

Darstellung von Eishockeystatistiken mit parallelen Koordinatensystemen

Informatikpraktikum an der TU Wien

Institut für Computergrafik und Algorithmen

Erstellt von Benjamin Beer

Betreuer: Prof. Meister Eduard Gröller und Dr. Matej Mlejnek

Datum: 17.10.2011

Inhalt

Abstract	2
1 Einleitung.....	3
2 Implementierung.....	3
2.1 Parvis	3
2.2 GUI.....	3
2.3 Ordnungen der Daten.....	3
2.4 Berechnungen von ähnlichen Datensätzen.....	4
2.5 Erweiterung der Brush Funktion	5
2.6 Sonstiges.....	7
3 Fragestellungen	8
3.1 Verletzte Torjäger (NHL).....	8
3.2 Harte, aber faire Spieler (NHL)	9
3.3 Offensive Powerplay Abwehrspieler (NHL)	9
3.4 Wichtigkeit von Verlängerungen (EBEL).....	11
4 Kurz-Bedienungsanleitung	13
5 Zusammenfassung.....	14

Abstract

In diesem Projekt wurde die Software "Parvis", ein Programm zur Darstellung von parallelen Koordinaten, so erweitert, dass es sich gut zur Darstellung von Eishockeystatistiken eignet. Die Interaktions- und Darstellungsmöglichkeiten wurden dahingehend ausgebaut, dass man interessante Fragestellungen beantworten kann. Besonders wichtig sind die Möglichkeiten des Brushings und des Hervorhebens bestimmter Datensätze. Außerdem können die Achsen so sortiert werden, dass es möglichst wenig Überschneidungen gibt, um die Übersichtlichkeit zu erhöhen.

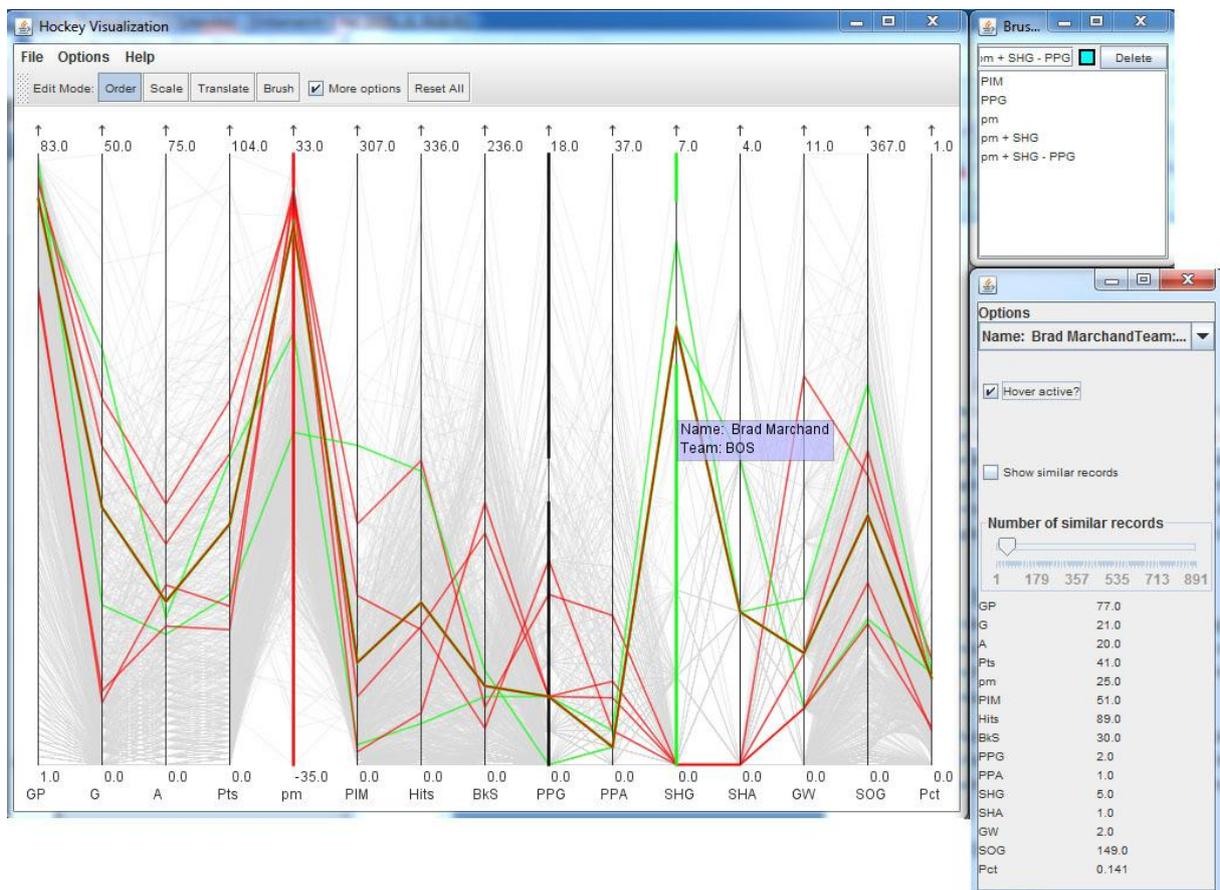


Abb. 1 Die GUI der Hockey Visualization Software

1 Einleitung

Im Sport werden große Mengen an statistischen Daten gesammelt. Meist werden diese in einfachen Tabellen gesammelt, die dann von Interessenten mühsam nach interessanten Zusammenhängen durchsucht werden müssen. Ziel des Praktikums, war die Entwicklung eines parallelen Koordinatensystems, um diesen Missstand speziell für Eishockeystatistiken zu beheben. Die Software zur Darstellung des parallelen Koordinatensystems sollte nicht nur die reine Visualisierung der Tabellen übernehmen, sondern auch verschiedene Interaktionsmöglichkeiten bieten, um interessante Informationen aus den Daten herauslesen zu können.

2 Implementierung

2.1 Parvis

Grundsätzlich basiert das Programm auf Parvis, einem Projekt zur Darstellung von parallelen Koordinatensystemen. Parvis erlaubt das Einlesen und Darstellen von Daten und implementiert einige grundlegende Operationen, die für parallele Koordinatensysteme wichtig sind:

- Ein- und Ausblenden von Achsen
- Sortierung, Skalierung und die Translation von Achsen.
- Brushing: auf einer Achse kann durch das Ziehen mit der Maus ein Wertebereich ausgewählt werden. Es werden dann nur noch die Datensätze angezeigt, die sich in diesem Bereich befinden.
 - o Durch drücken von Alt, Strg oder Shift während der Auswahl, können Brushes auch addiert, subtrahiert und geschnitten werden.
 - o Im ursprünglichen Parvis gibt es auch einen „angular brush“, dieser wird aber nicht berücksichtigt, da sich für dieses Projekt kein Anwendungsfall eignet und wurde deshalb auch deaktiviert.
 - o Brushes können auch mit unterschiedlichen Farben in einer Liste abgespeichert und wieder aufgerufen werden.
- Details der Datensätze mit Mouse-Over: aktiviert man den Mouse-Over Effekt, wird der Datensatz, über dem gerade die Maus steht, hervorgehoben und der Name des Datensatzes, sowie auf Wunsch auch die genauen Werte der jeweiligen Achsen angezeigt.

