Also publ. in: IEEE Transactions on Visualization and Computer Graphics 12 (2006), 6, pp. 1486-1499

Summarizing Dynamic Bipolar Conflict Structures*

Ulrik Brandes† Daniel Fleischer‡ Jürgen Lerner§

Department of Computer & Information Science
University of Konstanz, Germany

ABSTRACT

We present a method for visual summary of bilateral conflict structures embodied in event data. Such data consists of actors linked by time-stamped events, and may be extracted from various sources such as news reports and dossiers. When analyzing political events, it is of particular importance to be able to recognize conflicts and actors involved in them. By projecting actors into a conflict space, we are able to highlight the main opponents in a series of tens of thousands of events, and provide a graphic overview of the conflict structure. Moreover, our method allows for smooth animation of the dynamics of a conflict.

Keywords: information visualization, text mining, event analysis, time-dependent visualization

A preliminary version of this paper has been published in [5].

1 INTRODUCTION

Event data are among the most widely used indicators in quantitative international relations research. While such data can be extracted efficiently from news reports and dossiers, it has to be presented to the analyst in such a way that the contained information can be captured without too much effort. Even when aggregating actors (like different people representing the same organization) and focusing on specific global regions (like the Balkans or the Persian Gulf) and periods of time (like 10 or 20 years) there are typically several hundreds of actors engaged in tens of thousands of events.

Of special interest in social and political event analysis is the identification of conflicts and the division of parties engaged into opposing sides. While the ultimate goal is conflict prediction, a first step consists in the detection and analysis of historical or ongoing conflicts. This is particularly difficult if the parties involved are not, or only partially, known to the analyst.

Previous work. There is much work on analyzing and summarizing (streams of) news reports. Goals include the automatic generation of warnings about political crisises, topic detection and tracking, and the identification of frequent patterns in the development of conflicts [20, 2, 1, 11, 3]. However, most of the proposed methods are restricted to the computation of certain statistics, whereas visual support to the analyst is either absent or bounded to plotting the timeseries of these statistics. Exceptions include the following two: Best et al. [3] used a geographic information system to generate maps that highlight countries having the most co-occurrences with certain keywords. Wong et al. [19] proposed a method to generate animated scatterplots from data streams like, e. g., sequences of news articles. (Scatterplots are widely used in statistical graphics, see, e. g., [7, 8].) However, the scatterplots in [19] show similarities between documents and not hostile relationships between political actors as will be done in this paper. Moreover, most previous work (except, e. g., [18]), is limited to analyze individual dyads of actors separately. In contrast, we will exploit the macroscopic network structure of bilateral events.

Contributions. We present a method that, given a list of events, constructs a sequence of networks which in turn is converted into an animated scatterplot. The resulting video summarizes graphically the dynamics of major conflicts over a potentially long period of time. From this video, an analyst can recognize or discover the major actors engaged in conflict during certain periods of time. The observer is also enabled to detect time-points where the conflict structure changes significantly. Since our animation is smooth by design, it can be recognized easily which actors enter or leave a conflict during transitions. After recognizing important (groups of) actors and time points, the analyst can explore the conflict network surrounding these actors, thus obtaining additional structural information about how they are linked together. Furthermore, the lists of events associated to selected dyads of actors can be printed and news reports associated to crucial events can be directly shown.

The gain from the proposed method is threefold. Firstly, efficiency of data analysis is augmented by enabling visual mining of huge sets of event data supported by a sophisticated preprocessing. Secondly, the presented partial networks give additional information about indirect ties (e. g. enemies of enemies) and about density, complexity, and structure of the actors' environments. Last but not least, the analyst can present his/her insights much more conveniently to others by showing (parts of) the video or printed still images than by large statistical tables. Thus, usage scenarios for our method include the exploration of unfamiliar political situations, identification of crucial turning points, and briefing of decision makers.

An outstanding additional property of our method is that it allows for a rigorous analysis of robustness to noise in the input data. High stability gives the certainty that the derived representation is indeed meaningful and not caused by errors. On the other hand, significant drops in the stability indicator can draw attention to rapid changes in the conflict structure as well as to inconsistencies in the input data.

Outline of this paper. In Sect. 2 we provide background information on the type of data that is being analyzed. Our method for visualizing the conflict structure embodied in a set of events is introduced in Sect. 3 and extended to smooth animation of event series in Sect. 4. Section 5 describes how an analyst can explore the local conflict networks of prominent actors and trace back the original events. In Sect. 6 we analyze the robustness of our visualization technique. The utility of our method is illustrated on event data from the Balkans and the Persian Gulf in Sect. 7. We conclude with a discussion of open problems and future work.

*Research partially supported by DFG under grant Br 2158/1-2

†e-mail:Ulrik.Brandes@uni-konstanz.de

‡e-mail:fleische@inf.uni-konstanz.de

§e-mail:lerner@inf.uni-konstanz.de

2 Event Data

Our method is applicable to event data given as a series of pairwise interactions. Although it is independent of the data format, we will focus on a particular coding scheme to make the exposition more concrete. The Kansas Event Data System (KEDS) [15] is a software tool that automatically extracts events from text. In Sect. 7 we will use KEDS data for the Persian Gulf and Balkans regions. Formally, an event series is a sequence $a_1,\ldots,a_k$ of tuples $a_i = (t_i, s_i, o_i, c_i)$, where

- t_i is the *time-stamp* (date),
- s_i is the *subject* (source),
- o_i is the *object* (target), and
- c_i is the *code* (type)

of event a_i. Dates are given by the day and actors may be aggregated. We say that actors s_i and o_i are *involved* in event a_i. Events are classified using to the World Event/Interaction Survey (WEIS) codes [14]. Each event is assigned Goldstein weights [9] $-10 \le \omega(a_i) = \omega(c_i) \le 10$, which are psychometrically determined scores depending only on the type of event. A positive (negative) weight indicates the degree of *cooperation* (*hostility*) of the corresponding type of event.

The following excerpt indicates the coding of actors in the Balkans data.

```
NATO_OFFICIAL  [NAT]
NATO-LED_STABILIZATION_FORCE_IN_BOSNIA  [NAT]
SERBS_IN_BOSNIA  [BOSSER]
RATKO_MLADIC  [BOSSER]
MILOSEVIC  [SERGOV 890101-971230]
           [FRYGOV 971231-001005]
           [SERSM >001006]
```

Several tokens in the news may be interpreted as referring to the same aggregated actor. In the above excerpt, NATO (`NAT`) is represented by (among others) potentially unnamed officials and SFOR[1]. Similarly, the actor `BOSSER` is represented by (among others) the general term "Serbs in Bosnia", as well by specific persons like Mladić[2]. On the other hand, the same token may represent different actors at different times. For instance, Slobodan Milošević represents the Serbian government until December 1997, the government of the Federal Republic of Yugoslavia until October 2000, and after being replaced by opposition-list leader Vojislav Koštunica only himself.

Given an actor coding, textual statements are parsed into events like the following, dated July 1995.

```
950710 NAT  BOSSER  173  (SPECIF THREAT)
                                POSSIBLE AIR STRIKES
950710 FRN  MOS     054  (ASSURE)
                          CONDEMNED ATTACK ON MOSLEM
950710 UNO  BOS     102  (URGE)    MUST
950710 UNO  BOSSER  160  (WARN)    WARNED
950711 NAT  BOSSER  223  (MIL ENGAGEMENT) WARPLANES STRUCK
```

The first event is an action initiated by the NATO (active) and directed at the Serbs in Bosnia (passive). In addition to the event code (173), a textual description of the type of event is given in parentheses. The rest of the line is the stemmed form of the text fragment that has been turned by the KEDS parser to the corresponding event (line breaks are added for space constraints). Often, this text gives valuable additional information, e. g. for the first event, information about the nature of NATO's threat.

