovnoměrnému temperovanému ladění

	-100 cent	+100 cent	Hz
c	◀	▶	16 264,05
des	◀	▶	27 281,54
d	◀	▶	20 297,08
dis	◀	▶	-10 309,34
e	◀	▶	2 330,01
f	◀	▶	14 352,06
fis	◀	▶	6 371,28
g	◀	▶	18 396,09
as	◀	▶	29 422,32
a	◀	▶	0 440,00
hes	◀	▶	33 475,13
h	◀	▶	4 495,03

4.2.2. Vlastní výběr tónů z dlaždicového prostoru

Tento způsob umožňuje vlastní výběr tónů z dlaždicového ladění. Důvodem použití je, že typické nastavení nemusí uživateli vždy vyhovovat. Uživatel má možnost vybrat si jakékoliv tóny z dlaždicového ladění s omezením, že z enharmonických tónů se vždy vybírá pouze jeden z nich.

Obr. 14: Výběr vlastních tónů z dlaždicového ladění



305,56 -31dis	229,77 -29ais	343,75 -27dis	257,81 -25his	386,72 -23fisis	290,3 -22cisis
244,44 -18h	366,67 -16fis	275,0 -14cis	412,50 -12gis	309,37 -10dis	232,3 -8ais
391,11 -4g	293,33 -2d	440,0 0a	330,0 2e	247,50 4h	371,25 6fis
312,89 10es	234,67 12hes	352,0 14f	264,0 16c	396,0 18g	297,0 20d
250,31 23ces	375,47 25ges	281,60 27des	422,40 29as	316,80 31es	237,60 33hes

Frekvence tónů, které si vybereme, se opět schematicky zobrazí pomocí centového posunu jednotlivých tónů oproti rovnoměrnému temperovanému ladění.

Obr. 15: Zobrazení centového posunu oproti rovnoměrnému temperovanému ladění

	-100 cent	+100 cent	Hz
c			16 264,05
cis			-14 274,95
d			-2 293,33
dis			-31 305,61
e			2 330,01
f			14 352,06
fis			-16 366,59
g			18 396,09
as			29 422,32
a			0 440,00
hes			12 469,41
h			-18 488,77

4.2.3. Vlastní nastavení centového posunu tónů

Nejobecnějším nastavením je možnost vlastního centového posunu oproti rovnoměrnému temperovanému ladění. Toto nastavení se provádí přímo ovládacími prvky v dialogovém okně ladění.

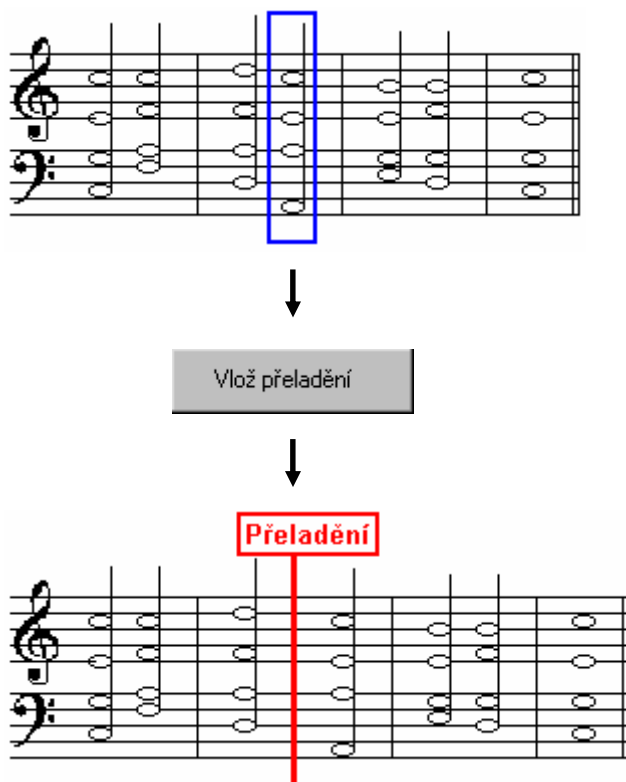
Obr. 16: Vlastní nastavení centového posunu tónů

	-100 cent	+100 cent	Hz
c		-4	261,02
cis		14	279,43
d		-2	293,33
dis		-8	309,69
e		-24	325,09
f		18	352,88
fis		-10	367,86
g		10	394,27
as		10	417,71
a		-2	439,49
hes		2	466,70
h		-12	490,47

4.2.4. Vložení zvoleného přeladění do skladby

At' už uživatel zvolí jakýkoliv postup při volbě ladění, postup při vkládání ladění do skladby je vždy stejný – přeladění je možné vložit na jakékoliv místo skladby.

Obr. 17: Vložení přeladění do skladby



4.3. Automatické nastavení ladění

Při automatickém nastavení ladění bude probíhat analýza skladby podle aktuálně znějící harmonie. Aby rozpoznávání mohlo být korektní, je potřeba nejprve určit melodické a harmonické tóny. To je možné provést automaticky na základě volitelných kritérií, nebo ručně určit vlastní volbu melodických tónů. Poté může proběhnout harmonická analýza skladby a odhad vhodných přirozených ladění.

4.3.1. Vlastní klasifikace harmonických a melodických tónů

Vlastní klasifikace zde znamená ruční stanovení melodických tónů. Pomocí myši je možné označit tóny jako melodické, neoznačené tóny budou považovány za harmonické. To může být u obsáhlých skladeb velmi pracné, ale umožňuje to jakékoliv rozlišení harmonických a melodických tónů.

Obr. 18: Ruční nastavení melodických tónů
(melodické tóny jsou uživatelem označeny červeně)



4.3.2. Automatická klasifikace harmonických a melodických tónů

Automatická klasifikace umožňuje rozpoznávání harmonických a melodických tónů na základě uživatelem stanovených kritérií.

U jednotlivých kritérií si uživatel stanoví *váhy*. Dále stanoví *prahovou hodnotu*, neboli mez, jež stanovuje hranici mezi melodickými a harmonickými tóny.

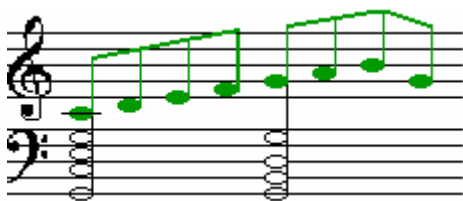
Kritéria jsou rozdělena do dvou druhů – kritéria melodičnosti (kladná) a kritéria harmoničnosti (záporná). Součtem vah jednotlivých kritérií pro každý tón bude klasifikační hodnota. Tóny, které budou klasifikovány vyšší hodnotou, než je práh, budou označeny jako melodické, ostatní tóny nebudou označeny a budou tedy považovány za harmonické.

Obr. 19: Příklad nastavení vah kritérií



Váha příznaků	
Příznaky MELODIČNOSTI	Příznaky HARMONIČNOSTI
Tvorba sekundy(harmonie)	Těžká doba..
Soprán-Tón vyšší než...	Bas-Tón nižší než...
Soprán-Nejvyšší tón	Bas-Nejnižší tón
Samostatnost	Nesamostatnost
Tvorba sekundy(melodie)	Dlouhý tón-delší než...
Kanály melodických tónů	Kanály harmonických tónů
Čísla kanálů melod.tónů	Čísla kanálů harmon.tónů
Práh pro označení tónu jako melodického: 60	

Obr. 20: Příklad výsledku automatické klasifikace melodických tónů (automaticky klasifikované melodické tóny je označeny zeleně)

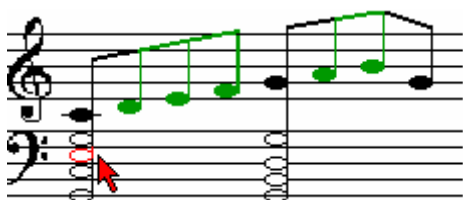


4.3.3. Kombinace automatické a ruční klasifikace melodických tónů

Protože automatická klasifikace melodických tónů nemusí uživateli vždy vyhovovat, je možné její výsledky ručně upravovat. Je možné u některých tónů klasifikovaných jako melodické zrušit označení a tím je prohlásit za harmonické.

Naopak je možné také označení dalších melodických tónů.

Obr. 21: Ruční oprava výsledku automatické klasifikace melodických tónů (automaticky klasifikované melodické tóny je označeny zeleně, uživatelem označené melodické tóny jsou označeny červeně)



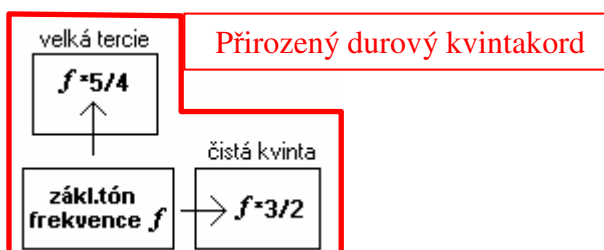
4.3.4. Automatická harmonická analýza skladby

Před touto funkcí by měla být provedena klasifikace harmonických a melodických tónů, protože harmonická analýza skladby bere v úvahu pouze harmonické tóny.

Algoritmus je založen na myšlence hledání vhodného přirozeného ladění pro daný souzvuk tónů. Pokud v určitý okamžik zní durový nebo mollový kvintakord, je použito takové ladění, aby tóny tohoto kvintakordu vytvářely čisté přirozené intervaly.

Jinými slovy: Cílem je, aby z dlaždicového prostoru byly použity sousední tóny, a to takto:

Obr. 22: Durový kvintakord v dlaždicovém prostoru



Obr. 23: Mollový kvintakord v dlaždicovém prostoru



Algoritmus prochází jednotlivé tóny a na základě souzvuku zjišťuje, zda tóny akordu jsou v dlaždicovém prostoru v odpovídajícím vztahu.

Pokud jsou požadované vztahy splněny, ladění zůstává beze změny – přirozené intervaly jsou čisté. Pokud ne, je zkoumána možnost změny ladění.

Možná ladění jsou pro vybírání z dlaždicového prostoru jako symetrická přirozená ladění (viz kapitola 2.2.3.: Přirozená ladění) s různými základními tóny. Změna ladění tedy znamená změnu základního tónu symetrického přirozeného ladění.

Pokud je nalezen základní tón, který požadované vztahy splňuje, vloží program na dané místo skladby změnu ladění. Algoritmus pro harmonickou analýzu skladby je podrobně popsán v kapitole č.6.

Poznámka: Uvedený postup hledání vhodného ladění není jediným možným a nemusí dokonce ani vždy být vhodný. Bylo by možné postupovat i jinak, například určovat každý tón zvlášť, nebo různě ladit harmonické a melodické tóny. Rozbor těchto a dalších postupů již ale přesahuje rámec této práce. V dalším textu budeme nadále pracovat pouze s výše popsaným způsobem změny ladění – změnou základního tónu symetrického přirozeného ladění.

Uživatel má možnost tato přeladění korigovat, a to tak, že po provedení automatické analýzy může některá přeladění odstranit, editovat, nebo naopak vložit další přeladění dle svého uvážení.

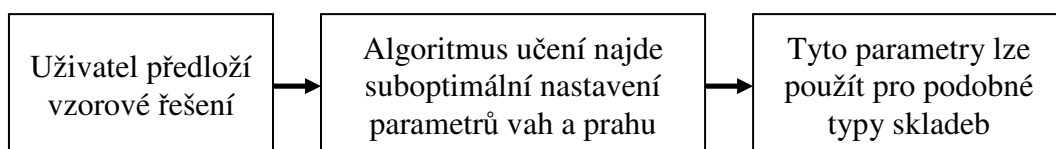
4.4. Postup při učení parametrů harmonických a melodických tónů

Pro nalezení suboptimálního nastavení vah kritérií a prahu pro klasifikaci melodických tónů používáme principu učení s učitelem.

Ve vzorové skladbě označí uživatel melodické tóny. Na základě tohoto vzoru se spustí genetický algoritmus, který najde suboptimální nastavení parametrů tak, aby klasifikace provedená na základě těchto parametrů dostatečně dobře odpovídala vzorovému označení melodických tónů.

Nalezené parametry by měly s minimálním počtem chyb klasifikovat melodické tóny ve skladbách podobného typu.

Obr. 24: Aplikace principu učení s učitelem



Obr. 25: Předložené vzorové řešení (uživatel označí melodické tóny červeně stejně jako při ručním nastavení melodických tónů)



Obr. 26: Možný příklad výsledného nastavení parametrů

Váha příznaků

Příznaky MELODIČNOSTI	Příznaky HARMONIČNOSTI
Tvorba sekundy(harmonie)	První doba taktu
106	-21
Soprán-Tón vyšší než...	Bas-Tón nižší než...
5	-11
Soprán-Nejvyšší tón	Bas-Nejnižší tón
90	-85
Samostatnost	Nesamostatnost
32	-39
Tvorba sekundy(melodie)	Dlouhý tón-delší než...
80	-74
Kanály melodických tónů	Kanály harmonických tónů
0	0
Čísla kanálů melod.tónů	Čísla kanálů harmon.tónů
0 4 8 12	0 4 8 12
1 5 9 13	1 5 9 13
2 6 10 14	2 6 10 14
3 7 11 15	3 7 11 15
Práh pro označení tónu jako melodického:	
142	

Podrobněji je implementace algoritmu učení pro stanovení vah melodických tónů popsána v kapitole 6 (Algoritmus učení stanovení vah melodických tónů).

5. Algoritmus automatické harmonické analýzy skladby

Automatická harmonická analýza skladby vychází z myšlenky, že tóny akordu znějí nejlépe, pokud tvoří přirozené intervaly, tj. pokud frekvence těchto tónů jsou v poměru malých celých čísel.

Odhlédněme nyní od toho, že uvedené tvrzení není absolutní pravdou – opět se totiž dostáváme na tenký led subjektivního vnímání hudby. Popsaný cíl totiž vůbec nemusí být hudebníkem požadován, pro některé skladby je dokonce jakákoliv aplikace přirozeného ladění nepoužitelná.

Proto se tedy dále budeme zabývat případy, kdy **chceme** dosáhnout přirozených intervalů v akordech.

Budeme používat přirozené intervaly kvinty, velké tercie, malé tercie a jejich obraty. Základním požadavkem bude, aby durový kvintakord byl složen z velké a malé tercie; obdobně budeme požadovat malou a velkou tercii u mollového kvintakordu.

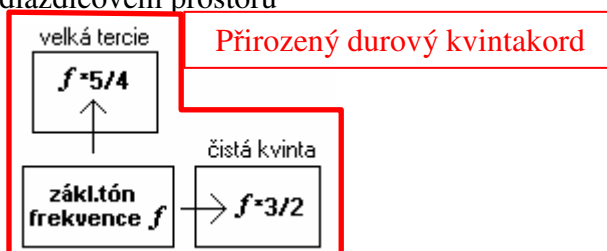
Uveďme si jako příklad symetrické přirozené ladění se základním tónem *c*:

Obr. 27: Symetrické přirozené ladění se základním tónem *c*

244,44 h	366,67 fis	275,0 cis	412,50 gis	309,37 dis	232,3 ais	348,5 eis	261,3 his
391,11 g	293,33 d	440,0 a	330,0 e	247,50 h	371,25 fis	278,43 cis	417,65 gis
312,89 es	234,67 hes	352,0 f	264,0 c	396,0 g	297,0 d	222,75 a	334,12 e
250,31 ces	375,47 ges	281,60 des	422,40 as	316,80 es	237,60 hes	356,40 f	267,30 c
400,50 asas	300,37 eses	225,28 heses	337,92 fes	253,44 ces	380,16 ges	285,12 des	427,68 as

Durové a mollové přirozené kvintakordy jsou v dlaždicovém prostoru vyjádřeny následujícími polohami:

Obr. 28: Durový kvintakord v dlaždicovém prostoru



Obr. 29: Mollový kvintakord v dlaždicovém prostoru



Vidíme, že v symetrickém přirozeném ladění můžeme pomocí aktivních tónů v dlaždicovém prostoru (tj. na obrázku označených zeleně a červeně) vytvořit 6 přirozených durových kvintakordů (f-a-c, c-e-g, g-h-d, des-f-as, as-c-es, es-g-b) a 6 mollových (a-c-e, e-g-h, h-d-fis, f-as-c, c-es-g, g-b-d).

Nejjednodušší případ nastane, když skladba obsahuje pouze akordy, které jsou přirozené v rámci jednoho naladění. Pokud se ve skladbě jiné akordy nenacházejí, je toto ladění možné použít pro celou skladbu a všechny akordy budou obsahovat přirozené intervaly.

Problém nastává, pokud skladba obsahuje i jiné akordy. Například mollový kvintakord d-f-a v tomto ladění neobsahuje přirozené intervaly – pro tón *d* je použita frekvence 297 Hz, ale pro dosažení přirozených intervalů s tóny *f* a *a* by bylo nutné použít sousedního tónu *d* s frekvencí 293,33 Hz.

Abychom žádaného přirozeného ladění dosáhli, budeme v průběhu skladby ladění dynamicky měnit. Změnou pochopitelně přirozené intervaly u některých kvintakordů ztratíme, u jiných je ale naopak získáme.

Poznámka: Při použití symetrického přirozeného ladění bude vždy 6 durových a 6 mollových kvintakordů přirozených.

Kroky automatické harmonické analýzy skladby tedy budou následující:

- A. roztrždit tóny na melodické a harmonické a dále pracovat pouze s harmonickými (viz kapitola 6.1.);
- B. u každého souzvuku stanovit, zda je založen na durovém nebo mollovém kvintakordu. Na základě toho určit **požadovaný** vztah tónů v dlaždicovém prostoru (viz kapitola 6.2.);
- C. pokud je požadovaný vztah splněn, ladění se nemění. Pokud požadovaný vztah není splněn, zvolit takové ladění, aby akord obsahoval přirozené intervaly. Nové ladění by mělo být co nejbližší předchozímu ladění, aby nedocházelo ke znatelným skokům výšek tónů (viz kapitola 6.3.).

Tyto kroky si nyní popíšeme.

5.1. Krok A: Rozdělení tónů na melodické a harmonické

Jak jsme si již popsali v kapitole 5, můžeme provést rozdělení tónů na harmonické a melodické buď ručně, nebo použít automatickou klasifikaci. Označení ručním i automatickým způsobem se sice z důvodu přehlednosti zobrazují různě (ruční červeně, automatické zeleně), ale co se týče zařazení tónů do skupin harmonických a melodických, jsou oba způsoby označení melodických tónů ekvivalentní.

Po označení melodických tónů zůstávají harmonické tóny zobrazeny černě. S těmito tóny budeme dále pracovat.

5.1.1. Aktuálně znějící harmonie

Abychom mohli pro každý tón určit, jaké další tóny ve stejném okamžiku znějí, je pro každý tón automaticky zapamatována jeho aktuálně znějící harmonie – to znamená tóny, které s ním současně znějí v jednom okamžiku. Tuto harmonii si v programu můžeme zobrazit pomocí zaškrtávacích tlačítek *Zobrazit harmonii v horní osnově* a *Zobrazit harmonii v dolní osnově*. Tóny, které byly stisknuty dříve a ještě znějí, jsou zobrazeny světle modrou barvou.

Obr. 30: zobrazení harmonie v horní osnově



Obr. 31: zobrazení harmonie v dolní osnově



Poznámka: v horní osnově obsahující houslový a basový klíč je zobrazen standardní notový zápis. V dolní osnově jsou všechny tóny zobrazeny v rámci jednočárkované oktávy.

Aktuálně znějící harmonie je v programu definována jako bitové pole o dvanácti prvcích, které pro každý tón udává, které tóny chromatické stupnice (v rámci jedné oktávy) jsou aktuálně stisknuty a které ne.

Zdrojový kód: definice aktuálně znějící harmonie v programu

```
type TZjedenHarmonie=array[0..11] of boolean;
{udává které tóny znějí - u každého tónu true/false}
```

Obr. 32: aktuálně znějící harmonie pro jednotlivé souzvuky



Tab. 2: příklad dat aktuálně znějící harmonie pro jednotlivé souzvuky

Tón	c	cis/des	d	die/es	e	f	fis/ges	g	gis/as	a	ais/b	h
Číslo	0	1	2	3	4	5	6	7	8	9	10	11
1.souzvuk	true	false	false	false	true	false	false	true	false	false	false	false
2.souzvuk	true	false	true	false	true	false	false	true	false	false	false	false
3.souzvuk	true	false	false	false	true	false	false	true	false	false	false	false
4.souzvuk	true	false	false	false	true	true	false	true	false	false	false	false

5.2. Krok B: Rozpoznání typu akordu a umístění tónů v dlaždicovém prostoru

Na základě informací o aktuálně znějící harmonii probíhá rozpoznávání typu akordu a to postupně pro každý tón, který byl klasifikován jako harmonický (tj. nebyl označen jako melodický).

5.2.1. Pravidla pro rozpoznání typu akordu

Informace pro rozpoznávání typu akordu a požadovaných vztahů v dlaždicovém prostoru jsou uloženy v datovém souboru [pravidla.dat](#). Tento soubor obsahuje pravidla pro rozpoznávání různých typů akordů a jejich požadované umístění.

Typ akordu je rozpoznáván podle relativních vztahů aktuálně znějících tónů v chromatické stupnici. Není tedy nutno definovat zvlášť tóny akordů v různých tóninách, ale pouze relativní polohy tónů akordu – tento popis je univerzální pro všechny tóniny.

Formát souboru *pravidla.dat* je textový. Soubor obsahuje postupně jednotlivá pravidla, která jsou automaticky načtena po spuštění programu *Přelad*. Pravidla jsou načítána postupně, dokud program nenarazí na konec souboru. Každé pravidlo má následující tvar:

Na prvním řádku každého pravidla je číslo udávající počet tónů souzvuku. Na druhém řádku pravidla jsou relativní polohy těchto tónů v chromatické stupnici. Prvním z nich je základní tón akordu, který je označen polohou 0. Následující čísla udávají relativní polohy dalších tónů oproti základnímu tónu. Čísla jsou zapsána v rámci jedné oktávy a mohou nabývat hodnot 1 až 11 – znamená to rozdíl chromatické stupnice.

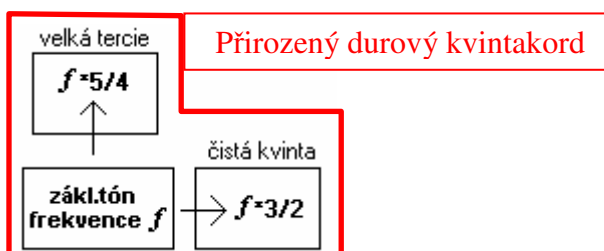
Dále následuje pro každý tón souzvuku řádek obsahující dvě čísla, a to relativní polohu řádku a sloupce v dlaždicovém prostoru oproti základnímu tónu. Pro základní tón musí být tato dvě čísla nulová. Pro další tóny mohou tyto hodnoty nabývat kladných i záporných hodnot.

