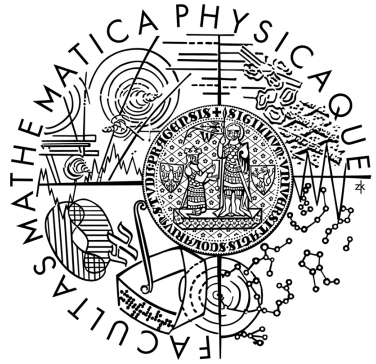
Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE



Michal Karásek

Tenisová databáze

Katedra softwarového inženýrství
Vedoucí bakalářské práce: RNDr. Michal Kopecký, Ph.D.
Studijní program: Aplikovaná informatika
2007

Děkuji RNDr. Michalu Kopeckému, Ph.D. za pomoc při tvorbě práce a zejména za trpělivost, kterou mi během té doby prokazoval.

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

V Praze dne

Michal Karásek

Obsah

Název práce: Tenisová databáze
Autor: Michal Karásek
Katedra (ústav): Katedra softwarového inženýrství
Vedoucí diplomové práce: RNDr. Michal Kopecký, Ph.D.
E-mail vedoucího: kopecky@ksi.ms.mff.cuni.cz

Abstrakt: Na základě naší představy o tenise, která je vymezena tenisovými pravidly a získanými empirickými daty, jsme vytvořili matematický model tenisového zápasu. Jeho chování je ovlivněno dvěma základními parametry: pravděpodobností výhry obou hráčů v rozehrách, v nichž podávají. To jsme ilustrovali na několika příkladech. Ukázali jsme metodu, jak zkombinovat reálné údaje o podání hráčů tak, abychom z nich mohli odvodit parametry modelu. Výsledky jeho počítačové implementace nám umožnily mimo jiné kvantifikovat odhady očekávaného počtu her v zápase, počtu a rozložení setů a výsledku zápasu. Simulovaná data jsme porovnávali se získanými empirickými daty. Zjistili jsme, že náš model předpovídá více her a obecně delší zápasy, než je ve skutečnosti pozorováno. Jedním z důvodů této odlišnosti, může být předpoklad statických pravděpodobností zisku rozehry během celého průběhu zápasu. Empirická data jsou získávána námi vytvořenými nástroji z publikovaných textových souborů na Internetu, které jsou převedeny do textové databáze. Tu lze snadno importovat do tabulkového kalkulátoru či prohlížet dalšími našimi aplikacemi.

Klíčová slova: tenis, matematický model, databáze, textový soubor ,U.S. Open

Title: Tennis database
Author: Michal Karásek
Department: Department of Software Engineering
Supervisor: RNDr. Michal Kopecký, Ph.D.
Supervisor's e-mail address: kopecky@ksi.ms.mff.cuni.cz

Abstract: In this thesis we made mathematical tennis match model, which is delimitated by the tennis rules and collected empirical data. The model behavior is determined by two parameters: probabilities of winning the rally by both players on serve. That was illustrated by examples. We showed the method how the real serving statistics data can be combined to derive model parameters. The results of its computer implementation allowed us to express numerically predictions of expected number of games, number and distribution of the sets, and match outcome. We compared simulated and real data. We have found that there are more games forecasted by our model than it was observed, that could be caused by the pre-requisite of static probabilities of winning the point during whole match. We collected empirical data using our tools from public available text files found on Internet. This data source is converted to one table database, which can be easily imported to spreadsheet program or our other developed applications.

Keywords: tennis, mathematical model, database, text file, U.S. Open

1 Úvod

V moderním světě jsou informace cenou komoditou. Míra užitečnosti informace stoupá s možností jejího začlenění do celku tak, aby lépe popisovala reálný svět. Zde se budeme snažit popsat, jak už z názvu práce vyplývá, svět tenisu.

Tato práce byla vytvářena po velmi dlouhou dobu. Prvotním motivací bylo získat výsledky tenisových zápasů z důvěryhodného zdroje tak, aby je bylo možno dále zpracovávat. Nejdříve s myšlenkou pouze předkládání dat ve srozumitelné formě lidské mysli. Později spíše s důrazem na prozkoumání matematických vlastností tenisového zápasu, kde lidská mysl má jen úkol porovnat matematické výsledky s realitou.

Cílem této práce tedy bylo vytvořit nástroj, který umožní přenesení specifických informací obsažených v strukturovaných textových souborech s výsledky tenisových zápasů do formátu vhodného k dalšímu zpracování. Data lze proto využít jako jeden z podkladů pro matematický model tenisového zápasu, který ve svém důsledku umožňuje předpovídat vítěze či jiné charakteristiky zápasu. Za pomoci uživatelského rozhraní může taková databáze sloužit jako zdroj historických výsledků.

Abychom čtenáři přiblížili tento abstraktní rámec, shrňme si, co nás čeká v dalších kapitolách. Druhá kapitola nám přiblíží pravidla tenisu a systém tenisových turnajů. Ve třetí kapitole se pokusíme zkonstruovat matematický model tenisového zápasu a ukázat jeho některé vlastnosti. V další kapitole si ukážeme, jak lze náš model aplikovat na reálná data, a provedeme simulaci výsledků tenisového zápasu. Pátá kapitola nás seznámí s tenisovými žebříčky a s některými dalšími metodami předpovídání výsledků. Šestá kapitola je věnována vyvinutému software, který slouží nejen k získání empirických dat, využitelných jako jeden z podkladů pro vývoj matematického modelu, ale také pro jejich zobrazení uživateli. V sedmé kapitole je shrnuto několik důležitých informací o vyvinutých aplikacích z hlediska uživatele.

2 Tenis

Chceme-li pochopit chování nějakého systému entit, musíme si o něm utvořit představu ve své mysli. Pokud však je daný systém složitý, nezbude nám nic jiného než se uchýlit k velkým zjednodušením. Postupný matematický popis jednoduchých pravidel, která se nám podaří v systému objevit, nám dává možnost odpoutat se od jeho složitosti a svou představu můžeme vytvořit dle chování matematického modelu. Systémy, které mají velkou dynamiku vývoje, je nejlepší pozorovat v nějaké počítačové reprezentaci modelu. Ze získaných výsledků pak můžeme lépe pochopit zákonitosti, které v modelu platí.

Přestože je tenis obecně známá hra, pro vytvoření plausibilního modelu tenisového zápasu je potřeba se důkladně seznámit se všemi aspekty hry. To mimo jiné předpokládá dostatečnou všeobecnou orientaci ve sportovním dění a jistý náhled do činností profesionálního sportovce. Pro přesné porozumění pravidlům tenisu je vhodné využít [1]. Systém losování a nasazování do turnajů je popsán např. v [2], [3]. Časté změny v systému výpočtu žebříčků ovšem činí tyto publikace v tomto ohledu neaktuální, proto je nezbytné použít [4].

2.1 Průběh tenisového utkání

Tenis je hra, kterou hrají dva hráči, v případě dvouhry („singles“), či čtyři hráči, v případě čtyřhry („doubles“). Úkolem hráčů je pomocí vypletené rakety udeřit plstí pokrytý gumový míček tak, aby se přes síť dostal do soupeřova pole. Míček do rozehry uvádí vždy jeden z hráčů („server“) na kříž do prostoru vymezeného v soupeřově části pole. Přijímající hráč („receiver“) musí mezi prvním a druhým dopadem míče vrátit míček jedním úderem přímo na druhou stranu. Podávající hráč nyní musí předtím než míč podruhé dopadne opět vrátit míč přes síť. Rozehra podobným stylem pokračuje a končí, jakmile jeden z hráčů nedokáže legálně vrátit míček na druhou stranu. Soupeři chybujícího hráče je připsán bod a hra pokračuje další rozehrou.

2.1.1 Standardní hra

V tenise, na rozdíl od většiny ostatních sportů, platí, že všechny rozehry během jedné hry zahajuje stále stejný hráč (s výjimkou uvedenou v části 2.1.2). Podání, na které má maximálně dva pokusy, provádí střídavě z pravé a z levé poloviny kurtu.

Struktura přidělování bodů je definována následovně. Standardní „hra“ („Game“) začíná za stavu „0-0“ („Love-all“). První bod je zaznamenán jako „15“, druhý jako „30“, třetí jako „40“. Pokud má první hráč, který získá čtvrtý bod, dva body náskok, získává hru. V opačném případě se hraje neomezeně dlouho až do doby, kdy jeden z hráčů získá dvoubodový náskok, a tím pádem i hru.

Stav, kdy oba hráči dosáhnou tří, čtyř či více bodů, je v tenisové terminologii nazýván „shoda“ („Deuce“). Po shodě má hráč, který vyhraje další bod „výhodu“ („Advantage“). Pokud ten samý hráč získá i další bod, pak vyhrává hru; pokud další bod vyhraje opačný hráč, pak je stav opět shoda. Hráč tedy musí po shodě vyhrát dva body za sebou, aby vyhrál hru.

2.1.2 Tie-break

Tie-break bývá občas v češtině nepřesně nazýván zkrácená hra. V průběhu tie-breaku jsou dosažené body uváděny jako 0, 1, 2, … Pokud první hráč, který získá sedm bodů, vede nad soupeřem rozdílem dvou bodů, vyhrává hru (a sadu). V opačném případě tie-break pokračuje, dokud není tohoto rozdílu dosaženo.

Hráč, který je na řadě s podáním podává první bod tie-breaku. Následující dva body podává soupeř. A oba hráči dále podávají střídavě dva body za sebou až do konce tie-breaku.

2.1.3 Sada

Existuje několik způsobů počítání her v sadě („set“). Dvě hlavní metody se nazývají „sada hraná s rozdílem dvou her" („Advantage Set“) a "tie-break sada" („Tie-break Set“). Pokud je použita metoda "tie-break sady", pak může být rozhodující závěrečná sada hrána jako "tie-break sada" nebo jako "sada hraná s rozdílem dvou her".

Do každého setu se vstupuje za stavu 0-0. V sadách hraných stylem „Advantage Set“ první hráč, který získá šest her, získává i danou sadu, za předpokladu, že vede nad soupeřem rozdílem alespoň dvou her. Jinak sada pokračuje, dokud není tohoto rozdílu dosaženo. Celkový počet her v sadě tedy není nijak omezen.

Tie-break sady byly zavedeny, aby odstranily tuto nepříjemnost. Počítání probíhá stejně jako u metody „Advantage set“, počet standardních her je však omezen na 12. Pokud je dosažen stav „6-6“, rozhoduje o vítězi sady tie-break (viz část 2.1.2).

2.1.4 Utkání

V utkáních hraných stylem „best of three“ vítězí ten, kdo první dosáhne dvou sad. Podobně, pokud se hraje stylem „best of five“, je vítězem ten, kdo získá první tři sady. Množství zápasů, které jsou tenisti nuceni během roku odehrát způsobuje, že většina utkání je hrána na dvě vítězné sady. Na tři vítězné sady jsou hrána jen Davis Cupová útkání, zápasy mužů na Grand Slamech a některá finále důležitých tenisových turnajů. Ženy hrají všechna svá utkání výhradně na dvě vítězné sady.

Většina utkání je hrána se všemi sadami počítanými jako „tie-break sady“. Jedině při zápasech Wimbledonu, Roland-Garros a Australian Open jsou případné páté sady stylem „advantage set“.

2.2 Tenisové turnaje

Téměř všechna soutěžní tenisová utkání jsou odehrána během turnajů. Ty obvykle trvají jeden týden, kdy první kolo je odehráno během pondělka či úterka, a finále se koná v neděli. Jedinou výjimkou jsou největší světové turnaje, čtyři Grand Slamy: Australian Open v Melbourne, Roland-Garros (French Open) v Paříži, Wimbledon na kurtech All England Lawn Tennis and Croquet Club v Londýně a U.S. Open ve Flushing Meadows v New York City, na jejichž odehrání jsou vyhrazeny dva týdny. Tenisová sezóna začíná v lednu, kdy se během léta na jižní polokouli koná první Grand Slam sezóny, a končí v listopadu v Šanghaji při turnaji mistrů.

Hráči na turnajích, které jsou rozděleny do několika kategorií, bojují kromě peněžitých odměn o body do žebříčku. Platí, že čím vyšší jsou vyplacené peněžité odměny, tím vyšší kategorie je turnaji přidělena, a tím lépe je turnaj bodově ohodnocen.

Protože počet účastníků je pevně dán, od 32 hráčů u malých turnajů po až 128 hráčů u Grand Slamů, jsou turnaje jsou hrány kaskádovým způsobem. To znamená, že hlavní soutěži předchází kvalifikace, která je dokonce u některých turnajů předcházena před-kvalifikací. Systém výběru do hlavní soutěže je pak následující. Z přihlášených hráčů je dle oficiálního žebříčku pro nasazování vybráno tolik účastníků, aby naplnili přibližně 90% losu. Několik dalších míst je vyhrazeno pro tzv. divoké karty („Wild Card“) udělované organizátory nadějným domácím hráčům či vracejícím se hvězdám. O zbytek volných míst se bojuje v kvalifikaci. Asi k nejznámějšímu úspěchu hráče s divokou kartou došlo roku 2001 na Wimbledonu, kde Goran Ivaniševič porazil ve finále dvouhry Patricka Raftera 6-3 3-6 6-3 2-6 9-7.

Turnaje se většinou hrají vyřazovacím způsobem. Aby bylo zajištěno rovnoměrnější rozložení kvality hráčů v pavouku, je čtvrtina nejlepších hráčů do turnaje nasazena. Nasazená jednička začíná na horní polovině pavouka, na opačném konci je vyhrazeno místo pro nasazenou dvojku. Podobně se postupuje, až jsou všichni nasazení hráči rozmístěni co nejdále od sebe přiřazeno pevné místo.

Úspěšnost hráčů v zápasech závisí mimo jiné na druhu hracího povrchu. Existují hráči, kterým lépe vyhovuje klouzavá a pomalá antuka, jiní zase preferují rychlými umělými povrchy pokrytý beton. Nejlépe podávajícím hráčům vyhovuje nejlépe styl „servis - volley“ typický pro travnaté dvorce.