Examples for Goldstein weights associated with event types are the following.

```
072  EXTEND MIL AID     8.3
054  ASSURE             2.8
160  WARN              -3.0
223  MIL ENGAGEMENT   -10.0
```

Apparently, extending military aid is a highly cooperative action, whereas warnings are mildly and military engagement is extremely hostile. To analyze conflict, we will only make use of negatively weighted events, i. e. hostile actions.

To detect emergent patterns and utilize indirect relations, we transform the data into a network. Any set $\{a_1,\ldots,a_k\}$ of events gives rise to a directed interaction graph $G = (V,E)$ in the following way. Every actor involved in any event constitutes a vertex, i. e. $V = \bigcup_{i=1}^{k} \{s_i, o_i\}$. There is a directed edge $e = (u,v)$ if there is an event with source u and target v, and we assign a weight $\omega(e)$ that is minus the sum of all negative weights on events initiated by u and directed to v. Figure 1 shows an example of a conflict graph drawn by standard force-directed layout techniques [12]. The complexity of Fig. 1 already indicates the insufficiency of general-purpose graph-drawing techniques and the need for other analysis and visualization methods that are more appropriate for this application. In Sect. 3 we develop a new method of network analysis [4] that is able to extract the dominant conflict structure, filters out minor actors, and produces a less complex image that is easy to understand.

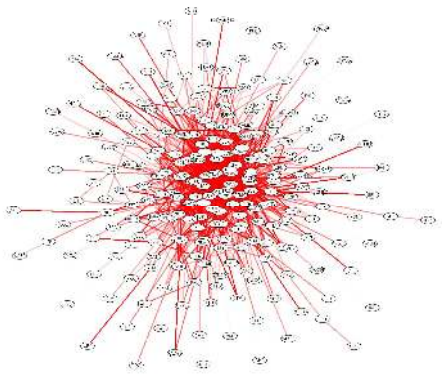

Figure 1: Hostile interaction in the Balkans from 1991 until 1997 (edge directions are not shown). Saturation of the edges is proportional to cumulative hostility weights.

It is unlikely that a focused data set yields an interaction graph with more than one significant non-trivial connected component. However, since components can be analyzed separately, we may safely assume that our interaction graphs are connected, anyway.

3 Visualizing Bilateral Conflicts

In this section we focus on extracting the structure of conflicts from static event data, i. e. we ignore time-stamps and consider the data to be given as a set (rather than a sequence). We will use the corresponding interaction graph to determine the main opposition of actors, where the assumption is that conflicts are predominantly bilateral. The static methods that are developed in this section will be augmented to include dynamics in Sect. 4.

During the computation of the conflict network's group structure we will ignore edge directions. The rationale behind this is that if there is a strong negative (hostile) edge between actors u and v, then

[1] The *Stabilization Force* (SFOR) was a NATO-led multinational force in Bosnia and Herzegovina.

[2] Ratko Mladić was the leader of the Bosnian Serb Army.

u and v should be in different groups—independent of whether the edge is directed from u to v or vice versa. However, edge directions will be taken into account when determining whether an actor is more active or more passive (see Sect. 3.3) and highly asymmetric edges will also be shown as such (see Sect. 5).

A straightforward attempt to determine the two opponent groups of a bilateral conflict is to try to divide the actor set V into two disjoint subsets U and W, such that for each edge $e = (u,w)$, or $e = (w,u)$ we have $u \in U$ and $w \in W$, i. e. conflicts are only between and not within the two groups. See Fig. 2 for an example partition of a small subset of actors in the Balkans and selected events among them.

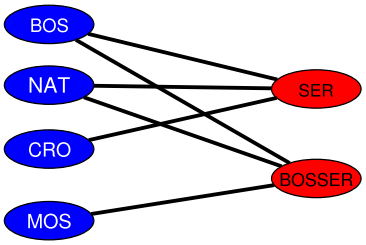

Figure 2: The subgraph of hostile interaction induced by selected Balkan events constitutes a bipartite conflict structure.

The existence of a perfect partition into such sets corresponds to the graph being bipartite and is thus easy to test for. However, larger data sets will seldomly result in purely bipartite structures. In fact, our experiments provide evidence that the interaction graphs are not bipartite for almost all reasonable selections of actors and periods of time. Thus, in order to make the idea applicable to empirical data, the concept of a bipartition must be relaxed.

A possible relaxation is to determine a partition $V = U \cup W$ such that the sum of edge weights between the two sets is maximized (in a perfect bipartition, this sum is over all edges). However, this is the well-known MAXCUT problem, which is highly inappropriate for our purposes: It is NP-hard, not robust to noise, requires actors to be purely in one group or the other, and reveals no prominence of actors. We make use of a different relaxation that poses no algorithmic problems, is stable, can handle actors that are members of both groups, and filters out unimportant actors on the fly.

3.1 Structural Projections

To arrive at a relaxed bipartition, we employ the recently introduced concept of structural projections (and the closely related structural similarities) [6], which have a sound theoretical basis and lead to an efficient algorithm that is easy to implement. We first give an intuition of structural projections, and then introduce them formally.

Instead of mapping actors to one class or the other, structural projections yield real-valued degrees of membership in classes. For a relaxed bipartition, actors are projected into a two-dimensional conflict space such that actors mapped mostly into one dimension have major conflicts with actors mapped mostly into the other dimension, but only minor conflicts with actors mapped into their own dimension.

An example of such a real-valued projection is shown in Fig. 3. The figure shows the conflict structure of all actors involved in the Balkans. Note that the degree of membership assigned to actors varies. E. g., Bosnia (BOS) is a much stronger member of the blue group than, e. g., the Moslem ethnic group (MOS). On the other hand UNO, though closer to the blue group, does not fit exactly into the bipartite structure, because conflicts with other blue actors (e. g., with Bosnia) are reported. Many of the unimportant actors close to the origin are filtered out because their level of hostility is not sufficient to place them prominently in one group or the other. Thus, our method not only determines a relaxed bipartition, but also indicates which actors are most responsible for the division.

The method sketched above is a specific use of a more general framework. We next introduce its essentials, and refer to [6] for further details.

A weighted graph $G = (V,E)$ is represented by its symmetric adjacency matrix $A = (a_{uv})_{u,v \in V}$ with rows and columns indexed by V and entries $a_{uv} = \omega(u,v) + \omega(v,u)$ corresponding to the weight of the two directed edges between the two endpoints (if an edge is not present, the weight is simply equal to zero). The following definition specifies how vertices (i. e. actors) are projected continuously to k classes, and thus generalizes k-partitions.

Definition 1 *Let G be a graph with n vertices and adjacency matrix A. A* projection *is a real $k \times n$ matrix P that has orthonormal rows. The entry P_{iv} is the* degree of membership *of actor v in class i. The* quotient *of G modulo P is the graph that has the k different classes as vertex set and whose adjacency matrix is $B = PAP^{\mathrm{T}}$.*

Quotients model the relation between classes induced by P and G. Two classes C_1 and C_2 are connected by an edge whose weight is the average weight of the edges connecting actors in C_1 to actors in C_2 (weighted by their respective degree of membership). In Fig. 2, e. g., the quotient of the projection to the blue and red class is a single edge connecting the two.

The next condition ensures that a projection is compatible with the graph structure.

Definition 2 *A projection P is called* structural *for G if $PA = BP$.*

The above condition requires that equivalent actors must be equally connected to equivalent actors. In Fig. 2, e. g., all blue actors have ties to some (but not necessarily the same) red actors .