Formát souboru pravidla.dat si ukážeme na příkladu definice durového a mollového kvintakordu.

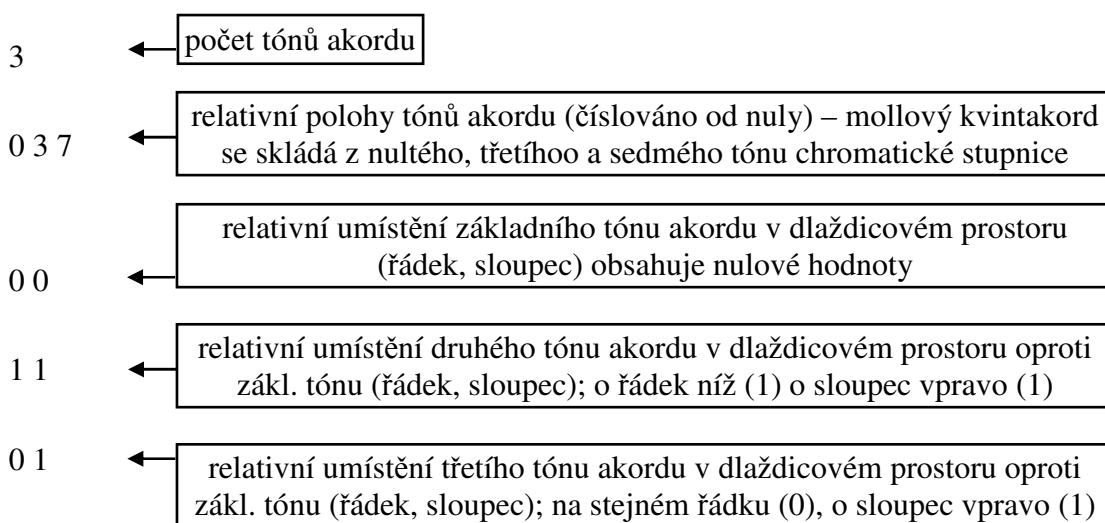
Příklad: pravidlo pro durový kvintakord v souboru *pravidla.dat*

3	←	počet tónů akordu
0 4 7	←	relativní polohy tónů akordu (číslováno od nuly) – durový kvintakord se skládá z nultého, čtvrtého a sedmého tónu chromatické stupnice
0 0	←	relativní umístění základního tónu akordu v dlaždicovém prostoru (řádek, sloupec) obsahuje nulové hodnoty
-1 0	←	relativní umístění druhého tónu akordu v dlaždicovém prostoru oproti zákl. tónu (řádek, sloupec); o řádek výš (-1) ve stejném sloupci (0)
0 1	←	relativní umístění třetího tónu akordu v dlaždicovém prostoru oproti zákl. tónu (řádek, sloupec); na stejném řádku (0), o sloupec vpravo (1)

Obr. 33: Durový kvintakord v dlaždicovém prostoru



Příklad: pravidlo pro mollový kvintakord v souboru *pravidla.dat*



Obr. 34: Mollový kvintakord v dlaždicovém prostoru



Poznámka: Podobně mohou být definována např. pravidla pro septakordy a zmenšené akordy. U septakordu je nutné si vybrat, který tón z dlaždicového prostoru chceme použít. Například u dominantního septakordu g-h-d-f můžeme jako tón f zvolit buď jako malou tercii nad tónem d (který je ale cizí v tónině c dur a muselo by dojít k přeladění) nebo jako subdominantu v tónině c dur, která je navíc bližší sedmé harmonické tónu g.

5.2.2. Porovnání aktuálně znějící harmonie s pravidly

Pro každý harmonický tón algoritmus porovnává, zda aktuálně znějící harmonie (viz kapitola 5.1.1.) odpovídá některému z použitých pravidel.

Algoritmus prochází všechny harmonické tóny. Pro každý z nich prochází všechna pravidla a hledá s nimi shodu.

Aby porovnání bylo univerzální pro každou tóninu, je pravidlo při hledání shody postupně přikládáno na všechny možné pozice tónů v rámci oktávy.

Postup při přikládání pravidla ke všem tónům v rámci oktávy:

Jak je popsáno v kapitole 5.1.1. v tab. 2, můžeme aktuálně znějící harmonii vyjádřit následujícím bitovým polem:

Tab. 3: Bitové pole aktuálně znějící harmonie v rámci jedné oktávy

Tón	c	cis/des	d	die/es	e	f	...	h
Číslo	0	1	2	3	4	5	...	11
Harmonie	true/false	true/false	true/false	true/false	true/false	true/false	...	true/false

Stejným bitovým polem můžeme popsat také pravidlo (viz kapitola 5.2.1.), přičemž polohy tónů jsou relativní vzhledem k základnímu tónu. Ten je vždy definován na pozici 0:

Tab. 4: Vyjádření pravidla pomocí bitového pole

Číslo tónu pravidla	0	1	2	3	4	5	...	11
Tón je součástí pravidla	true (základní tón)	true/false	true/false	true/false	true/false	true/false	...	true/false

Pravidlo je postupně přikládáno k aktuálně znějící harmonii, přičemž při posunu pravidla vpravo je číslo tónu pravidla zvýšeno o 1 a je na ně provedena operace modulo 12. Díky tomu je posloupnost tónů pravidla cyklická a porovnání je univerzální pro všechny tóniny. Hledáme tedy absolutní pozici základního tónu v aktuálně znějící harmonii.

Tab. 5: Schéma postupného přikládání pravidla k aktuálně znějící harmonii

Har-monie	Tón	c	cis/des	d	die/es	e	f	fis/ges	g	gis/as	a	ais/b	h
	Číslo	0	1	2	3	4	5	6	7	8	9	10	11
	Harmonie	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F
První pokus	Tón pravidla	0	1	2	3	4	5	6	7	8	9	10	11
	Je součástí pravidla	true (zákl.)	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F
Druhý pokus	Tón pravidla	11	0	1	2	3	4	5	6	7	8	9	10
	Je součástí pravidla	T/F	true (zákl.)	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F
Třetí pokus	Tón pravidla	10	11	0	1	2	3	4	5	6	7	8	9
	Je součástí pravidla	T/F	T/F	true (zákl.)	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F
...													
Dvanáctý pokus	Tón pravidla	1	2	3	4	5	6	7	8	9	10	11	0
	Je součástí pravidla	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	T/F	true (zákl.)

Algoritmus postupného přikládání pravidla k aktuálně znějící harmonii je implementován v unitě AutoDynamLadeni.pas a jeho schematický zápis vypadá takto:

```
var ZjedenLadeni:array[0..11] of boolean; {vstup.data: akt.znějící harmonie}
SouzvukPravidla:array[0..11] of boolean; {vstup.data-tóny pravidla}
PocetTonuPravidla:integer; {vstupní data-počet TRUE hodnot v pravidle}
ZaklTonSouzvuku:integer; {pom.proměnná-průchod všemi pokusy}
j:integer; {pom.proměnná-pro ověření shody tónů}
shoda:integer; {pomocná proměnná pro výpočet
                počtu tónů pravidla obsažených v souzvuku}

begin
  for ZaklTonSouzvuku:=0 to 11 do begin {zkouší přikládat šablonu na
                                         všech 12 pozic tónů v oktávě}

    shoda:=0;
    for j:=0 to 11 do {kontrola všech dvanácti tónů}
      if (ZjedenLadeni[j] = true) and
        (SouzvukPravidla[ (j-ZaklTonSouzvuku+12) mod 12] = true)
      then shoda:=shoda+1;

      if shoda=PocetTonuPravidla then begin
        ShowMessage('Shoda na tónu '+IntToStr(ZaklTonSouzvuku));
        Break; {Nalezena shoda akt.znějící harmonii s pravidlem}
      end;
    end; {end for}
  end;
end;
```

Pokud je po skončení tohoto algoritmu hodnota proměnné *shoda* rovna hodnotě proměnné *PocetTonuPravidla*, pak byla nalezena shoda aktuálně znějící harmonie s pravidlem, přičemž základním tónem souzvuku je hodnota *ZaklTonSouzvuku*.

Vidíme, že pravidlo je splněno vždy, když aktuálně znějící harmonie obsahuje všechny tóny pravidla. Není ale vyloučeno, že aktuálně znějící harmonie obsahuje také další tóny.

Podmínka, že ke každému tónu pravidla zní tento tón v aktuální harmonii, je tedy podmínkou nutnou i postačující.

Opačná podmínka neplatí – ne nutně každý tón aktuálně znějící harmonie musí být obsažen v pravidle, harmonie může obsahovat více tónů než pravidlo.

Může se stát, že pravidlo bude vyhovovat více základním tónům. V takovém případě bude vybrán základní tón s nižším číslem – jakmile poprvé dojde ke shodě souzvuku, je proveden příkaz Break a další možnosti základních tónů se nezkoušejí.

Například v následujícím příkladě by pravidlo durového akordu mohlo být přiloženo k tónu *c* nebo *g*, ale protože ke shodě dojde již u tónu *c*, bude pravidlo splněno a tón *c* bude označen jako nalezený základní tón souzvuku.

Obr. 35: Víceznačný základní tón pro jedno pravidlo



Tab. 6: Bitové pole harmonie dle obr. 30

Harmonie	Tón	c	cis/des	d	die/es	e	f	fis/ges	g	gis/as	a	ais/b	h
	Číslo	0	1	2	3	4	5	6	7	8	9	10	11
	Harmonie	true	false	true	false	true	false	false	true	false	false	false	true

Může se také stát, že některý souzvuk bude vyhovovat více pravidlům. Pravidla jsou aplikována postupně, a po prvním nalezení shody je další prohledávání ukončeno. Protože pravidla jsou při načítání postupně ukládána za sebe, musí být důležitější pravidla v souboru *pravidla.dat* uvedena na přednějších místech.

Např. pokud bychom definovali pravidlo pro septakord, musí být v souboru *pravidla.dat* umístěno dříve, než pravidla pro durový a mollový kvintakord. Kvintakordy jsou totiž součástí septakordů a pokud by byla pravidla pro durový a mollový kvintakord prohledávána dříve, byla by nalezena shoda a pravidlo pro septakord by vůbec nebylo aplikováno.

Obr. 36: Aplikace algoritmu hledání akordu pro mollový septakord g7



Tab. 7: Bitové pole harmonie dle obr. 31

Harmonie	Tón	c	cis/des	d	die/es	e	f	fis/ges	g	gis/as	a	ais/b	h
	Číslo	0	1	2	3	4	5	6	7	8	9	10	11
	Harmonie	false	false	true	false	false	true	false	true	false	false	true	false

V případě, že budeme mít definováno pravidlo pro septakord, bude nalezena shoda při základním tónu *g*. Pokud nebudeme mít definováno pravidlo pro septakord a dříve budeme mít definováno pravidlo pro durový kvintakord, bude nalezena shoda s durovým kvintakordem při základním tónu *b*. Pokud nebudeme mít definováno pravidlo pro septakord a dříve bude definováno pravidlo pro mollový kvintakord, bude s ním nalezena shoda při základním tónu *g*.

5.3. Krok C: Změna ladění

Předchozí krok nám zjistil, zda aktuálně znějící harmonie odpovídá některému pravidlu, a pokud ano, tak s jakým základním tónem.

Pokud nebyla zjištěna shoda s žádným pravidlem, program nemění ladění – není jisté, jak by taková změna měla proběhnout.

Pokud byla zjištěna shoda s některým pravidlem na určitém základním tónu, musíme ověřit, zda jsou splněny požadované vztahy v dlaždicovém prostoru definované pravidlem.

Pokud zjistíme, že všechny tóny pravidla v požadovaném vztahu jsou v dlaždicovém prostoru aktivní, nedochází ke změně ladění, protože požadované přirozené intervaly budou znít správně.

Pokud některý z tónů pravidla nebude v požadovaném vztahu v dlaždicovém prostoru aktivní, budeme zkoušet možné změny ladění.

Popis těchto změn je definován v souboru *zmenaladeni.dat*. Soubor obsahuje postupně návrhy na změnu základního tónu symetrického přirozeného ladění.

Tyto změny symetrického přirozeného ladění jsou postupně zkoušeny a pro každý pokus je znovu ověřeno, zda jsou tóny pravidla v požadovaném vztahu v dlaždicovém prostoru aktivní. Pokud je při některém pokusu změny ladění dosaženo shody, je tato změna aplikována a nalezené přeladění je vloženo do skladby. Je tím zajištěno, že přirozené intervaly budou znít správně.

5.3.1. Pořadí pokusů změn ladění

Je žádoucí, aby prováděná změna byla co nejmenší. Jako úplně první ladění skladby je navrženo přirozené symetrické ladění se základním tónem podle tóniny dané předznamenáním.

Pořadí návrhů na změnu ladění je dáno obsahem souboru [zmenaladeni.dat](#). Formát tohoto souboru je textový; soubor obsahuje následující data:

Na prvním řádku je uveden počet pokusů změny ladění. Dále následuje tolik řádků, kolik je těchto pokusů.

Každý z těchto řádků obsahují vždy dvě čísla udávající navrhovanou změnu základního tónu: první z nich udává navrhovaný posun aktuálního naladění v řádcích a druhé číslo navrhovaný posun ve sloupcích. Čísla mohou nabývat kladných i záporných hodnot.

První pokus musí mít hodnoty 0 0 – protože jako první je vždy zkoušena možnost zachování stávajícího ladění; bylo by zbytečné provádět změnu, když by současné ladění vyhovovalo.

U posledního pokusu je nastavení hodnot 0 0 doporučeno – pokud jsme nenalezli žádné ladění, kde aktuálně znějící tóny vyhovují pravidlu, neměli bychom ladění měnit – znamenalo by to pouze změnu z jednoho nevyhovujícího ladění do jiného, taktéž nevyhovujícího.

Příklad: Obsah souboru [zmenaladeni.dat](#)

```
13          //počet pokusů
0 0         //1.pokus-zachovat současné ladění
0 -1        //2.pokus-posun v dlažd.prostoru o 1 doleva-na subdominantu
0 1         //3.pokus-posun v dlažd.prostoru o 1 doprava-na dominantu
0 -2        //4.pokus-na subdominantu subdominanty
0 2         //5.pokus-na dominantu dominanty
-1 0        //6.pokus-o velkou tercii výš
1 0         //7.pokus-o velkou tercii níž
1 1         //8.pokus-o malou tercii výš
-1 -1       //9.pokus-o malou tercii níž
-1 1        //10.pokus-o půltón výš
1 -1        //11.pokus-o půltón níž
-1 2        //12.pokus-o zmenšenou kvintu výš
0 0         //13.pokud žádný z předchozích pokusů nevyjde,vrátíme se zpět
```

5.3.2. Příklad změn ladění pro konkrétní skladbu

Následující příklad ukazuje praktický výsledek použití algoritmu pro následující kadenci:

Obr. 37: Příklad kadence pro harmonickou analýzu



Obr. 38: Výsledek harmonické analýzy – automatický návrh přirozeného ladění



Protože skladba je bez předznámenání, je jako první zvoleno přirozené ladění *c*. Toto ladění vyhovuje pro první a druhý souzvuk – v prvním je rozpoznán durový kvintakord *c-e-g* a v druhém durový kvintakord *f-a-c*. Tyto tóny mají v dlaždicovém prostoru pro symetrické přirozené ladění *c* správné vztahy:

Obr. 39: Přirozené symetrické ladění se základním tónem *c*

244,44 -18h	366,67 -16fis	275,0 -14cis	412,50 -12gis	309,37 -10dis	232,3 -8 ais	348,5 -6 eis	261,3 -4 his
391,11 -4 g	293,33 -2 d	440,0 0 a	330,0 2 e	247,50 4 h	371,25 6 fis	278,43 8 cis	417,65 10 gis
312,89 10 es	234,67 12 hes	352,0 14 f	264,0 16 c	396,0 18 g	297,0 20 d	222,75 22 a	334,12 23 e
250,31 23 ces	375,47 25 ges	281,60 27 des	422,40 29 as	316,80 31 es	237,60 33 hes	356,40 35 f	267,30 37 c
400,50 37 asas	300,37 39 eses	225,28 41 hese	337,92 43 fes	253,44 45 ces	380,16 47 ges	285,12 49 des	427,68 51 as

O následujícím souzvuku již totéž říci nelze. Je v něm rozpoznán mollový kvintakord *d-f-a*, jehož tóny ale nejsou v dlaždicovém prostoru v požadovaných polohách – tóny *d-f-a* spolu v dlaždicovém prostoru vůbec nesousedí. Dochází tedy k testování změny ladění. Pokud je pořadí pokusů změn ladění jako v 5.3.1., zkusíme nejprve posun o jedno pole doleva:

Obr. 40: Přirozené symetrické ladění se základním tónem *f*

244,44 -18h	366,67 -16fis	275,0 -14cis	412,50 -12gis	309,37 -10dis	232,3 -8 ais	348,5 -6 eis	261,3 -4 his
391,11 -4 g	293,33 -2 d	440,0 0 a	330,0 2 e	247,50 4 h	371,25 6 fis	278,43 8 cis	417,65 10 gis
312,89 10 es	234,67 12 hes	352,0 14 f	264,0 16 c	396,0 18 g	297,0 20 d	222,75 22 a	334,12 23 e
250,31 23 ces	375,47 25 ges	281,60 27 des	422,40 29 as	316,80 31 es	237,60 33 hes	356,40 35 f	267,30 37 c
400,50 37 asas	300,37 39 eses	225,28 41 hese	337,92 43 fes	253,44 45 ces	380,16 47 ges	285,12 49 des	427,68 51 as

Jak vidíme, již v tomto prvním pokusu jsme dosáhli toho, aby tóny mollového kvintakordu *d-f-a* byly v dlaždicovém prostoru v požadovaných polohách. Změna ladění je navíc minimální možná – nedochází tudíž ke znatelným skokům ve výšce tónů. Ještě přesněji – nedochází k žádným skokům ve výšce tónů, protože tóny *d*, *fis* a *b* dosud nezazněly.

Ve čtvrtém souzvuku je rozpoznán mollový kvintakord *a-c-e*. Polohy těchto tónů v dlaždicovém ladění odpovídají požadovaným vztahům, ladění se tedy nemění.

I v pátém souzvuku, kde je rozpoznán mollový kvintakord *e-g-h*, frekvence tónů stále vytvářejí přirozené intervaly.

Jinak tomu ale je u šestého kvintakordu – přirozený kvintakord *g-h-d* již není součástí přirozeného symetrického ladění se základním tónem *f*. Zkoušíme tedy změnu ladění. Ale ani první pokus změny ladění nevede k požadovanému výsledku – pro přirozené symetrické ladění se základním tónem *b* také nejsou tóny kvintakordu *g-h-d* v požadovaných vztazích.

Podle pořadí pokusů z příkladu v kapitole 5.3.1. zkusíme jako další možnost posun o jedno políčko doprava oproti současnému ladění. Tím je návrat zpět na přirozené ladění *c*.

Obr. 41: Přirozené symetrické ladění se základním tónem *c*

244,44 -18h	366,67 -16fis	275,0 -14cis	412,50 -12gis	309,37 -10dis	232,3 -8 ais	348,5 -6 eis	261,3 -4 his
391,11 -4 g	293,33 -2 d	440,0 0 a	330,0 2 e	247,50 4 h	371,25 6 fis	278,43 8 cis	417,65 10 gis
312,89 10 es	234,67 12 hes	352,0 14 f	264,0 16 c	396,0 18 g	297,0 20 d	222,75 22 a	334,12 23 e
250,31 23 ces	375,47 25 ges	281,60 27 des	422,40 29 as	316,80 31 es	237,60 33 hes	356,40 35 f	267,30 37 c
400,50 37 asas	300,37 39 eses	225,28 41 hese	337,92 43 fes	253,44 45 ces	380,16 47 ges	285,12 49 des	427,68 51 as

V tomto ladění již je vztah *g-h-d* splněn. Je tedy vloženo přeladění na symetrické přirozené ladění *c*.

Podívejme se na další, v pořadí již sedmý souzvuk. V něm program rozpozná durový kvintakord *d-fis-a*. Pro tyto tóny opět nejsou splněny požadované vztahy a úspěšný v hledání vhodného ladění bude opět druhý pokus – tedy opět posun o jedno políčko doprava:

Obr. 42: Přirozené symetrické ladění se základním tónem *g*

244,44 -18h	366,67 -16fis	275,0 -14cis	412,50 -12gis	309,37 -10dis	232,3 -8 ais	348,5 -6 eis	261,3 -4 his
391,11 -4 g	293,33 -2 d	440,0 0 a	330,0 2 e	247,50 4 h	371,25 6 fis	278,43 8 cis	417,65 10 gis
312,89 10 es	234,67 12 hes	352,0 14 f	264,0 16 c	396,0 18 g	297,0 20 d	222,75 22 a	334,12 23 e
250,31 23 ces	375,47 25 ges	281,60 27 des	422,40 29 as	316,80 31 es	237,60 33 hes	356,40 35 f	267,30 37 c
400,50 37 asas	300,37 39 eses	225,28 41 hese	337,92 43 fes	253,44 45 ces	380,16 47 ges	285,12 49 des	427,68 51 as

S tímto laděním již vystačíme až do konce – následující mollový kvintakord *d-f-a*, poté následující kvintakord *g-h-d* a závěrečný kvintakord *c-e-g* již v tomto ladění splňují požadované vztahy.

Poznámka: všimněme si, že v průběhu kadence zazněly dva odlišné akordy d-f-a. V tónině c-dur je první z nich je neúplnou subdominantou a druhý z nich je střídavou dominantou. Rozdíl výšky mezi těmito dvěma souzvuky je přibližně 22 centů. Různost těchto akordů zdůrazňuje i Šín ve své Nauce o harmonii [8] v §10 – Mínotonální akordy.

6. Algoritmus učení pro stanovení vah parametrů analýzy melodických tónů

V této kapitole je popsán genetický algoritmus pro stanovení vah parametrů pro analýzu melodických tónů. Algoritmus je implementován v software Přelad'.

Cílem tohoto algoritmu je na základě vzorového řešení najít vhodné nastavení parametrů analýzy melodických a harmonických tónů, které při automatické analýze korektně označí harmonické a melodické tóny. Nalezené nastavení parametrů pak může být použito pro podobné typy skladeb.

Z uživatelského hlediska je tento postup popsán v kapitole 4.4. Uživatel označí melodické tóny ve vzorové skladbě. Tím stanovuje vzorové řešení. Poté uživatel spustí genetický algoritmus učení, který se snaží optimalizovat nastavení parametrů pro vzorové řešení.