3 Matematický model tenisového zápasu

Jeden z prvním modelů tenisu a jiných hierarchických her byl navržen v práci Kemeny a Snell [5]. Jimi popisovaný model měl pouze jeden parametr, pravděpodobnost zisku bodu podávajícím hráčem. Tato pravděpodobnost byla neměnná po celý zápas a nezávisela na podávajícím hráči. Fischer [6], Carter a Clarke [7] navrhli model, který více respektoval odlišnost mezi soupeři. Vyjádřili pravděpodobnost získání bodu hráčem jako průměr pravděpodobnosti výhry rozehry při podání a při příjmu. V moderním tenise je právo podávat velkou výhodou (viz např. tabulka 4.1), proto je pro věrohodnější přenesení reality do modelu nezbytné rozlišovat mezi schopnostmi obou hráčů na podání a příjmu. Hsi a Burych [8], Carter a Crews [9] vyjádřili algebraicky pravděpodobnost vítězství v sadě a zápase stanovením konstantní pravděpodobnosti výhry bodu při podání obou hráčů. Pollard [10] popsal další statisticky zajímavé údaje, jakými jsou například střední délka hry, sady a zápasu. Barnett a Clarke [11] zveřejnili návod, v němž na jednom příkladu ilustrovali, jak z údajů o úspěšnosti příjmu a podání soupeřů odvodit pravděpodobnosti výhry bodu při jejich vzájemném zápase. [12]

Všechny předchozí práce předpokládají, že rozehry v tenise se dají považovat za nezávisle stejně rozdělenou (iid) náhodnou proměnnou. Což znamená, že považují pravděpodobnosti výhry hráčů v jednotlivých rozehrách za konstantní. Některé další analýzy však naznačily, že tomu tak být nemusí. Magnus a Klassen [13] analyzovali 90 000 bodů odehraných během Wimbledonu a dokázali statistickou významnost jevu známého jako „first game effect“. Tedy, že nejtěžší je prolomit protivníkovo podání v první hře zápasu.

Jackson a Mosurski [14] na dalších podobných případech popsali, že výsledky rozehry v tenise nemusí být nutně nezávislé. Podařilo se jim totiž ukázat, že vítězství v rozehře má pozitivní vliv na úspěšnost v rozehře po ní bezprostředně následující. Tedy, že četnost výskytu výher po již vyhrané rozehře je vyšší než teoretická. Tento efekt je obvykle nazýván „hot-hand effect“. Zjistili také, že se v jimi analyzovaných datech objevuje „back-to-the-wall effect“, jak je nazývána (údajná) psychologická výhoda hráče, který je o bod zpět.

Magnus a Klassen [15] ve své další práci uvádějí, že body v tenise sice nejsou iid, nicméně lze toto rozdělení považovat za iid blízké. Je nutné si uvědomit, že důležitým aspektem přesnosti výsledků je velikosti analyzovaných dat. U menších souborů se může odchylka od iid projevit výrazněji. [16]

3.1 Markovovy řetězce

Model popsaný níže charakterizuje průběh tenisového zápasu jako řadu náhodných událostí, jejichž sémantický význam, přechod mezi jednotlivými stavy, je dán omezujícími podmínkami. Tento druh stochastického procesu je znám jako Markovovy řetězce [17]. Model je rozdělen do logických celků: modelu hry, modelu sady a modelu zápasu. Různorodost tenisových pravidel (viz část 2.1) způsobila, že bylo nutno zkonstruovat několik různých variant uvedených logických celků. Plausabilita modelu je ověřena v části 4.2 na základě dostupných empirických dat (viz kapitoly 6 a 7).

3.2 Model hry

V částech 2.1.1 a 2.1.2 jsme popsali průběh a pravidla počítání v jedné tenisové hře. Sestrojme nejdříve model standardní hry.

3.2.1 Model standardní hry

Definice 3.1: *Mějme dva hráče S a R. Nechť výsledek rozehry těchto dvou hráčů je výsledek náhodného pokusu s dvěmi možnými jevy. Vítězství hráče S, které nastává s pravděpodobnostmi* p_S^R*, či vítězství hráče R, které nastává s pravděpodobností* $1\text{-}p_S^R$.

Přestože jsou tenisové stavy počítány jako „love“, „15“, „30“, ..., zdá se praktičtější modelovat je jako „0“, „1“, „2“, ... Jediným problémem je jak budeme modelovat cyklus „shoda“ -> „výhoda“. Počet rozeher v jedné hře totiž nelze nijak shora omezit. Naštěstí existuje elegantní řešení. Aby hráč získal ze „shody“ hru, musí vyhrát dvě rozehry v řadě. To znamená, že pravděpodobnosti získání hry hráčem S resp. R jsou vůči sobě v poměru druhých mocnin pravděpodobnosti získání rozehry odpovídajícím hráčem (1). Rovnice (2) po jednoduché algebraické úpravě vyjadřuje pravděpodobnost vítězství jako:

$$P((4,3)\,|\,(3,3)) = \frac{1}{1+\frac{(1-p)^2}{p^2}} \qquad (\mathbf{1})$$

$$P((4,3)\,|\,(3,3)) = \frac{p^2}{2p^2-2p+1} \qquad (\mathbf{2})$$

Pravděpodobnost výhry druhým hráčem je zachycena rovnicí (3):

$$P((3,4)\,|\,(3,3)) = \frac{(1-p)^2}{2p^2-2p+1} \qquad (\mathbf{3})$$

Definice 3.2: *Modelem hry nazveme Markovův řetězec s následujícími parametry:*

dvojice (s,r) $0 \le s, r \le 4$ jsou stavy hry;

stav (s, r) ~ (0, 0) je počáteční stav;

přechodová funkce je definována:

(s, r) -> (s+1, r) s pravděpodobností p_S^R, pro s <4 & r<3;

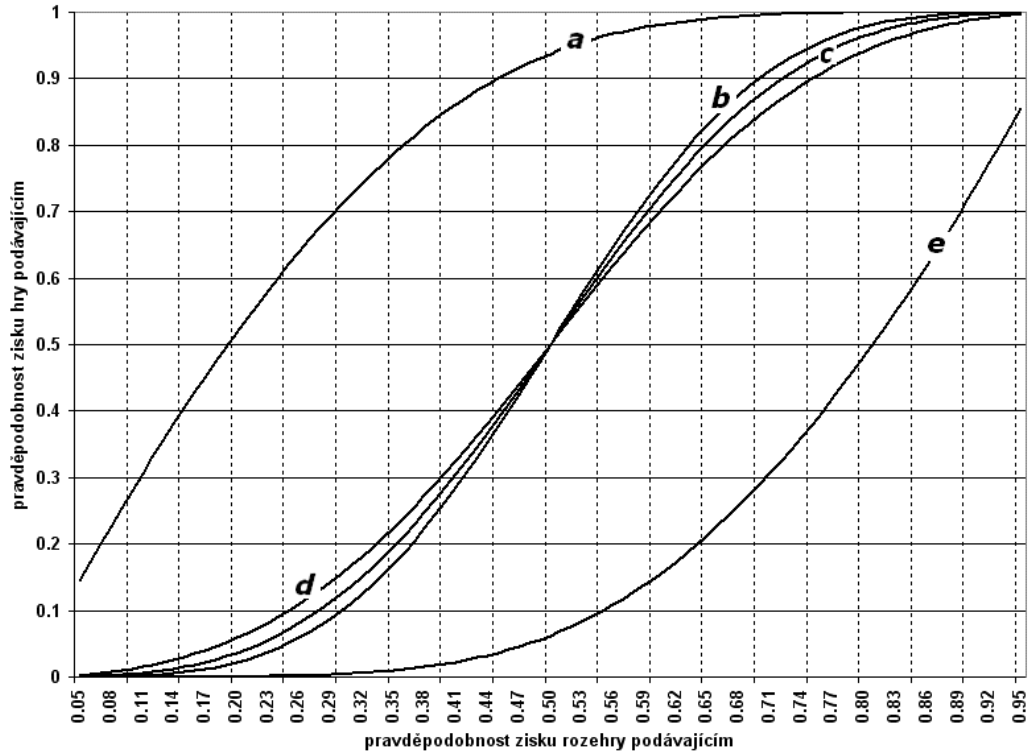
(s, r) -> (s, r+1) s pravděpodobností 1-p_S^R, pro r<4 & s<3;

(3,3) -> (4,3) s pravděpodobností dle (2), kde p = p_S^R;

(3,3) -> (3,4) s pravděpodobností dle (3)*, kde p = p_S^R.*

Absorpční jsou stavy (4,r), r<4 a (s,4), s<4. Pravděpodobnost p_S^G vítězství podávajícího ve hře lze vyjádřit jako: $\mathrm{p}_S^G = \sum_{r=0}^{3} P((4,r))$.

Takto zkonstruovaný konečný model se dá jednoduše přenést do tabulkového procesoru (viz také kapitola 6.3), což dovoluje snadno zjistit některé zajímavé hodnoty. Na obrázku 3.1 můžeme vidět křivky pravděpodobnosti zisku hry podávajícím v závislosti stavu a na pravděpodobnosti zisku rozehry.



Obrázek 3.1 - Rozložení pravděpodobnosti zisku hry podávajícím v závislosti na pravděpodobnosti zisku rozehry. Křivka odpovídající stavu a) 40:0 b) 0:0 c) 15:15 d) 30:30 (nebo shodě) e) 0:40.

3.2.2 Model tie-breaku

Nyní se pokusíme zkonstruovat podobný model i pro tie-break. Délku tie-breaku, podobně jako délku standardní hry nelze shora omezit, ale i zde platí, že použitím vhodné konstrukce lze tie-break modelovat konečným počtem stavů. Zásadní rozdíl mezi oběma modely bude způsobený tím, že se v tie-breaku oba hráči střídají na podání.

Mějme dva hráče A a B. Můžeme bez újmy na obecnosti předpokládat, že hráč A podává v tie-breaku první. Pravděpodobnosti, že hráč A resp. hráč B zvítězí v rozehře, ve které podává označme p_A^S resp. p_B^S.

Podobnou úvahou jako výše zjistíme, že rozložení pravděpodobností je při stavech „6-6“, „8-8“, „10-10“, ... pro oba hráče stejné. Aby hráč zvítězil v tie-breaku musí získat dvě rozehry v řadě, a to jednu při svém a druhou při soupeřově servisu. Stav „6,6“ můžeme tedy považovat za analogií „shody“ ve standardní hře. Z předchozího plyne, že poměr pravděpodobností vítězství hráčů ze stavu (6,6) je roven poměru pravděpodobností získání dvou rozeher jedním hráčem v řadě. To lze vyjádřit rovnicí (3). Teď již lehce zformulujeme rovnici (5), která vyjadřuje pravděpodobnost vítězství hráče A ze stavu (6,6). Rovnice (6) vyjadřuje totéž pro hráče B.

$$P((7,6)\,|\,(6,6)) = \frac{1}{1+\frac{p_B(1-p_A)}{p_A(1-p_B)}} \qquad (4)$$

$$P((7,6)\,|\,(6,6)) = \frac{p_A(1-p_B)}{p_A - 2p_Ap_B + p_B} \qquad (5)$$

$$P((6,7)\,|\,(6,6)) = \frac{p_B(1-p_A)}{p_A - 2p_Ap_B + p_B} \qquad (6)$$

Definice 3.3: *Modelem tie-breaku nazveme Markovův řetězec s následujícími parametry:*

dvojice (a,b) $0 \le a, b \le 7$ jsou stavy tie-breaku;

stav (a, b) ~ (0, 0) je počáteční stav;

přechodová funkce je definována:

*if (a + b = 4k+3) or (a + b = 4k) k ϵ **N** then (podává hráč A)*

(a, b) -> (a+1, b) s pravděpodobností p_A^S, pro a <7 & b<6;

(a, b) -> (a, b+1) s pravděpodobností 1-p_A^S, pro b<7 & a<6;

(6,6) -> (7,6) s pravděpodobností dle (5)*, kde $p_A = p_A^S$ a $p_B = p_B^S$;*

(6,6) -> (6,7) s pravděpodobností dle (6)*, kde $p_A = p_A^S$ a $p_B = p_B^S$;*

fi;

if (a + b = 4k+1) or (a + b = 4k+2) k ϵ ***N*** *then (podává hráč B)*

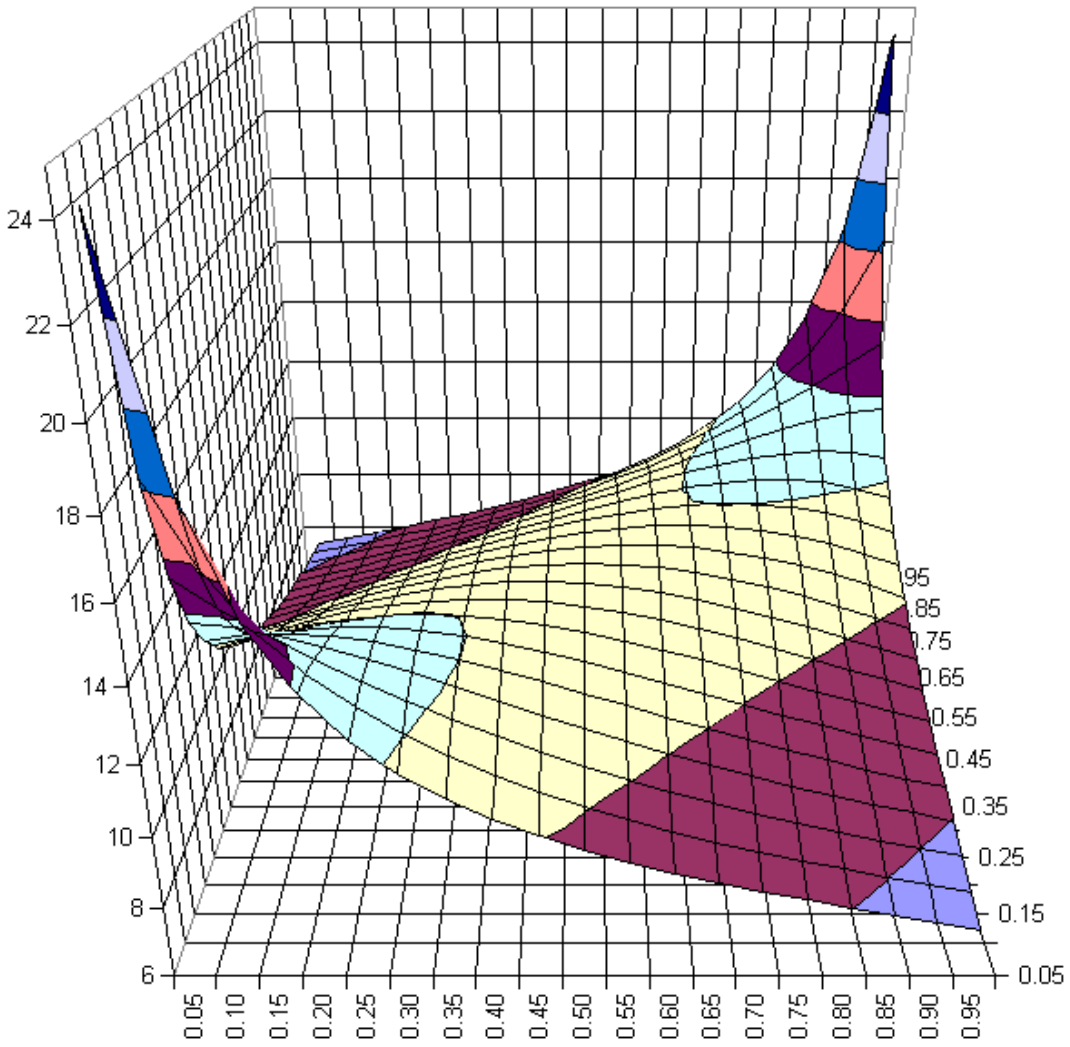
(a, b) -> (a+1, b) s pravděpodobností p_B^S*, pro a <7 & b<6;*

(a, b) -> (a, b+1) s pravděpodobností $1-p_B^S$*, pro b<7 & a<6;*

fi;

Absorpční jsou stavy (7,b), b<7 a (a,7), a<7. Pravděpodobnost p_A^T *vítězství prvního podávajícího v tie-breaku lze vyjádřit jako:* $\mathrm{p}_\mathrm{A}^\mathrm{T} = \sum_{b=0}^{6} P((7,b))$.