The following theorem is a characterization of all structural projections for a given quotient and hence essential for our purposes. It shows how a projection can be chosen such that it yields a prespecified quotient.

Theorem 3 ([6]) *Let G and R be two weighted graphs with adjacency matrices A and B. A projection P is structural for G with quotient R, if and only if the row-space of P is generated by eigenvectors of A associated with all eigenvalues of B.*

We will use this theorem in the next section to determine a structural projection into two-dimensional conflict space such that the resulting quotients are "as bipartite as possible."

3.2 Projecting into Conflict Space

Consider the parameterized graph in Fig. 4. It consists of two vertices representing the opposition of two groups, an edge with large weight w representing the hostility of events involving actors in opposite groups, and two edges with small weight c representing the hostility of events involving actors within a group.

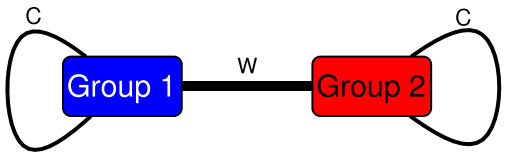

Figure 4: Quotient R representing the conflict space. Weights w and c are average weights of edges between and within the two groups.

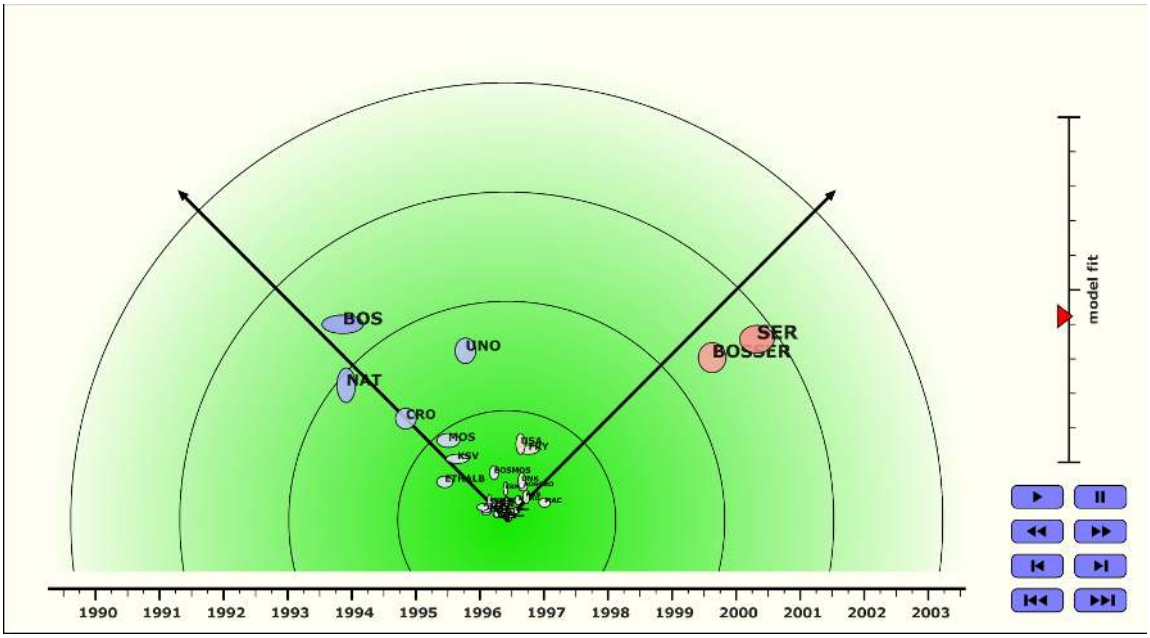

Figure 3: Structural projection of Balkan conflict 1989–2003. Dominant actors include those of Fig. 2. Actors are members of the first (second) group to the extent that they are mapped in direction of the left (right) coordinate axis. Angle (left vs. right) and color (blue vs. red) encodes the ratio between the two group membership values. Importance is proportional to the distance from the origin. The aspect ratio (shape) of an actor encodes the ratio between activeness (height) and passiveness (width).

The single-edge quotient of a bipartite graph is augmented by loops of weight c, because the interaction graph will almost never be bipartite at any given point in time. For this quotient, we need to find a (continuous) projection of actors onto these two classes, such that w is as large as possible and c as small as possible.

The eigenvalues of the adjacency matrix B of the quotient shown in Fig. 4 are

$$\lambda = w + c \qquad \text{and} \qquad \mu = -w + c \ .$$

From a different perspective, parameters w and c are given by the two eigenvalues λ and μ of B as

$$w = \frac{\lambda - \mu}{2} \qquad \text{and} \qquad c = \frac{\lambda + \mu}{2} \ .$$

By Theorem 3, a projection P onto the graph R is found by eigenvectors of A associated with λ and μ. Since our goal is to maximize w, we choose λ as large and μ as small as possible, i. e. we take eigenvectors of A associated with the largest and smallest eigenvalue $\lambda_{\max}$ and $\lambda_{\min}$. We thus have the following result.

Theorem 4 *The structural projection with quotient R from Fig. 4 that maximizes w is the orthogonal projection onto the two eigenvectors of A associated with the largest and smallest eigenvalue.*

The above derivation also shows that $c = 0$ if and only if $\lambda_{\min} = -\lambda_{\max}$, which is well-known to hold if and only if the graph is bipartite. Although this is almost never the case in empirical data, our experiments provide evidence that often c is very small compared to w. To assess the degree to which the data matrix A is bipartite, i. e., to which degree does the bilateral model fit the data, we introduce the following index.

Definition 5 *Let $\lambda_{\max}$ and $\lambda_{\min}$ be the largest and smallest eigenvalue of the adjacency matrix of a graph G. The* bipartiteness *of G (or* model fit *of the projection) is defined as*

$$\beta(G) = \left| \frac{\lambda_{\min}}{\lambda_{\max}} \right| \ .$$

Index $\beta(G)$ is between zero and one. It is one if and only if the graph is bipartite (i. e. if the model fits perfectly) and it is zero if and only if c equals w, i. e. if there are as many edges within the groups as there are in-between.

3.3 Graphing Conflict Space

Using the projection specified in Theorem 4, we obtain a representation of the interaction graph in a two-dimensional conflict space, i. e. a scatterplot of the vertices with coordinates given by the eigenvectors of the adjacency matrix of the graph. So far, actors are drawn without the connecting edges to avoid clutter (compare Fig. 1).

The coordinates of the scatterplot are transformed to indicate the degree of membership to each of the two classes. To do so, first consider what happens when the ideal, single-edge, conflict graph (i. e., the graph R in Fig. 4 with $c = 0$) is projected onto itself. The normalized eigenvectors of a single edge are

$$x = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \end{pmatrix} \quad \text{and} \quad y = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ -1 \end{pmatrix} ,$$

so that

$$\frac{x+y}{\sqrt{2}} = \begin{pmatrix} 1 \\ 0 \end{pmatrix} \quad \text{and} \quad \frac{x-y}{\sqrt{2}} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} .$$

If x, y are the first and last eigenvectors of an arbitrary input graph, the lefthand side transformations thus yield real-valued degrees of membership in the two conflict groups. To obtain a horizontal opposition, we finally rotate the result by 45 degrees (compare e. g., Fig. 3). The rotated coordinate system prevents, in contrast to the more usual one (one axis vertical and one horizontal), misguided

interpretation of superiority of one group over the other. The proposed coordinate system differs from that of [5] which has two horizontal, antiparallel axes. In the new coordinate system actors that are logically between the two groups (i. e., those that have strong membership to both groups) are also visually between those groups (see, e. g., UNO in Fig. 15). This does not hold for the coordinate system in [5].