Dies sind die wichtigsten Funktionen von Parvis, die auch in das neue Projekt, zumindest teilweise, übernommen werden.

2.2 GUI

Da nicht alle Funktionen benötigt werden, die in Parvis vorhanden sind, aber dafür weitere hinzugefügt werden sollen, wird eine komplett neue GUI Klasse erstellt. Die GUI wird in Kapitel 4 näher beschrieben.

2.3 Ordnungen der Daten

Es wurden zwei Methoden implementiert, um die Achsen auf eine intelligente Art anzuordnen. Im Allgemeinen gilt, je weniger Überschneidungen im Graphen sind, desto übersichtlicher wird er. Um die Berechnungen sinnvoll durchführen zu können, werden die Werte aller Daten immer zuerst normiert zwischen 0 und 1.

Kleinsten Weg berechnen:

Die erste Möglichkeit ist zu versuchen, dass die Achsen so angeordnet werden, dass die Gesamtwege aller Datensätze von Achse zu Achse minimal werden. Dafür muss für jede Permutation der Achsen der Gesamtweg berechnet werden (da die x-Abstände der Achsen immer gleich sind, wird nur der normierte y-Wert zwischen den Punkten genommen, statt dem konkreten Abstand).

Wenigste Überschneidungen berechnen:

Bei diesem Ansatz wird für jede Permutation der Achsen die Anzahl an Überschneidungen der Datensätze berechnet und die Achsen schlussendlich so angeordnet, dass die Überschneidungen minimal sind. Um den Berechnungsvorgang zu beschleunigen, werden die Überschneidungen zwischen jeweils zwei Achsen vorberechnet, denn um alle Möglichkeiten zu erhalten, reicht die Kombination (ohne Wiederholung) von zwei Achsen aus allen Achsen. So gibt es zum Beispiel bei vier Achsen lediglich 6 verschiedene Summen der Überschneidungen aber 24 Möglichkeiten, wie die Achsen angeordnet werden können (Überschneidungen von Achse 1 und 2 = Anzahl Überschneidungen von Achse 2 und 1, etc., somit gibt es folgende Möglichkeit: (1,2), (1,3), (1,4), (2,3), (2,4), (3,4). Aus diesen kann jede beliebige Permutation (etwa 4,3,2,1 benötigt (1,2), (2,3) und (3,4)) berechnet werden.

2.4 Berechnungen von ähnlichen Datensätzen

Es wurde auch eine Funktion eingebaut, um ähnliche Datensätze berechnen zu können. Als Maßstab für die Ähnlichkeit wird die normierte Gesamtdistanz auf allen Achsen zum jeweiligen Ausgangsdatsatz herangezogen. Über einen Schieberegler kann man festlegen, wie viele ähnliche Datensätze berechnet werden sollen. Abb. 2 Berechnung ähnlicher Datensätze zeigt ein Beispiel dieser Funktion.

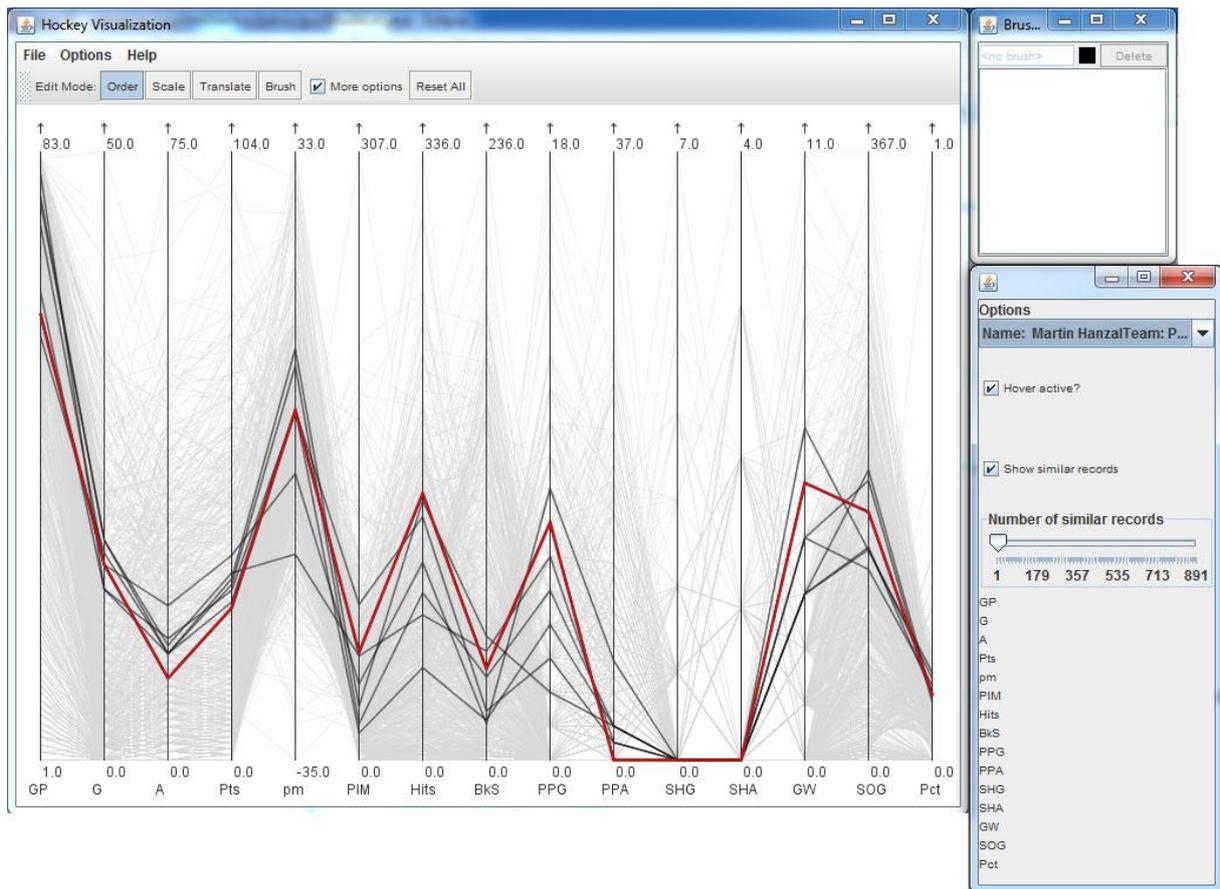


Abb. 2 Berechnung ähnlicher Datensätze

2.5 Erweiterung der Brush Funktion

Ein wesentlicher Fokus der Arbeit lag auf der Erweiterung und Verbesserung der Brush Funktion. Folgende Funktionen wurden hinzugefügt:

- Markierung der Achsen: Um besser zu sehen, welcher Bereich auf den Achsen gebrushed wurde, werden die Achsen nun oberhalb und unterhalb des Brushes markiert (der Bereich, der gebrushed wurde ist „durchlässig“). Siehe auch Abb. 3 Markierung der Achsen.
- Brushes werden nun automatisch in der Brushliste abgespeichert und es wird ihnen automatisch eine Farbe zugewiesen. In der Brushliste kann nun zwischen allen bisherigen Brushes navigiert werden. Die Brushes können auch wieder aus der Liste entfernt werden. (siehe Abb. 4 Brushliste wird automatisch erweitert)
- Durch das Drücken verschiedener Tasten können Brushes kombiniert werden (Shift = addieren, Strg = Schnitt und Alt = subtrahieren von Brushes). Auch diese Operationen werden automatisch in die Brushliste übernommen bzw. wird beim Drücken einer dieser Tasten eine neue Auswahl in der Liste getätigt, wird auch so die Operation durchgeführt.
- Um das Addieren von zwei Brushes besser zu erkennen, werden bei der Addition zwei unterschiedliche Farben für die jeweiligen Brushes gewählt. So erkennt man besser, welcher Bereich wo ausgewählt wurde.
- Um genauere Brushes zeichnen zu können, wird der Wert der Position des Mauszeigers neben dem Zeiger eingeblendet (nur wenn sich die Maus über einer Achse befindet).

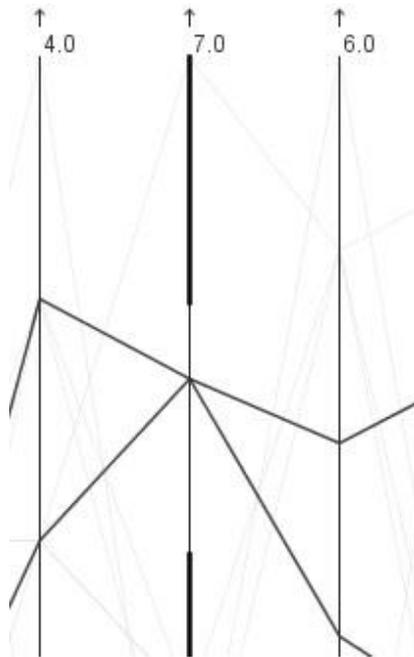


Abb. 3 Markierung der Achsen

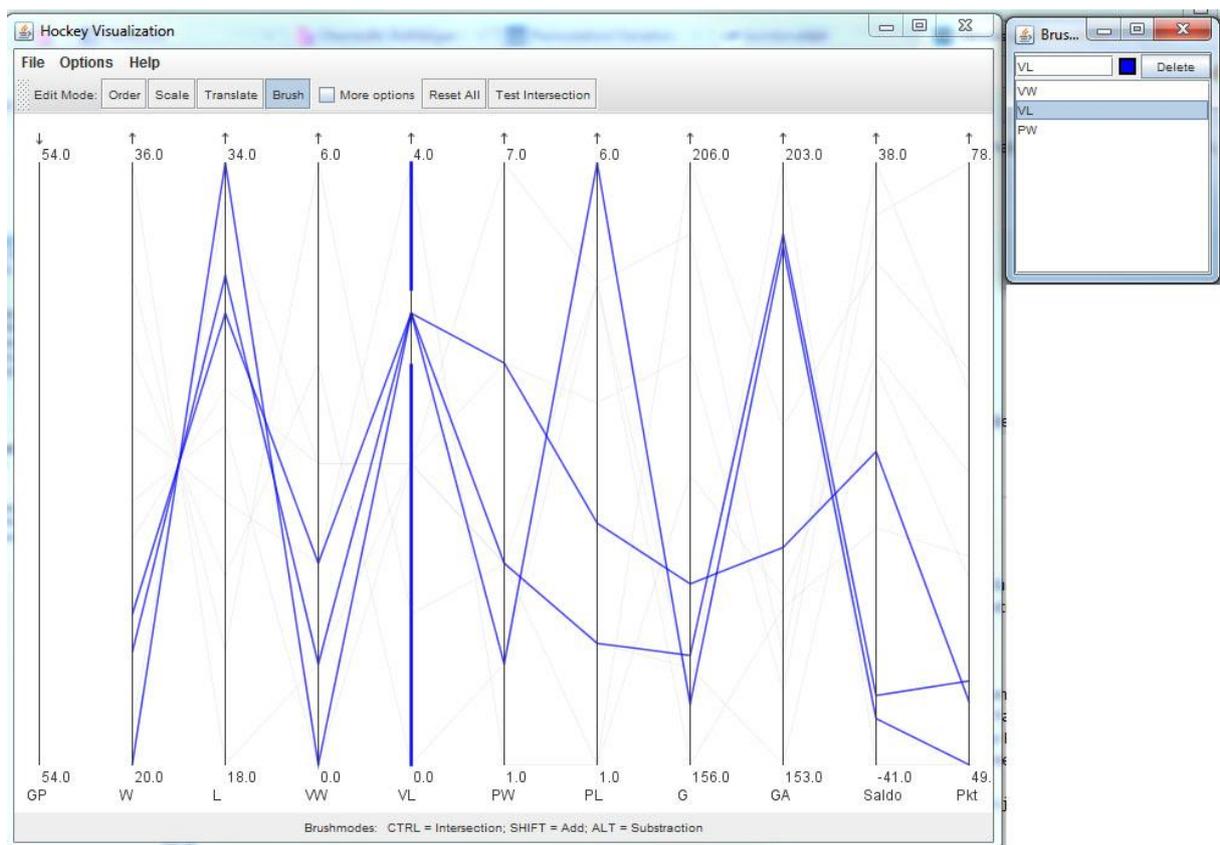


Abb. 4 Brushliste wird automatisch erweitert

2.6 Sonstiges

In Parvis ist ein sogenannter Hovereffekt verfügbar. Bei diesem wird der Datensatz farblich hervorgehoben, über dem gerade die Maus platziert ist. Darüber hinaus können für diesen Datensatz dann die jeweiligen Werte bei den jeweiligen Achsen angezeigt werden.

Diese Hoverfunktion wurde in mehreren Punkten abgeändert. Die hervorzuhebenden Datensätze können nun nicht mehr nur mit einem Mouse-Over Effekt ausgewählt werden, sondern auch in einer Liste. Die genauen Werte auf den jeweiligen Achsen werden nun auch in einer externen Liste angezeigt. Dadurch gibt es keine Überdeckungen von möglicherweise interessanten Datensätzen.

Sind sowohl der Hovereffekt als auch die Berechnung von ähnlichen Datensätzen gleichzeitig aktiv, so wird der jeweilige Datensatz hervorgehoben, zu dem die ähnlichen Datensätze berechnet werden.

Wenn Datensätze durch einen Brush ausgewählt wurden und der Hovereffekt aktiviert ist, so werden jeweils nur die Datensätze markiert, die im aktuellen Brush vorhanden sind. Von Datensätzen, die sich beim aktuellen Brush im Hintergrund befinden, wird nur der Name angezeigt, wenn sich die Maus dort befindet.