Na obrázku 3.2 můžeme vidět vizualizaci očekávaného počtu rozeher odehraných během tie-breaku. Můžeme si povšimnout, že se jejich počet zvyšuje s rostoucí vyrovnaností hráčů a se zvyšující se pravděpodobností výhry podávajícího hráče.



Obrázek 3.2 - Graf očekávaného počtu rozeher v tie-breaku (vertikální osa) v závislosti na pravděpodobnosti výhry podávajících hráčů v rozehře (horizontální osy).

3.3 Model sady

Nyní můžeme postoupit o úroveň výše k modelování sady. Opět lze bez újmy na obecnosti předpokládat, že v sadě začne podávat hráč A. Pravděpodobnost výhry podávajícího hráče ve hře označujeme p^G, tedy $p_A{}^G$ pro hráče A resp. $p_B{}^G$ pro hráče B, a pravděpodobnost výhry tie-breaku hráčem, který začne v tie-breaku podávat, označujeme p^T ($p_A{}^T$, $p_B{}^T$).

Definice 3.4: *Modelem tie-break sady nazveme Markovův řetězec s následujícími parametry:*

dvojice (a,b) $0 \le a, b \le 7$ *jsou stavy sady;*

stav (a, b) ~ (0, 0) je počáteční stav;

přechodová funkce je definována:

if (a + b) mod 2 = 0 then (podává hráč A)

(6,6) -> (7,6) s pravděpodobností $p_A{}^T$*;*

(6,6) -> (6,7) s pravděpodobností 1- $p_A{}^T$*;*

pro a <6 & b<6 nebo a + b = 11:

(a, b) -> (a+1, b) s pravděpodobností $p_A{}^G$*;*

(a, b) -> (a, b+1) s pravděpodobností 1-$p_A{}^G$*;*

fi;

if (a + b) mod 2 = 1 then (podává hráč B)

(a, b) -> (a+1, b) s pravděpodobností $p_B{}^G$*, pro a <6 & b<6;*

(a, b) -> (a, b+1) s pravděpodobností 1-$p_B{}^G$*, pro a <6 & b<6;*

fi;

Absorpční jsou stavy (6,b), b<5; (a,6), a<5; (7,5),(7,6),(5,7),(6,7). Pravděpodobnost $p_A{}^S$ *vítězství prvního podávajícího v sadě lze vyjádřit jako:*

$$p_A^S = \sum_{b=0}^{4} P((6,b)) + P(7,5) + P(7,6)\,.$$

Pravděpodobnost $p_A{}^{SSC}$ *vítězství prvního podávajícího v sadě spolu ze změnou prvního podávajícího v následující sadě je:* $p_A^{SSC} = P(6,1) + P(6,3) + P(7,6)\,.$

Konečně pravděpodobnost $p_A{}^{SrSC}$ *vítězství prvního přijímajícího v sadě spolu ze změnou prvního podávajícího v sadě následující je:* $p_A^{SrSC} = P(1,6) + P(3,6) + P(6,7)\,.$

Poslední a předposlední větou definice 3.4 jsme definovali pravděpodobnosti změny prvního podávajícího hráče v sadě, což nám umožní vytvořit model zápasu.

3.4 Model zápasu

Jak již bylo zmíněno v části 2.1.3, tenisové zápasy se hrají na dva nebo tři vítězné sady. I když se zápas zdá z hlediska složitosti tvorby matematického modelu nejjednodušší entitou, dělají ho pravidla změny podání o něco složitější. Opět můžeme předpokládat, že v sadě začne podávat hráč A. Pokud by nám vadilo, že v reálném tenisovém zápase dochází k losu, můžeme snadno model doplnit o stavy zajišťující los a volbu strany. Abychom se snadno mohli odkázat na již vymodelované, připomeňme p_A^S pravděpodobnost výhry sady hráčem, který v něm začne podávat jako první, p_A^{SSC} je pravděpodobnost téhož sdružená s pravděpodobností změny prvního podávajícího. Potom 1- p_A^S odpovídá zisku sady prvním přijímajícím a p_A^{SrSC} je pravděpodobnost téhož jevu spolu se změnou prvního hráče na podání v další sadě. Sestavme nyní model tie-break zápasu „best of five“.

Definice 3.5: *Modelem tie-break zápasu „best of five“ nazveme Markovův řetězec s následujícími parametry:*

trojice (a,b,s) $0 \le a, b \le 3$ jsou stavy zápasu, $0 \le s \le 1$ označuje podávajícího hráče, 0 odpovídá hráči A, 1 hráči B ;

stav (a, b) ~ (0, 0, 0) je počáteční stav;

přechodová funkce je (pro a < 3, b < 3) definována:

(a,b,0) -> (a+1,b,0) s pravděpodobností $p_A^S - p_A^{SSC}$;

(a,b,0) -> (a+1,b,1) s pravděpodobností p_A^{SSC};

(a,b,0) -> (a,b+1,0) s pravděpodobností $1-p_A^S - p_A^{SrSC}$;

(a,b,0) -> (a,b+1,1) s pravděpodobností p_A^{SrSC};

(a,b,1) -> (a,b+1,1) s pravděpodobností $p_B^S – p_B^{SSC}$;

(a,b,1) -> (a,b+1,0) s pravděpodobností p_B^{SSC};

(a,b,1) -> (a+1,b,1) s pravděpodobností $1-p_B^S - p_B^{SrSC}$;

(a,b,1) -> (a+1,b,0) s pravděpodobností p_B^{SrSC}.

Absorpční jsou stavy (3,b), b<3 a (a,3), a<3. Pravděpodobnost p_A^M vítězství hráče, který v zápase podával jako první lze vyjádřit jako:

$$p_A^M = \sum_{b=0}^{2} P((3,b,0)) + P((3,b,1)).$$

Tímto jsme dokončili tvorbu model tie-break zápasu „best of five“. Model tie-break zápasu „best of three“ je obměnou modelu definovaného v 3.5. Stačí jen snížit horní omezení dostupných stavů z tří na dva a upravit analogicky i přechodovou funkci a vyjádření celkové pravděpodobnosti.

Definice 3.6: *Modelem tie-break zápasu „best of three“ nazveme Markovův řetězec s následujícími parametry:*

trojice (a,b,s) $0 \leq a, b \leq 2$ jsou stavy zápasu, $0 \leq s \leq 1$ označuje podávajícího hráče, 0 odpovídá hráči A, 1 hráči B ;

stav (a, b) ~ (0, 0, 0) je počáteční stav;

přechodová funkce je (pro a < 2, b < 2) definována:

(a,b,0) -> (a+1,b,0) s pravděpodobností p_A^S - p_A^{SSC};

(a,b,0) -> (a+1,b,1) s pravděpodobností p_A^{SSC};

(a,b,0) -> (a,b+1,0) s pravděpodobností 1-p_A^S - p_A^{SrSC};

(a,b,0) -> (a,b+1,1) s pravděpodobností p_A^{SrSC};

(a,b,1) -> (a,b+1,1) s pravděpodobností p_B^S – p_B^{SSC};

(a,b,1) -> (a,b+1,0) s pravděpodobností p_B^{SSC};

(a,b,1) -> (a+1,b,1) s pravděpodobností 1-p_B^S - p_B^{SrSC};

(a,b,1) -> (a+1,b,0) s pravděpodobností p_B^{SrSC};

Absorpční jsou stavy (2,b), b<2 a (a,2), a<2. Pravděpodobnost p_A^M vítězství hráče, který v zápase podával jako první lze vyjádřit jako:

$$\mathrm{p}_\mathrm{A}^\mathrm{M} = \sum_{b=0}^{1} P((2,b,0)) + P((2,b,1)).$$

3.5 Shrnutí

S využitím Markovovo řetězců se nám podařilo zkonstruovat matematický model tenisového zápasu. Nejdříve jsme definovali model hry a tie-breaku, odkud jsme dále postupovali směrem ze zdola nahoru. Jakmile se nám podařilo matematicky vymezit pojem „tie-break sady“, mohli jsme konečně v definicích 3.5 a 3.6 popsat celý tenisový zápas.

4 Aplikace matematického modelu

V této kapitole si nejprve ukážeme jak lze získat z dat o úspěšnosti hráčů pravděpodobnost jejich výhry ve vzájemném zápase. Tím získáme metodu jak nastavovat parametry našeho matematického modelu, což nám umožní se pokusit o jeho aplikaci na reálná data.

4.1 Získání parametrů modelu

4.1.1 Osobní charakteristika hráčů

Barnett a Clarke [11] navrhují získat pravděpodobnosti výhry rozehry dvou konkrétních soupeřů následujícím postupem. Na internetových stránkách ATP www.atptour.com/en/media/rankings/matchfacts.pdf, lze každý týden nalézt data o aktuálních statistikách nejlepších 200 hráčů světa dle žebříčku ATP Ranking. Po vhodném zpracování publikovaných údajů lze získat úspěšnosti při prvním a druhém podání.

Výpočet úspěšnosti podávajícího hráče s_i je přímočaré, a je uvedeno v následující rovnici:

$$s_i = a_i b_i + (1 - a_i) c_i , \qquad (7)$$

kde pro hráče i je a_i úspěšnost prvního podání, b_i úspěšnost v rozehře po povedeném prvním podání (včetně es), c_i úspěšnost v rozehře po nepovedeném prvním podání (včetně dvojchyb).

Výpočet téže pravděpodobnosti r_i pro přijímajícího hráče je složitější. Úspěšnost prvního podání a_i jsme nuceni aproximovat průměrem všech ostatních hráčů a_{avg}. Nyní můžeme úspěšnost vyjádřit v rovnici:

$$r_i = a_{avg} d_i + (1 - a_{avg}) e_i , \qquad (8)$$

kde d_i je úspěšnost hráče i v rozehře při příjmu prvního podání a e_i je úspěšnost téhož hráče v rozehře při příjmu druhého podání. Pro data platná v době psaní této práce je koeficient a_{avg} roven 0,609.

4.1.2 Zápas dvou hráčů

Předchozí odstavce nám daly návod jak získat charakteristiky jednotlivých hráčů. Abychom mohli náš model aplikovat, potřebujeme mít možnost do něj vložit úspěšnosti odpovídající jejich potenciálnímu vzájemnému zápasu, které zatím neznáme. Jinými slovy musíme rozhodnout, co se stane, když dobře podávající hráč narazí na dobře přijímajícího. Tento problém s kvantifikací pravděpodobností dvou protichůdných charakteristik musí být nějakým způsobem řešen při v každé simulaci sportu. Podobnou otázku, vyjádření věrohodných pravděpodobnosti zisku rozehry

podávajícím respektive přijímajícím družstvem, pozorujeme během modelování volejbalového utkání. Je nutné zejména vyřešit komplementárnost obou pravděpodobností, tedy fakt, že součet úspěšnosti jednoho hráče na podání s úspěšností druhého na příjmu musí být roven jedné.

Dalším důležitým faktorem ovlivňující úspěšnost je druh hracího povrchu, což lze nahlédnout v tabulce 4.1, která uvádí statistiky všech čtyř Grand Slamů. Podávající hráči jsou nejúspěšnější na Wimbledonu, který se hraje na trávě, následují tvrdé kurty na Australian Open a U.S. Open. Nejhorším povrchem pro podávajícího je antuka na Roland-Garros. Rozdíl mezi U.S. Open a Australian Open je způsoben různými typy umělého povrchu. DecoTurf, pokrývající kurty ve Flushing Meadows, je rychlejší než v Melbourne Park položený Rebound Ace.

	Ženy				Muži			
	průměr	2002	2003	2004	průměr	2002	2003	2004
Roland-Garros	**53,2**	-	53,4	53,0	**60,0**	60,4	60,1	59,4
Australian Open	**54,9**	54,4	54,9	55,3	**62,1**	61,7	61,7	63,0
US Open	**56,1**	55,9	56,1	56,2	**62,8**	62,6	63,6	62,1
Wimbledon	**57,4**	57,0	58,0	57,2	**64,6**	63,8	64,4	65,6

Tabulka 4.1 - Úspěšnost podávajícího hráče v rozehře na jednotlivých Grand Slamech v letech 2002 - 2004, zdroj dat [12].

Výše zmiňované charakteristiky hráčů jsou utvářeny pro všechny povrchy. Aby náš model kvalitně popisoval tenisový zápas, je nutné jeho parametry upravovat v závislosti na místě jeho konání. Elegantně se tyto rozdíly podařilo vyřešit v již zmiňované [11]. Pravděpodobnost výhry podávajícího v rozehře lze vyjádřit jako součet:

- poměru bodů, které všichni hráči vyhrávali v minulosti při svých podáních na turnaji, kde je simulovaný zápas odehrán;
- rozdílu mezi celkovou schopností hráče vyhrávat při svém podání a celkovým průměrem všech hráčů při této činnosti;
- rozdílu mezi celkovým průměrem všech hráčů na příjmu a individuální schopností soupeře v této činnosti.

První sčítanec reprezentuje vlastnosti povrchu, druhý schopnosti podávajícího hráče, třetí schopnosti hráče na příjmu. Můžeme tedy vyslovit následující definici.

Definice 4.1: *V zápase dvou hráčů i a j stanovme pravděpodobnost zisku rozehry podávajícím hráčem i s_{ij} rovnou následujícímu výrazu:*

$$s_{ij} = s_{trn} + (s_i - s_{avg}) + (r_{avg} - r_j), \qquad (9)$$

kde s_{trn} je průměrná úspěšnost podávajících hráčů v rozehře na daném turnaji, s_i je celková úspěšnost podávajícího hráče ve všech zápasech, s_{avg} je celková průměrná úspěšnost všech podávajících hráčů ve všech zápasech, r_j je celková úspěšnost přijímajícího hráče ve všech zápasech, r_{avg} je celková průměrná úspěšnost všech podávajících hráčů ve všech zápasech.

Analogicky lze vyjádřit i úspěšnost na příjmu, což je shrnuto v definici 4.2.

Definice 4.2: *V zápase dvou hráčů i a j stanovme pravděpodobnost zisku rozehry přijímajícím hráčem j r_{ji} rovnou následujícímu výrazu:*

$$r_{ji} = r_{trn} + (r_j - r_{avg}) + (s_{avg} - s_i), \qquad (\mathbf{10})$$

kde r_{trn} je průměrná úspěšnost přijímajících hráčů v rozehře na daném turnaji, ostatní proměnné mají stejný význam jako v předchozí definici.