Because of an intricate relation between structural similarities and vertex centrality [6], the distance from the origin (i. e., the norm of the actor's column in P) is an indicator of the actor's importance. As is apparent in Fig. 3, our projections thus not only classify actors to one group or the other, but also distinguish between major and minor members of each group. We draw circles around the origin to facilitate the recognition of actors with the same level of importance.

The graphical attributes of our visualization are determined as follows. The actor's position in the two-dimensional drawing indicates its group membership and importance: Actors are members of the first (second) group to the extent that they are mapped in direction of the left (right) coordinate axis. Importance is proportional to the distance from the origin. Angle (left vs. right) and color (blue vs. red) encodes the ratio between the two group membership values. Importance is further emphasized by the saturation of the actor's color. Vertex shape and size are used to add information about activeness of actors. *Activeness* is defined as the net weight of the events in which an actor is involved as the subject initiating the event. Symmetrically, *passiveness* adds weights of events received. The ratio between activeness and passiveness determines the aspect ratio of a vertex, so that aggressive actors that initiate hostile interactions, but are not the subject of retaliation are high and narrow. The size of a vertex is the sum of the two and thus represents an actors *involvement* in a conflict structure. Finally, we indicate the fit of the bipartite model using a bipartiteness gauge on the righthand side of the frames.

We summarize the algorithm for scatterplots representing the conflict structure of a graph G of hostile interactions with adjacency matrix A:

1. Compute maximum and minimum eigenvalues $\lambda_{\max}$ and $\lambda_{\min}$ of A together with associated normalized eigenvectors $v_{\max}$ and $v_{\min}$.

2. The projection is given by the $2 \times n$ matrix P with $x = (v_{\max} + v_{\min})/\sqrt{2}$ in the first and $y = (v_{\max} - v_{\min})/\sqrt{2}$ in the second row. Rotate coordinates by 45 degrees.

3. The bipartiteness is given by $\beta = -\lambda_{\min}/\lambda_{\max}$.

4. Draw each actor v as an ellipse with coordinates and color values proportional to the two values in the v'th column of P, height proportional to activeness and width proportional to passiveness.

Any eigenvector algorithm for real symmetric matrices can be used in Step 1 (see, e. g., [10]), and there are many readily available software packages.

Figure 3 shows the projection into conflict space for events of the Balkan Conflict from 1989 until 2003. The circles around the origin facilitate to recognize that the most important actor during the whole period of time is Serbia (SER), closely followed by the Serbs in Bosnia (BOSSER) and Bosnia (BOS). The bipartiteness of this projection is rather low (only around 0.42), indicating many conflicts within the two groups. Despite of the low bipartiteness, our method still yields two reasonable opponent groups: Serbia and the Serbs in Bosnia opposed to Bosnia and Croatia (CRO). The NATO (NAT) is opposed to SER and BOSSER, due to the massive air strikes in 1994 and 1995. Since NATO initiated more events than it receives, it shows as a high and narrow actor.

Figure 5 shows the projection for the conflicts in the Persian Gulf from 1979 until 1999. The two major opponents for the whole period of time are clearly the Iran and Iraq, due to their war from 1980 to 1988. The bipartiteness is, although much higher than in Fig. 3, considerably distant from one. The reason for this is that the USA have strong negative ties to both Iran and Iraq, thus these three actors form a hostile triangle. The fact that the aggregated weight of hostile events between the USA and Iraq (mostly following Iraq's invasion of Kuwait in 1990) is larger than the weight between the USA and Iran, is the cause that USA is mapped farer from Iraq and thus, necessarily, closer to Iran .

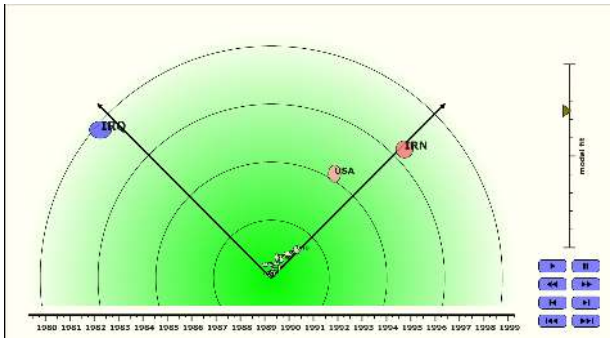

Figure 5: Gulf Conflict 1979–1999.

4 Animating Conflict Dynamics

The images generated as described in Sect. 3 already reveal the actors and conflicts that are dominant over the whole period of time. However, due to changing oppositions and alliances these images might not well represent the structure at specific time-points. Likewise, conflicts of short duration might be filtered out. To obtain a more detailed insight into the evolution of conflicts, we will introduce a technique for smooth animation of the above type of scatterplots for limited periods of time.

The event graph G is used to generate a sequence of graphs G_t, each of which represents the view on the set of events at the specific time t. A graph G_t yields one frame of the final video and this frame shows a detailed image of the situation at time t. How the events are viewed at a certain time-point is determined by a *scaling function* $\eta\colon \mathbb{R} \to \mathbb{R}_{\geq 0}$, which models how events move into the data when time increases and how they fade out. Examples of possible scaling functions are triangular shaped scaling functions with time radius r

$$\eta_r(t) = \begin{cases} (t+r)/r & \text{if } |t| \leq r \text{ and } t < 0 \\ -(t-r)/r & \text{if } |t| \leq r \text{ and } t \geq 0 \\ 0 & \text{if } |t| > r \ . \end{cases} \qquad (1)$$

The function η_r does not consider events with a time-stamp more then r away from the current time-point. Events move into G_t linearly until t is larger than their time-stamp. Then, they fade out linearly until they have zero weight.

For a fixed η and t, the graph $G_t = (V, E, \omega_t)$ is defined as follows. The actor set V and the edge (or event) set E are the same as for the input graph G. The weight $\omega_t(e)$ of an event e at time t is defined to be $\omega_t(e) = \omega(e) \cdot \eta(t_e - t)$, that is, the weight of e at time t is its absolute weight $\omega(e)$ times a scaling factor which is dependent on the difference between the time-stamp t_e of the event and the current time t. The graph G_t may be reduced by removing events with zero weight, as well as isolated actors, since these do not influence the analysis and would be invisible in the final video.

Given a graph G, representing a list of events, the movie is generated by the following steps.

1. Select a sequence of time-points $t_1 < \cdots < t_N$ in a given time interval.

2. For each i from 1 to N

   Compute the visualization of the graph G_{t_i}.

3. The images for all time-points yield the frames of the video.

In order to maintain the overall appearance of the frames one further detail has to be taken into consideration. If v is an eigenvector of A associated to eigenvalue λ, then so is $-v$. Thus the eigensolver algorithm could return either v or $-v$ as a solution to the eigenvalue problem. To prevent that this assignment switches from one frame to another (which would result in interchanging the axes of the coordinate system from one frame to another) we have to ensure that the eigenvectors we use point in a well-defined direction.

The canonical direction for the eigenvector $v_{\max}$ associated to the largest eigenvalue is simply the direction in which each entry of $v_{\max}$ is positive. (It is standard knowledge in algebraic graph theory that all entries of this eigenvector have the same sign.) We define the canonical direction for the eigenvector $v_{\min}$ for time-step t recursively by the direction of this eigenvector for time-step $t-1$. The direction of $v_{\min}$ is chosen such that the angle between $v_{\min}$ at time t and $v_{\min}$ at time $t-1$ is smaller than 90 degrees. Thus, only the direction of $v_{\min}$ for the very first time-step is arbitrary. This translates to the fact that there is no absolute meaning attached to the two opponent groups. A second computation of the movie could reverse the red and the blue groups, but then it has to reverse the assignment for all actors and at all time points, which results in the same opponents.