Die oben beschriebenen Funktionen werden in Abb. 5 Verbesserungen des Hovereffekts, dargestellt.

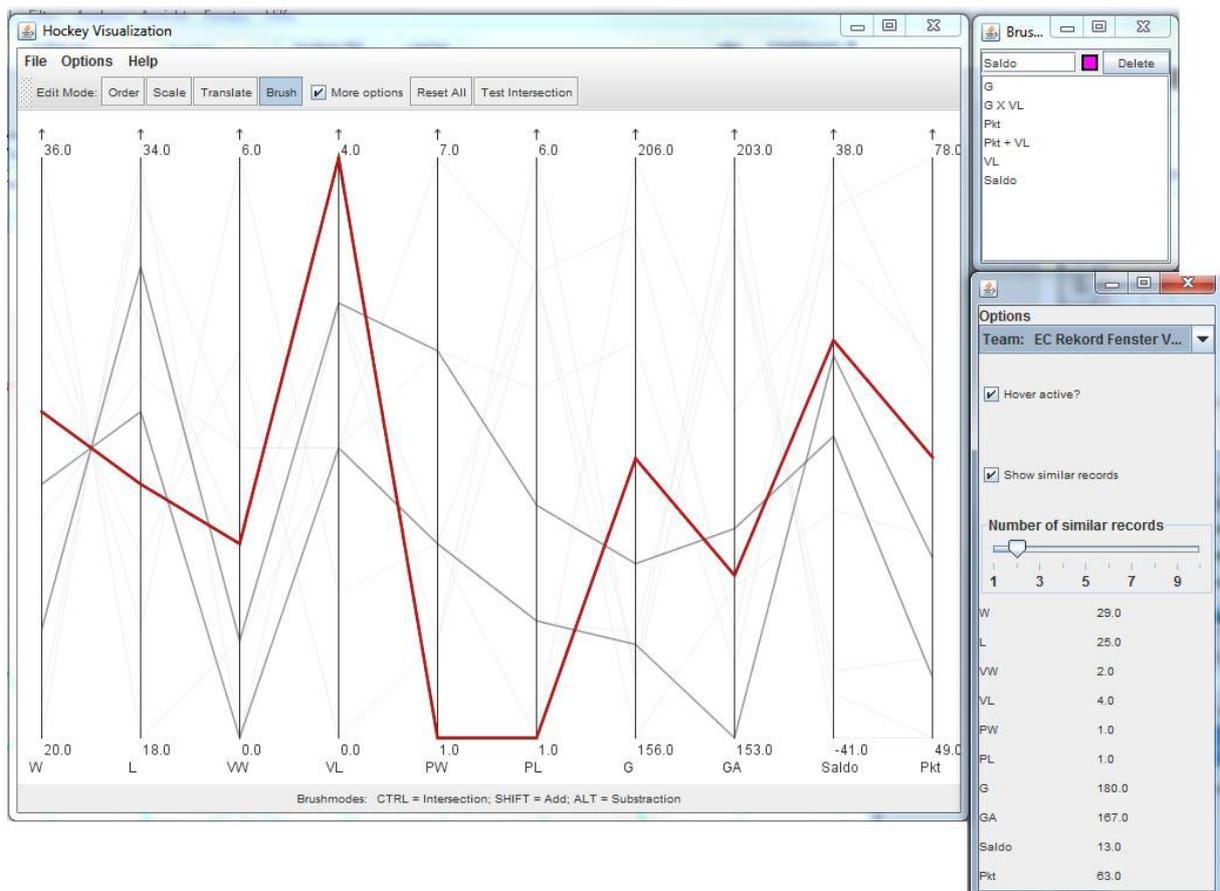


Abb. 5 Verbesserungen des Hovereffekts

3 Fragestellungen

Im folgenden Kapitel wird versucht, mit Hilfe des parallelen Koordinatensystems und den oben beschriebenen Techniken, einige Fragestellungen zu lösen. Als Grundlage der Daten dienen die Statistiken der Spieler aus dem NHL Grunddurchgang bzw. Statistiken über die Teams der Erste Bank Eishockeyliga. Um die Berechnung der wenigstens Überschneidungen zu verkürzen, wurden nur die jeweils wichtigen Statistiken verwendet.

3.1 Verletzte Torjäger (NHL)

Finde Spieler, die weniger als 65 Spiele gespielt haben, aber trotzdem 30 und mehr Tore geschossen haben:

Für diese Fragestellung sind die Dimensionen „Anzahl Spiele“ (GP) und „Tore“ (G) wichtig.

Um das gewünschte Ergebnis zu erhalten wird zuerst ein Brush auf der GP Achse von 0 bis etwa 65 gezogen und dann wird dieser Brush geschnitten mit einem Brush auf der „G“ Achse von 30 bis zum höchsten Wert. Über bleiben genau zwei Spieler: Sidney Crosby von den Pittsburgh Penguins mit 32 Toren in gerade einmal 41 Spielen und Drew Stafford von den Buffalo Sabres mit 31 Treffern in 62 Spielen.

(siehe Abb. 6 Verletzte Torjäger)