Korektnost uvedených definic můžeme ověřit následujícím tvrzením:

Tvrzení 4.1: *V zápase dvou hráčů i a j dle výše uvedených definic platí, že součet pravděpodobností s_{ji} a r_{ji} je roven jedné.*

Důkaz: Z konstrukce průměrných pravděpodobností s_{trn} a r_{trn} je zřejmé, že platí rovnost (11):

$$s_{trn} + r_{trn} = 1. \qquad (\mathbf{11})$$

Vyjádříme-li pravděpodobnosti s_{trn} a r_{trn} z rovnic (9) resp. (10), a dosadíme-li je do rovnice (11), dostaneme rovnici (12). Po jednoduché úpravě hned získáme dokazovanou rovnost.

$$s_{ij} + r_{ji} - [(s_i - s_{avg}) + (s_{avg} - s_i)] - [(r_{avg} - r_j) + (r_j - r_{avg})] = 1, \qquad (\mathbf{12})$$

□

Pravděpodobnosti s_{ji} a r_{ji} dle definic 4.1 a 4.2 můžeme nyní použít jako vstup do našeho modelu tenisového zápasu.

4.2 Aplikace modelu

V této části se pokusíme zjistit, do jaké míry odpovídá námi zkonstruovaný model tenisového zápasu realitě. Soubor programů (jejich podrobnější popis a uživatelskou příručku lze nalézt v kapitolách 6 a 7) nám umožnil získat empirická data, výsledky mužských tenisových dvouher. Počítačovou implementaci matematického modelu jsme pak zkonstruovali v tabulkovém procesu.

Abychom mohli porovnat model se získanými empirickými daty, potřebujeme znát dva vstupní parametry modelu: úspěšnost přijímajících a podávajících hráčů v rozehře. Tento požadavek však značně omezil rozsah dat, která je možno využít. Dostatečné množství potřebných údajů se podařilo získat pro turnaj U.S. Open, který je jako jediný z turnajů Grand Slam hrán stylem tie-break „best of five“.

U.S. Open hraje 128 hráčů, turnaj má tedy 7 kol a odehraje se v něm 127 zápasů, to odpovídá 381 zápasům během tří let. Z empirických dat jsme vyřadili 22 zápasů, během nichž jeden z hráčů vzdal, což bylo 21 případů, či nenastoupil k zápasu. Tabulka 4.2 uvádí zjištěné souhrnné statistiky o sledovaném souboru 359 zápasů.

	Odehráno							Zápasů
	sad	her	tie-breaků	bodů TB	H / S	TB / S	B / TB	ukončeno
1.sada	359	3474	60	714	9,68	0,17	11,90	-
2.sada	359	3548	69	756	9,88	0,19	10,96	-
3.sada	359	3406	40	477	9,49	0,11	11,93	161
4.sada	198	1926	34	424	9,73	0,17	12,47	122
5.sada	76	740	11	121	9,74	0,14	11,00	76
celkem	1351	13094	214	2492	9,69	0,16	11,64	359

Tabulka 4.2 - Souhrnná statistika U.S. Open 2002-2004, z empirických dat získaných pomocí programu Tennis Builder.

4.2.1 Jednoduché odhady

K určení parametrů modelu jsme využili údaje z tabulky 4.1, kde je průměrná pravděpodobnost zisku rozehry podávajícím hráčem na U.S. Open v letech 2002 – 2004 vypočtena na 62.8%.

Nejprve se pokusme odhadnou celkový počet her odehraných během sledovaného období. Protože očekávaný počet her v zápase je nejvyšší u vyrovnaných soupeřů, umožní nám to snadno získat horní odhad očekávané hodnoty. Dosazením do modelu nám vychází hodnota 15 062 her za uvedené období, což je mnohem více než pozorovaných 13 094.

Podobným způsobem lze shora odhadnout očekávaný počet tie-breaků. Při nezměněných vstupních parametrech dostáváme z modelu 311 tie-breaků při jen 214 pozorovaných.

Konečně náš model předpovídá 90 výsledků 3:0, 135 výsledků 3:1 a shodně 135 výsledků 3:2. Empiricky však získáváme 161 výsledků 3:0, 122 výsledků 3:1 a 76 zápasů skončilo 3:2. Důvod uvedeného přeceňování délky zápasu je však zřejmý. Nelze u všech 359 zápasů očekávat naprosto vyrovnané soupeře. Chceme-li dosáhnout přesnějších odhadů, nezbude nám nic jiného než zlepšit distribuci pravděpodobností, aby se blížila simulovanému souboru.

4.2.2 Simulace většího počtu zápasů

K získání parametrů využijeme poznatky z části Získání parametrů modelu. Ze statistických údajů o 200 nejlepších hráčích, které jsme převedli do tabulkového procesoru, jsme vyřadili hráče s neúplnými či nulovými údaji. Redukovaný soubor 189 hráčů jsme dále zpracovali, abychom získali jejich úspěšnosti v rozehře při podání a příjmu. Dále jsme dle definice 4.1 vytvořili matici pravděpodobností odpovídající všem možnostem vzájemných zápasů. Jednoduchá simulace napsaná v programovacím jazyku Visual Basic for Application (podrobněji o ní v subkapitole 6.3) nám umožnila simulovat násobek 359 zápasů náhodným výběrem z 17 766 možných, nebo přesněji 35 532, protože náš model rozlišuje, kdo ze soupeřů podává v zápase jako první. Abychom získali přesnější výsledky, bylo nasimulováno celkem 3 590 zápasů.

Výsledky simulace přepočtené pro 359 tenisových utkání jsou uvedeny v tabulkách 4.3 a 4.4. Srovnejme jednotlivé charakteristiky mezi výsledkem simulace a empirickými daty. Rozdíly mezi teorii a praxí jsou na první pohled patrné. Vidíme, že náš matematický model předpověděl celkem o 800 her, tj. o přibližně 6%, více než bylo ve skutečnosti pozorováno. Nejvýraznější rozdíly lze pozorovat ve čtvrtém a pátém setu.

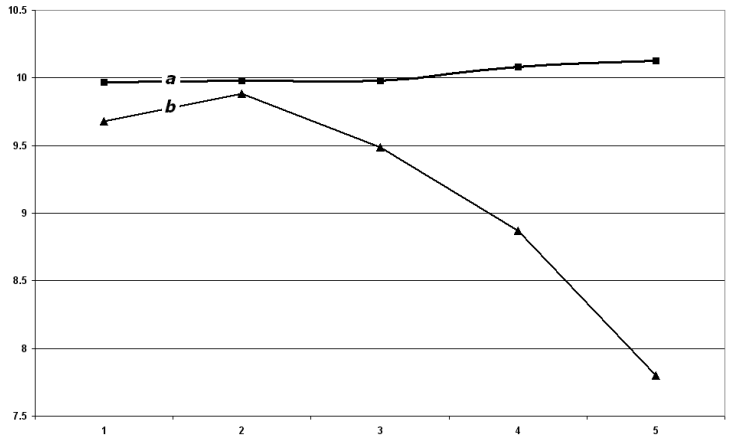
Sada	1	2	3	4	5	celkem
Očekávaný počet her	3578.8	3582.5	3582.4	2190.0	960.9	**13894.6**
Skutečný počet her	3474.0	3548.0	3406.0	1926.0	740.0	**13094.0**
Očekáváný počet tie-breaků	70.5	70.5	70.5	45.4	20.3	**277.3**
Skutečný počet tie-breaků	60.0	69.0	40.0	34.0	11.0	**214.0**

Tabulka 4.3 - Výsledky simulace a empiricky zjištěná data, očekávané hodnoty počtu her a počtu tie-breaků v 359 zápasech.

	Simulovaná data	Empirická data
3:0	141.84	161
3:1	122.26	122
3:2	94.90	76
celkem	**359**	**359**

Tabulka 4.4 - Výsledky simulace a empirická data, rozložení délky zápasů v setech.

Nesouvisí to však pouze s nižším počtem ve skutečnosti odehraných čtvrtých a pátých setů oproti teoretickému rozložení. Na obrázku 4.1, kde je graficky znázorněn průměrný počet odehraných her v jednotlivých sadách, můžeme pozorovat snižování délky jednotlivých sad s postupující délkou zápasu. To je v přímém rozporu s teoretickými předpoklady. Není to však zcela překvapující, protože lze předpokládat, že při takto dlouhých zápasech působí na skutečné tenisové hráče další faktory, jako je např. únava, které náš model nedokáže zachytit.



Obrázek 4.1 - Průměrný počet odehraných her v sadě: (a) simulovaná data, (b) empirická data.

Celkové zkrácení délky zápasu si je možno povšimnout také ve srovnání v tabulce 4.4, kdy model předpovídá 142 výsledků 3:0, 122 výsledků 3:1 a 95 výsledků 3:2. Ve skutečnosti pozorujeme o 19 výsledků 3:0 více a o 19 výsledků 3:2 méně. V sledovaných zápasech je také odehrán nižší než teoretický počet tie-breaků.

5 Jiné způsoby předpovědi průběhu a výsledku tenisových zápasů

5.1 Oficiální tenisové žebříčky

V mnoha sportech lze nalézt jistou formu oficiálního žebříčku, který udává pořadí hráčů či týmů. Fotbalové reprezentační týmy jsou seřazeny v žebříčku Fédération Internationale de Football Association (FIFA), ve volejbale světové pořadí udává žebříček FIVB World Ranking, tenisový žebříček mužů vydává Association of Tennis Professionals (ATP).

Všechny tyto žebříčky jsou založeny na postupné kumulaci bodů na základě výsledků odehraných zápasů. Přesné bodové ohodnocení každého výsledku je však ovlivněno různými dalšími kvantitativními a kvalitativními charakteristikami. Takto získaná pořadí nejsou používána pouze jako vzájemné dlouhodobé porovnání týmů a pro jednoduchou prezentaci výsledků sportovní veřejnosti. Umožňují navíc sportovním asociacím zaručit, spolu se systémy kvalifikací a nasazování, spravedlivější los v jimi pořádaných soutěžích.

Kategorie turnaje	Prize money	W	F	SF	QF	R16	R32	R64	R128	Vítězům kvalifikace
Grand Slam		200	140	90	50	30	15	7	1	+ 3
Masters Series		100	70	45	25	15	7	1(4)	(1)	+ 3 (+ 1)
Int. Series Gold	$1,000,000	60	42	27	15	5	3	(0)	---	+ 2 (+ 1)
Int. Series Gold	$800,000	50	35	22	12	5	3	(0)	---	+ 2 (+ 1)
Int. Series	$1,000,000	50	35	22	12	5	3	(0)	---	+ 2 (+ 1)
Int. Series	$800,000	45	31	20	11	4	2	(0)	---	+ 2 (+ 1)
Int. Series	$600,000	40	28	18	10	3(4)	(2)	---	---	+ 1
Int. Series	$400,000	35	24	15	8	3	0	---	---	+ 1
Masters Cup		+ 20 za výhry ve skupině, + 40 finalistům, + 50 za vítězství								

2007 Round Robin		W	F	SF	QF	R16	R32	RR W	E W	Vítězům kvalifikace
Int. Series Gold	$800,000	50	35	22	12	---	---	6	4	+ 2
Int. Series	$800,000	45	31	25	13	---	---	5	3	+ 1
Int. Series 48	$600,000	40	28	18	10	4	---	2	---	+ 1
Int. Series 32	$600,000	40	28	18	10	---	---	4	2	+ 1
Int. Series	$400,000	35	24	15	8	---	---	4	2	+ 1

Tabulka 5.1 - Rozdělení získaných bodů v ATP Race dle očekávané obtížnosti zápasu, zdroj [4].

Aby ATP zajistila všechny jmenované principy předkládá veřejnosti dva různé žebříčky. ATP Race je formou klasických systémů hodnocení ve sportu, které jsou založeny na kalendářním základě. Každý hráč vstupuje do nové sezóny, jejíž začátek se zde shoduje s počátkem kalendářního roku, s nulovým počtem bodů. Každému hráči se započítává 18 nejlepších výsledků dosažených během roku, případě navíc ještě výsledek dosažený na Tennis Masters Cup, po jehož skončení je vyhlášen celkový vítěz.

Množství bodů získaných za odehraný zápas se liší dle předpokládané obtížnosti utkání, která je odhadována dle významnosti turnaje a roste během jednotlivých kol turnaje. Například vítěz Grand Slamu získává 200 bodů, což odpovídá vítězství na čtyřech menších turnajích. Podrobnou distribuci bodů lze nahlédnout v tabulce 5.1.

Kategorie turnaje	Prize money	W	F	SF	QF	R16	R32	R64	R128	Vítězům kvalifikace
Masters Series		1000	700	450	250	150	75	35	5	+ 15
Int. Series Gold		500	350	225	125	75	35	5(20)	(5)	+ 15 (5)
Int. Series Gold	$1,000,000	300	210	135	75	25	15	5	---	+ 10 (5)
Int. Series	$800,000	250	175	110	60	25	15	0	---	10 (5)
Int. Series	$1,000,000	250	175	110	60	25	15	0	---	10 (5)
Int. Series	$800,000	225	155	100	55	20	10	0	---	+ 10 (5)
Int. Series	$600,000	200	140	90	50	15	0	---	---	+ 5
Int. Series	$400,000	175	120	75	40	15	0	---	---	+ 5
Challenger	$150,000+H	100	70	45	23	10	0	---	---	+ 3
Challenger	$150,000	90	63	40	21	9	0	---	---	+ 3
Challenger	$125,000	80	56	36	19	8	0	---	---	+ 3
Challenger	$100,000	70	49	31	16	7	0	---	---	+ 3
Challenger	$75,000	60	42	27	14	6	0	---	---	+ 3
Challenger	$37,500+H	55	38	24	13	5	0	---	---	+ 2
Challenger	$50,000	50	35	22	12	5	0	---	---	+ 2
Futures	$15,000+H	24	16	8	4	1	0	---	---	0
Futures	$15,000	18	12	6	3	1	0	---	---	0
Futures	$10,000	12	8	4	2	1	0	---	---	0
Masters Cup		+100 za výhry ve skupině, +200 finalistům, +250 za vítězství								

2007 Round Robin		W	F	SF	QF	R16	R32	RR W	E W	Vítězům kvalifikace
Int. Series Gold	$800,000	250	175	110	60	---	---	30	20	+ 10
Int. Series	$800,000	225	155	100	55	---	---	25	15	+ 5
Int. Series 48	$600,000	200	140	90	50	20	---	10	---	+ 5
Int. Series 32	$600,000	200	140	90	50	---	---	20	10	+ 5
Int. Series	$400,000	175	120	75	40	---	---	20	10	+ 5

Tabulka 5.2 - Rozdělení získaných bodů v ATP Ranking dle očekávané obtížnosti zápasu, zdroj [4].