5 Exploring the Conflict Network

In summary, the visualization method described in Sects. 3 and 4 settles the important problem of identifying and displaying the most important actors out of a large set of event data. Moreover, it shows how these actors are grouped together, and how the group structure changes over time. This information serves as the starting-point to analyze visually and interactively the given event data. In this section we describe how to augment our system by providing interactive facilities to explore the conflict network around prominent actors. We give first an overview about the interaction possibilities and explain them in more detail in the remainder of this section.

The first (rather obvious) interaction possibility is the control of the animation and selection of time points by the buttons in the lower right corner or by clicking on a specific point in the timeline. The other types of interaction enable exploration of the network structure and tracing of events (also see Fig. 6): Focusing on a small number of important actors, the analyst can explore the edge structure (conflict structure) surrounding these actors. The events that are associated to selected edges can be printed in textual form. Finally, the original news reports are obtained by selecting crucial (like military engagements) or unexpected (depending on the analyst's previous knowledge) events. Thus, the analyst can trace back the (reports of the) most serious political, military, or other activities, involving the chosen actors.

In our examples (see Sect. 7), printing of the events is performed by a separate program and the presentation of news reports is not included in our prototypical implementation. However, it should be obvious that the realization of these functionalities is straightforward.

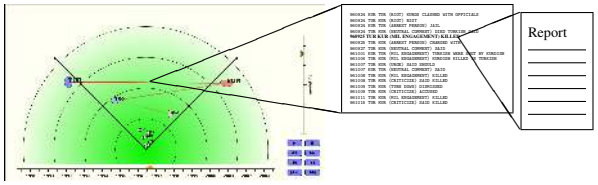

Figure 6: Schematic illustration of exploration possibilities. Edges of selected actors can be drawn. The event list of the selected edge (TUR,KUR) is printed. The news report associated to a specified event is presented.

Displaying selected network-structure. A scatterplot obtained by the algorithm in Sect. 3 reveals the key actors and the oppositions they form. However, it does not reveal the exact interaction structure. For instance, in Fig. 7, the analyst can clearly distinguish the prominent members of two opposing groups. However, from this picture it does not become clear whether, e. g., Turkey (TUR) has hostile interactions with the Kurds (KUR), the USA, or with both.

Our system can show the edges incident to a selected subset of actors. By clicking on an actor, all edges in which this actor is either the source or the target are displayed with saturation proportional to the edge weight. For instance, selecting the prominent actors in Fig. 8 shows their local network structure. It can be seen that not all edges between blue and red actors are present and not all edges have the same weight.

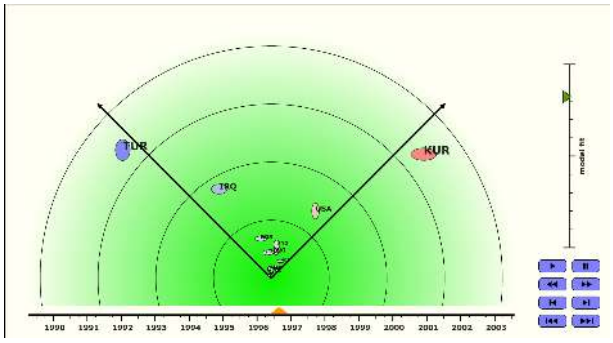

Figure 7: Balkans in 1996. Two opposing groups become apparent - but which actor is in conflict with whom?

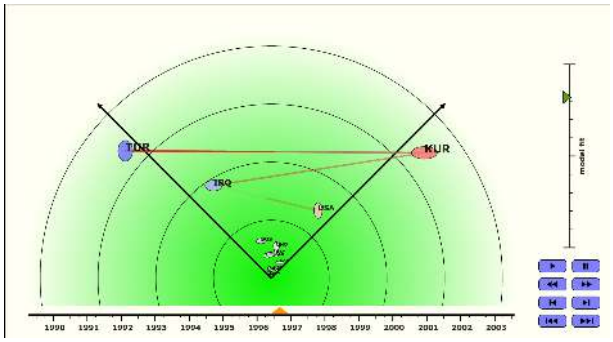

Figure 8: Same situation as in Fig. 7 with edges of selected actors shown. (Only edges with weight above 100 are included.)

The possibility to show and hide selected edges interactively is much preferable to simply drawing all (strong) edges. For instance,

showing all edges whose weight is above 1000 in the scatterplot for the complete Balkan conflict (Fig. 9), yields a rather confusing image. By giving the analyst the possibility to select actors whose edges are to be shown, our system can present a less complex but complete visualization of a part of the conflict network. See Fig. 10 where only edges incident to BOSSER are shown.

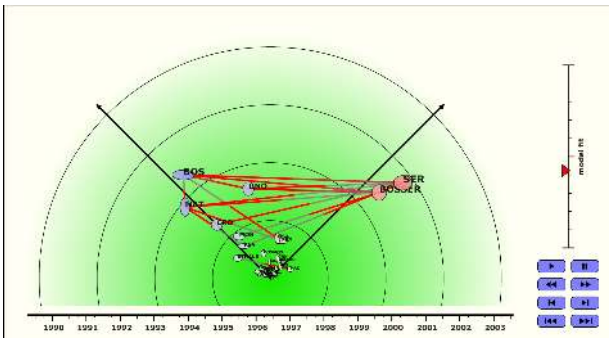

Figure 9: Balkans conflict from 1989–2003 including all edges with weight above 1000. The complexity of the edge-structure rather hides than reveals information.

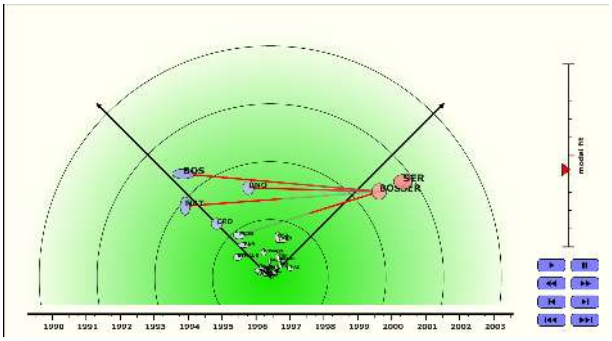

Figure 10: Same picture as Fig. 9 where only edges targeted to or from the Serbs in Bosnia (BOSSER) are shown. BOSSER's main opponents become clearly visible. Edge coloring indicates the main direction.

Asymmetry of edges is shown to the analyst by coloring the end incident to the main initiator (source) in red and the end incident to the main receiver (target) in grey. In Fig. 10, e. g., it can be observed that BOSSER initiated many hostilities targeted to the Moslem ethnic group (MOS), receives a lot from NATO (NAT), and is mostly on equal terms with Bosniaks (BOS) and the UNO.

Showing crucial events. The exploration of the local network around selected actors might draw the attention to strong, significant, or unexpected edges. To go deeper into the meaning and the origin of these edges, the analyst is enabled to print the hostile events and the news reports that generated it.

By selecting a pair of actors (an edge) and a time-frame the analyst can print all (or if too many, the strongest) hostile events involving those two actors. Figure 11 shows hostile events between Turkey and the Kurds (compare Fig. 8) from September 15'th until October 15'th 1996. (Note that individual events might be coding-errors. We will treat the issue of errors in Sect. 6.)

Finally, by selecting important or unexpected events from such lists, the analyst can obtain those news reports that contain the original text associated to the generated events.