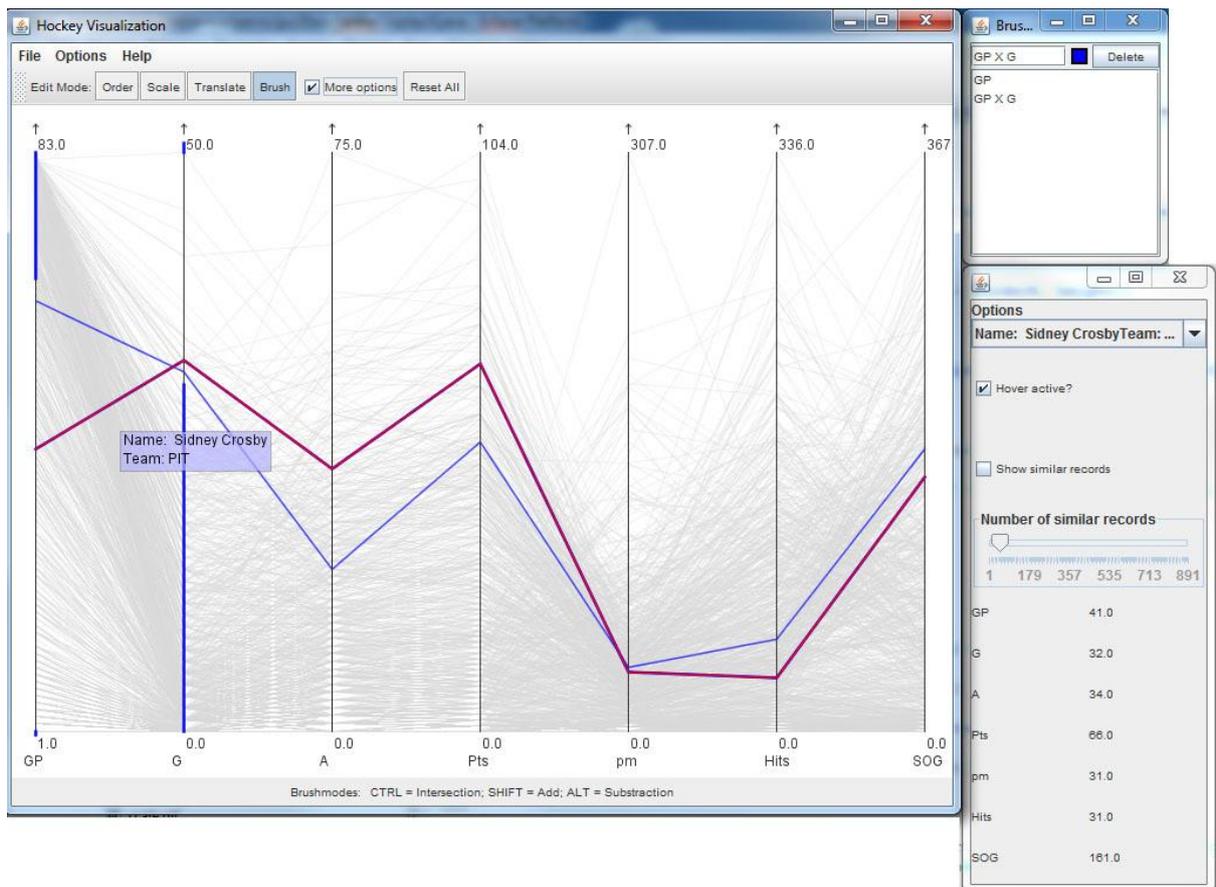


Abb. 6 Verletzte Torjäger

3.2 Harte, aber faire Spieler (NHL)

Finde Spieler, die viele Hits aber wenige Strafminuten (PIM) haben.

Ein Brush wird auf der Hits-Achse gelegt, und zwar werden nur Spieler gewählt, die über 280 Hits haben. Hier bleiben nur noch wenige Spieler über. Der Brush wird nun noch mit einem Brush auf der „pm“ Achse im unteren Bereich geschnitten. So bleiben drei Spieler über:

- Cal Clutterbuck von Minnesota Wild mit 336 Hits und nur 79 Strafminuten.
- Tuomo Ruutu von den Carolina Hurricanes mit 309 Hits aber nur 54 Strafminuten.
- Dustin Brown von den Los Angeles Kings mit 300 Hits und 67 Strafminuten.

(siehe Abb. 7 Harte aber faire Spieler)

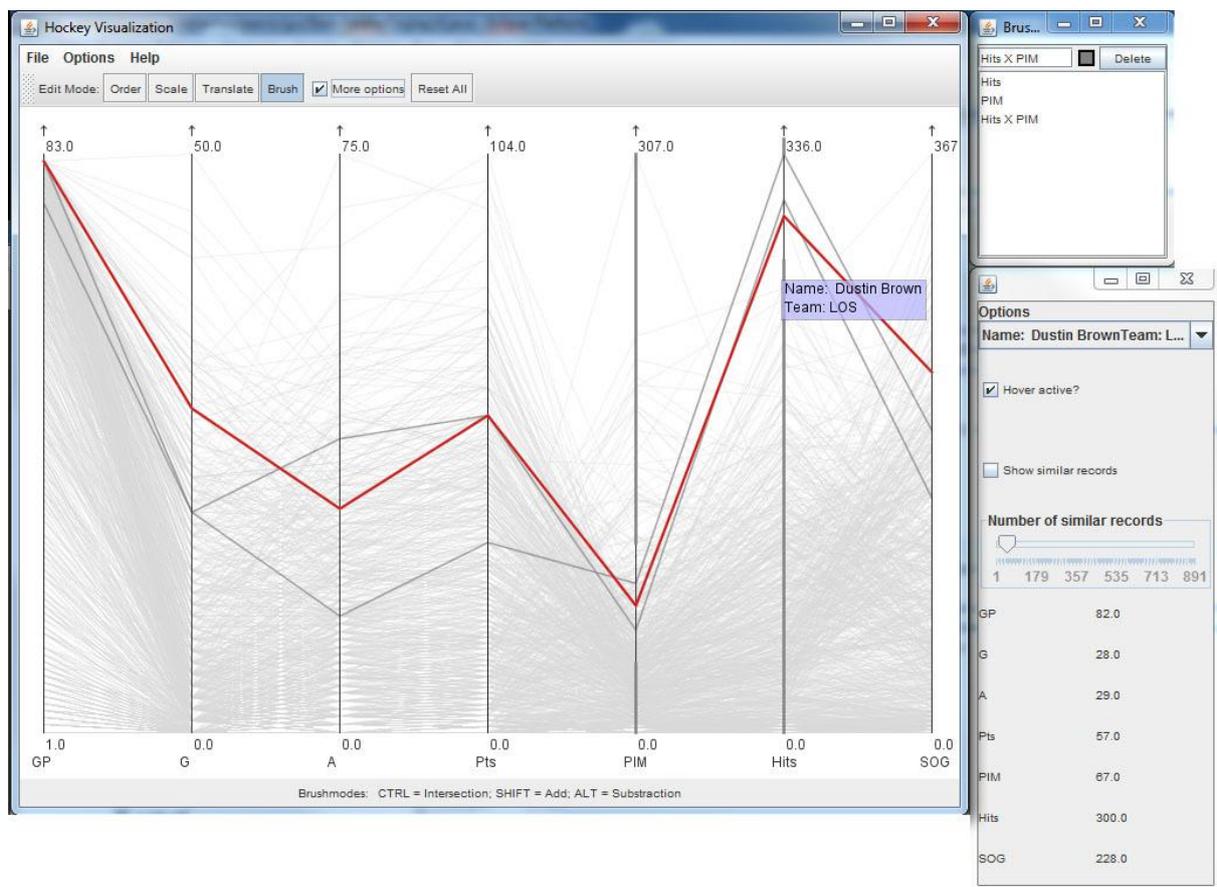


Abb. 7 Harte aber faire Spieler

3.3 Offensive Powerplay Abwehrspieler (NHL)

Find Spieler mit zumindest 3 SH Goals, die aber keine SH Assists haben.

Es wird wieder auf die gleiche Methode wie beim letzten Beispiel zurückgegriffen (Brush auf „SHA“, so dass alle Spieler mit keinem SH Assist ausgewählt sind, und diesen geschnitten mit Brush auf „SHG“ mit allen Spielern mit mindestens 3 SH Goals).

Es bleiben vier Spieler über:

- Matt Cullen (Position Center) von Minnesota mit 4 SHG.
- Milan Michalek (Position Left Wing) von Ottawa mit ebenfalls 4 SHG.

- Eric Staal (Position Center) von Carolina mit 3 SHG.
- R. J. Umberger (Position Left Wing) von Columbus.