Hráč	ATP Ranking k 6.8.2007				ATP Race k 6.8.2007			
	Pořadí	Body	% max	Započteno turnajů	Pořadí	Body	% max	Započteno turnajů
Federer, Roger (SUI)	1	7290	**100.0**	17	2	801	**85.6**	9
Nadal, Rafael (ESP)	2	5455	**74.8**	20	1	936	**100.0**	13
Roddick, Andy (USA)	3	3290	**45.1**	21	4	376	**40.2**	14
Djokovic, Novak (SRB)	4	3200	**43.9**	21	3	561	**59.9**	14
Davydenko, Nikolay (RUS)	5	3075	**42.2**	26	5	325	**34.7**	16
Gonzalez, Fernando (CHI)	6	2695	**37.0**	17	6	293	**31.3**	10
Robredo, Tommy (ESP)	7	2200	**30.2**	24	8	275	**29.4**	15
Gasquet, Richard (FRA)	8	2085	**28.6**	22	10	257	**27.5**	14
Blake, James (USA)	9	1995	**27.4**	25	22	185	**19.8**	16
Berdych, Tomas (CZE)	10	1975	**27.1**	20	14	244	**26.1**	13

Tabulka 5.3 - Porovnání žebříčků ATP Race a ATP Ranking.

ATP Ranking určuje, kteří z přihlášených hráčů se kvalifikují do hlavní soutěže každého turnaje nebo budou připuštěni do kvalifikace, a jejich případné nasazení v losu. Žebříček zahrnuje výsledky za uplynulých 52 týdnů, což zajišťuje nezávislost na kalendáři. Způsob přidělování bodů hráčům je v ATP Ranking (viz tabulka 5.2) obdobné jako u ATP Race, započítávají se však výsledky dosažené na všech turnajích zařazených do okruhu.

Rozdíl mezi oběma žebříčky je patrný v tabulce 5.3, která zachycuje odlišnosti v pořadí 10 nejlepších hráčů. Lze si také všimnout, že se oba způsoby výpočtu liší relativní vzdáleností mezi hráči. Je proto zřejmé, že z hlediska modelování a předpovídání výsledků zápasů je vhodnější použít ATP Ranking, který zejména počátkem roku přesněji odráží celkovou výkonnost hráčů.

Existuje však způsob jak sblížit sémantiku ATP Race a ATP Ranking. Rovnice (13) definuje modifikovaný žebříček tak, že aktuální bodovou hodnotu ATP Race navýší o body, které hráč získal v loňském roce ve zbytku sezóny.

$$ATPRace_{mod} = ATPRace(now) + ATPRace(year\_end) - ATPRace(now\text{-}1Y) \quad (\mathbf{13})$$

Ze způsobu konstrukce ATP Race však plynou jisté limity úspěšnosti této modifikace. Protože je celkový počet turnajů započtených do žebříčku omezen, je nutné předpokládat, že hráči v obou letech odehráli do odpovídajícího kalendářního období stejný či podobný počet turnajů. V opačném případě lze očekávat odchylky od korektního výsledku. Část modifikovaného žebříčku k 6.8.2007 a jeho srovnání s ATP Ranking je uvedeno v tabulce 5.4. Jediným významnějším rozdílem je prohození pořadí Jamese Blakea a Tomáše Berdycha, které je způsobeno nižším počtem v tomto roce započtených turnajů u druhého z hráčů.

Hráč	ATP Ranking		ATP Race k			upravená ATP Race			
	pořadí	% max	6.8.07	7.8.06	20.11.06	rozdíl	součet	pořadí	% max
Federer, Roger	**1**	**100.0**	801	1017	1674	657	1458	**1**	**100.0**
Nadal, Rafael	**2**	**74.8**	936	735	894	159	1095	**2**	**75.1**
Roddick, Andy	**3**	**45.1**	376	194	483	289	665	**3**	**45.6**
Djokovic, Novak	**4**	**43.9**	561	192	276	84	645	**4**	**44.2**
Davydenko, Nikolay	**5**	**42.2**	325	308	565	257	582	**5**	**39.9**
Gonzalez, Fernando	**6**	**37.0**	293	175	403	228	521	**6**	**35.7**
Robredo, Tommy	**7**	**30.2**	275	292	475	183	458	**7**	**31.4**
Gasquet, Richard	**8**	**28.6**	257	106	273	167	424	**8**	**29.1**
Blake, James	**9**	**27.4**	185	301	506	205	390	**10**	**26.7**
Berdych, Tomas	**10**	**27.1**	244	188	341	153	397	**9**	**27.2**

Tabulka 5.4 - Ukázka výsledku modifikace hodnot žebříčku ATP Race.

5.2 Alternativní žebříčky

U ostatních sportů se můžeme setkat s podobnými ad hoc vytvořenými žebříčky. Asi nejdůležitějším z nich je Elo žebříček [18], který byl vyvinut jako nástroj pro porovnávání výkonnosti šachistů. Ten mimo jiné umožňuje teoretické porovnání výkonnosti současných hráčů s hráči již nehrajícími. Jeho konstrukce vychází

z předpokladu, že výkonnost každého hráče v každé hře je normálně rozdělená náhodná proměnná.

Elo systém je založen na exponenciálně vyrovnávaných hodnoceních jednotlivých hráčů, která jsou získána porovnáním skutečného poměru jejich vítězství k očekávanému, tak jak by to odpovídalo dosavadní výkonnosti jejich soupeřů. Navíc žebříček udává přímý vztah mezi rozdílem v hodnocení hráčů v žebříčku a pravděpodobností jejich porážky či vítězství. Rozdíl 200 bodů odpovídá přibližně 0,75 pravděpodobnosti výhry a 0,25 pravděpodobnosti prohry, rozdílu 100 bodů mezi hráči odpovídají pravděpodobnosti 0,64 resp. 0,36.

O výhodnosti použití Elo systému svědčí jeho využití i v ostatních sportech. Strauss a Arnold [19] doporučili použití podobného hodnotícího systému pro racquetball, Clarke [20] upravil systém pro squash. Jedním z problémů adaptace Elo systémů pro hodnocení výkonnosti hráčů tenisu je fakt, že se tenisové turnaje hrají na různém povrchu (antuka, tráva, umělá tráva, beton, různé syntetické povrchy, …), a navíc se mohou konat venku i v hale. Většina hráčů má svůj oblíbený povrch a jejich výkonnost se mezi různými povrchy významně liší.

Jedním z podobných případů statisticky významné nerovnosti podmínek je výhoda domácího prostředí, která byla popsána u např. u australského fotbalu [21]. Přesto se podařilo vytvořit několik systémů pro předpověď výsledků fotbalových utkání, jejichž úspěšnost byla srovnatelná s úspěšností tipů expertů [22,23].

5.3 Předpovědi dle ATP ranking

Clarke a Dyte [3] testovali, zda lze předpovídat výsledky na základě žebříčku ATP Ranking. Jejich metoda byla založena na předpokladu, že výše nasazení hráči vyhrávají zápasy nad níže nasazenými. Např. v roce 1997 hráči s vyšším postavením v žebříčku vyhráli na Australian Open 69,7%, na Roland-Garros 60,9% zápasů, na Wimbledonu 64,1% zápasů a na U.S. Open 62,5% zápasů.

Sestrojili proto model, který postupně testovali na datech Wimbledonu 1998, U.S. Open 1998 a Australian Open 1999. Nejzdařilejší byla předpověď výsledků Wimbledonu, kde se správně podařilo určit pozdějšího vítěze Peta Samprase, když byl modelem před turnajem odhadován na vítězství s pravědpodobností 25%. Nicméně poraženému finalistovi Goranu Ivaniševiči, byly až do pozdních kol dávány zcela teoretické šance (<1%). Ve finále bylo vítězství Peta Samprase kvantifikováno na 90%, což však příliš neodpovídalo vyrovnanému průběhu zápasu 6-7(2-7) 7-6(11-9) 6-4 3-6 6-2. Na ostatních turnajích byly výsledky předpovědí méně přesné.

Scheibehenne a Bröder [24] ve studii předpovědí výsledků Wimbledonu 2005 srovnávali úspěšnost mezi předpověďmi dle žebříčku ATP Ranking, tipy tenisových amatérů, tipy laiků a tipů s využitím kurzů sázkových kanceláří. Zaznamenali, že první tři skupiny dosahovali podobně úspěšnosti 70%. Není asi velkým překvapením, že se nejlépe podařilo výsledky určit dle kurzů on-line sázkovým kanceláří, kde byla dosažena úspěšnost 76%.

6 Empirická data a práce s nimi

Aby bylo možné náš teoretický model nějak využít, je za potřebí nastavit jeho parametry. K tomu je mimo jiné potřeba získávat aktuální data o tenisových zápasech. Protože není v lidských silách, alespoň v těch rozumně vynaložených, sledovat výsledky všech tenisových zápasů, je potřeba tento proces nějakým způsobem zautomatizovat.

Na Internetu se dá najít velké množství různých stránek, které v nějaké formě shromažďují údaje o tenisových kláních, většina z nich má ale data neúplná, pozdě či nepravidelně publikovaná, nepřesná či jinak závadná. Naštěstí existuje jedna spolehlivá a dlouhodobě prověřená stránka: www.stevegtennis.com, kde najdeme rukama mnoha nadšenců udržované výsledky tenisových dvouher mužů od roku 1978 až do současnosti.

Data jsou publikována v lidskou rukou psaných strukturovaných textových souborech. Pro naše potřeby je nutné převést je do formátu vhodného k dalšímu zpracování. Při tvorbě našeho modelu se ukázalo výhodné použít tabulkový kalkulátor, jedním z požadavků proto bylo, aby data bylo možno do něj snadno importovat. Tato kritéria splňoval textový formát „CSV“ („comma-separated values“). Transformaci do něj zajišťuje aplikace TennisBase Builder. Pro kontrolu a úpravu vygenerovaných dat byl později vytvořen TennisBase Editor.

Tímto způsobem bylo postupně zpracováno přes 350 000 výsledků tenisových zápasů. O přístup do takto rozsáhlé databáze projevilo během několika let jejího provozování zájem několik sázkařů a tenisových fanoušků. Dle jejich požadavků byla vytvořena (záměrně) jednoduchá aplikace, TennisBase Viewer, určená, jak název napovídá, zejména k prohlížení výsledků. Každý rok v databázi přibývají údaje o desítkách tisíc zápasů. Nutností je proto zajistit možnost aktualizace dat. Tvorba aktualizačních balíčků zajišťuje TennisBase Convertor, který navíc provádí konverzi z formátu CSV do interního formátu TennisBase Viewer. Tomu byla, možná trochu nešťastně, dána přednost před nějakým standardizovaným formátem.

6.1 Zpracování vstupních dat

6.1.1 Analýza struktury dat

Abychom mohli sestrojit program zpracovávající vstupní data, je potřeba mít dobrou představu o jejich struktuře. Výhodou pro nás samozřejmě je, pokud jsme navíc seznámeni s jejich sémantikou. Připomeňme proto krátce, kde se příslušné informace nacházejí. Stručný popis tenisových pravidel spolu se základními informacemi o tenisových turnajích lze nalézt v kapitole 2. Systém tenisových žebříčků byl podrobněji rozebrán v části 5.1.

V první řadě je nutno prozkoumat použitou konvenci pojmenování vstupních souborů, protože jejich jména zde nesou důležité informace. To souvisí s tím, že většina tenisových turnajů je hrána kaskádovým způsobem. Tedy, že níže nasazeným hráčům je umožněno bojovat v kvalifikaci o několik míst v hlavní soutěži. Dokonce, u některých turnajů je hrána tzv. před-kvalifikace, jejímž vítězům je dána možnost účastnit se kvalifikace. Jména souborů jsou tvořena dvojicí: „PŘEDPONA-HLAVNÍČÁST“, kde „HLAVNÍČÁST“ je unikátní řetězec společný pro všechny turnaje jedné kaskády, a kde „PŘEDPONA“ označuje druh kaskády a pozici turnaje v kaskádě. Výsledek analýzy kaskádové struktury pojmenování, který je uveden v tabulce 6.1, nám umožňuje sdílet informace mezi jednotlivými částmi kaskády a doplnit tak případné chybějící či nepřesné údaje.

Předpona	Význam / Skupina dat	Nadřazená předpona
a	ATP Tour	neexistuje
q-a	kvalifikace na ATP Tour	a
pq-a	předkvalifikace na ATP Tour	q-a
ch	Challenger	neexistuje
q-ch	kvalifikace na Challenger	ch
pq-ch	předkvalifikace na Challenger	q-ch
f	Futures	neexistuje
q-f	kvalifikace na Futures	f
pq-f	předkvalifikace na Futures	q-f
sa	Satelitte Cirtuit	neexistuje

Tabulka 6.1 - Výsledek analýzy stromové struktury pojmenování vstupních dat.

Nyní můžeme přistoupit k analýze jednotlivých vstupních souborů. Ty jsou rozděleny do dvou logických celků: hlavičky turnaje a výsledků zápasů. Na obrázku 6.1 je uveden příklad hlavičky turnaje kategorie ATP Tour. Na prvním řádku je uveden název turnaje, na druhém místo konání. Třetí řádek obsahuje informace o datu konání turnaje. Ve čtvrtém jsou informace o počtu zúčastněných hráčů a o celkové dotaci turnaje. Poslední řádek prvního bloku obsahuje informace o hracím povrchu. Druhý blok nese pro nás redundantní informace o nasazených hráčích. Všimněme si vpravo omezených informací v hlavičce kvalifikace. Zde se ukazuje oprávněnost implementace propojení mezi jednotlivými turnaji kaskády.

```
China Open                                Beijing qualifying
Beijing, China                            September 13-19, 2004
September 13-19, 2004
32 Draw - $500,000                        Singles
Surface - Hard                            Seeds: (cut: NR)
                                          1. Peter Luczak
Singles                                   2. Arvind Parmar
Seeds: (cut: Jean-Rene Lisnard - 148)     3. Gilles Simon
1. Carlos Moya                            4. Jamie Delgado
2. Juan Carlos Ferrero                    5. Jo-Wilfried Tsonga
3. David Nalbandian                       6. Tasuku Iwami
```

Obrázek 6.1 - Část hlavičky turnaje kategorie ATP Tour; vlevo ze souboru hlavní soutěže, vpravo ze souboru kvalifikace.

Hlavičky turnajů dalších kategorií (viz obrázky 6.2 a 6.3) jsou obdobné. Jejich struktura je mírně pozměněna, obsahují však stejné informace.

```
Prague Open 2004
Prague, Czech Republic
May 17-23, 2004
Clay - $125,000+H

Singles
Seeds: (cut: Timo Nieminen - 276)
1. Fabrice Santoro
2. Karol Kucera
3. Jan Vacek
4. Michael Llodra
5. Alex Bogomolov
6. Todd Reid
7. Bohdan Ulihrach
8. Jan Hernych
```

```
Prague challenger qualifying
May 17-23, 2004

Singles
Seeds: (cut: Jaroslav Trojan - NR)
1. Joseph Sirianni
2. Robin Vik
3. Lukas Dlouhy
4. Petr Luxa
5. Emilio Benfele-Alvarez
6. Jan Masik
7. Martin Slanar
8. Ota Fukarek
```

Obrázek 6.2 - Část hlavičky turnaje kategorie Challenger; vlevo ze souboru hlavní soutěže, vpravo ze souboru kvalifikace.

```
Bahrain F1 - Power Horse ITF Men's Cup
Manama, Bahrain
January 26-February 1, 2004
Surface: Hard
Main Draw      : 32
Qualifying Draw: 32
$15,000

Singles
Seeds: (cut: Michihisa Onoda - 527)
1. Uros Vico
2. Marco Chiudinelli
3. Victor Bruthans
```

```
Bahrain F1 qualifying
January 26-February 1, 2004

Singles
Seeds: (cut: none)
1. Sunil Kumar Sipaeya
2. Jasper Smit
3. Mohamed Maamoun
4. Alexander Hartman
5. Karim Maamoun
6. Mustafa Ghouse
7. Rameez Junaid
8. Baptiste Dupuy
```

Obrázek 6.3 - Část hlavičky turnaje kategorie Futures; vlevo ze souboru hlavní soutěže, vpravo ze souboru kvalifikace.