The advantage of showing the events in their coded form before presenting the actual reports is that the analyst can get a rough

```
960924 KUR TUR (RIOT) KURDS CLASHED WITH OFFICIALS
960924 TUR KUR (RIOT) RIOT
960924 KUR TUR (ARREST PERSON) JAIL
960924 TUR KUR (NEUTRAL COMMENT) DIED TURKISH SAID
960925 TUR KUR (MIL ENGAGEMENT) KILLED
960925 TUR KUR (ARREST PERSON) CHARGED WITH
960927 TUR KUR (NEUTRAL COMMENT) SAID
961001 KUR TUR (MIL ENGAGEMENT) TURKISH WERE SHOT BY KURDISH
961006 TUR KUR (MIL ENGAGEMENT) KURDISH KILLED IN TURKISH
961007 TUR KUR (URGE) SAID SHOULD
961007 KUR TUR (NEUTRAL COMMENT) SAID
961008 TUR KUR (MIL ENGAGEMENT) KILLED
961008 TUR KUR (CRITICIZE) SAID KILLED
961009 TUR KUR (TURN DOWN) DISMISSED
961009 TUR KUR (CRITICIZE) ACCUSED
961011 TUR KUR (MIL ENGAGEMENT) KILLED
961015 TUR KUR (CRITICIZE) SAID KILLED
```

Figure 11: Hostile events between TUR and KUR from 09/15/96 to 10/15/96.

overview of the succession of events before starting the (time consuming) task of reading the news articles. For instance the above 17 events can be read "in a minute", whereas even skipping through several reports written on different days will take much longer. In addition, by looking at the sequence of events, it might become clear that some events are of minor importance and considering the corresponding articles can be omitted or postponed.

6 Robustness on Noisy Data

In this section we tackle the issue of stability of our method. Stability, i. e., robustness to errors in the input data, is one of the most crucial properties of any method that analyzes and/or visualizes empirical data, because the vast majority of application data does contain errors.

Reasons for errors are manifold—we will exemplary mention only two of them that are related to our data. First of all the news reports that have been used as input to the KEDS parser cannot be assumed to be objectively correct in all cases. Due to several reasons, some events might be more intensively reported than others, or journalists might not be aware of certain events. Secondly, the automatic coding of natural language text is certainly not perfect. According to the KEDS web-site, "machine-coded event data has a 15% to 20% coding error rate" [3]. Thus, the list of events that forms the input data of our system is only a perturbed coding of the "true" relations between actors.

In the ideal case, our method would turn the perturbed input data into an image that gives the same (or very similar) visual information to the analyst, as if the input data contained no errors. One of the most desirable features of our method is that we can provide a quantitative indicator that tells to what extent this ideal case is reached. It can be proved that, for a given value of this indicator, errors up to a certain magnitude are harmless. The benefits of the stability indicator are twofold. A high parameter gives the certainty that the representation of the data is indeed meaningful, because it is close to the representation of the unperturbed data. On the other hand, a low parameter draws the attention to unstable states of the conflict network which in turn might draw the attention to unstable political situations, as well as to inconsistencies in the input data. Furthermore, the magnitude of the stability parameter also provides bounds on how much does the visualization change from one frame to the next one, thus giving guarantees for the smoothness of the animation.

The next subsection provides a more formal model for the stability analysis and introduces the notation used in the following.

[3] http://www.ku.edu/~keds/data.dir/balk.html

6.1 Modeling Stability

We assume that the true conflict network is described by an $n \times n$ matrix A. Given A, our method would compute an image whose coordinates are the entries of the projection matrix P. However, A is not known. Instead, the input data is a perturbed matrix $A' = A + \mathscr{E}$, where $\mathscr{E}$ is the error matrix. Given A', our method computes an image whose coordinates are the entries of the projection matrix P'. The problem is to clarify how much does P' differ from P.

A first solution to this problem is that we define in Sect. 6.2 the *absolute stability* σ_0 (dependent only on A'), such that the difference between P and P' can be bounded in terms of σ_0 and the norm of $\mathscr{E}$.

However, σ_0 is not appropriate to compare the stability of networks with different number of actors and different average edge weight. As a more useful indicator of stability, we define in Sect. 6.3 the *relative stability* σ which builds on the assumption that errors for individual edges are uncorrelated and do not exceed a certain factor δ. (In the light of the stated 20% coding error in the KEDS data, this would translate to the assumption that each matrix entry is perturbed independently by up to 0.2 times the average magnitude of all entries.) The parameter σ is only dependent on A' and δ. Given the independence assumption on $\mathscr{E}$, the difference between P and P' can be bounded in terms of σ alone.

Since the true, unperturbed data is not known, the independence assumption can never be proved. However, assuming independence of errors is very common in statistical data analysis and is motivated from the belief that errors are not distributed in a "worst case" manner.

6.2 General Stability Results

We recall the definition of two matrix norms. The *2-norm* $\|M\|_2$ of a Matrix M is defined to be

$$\|M\|_2 = \max_{\|x\|_2=1} \|Mx\|_2 \ ,$$

where $\|x\|_2$ denotes the 2-norm (Euclidean length) of the vector x. The *Frobenius norm* $\|M\|_F$ of a Matrix M is defined to be

$$\|M\|_F = \sqrt{\sum_{i,j} M_{ij}^2} \ .$$

As specified in Theorem 4, our method projects to the eigenvectors associated to the largest eigenvalue $\lambda_{\max}$ and the smallest eigenvalue $\lambda_{\min}$. The absolute stability is the smallest distance of one of these two eigenvalues to any other eigenvalue of the adjacency matrix.

Definition 6 *Let A' be the adjacency matrix of the given conflict network and let $\lambda_1 \leq \cdots \leq \lambda_n$ be the ordered sequence of its eigenvalues (in particular, $\lambda_1 = \lambda_{\min}$ and $\lambda_n = \lambda_{\max}$). The* absolute stability σ_0 *of the projection P' to conflict space is defined to be*

$$\sigma_0 = \min\{|\lambda_1 - \lambda_2|, |\lambda_{n-1} - \lambda_n|\} \ .$$

The inverse of σ_0 measures to what extent errors in the data propagate to errors in the visualization of the data:

Theorem 7 *Let σ_0 be the absolute stability of the projection to conflict space, $\varepsilon = \|\mathscr{E}\|_2$ the norm of the error matrix, and let P and P' be the projections computed according to A and A', respectively. Set $\sigma_0' = \sigma_0 - \varepsilon$. Then, if $\varepsilon < \sigma_0'/2$, it is*

$$\|P - P'\|_F \leq \frac{2\varepsilon}{\sigma_0'} \ .$$

The proof of Theorem 7 follows in a straightforward manner from Theorem V.3.4 of [17].

The parameter σ_0 as defined in Def. 6 has the drawback that it is not possible to compare values of σ_0 across different networks, as σ_0 normally increases with increasing number of actors and with increasing average edge weight. In the next section we define the *relative stability* σ by a normalization of σ_0.

6.3 Uncorrelated Errors

The appropriate normalization of the stability indicator can be derived under the assumption that the errors of individual edges are uncorrelated. We give a model for such a random error process in the following definition.

Definition 8 *Let* $\mathrm{avg}(A')$ *denote the average of the entries of A', and let δ be a given parameter ($0 < \delta < 1$). We say that $\mathscr{E}$ satisfies the* independence assumption *with* error rate δ *if the entries of $\mathscr{E}$ are uniformly and independently drawn from the interval $[-\delta \cdot \mathrm{avg}(A'), \delta \cdot \mathrm{avg}(A')]$.*

The 2-norm of random matrices with bounded variance can be well estimated. The following theorem can be proved using results from [13].