Wenn man nur die SHG und SHA Achsen betrachten würde, könnte man nur zwischen 2 Spielern unterscheiden (2mal 0 SHA und 4 SHG und 2mal 0 SHA und 3 SHG). Erst durch den Blick auf die anderen vorhandenen Achsen erkennt man die vier Spieler.

(siehe Fehler! Verweisquelle konnte nicht gefunden werden.)

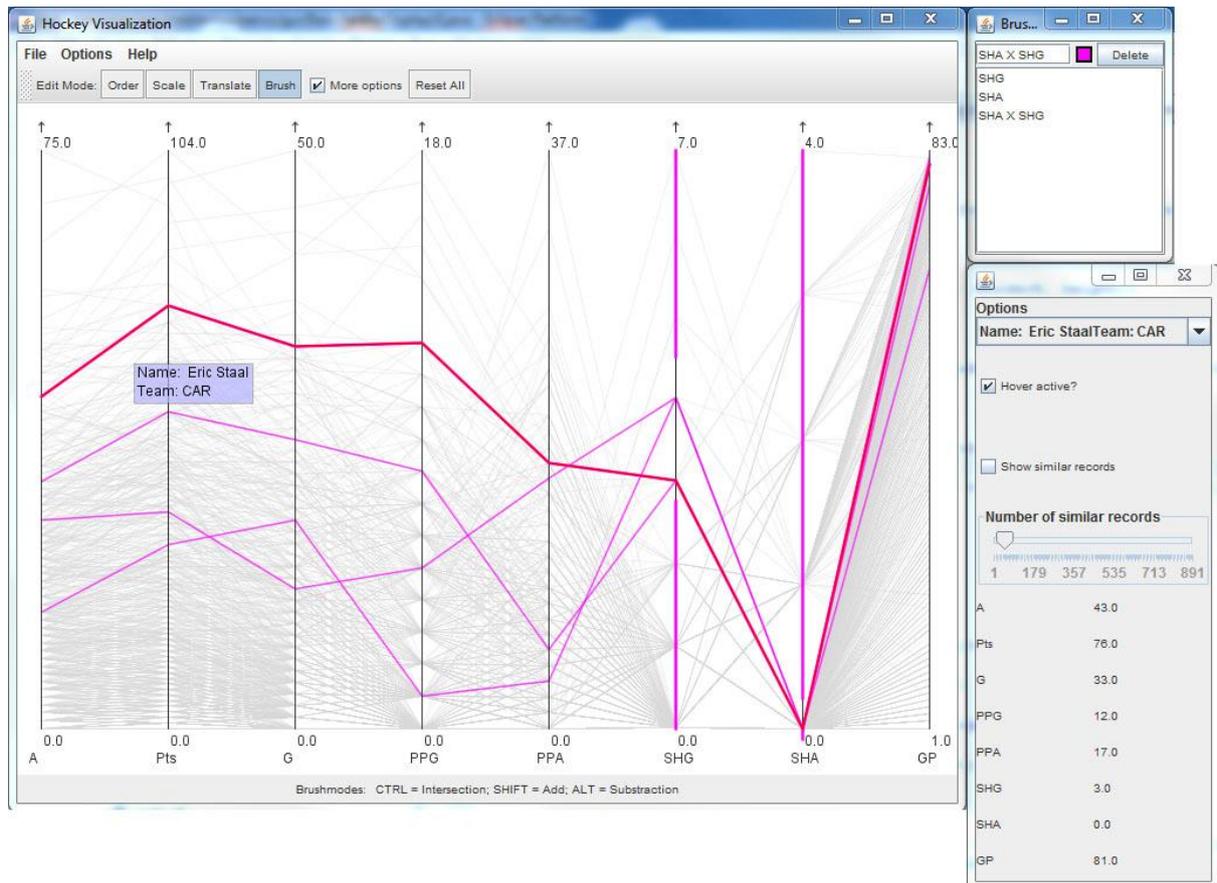


Abb. 8 Offensive Powerplay Abwehrspielen

Finde Spieler mit zumindest 45 Toren oder 65 Assists, oder 300 Hits, oder 320 Shots on Goal. Welcher von diesen Spielern hat am fairsten gespielt (am wenigstens Strafminuten (PIM))?

- **> 45 Tore:** nach einem entsprechenden Brush stellt man fest, dass es nur 2 Spieler mit mehr als 45 Toren gibt (Corey Perry und Steven Stamkos). Steven Stamkos hat aber nur 74 Strafminuten (Perry im Vergleich 104).
- **> 65 Assists:** Wieder gibt es nur 2 Spieler, die über 65 Assists haben (Henrik Sedin und Martin St. Louis). Louis hat dabei nur 12 Strafminuten (Sedin immerhin 40).
- **> 300 Hits:** Cal Clutterbuck und Tuomo Ruutu haben über 300 Hits, Ruutu hat dabei nur 54 Strafminuten gesammelt (Clutterbuck 79).
- **> 320 Shots on Goal:** Immerhin vier Spieler haben über 320 Schüsse aufs Tor, am wenigsten Strafminuten hat dabei Phil Kessel mit 24 (Ovechkin hat 42, Byfuglien 93 und Carter 39).

Diese Tests wurden mit einzelnen Brushes durchgeführt. Außerdem wurde versucht, alle Kriterien auf einmal zu erfüllen. Dafür werden auf der Tore, Assists, Hits und Shots on Goal Achse jeweils Brushes

addiert und anschließend der unterste markierte Record auf der Strafminuten Achse gehovort. Dieser ist (wie man aus den obigen Tests bereits erkennen kann) Martin St. Louis mit gerade einmal 12 Strafminuten. (siehe Abb. 9 Fairste Spieler unter verschiedenen Bedingungen)