Druhým logickým celkem jsou výsledky jednotlivých tenisových zápasů. Následují bezprostředně po hlavičce a jsou sdruženy do bloků po jednotlivých kolech turnaje. Každý blok je uveden řádkem obsahujícím řetězec s názvem kola např.: „First Round“, „Second Round“, „Finals“, ... Zbylé řádky bloku tvoří zápis výsledků zápasů.

Na obrázku 6.4 je uveden příklad jednoho takového bloku. Informace o zápase jsou zaznamenány v řetězci na jednom řádku. Ten lze rozdělit do pěti částí, zleva doprava to jsou:

- oblast s informacemi o vítězi resp. hráči, který v přerušeném zápase vede;
- oddělovač „d.“ resp. „def.“ s významem porazil, oddělovač „leads“ s významem vede nad, či oddělovač „vs.“, který je umístěn mezi hráči, jejichž utkání ještě nezačalo;
- oblast s informacemi poraženém resp. druhém hráči;
- numerické vyjádření výsledku zápasu;
- v hranatých závorkách uvedený komentář.

Oblast s informacemi o hráčích obsahuje, kromě jejich jména a příjmení, i další důležité údaje. Zkratka, uvedená v závorkách za jménem, informuje o hráčově národnosti. Další skupinu závorek lze najít před jmény hráčů. Může v nich být uvedeno číslo, které má význam nasazení během losu turnaje, nebo písmena, která specifikují, jakým způsobem se hráč dostal do turnaje.

Zkratka „WC“ znamená, že mu byla organizátory udělena divoká karta. Hráči uvedení písmenem „q“ postoupili do soutěže jako vítězové kvalifikace. „LL“ je v kvalifikaci poražený hráč, který však postoupil jako náhradník zraněného hráče. Písmeny „ALT“ jsou označeni ostatní hráči, kteří nastoupili do soutěže jako náhradníci. Konečně, „SE“ můžeme najít u hráče, který hrál v době kvalifikace finále či semifinále jiného turnaje.

Obrázek 6.4: Blok zápasů druhého kola.

6.1.2 Výstupní formát

Pro naše účely se ukázal vhodný formát CSV, zejména pro jeho snadné importování do tabulkového kalkulátoru. Je to typ textového souboru, navržený k uchovávání tabulkových dat. Implementována byla varianta tohoto formátu, kterou lze snadno importovat do české verze programu Microsoft Excel. Jeden soubor tvoří jedna tabulka, jejíž jednotlivé položky na jednom řádku jsou odděleny „;”.

Jednotlivé položky jsou:

- datum konání turnaje;
- údaje o nasazení vítěze;
- jméno a příjmení vítěze;
- údaje o nasazení poraženého;
- jméno a příjmení poraženého;
- počet získaných setů vítěze;
- počet získaných setů poraženého;
- přesný výsledek;
- zlomkem vyjádřené kolo turnaje (tedy „1/1“ znamená finále, „1/2“ semifinále, „1/4“ čtvrtfinále, atd.);
- slovem vyjádřené kolo turnaje;
- název turnaje;
- místo konání turnaje;
- celková finanční dotace turnaje;
- hrací povrh;
- kategorie kaskády turnajů;
- pozice turnaje v kaskádě.

Přestože se ukázalo toto uložení dat výhodné k účelu zpracování v dalších programech, je zřejmé, že přílišná redundance dat činí tento způsob jejich zápisu, neoptimální. Aplikace TennisBase Viewer proto již využívá databázi s více tabulkami (viz odstavec 6.2.3).

6.1.3 Transformační automat

K transformaci textových dat do tabulky byl navržen automat, který načítá z konfiguračního souboru instrukce (popis jejich syntaxe viz dodatek B) a vykonává je na vstupních datech. Ty zpravidla načítá do mezipaměti po řádcích či blocích jako řetězce. S řetězci uloženými v mezipaměti lze provádět různé textové operace, jako je dělení, spojování, vyhledávání, můžou být z mezipaměti uloženy do databáze nebo smazány.

Abychom čtenáři přiblížili způsob fungování transformačního automatu uveďme několik podrobností.

Stav automatu je definován:

- ukazatelem do souboru instrukcí;
- ukazatelem do zpracovávaného souboru;
- řetězci uloženými v mezipaměti;
- pomocnými proměnnými jako je separátor bloku;
- řetězci připravenými pro zápis do databáze.

Chod automatu je řízen příkazy, uveďme si proto jejich krátký přehled, v němž jsou rozděleny dle své logické funkce.

Implementovány jsou následující skupiny řídících příkazů:

- příkaz skoku na návěstí („GOTO“);
- příkaz skoku o daný počet instrukcí dopředu („JUMP“);
- podmínkové příkazy („IFBUFFREE“, „IFNPOS“, „IFBUFLENLWR“, „IFEOF“, ...);
- příkaz ukončení zpracování („HALT“).

Zpracování textu je řízeno následující skupinou příkazů:

- příkazy vyhledávání ve vstupním souboru („FIND“, „FIND+“);
- příkazy skoku ve vstupním souboru („SKIPLINE“, „AGAIN“);
- technickým příkazem nastavení oddělovače mezi bloky („BLOCK_SEP“);
- příkazem čtení řetězců ze vstupního souboru („READBLOCK“, „READLINE“);
- příkazem přidání řetězce do mezipaměti („ADD“);
- příkazy pro vymazání řetězců z obsahu mezipaměti („CLEARBUFFER“, „FLUSH“,);
- příkazy pro uložení rozpoznaných charakteristik o turnaji („SAVE _SURFACE“, „SAVE_TTYPE“, „SAVE_TTYPE2“, „SAVE_NAME“, „ROUNDNAME“, „SAVE_DATE“, „SAVE_DOTATION“, „SAVE _LOCATION“);
- příkazy pro upravování řetězců v mezipaměti („PARSE“, „PARSE+“, „PARSE!“, „PARSE+!“);
- příkaz pro získání informací o zápasech z řetězců v mezipaměti („PARSE GAMES“, „NEXTROUND“, „REPEATROUND“, „PARSEGAME“);
- příkaz pro uložení všech získaných údajů o turnaji do databáze („CLEAR TOURNAMENT“).

6.1.4 Oddělení jména od příjmení

Jak je zřejmé z obrázku 6.4, vstupní data obsahují jméno a příjmení hráče spojené v jednom řetězci bez jakéhokoliv oddělovače. Navrhli jsme proto jednoduchou gramatiku, na jejímž základě je oddělení obou položek provedeno.

Příjmení hráče je definováno jako spojení tří složek: předpony, těla a přípony. Tělo je specifikováno jako řetězec znaků bez mezer, což využívá faktu, že ve vstupních datech jsou víceslovná příjmení propojena pomlčkami (např. „Navarro-Pastor“). Předpony a přípony jsou od těla odděleny mezerou a jejich seznam je přesně vymezen. V určitých případech jsou gramatikou povoleny i jejich kombinace.

Celý postup oddělování pak probíhá následovně. V prvním kroku se vstupní řetězec otestuje s databází již rozpoznaných dvojic. To umožňuje případná zcela mimo gramatiku vybočující jména jednorázově doplnit do databáze. V druhém kroku dojde k otestování posledního slova se seznamem známých přípon. V případě kladného výsledku je poslední slovo uloženo jako přípona a předposlední jako tělo, jinak je poslední slovo uloženo jako tělo. Odpovídající slova jsou pak z řetězce odstraněna. Dále je aktuálně poslední slovo otestováno jako předpona a případně uloženo. Nakonec je zbylá část řetězce uložena jako jméno. To nám umožní se vypořádat se jmény jako: „Sonny van der Velden“, „Jose Enrique de la Torre“, „Jose Carlos Pinto Jr.“, apod.

Se jménem a příjmením je spojena ještě jedna nepříjemnost. Může se samozřejmě stát, že tenis hrají dva hráči stejného jména. Ve vstupních datech je u těch hráčů, jejichž pojmenování není jednoznačné, uveden za jménem v závorce rok narození. Převodní algoritmus uvedenou skutečnost zkontroluje, a pokud to je potřeba, pozmění příslušným způsobem údaje v databázi.

6.1.5 Seřazení dle předpokládaného data odehrání

Je přirozeným požadavkem mít zápasy seřazeny dle data odehrání. Vstupní data obsahují pouze interval ve kterém byly zápasy odehrány. Jistě lze rozumně předpokládat, že během kaskádových turnajů jsou nejdříve odehrány kvalifikace a před-kvalifikace, a až po nich následuje hlavní soutěž. Aby toto řazení bylo zachováno, přiřadíme ke každému možnému datu konání zápasu odpovídající index. Vzniklo nám tedy „3 * počet dnů našeho kalendáře“ kategorií. Tato množina je pro reálné aplikace konečná a platí, že každý zápas je jednoznačně zařazen do jedné kategorie. To nám umožňuje použít pro třídění variantu rychlého algoritmu Bucket Sort [25].

6.2 Poznámky k implementaci

6.2.1 Jazyk a vývojové prostředí

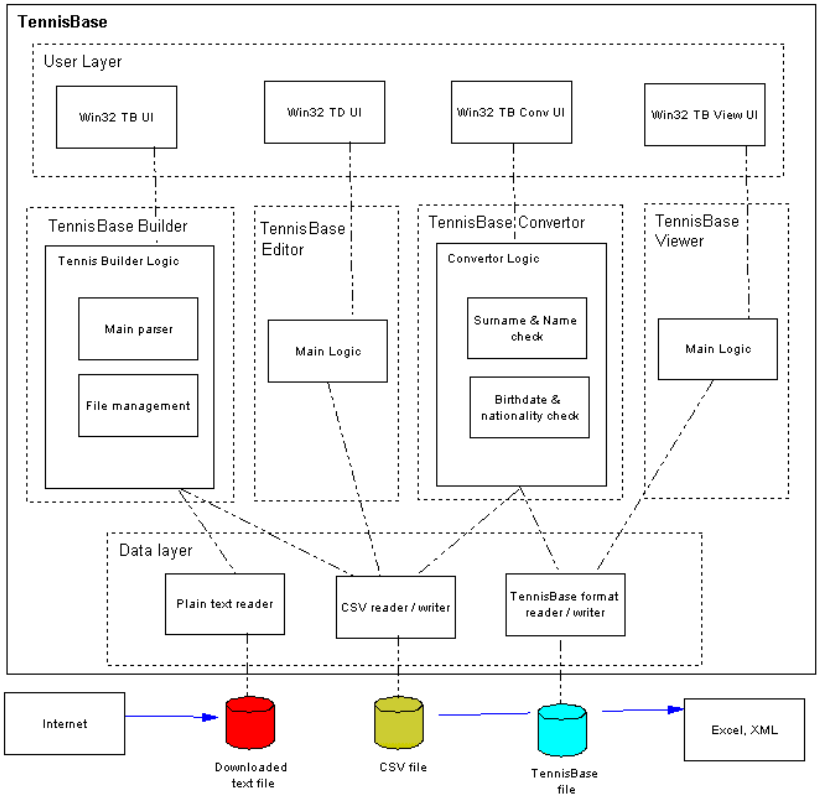
Aplikace byly vytvářeny ve vývojovém prostředí firmy Borland. První iterace TennisBase Builderu byla vytvořena v prostředí Borland C++ 3.1. Další vývoj však pokračoval jen pro platformu Win32. TennisBase Builder a TennisBase Editor byly

napsány v jazyce C++ za pomoci vývojového prostředí Borland C++ Builder 3.0. TennisBase Convertor a TennisBase Viewer jsou vytvořeny v jazyce Object Pascal ve vývojovém prostředí Borland Delphi 4.0.

V prostředí těchto vývojových nástrojů je vývoj všech aplikací, zejména v částech určených k interakci s uživateli, usnadněn předem připravenými vzory. Ty stačí „jen“ vhodně modifikovat a doplnit o vhodné komponenty a vlastní funkčnost. Kromě toho, že se nemusíme starat o volaní služeb Windows API, můžeme rozšiřovat vývojové možnosti o komponenty třetích stran.

6.2.2 Funkce a propojení jednotlivých částí

Přestože aplikace na sebe logicky navazují, nejsou typicky používány jedním uživatel všechny jako jediný celek. Uživatel Sázkař bude používat jen program TennisBase Viewer, pro vytváření modelů tenisového zápasu nám zcela vystačí dvojice TennisBase Builder a TennisBase Editor. Proto je pro snazší utvoření nadhledu na obrázku 6.5 k dispozici zjednodušené schéma jejich celkové funkce.



Obrázek 6.5 - Celkový pohled na aplikace TennisBase.

TennisBase Builder je implementací v odstavci 6.1.3 popsaného transformačního automatu. Navíc zajišťuje postupné načítání jednotlivých turnajů kaskády ve správném pořadí.

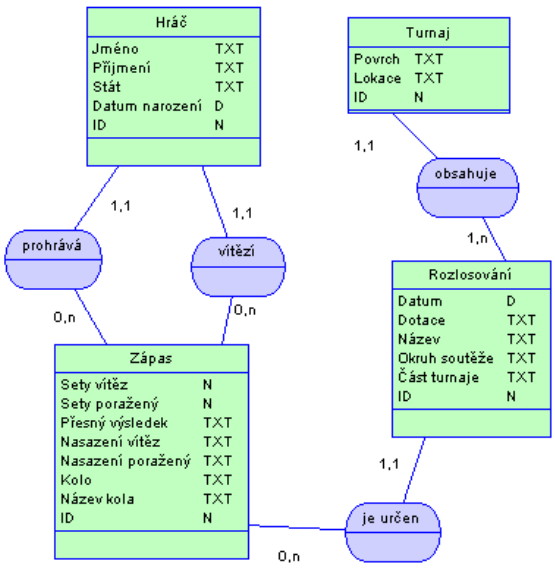
TennisBase Editor je jednoduchý tabulkový editor tenisové databáze. Kromě základní funkcí jako je přidávání a odebírání záznamů, editace položek záznamu, a pod. jsou v něm implementovány jednoduché filtrovací dotazy. Dále umožňuje seřadit opravené záznamy (jak je popsáno v sekci 6.1.5).