Theorem 9 *Let $\mathscr{E}$ be a symmetric matrix whose entries are drawn independently from a probability distribution with variance* var *and mean zero. Then with high probability the 2-norm of $\mathscr{E}$ is bounded by*

$$\|\mathscr{E}\|_2 \leq 3\sqrt{\mathrm{var}\, n} \ .$$

The term "with high probability" means: with probability that tends to one as n increases.

Drawing numbers uniformly from an interval $[-\Delta, \Delta]$ has mean zero and variance $\Delta^2/3$. We conclude that if $\mathscr{E}$ satisfies the independence assumption with error rate δ then with high probability its 2-norm is bounded by

$$\|\mathscr{E}\|_2 \leq \delta \mathrm{avg}(A')\sqrt{3n} \ . \tag{2}$$

The above bound, together with Theorem 7 motivates the definition of the relative stability.

Definition 10 *Let A be the true adjacency matrix of a conflict network, $A' = A + \mathscr{E}$ the noisy input matrix, σ_0 the absolute stability, and δ a given error rate. Further, let $\varepsilon = \delta \mathrm{avg}(A')\sqrt{3n}$ be the upper bound for $\|\mathscr{E}\|_2$. The* relative stability σ *of the projection to conflict space is defined to be*

$$\sigma = \frac{\sigma_0 - \varepsilon}{2\varepsilon} \ ,$$

if $\sigma_0 > \varepsilon$. (Otherwise σ is undefined.) We call σ^{-1} the volatility *of the projection, if σ is defined.*

The key perturbation-theorem for the case of uncorrelated errors follows from Theorem 7 and Eq. (2) .

Theorem 11 *If $\mathscr{E}$ satisfies the independence assumption and if $\sigma >$ 1.0, then with high probability $\|P - P'\|_F \leq \sigma^{-1}$.*

Measuring $P - P'$ is done in the Frobenius norm since $\|P - P'\|_F$ is closely related to the difference between the unperturbed and perturbed drawing: If $\Delta(v) = \sqrt{(P_{1v} - P'_{1v})^2 + (P_{2v} - P'_{2v})^2}$ denotes the Euclidean distance between the positions of actor v in the two drawings (where the unit length is determined as being the length of the rows of P and P'), then

$$\|P - P'\|_F^2 = \sum_{v \in V} \Delta(v)^2$$

is the sum of the squared distances over all actors. Taking the squared distances has the advantage of counting a few large deviations more heavily than many small deviations. Indeed, moving one actor to a very distant place distorts the perceived information more seriously than moving all actors only slightly.

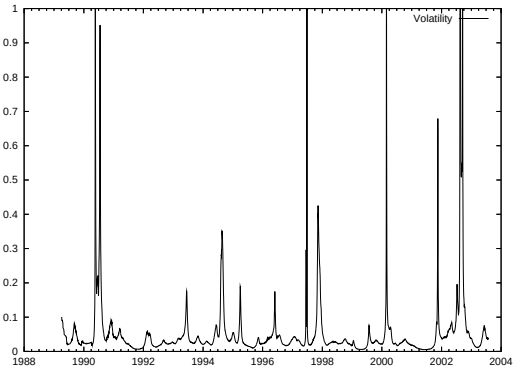

Figure 12: Volatility profile of the Balkan conflict.

As an example, Fig. 12 shows the volatility of the analysis of the Balkan conflict computed with $\delta = 0.2$. The volatility is well below 0.1 for most time steps. It reaches the threshold 1.0 (where Theorem 11 provides no bounds) only on a small number of days. These peaks in the volatility curve are all mirrored in the video by a sudden movement of the actors, resulting in a rapid change of the dominant conflict structure.

We conclude that the stated 20% coding errors in the KEDS data do only slightly affect the analysis for the majority of time-steps.

In the video we visualize a high volatility by changing the background-color from green to red. Coding volatility (similar to the model-fit) by a "volatility gauge" would not be appropriate due to the very short peaks that could easily be overseen. In contrast, a red background is easy to recognize and warns the analyst that the representation is currently very sensitive to errors and that possibly a major change in the conflict structure is going on.

7 Application Examples

We apply our method to visualize two selected data sets from the Kansas Event Data System (KEDS) [15] in a prototypical implementation. We generate the graphs G_t with the triangular shaped scaling function η_r which is defined in (1). For the two investigated data sets we used for the time radius r a period of 90 days. This simple but continuous scaling function already yields a smooth animation, since the coordinates depend continuously on the entries of the input matrix (lest σ^{-1} becomes greater than one).

In the video, the current time is shown by a triangle on the horizontal time-scale at the bottom. The current bipartiteness β_t is shown in a scale ranging from zero to one. This scale is green if β_t is close to one and red if it becomes low.

The animations[4] are realized in SVG (Scalable Vector Graphics, see W3C Recommendation[5]) format, thus they can be viewed on any web browser with an appropriate plug-in. The file for the Balkan region, e. g., covers about 5000 days using 1000 key frames to ensure smooth animation.

[4]The animations referred to in this paper are available from `http://www.inf.uni-konstanz.de/algo/research/conflict/`

[5]`http://www.w3.org/TR/SVG/`

7.1 Balkan

The KEDS Balkans data[6] contains 78,668 events for the major actors (including ethnic groups) involved in the conflicts in the former Yugoslavia. Coverage is April 1989 through July 2003. The varying degree of polarization can be inferred from the model fit indicator curve in Fig. 13. Although there is great variation in the magnitude of the model fit, it is often close to one and at all time points considerably distant from zero. Thus, the simplistic assumption of bipartite conflicts already fits the data sufficiently well. The following figures show selected time-points of the Balkans conflict.

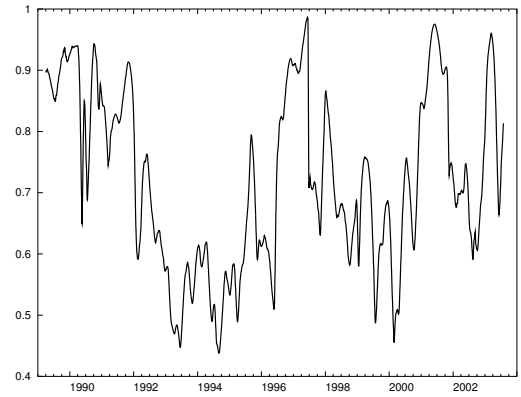

Figure 13: Bipartiteness (model fit) profile of the Balkan Conflict.

Important changes in the conflict structure took place in 1995 and 1996. Figure 15 shows the war in Bosnia, where Serbia and the Serbs in Bosnia (`BOSSER`) are opposed to Bosniaks and Croats. The UNO, which is trying to install peace in Bosnia, has conflicts of similar strength to all of them. This changes when troops of the Bosnian Serbs captured weapons from UN peace keepers and declined to return them (Fig. 16). After the Bosnian Serbs did not respond to an ultimatum, the NATO started air strikes under the order of the UN (Fig. 17). The opposing parties finally participated in peace talks which took place in Dayton/Ohio and where signed in December 1995 (Dayton Peace Agreement, Fig. 18). After this, events in the Balkans calmed down and the media focused on the conflict between Turkey and the Kurds (already shown in Figs. 7 and 8).

The conflict between Turkey and the Kurds also exemplifies a problem with the data that we were not aware of before seeing the animation. In July 1997, there is an abrupt change in media coverage in the sense that reports on hostilities between Turks and Kurds are suddenly missing. Figures 19 and 20 show the conflict structure in the Balkans with only a few days in between. The change is also visible in a significant drop in the bipartiteness curve (Fig. 13), where the highly bipolar situation rapidly changes into a more complex one and in a peak in the volatility curve (Fig. 12), which becomes visible to the analyst by the red background in Fig. 20.