6.1. Obecný popis práce genetických algoritmů

Genetické algoritmy jsou heuristickou prohledávací metodou inspirovanou přírodními principy selekce, křížení a mutace. Používají se pro řešení problémů optimalizace, zvláště v případech, kdy je parametrický prostor příliš velký pro prohledávání všech možností.

V následujícím textu si popíšeme obecné charakteristiky genetických algoritmů. Podrobnější popis genetických algoritmů je možné nalézt například v publikaci Teoretické otázky neuronových sítí [7] v kapitole 14.3.

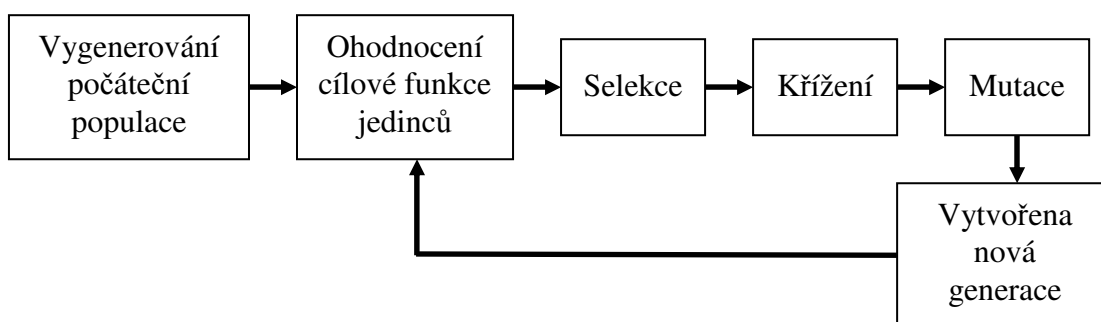
Základním prvkem genetického algoritmu je *jedinec*, který je definován jako zakódované nastavení parametrů. Parametry jedince se nazývají *geny* a typicky jsou kódovány binárním řetězcem. Pro každého jedince lze vypočíst hodnotu *cílové funkce*, která udává kvalitu jedince. Genetický algoritmus pracuje v jednom okamžiku s celou *populací* řešení (populace je definována jako *množina jedinců*).

Populace je na začátku algoritmu typicky vygenerována náhodně. Populace se postupně vyvíjí – jedinci zanikají a naopak vznikají noví. Tento proces je řízen třemi základními operátory:

- operátor *selekce*
Má za úkol provést přírodní výběr – neboli zánik horších jedinců a přežití lepších. Selekcce může být založena na pravděpodobnostních metodách – v tom případě mohou přežít i horší jedinci, ale s nižší pravděpodobností (úměrnou hodnotě jejich cílové funkce).
- operátor *křížení*
Provádí kombinaci dvou existujících jedinců, čímž vznikají noví jedinci. Nový jedinec dědí geny svých rodičů.
- operátor *mutace*
Zajišťuje zachování variability pomocí náhodných změn genů jedinců.

Genetický algoritmus probíhá v nekonečném cyklu. Na populaci jsou postupně aplikovány operátory selekce, křížení a mutace. Po provedení těchto operátorů vzniká nová populace. Říkáme, že postupně vznikají nové *generace* jedinců.

Obr. 43: Schéma práce genetického algoritmu



Algoritmus může být ukončen po nalezení dostatečně kvalitního řešení, po provedení stanoveného počtu cyklů nebo po zásahu uživatele.

Při konkrétní aplikaci genetického algoritmu je nutno definovat jeho následující části:

- genetickou reprezentaci možných řešení (zakódování parametrů)
- způsob vytváření počáteční populace
- výpočet cílové funkce
- genetické operátory vytvářející novou generaci
- parametry (velikost populace, pravděpodobnosti operátorů)

6.2. Genetický algoritmus pro stanovení vah parametrů analýzy melodických tónů

Zde si popíšeme konkrétní aplikaci genetického algoritmu použitou v programu Přelad'. Postupně zde popíšeme všechny výše popsání části genetického algoritmu.

6.2.1. Genetická reprezentace – kódování parametrů

Použité parametry analýzy melodických a harmonických tónů, které byly popsány v kapitole 3, je nutno zakódovat do podoby řetězce genů. Pro zakódování všech parametrů nám bude stačit 21 bytů, přičemž každý byte bude reprezentovat jeden gen. Každý gen tedy může nabývat hodnot 0..255.

V unitě AlgoritmusLadeni.pas je definován typ TJedinec obsahující především následující součásti:

```
Type TJedinec=object
  Value:array[1..21] of byte;    {zakódované parametry - geny}
  PocetChyb:longint;             {cíl.funkce: počet chyb. Cíl:minimalizovat}
end;
```

Většina vah parametrů také nabývá hodnot 0..255, pak je příslušný gen přímo roven hodnotě parametru. Váhy parametrů nabývajících hodnot -255..0 jsou pak v genetickém kódu reprezentovány svou opačnou hodnotou. U parametrů nabývajících jiného rozsahu hodnot budeme provádět překódování tak, abychom opět dosáhli hodnot genu 0..255.

Způsob kódování parametrů popisuje následující tabulka:

Tab. 8: Kódování parametrů analýzy melodických tónů do řetězce bytů

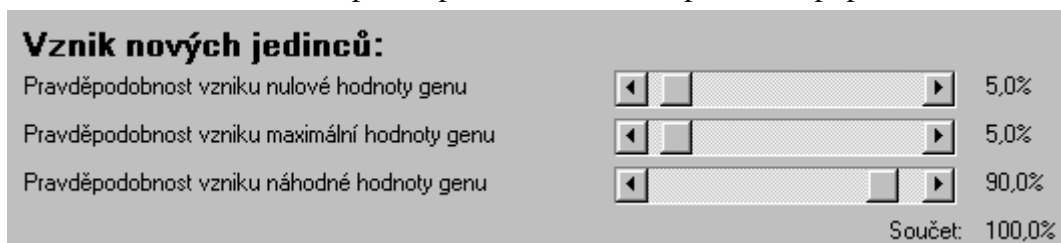
Pořadí Bytu	Kódovaný parametr	Hodnoty váhy / parametru	Způsob dekodování genu do parametru
1.	tvorba sekundy v harmonii	0..255	$\text{váha} := \text{gen}$
2.	soprán – tón vyšší než ..	0.255	$\text{váha} := \text{gen}$
3.	parametr tónu pro soprán – tón vyšší než ..	0..127	$\text{parametr} := \text{gen} \text{ div } 2$
4.	soprán – nejvyšší tón	0.255	$\text{váha} := \text{gen}$
5.	samostatnost	0..255	$\text{váha} := \text{gen}$
6.	tvorba sekundy v melodii	0..255	$\text{váha} := \text{gen}$
7.	těžká doba	-255..0	$\text{váha} := -\text{gen}$
8.	parametr – počet těžkých dob v taktu	1..8	$\text{parametr} := (\text{gen} \text{ div } 64) + 1$
9.	bas – tón nižší než ..	-255..0	$\text{váha} := -\text{gen}$
10.	parametr tónu pro bas – tón nižší než ..	0..127	$\text{parametr} := \text{gen} \text{ div } 2$
11.	bas – nejnižší tón	-255..0	$\text{váha} := -\text{gen}$
12.	nesamostatnost	-255..0	$\text{váha} := -\text{gen}$
13.	dlouhý tón	-255..0	$\text{váha} := -\text{gen}$
14.	parametr dlouhého tónu – počet dvaatřicetin	1..128	$\text{parametr} := (\text{gen} \text{ div } 2) + 1$
15.	kanály melod.tónů	0..255	$\text{váha} := \text{gen}$
16.,17	parametr – melodické kanály	16 bitů	jednotlivé bity reprezentují zaškrtnutí 16 kanálů
18.	kanály harmon.tónů	-255..0	$\text{váha} := -\text{gen}$
19.,20	parametr – harmonické kanály	16 bitů	jednotlivé bity reprezentují zaškrtnutí 16 kanálů
21.	práh	-255..255	$\text{hodnota} := (\text{gen} * 2) - 255$

Poznámka: U parametrů 3, 8, 10, 14 nevyužíváme plně potenciál genu, protože pomocí celého bytu kódujeme sedmibitové číslo (u parametru 8 dokonce pouze tříbitové). U parametru 21 zase ztrácíme na přesnosti a je možné zakódovat tak pouze lichá čísla. Naopak, výhodou tohoto způsobu je přehlednost při kódování parametrů; každý parametr je zakódován právě jedním bytem.

6.2.2. Vytvoření počáteční populace

Počáteční populace je vytvořena náhodně. Vytváření každého genu se řídí pravděpodobnostmi definovanými ve formuláři parametrů genetického algoritmu. Geny mohou vznikat s nulovou, maximální nebo náhodnou hodnotou. Typické je nastavení vysoké pravděpodobnosti pro náhodnou hodnotu genu.

Obr. 44: Příklad nastavení pravděpodobností vzniku počáteční populace



Vznik nových jedinců:	
Pravděpodobnost vzniku nulové hodnoty genu	5,0%
Pravděpodobnost vzniku maximální hodnoty genu	5,0%
Pravděpodobnost vzniku náhodné hodnoty genu	90,0%
Součet: 100,0%	

6.2.3. Výpočet cílové funkce

Předpokladem pro výpočet cílové funkce je podmínka, že uživatel předložil vzorové řešení, neboli že označil tóny, které mají být považovány za melodické.

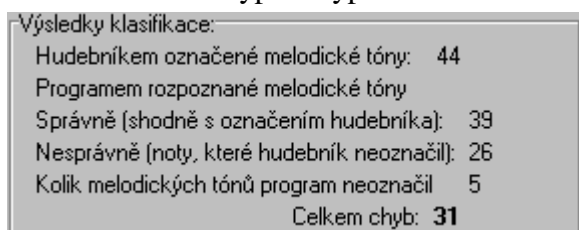
Při výpočtu cílové funkce se příslušný jedinec dekoduje do nastavení vah kritérií, jejich parametrů a prahu. Poté proběhne spuštění automatické klasifikace melodických tónů podle těchto parametrů.

Výsledek této klasifikace je porovnán s uživatelem nastaveným vzorovým řešením a jsou zjištěny následující odchylky:

- nesprávně označené melodické tóny (tóny, které program označil, ale uživatel ne)
- chybějící označení melodických tónů (tóny, které program neoznačil, ale uživatel ano)

Součet těchto hodnot udává počet chyb pro dané nastavení parametrů a je cílovou funkcí příslušného jedince. Cílem je tuto hodnotu minimalizovat.

Obr. 45: Příklad výpisu výpočtu cílové funkce



Výsledky klasifikace:	
Hudebníkem označené melodické tóny:	44
Programem rozpoznané melodické tóny	
Správně (shodně s označením hudebníka):	39
Nesprávně (noty, které hudebník neoznačil):	26
Kolik melodických tónů program neoznačil	5
Celkem chyb: 31	

6.2.4. Operátor selekce

Tento operátor provádí přirozený výběr jedinců. Jeho použití se řídí parametry nastavitelnými ve formuláři parametrů genetického algoritmu.

Pro selekci jsou definovány následující parametry:

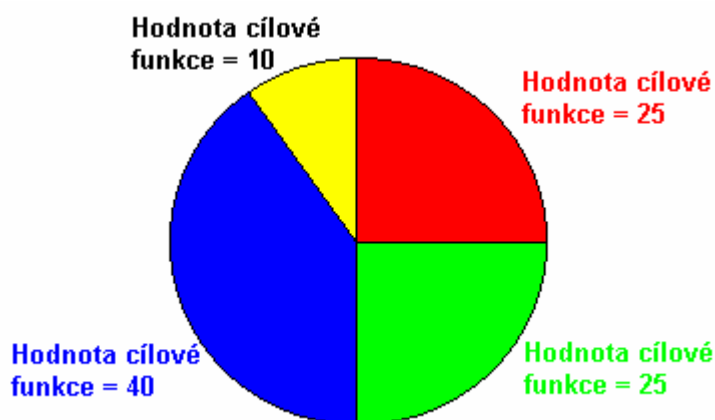
- počet elitních jedinců (tento počet nejlepších jedinců vždy přežívá)
- počet jedinců přežívajících ruletovým výběrem

Jedinci jsou seřazeni vzestupně podle hodnoty cílové funkce, tedy nejlepší jedinci jsou na předních místech. Prvních m jedinců (m = parametr počtu elitních jedinců) vždy přežívá. Tím je zaručeno, že nejlepší dosud nalezené řešení nemůže zaniknout a minimální hodnota cílové funkce s časem nestoupá.

Dalších n jedinců (n = parametr počtu jedinců přežívajících ruletovým výběrem) je pro přežití vybráno z ostatních jedinců. Ruletový mechanismus znamená, že jedinci s lepší hodnotou cílové funkce mají vyšší pravděpodobnost, že přežijí.

V našem případě si můžeme představit, že hodnoty cílové funkce jedinců udávají poměr velikostí kruhových výsečí rulety. Ruleta pak podle příslušných pravděpodobností vybírá řešení k vyřazení. Čím je tedy hodnota cílové funkce vyšší (=horší), tím větší má jedinec pravděpodobnost, že bude ruletovým výběrem vyřazen.

Obr. 46: Příklad vyřazovacího ruletového výběru pro 4 jedince



Ruletový výběr probíhá vícekrát. Dojde přitom k vyřazení příslušného počtu jedinců tak, aby přežil právě požadovaný počet jedinců.

Místo vyřazených jedinců vzniknou jedinci noví. Součástí formuláře parametrů genetického algoritmu je způsob vytváření nových jedinců. Parametry stanovují, kolik jedinců má vzniknout náhodně a kolik křížením. Vytváření nových náhodných jedinců zajišťuje větší variabilitu řešení. Hodnota tohoto parametru ale nesmí být příliš vysoká, protože by znehodnotila přednosti křížení (pokud by dokonce všichni noví jedinci vznikali náhodně, algoritmus by pouze nahodile zkoušel náhodná řešení).

Obr. 47: Příklad nastavení parametrů selekce a vzniku nových jedinců

Selekce:	
Elita - počet nejlepších přežívajících jedinců	4
Počet jedinců přežívajících ruletovým výběrem	6
Doplnění generace:	
Počet jedinců vzniklých křížením	8
Počet nových náhodných jedinců	2
Součet: 20	

6.2.5. Operátory křížení a mutace

V programu Přelad' jsou operátory křížení a mutace použity společně – mutace je aplikována pouze při vytváření nových jedinců jako speciální případ křížení genů.

Křížení probíhá následujícím způsobem: z jedinců, kteří přežili selekci, jsou náhodně vybráni dva jedinci (označíme je jako otce a matku, i když jejich role jsou rovnocenné). Z těchto dvou jedinců vznikne nový jedinec pomocí uniformního křížení. Uniformní křížení znamená, že pro každou základní jednotku (tj. 1 gen, tj. 1 byte) nového jedince je vybrán způsob, jakým vznikne.

V programu jsou použity následující možnosti křížení genů:

- zdědění genu po některém z rodičů
- gen nového jedince vznikne jako aritmetický průměr genů rodičů
- při křížení dojde k mutaci genu na nulovou hodnotu
- při křížení dojde k mutaci na maximální hodnotu
- při křížení dojde k mutaci na náhodnou hodnotu

Pravděpodobnosti těchto možností jsou stanoveny ve formuláři parametrů genetického algoritmu. Pravděpodobnosti mutace by neměly být příliš vysoké, jinak by místo křížení opět docházelo k pouhému náhodnému zkoušení možností.

Obr. 48: Příklad nastavení parametrů křížení a mutace

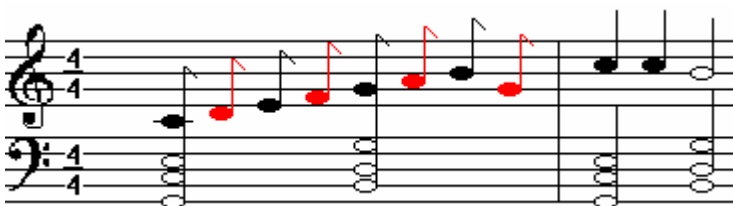
Parametry křížení:	
Pravděpodobnost křížení zděděním genu některého z rodičů	90,0%
Pravděpodobnost křížení průměrem hodnoty genů rodičů	7,0%
Mutace:	
Pravděpodobnost mutace na nulovou hodnotu při křížení	1,0%
Pravděpodobnost mutace na maximální hodnotu při křížení	1,0%
Pravděpodobnost mutace na náhodnou hodnotu při křížení	1,0%
Součet: 100,0%	

6.3. Dosažené výsledky

Použitá implementace genetického algoritmu poměrně dobře řeší detekci některého ze základních kritérií. Pokud jako vzorová data předložíme melodické tóny přesně odpovídající některému z kritérií, algoritmus poměrně brzy nalezne optimální nastavení vah kritérií.

Jako příklad si ukážeme vzorová data odpovídající kritériu těžké doby s parametrem 4 těžkých dob v taktu. Toto nastavení algoritmus s vysokou pravděpodobností detekuje správně již během několika prvních generací.

Obr. 49: Příklad dominance kritéria těžké doby (parametr: 4 těžké doby v taktu)



Podobně algoritmus funguje i pro další jednoduchá nastavení. I s detekcí kombinace několika kritérií si algoritmus umí poměrně dobře poradit.

Naopak, algoritmus většinou nenalezne ideální řešení, pokud vzorová data neobsahují logiku odpovídající nějaké kombinaci kritérií analýzy melodických tónů.

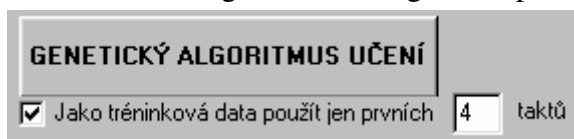
Algoritmus například zaručeně selže, pokud vzorová data obsahují na různých místech dva stejné tóny ve stejném kontextu (tj. označení těchto dvou tónů jako melodických nebo harmonických je ekvivalentní při jakýchkoliv parametrech) a uživatel jeden z nich označí jako melodický a druhý jako harmonický. Program pak nedokáže nalézt nastavení kritérií odpovídající vzorovým datům, jednoduše proto, že takové nastavení kritérií neexistuje.

Problémem je také, pokud je pro vzorové řešení některé kritérium irelevantní. Znamená to, že kritérium je buď splněno pro všechny tóny nebo naopak není splněno pro žádný tón. Pak hodnota tohoto kritéria pro jedince v populaci v průběhu výpočtu osciluje, protože kvalita řešení na něm není přímo závislá a hodnota váhy kritéria plní pouhou funkci absolutního posunu výsledného součtu vah kritérií.

Naopak, pokud má některé kritérium pozitivní vliv na kvalitu řešení, hodnota tohoto kritéria pro jedince v populaci konverguje k vysoké hodnotě váhy kritéria. Podobně, pokud má některé kritérium negativní vliv na kvalitu řešení, hodnota jeho váhy konverguje k nule.

Pro praktické použití se může hodit možnost nastavení vzorového řešení jen pro několik počátečních taktů. Uživatel může označit melodické tóny pouze v několika počátečních taktech a zaškrtnout volbu analýzy pouze několika počátečních taktů:

Obr. 50: Použití genetického algoritmu pouze pro zadaný počet počátečních taktů



GENETICKÝ ALGORITMUS UČENÍ

☒ Jako tréninková data použít jen prvních taktů

Pokud genetický algoritmus nalezne vhodné nastavení parametrů pro toto vzorové řešení, je možné jej aplikovat pro celou skladbu. Tím je možné ušetřit mnoho práce oproti ručnímu označení melodických tónů v celé skladbě.

7. Příloha A: popis standardu MIDI

MIDI (Musical Instrument Digital Interface) je standardem komunikace elektronických klávesových nástrojů a dalších zařízení. Užitečným pomocníkem při tvorbě software i tohoto textu mi byla Forróova publikace Svět MIDI [1], kde je standard MIDI podrobně popsán a vysvětlen na příkladech. Oficiální stránky standardu MIDI se nacházejí na adrese www.midi.org

7.1. Formát SMF – Standard MIDI File (*.mid)

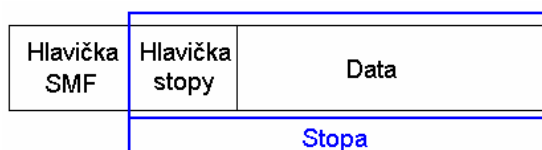
Formát SMF je standardem zápisu MIDI souborů. Slouží jako univerzální komunikační a archivační formát software pracující s MIDI. Soubory standardně používají příponu MID.

Jsou používány dva formáty SMF:

- Formát 0 – obsahuje jednu stopu, která obsahuje všechny MIDI kanály.
- Formát 1 – obsahuje jednu nebo více současně znějících stop. Typicky se používá uložení každého kanálu do jedné stopy. Pokud se v některé stopě přesto nacházejí multikanálová data, používá se pro takovýto soubor termín *nečistý formát 1*.

Data se skládají z hlavičky SMF, po ní následují jednotlivé stopy, přičemž každá stopa je zahájena svou hlavičkou.

Obr. 51: Schéma SMF souboru – formát 0



Obr. 52: Schéma SMF souboru – formát 1



7.1.1. Hlavička SMF

Hlavička SMF se skládá ze 14 bytů. Znázorníme si ji schematicky:

„M“	„T“	„h“	„d“
				L				F	N	R			

Popis:

- „MThd“ = počáteční konstantní textový řetězec
- L (length) = 32-bitové číslo udávající délku následujících dat hlavičky v bytech. Tento údaj teoreticky umožňuje budoucí rozšíření SMF normy, ale v praxi obsahuje typicky hodnotu 6.

- F (format) = 16-bitové číslo udávající formát SMF – obsahuje hodnotu 0 nebo 1
- N (number of tracks) = 16-bitové číslo udávající počet stop. Pro formát 0 je tato hodnota vždy rovna 1
- R (resolution) = 16-bitové číslo udávající rozlišení – vztažnou hodnotu pro kódování času. Jako abstraktní jednotka pro měření času v SMF se používá tzv. tik. Pokud je první bit R roven 0, pak číslo R udává počet tiků za jednu čtvrtřovou dobu. Tato varianta významu R se používá nejčastěji.
Pokud je první bit R roven 1, pak první byte udává SMPTE/EBU formát (formát používaný pro synchronizaci s videodaty) a druhý byte udává počet tiků na okénko. Podrobněji je tato varianta popsána v Forróově Světě MIDI [1] na stranách 92 a 222.