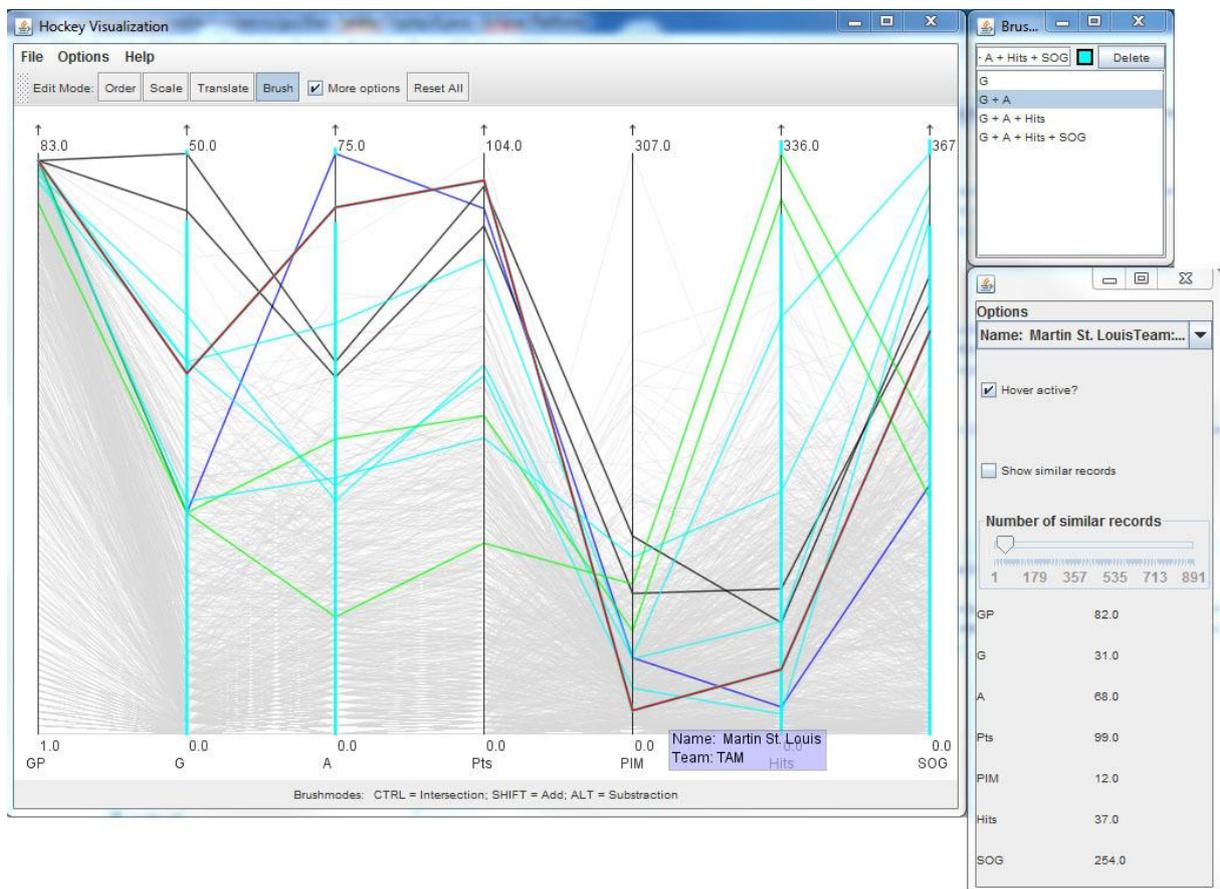


Abb. 9 Fairste Spieler unter verschiedenen Bedingungen

3.4 Wichtigkeit von Verlängerungen (EBEL)

Welche Mannschaft ist die Beste von denen, die nie in der Verlängerung gewonnen haben, bzw. welche ist die am schlechtesten Platzierte von denen, die nie in der Verlängerung verloren haben?

- **Nie in der Verlängerung gewonnen:** Es gibt nur zwei Teams die nie in der Verlängerung gewonnen haben, Graz und Jesenice. Dabei ist Graz die besser platzierte (Jesenice schloss als 10. und somit letzter ab, Graz als 6.). Wobei zu beachten ist, dass Graz in der Penaltywertung wesentlich besser abgeschnitten hat und insgesamt seltener in die Overtime musste (sieben Mal, während Jesenice 11 mal in der Overtime war). Siehe dazu auch Abb. 10 Wichtigkeit der Overtime)
- **Nie verloren aber am schlechtesten platziert:** Nur ein Team hat nie in der Overtime verloren (Vienna Capitals), dafür haben sie bei sieben Penaltyschießen nur zwei gewonnen (und vier Mal in regulärer Overtime gewonnen).