TenisBase Convertor slouží ke správě aktualizací pro TennisBase Viewer. Dokáže transformovat databázi z formátu CSV do formátu TennisBase a vytvořit rozdílový aktualizační balíček.

TennisBase Viewer je určen k jednoduchému prohlížení TennisBase databáze případnými koncovými uživateli.

6.2.3 Datová vrstva TennisBase Viewer

Výstupem TennisBase Builderu je databáze uložená v jedné tabulce. I když to přináší některé nesporné výhody, z hlediska databázové teorie a praxe není takto rozsáhlý redundantní soubor dat uložený v jedné tabulce ničím chvályhodným. Proto byla i naše databáze přetransformována do lepší struktury, což umožnilo zajistit snazší správu databáze a usnadnilo provádění aktualizací.



Obrázek 6.6 - E-R diagram databázové struktury používané programem TennisBase Viewer.

Strukturu databáze lze nahlédnout na E-R diagramu (viz obrázek 6.6). Schéma obsahuje čtyři tabulky a čtyři relace. Při konstrukci databáze dle konceptuálního modelu bývají relace implementovány různým způsobem v závislosti jejich násobnosti ve vztahu k ostatním entitám.

V našem případě lze každou relaci přiřadit do některé z tabulek jako další atribut. Relace *je určen*, stejně jako relace *prohrává* a *vítězí* jsou přidány do tabulky *Zápas*. Relace *obsahuje* je transformována jako další atribut tabulky *Rozlosování*.

Hráč je tabulka s pěti atributy. Tři textové atributy obsahující údaje o hráčově jménu, příjmení a národnosti, další atribut nese informaci o datu narození. Atribut *ID* je umělým klíčem.

Tabulka *Zápas* obsahuje záznamy s informacemi o zápasech. Kromě relačních atributů odkazujících na vítěze a poraženého a umělého klíče, obsahuje pět textových a dva numerické atributy.

Turnaj je tabulka sdružuje údaje o jedné kaskádě turnajů. Textové atributy *Povrch* a *Lokace* jsou oba klíčové. Pro rychlejší chod databáze a snadnější odkazování relací *obsahuje* je přidán navíc umělý klíč.

Tabulka *Rozlosování* zaznamenává údaje o jednom konkrétním turnaji kaskády. Kromě data konání turnaje uchovává ještě čtyři textové atributy. Relační atribut odkazuje na *Turnaj.ID*. Navíc je i zde je doplněn umělý klíčový atribut *ID*.

Fyzicky je tato databáze uložena na disku ve čtyřech souborech „td_001.dat“ (tabulka *Turnaj*), „td_002.dat“ (tabulka *Rozlosování*), „td_003.dat“ (tabulka *Hráč*) a „td_004.dat“ (tabulka *Zápas*). V každém z nich jsou jednotlivé záznamy uloženy sekvenčně a jsou seřazeny dle umělých klíčových atributů. Jednotlivé záznamy jsou navíc změněny jednoduchou šifrou. Největší síla ochrany je dána tím, že útočník nezná implementovaný způsob šifrování. Celá tato ochrana je koncipována zejména jako ochrana před náhodným pokusem o napadení.

Pro vlastní běh programu TennisBase Viewer jsou ještě používány další pomocné soubory: „td_000.dat“ s číslem licence a jako indikátor, zda byla databáze již použita; „td_005.dat“ s informacemi o povolených licencích; „td_006.dat“ s posledním nastavením programu.

6.3 Simulace ve VBA

K počítačovému sestrojení matematického modelu byla využita česká verze aplikace Microsoft Office Excel 2003. Tento produkt umožňuje snadné, byť ne bezpracné, sestrojení modelu Markovových řetězců. Vhodným postupem se dá jen s použitím statických vzorců zkonstruovat model celého zápasu s dvěma parametry.

Zpracování většího množství dat, ať už jako podklad pro sestrojení tabulek či grafů, nebo jako výsledek simulace většího počtu zápasů, už nezbytně předpokládá použití programovacího jazyka. Nejjednodušší je volbou je využít v Excelu integrovaného Visual Basic for Application (VBA). Pomocí něj bylo napsáno několik jednoduchých procedur, které umožnili uvedené operace realizovat.

7 Uživatelská dokumentace

Balík aplikací TennisBase je tvořen čtyřmi samostatnými aplikacemi. Není předpokládáno, že jeden uživatel bude používat všechny z nich. Můžeme rozlišovat mezi třemi způsoby jejich využití.

Pro získání dat v jednoduché databázové formě, která je dostačující pro využití s matematickým modelem, slouží TennisBase Builder. Jeho výstup, CSV soubor, je možno snadno zpracovat v tabulkových kalkulátorech nebo také v programu TennisBase Editor.

Uživatele sázkaře nezajímají tyto technické detaily, chce mít přístup k historickým výsledkům, pokud možno aktuálním. Typicky porovnává výsledky dvou hráčů, a proto nepožaduje žádné složité filtrování či rozsáhlé statistiky. Tyto funkce jsou poskytovány aplikací TennisBaseViewer.

K tvorbě aktualizačních balíčků pro začlenění nových výsledků z CSV souborů do databáze slouží jejímu správcovi aplikace TennisBase Convertor.

7.1 Hardwarové a softwarové nároky

Program ke svému chodu potřebuje počítač s operačním systémem Microsoft Windows 2000 nebo novější. K instalaci, která probíhá z přiloženého média, je potřeba mít na disku k dispozici alespoň 50 MB volného místa. Není vyžadována přítomnost žádného dalšího speciálního software. Přesné minimální hardwarové nároky nebyly stanoveny, konfigurace počítače by měla odpovídat použitému operačnímu systému. Funkční otestovaná konfigurace je uvedena v tabulce 7.1.

PC AT:	AMD Athlon 1,2 GHz
RAM:	256 MB
Grafická karta:	NVidia GeForce2 400MX 32MB
Operační systém:	MS Windows XP Professional SP1

Tabulka 7.1 - Funkční otestovaná konfigurace.

7.2 Instalace

Pro účely této práce jsou všechny aplikace distribuovány v jednom instalačním balíku. Ten je umístěn na přiloženém CD v souboru „setup.exe“ v kořenovém adresáři. Instalační program umožňuje snadné a uživatelsky přívětivé nainstalování a odinstalování aplikace. Pro případné zájemce o manuální způsob instalace jsou na CD ve složce „/bin” připraveny potřebné soubory.

Instalační balíček obsahuje následující:

- TennisBase Builder s potřebnými filtry pro převod a vzorovými daty;
- TennisBase Editor s několika připravenými databázemi;
- TennisBase Convertor s připravenou rozdílovou databází;
- TennisBase Viewer s databází.

Instalační program nakopíruje uvedené programy na disk a vytvoří zástupce na místech dle volby uživatele.

7.3 TennisBase Builder

TennisBase Builder je určen k převodu textových souborů s výsledky tenisových zápasů do CSV databáze. Ta může být dále zpracována tabulkovým procesorem, TennisBase Editorem, či v rámci aktualizačního balíčku vytvořeného programem TennisBase Convertor poskytnuta uživatelům TennisBase Viewer.

V intuitivním uživatelském prostředí můžeme snadno editovat konfigurační sekvenci určující, které ze vstupních souborů budou zpracovány dle kterého souboru příkazů. Tuto sekvenci můžeme také uložit na disk nebo z disku nahrát. Dvě políčka nám umožňují specifikovat množinu vstupních souborů a soubor výstupní databáze. Převod lze spustit stisknutím tlačítka „Start“.

7.3.1 Konfigurační sekvence

Konfigurační sekvence se skládá z instrukcí dvou druhů:

- instrukce pro vytvoření struktury zpracování zapsaná ve tvaru „*Začátek_jména_souboru#Název_části_soutěže#Soubor_s_ instrukcemi*“, kde speciální znak „$“ zastupuje řetězec *Začátek_jména_souboru* hodnotu s nulovou délkou;
- instrukce pro začátek zpracování jedné kategorie turnajů zapsané ve tvaru „*#Označení_kategorie_turnajů*“.

Ukažme si to na příkladě:

```
a-#Main Draw#atp.tbf
q-#Qualification#atp-q.tbf
pq-#Pre-Qualification#atp-pq.tbf
#ATP Tour#
```

Uvedená konfigurační sekvence se zpracovává následovně. V prvním kroku je pro každý vstupní soubor provedena kontrola, zda odpovídá masce „a-*.*“. Dejme tomu, že je nalezen soubor „a-adelaide“. V tom případě je zpracován dle souboru s instrukcemi „atp.tbf“. Program se poté pokusí najít soubory „q-adelaide.txt“ a „pq-adelaide.txt“. Ty jsou, v případe že existují, zpracovány pomocí souborů s instrukcemi z odpovídajících souborů, zde to jsou „atp-q.tbf“ resp. „atp-pq.tbf“. V ostatních případech se postupuje obdobně.

7.3.2 Soubory s instrukcemi

Soubor s instrukcemi je vlastně návod, který transformačnímu automatu říká, jakým způsobem má zpracovat vstupní data. Jelikož je s programem TennisBase Builder dodávána kompletní sada těchto instrukčních souborů, není pro běžné používání znalost jejich struktury potřeba. Podrobnější popis transformačního automatu lze nalézt v části 6.1.3. Syntaxe příkazů je pro případné zájemce uvedena jako dodatek B.

Přesto si v krátkosti prohlédněme strukturu souboru s instrukcemi, jehož část je na obrázku 7.1. Jednotlivé příkazy jsou interpretovány po sloupcích zleva doprava a jsou pro větší názornost odděleny horizontální čarou. První dva sloupce definují čtení hlavičky turnaje. Instrukcí „CLEARBUFFER“ ve spodní části druhého sloupce začíná fáze čtení těla turnaje. Sekce od návěstí „#zapasy“ po navěstí „#konec“ odpovídá zpracování jednotlivých kol turnaje. Příkaz „CLEARTOURNAMENT“ uloží výsledky zpracování do databáze a příkaz „HALT“ zpracování ukončí.

```
BLOCK_SEP
_________
READBLOCK
ADD
0
ATP Tour
ADD
1
Main Draw
SAVE_TTYPE
1
SAVE_TTYPE2
2
SAVE_NAME
3
SAVE_LOCATION
4
SAVE_DATE
5
_________
```

```
PARSE!
7
-
1
0
SAVE_SURFACE
8
PARSE!
6
-
1
0
SAVE_DOTATION
7
CLEARBUFFER
FIND
SINGLES
IFEOF
1
HALT
____
```

```
READBLOCK
CLEARBUFFER
#zapasy
READBLOCK
IFBUFFREE
2
GOTO
konec
NEXTROUND
1
FLUSH
1
PARSEGAMES
GOTO
zapasy
#konec
CLEARTOURNAMENT
HALT
```

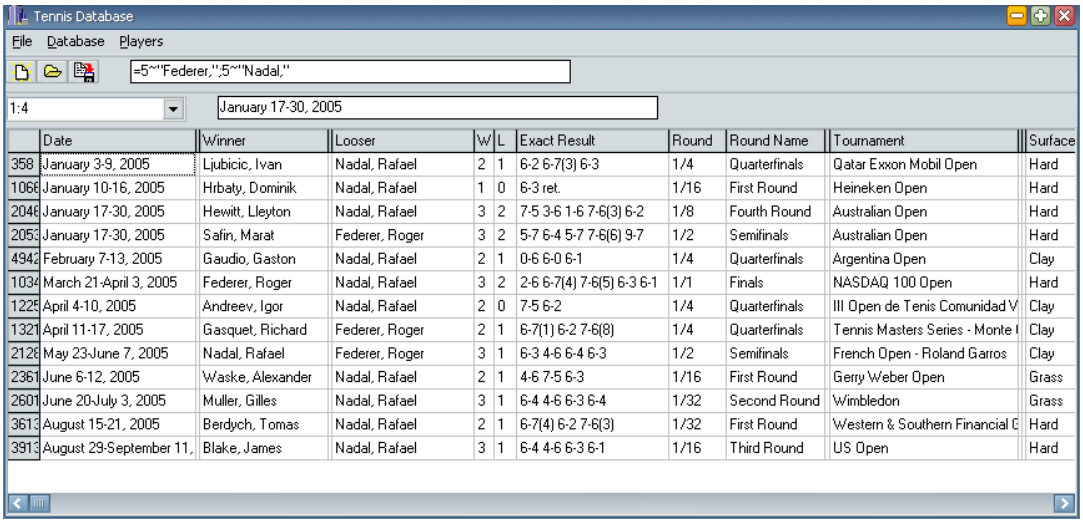
Obrázek 7.1 - Příklad instrukčního souboru TennisBase Builder. Instrukce jsou interpretovány po sloupcích zleva doprava.

7.4 TennisBase Editor

TennisBase Editor slouží ke správě databáze vytvořené programem TennisBase Builder. K obsluze programu slouží jednoduché grafické rozhraní s několika ovládacími prvky (viz obrázek 7.2).

Nejdůležitější implementovaná funkce:

- filtrování tabulky s využitím jednoduchého dotazovacího jazyku, tedy např. dle jmen hráčů, dle vítěze či poraženého, dle počtu odehraných setů, místa konání či názvu turnajů, hracího povrchu, kategorie soutěže, ...;
- editovací funkce (vložení zápasu, úprava zápasu, řazení, ...).



	Date	Winner	Looser	W	L	Exact Result	Round	Round Name	Tournament	Surface
358	January 3-9, 2005	Ljubicic, Ivan	Nadal, Rafael	2	1	6-2 6-7(3) 6-3	1/4	Quarterfinals	Qatar Exxon Mobil Open	Hard
1068	January 10-16, 2005	Hrbaty, Dominik	Nadal, Rafael	1	0	6-3 ret.	1/16	First Round	Heineken Open	Hard
2048	January 17-30, 2005	Hewitt, Lleyton	Nadal, Rafael	3	2	7-5 3-6 1-6 7-6(3) 6-2	1/8	Fourth Round	Australian Open	Hard
2053	January 17-30, 2005	Safin, Marat	Federer, Roger	3	2	5-7 6-4 5-7 7-6(6) 9-7	1/2	Semifinals	Australian Open	Hard
4942	February 7-13, 2005	Gaudio, Gaston	Nadal, Rafael	2	1	0-6 6-0 6-1	1/4	Quarterfinals	Argentina Open	Clay
1034	March 21-April 3, 2005	Federer, Roger	Nadal, Rafael	3	2	2-6 6-7(4) 7-6(5) 6-3 6-1	1/1	Finals	NASDAQ 100 Open	Hard
1225	April 4-10, 2005	Andreev, Igor	Nadal, Rafael	2	0	7-5 6-2	1/4	Quarterfinals	III Open de Tenis Comunidad V	Clay
1321	April 11-17, 2005	Gasquet, Richard	Federer, Roger	2	1	6-7(1) 6-2 7-6(8)	1/4	Quarterfinals	Tennis Masters Series - Monte	Clay
2128	May 23-June 7, 2005	Nadal, Rafael	Federer, Roger	3	1	6-3 4-6 6-4 6-3	1/2	Semifinals	French Open - Roland Garros	Clay
2361	June 6-12, 2005	Waske, Alexander	Nadal, Rafael	2	1	4-6 7-5 6-3	1/16	First Round	Gerry Weber Open	Grass
2601	June 20-July 3, 2005	Muller, Gilles	Nadal, Rafael	3	1	6-4 4-6 6-3 6-4	1/32	Second Round	Wimbledon	Grass
3613	August 15-21, 2005	Berdych, Tomas	Nadal, Rafael	2	1	6-7(4) 6-2 7-6(3)	1/32	First Round	Western & Southern Financial G	Hard
3913	August 29-September 11,	Blake, James	Nadal, Rafael	3	1	6-4 4-6 6-3 6-1	1/16	Third Round	US Open	Hard

Obrázek 7.2 - Uživatelské prostředí programu TennisBase Editor.