7.1.2. Hlavička stopy

Hlavička stopy je 8-bytová. Pro přehlednost ji také znázorníme jako tabulku:

„M“	„T“	„r“	„k“
				L			

Popis:

- „MTrk“ = počáteční konstantní textový řetězec
- L (length) = 32-bitové číslo udávající délku následujících dat stopy. Tento údaj musí souhlasit se skutečností – stopa nesmí být delší ani kratší, jinak je SMF soubor neplatný.

7.1.3. Data stopy

Základním stavebním kamenem MIDI dat je *událost* neboli zpráva o změně stavu. Typickými příklady událostí jsou například stisk nebo uvolnění klávesy. V SMF jsou dále používány tzv. metaudálosti neboli speciální informační události specifické pro SMF. Pro přehlednost budeme s metaudálostmi pracovat jako s druhem události.

Data stopy obsahují posloupnost MIDI událostí. Informace o každé události je definována posloupností obsahující *deltačas*, *typ události* a *data události*.

Obr. 53: Data stopy v SMF

Deltačas	Typ události	Data události	Deltačas	Typ události	Data události	Deltačas	Typ události	Data události	...
----------	--------------	---------------	----------	--------------	---------------	----------	--------------	---------------	-----

7.1.4. Kódování deltačasu

Deltačas je údaj o relativním časovém umístění události vzhledem k předcházející události. Udává se v ticích. Absolutní čas je dán jednak udanou hustotou tiků v hlavičce SMF a také údajem o tempu, které se nastavuje pomocí metaudálosti Tempo (viz kapitola 7.1.8.: Metaudálosti).

Počet bytů pro kódování deltačasu je proměnlivý a rozpoznáme jej podle prvního bitu. Poslední byte deltačasu má první bit roven 0, jemu předcházející byty mají první bit roven 1. Samotná hodnota deltačasu je postupně uložena do zbylých sedmi bitů každého bytu.

Pro názornost následuje konkrétní příklad zakódovaného deltačasu. Algoritmy pro kódování a dekódování deltačasu jsou podrobněji popsány v programátorské dokumentaci.

Příklad: Dekódování deltačasu

Uvažujme, že se v SMF na místě deltačasu vyskytne sekvence ...\$E0 \$39...

Napišme si údaje ve dvojkové soustavě:

1 110 0000	0 011 1001
------------	------------

První bit je roven 1 –
kódování deltačasu
pokračuje dalším bytem

První bit je roven 0
– jedná se o poslední
byte deltačasu

Zbývající sedmibitové hodnoty se spojí v jedno číslo, které udává hodnotu deltačasu

$(00)11\ 0000\ 0011\ 1001_{(\text{bin})} = 30\ 39_{(\text{hex})} = 12\ 345_{(\text{dec})}$
--

Uložená hodnota deltačasu je tedy rovna hodnotě 12 345 tiků.

7.1.5. Kódování typu a dat události

Typ události je vysílán v podobě tzv. *stavového bajtu*. Stavový bajt je charakteristický tím, že hodnota jeho prvního bitu je vždy rovna 1, tedy stavový bajt může nabývat hodnot 128..256 (\$80..\$FF). Druhý až čtvrtý bit stavového bajtu určují typ události. Existuje tedy $2^3=8$ typů událostí. Označují se podle první číslice zápisu v šestnáctkové soustavě jako \$8, \$9, \$A, \$B, \$C, \$D, \$E a \$F.

Události typu \$8 až \$E se nazývají *kanálová data*, což odpovídá charakteru dat MIDI – jednotlivé nástroje jsou vysílány v různých kanálech. Číslo kanálu, ke kterému se událost váže, je vyjádřeno zbylými čtyřmi bity stavového bajtu (je tedy možné použít $2^4=16$ kanálů).

Události typu \$F znamená tzv. *systémová data*, kde zbylé čtyři bity stavového bajtu nevyjadřují číslo kanálu, ale různé typy příkazů.

Po stavovém bajtu následuje podle typu události proměnlivý počet *databajtu*. U každého databajtu naopak hodnota prvního bitu rovna 0, tedy databajt může nabývat hodnot 0..127 (\$00..\$7F).

V následující tabulce jsou uvedeny nejdůležitější typy událostí a jejich databajty. Parametr *n* u stavového bajtu udává číslo kanálu.

Tab. 9: Události MIDI používané v SMF

	Stavový byte	Funkce	Počet databytů	1. databyte	2. databyte
Kanálová data	\$8n	Nota vypnuta (uvolnění klávesy)	2	Číslo noty	Síla
	\$9n	Nota zapnuta (stisk klávesy)	2	Číslo noty	Síla
	\$An	Individuální tlaková citlivost	2	Číslo noty	Hodnota
	\$Bn	Kontroler	2	Číslo kontroleru	Hodnota
	\$Cn	Volba programu	1	Číslo programu	-
	\$Dn	Společná tlaková citlivost	1	Hodnota	-
	\$En	Ohýbání tónu	2	LSB*	MSB*
Systémová data	\$FF	Metaudálost SMF	2**	Typ	Délka [byte]
	\$F0	Začátek zvláštních systémových dat	1***	Délka [byte] kódovaná obdobně jako deltačas	-
	\$F7	Konec zvlášť. systémových dat	0	-	-

Poznámky k tabulce MIDI událostí:

* - po odstranění prvního (nulového) bitu u hodnot LSB a MSB a jejich spojením (LSB jako nižší, MSB jako vyšší) získáváme 14-bitové číslo udávající hodnotu parametru události ohýbání tónu.

** - po bajtu \$FF a jeho dvou databajtech následují další data. Jejich délka v bajtech je dána přímo hodnotou v druhém databajtu.

*** - po bajtu \$F0 následuje hodnota délky zvláštních systémových dat (zakódovaná obdobně jako deltačas). Následují zvláštní systémová data. Sekvence musí být ukončena stavovým bajtem \$F7.

Poznámka: Stavové bajty \$F1, \$F2, \$F3, \$F6, \$F8, \$FA, \$FB, \$FC a \$FE se v SMF nepoužívají – jsou to systémová data reálného času. Stavové bajty \$F4, \$F5, \$F9 a \$FD nejsou definovány. Podrobný výčet MIDI událostí je ve Forróově Světě MIDI [1] na str. 31.,

7.1.6. Popis formátu událostí

Nyní vyjmenujeme důležité druhy událostí z hlediska výšky jednotlivých tónů. Popsány budou události **Nota zapnuta**, **Nota vypnuta**, **Ohýbání tónu** a **Kontroler**. Ostatní události (*Individuální tlaková citlivost*, *Volba programu* a *Společná tlaková citlivost*) zde podrobně nepopisují; je možné je nalézt např. ve Forróově Světě MIDI [1].

- **Nota zapnuta (událost \$8n)**

\$8n	.	.
	T	S

V kanálu číslo *n* stiskne notu číslo T (0..127) silou S (0..127; S se také někdy značí jako rychlost stisku). Číslování not je popsáno v dodatku Tabulky MIDI not.

- **Nota vypnuta (událost \$9n)**

\$9n	.	.
	T	S

V kanálu číslo *n* vypne notu číslo T (0..127) silou S (0..127; S značí rychlost uvolnění)

- **Ohýbání tónu (událost \$En)**

\$En	.	.
	LSB	MSB

Tato událost mění výšku všech tónů v kanálu *n* podle parametrů LSB (0..127) a MSB (0..127). Po odstranění prvního (nulového) bitu u hodnot LSB a MSB a jejich spojením (LSB jako nižší, MSB jako vyšší) získáváme 14-bitové číslo udávající hodnotu parametru události ohýbání tónu.

Výpočet můžeme vyjádřit následujícím vzorcem:

$$\text{Hodnota_Ohýbání_Tónu} = \text{LSB} + 128 \cdot \text{MSB}$$

Toto číslo vyjadřuje hodnotu v rozmezí 0..16383. Hodnota 8192 znamená žádné ohýbání, kdy tóny kanálu *n* zůstávají ve standardní výšce.

Maximální hodnota 16383 znamená posun o celý tón výše (tj. o 200 centů výše), minimální hodnota 0 znamená posun o celý tón níže (tj. o 200 centů níže).

Tuto událost je možné použít například pro *glissando*, nebo v našem případě pro mikroladění (viz kapitola 7.2.: Zápis mikroladění do MIDI).

Poznámka: Rozsah ohýbání tónu je standardně ±200 centů, ale pomocí kontroleru RPN je možné tento rozsah změnit (viz kapitola 7.2.: Zápis mikroladění do MIDI).

- **Kontroler (událost \$Bn)**

\$Bn	.	.
	N	H

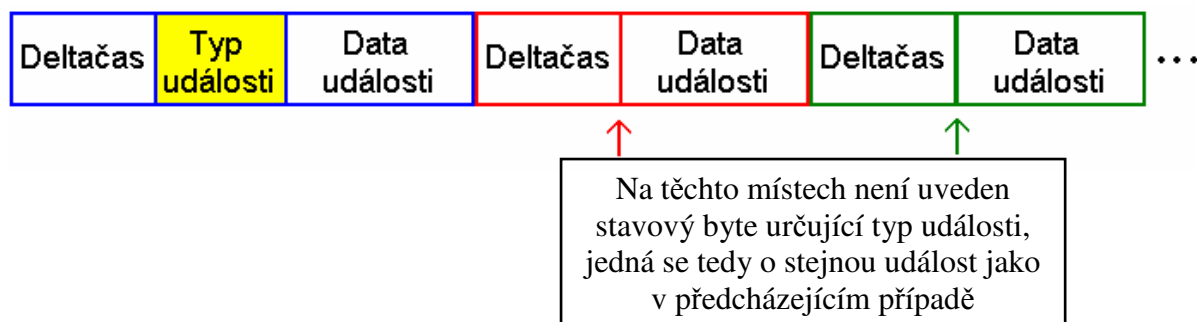
Kontrolery umožňují měnit parametry přehrávání v reálném čase. Číslo N (0..127) udává číslo kontroleru, číslo H jeho hodnotu (0..127). Pomocí události kontroleru jsou například zapisovány informace o stisku pedálu, přepínání skupin nástrojů apod. Nebudeme zde podrobně popisovat všechny definované kontrolery, podrobný popis všech kontrolerů je možné nalézt například ve Forróově Světě MIDI v kapitole 2.4.1.3.: Kontrolery. V dalším textu budeme používat kontrolery v kapitole 7.2.: Zápis mikroladění do MIDI.

7.1.7. Trvající stavový byte

Pro snížení množství dat v MIDI se používá tzv. trvající stavový byte. Znamená to, že pokud se posílá vícekrát událost stejného typu, uvede se stavový byte pouze jednou. U následujících událostí je dovoleno uvést pouze deltačas a databyty – nemusí se znovu vysílat stejný stavový byte. Trvající stavový byte platí do té doby, než se objeví jiný stavový byte.

Tento postup je možný díky prostému faktu, že u stavových bytů je hodnota prvního bitu rovna 1, zatímco u databytů je první bit roven nule. Když tedy po deltačasu následuje byte, který není stavovým bytem, je to počátek databytů stejného typu události, jako byla předchozí.

Obr. 54: Trvající stavový byte v SMF



Poznámka: Vzhledem k tomu, že místo události „Nota vypnuta“ můžeme vysílat událost „Nota zapnuta silou 0“, je možné tato základní hudební data posílat ve formě trvajících stavových bytů. Událost „Nota vypnuta“ tedy ztratila na významu, důvodem její trvající existence je pouze zpětná kompatibilita.

7.1.8. Formát metaudálostí

Metaudálosti reprezentují různá speciální data skladby, jako tempo, metrum, tóninu, textové popisy apod. Každá metaudálost má následující formát:

\$FF
	T	L	Data				

- T (type) – 8-bitové číslo udávající typ metaudálosti. První bit T je vždy roven 0, tedy T může nabývat hodnot 0..127 (\$00..\$7F)
- L (length) – 8-bitové číslo udávající délku dat metaudálosti v bytech. První bit L je vždy roven 0, tedy maximální možná délka dat je 127.
- Data – posloupnost bytů, jejich délka je dána hodnotou L.

7.1.9. Nejdůležitější typy metaudálostí SMF

Uvádím zde pouze metaudálosti podstatné pro správnou reprodukci skladby v SMF. Úplný výčet typů metaudálostí je možné nalézt např. ve Forróově Světě MIDI [1] v kapitole 2.5.2.1.1 (Formát SMF).

- **Konec stopy (metaudálost \$2F)**

\$FF	\$2F	\$00
	T	L

Touto metaudálostí musí být povinně zakončena každá stopa, jinak je SMF soubor neplatný. Tato metaudálost nemá další parametry.

- **Tempo (metaudálost \$51)**

\$FF	\$51	\$03	.	.	.
	T	L	Tempo		

Tato událost by měla být uvedena na začátku každé skladby. Parametr **Tempo** je 24-bitové číslo udávající počet mikrosekund za jednu čtvrtovou dobu. V průběhu skladby je možné tuto metaudálost volat znovu a tím docílit změny tempa. Pokud SMF soubor

neobsahuje metaudálost tempa, standardně se použije tempo 120 BPM (odpovídající délce trvání jedné čtvrt'ové $T = \$07 \$A1 \$20 = 500\,000\,\mu s = 0,5\text{ sekundy}$)

Poznámka: důležitým údajem pro stanovení skutečného tempa je také rozlišení definované v hlavičce SMF – viz kapitola 7.1.1.: Hlavička SMF.

- **Metrum (metaudálost \$58)**

\$FF	\$58	\$04
	T	L	N	D	C	B

Tato událost by také měla být uvedena na začátku skladby. Všechny parametry jsou 8-bitová čísla. Parametr N udává přímo hodnotu čitatele metrického údaje, parametr D pak kóduje jmenovatele takto: 0 je celá hodnota, 1 půlová, 2 čtvrt'ová, 3 osminová, atd. Parametry C a B souvisí s metronomem – C udává počet MIDI synchronizací v jednom ťuknutí metronomu (def. synchronizací: 1 čtvrt'ová doba se skládá z 24 synchronizací), parametr B označuje počet psaných dvaatřicetinek v jedné čtvrt'ové (pro případ, že má některý program tento parametr nastavitelný – standardně je B=8). Pokud SMF soubor neobsahuje metaudálost metrum, je standardně použit 4/4 takt s metronomem na každou čtvrt'ovou, tj. jako by byla zavolána událost metrum ve tvaru \$FF \$58 \$04 \$04 \$02 \$18 \$08.

- **Předznamenání a tónorod (metaudálost \$59)**

\$FF	\$59	\$02	.	.
	T	L	SF	MI

SF je 8-bitové číslo se znaménkem (první bit určuje znaménko, přičemž záporná čísla jsou kódována dvojkovým doplňkem: -1 je kódováno jako \$FF, -2 jako \$FE, atd.).

Záporná hodnota SF udává počet běček, kladná počet křížků.

MI je 8-bitové číslo určující tónorod, nabývat může pouze dvou hodnot: 0 (dur) nebo 1 (moll).

7.1.10. Zvláštní systémová data

U tohoto typu dat je součástí MIDI standardu pouze hlavička a ukončující příkaz. Vlastní obsah dat si definují jednotliví výrobci pro různé účely.

Obrázek: Zvláštní systémová data

\$F0	(var)	\$F7
	L	ID	Data					

Po stavovém bytu \$F0 následuje parametr L proměnlivé délky – tento parametr udává délku následujících dat (včetně zakončujícího databytu \$F7) a je kódován stejně jako deltačas (viz kapitola 7.1.4.: Kódování deltačasu). Následuje byte ID udávající kód výrobce a poté následují výrobcem specifikovaná data.

Ze zvláštních systémových dat pro nás bude zajímavý mikrointervalový standard MTS, který podrobně popisujeme v následující kapitole 7.2.2.: Speciální instrukce mikroladění. Podrobným popisem dalších typů zvláštních systémových dat se nebudeme podrobněji zabývat. Podrobné informace je možné nalézt například ve Forróově Světě MIDI [1].

7.2. Zázpis mikroladění do MIDI

Standardně jsou tóny v MIDI naladěny podle rovnoměrného temperovaného ladění. Abychom mohli mikrointervalově nastavovat výšky tónů, musíme použít některý ze způsobů změny výšky tónu.

Uvedeme si zde dva způsoby mikroladění v MIDI. První z nich používá událost ohýbání tónu a druhý vysílá zvláštní systémová data pro mikroladění.

7.2.1. Mikroladění pomocí ohýbání tónu

Jak již bylo popsáno ve výčtu typů událostí, umožňuje událost ohýbání tónu měnit výšku všech tónů v určitém kanálu. V rámci každého kanálu je tedy zachováno rovnoměrné temperované ladění, ale všechny tóny kanálu jsou posunuty výš nebo níž.

Vidíme tedy, že mikroladění pomocí ohýbání tónu má určitá omezení. Protože počet kanálů je 16, můžeme definovat nejvýše 16 možných posunů. Navíc musíme vystačit s jediným nástrojem, jehož tóny podle potřeby vysíláme v různých kanálech.

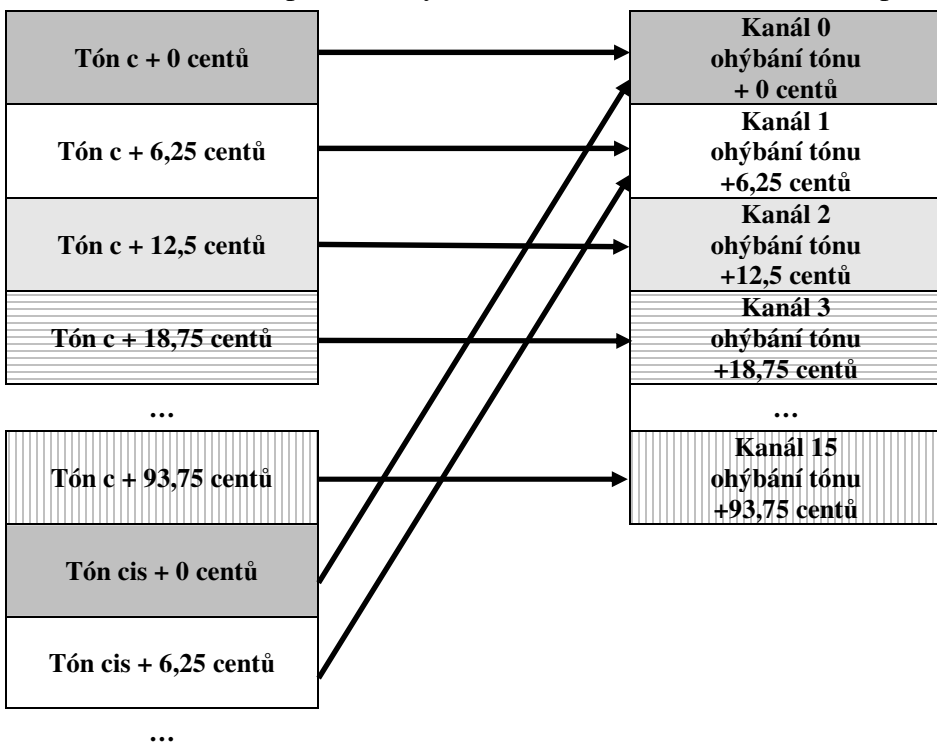
Oproti tomu je výhodou, že událost ohýbání tónu je implementována v naprosté většině zvukových karet a v dalších zařízeních. Máme tedy téměř stoprocentní jistotu, že takto zapsaný SMF soubor bude přehráno korektně.

Otázkou tedy zůstává, jak nastavit posuny jednotlivých kanálů a jak rozmístit jednotlivé tóny. Popíšu zde dva možné postupy.

1. Rovnoměrné rozdělení půltónu na jemnější intervaly

Forró ve Světě MIDI [1] v kapitole 2.4.1.5. (Ohýbání tónu) popisuje možnost rovnoměrného rozložení posunu 16 kanálů v rozahu půltónu. Do každého kanálu mohou být vysílány jakékoliv tóny. Tím se získává poměrně jemné rozlišení mikroladění – oktáva se rozděluje na 192 dílků. To je dobře použitelné pro zjemnění temperovaného ladění. Vzhledem k tomu, že každý půltón lze rozdělit na 16 dílků, získáváme jemnost rozlišení 6,25 centů.

Obr. 55: Mikroladění pomocí ohýbání tónu – rovnoměrné rozdělení půltónu na 16 dílků



2. Zařazení každého tónu stupnice do samostatného kanálu

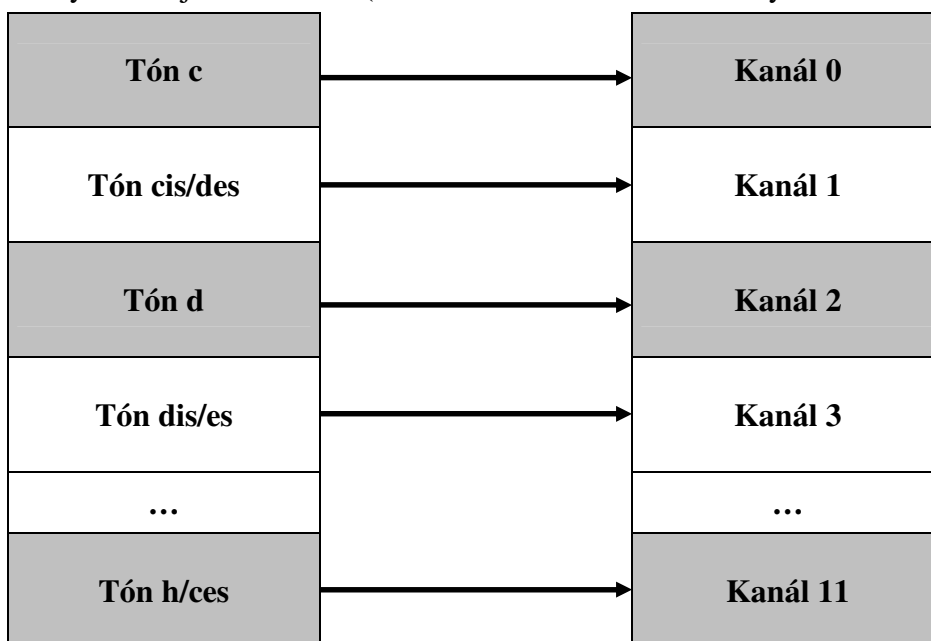
Pokud nám rozlišení popsané v předchozím případě nestačí, máme možnost použít následující způsob, který popisuje např. John Sankey (www.sankey.ws/miditune.html). Každý z 12 tónů oktávy umístíme do vlastního kanálu. Na tento kanál aplikujeme příslušné mikroladění podle naší potřeby – pro celý kanál můžeme použít potřebné ohýbání tónu.

Tento postup zápisu do MIDI je použit v programu „Přelad“. Lze nastavit zaznění tónů v určité výšce, poté přeladit kanál na jinou výšku a zahrát tón v pozměněné výšce.