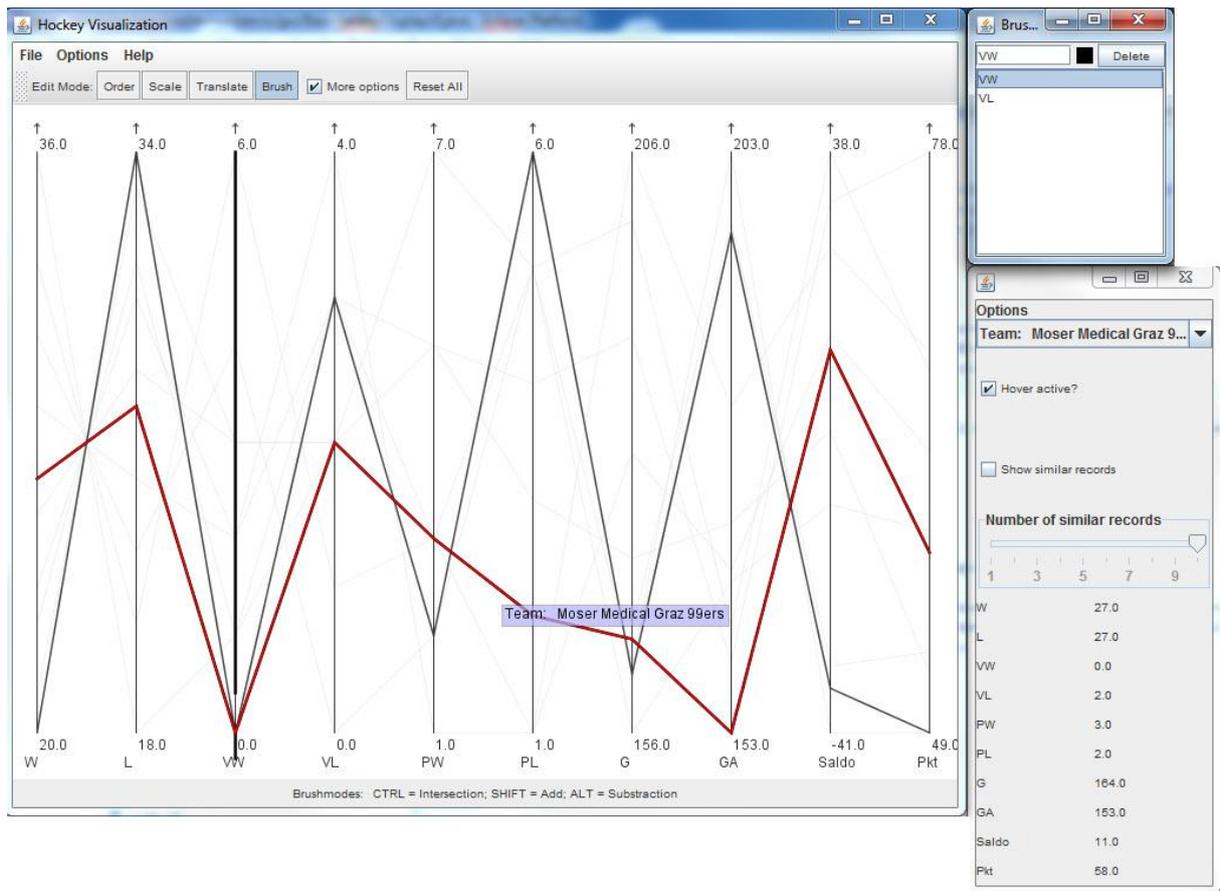


Abb. 10 Wichtigkeit der Overtime

4 Kurz-Bedienungsanleitung

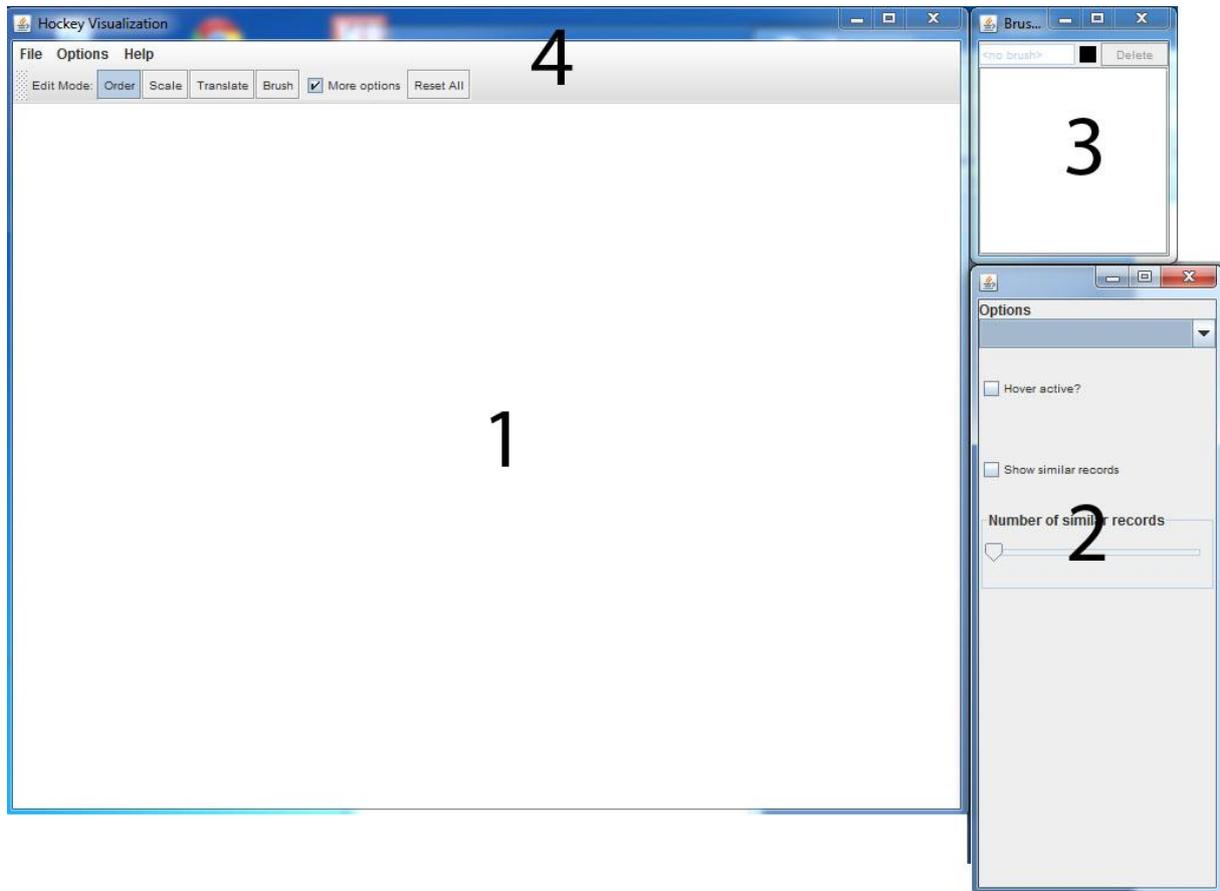


Abb. 11 Die 4 Hauptteile der GUI

Im folgenden werden die 4 Hauptteile der GUI (siehe Abb. 11 Die 4 Hauptteile der GUI) kurz beschrieben.

1.) Das Hauptfenster:

In diesem Fenster wird die eigentliche grafische Darstellung des parallelen Koordinatensystems gezeichnet. Hier können auch alle Mausektionen getätigt werden (Brushes ziehen, Achsen ändern, etc.)

2.) Das Options Fenster:

Dieses muss nach dem Start von Parvis durch einen Klick auf "More options" in der Menüleiste (4) aktiviert werden. Hier kann eingestellt werden, ob der "Hover" Effekt aktiv ist, also ob ein Datensatz hervorgehoben wird, wenn man mit der Maus darüber fährt. Ist dieser aktiviert, so werden unterhalb auch alle Daten im Detail eingeblendet.

Weiters können hier die ähnlichen Datensätze aktiviert werden und über einen Schieberegler eingestellt werden, wie viele ähnliche Datensätze angezeigt werden sollen. Um ähnliche Datensätze anzuzeigen muss im oberen Drop-Down Menü ein Datensatz vorausgewählt werden.

3.) Das Brush Fenster:

Hier werden alle getätigten Brushes automatisch abgespeichert und können so im Nachhinein wieder ausgewählt werden. Die Add, Subtract und Intersect Aktionen können auch direkt in diesem Fenster durchgeführt werden (so können alte und neue Brushes kombiniert werden). Über den Delete Button können gespeicherte Brushes wieder gelöscht werden.

4.) Die Menüzeile:

Im File Menü kann man neue Datensätze öffnen. Options bietet die Möglichkeit, die zwei verschiedenen Ähnlichkeitsberechnungen durchführen zu lassen. Durch Auswahl der Buttons Order, Scale, Translate und Brush wird zwischen den verschiedenen Bearbeitungsmodi umgeschaltet. Mit der Checkbox "More Options" wird das Optionenfenster aktiviert bzw. deaktiviert.

5 Zusammenfassung

Das interaktive parallele Koordinatensystem ist eine recht interessante Möglichkeit, Eishockeystatistiken zu betrachten. Es eignet sich vor allem gut dazu, Informationen herauszusuchen, die man in Tabellen lange suchen würde. Besonders gut eignen sich hier die beiden Werkzeuge "Brush" und "Hover". Durch das anzeigen des derzeitigen Wertes während des Ziehens eines Brushes, kann man auch erstaunlich genau arbeiten.

Mögliche Verbesserungen sind noch bessere Methoden zur Berechnung der wenigsten Überschneidungen, die erstens genauer und zweitens schneller arbeiten. Derzeit muss die Anzahl der Achsen bei entsprechend großer Anzahl an Datensätzen auf unter 10 reduziert werden, um Ergebnisse in halbwegs annehmbarer Zeit zu bekommen.

Der große Vorteil dieser Darstellungsart ist, dass der Benutzer relativ einfach und schnell einen Überblick über die verschiedensten Statistiken bekommt und vor allem Wertebereiche sehr einfach eingrenzen kann.

Ein Nachteil dieser Darstellungsart ist aber sicher, dass die Übersichtlichkeit bei vielen Datensätzen schnell verloren geht. Wenn man keine Einschränkungen durch Brushes oder Ähnlichkeitsberechnungen vornimmt, können zum Beispiel bei den über 800 Spielern der NHL wenig Informationen aus der Darstellung abgelesen werden.