7.4.1 Zadávání filtrů

Nejsnazší možností je zadávat dotazy přímo z menu aplikace. Zde jsou předem připraveny tři nejčastější dotazy: zápasy jednoho hráče, zápasy dvou hráčů proti sobě a výpis všech zápasů dvou hráčů.

Druhou možností je možnost položení dotazu přímo do vstupního boxu. Dotazy nnejsou citlivé na velikost písmen a rozlišují se na dva typy. Dotazu, který nezačíná znakem „=“, vyhovují všechny řádky, kde se v jakémkoliv sloupci vyskytuje řetězec dotazu jako podřetězec. Dotazy uvedené znakem „=“ jsou složitější a mají následující strukturu:

- znak „,“ má význam logické spojky AND;
- znak „;” má význam logické spojky OR;
- řetězci „*číslo ~ dotaz*“ vyhovují řádky, kde sloupec *číslo* obsahuje řetězec *dotaz* jako podřetězec;
- řetězci „*číslo = dotaz*“ vyhovují řádky, kde sloupec *číslo* obsahuje právě řetězec *dotaz;*
- řetězec *dotaz* může být zadán v uvozovkách, aby došlo k zamaskování speciálních znaků.

Např. dotazu zapsanému jako „*=3="roddick, andy",5~coria,g*“ vyhovují všechny řádky, kde se třetí sloupec rovná řetězci „*roddick, andy*“, kde se v pátém sloupci vyskytuje řetězec *coria* a kde nějaký sloupec obsahuje písmeno „*g*“.

7.5 TennisBase Convertor

TennisBase Convertor je určen k převodu CSV databáze do databáze vhodné pro TennisBase Viewer. V přehledném grafickém rozhraní lze nastavit zdrojové a cílové soubory. Po stisknutí tlačítka „Read“ je do paměti načtena databáze ze souborů definovaných v položkách uvedených nad tlačítky. Po stisknutí tlačítka „Start !“ dojde ke zpracování souboru uvedeného v položce „CSV format file“ vůči načtené databázi. Výsledkem jsou nové aktualizační soubory s příponou definovanou v položce „New file extension“. Aktualizace obsahuje pouze nové záznamy či záznamy obsahující změněné položky databáze.

7.6 TennisBase Viewer

Uživatelské prostředí TennisBase Viewer je tvořeno jedním oknem. V něm jsou k dispozici čtyři vstupní boxy, určené pro zadání jmen soupeřů, hracího povrchu a data odehrání zápasu. Filtrování dle data a povrchu probíhá okamžitě. Pokud provádíme výběr hráče zápisem jeho jména, je nutno volbu potvrdit stisknutím šipky dolů. Výsledek hledání se objevuje ve třech záložkách:

- „Player 1 Profile“, kde jsou uvedeny údaje o prvním hráči a jeho zápasové výsledky, které jsou případně omezeny dle nastavení filtru data a povrchu;
- „Player 2 Profile“, kde jsou uvedeny stejným způsobem informace o druhém hráči;
- „Head To Heads“, kde jsou uvedeny nefiltrované výsledky jejich vzájemných zápasů.

Čtvrtá záložka „Tournaments“ slouží k vyhledávání informací o kategoriích turnajů a jejich konkrétních instancích.

Uživatelské prostředí dovoluje modifikovat font použitý k zobrazení výsledků, samozřejmostí je také možnost přizpůsobit si zobrazovanou šířku sloupců tabulky. Pokud máme k dispozici aktualizační balíček, můžeme databázi volbou z menu snadno aktualizovat.

Aplikaci je možno využít k exportu aktuálního pohledu do formátu CSV. Po vybrání příslušné položky v menu je na místě určeném uživatelem vytvořen CSV soubor s údaji o zápasech, které jsou obsaženy v právě aktivní záložce.

8 Závěr

Na základě naší představy o tenise, která je vymezena tenisovými pravidly a získanými empirickými daty, jsme vytvořili matematický model tenisového zápasu. Jeho chování je ovlivněno dvěma základními parametry: pravděpodobností výhry obou hráčů v rozehrách, v nichž podávají. To jsme ilustrovali na několika příkladech. Ukázali jsme metodu, jak zkombinovat reálné údaje o servisu hráčů tak, abychom z nich mohli odvodit parametry modelu. Výsledky jeho počítačové implementace nám umožnily mimo jiné kvantifikovat odhady očekávaného počtu her v zápase, počtu a rozložení setů a výsledku zápasu. To nám dovolilo porovnat simulovaná data se získanými empirickými daty. Zjistili jsme ale, že náš model předpovídá více her a obecně delší zápasy, než je ve skutečnosti pozorováno. Jedním z důvodů této odlišnosti, může být předpoklad statických pravděpodobností zisku rozehry během celého průběhu zápasu. V jedné z kapitol jsme si představili další možné způsoby předpovídání výsledků.

Empirická data jsou získávána námi vytvořenými nástroji z publikovaných textových souborů na Internetu, které jsou převedeny do textové databáze. Tu lze snadno importovat do tabulkového kalkulátoru či prohlížet dalšími našimi aplikacemi.

Podařilo se nám tedy splnit náš cíl, vytvořit počítačový nástroj pro získávání tenisových výsledků ve formátu použitelném k dalšímu zpracování. Matematický model, který jsme s využitím získaných dat vystavěli, umožňuje předpovídat různé charakteristiky tenisového zápasu. Budoucí vývoj modelu by měl směrovat k jeho úpravě, která by umožnila vysvětlit v jeho rámci zatím nevysvětlitelné anomálie, jako je např. únava hráčů. Jeho další modifikací by mohla být online přizpůsobení parametrů během průběhu zápasu.

9 Literatura

[1] International Tennis Federation, 2006. Rules of Tennis 2007. ITF Ltd., London.

[2] Stefani, R.T., 1997. Survey of the major world sport rating systems. *Journal of Applied* Statistics 24, 635-646.

[3] Clarke, S.R., Dyte, D., 2000. Using official ratings to simulate major tennis tournaments. *International Transactions in Operational Research* 7, 585-594.

[4] ATP Tour, 2007. The 2007 ATP® Official Rulebook. ATP Tour Inc., United States of America.

[5] Kemeny J.G., Snell, J.L., 1960. Finite Markov chains. D. Van Nostrand, Princeton, New Jersey.

[6] Fischer, G., 1980. Exercise in probability and statistics, or the probability of winning at tenis. *American Journal of Physics*. 48(1), 14–19.

[7] Carter W.H., Crews, S.L., 1974. An analysis of the game of tenis. *The American Statistician* 28 (4), 130–134.

[8] Hsi, B.P., Burych, D.M., 1971. Games of two players. *Applied Statistics* 20, 86–92.

[9] Carter, W.H., Crews, S. L., 1974. An analysis of the game of tennis. *American Statistician* 28(4), 130–134.

[10] Pollard, G.H., 1983. An analysis of classical and tie-breaker tenis. *The Australian Journal of Statistics* 25(3), 496–505.

[11] Barnett, T.., Clarke, S.R., 2005. Combining player statistics to predict outcomes of tennis matches. *IMA Journal of Management Mathematics* 16, 113−120.

[12] Barnett, T.J., 2006. Mathematical modelling in hierarchical games with specific reference to tennis. Swinburne University, Melbourne.

[13] Magnus, J.R., Klaassen, F. J. G. M., 1999. On the advantage of serving first in a tennis set: Four years at Wimbledon, *The Statistician* 48, 247–256.

[14] Jackson, D., Mosurski, K., 1997. Heavy defeats in tennis: Psychological momentum or random effects. *Chance* 10, 27–34.

[15] Klaassen, F. J. G. M., Magnus, J.R., 2001. Are points in tennis independent and identically distributed? Evidence from a dynamic binary panel data model, *Journal of the American Statistical Association* 96 (454), 500–509.

[16] Newton, P.K., Keller J.B., 2005. Probability of Winning at Tennis I. Theory and Data. *Studies in Applied Mathematics* 114 (3), 241–269.

[17] Dupač, D., Dupačová, J., 1980. Markovovy procesy I. SPN Praha, Praha.

[18] Elo, 1978. The Rating of Chess Players, Past and Present. Batsford, London.

[19] Strauss, D., Arnold, B.C., 1987. The rating of players in racketball tournaments. *Journal of Applied Statistics* 36, 163-173.

[20] Clarke, S.R., 1994. An adjustive rating system for tennis and squash players. In: de Mestre, N. (Ed.), *Mathematics and Computers in Sport*. Bond University, Gold Coast, Qld, 43-50.

[21] Courneya, K., Carron, A., 1992. The Home Advantage in Sport Competitions: a Literature Review. *Journal of Sport and Exercise Psychology* 14: 13-27

[22] Stefani, R.T., Clarke, S.R., 1992. Predictions and Home advantage for Australian Rules Football. *The Journal of Applied Statistics*. 19, 2, 251-261.

[23] Harville, D.A., 1980. Predictions for national football league games via linear-model methodology. *Journal of the American Statistical Association* 75, 516-524.

[24] Scheibehenne, B., Bröder, A., 2007. Predicting Wimbledon tennis results 2005 by mere player name recognition. International Journal of Forecasting.

[25] Töpfer, P., 1995. Algoritmy a programovací techniky. Prometheus, Praha.

Dodatek A Obsah přiloženého CD-ROM

Přiložené CD-ROM obsahuje soubory balíku TennisBase se zdrojovými kódy, ukázky implementace modelu, a elektronickou formu této práce ve formátu PDF.

Adresářová struktura obsahuje:

/ – soubor setup.exe pro instalaci celého balíku TennisBase;

/bin – soubory potřebné pro volitelnou manuální instalaci;

/data – ukázky zdrojových dat, souborů převodních instrukcí transformačního automatu TennisBuilder, výstupních CSV databází;

/sim – ukázky implementace matematického modelu v programu Microsoft Office Excel;

/src – zdrojové kódy programů balíku TennisBase;

/text – elektronickou verzi práce ve formátu PDF.

Dodatek B Syntaxe příkazů

Syntaxe příkazů použitých v instrukčních souborech transformačního automatu v programu TennisBase Builder.

```
HALT            - ukončí parsování

IFBUFLENLWR     - pokud je počet řetězců v bufferu menší než cele_číslo1,
cele_cislo1       nastaví začátek čtení za řádek cele_cislo2,
cele_cislo2       jinak o cele_cislo2 řádků dál

IFBUFFREE       - pokud je buffer prázdný,
cele_cislo1       nastaví začátek čtení za řádek cele_cislo1,
                  jinak o cele_cislo1 řádků dál

SKIPLINE        - přeskočí ve vstupním souboru následující řádku

GOTO            - najde v souboru filtru řádek, který se rovná: #string1
string1           a nastaví začátek čtení za tento řádek

JUMP            - přeskočí v souboru filtru následujících cele_cislo1 řádek
cele_cislo1

IFEOF           - pokud je vstupní soubor celý přečten,
cele_cislo1       nastaví začátek čtení za řádek cele_cislo1,
                  jinak o cele_cislo1 řádků dál

FIND            - čte vstupní soubor dokud nenajde se nějaká řádka rovnající
string1           se string1 bez ohledu na velikost znaků

FIND+           - čte vstupní soubor dokud nenajde se nějaké řádce slovo
string1           string1 bez ohledu na velikost znaků

AGAIN           - nastaví čtení vstupního souboru na jeho začátek

ADD             - přidá na cele_cislo1 pozici bufferu text string1
cele_cislo1
string1

READBLOCK       - přečte do bufferu vstupní soubor až po řádku ekvivalentní
                  s separátorem bloku (viz BLOCK_SEP), implicitně prázdná
                  řádka bez ohledu na velikost znaků

BLOCK_SEP       - nastaví separátor bloku na hodnotu string1
string1

READLINE        - přečte do bufferu jednu řádku ze vstupního souboru
```

```
PARSE          - rozdělí cele_cislo1 řetězec bufferu na maximálně cele_cislo2
cele_cislo1      řetězců dle separátoru string1 (pokud cele_cislo1 = -1, pak
string1          nekonečně), vynechá cele_cislo3 výskytů separátoru string1,
cele_cislo2      výsledných několik řetězců (>=1) nahradí původní řetězec
cele_cislo3      bufferu, implicitně separátor není ve výsledku (viz. PARSE+)

PARSE+         - změní stav toho, zda separátor je ve výsledku

PARSE+! či PARSE!+ provede PARSE+ a zavolá PARSE!

PARSE!         - jako PARSE, jen cele_cislo2 není maximum, ale přesný počet
cele_cislo1      výsledných řetězců (jsou případně doplněny prázdné řetězce)
string1
cele_cislo2
cele_cislo3

SAVE_SURFACE   - uloží jako hrací povrch cele_cislo1 řetězec bufferu
cele_cislo1

SAVE_NAME      - uloží jako jméno turnaje cele_cislo1 řetězec bufferu
cele_cislo1

SAVE_LOCATION  - uloží jako místo konání turnaje cele_cislo1 řetězec bufferu
cele_cislo1

SAVE_TTYPE     - uloží jako typ turnaje (okruh) cele_cislo1 řetězec bufferu
cele_cislo1

SAVE_TTYPE2    - uloží jako druh losu (kvalifikace/hlavní/..) cele_cislo1
cele_cislo1      řetězec bufferu

SAVE_DATE      - uloží jako datum cele_cislo1 řetězec bufferu
cele_cislo1

SAVE_DOTATION  - uloží jako dotaci (prize money) turnaje cele_cislo1
cele_cislo1      řetězec bufferu

SAVE_PRIZEMONEY funguje jako SAVE_DOTATION
cele_cislo1

CLEAR_BUFFER   - vyprázdní buffer

CLEAR_TOURNAMENT - zpracuje turnaj (zapíše na výstup)

FLUSH          - vymaže z bufferu řetězec číslo cele_cislo1

NEXTROUND      - přidá do turnaje další kolo

PARSEGAMES     - rozparsuje řetězce z bufferu jako zápasy 1 kola,
                 každý řetězec je 1 zápas

PARSEGAME      - totéž, ale jen první řetěz bufferu
```