Nevýhodou oproti předchozímu řešení je menší počet tónů použitých v rámci oktávy. Můžeme tedy použít pouze 12 tónů; nemůžeme například snadno zajistit souzvuk enharmonicky záměnných tónů (např. cis/des), protože tyto tóny jsou vzhledem ke stejnému číslu MIDI noty umístěny do stejného kanálu.

Naopak výhodou je, že výšku jednotlivých tónů můžeme nastavit jakkoliv a v průběhu skladby ji měnit. Další výhodou je, že můžeme změny ladění snadno aplikovat na existující MIDI nahrávku; pouze změníme ohýbání tónu pro jednotlivé kanály.

Obr. 56: Postup pro mikroladění použitý v programu: Mikroladění pomocí ohýbání tónu, každý tón do jiného kanálu (v každém kanálu nastavíme ohýbání tónu dle potřeby)



7.2.2. Instrukce mikroladění – standard MTS

Mikrointervalový standard MTS (Midi Tuning Standard), který je dodatkem MIDI normy, je založen na možnosti přeladit libovolný tón na jakoukoliv výšku. Hodnoty přeladění jsou uloženy v tabulkách přeladění, kde je ke každému tónu uvedena jeho požadovaná výška.

MTS umožňuje přepínání různých ladění. Tato ladění jsou uložena v paměti ve formě tabulek; každá tabulka znamená určité naladění. Standard MTS předpokládá možnost uložení 128 skupin tabulek (*banků*), kde se v každé skupině nachází 128 tabulek. Je tedy možné uložit až 16 384 tabulek ladění.

V každé tabulce je ke každému tónu přiřazena frekvence. Ta je při přenosu dat kódována třemi byty. První byte je tzv. hrubé přeladění a udává nejvyšší nižší číslo MIDI tónu (0..127) na který se má přeladit. Druhý a třetí byte určují jemné přeladění (*fine tuning*), které udává zvýšení tohoto tónu oproti standardní frekvenci.

Jemné přeladění je kódováno 14-bitově; po odstínění prvního (nulového) bitu obou bytů získáme dvě sedmibitová čísla, jejichž složením (první byte jako vyšší řád, druhý byte jako nižší řád) získáme 14-bitové číslo udávající zvýšení tónu.

Nejmenší hodnota (0) znamená standardní výšku tónu, nejvyšší hodnota (16 384) znamená zvýšení o temperovaný půltón (tj. o 100 centů). Vidíme tedy, že MTS umožňuje rozlišení $100/16384 =$ přibližně 0,0061 centů.

Poznámka: Hodnota přeladění \$7F \$7F \$7F je rezervována pro příkaz „žádná změna“, tj. ponechání tónu bez přeladění.

Tab. 11: Příklady dat mikroladění

Číslo MIDI noty	Jemné přeladění		Frekvence	Poznámka
\$00	\$00	\$00	8,175799 Hz	Tón 0: MIDI C0 (nejnižší tón)
\$45	\$00	\$00	440 Hz	Tón 69: MIDI A5
\$45	\$00	\$01	440,0016 Hz	Tón 69 zvýšený o 0,0061 centu (tj. jemné přeladění = 1)
\$45	\$00	\$A4	440,2542 Hz	Tón 69 zvýšený o 1 cent (tj. jemné přeladění = 164)
\$45	\$40	\$00	452,8930 Hz	Tón 69 zvýšený o 50 centů (čtvrttón) (tj. jemné přeladění = 8192)
\$7F	\$00	\$00	12543,85 Hz	Tón 127: MIDI G10
\$7F	\$7F	\$7E	13289,70 Hz	Tón 127 zvýšený o maximum (nejvyšší možná frekvence)
\$7F	\$7F	\$7F	-	Žádná změna

Poznámka: hodnoty frekvence jsou v tabulce zaokrouhleny na 7 platných cifer, aby byl názorně vidět rozdíl změny jemného přeladění. Přesnost 0,0061 centu je velmi jemná, lidským sluchem jsou rozlišitelné teprve hodnoty více než stokrát větší.

MTS specifikuje možnost výměny tabulek mezi zařízeními pomocí tzv. *non-realtime příkazů zvláštních systémových dat*. Jsou použity následující příkazy pro výměnu tabulek mezi zařízeními viz Forróův Svět MIDI [1] str. 141:

- **Žádost o vyslání tabulky**

Tento příkaz se má následující formát:

\$F0	\$7E	..	\$08	\$00	..	\$F7
ID	I				T	

ID \$7E je standardní návěstí univerzálních systémových dat. Číslo I (0..127) je identifikátor zařízení – každý přístroj zapojený na MIDI sběrnici je reprezentován jednou hodnotou. Číslo T (0..127) udává tabulku mikroladění.

- **Výpis tabulky**

Následující data jsou vyslána jako výpis tabulky:

\$F0	\$7E	..	\$08	\$0116byte..	\$F7
	ID	I			T	A	B1	B2		

128x

Čísla ID, I a T mají stejný význam, jako v předchozím případě. Poté následuje 16 ASCII znaků udávajících název ladění a dále pokračuje 128x opakovaná posloupnost čísel B1 (0..127) a B2 (14-bitové číslo) udávající parametry jednotlivých MIDI tónů dle MTS; B1 je číslo nejbližší nižší MIDI noty B2 je 14-bitová hodnota udávající jemné rozladění.

Dále je možné použít následující *realtime příkaz zvláštních systémových dat* pro změnu tabulky některého ze zařízení (viz Forróův Svět MIDI [1], str. 142):

- **Přeladění tónu**

\$F0	\$7E	..	\$08	\$02	\$F7
	ID	I			T	L	K	B1	B2		

L-krát

Čísla ID, I a T mají stejný význam, jako v předchozím případě. Hodnota L udává počet přeladovaných not a poté následuje L-krát opakovaná posloupnost čísel K, B1 a B2. Hodnota K je číslo MIDI noty, která se přeladuje, hodnoty B1 a B2 stejně jako v předchozím případě udávají hodnotu, na kterou se má nota přeladit.

Pro aplikaci mikroladění pomocí MTS je potřeba použít volání určených kontrolerů. Popíšeme si zde kontrolery 101, 100, 6 a 38, které společně tvoří posloupnost kontrolerů nazvanou RPN (Registered Parametr Number). RPN umožňuje nastavovat rozsah ohýbání tónu, ladit výšku tónů v kanálu a přepínat tabulky mikroladění. Uvedené čtyři kontrolery jsou typicky vyslány postupně za sebou v pořadí 101, 100, 6 a 38. Funkce RPN je různá pro různé hodnoty vyslané v kontroleru 101 a 100. Standardně jsou pro následující kombinace hodnot kontrolerů 101 a 100 definovány tyto funkce:

Tab. 12: RPN

Hodnota kontroleru 101	Hodnota kontroleru 100	Funkce
\$00	\$00	Rozsah ohýbání tónu
\$00	\$01	Jemné ladění
\$00	\$02	Hrubé ladění
\$00	\$03	Výběr tabulky mikroladění
\$00	\$04	Výběr skupiny tabulek (<i>banku</i>) mikroladění
\$7F	\$7F	Deaktivovat používanou funkci RPN

Parametry těchto funkcí jsou vysílány pomocí kontrolerů 6 a 38. Význam kontrolerů 6 a 38 závisí na zvolené funkci kontrolerů 101 a 100:

- **Rozsah ohýbání tónu** (kontroler 101: \$00, kontroler 100: \$00)
Tato funkce mění rozsah, ve kterém bude probíhat ohýbání tónu. Hodnota kontroleru 6 udává velikost tohoto rozsahu v půltónech, hodnota kontroleru 38 se nepoužívá.

Příklad: rozsah ohýbání tónu ± 1 oktáva (tj. 12 půltónů) v kanálu 0:

\$B0	\$65	\$00	\$B0	\$64	\$00	\$B0	\$06	\$0C
	(101)	(0)		(100)	(0)		(6)	(12)

Poznámka: standardní hodnotou rozsahu ohýbání tónu je rozsah 2 půltónů, tedy celého tónu, odpovídající možnosti ohýbání tónu ± 200 centů.

- **Jemné ladění** (kontroler 101: \$00, kontroler 100: \$01)
Tato funkce slouží k jemnému naladění kanálu v rozmezí ± 100 centů. Hodnota kontroleru 6 udává vyšších 7 bitů (hodnota MSB), hodnota kontroleru 38 udává nižších 7 bitů (hodnota LSB). Celkem tedy získáváme 14-bitové číslo udávající hodnotu jemného ladění.
Výpočet můžeme vyjádřit následujícím vzorcem:
$$\text{Jemné_Ladění} = \text{LSB} + 128 \cdot \text{MSB}$$

Toto číslo vyjadřuje hodnotu v rozmezí 0..16383. Hodnota 8192 znamená žádný posun oproti temperovanému ladění, kdy tóny kanálu *n* zůstávají ve standardní výšce. Maximální hodnota 16383 znamená posun o půltón výše (tj. o 100 centů výše), minimální hodnota 0 znamená posun o půltón níže (tj. o 100 centů níže).
- **Hrubé ladění** (kontroler 101: \$00, kontroler 100: \$02)
Tato funkce změní výšky tónů v kanálu po půltónech v rozsahu ± 24 půltónů. Hodnota kontroleru 6 udává posun v půltónech následujícím způsobem: hodnota \$40 znamená standardní výšku, hodnota \$28 je posun o 24 půltónů níže a hodnota \$58 znamená posun o 24 půltónů výše
Kontroler 38 se zde nepoužívá.
- **Výběr tabulky mikroladění** (kontroler 101: \$00, kontroler 100: \$03)
Hodnota kontroleru 6 udává vyšších 7 bitů (hodnota MSB), hodnota kontroleru 38 udává nižších 7 bitů (hodnota LSB). Výsledné číslo určuje volbu tabulky MTS.
- **Výběr banku mikroladění** (kontroler 101: \$00, kontroler 100: \$04)
Hodnota kontroleru 6 udává vyšších 7 bitů (hodnota MSB), hodnota kontroleru 38 udává nižších 7 bitů (hodnota LSB). Výsledné číslo určuje volbu banku MTS.
- **Deaktivace RPN** (kontroler 101: \$7F, kontroler 100: \$7F)
Tento povel deaktivuje aktuálně používané nastavení. Po jeho spuštění již kontrolery 6 a 38 nemění popsané parametry RPN.

7.3. Dodatek – tabulky MIDI not

7.3.1. Tabulka číslování MIDI not

Tab. 13: Čísla MIDI not

Poř.číslo oktávy	Ozn. oktávy	Hudební označení	C	C#	D	D#	E	F	F#	G	G#	A	A#	H
0	-2	sub-subkontra	0	1	2	3	4	5	6	7	8	9	10	11
1	-1	subkontra	12	13	14	15	16	17	18	19	20	21	22	23
2	0	kontra	24	25	26	27	28	29	30	31	32	33	34	35
3	1	velká	36	37	38	39	40	41	42	43	44	45	46	47
4	2	malá	48	49	50	51	52	53	54	55	56	57	58	59
5	3	jednočárkovaná	60	61	62	63	64	65	66	67	68	69	70	71
6	4	dvoučárkovaná	72	73	74	75	76	77	78	79	80	81	82	83
7	5	tříčárkovaná	84	85	86	87	88	89	90	91	92	93	94	95
8	6	čtyřčárkovaná	96	97	98	99	100	101	102	103	104	105	106	107
9	7	pětičárkovaná	108	109	110	111	112	113	114	115	116	117	118	119
10	8	šestičárkovaná	120	121	122	123	124	125	126	127				

7.3.2. Tabulka frekvencí MIDI not

Tab. 14: Frekvence MIDI not

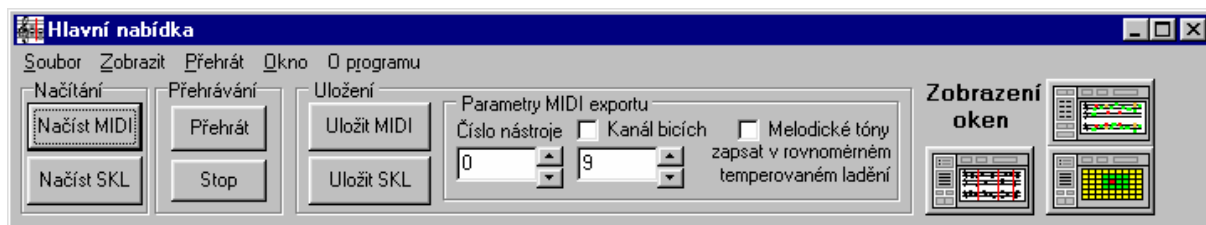
Oktáva	Název noty	Číslo MIDI noty (i)	Frekvence (f) [Hz]	Oktáva	Název noty	Číslo MIDI noty (i)	Frekvence (f) [Hz]
sub-subkontra	C	0	8,176	dvoučárkovaná	C	72	523,251
	C#	1	8,662		C#	73	554,365
	D	2	9,177		D	74	587,330
	D#	3	9,723		D#	75	622,254
	E	4	10,301		E	76	659,255
	F	5	10,913		F	77	698,456
	F#	6	11,562		F#	78	739,989
	G	7	12,250		G	79	783,991
	G#	8	12,978		G#	80	830,609
	A	9	13,750		A	81	880,000
	A#	10	14,568		A#	82	932,328
subkontra	H	11	15,434		H	83	987,767
	C	12	16,352	tříčárkovaná	C	84	1 046,502
	C#	13	17,324		C#	85	1 108,731
	D	14	18,354		D	86	1 174,659
	D#	15	19,445		D#	87	1 244,508
	E	16	20,602		E	88	1 318,510
	F	17	21,827		F	89	1 396,913
	F#	18	23,125		F#	90	1 479,978
	G	19	24,500		G	91	1 567,982
	G#	20	25,957		G#	92	1 661,219
	A	21	27,500		A	93	1 760,000
	A#	22	29,135		A#	94	1 864,655
kontra	H	23	30,868		H	95	1 975,533
	C	24	32,703	čtyřčárkovaná	C	96	2 093,005
	C#	25	34,648		C#	97	2 217,461
	D	26	36,708		D	98	2 349,318
	D#	27	38,891		D#	99	2 489,016
	E	28	41,203		E	100	2 637,020
	F	29	43,654		F	101	2 793,826
	F#	30	46,249		F#	102	2 959,955
	G	31	48,999		G	103	3 135,963
	G#	32	51,913		G#	104	3 322,438
	A	33	55,000		A	105	3 520,000
	A#	34	58,270		A#	106	3 729,310
velká	H	35	61,735		H	107	3 951,066
	C	36	65,406	pětáčárkovaná	C	108	4 186,009
	C#	37	69,296		C#	109	4 434,922
	D	38	73,416		D	110	4 698,636
	D#	39	77,782		D#	111	4 978,032
	E	40	82,407		E	112	5 274,041
	F	41	87,307		F	113	5 587,652
	F#	42	92,499		F#	114	5 919,911
	G	43	97,999		G	115	6 271,927
	G#	44	103,826		G#	116	6 644,875
	A	45	110,000		A	117	7 040,000
	A#	46	116,541		A#	118	7 458,620
malá	H	47	123,471		H	119	7 902,133
	C	48	130,813	šestičárkovaná	C	120	8 372,018
	C#	49	138,591		C#	121	8 869,844
	D	50	146,832		D	122	9 397,273
	D#	51	155,563		D#	123	9 956,063
	E	52	164,814		E	124	10 548,082
	F	53	174,614		F	125	11 175,303
	F#	54	184,997		F#	126	11 839,822
	G	55	195,998		G	127	12 543,854
	G#	56	207,652				
	A	57	220,000				
	A#	58	233,082				
jednočárkovaná	H	59	246,942				
	C	60	261,626				
	C#	61	277,183				
	D	62	293,665				
	D#	63	311,127				
	E	64	329,628				
	F	65	349,228				
	F#	66	369,994				
	G	67	391,995				
	G#	68	415,305				
	A	69	440,000				
	A#	70	466,164				
	H	71	493,883				

8. Příloha B: Uživatelská dokumentace

Uživatelská dokumentace k programu *Přelad'* popisuje funkce ovládacích prvků jednotlivých oken. Vzhledem k tomu, že funkce programu byly popsány již v kapitole 4, některé části popisu se zde opakují.

8.1. Okno Hlavní nabídka

Obr. 57: Okno Hlavní nabídka



Tlačítka práce se soubory:

- **Načíst MIDI:** Importuje MIDI soubor do programu Přelad'.
- **Načíst SKL, Uložit SKL:** Načte nebo uloží skladbu ve vlastním formátu SKL. Tento formát obsahuje také informace o označení melodických tónů a vložených laděních.
- **Uložit MIDI:** Exportuje MIDI soubor včetně přeladění pomocí ohýbání tónu popsaného v kapitole 7.2.1. část 2 (zařazení každého tónu do samostatného kanálu). Z exportovaného MIDI souboru už program neumožňuje zpětné získání informací o dlaždicovém ladění a o označení melodických tónů.
- **Přehrát, Stop:** Přehraje aktuálně načtenou skladbu včetně případných změn ladění.
- Tyto funkce je možné volat také výběrem položek menu **Soubor** a **Přehrát**.

Parametry MIDI exportu:

- **Číslo nástroje:** Vzhledem ke způsobu přeladění se pro všechny kanály používá jeden nástroj. Tento parametr udává číslo MIDI nástroje exportovaného do MIDI souboru.
- **Kanal bicích:** Pokud skladba obsahuje bicí nástroje, zaškrtnutím tohoto tlačítka a výběrem kanálu bicích nástrojů zachováme jeho tóny v původním kanálu.
- **Melodické tóny v rovnoměrném temperovaném ladění:** Umístí melodické tóny do samostatného kanálu, který není přeladěn.

O programu:

- Položka menu **O programu** informuje o verzi programu Přelad'.

Zobrazení oken:

- Menu **Zobrazit:** Výběrem položky tohoto menu zobrazíme na popředí příslušné okno.
- Menu **Okno – Uspořádat okna:** Nastaví oknům polohy a velikosti typické pro standardní rozlišení obrazovky 1024x768.

- Tlačítka typických zobrazení:



Typické zobrazení pro manuální ladění a jeho vkládání do skladby.



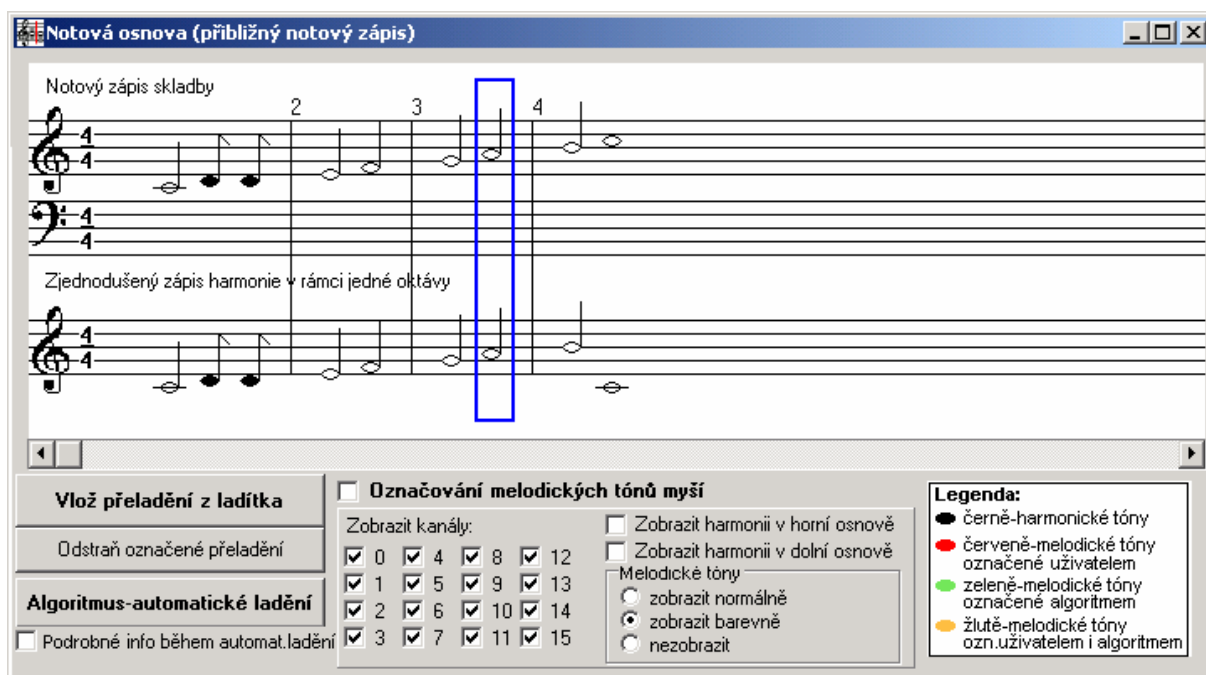
Typické zobrazení pro nastavení ladění v dlaždicovém prostoru.



Typické zobrazení pro analýzu harmonických a melodických tónů.

8.2. Okno Notová osnova

Obr. 58: Okno Notová osnova



V okně Notová osnova se zobrazuje aktuálně načtená skladba. Horní dvě notové osnovy (dále jen Horní osnova) obsahují notový zápis skladby. Dolní notová osnova (dále jen Dolní osnova) obsahuje zjednodušený zápis harmonie v rámci jedné oktávy – tóny jsou zobrazeny v jednočárkované oktávě.

Kliknutím myší do prostoru zobrazení notových osnov nastavujeme na příslušné místo kurzor, který slouží pro nastavení místa pro vložení instrukce přeladění. Při kliknutí na symbol instrukce přeladění se příslušně změní nastavení v oknech Ladítka a Dlaždice.

Okno Notová osnova obsahuje následující ovládací prvky:

Možnosti zobrazení:

- **Zobrazení kanálů:** Zobrazí pouze tóny, které byly v originálním MIDI souboru v zaškrtnutých kanálech.
- **Zobrazit harmonii v horní/dolní osnově:** Zobrazí aktuálně znějící souzvuk pro každý tón, a to v horní nebo dolní osnově.
- **Zobrazení melodických tónů:** Umožňuje zobrazit nebo skrýt tóny, které byly označeny jako melodické.

Přepínače:

- **Označování melodických tónů myši:** Pokud je tento přepínač zaškrtnut, při kliknutí na notu se mění její označení jako melodické nebo harmonické. Nota, která je označena jako harmonická (černě), se po kliknutí označí jako uživatelem označená melodická nota (červeně). Nota, která je programem nebo uživatelem označena jako melodická (červeně nebo zeleně) se po kliknutí označí jako harmonická (černě). Nota, která byla označena jako melodická programem i uživatelem (žlutě) se po kliknutí změní na označenou pouze uživatelem (červeně). Pokud tento přepínač není zaškrtnut, kliknutí myši funguje standardně, tj. pouze nastavuje umístění kurzoru ve skladbě.
- **Podrobné info během automatického ladění:** Přepíná zobrazování podrobných informací během automatického doplnění ladění.

Tlačítka jednotlivých funkcí:

- **Vložit přeladění z ladítka:** Na místo kurzoru, které jsme vybrali kliknutím myši do skladby, umístí přeladění nastavené v okně Ladítko.
- **Odstraň označené přeladění:** Odstraní ze skladby přeladění na místě kurzoru.
- **Algoritmus – automatické ladění:** Spustí algoritmus automatického návrhu přirozeného ladění popsaného v kapitole 5. Před spuštěním tohoto algoritmu by měla být provedena klasifikace harmonických a melodických tónů.

8.3. Okno Hledání melodických tónů

Obr. 59: Hledání melodických tónů

Toto okno umožňuje spuštění automatické klasifikace melodických tónů na základě nastavitelných vah kritérií. Použitá kritéria jsou podrobně popsána v kapitole 3.

Kritéria *Soprán-Tón vyšší než..*, *Těžká doba*, *Bas-Tón nižší než..* a *Dlouhý tón* mají další parametry, které je možné nastavit kliknutím na tlačítko

Parametry kritérií kanálů harmonických a melodických tónů se nastavují pomocí zaškrtnutých tlačítek. Okno dále obsahuje následující ovládací prvky:

Přepínače:

- **Způsob klasifikace:** Nastavení, zda se mají před klasifikací melodických tónů označit všechny tóny jako harmonické, nebo k nyní označeným melodickým tónům přidat další.
- **Jako tréninová data použít prvních N taktů:** Při zaškrtnutí tohoto přepínače bude klasifikace melodických tónů provedena pouze pro stanovený počet taktů. Toho lze využít například pro zkrácení vzorových dat pro genetický algoritmus učení.
- **Používat data o čísle kanálu tónů:** Určuje, zda se budou aplikovat kritéria kanálů melodických a harmonických tónů.

Tlačítka jednotlivých funkcí:

- **Klasifikace melodických tónů:** Spustí klasifikaci melodických tónů na základě nastavených kritérií a prahové hodnoty. Výsledek se zobrazí v okně Notová osnova.
- **Načti parametry vah, Ulož parametry vah:** Umožňuje uložit a načíst nastavené váhy kritérií a všechny jejich parametry.
- **Vynuluj:** Nastaví počáteční hodnoty vah kritérií a jejich parametrů.
- **Genetický algoritmus učení:** Spustí genetický algoritmus učení popsany v kapitole 6. Parametry genetického algoritmu je možné nastavit po stisknutí tlačítka Parametry. V průběhu algoritmu dochází ke zkoušení různých nastavení vah kritérií a parametrů. Při potvrzení ukončení výpočtu program nastaví nejlepší nalezené řešení vah a parametrů přímo do ovládacích prvků okna.

8.4. Okno Ladítka

Obr. 60: Okno Ladítka

Ladítka

Druh ladění

- ☒ Rovnoměrně temperované ladění
- ☐ Přirozená ladění - použít dlaždice
- ☐ Vlastní ladění

Výstupní ladění

	-100 cent	+100 cent	Hz
c	<input type="text"/>	<input type="text"/>	0 261,63
c#	<input type="text"/>	<input type="text"/>	0 277,18
d	<input type="text"/>	<input type="text"/>	0 293,66
d#	<input type="text"/>	<input type="text"/>	0 311,13
e	<input type="text"/>	<input type="text"/>	0 329,63
f	<input type="text"/>	<input type="text"/>	0 349,23
f#	<input type="text"/>	<input type="text"/>	0 369,99
g	<input type="text"/>	<input type="text"/>	0 392,00
g#	<input type="text"/>	<input type="text"/>	0 415,30
a	<input type="text"/>	<input type="text"/>	0 440,00
a#	<input type="text"/>	<input type="text"/>	0 466,16
b	<input type="text"/>	<input type="text"/>	0 493,88

Navrhovaný název ladění:
Rovnoměrně temperované

Název ladění:
Rovnoměrně temperované

Automatický absolutní posun
☐ Nastavit absolutní posun aby fixoval tón
(pevné nastavení frekvence tónu na temperované ladění)
Z toho plyne abs. posun o: 0 centů

Ruční absolutní posun

Toto okno umožňuje výběr druhu ladění i nastavení vlastního ladění. Hodnoty posuvných ovládacích prvků jednotlivých tónů udávají změnu výšky v centech oproti rovnoměrnému temperovanému ladění. Tyto hodnoty jsou rozhodující pro vkládání ladění do skladby.

Okno obsahuje následující ovládací prvky:

Přepínače:

- **Rovnoměrně temperované ladění:** Nastaví všechny posuvníky na stejnou hodnotu, nastaví tedy rovnoměrně temperované ladění.
- **Přirozená ladění – dlaždice:** Nastaví hodnoty podle tónů vybraných v okně Dlaždice.
- **Vlastní ladění:** Umožňuje vlastní nastavení.
- **Automatický absolutní posun:** Posouvá hodnoty posuvníků tak, aby měl vybraný tón nulový posun oproti temperovanému ladění.
- **Ruční absolutní posun:** Umožňuje posun všech posuvníků o nastavenou hodnotu.

Tlačítka jednotlivých funkcí:

- **Načíst / Uložit ladění:** Nastavené ladění se ukládá do souboru včetně informací o nastavení polohy dlaždic. Součástí uložených informací je také název ladění.
- **Použít název:** Nastaví navrhovaný název ladění do editačního políčka.
- **Cyklický posun:** Posune všechny hodnoty posuvníků cyklicky o jedno místo dál.

8.5. Okno Dlaždice

Obr. 61: Okno Dlaždice

Přirozené ladění - dlaždice

377,22 -66fisis	282,93 -65cisis	424,39 -63gisis	318,28 -61disis	238,71 -59aisis	358,8 -57eisis	268,56 -55his	402,83 -53	302,12 -51	226,59 -49	339,89 -47	254,91 -45	382,38 -43	286,79 -41	430,18 -39	322,63 -37	241,97 -35
301,78 -53dis	226,33 -51ais	339,50 -49eis	254,63 -47his	381,94 -45fisis	286,45 -43cisis	429,68 -41gisis	322,27 -39disis	241,69 -37aisis	362,55 -35eisis	271,92 -33his	407,86 -31	305,91 -29	229,42 -27	344,13 -25	258,11 -23	387,15 -22
241,43 -39h	362,14 -37fis	271,61 -35cis	407,40 -33gis	305,56 -31dis	229,17 -29ais	343,75 -27eis	257,81 -25his	386,72 -23fisis	290,3 -22cisis	435,6 -20gisis	326,30 -18disis	244,72 -16aisis	367,9 -14eisis	275,32 -12his	412,97 -10	309,72 -8
386,29 -25g	289,72 -23d	434,57 -22a	325,93 -20e	244,44 -18h	366,67 -16fis	275,0 -14cis	412,50 -12gis	309,37 -10dis	232,3 -8ais	348,5 -6eis	261,3 -4his	391,56 -2fisis	293,67 0cisis	220,24 2gis	330,38 4dis	247,77 6ais
309,2 -12es	231,77 -10hes	347,65 -8f	260,75 -6c	391,11 -4g	293,33 -2d	440,0 0a	330,0 2e	247,50 4h	371,25 6fis	278,43 8cis	417,65 10gis	313,24 12dis	234,94 14ais	352,39 16eis	264,29 18his	396,44 20fisis
247,23 2ces	370,83 4ges	278,12 6des	417,18 8as	312,89 10es	234,67 12hes	352,0 14f	264,0 16c	396,0 18g	297,0 20d	222,75 22a	334,12 23e	250,60 25h	375,90 27fis	281,92 29cis	422,88 31gis	317,16 33dis
395,55 16asas	296,66 18eses	222,50 20hese	333,74 22fes	250,31 23ces	375,47 25ges	281,60 27des	422,40 29as	316,80 31es	237,60 33hes	356,40 35f	267,30 37c	400,95 39g	300,71 41d	225,53 43a	338,31 45e	253,72 47h
316,45 29	237,33 31	355,99 33gese	267,0 35dese	400,50 37asas	300,37 39eses	225,28 41hese	337,92 43fes	253,44 45ces	380,16 47ges	285,12 49des	427,68 51as	320,76 53es	240,57 55hes	360,85 57f	270,64 59c	405,97 61g
253,15 43	379,73 45	284,79 47	427,20 49	320,39 51	240,30 53	360,45 55gese	270,33 57dese	405,50 59asas	304,13 61eses	228,9 63hese	342,14 65fes	256,60 66ces	384,92 68ges	288,68 70des	433,2 72as	324,76 74es
405,5 57	303,79 59	227,84 61	341,75 63	256,32 65	384,48 66	288,35 68	432,54 70	324,40 72	243,30 74	364,96 76gese	273,71 78dese	410,58 80asas	307,93 82eses	230,94 84hese	346,43 86fes	259,82 88ces

Ladění
☒ Použít typické nastavení
☐ Vlastní nastavení

Vlastní nastavení
☒ Nastavit základní tón
☐ Nastavit tóny ve stupnici

Typické nastavení
 Tónorod:
☒ DUR
☐ MOLLI

Stupnice
☒ Přirozená diatonická stupnice
☐ Přiroz.stup.,chrom.tóny lokálně temperované
☐ Přiroz.stup.,chrom.tóny symetrické
☐ Přiroz.stup.,chromatické tóny Aristoxenes
☐ Přiroz.stup.,chromatické tóny Ptolemaios
☐ Přiroz.stup.,chromatické tóny Delezenne
☐ Pythagorejské ladění

Legenda:
 frekvence [Hz] → 412,50
 posun vzhledem k temper.lad.[cent] → -12
 název tónu ↓ gis
 Červeně je označen základní tón ladění
 Zeleně jsou označeny další použité tóny ladění
 Žlutě a bíle jsou označeny tóny, které aktuálně nejsou používány

Diagramy intervalů:
 čistá kvarta: $f \cdot 4/3$
 čistá kvinta: $f \cdot 3/2$
 malá tercie: $f \cdot 6/5$
 velká tercie: $f \cdot 5/4$
 tón frekvence f

Frekvence tónů je oktávovými posuny udržována v rozmezí 220 Hz - 440 Hz

Toto okno umožňuje výběr tónů z dlaždicového prostoru. Zobrazení tónů v dlaždicovém prostoru je podrobně popsáno v kapitole 2. Každé políčko obsahuje název tónu, jeho frekvenci a centový posun oproti rovnoměrnému temperovanému ladění. Právě tato hodnota centového posunu se v případě použití přirozeného ladění předává do posuvníků okna Ladítka.

Aktivní tóny dlaždicového prostoru jsou označeny červeně (základní tón) a zeleně (ostatní aktivní tóny). Při předávání dat do okna Ladítka se přenášejí informace o aktivních tónech. Pokud některý tón není aktivní, do příslušného posuvníku okna Ladítka se nastaví výška jako v rovnoměrném temperovaném ladění (výjimkou jsou lokálně temperované chromatické tóny – viz níže).

Žlutě a bíle jsou označeny tóny, které nejsou aktivní. Bíle jsou označeny tóny, které jsou již příliš vzdáleny tónině c-dur a jejichž název by tedy již byl příliš dlouhý.

Ovládací prvky umožňují výběr **typického nastavení**. Pak se kliknutím do dlaždicového prostoru mění pouze základní tón příslušného ladění.

Při zaškrtnutí volby *Přiroz.stup.chrom.tóny lokálně temperované* se pro diatonické tóny nastaví příslušný posun, zatímco pro chromatické tóny se nastaví centový posun jako průměr hodnot sousedních tónů ve stupnici.

Obr. 62: Výběr typického ladění a základního tónu:

Výběr přednastaveného ladění

Typické nastavení

Tónorod
☒ DUR
☐ MOLL

Stupnice
☐ Přirozená diatonická stupnice
☐ Přiroz.stup.,chrom.tóny lokálně temperované
☐ Přiroz.stup.,chrom.tóny symetrické
☐ Přiroz.stup.,chromatické tóny Aristoxenes
☒ Přiroz.stup.,chromatické tóny Ptolemaios
☐ Přiroz.stup.,chromatické tóny Delezenne
☐ Pythagorejské ladění

Výběr základního tónu

366,67 -16fis	275,0 -14dis	412,50 -12gis	309,37 -10dis	232,3 -8 ais	348,5 -6 eis
293,33 -2 d	440,0 0 a	330,0 2 e	247,50 4 h	371,25 6 fis	278,43 8 cis
234,67 12 hes	352,0 14 f	264,0 16 c	396,0 18 g	297,0 20 d	222,75 22 a
375,47 25 ges	281,60 27 des	422,40 29 as	316,80 31 es	237,60 33 hes	356,40 35 f

Pokud si vybereme volbu **Vlastní nastavení**, můžeme si při výběru volby *Nastavit tóny* přímo vybrat kliknutím myší, které tóny z dlaždicového prostoru chceme používat.

Obr. 63: Výběr vlastních tónů z dlaždicového ladění

305,56 -31dis	229,77 -29ais	348,75 -27fis	257,81 -25his	386,72 -23fisis	290,3 -22cisis
244,44 -18h	366,67 -16fis	275,0 -14cis	412,50 -12gis	309,37 -10dis	232,3 -8 ais
391,11 -4 g	293,33 -2 d	440,0 0 a	330,0 2 e	247,50 4 h	371,25 6 fis
312,89 10 es	234,67 12 hes	352,0 14 f	264,0 16 c	396,0 18 g	297,0 20 d
250,31 23 ces	375,47 25 ges	281,60 27 des	422,40 29 as	316,80 31 es	237,60 33 hes

9. Příloha C: Programátorská dokumentace

Software *Prelad* byl napsán a odladěn v prostředí Borland Delphi 7. Zdrojový kód software *Prelad* se skládá z unit, které jsou mezi sebou provázány. Strukturu jednotlivých unit znázorňuje obrázek č. 64 na následující straně. V programátorské dokumentaci jsou popsány nejdůležitější datové struktury a algoritmy použité v programu *Přelad*. Dokumentace obsahuje také dodatek s popisem vlastních formátů datových souborů.

9.1. Datové struktury

V následující části budou popsány důležité datové struktury použité v programu *Přelad*. Datové struktury budou popsány samostatně pro následující oblasti:

- Datové struktury pro definici skladby
- Datové struktury pro nastavení ladění
- Datové struktury pro klasifikaci melodických tónů
- Datové struktury genetického algoritmu

Poznámka: pro přehlednost zde uvádím datové struktury zjednodušeně – píšu u nich pouze důležité položky a důležité metody. Pomocné položky a metody zde v textu neuvádím. U některých metod zde vynechávám seznam parametrů, pokud není podstatný pro pochopení toho, co metoda provádí. Kompletní definice jsou samozřejmě součástí samotného zdrojového kódu. K jednotlivým položkám a metodám jsou ve zdrojovém kódu napsány komentáře tak, aby bylo zřejmé, k čemu slouží, a jak se používají.

9.1.1. Datové struktury pro definici skladby

Při popisu datových struktur pro definici skladby začneme základními typy, ze kterých pak budeme skládat typy složitější. Základním prvkem skladby v programu je abstraktní typ *TUdalost*, který zastřešuje různé typy událostí.

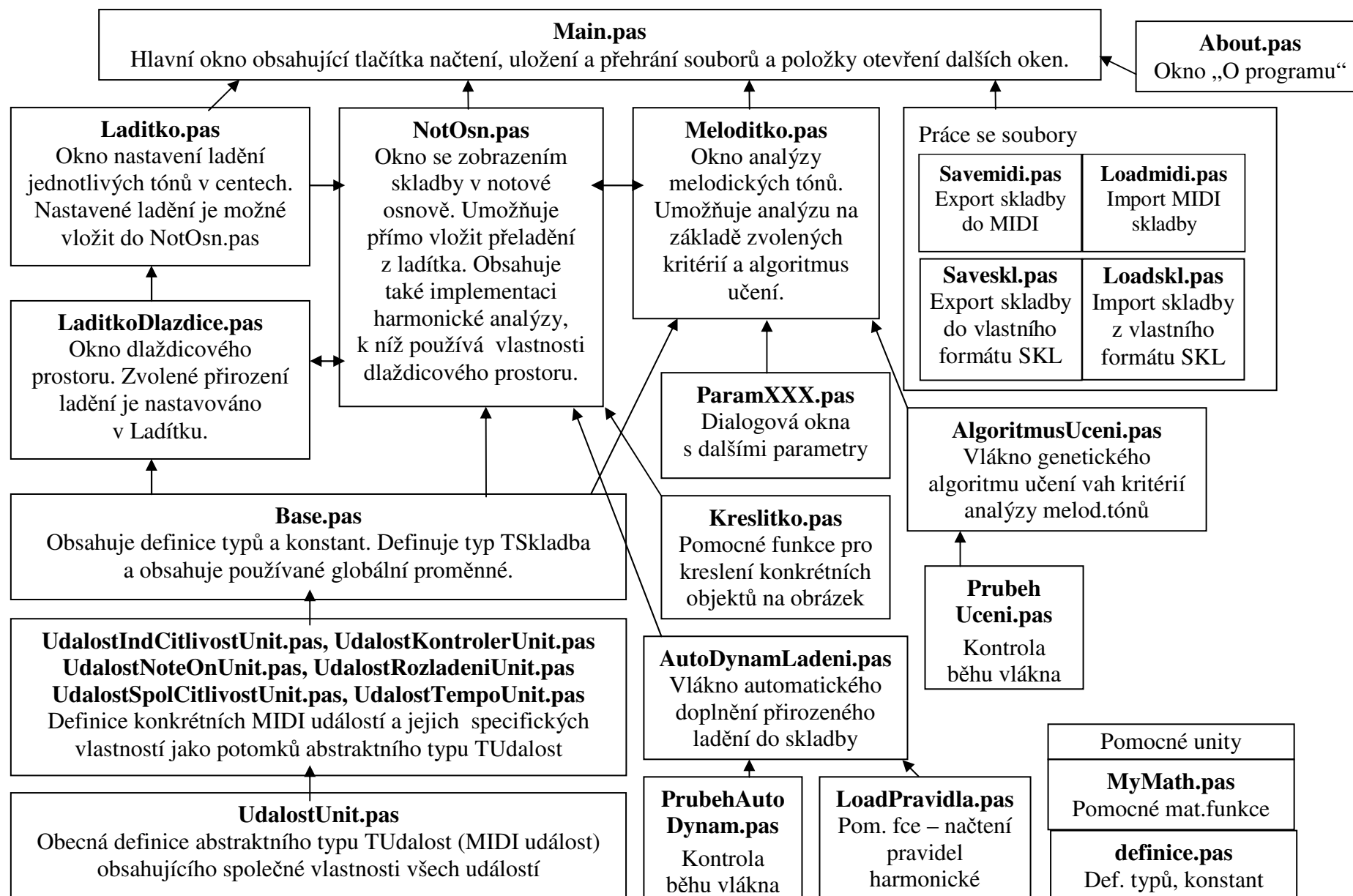
TUdalost zde není ekvivalentní MIDI události; ve většině případů sice *TUdalost* odpovídá MIDI události, není to ale pravidlem – jak si ukážeme, například typ *TUdalostPreladeni* je definován tak, že při zápisu do MIDI vytvoří celkem 12 MIDI událostí, které jsou zapsány za sebe.

Typ *TUdalost* je definován v unitě *UdalostUnit.pas*. *TUdalost* obsahuje společné vlastnosti všech událostí a to čas události a kanál, ve které byla událost zapsána. Společnými metodami pro všechny události jsou metody *VratDelku* a *VratRetezec* umožňující snadný zápis do SMF souboru, a také *NakresliSe* umožňující opět snadný zápis události do notové osnovy.

```
type TUdalost=class
  cas:longint;      {absolutní čas - počet tiků od začátku skladby}
  kanal:byte;      {MIDI kanál, který byl načten ze souboru}

  function VratDelku:longint; virtual; {vrátí svou délku zápisu v SMF}
  function VratRetezec:string; virtual; {vrátí výstupní řetězec do SMF}
  procedure NakresliSe(var image:TImage);virtual;
                                     {podle svého typu události volá příslušné
                                     funkce z unity Kreslitko.pas}
end;
```

Obr. 64: Struktura unit software *Prelad*



Potomky abstraktního typu `TUdalost` jsou následující události:

```
TUdalostNoteOn      {událost NoteOn}
TUdalostTempo       {metaudálost tempo}
TUdalostKontroler   {událost Kontroler}
TUdalostIndCitlivost {Individuální citlivost tónu - neimplementováno}
TUdalostSpolCitlivost {Společná citlivost tónu - neimplementováno}
TUdalostRozladieni  {soubor instrukcí přeladění}
```

Tyto události jsou definovány v samostatných unitách. Použité události si zde popíšeme:

```
type TUdalostNoteOn=class(TUdalost)
  cislonyty:byte;      {číslo noty: 0..127}
  sila:byte;          {síla: 0..127}
  delka:longint;      {vypočítávaná hodnota - délka v ticích}
  harmonie:THarmonie; {array[0..127] of boolean}
  zjednharmonie:TZjednHarmonie; {v rámci 1 oktávy:array[0..11] of bool.}
  melodickavaha:longint; {součet vah splněných kritérií melod.tónů}
  melodicka:boolean; {označení melod.tónů algoritmem-překročení prahu}
  uzivmelod:boolean; {označení melod.tónů uživatelem}
end;

type TUdalostTempo=class(TUdalost)
  tempo:longint; {počet mikrosekund za čtvrtovou dobu}
end;

type TUdalostKontroler=class(TUdalost)
  cislo:byte;      {číslo kontroleru}
  hodnota:byte;    {hodnota}
end;

type TUdalostRozladieni=class(TUdalost)
  rozladieni:TRozladieni;
end;
```

Typ `TRozladieni`, který je použit v třídě `TUdalostRozladieni`, je definován v unitě `Definice.pas` a má následující podobu:

```
type TRozladieni=record
  centposun:array[0..11] of integer; {pamatuje si centový posun tónů}
  rozl:array[0..11] of byte; {MSB-hlavní byte ohýbání tónu pro kanály}
  rozllb:array[0..11] of byte; {LSB-nižších 7 bitů}
  popis:string; {název přeladění}
  DefDlzdice:boolean; {pokud je to přirozené ladění z dlaždic:True}
  ... {další položky: informace o umístění v dlaždicovém prostoru}
end;
```

Datovou strukturou použitou pro uložení skladby je typ `TSkladba` definovaný v unitě `Base.pas`. Instance typu `TSkladba` obsahuje celou skladbu. Při načítání stop formátu 1 se používá pro každou stopu jedna instance `TSkladba`, přičemž poté jsou všechny stopy sloučeny do jedné.

Typ **TSkladba** obsahuje položky **metrum**, počet tiků za dobu, dopočítávanou hodnotu počtu tiků za takt, informace zda byl již načten takt, a především tři seznamy událostí.

```
type TSkladba=object
  metrum:TMetrum;      {určuje metrum}
  TikuZaDobu:word;      {počet tiků za jednu dobu}
  TikuZaTakt:word;      {dopočítávaná hodnota:=metrum.citatel*TikuZaDobu}
  TaktDefined:boolean;  {určuje, zda bylo z MIDI souboru načteno metrum}

  UdalostList:TUdalostList;          {obsahuje VŠECHNY udalosti}
  NotaList:TUdalostNoteOnList;      {obsahuje POUZE udalosti NOTA}
  HarmonList:TUdalostNoteOnList;    {obsahuje POUZE HARMONICKÉ TÓNY.
                                     Vytváří se až před automatickým
                                     doplněním rozladění}

end;
```

Následuje definice pomocného typu **TMetrum**. Obsah jeho položek je zřejmý; vychází z metaúdlosti SMF \$58 udávající metrum skladby.

```
type TMetrum=object
  citatel,jmenovatel:byte; {skutečná čísla čitatele a jmenovatele taktu}
  pocetsynchrotik:byte;   {počet MIDI synchrotik v 1 tukunutí metronomu}
  pocet32:byte;           {počet psaných 1/32 not v jedné čtvrtové (typicky=8)}
end;
```

Typ **TUdalostList** je seznam všech událostí skladby. **TUdalostList** je definován v unitě **UdalostUnit.pas** a je potomkem obecného typu **TClassList**.

```
type TUdalostList=class(TClassList)
  function AddUdalost(aUdalost: TUdalost): Integer;
  procedure InsertUdalost(Index:longint; aUdalost: TUdalost);
  function UdalostItems(Index:longint):TUdalost;
end;
```

UdalostList ve skladbě program vytváří při načítání MIDI nebo SKL souboru. Stejným způsobem se vytváří také událost **NotaList** typu **TUdalostNoteOnList**. Ta ale obsahuje pouze události typu **NoteOn**.

```
type TUdalostNoteOnList=class(TClassList)
  function AddNoteOn(aNoteOn: TUdalostNoteOn): Integer;
  procedure InsertNoteOn(Index:longint; aNoteOn: TUdalostNoteOn);
  function NoteOnItems(Index:longint):TUdalostNoteOn;
end;
```

Položka typu **TSkladby** s názvem **HarmonList** je také typu **TUdalostNoteOnList**, ale obsahuje pouze harmonické tóny. Tento seznam se na základě označených harmonických a melodických tónů vytváří vždy znovu na začátku automatické harmonické analýzy skladby.

V unitě Base.pas jsou definovány následující proměnné typu TSkladba:

```
var celaskladba:TSkladba; {celá skladba}
    stopa:array[1..32] of TSkladba; {stopy při načítání formátu 1}
```

Pokud je načítán MIDI formát 0, načte se skladba přímo do proměnné celaskladba. Pokud je načítán MIDI formát 1, načtou se jednotlivé stopy do proměnných Stopa a poté je provedeno sloučení dat do proměnné celaskladba.

9.1.2. Datové struktury pro nastavení ladění

Základní datovou strukturou pro nastavení ladění je typ TRozladieni, jehož popis byl již uveden v kapitole 9.1.1. Pro přehlednost jej uvádím znovu:

```
type TRozladieni=record
    centposun:array[0..11] of integer; {pamatuje si centový posun tónů}
    rozl:array[0..11] of byte; {MSB-hlavní byte ohýbání tónu pro kanály}
    rozllb:array[0..11] of byte; {LSB-nižších 7 bitů}
    popis:string; {název přeladění}
    DefDlazdice:boolean; {pokud je to přirozené ladění z dlaždic:True}
    ... {další položky: informace o umístění v dlaždicovém prostoru}
end;
```

Typ TRozladieni je použit jako součást typu TUDalostRozladieni. V unitě Base.pas je definována proměnná *predavanerozladieni* typu TRozladieni, která přímo udává hodnoty scrollbarů v okně Ladítka. Centový posun převzatý ze scrollbarů je uložen do položky *centposun*. Na základě této hodnoty program vypočte hodnoty *rozl* a *rozllb* udávající příslušné parametry události ohýbání tónu. Další položky typu TRozladieni udávají umístění základního tónu a ostatních aktivních tónů v dlaždicovém ladění.

V unitě LaditkoDlazdice.pas je pro definici dlaždicového prostoru použit následující typ definující jedno políčko:

```
TPolicko=object
    cislnoty:integer; {udává číslo noty 0..11}
    nazevnoty:string[7]; {zobrazovaný název noty}
    frekvence:real; {frekvence noty v rozmezí (220..440> Hz}
    centy:integer; {centový posun oproti temperovanému ladění}
    aktivni:boolean; {informace o tom, zda je toto políčko aktivní}
end;
```

Dlaždicový prostor je definován jako dvojrozměrné pole typu TPolicko:

```
var Policko:array[1..10, 1..17] of TPolicko;
```

Aktuální ladění v dlaždicovém prostoru je uloženo pomocí následujících proměnných:

```
var AktX,AktY:integer; {pozice základního tónu (červeně označený tón)}
    zaklton:integer; {číslo tohoto základního tónu (0..11, 0=c, 11=h)}
    Dlazdeni:array[0..11] of TLokace; {polohy aktivních tónů}
```

Pole Dlazdeni obsahuje relativní polohy aktivních tónů oproti základnímu tónu. Typ TLocace je definován v unitě Definice.pas jako typ record obsahující položky *r* a *s* (řádek a sloupec). Na první pozici (Dlazdeni[0]) je tedy poloha základního tónu vůči sobě, platí tedy, že Dlazdeni[0].r = 0 a Dlazdeni[0].s = 0.

Relativní jsou polohy v dlaždicovém prostoru i ve stupnici dané základním tónem, tedy na druhé pozici (Dlazdeni[1]) je poloha tónu o půltón vyššího, než je základní tón, atd.

Pokud některý tón není aktivní na žádném místě dlaždicového prostoru, je příslušná hodnota Dlazdeni rovna (0,0).

Při změně v dlaždicovém prostoru dochází k přepočítání následujícím způsobem: Všechny políčka dlaždicového ladění se nastaví jako neaktivní. Poté se políčko na pozici AktX a AktY označeno jako aktivní a jako základní tón. V jeho okolí jsou podle hodnot v poli Dlazdeni označeny jako aktivní další políčka.

Do okna Ladítka jsou následně přeneseny hodnoty centových posunů všech aktivních tónů. Pokud některý tón není na žádném místě dlaždicového ladění aktivní, nastaví se v okně Ladítka nulová hodnota centového posunu. Výjimkou je přirozená diatonická stupnice s lokálně temperovanými chromatickými tóny; v tom případě se hodnoty centového posunu u chromatických tónů nastaví jako aritmetický průměr okolních diatonických tónů stupnice.

9.1.3. Datové struktury pro klasifikaci melodických tónů

Při klasifikaci melodických tónů jsou procházen seznam Skladba.NotaList, tj. seznam všech událostí NoteOn. Pro každou tuto událost je u všech kritérií (viz kapitola 3) zkontrolováno, zda je příslušné kritérium splněno. Pokud ano, je jeho váha přičtena k proměnné *melodickavaha*, která je součástí typu TUdalostNoteOn. Tento typ je popsán v kapitole 9.1.1., položky hodnot pro klasifikaci melodických tónů zde pro přehlednost uvádím znovu:

```
type TUdalostNoteOn=class(TUdalost)
...
melodickavaha:longint; {součet vah splněných kritérií melod.tónů}
melodicka:boolean; {označení melod.tónů algoritmem-překročení prahu}
uzivmelod:boolean; {označení melod.tónů uživatelem}
...
end;
```

Pokud po součtu všech vah kritérií je hodnota *melodickavaha* vyšší nebo rovna než práh, je vlastnost melodicka nastavena na true. Jinak je nastavena na false. V případě, že je zaškrtnuta volba „Přidat ke stávajícím“, k nastavení na false nedochází. Tuto operaci provádí následující příkaz v unitě Meloditko.pas:

```
if celaskladba.NotaList.NoteOnItems(i).melodickavaha>=Threshold.Position
then celaskladba.NotaList.NoteOnItems(i).melodicka:=true
else if ZpusobMelod.ItemIndex=0
then celaskladba.NotaList.NoteOnItems(i).melodicka:=false;
```

Označení melodických tónů uživatelem je označeno v položce *uzivmelod*. Hodnoty položek *melodicka* a *uzivmelod* se používají při kreslení not, a především pro sestavení seznamu událostí harmonických tónů před algoritmem automatického návrhu změn ladění.

9.1.4. Datové struktury genetického algoritmu

Datové struktury genetického algoritmu jsou definovány v unitě AlgoritmusUceni.pas. Základním prvkem genetického algoritmu je jedinec:

```
type TJedinec=object
  Value:array[1..delkajedince] of byte; {zakódované parametry}
  PocetHudMel, {počet melodických tónů označených uživatelem}
  PocetMelOk, {programem správně rozpoznané melodické tony}
  PocetMelNavic, {to,co uživatel neoznačil a program ano}
  PocetNeoznac:longint; {co program neoznačil a uživatel ano}
  PocetChyb:longint; { = PocetMelNavic + PocetNeoznac }
end;
```

Způsob kódování parametrů popisuje tabulka v kapitole 6.2.1., pro přehlednost ji na následující straně uvádím znovu jako tab. 15.

Populace je definována jako pole jedinců, její velikost udává proměnná PocetJedincu. Nejlepší dosud nalezený jedinec se zapamatovává do proměnné BestJedinec.

```
type TPoleJedincu=array [1..MaxJedincu] of TJedinec;
var Jedinec:TPoleJedincu;
    PocetJedincu:longint;
    BestJedinec:TJedinec;
```

Při křížení jedinců dochází ke kombinaci odpovídajících hodnot pole Value na základě pravděpodobností definovaných jako parametry genetického algoritmu.

Tab. 15: Kódování parametrů analýzy melodických tónů do řetězce bytů

Pořadí Bytu	Kódovaný parametr	Hodnoty váhy / parametru	Způsob dekodování genu do parametru
1.	tvorba sekundy v harmonii	0..255	$\text{váha} := \text{gen}$
2.	soprán – tón vyšší než ..	0.255	$\text{váha} := \text{gen}$
3.	parametr tónu pro soprán – tón vyšší než ..	0..127	$\text{parametr} := \text{gen} \text{ div } 2$
4.	soprán – nejvyšší tón	0.255	$\text{váha} := \text{gen}$
5.	samostatnost	0..255	$\text{váha} := \text{gen}$
6.	tvorba sekundy v melodii	0..255	$\text{váha} := \text{gen}$
7.	těžká doba	-255..0	$\text{váha} := -\text{gen}$
8.	parametr – počet těžkých dob v taktu	1..8	$\text{parametr} := (\text{gen} \text{ div } 64) + 1$
9.	bas – tón nižší než ..	-255..0	$\text{váha} := -\text{gen}$
10.	parametr tónu pro bas – tón nižší než ..	0..127	$\text{parametr} := \text{gen} \text{ div } 2$
11.	bas – nejnižší tón	-255..0	$\text{váha} := -\text{gen}$
12.	nesamostatnost	-255..0	$\text{váha} := -\text{gen}$
13.	dlouhý tón	-255..0	$\text{váha} := -\text{gen}$
14.	parametr dlouhého tónu – počet dvaatřicetin	1..128	$\text{parametr} := (\text{gen} \text{ div } 2) + 1$
15.	kanály melod.tónů	0..255	$\text{váha} := \text{gen}$
16.,17	parametr – melodické kanály	16 bitů	jednotlivé bity reprezentují zaškrtnutí 16 kanálů
18.	kanály harmon.tónů	-255..0	$\text{váha} := -\text{gen}$
19.,20	parametr – harmonické kanály	16 bitů	jednotlivé bity reprezentují zaškrtnutí 16 kanálů
21.	práh	-255..255	$\text{hodnota} := (\text{gen} * 2) - 255$

Poznámka: U parametrů 3, 8, 10, 14 nevyužíváme plně potenciál genu, protože pomocí celého bytu kódujeme sedmibitové číslo (u parametru 8 dokonce pouze tříbitové). U parametru 21 zase ztrácíme na přesnosti a je možné zakódovat tak pouze lichá čísla. Naopak, výhodou tohoto způsobu je přehlednost při kódování parametrů; každý parametr je zakódován právě jedním bytem.

9.2. Algoritmy

V následující části budou popsány důležité algoritmy použité v programu Přelad'. Algoritmy budou popsány samostatně pro následující oblasti:

- Algoritmy načítání a ukládání MIDI
- Algoritmus automatické harmonické analýzy skladby
- Genetický algoritmus učení
- Některé pomocné výpočty

Poznámka: pro přehlednost zde uvádím algoritmy zjednodušeně – píšu pouze důležité příkazy a v některých případech používám zjednodušené pojmenování proměnných. U některých postupů vynechávám pomocné příkazy, pokud nejsou podstatné pro pochopení toho, co algoritmus provádí. Kompletní zápis je samozřejmě součástí samotného zdrojového kódu. K důležitým příkazům jsou ve zdrojovém kódu napsány komentáře tak, aby bylo zřejmé, k čemu slouží, a jak se používají.

9.2.1. Algoritmus načítání MIDI

Procedury a funkce pro načítání MIDI souborů jsou zapsány v unitě LoadMidi.pas. Hlavní procedurou při načítání MIDI je zde procedura *LoadMidiFile*. Její parametr *filename* udává název souboru, parametr *skladba* je proměnná typu TSkladba, do které se má načtená skladba zaznamenat.

Procedura *LoadMidiFile* otevře soubor a načte nejprve hlavičku MIDI. Pokud se jedná o MIDI soubor formátu 0, dojde následně k načtení jedné stopy. Pokud načítáme soubor MIDI formátu 1, pak se načtou jednotlivé stopy, které se nakonec sloučí v jednu. Procedura *LoadMidiFile* má následující tvar:

```
procedure LoadMidiFile(filename:string; var skladba:TSkladba);
begin
  NactiCislo(x,4);           {načte první 4 byty: MThd}
  NactiCislo(delka_hlavy,4); {načte 4 bytové číslo: délka hlavičky}
  NactiCislo(format,2);      {načte 2-bytové číslo: format 0/1}
  NactiCislo(pocetstop,2);   {načte 2-bytové číslo: pocet stop}
  case format of
    0:begin
      LoadTrack(skladba);    {načte jednu stopu přímo do proměnné skladba}
      skladba.VytvorNoteOnList;
    end;
    1:begin
      for i:=1 to pocetstop do LoadTrack(stopa[i]); {načte stopy}
      for i:=pocetstop downto 1 do MergeTracks(skladba, stopa[i] );
        {sloučí postupně všechny stopy do proměnné skladba}
      skladba.VytvorNoteOnList;
    end;
  end;
end;
```

Při načítání se používá pomocná funkce NactiCislo, která ze souboru načte binárně uložené číslo zadané délky:

```
procedure NactiCislo(var vysledek:longint; pocetbyte:byte);
{načte ze souboru číslo délky pocetbyte do proměnné vysledek}
```

Příkaz MergeTracks postupně prochází události dvou skladeb a zařazuje data z jedné skladby do druhé, tj. slučuje události do jedné stopy. Hlavička procedury má následující tvar:

```
procedure MergeTracks(var Skladba:TSkladba; var Skladba2:TSkladba);
{do proměnné Skladba přidá data z proměnné Skladba2}
```

Procedura *LoadTrack* načte z MIDI souboru jednu stopu. LoadTrack funguje následujícím způsobem: Nejprve načte hlavičku a délku stopy a poté v cyklu načítá jednotlivé události stopy:

```
NactiCislo(x,4); {nejprve načte hlavičku MTrk}
NactiCislo(delkastopy,4); {načte délku stopy}
While (file.Position < file.Size)
    and (LoadEvent(Skladba)) do begin end;
```

Funkce *LoadEvent* načte událost a přidá ji do proměnné Skladba. Funkce *LoadEvent* vrací False, pokud byla načtena metaudálost Konec stopy (viz kapitola 7.1.9.), jinak vrací True. Hlavička funkce LoadEvent má následující tvar:

```
function LoadEvent(var skladba:TSkladba):boolean;
{vrací false, pokud byl načten konec stopy, jinak vrací true}
```

Funkce LoadEvent načte typ události a poté rozdělí běh algoritmu do větví odpovídajících načtení jednotlivých typů událostí a metaudálostí.

9.2.2. Algoritmus uložení MIDI

Procedury a funkce pro ukládání MIDI souborů jsou zapsány v unitě SaveMidi.pas. Hlavní procedurou při načítání MIDI je zde procedura *SaveMidiFile*. Její parametr *filename* udává název souboru, parametr *skladba* je proměnná typu TSkladba, která se má uložit. Vzhledem k tomu, že při ukládání skladby je použit způsob, kdy jsou jednotlivé tóny rozděleny do příslušných kanálů, používá výstupní soubor pouze jeden nástroj. Jeho číslo je definováno pomocí proměnné *nastroj*.

Procedura *SaveMidiFile* nejprve vyplní proměnnou *ckan*, což je pole dvanácti čísel, udávající mapování jednotlivých tónů stupnice do kanálů. Nejjednodušší by bylo nastavit čísla kanálů od 0 do 11, ale vzhledem k možnosti použití kanálu bicích nástrojů je definováno toto mapování.

Procedura SaveMidiFile dále otevře soubor a zapíše do něj hlavičku SMF a hlavičku stopy. Soubor ukládáme ve formátu MIDI 0, tedy jako jednu stopu.

Poté už probíhá zápis jednotlivých událostí. Zde se plně využívá polymorfismu – v abstraktním typu TUDalost je definována virtuální funkce *VratRetezec*. Tato funkce je následně definována ve všech potomcích typu TUDalost jako override, tedy každý typ události má svůj vlastní způsob výpočtu návratové hodnoty. Funkce *VratRetezec* vrací přímo řetězec znaků pro zápis do MIDI (bez deltačasu, ten se zapisuje samostatně).

Procedura *SaveMidiFile* má následující tvar:

```
procedure SaveMidiFile(filename:string;Skladba:TSkladba;nastroj:byte);
begin
  {nejprve proběhne stanovení rozdělení tónů do kanálů}
  j:=0;
  for i:=0 to 11 do begin {určíme rozdělení tónů do kanálů}
    if j=kanal_bic then j:=j+1; {kanál bicích nelze použít pro přeladění}
    ckan[i]:=j;
    j:=j+1;
  end;
  {poté probíhá zápis hlavičky souboru a stopy. Ukládáme ve formátu 0.
   Jsou také zapsány události nastavení nástroje a tempa}
  ...
  {poté se zapisují jednotlivé události z proměnné Skladba}
  for i:=0 to skladba.UdalostList.Count-1 do begin
    ZapisDeltacas;
    FileWrite( skladba.UdalostList.UdalostItems(i).VratRetezec );
  end;
  FileWrite(chr($00)+chr($FF)+chr($2F)+chr($00)); {konec stopy}
end;
```

9.2.3. Algoritmus automatické harmonické analýzy skladby

Algoritmus zde popíšeme z hlediska zápisu zdrojového kódu. Podrobné vysvětlení základních principů a postupů tohoto algoritmu je uvedeno v kapitole 5.

Algoritmus je umístěn v unitě AutoDynamLadeni.pas. Výpočet probíhá v samostatném vlákně TVypocetDynamLadeni.

Hlavní procedura se tedy jmenuje *TVypocetDynamLadeni.Execute*. V této proceduře dojde nejprve k načtení pravidel pro návrhy posunů změn ladění. Volá se také procedura *skladba.VytvorHarmonList*, která vytvoří seznam obsahující pouze harmonické tóny. S tímto seznamem algoritmus dále pracuje.

Algoritmus postupně prochází všechny harmonické tóny. Vždy nejprve nalezne vhodné místo pro případné vložení přeladění – pokud je například tato událost v souzvuku s dalšími událostmi, je nutné vložit příslušné přeladění před první událost tohoto souzvuku.

Poté se volá funkce *NastavLadeni*, která podle pravidel pro souzvuky nalezne vhodné ladění. Funkce NastavLadeni nastaví toto ladění na Ladítku. Pokud se toto ladění liší od nyní používaného ladění, dojde ke vložení přeladění.

```
procedure TVypocetDynamLadeni.Execute;
begin
  h:=0;
  while (h<skladba.HarmonList.Count-1) do begin
    {nejdříve se nastaví příslušné místo pro případné vložení přeladění}
    ...
    {poté zjistíme typ ladění a základní tón}
    ladeni:=
      NastavLadeni(skladba.HarmonList.NoteOnItems(h).ZjednHarmonie,
                    lastladeni );
    if ladeni<>lastladeni then begin {pokud je nalezené ladění nové...}
      skladba.VlozRozladeni(predavanerozladeni, u);
      {predavanerozladeni: sdílené v Base.pas, píše do něj Ladítka}
    end;
    lastladeni:=ladeni; h:=h+1;
  end; {end while}
end; {end procedure}
```

Funkce *NastavLadeni* má parametr *ZjednLadeni* typu `array[0..11] of boolean` udávající aktuálně znějící harmonii tónu. Funkce vrací nalezený základní tón odpovídající některému z pravidel souzvuku. Funkce obsahuje parametr *lastladeni* z důvodu, že pokud nenalezne vhodné ladění, vrátí původní hodnotu *lastladeni*, aby nedošlo ke změně ladění. Deklarace funkce *NastavLadeni* vypadá následovně:

```
function NastavLadeni (ZjednLadeni:array of boolean;lastladeni:integer)
:integer;
```

Funkce *NastavLadeni* se postupně pokouší aplikovat pravidla souzvuků na aktuálně znějící harmonii. Schematický zápis algoritmu postupného přikládání pravidel k aktuálně znějící harmonii je vysvětlen v kapitole 5, pro přehlednost zde tento zápis uvádím znovu:

```
var ZjednLadeni:array[0..11] of boolean; {vstup.data: akt.znějící harmonie}
SouzvukPravidla:array[0..11] of boolean; {vstup.data-tóny pravidla}
PocetTonuPravidla:integer; {vstupní data-počet TRUE hodnot v pravidle}
ZaklTonSouzvuku:integer; {pom.proměnná-průchod všemi pokusy}
j:integer; {pom.proměnná-pro ověření shody tónů}
shoda:integer; {pomocná proměnná pro výpočet
                počtu tónů pravidla obsažených v souzvuku}
begin
  for ZaklTonSouzvuku:=0 to 11 do begin {zkouší přikládat šablonu na
                                        všech 12 pozic tónů v oktávě}
    shoda:=0;
    for j:=0 to 11 do {kontrola všech dvanácti tónů}
      if (ZjednLadeni[j] = true) and
        (SouzvukPravidla[ (j-ZaklTonSouzvuku+12) mod 12] = true)
      then shoda:=shoda+1;

      if shoda=PocetTonuPravidla then begin
        ShowMessage('Shoda na tónu '+IntToStr(ZaklTonSouzvuku));
        Break; {Nalezena shoda akt.znějící harmonii s pravidlem}
      end;
    end; {end for}
  end;
```

Pokud je po skončení tohoto postupu hodnota proměnné *shoda* rovna hodnotě proměnné *PocetTonuPravidla*, pak byla nalezena shoda aktuálně znějící harmonie s pravidlem, přičemž základním tónem souzvuku je hodnota *ZaklTonSouzvuku*.

Pak funkce *NastavLadeni* vrátí toto číslo. Pokud pro dané pravidlo není nalezena shoda, zkouší se další pravidlo. Pokud souzvuk neodpovídá při žádném základním tónu žádnému pravidlu, vrátí funkce *NastavLadeni* parametr *Lastladeni* – v tom případě nedojde ke změně ladění.

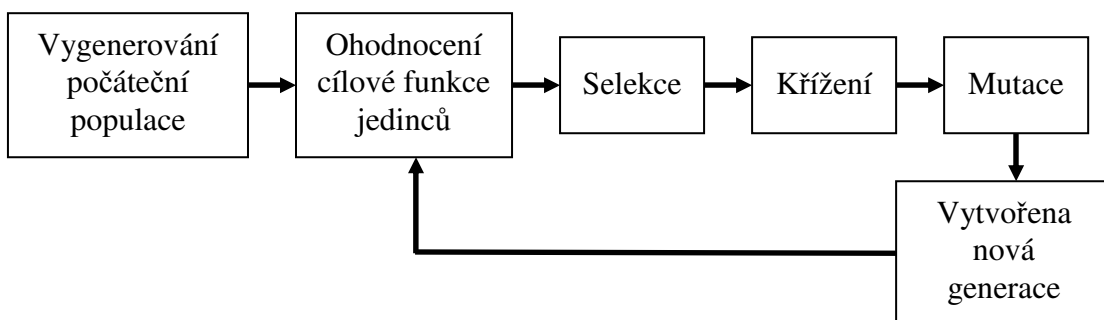
9.2.4. Genetický algoritmus pro stanovení vah kritérií

Algoritmus zde popíšeme z hlediska zápisu zdrojového kódu. Podrobné vysvětlení základních principů a postupů tohoto algoritmu je uvedeno v kapitole 6.

Algoritmus je umístěn v unitě *AlgoritmusUceni.pas*. Výpočet probíhá v samostatném vlákně *TAlgoritmusUceni*.

Hlavní procedura se tedy jmenuje *TAlgoritmusUceni.Execute*. Tato procedura si nejprve uloží potřebné parametry a poté probíhá výpočet v následujícím cyklu:

Obr. 65: Schéma práce genetického algoritmu



Zkrácený zápis zdrojového kódu funkce *TAlgoritmusUceni.Execute* vypadá následovně:

```

procedure TAlgoritmusUceni.Execute;
begin
  {nejprve si do svých proměnných uložíme parametry}
  ...
  {poté inicializujeme používané proměnné}
  BestJedinec.PocetChyb:=MaxLongInt; {aby ho na začátku každý přebil}
  Generace:=0;
  {vygenerujeme náhodnou počáteční populaci a ohodnotíme ji}
  for i:=1 to PocetJedincu do begin
    Jedinec[i].GenerujSeNahodne; {vytvoří náhodného jedince}
    Jedinec[i].DekodujSeDoMeloditka; {nastaví jej do okna Meloditko}
    Jedinec[i].OhodnotSe; {na základě okna Meloditko provede hodnocení}
    if Jedinec[i].PocetChyb < BestJedinec.PocetChyb then
      BestJedinec:=Jedinec[i];
  end; {end for}
  while not terminated do begin
    {--SELEKCE--}
    {nejprve setřídíme jedince podle kvality-elita bude na začátku}
    BubbleSort(Jedinec,PocetJedincu); {setřídí jedince podle PocetChyb}
    {nyní pomocí pole Kill určíme, kteří jedinci přežijí a kteří ne}
    for i:=1 to PocetJedincu do Kill[i]:=false; {pro začátek:všem milost}
    for i:=1 to PocetJedincu - PocetPrezivajících do begin
      ...{ruletový výběr-vyřazování jedinců (nastavování Kill:=true).Není
        dovoleno vyřadit elitu.U ostatních jedinců: Jedinci s horší
        cílovou funkcí mají větší pravděpodobnost vyřazení}
    end; {end for}
    {--KŘÍŽENÍ+MUTACE--}
    {jedinci, kteri maji KILL=true budou nahrazeni novymi}
    for i:=1 to PocetJedincu do
      if Kill[i] then begin
        Jedinec[i].GenerujSe...;
        {podle příslušných pravděpodobností vygeneruje nového jedince
          náhodně nebo křížením s příslušnou pravděpodobností mutace}
        Jedinec[i].DekodujSeDoMeloditka; {nastaví jej do okna Meloditko}
        Jedinec[i].OhodnotSe; {na základě okna Meloditko provede hodnocení}
        if Jedinec[i].PocetChyb < BestJedinec.PocetChyb then
          BestJedinec:=Jedinec[i];
      end; {end if Kill[i]}
    end; {end while}
  end; {end procedure}

```

Poté, co uživatel stiskne tlačítko Použít, algoritmus se ukončí a nejlepší nalezené řešení se nastaví do Melodítka.

9.2.5. Některé pomocné výpočty

Algoritmy pro kódování a dekódování deltačasu

V programu používám pro dekódování deltačasu následující algoritmus (pro přehlednost je v následujícím kódu použita fiktivní funkce NactiByte, která postupně načítá jednotlivé byty SMF souboru)

Vzorec: Délka kódovaného deltačasu

Deltačas označme proměnnou DeltaTime, požadovaný počet bytů SMF pro kódování deltačasu označme NumberOfBytes. Protože každých započatých 7 bitů čísla DeltaTime zabere celý byte v SMF a $2^7=128$, platí následující vztah:

$$\text{NumberOfBytes} = \lceil \log_{128} \text{DeltaTime} \rceil$$

Algoritmus: Kódování deltačasu

Číslo udávající deltačas zakódujeme do jednotlivých bytů pomocí následujícího algoritmu:

```
var DeltaTime:longint;           //udává kódovanou hodnotu deltačasu
    b:array[1..n] of byte;       //výstupní posloupnost bytů
    NumberOfBytes:longint;       //počet zakódovaných bytů
    i:longint;                   //pomocná proměnná
begin
    NumberOfBytes:= $\lceil \log_{128} \text{DeltaTime} \rceil$ ; //horní celá část logaritmu
    for i:=NumberOfBytes downto 1 do begin
        b[i]:=(DeltaTime mod 128); //posledních 7 bitů
        DeltaTime:=DeltaTime div 128; //posun o 7 bitů doprava
        b[i]:=b[i]+128;
        //první bity všech bytů budou nastaveny = 1
    end;
    b[NumberOfBytes]:=b[NumberOfBytes]-128;
    //poslední byte v pořadí bude mít první bit roven 0,
end;
```

Algoritmus: Dekódování deltačasu

Informace je zakódována v SMF souboru. Předpokládejme, že funkce NactiCislo vrátí vždy následující byte SMF souboru a postoupí v něm na další byte.

První bity všech bytů deltačasu jsou rovny jedné, kromě posledního bytu, který má první bit roven nule. Dekódování probíhá pomocí následujícího algoritmu, který výsledek umístí do proměnné **vysledek**:

```
var nacteno:byte;                //načtený byte
    pocitano:byte;              //pomocná proměnná-7-bitová hodnota pro načtení
    vysledek:longint;           //dekódovaná hodnota deltačasu
begin
    vysledek:=0;
    repeat
        vysledek:=vysledek shl 7; //posun na vyšší řád,pokračujeme nižším
        b:=NactiCislo; //postupné načítání bytů
        //ve všech bytech je první bit 1,v posledním bytu 0
        if nacteno>=$80 then
            //pokud je první bit roven 1, odstíníme ho odečtením
            pocitano:=nacteno-$80
        else
            //pokud je první bit roven 0, počítáme s touto hodnotou
            pocitano:=nacteno;

        vysledek:=vysledek+pocitano; //přičteme (7-bitové) číslo
    until nacteno<$80; //pokud byl první BIT nulový, cyklus se ukončí
```

Přepočet frekvence na centový posun oproti temperovanému ladění

Označíme-li si jednotlivé stupně centové míry čísla, můžeme pro ně stanovit frekvence. Přiřadíme MIDI notě č. 0 číslo stupně centové míry 0. Pak frekvence MIDI noty 1 bude odpovídat stupni centové míry 100, atd. Čísla centových stupňů MIDI not uvádí následující tabulka.

Tab. 16: Čísla centových stupňů pro MIDI noty

Poř.číslo oktávy	Ozn. oktávy	Hudební označení	C	C#	D	D#	E	F	F#	G	G#	A	A#	H
0	-2	sub-subkontra	0	100	200	300	400	500	600	700	800	900	1000	1100
1	-1	subkontra	1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200	2300
2	0	kontra	2400	2500	2600	2700	2800	2900	3000	3100	3200	3300	3400	3500
3	1	velká	3600	3700	3800	3900	4000	4100	4200	4300	4400	4500	4600	4700
4	2	malá	4800	4900	5000	5100	5200	5300	5400	5500	5600	5700	5800	5900
5	3	jednočárkovaná	6000	6100	6200	6300	6400	6500	6600	6700	6800	6900	7000	7100
6	4	dvoučárkovaná	7200	7300	7400	7500	7600	7700	7800	7900	8000	8100	8200	8300
7	5	tříčárkovaná	8400	8500	8600	8700	8800	8900	9000	9100	9200	9300	9400	9500
8	6	čtyřčárkovaná	9600	9700	9800	9900	10000	10100	10200	10300	10400	10500	10600	10700
9	7	pětičárkovaná	10800	10900	11000	11100	11200	11300	11400	11500	11600	11700	11800	11900
10	8	šestičárkovaná	12000	12100	12200	12300	12400	12500	12600	12700				

Nyní si definujeme posloupnost f_i – jako posloupnost frekvencí jednotlivých mikrotónů. Označme tedy f_i jako frekvenci i -tého stupně centové stupnice. Mezi sousedními tóny centové stupnice platí následující vztahy:

$$f_{i+1} = \sqrt[1200]{2} \cdot f_i$$

Tento vzorec můžeme zobecnit pro libovolné dva prvky posloupnosti f_i :

$$f_{i+k} = (\sqrt[1200]{2})^k \cdot f_i = 2^{\frac{k}{1200}} \cdot f_i$$

Vycházíme-li tedy z frekvence 440 Hz pro stupeň 6900 (jednočárkované a, MIDI 69), získáme následující obecný vzorec pro výpočet frekvence tónu centové stupnice:

$$f_i = 440 \cdot (\sqrt[1200]{2})^{i-6900} = 440 \cdot 2^{\frac{i-6900}{1200}} \quad \text{kde } i \text{ je číslo stupně v centové posloupnosti (} i \text{ v intervalu } 0..12800)$$

Jelikož ale chceme převádět naopak frekvenci na centy, potřebujeme zjistit opačný vztah, tedy vyjádřit si i pomocí f_i . Vzorec zlogaritmujeme a upravujeme:

$$2^{\frac{i-6900}{1200}} = \frac{f_i}{440}$$

$$\frac{i-6900}{1200} = \log_2 \left(\frac{f_i}{440} \right)$$

Získáváme následující vztah:

$$i = \log_2 \left(\frac{f_i}{440} \right) \cdot 1200 + 6900$$

Tím získáme absolutní pozici stupně v centové posloupnosti. Pokud budeme chtít uvádět výsledek v celých centech, zaokrouhlíme jej matematicky na nejbližší celé číslo.

Pokud budeme chtít určit relativní posun v centech oproti příslušnému MIDI tónu, stačí nám znát číslo tohoto tónu (definujme jej jako proměnnou *cislonoty*).

Pak můžeme relativní posun vyjádřit tímto vzorcem:

$$posun = \log_2\left(\frac{f_i}{440}\right) \cdot 1200 + 6900 - cislonoty \cdot 100$$

Tento vzorec se používá v unitě LadiťkoDlaždice pro určení posunu výšek tónů oproti temperovanému ladění. Ve zdrojovém kódu je zápis následující:

```
centy:=(Round(ln(frekvence/440)/ln(2)*1200+6900)-cislonoty*100+1200)
      mod 1200;
if policko[pocrad,j].centy>600 then centy:=centy-1200;
```

Poznámka: oproti výše uvedenému vzorci je zde následující rozdíl: K dosažené hodnotě přičítáme 1200 a poté provádíme operaci modulo 1200. To proto, že dlaždicovém ladění jsou frekvence v rozmezí (220 Hz..440 Hz) a v některých případech může nižší hodnota udávat centový posun nahoru – např. frekvence 221 Hz je zvýšením tónu a. Vzhledem k provedení operace modulo tedy musíme dodatečně určit, zda se jedná o posun nahoru nebo dolů. To rozhodneme podle podmínky, zda se výsledná hodnota nachází výše než o půl oktávy (posun v centech vyšší než 600). Tato hodnota se převede na příslušnou zápornou hodnotu o oktávu níže.

9.3. Dodatek: Vlastní formáty datových souborů

9.3.1. Formát pravidel pro změnu ladění – Pravidla.dat

Soubor Pravidla.dat je textový. Postupně za sebou jsou uvedena pravidla, která program načítá tak dlouho, dokud nenarazí na konec souboru. Pravidla, která mají být zkoušena dříve, jsou uvedena na přednějších místech.

Každé pravidlo má následující tvar:

```
POCET_TONU
TON1 TON2 TON3 ...
POZ_X_TON1 POZ_Y_TON1
POZ_X_TON2 POZ_Y_TON2
POZ_X_TON3 POZ_Y_TON3
...
```

POCET_TONU je číslo udávající počet tónů pravidla

TON1 má vždy hodnotu 0 – je to relativní poloha základního tónu vůči sobě

TON2, TON3... jsou relativní hodnoty ostatních tónů akordu oproti základnímu tónu

POZ jsou relativní polohy políček v dlaždicovém prostoru oproti základnímu tónu.

Poznámka: Vždy tedy platí, že $POZ_X_TON1 = POZ_Y_TON1 = 0$.

9.3.2. Formát pořadí pokusů změn ladění – Zmenaladeni.dat

Soubor Zmenaladeni.dat je textový. Na prvním řádku je uvedeno číslo udávající počet pokusů. Následuje tolik řádků, kolik je pokusů.

Na každém tomto řádku jsou dvě čísla:

```
RAD SLO
```

Číslo RAD udává navrhovaný posun v řádcích, SLO ve sloupcích. Čísla mohou nabývat kladných i záporných hodnot. První pokus musí mít hodnotu 0 0 (základní variantou je neměnit ladění), poslední pokus by měl mít také hodnotu 0 0 (pokud se nenalezne vhodné ladění, měli bychom zůstat u původního).

9.3.3. Formát *.lad - uložené ladění

Soubory *.lad jsou textové a uchovávají informace o ladění nastaveném v Ladítku a v Dlaždicích. Formát je následující:

1. řádek - název ladění (string)
2. řádek - typ ladění (string) - výběr 3 možností: "Temperovane", "Dlazdice", "Vlastni"
3. řádek - ScrollRad (číslo)
4. řádek - ScrollSlo (číslo) - scrolling dlaždicového ladění
5. řádek - AktX (číslo)
6. řádek - AktY (číslo) - poloha zákl. tónu v dlaždicovém ladění
7. řádek - zaklton (číslo) - číslo základního tónu (0..11)

Dále následuje 12 dvojic řádků postupně pro všechny tóny 0..11 (pořadí od c do h):

řádek 1: název tónu (string)

řádek 2: obsahuje tři čísla: POSUN_V_CENTECH, DL_X, DL_Y

kde DL_X a DL_Y znamenají hodnotu proměnné „Dlazdení“ v dlaždicích udávající relativní polohu tónu v dlaždicovém prostoru oproti základnímu tónu. Pokud tón není aktivní, čísla DL_X a DL_Y jsou 0,0.

Poznámka: pokud je použit typ Temperované nebo Vlastní, pak hodnoty dlaždic mají uloženu hodnoty dlaždic, jaké byly v době ukládání. Pro výšku ladění jsou ale v tomto případě podstatné pouze centové posuny.

9.3.4. Formát *.vah – Parametry klasifikace melodických tónů

Formát souboru je textový. Soubor obsahuje na každém řádku jedno číslo, postupně obsahuje tyto informace:

- 12 řádků vah příznaků (postupně příznaky 1..12)
- parametr výšky kritéria sopránového tónu
- parametr počtu těžkých dob v taktu
- parametr výšky kritéria basového tónu
- parametr délky noty pro Délku noty
- 16 čísel 0/1 udávající zaškrtnutí kanálů pro melodické tóny
- 16 čísel 0/1 udávající zaškrtnutí kanálů pro harmonické tóny
- hodnota parametru prahu

9.3.5. Formát *.pga – Parametry genetického algoritmu

Formát souboru je textový. Soubor obsahuje postupně 13 řádků, na každém je jedno číslo a to postupně hodnoty scrollbarů z formuláře parametrů genetického algoritmu.

9.3.6. Formát *.skl – Vlastní formát pro uložení skladby

Formát souboru je textový. Následující popis formátu bude používat tyto konvence:

- Informace zapsané v uvozovkách jsou popisky (uvozovky nejsou součástí formátu), tato data jsou obsažena v souboru SKL.
- Informace v závorkách jsou hodnoty. Tato data jsou také obsažena v souboru SKL. Hodnoty se vážou k předchozímu popisku, pokud není řečeno jinak. Pokud se nachází více hodnot na jednom řádku a pokud není řečeno jinak, jsou odděleny mezerou.
- Informace, které jsou uvozeny znaky //, jsou komentáře a nejsou součástí definice SKL souboru.

Popis formátu SKL:

```
//-----  
//Nejprve popíšeme prvních 20 řádků souboru SKL:  
//-----  
"MetrumCitatel"  
(integer)  
"MetrumJmenovatel"  
(integer)  
"PocetSynchronik"  
(integer)  
"PocetDvTr"  
(integer) //počet dvaatřicetin v jedné čtvrt'ové  
"TikuZaDobu"  
(integer)  
"TikuZaTakt"  
(integer)  
"TaktDefined"  
(integer: 0/1)  
"Predznameni"  
(integer: -7..7)  
"Tonorod"  
(integer: 0=dur,1=moll)  
"PocetUdalosti"  
(integer)  
//-----  
//V souboru SKL následuje seznam událostí skladby  
//-----  
//Každá událost začíná řádkem obsahujícím "-----"  
//Každá událost obsahuje následujících 6 řádků:  
  
"Time"  
(integer) //absolutní čas události v ticích  
"Kanal"  
(integer)  
"TypUdalosti"  
(integer)  
  
//---Použité číslování typů událostí:  
//TypNota=1; TypRozladieni=2; TypTempo=3;  
//TypIndCitlivost=4; TypSpolCitlivost=5; TypKontroler=6;
```

//Podle typu následují různé informace. Následuje popis formátování různých typů událostí:

//-----1.TypNota

//na jednom řádku jsou dvě čísla

(Výška:integer) (Síla:integer)

//na druhém a třetím řádku se nachází po jednom znaku

(x/M) //udává nastavení hodnoty Melodická: „x“=false, „M“=true

(x/U) //udává nastavení hodnoty UzivMelod: „x“= false, "M"=true

//-----2.TypRozladeni

(Název:string) //1.řádek: název přeladění

"Vlastni/Dlazdice"

(integer) //0 znamená nastavení volby vlastní,

//1 znamená nastavení volby Přirozené ladění (dlaždice)

"ScrollDlazdice"

(scrollRad:integer) (scrollSlo:integer) //scroll v dlaždicích

"PoziceZaklDlazdTonu"

(x:integer) (y:integer) //souřadnice tonu v dlaždicích

"ZaklTon"

(zaklton:integer) //0..11

//následuje 12 řádků udávajících rozladění tónů 0..11. Na každém řádku je 5 čísel

//typu integer:

(PosunVCentech) (posunPitchMSB) (posunPitchLSB) (Dlazdeni.x) (Dlazdeni.y)

//kde Dlazdeni.x a .y udávají relativní posun v dlaždění oproti základnímu tónu

//-----3.TypTempo

(tempo:integer)

//-----4.TypIndCitlivost

//neimplementováni - neobashuje další data

//-----5.TypSpolCitlivost

//neimplementováno - neobashuje další data

//-----6.TypKontroler

//obsahuje dvě čísla typu integer (v rozemzí 0..127). První z nich je číslo kontroleru,

//druhé udává hodnotu kontroleru

(integer) (integer)

10. Seznam použité literatury

- [1] Forró Daniel: Svět MIDI. Grada Publishing 1997
- [2] Geist Bohumil: Akustika. Muzikus 2005
- [3] Holan Tomáš: Delphi v příkladech. Ben – technická literatura 2002
- [4] Kofroň Jaroslav: Učebnice harmonie. Editio Bärenreiter Praha 2002
- [5] Mikulka Pavel, Juhová Kateřina, Soukenka Jiří: Turbo Pascal 7.0. Grada 1993
- [6] Obdržálek Jan: Teorie ladění; multimediální prezentace 2005
- [7] Šíma Jiří, Neruda Roman: Teoretické otázky neuronových sítí. Matfyzpress 1997
- [8] Šín Otakar: Úplná nauka o harmonii, díl první. Hudební matice Umělecké besedy 1943
- [9] Tvrzský Filip: Systémy ladění klávesových nástrojů. HAMU

Nápověda vývojového prostředí

Borland Delphi 7.0 – Delphi Object and Component Reference

Zdroje informací na Internetu

<http://www.borland.com>: Oficiální stránky firmy Borland

<http://www.midi.org>: Oficiální server MIDI

http://www-kiv.zcu.cz/~herout/html_sbo/midi/toc.html: Základní informace o MIDI

<http://www.sankey.ws/miditune.html>: Příklad mikroladění v MIDI pomocí ohýbání tónu