That this change is indeed supported by the data can be verified by printing the events involving `TUR` and `KUR`. During the period from May 10'th 1997 to June 10'th 1997 many hostile events between these two actors are reported (see Fig. 14). In contrast, from June 11'th 1997 until July 11'th 1997 there is *no* hostile event reported between `TUR` and `KUR`. There are no prominent historic events explaining this sudden "peace". However, turning to the data description gives the information that this is precisely the time when KEDS sources change from Reuters North America to

[6]`http://www.ku.edu/~keds/data.dir/balk.html`

```
970514 TUR KUR (MIL ENGAGEMENT) KILLED
970514 TUR KUR (MIL ENGAGEMENT) TROOPS CLASHED
970514 TUR KUR (MIL ENGAGEMENT) TURKISH PUSHED AGAINST KURDISH
970520 TUR KUR (MILITARY DEMO) HUNTING DOWN
970521 TUR KUR (MILITARY DEMO) BUILDING UP FORCES
970522 KUR TUR (MIL ENGAGEMENT) ATTACKED KILLING
970522 TUR KUR (MIL ENGAGEMENT) TURKISH PUSHED AGAINST KURDISH
970522 KUR TUR (NONMIL DEMO) STAGED PROTEST
970522 KUR TUR (DENIGRATE) CONDEMNATION
970524 KUR TUR (MIL ENGAGEMENT) ATTACKS ARMY
970526 TUR KUR (MIL ENGAGEMENT) TROOPS CLASHED
970527 TUR KUR (MIL ENGAGEMENT) BOMBED
970602 TUR KUR (MIL ENGAGEMENT) KURDISH KILLED IN TURKISH
970604 TUR KUR (MIL ENGAGEMENT) KURDISH KILLED IN TURKEY
970604 TUR KUR (ARREST PERSON) JAIL
970605 TUR KUR (MIL ENGAGEMENT) KILLED
970607 TUR KUR (DENY) DENIED
970609 TUR KUR (DENY) DENIED
970610 KUR TUR (DEMAND) DEMANDING
```

Figure 14: Hostile events between TUR and KUR from 05/10/97 to 06/10/97.

Reuters Business Briefing, with the latter apparently not covering the conflict. (Or TUR and KUR being filtered out during preparation of the data for the KEDS parser.) The different news sources are

01 Apr 1989 – 31 Dec 1990	Reuters Business Briefing
01 Jan 1991 – 10 Jun 1997	Reuters North America
11 Jun 1997 – 31 May 1999	Reuters Business Briefing
01 Jun 1999 – 31 Jul 2003	Agence France Press

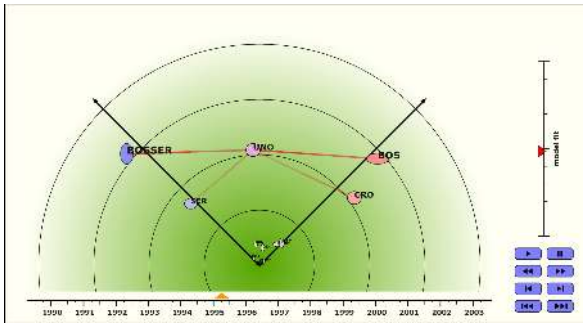

Figure 15: War in Bosnia (only edges incident to UNO are shown). Two opposing groups and UNO trying to mediate.

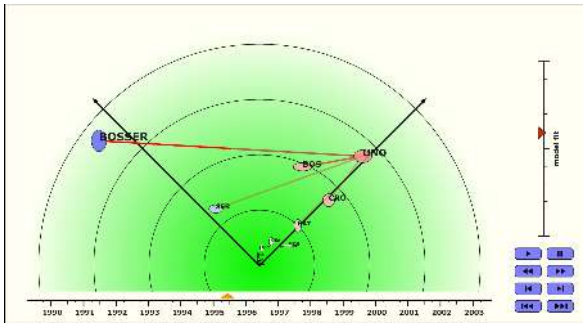

Figure 16: BOSSER's troops in conflict with the UN. The heavy edge pushes UNO to the red group.

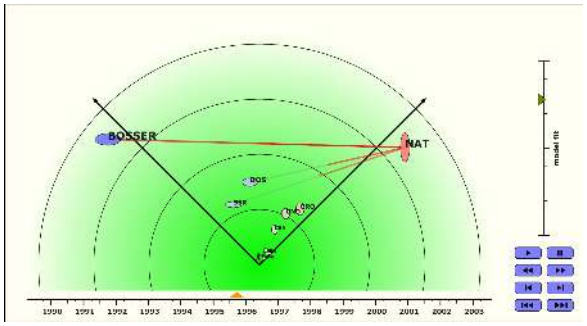

Figure 17: NATO bombing in Bosnia. Note that BOSSER changed from high and narrow (being the source of events) in Fig. 16 to broad and flat (being a target).

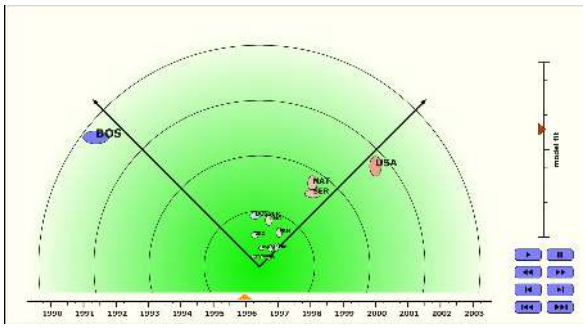

Figure 18: Dayton peace talks.

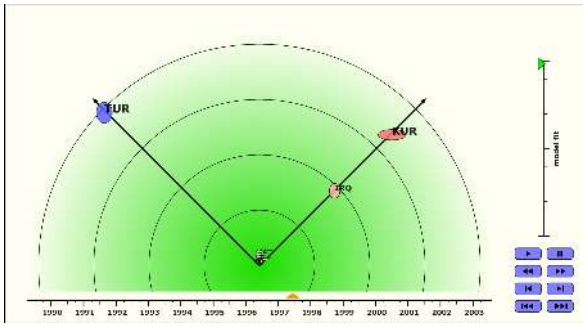

Figure 19: Conflict between Turkey and the Kurds before the change from Reuters North America to Reuters Business Briefing.

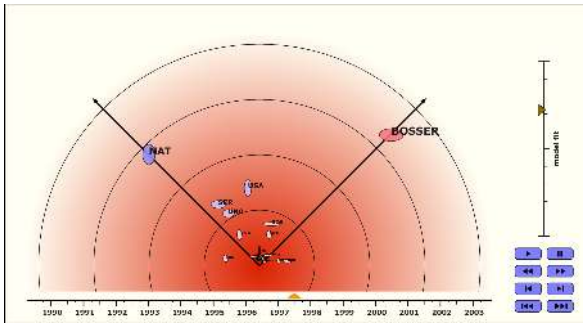

Figure 20: During the change: A red background indicates a highly unstable projection. TUR and KUR are still visible to the left and to the right of the origin and are rapidly moving towards it.

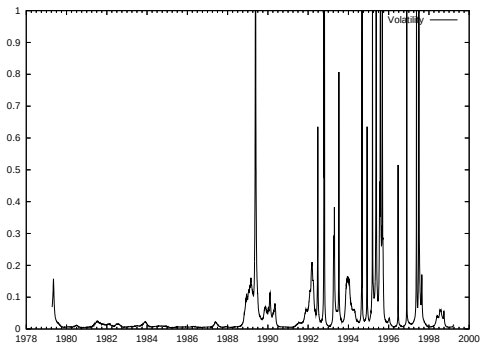

Figure 21: Volatility profile of the Gulf Conflict.

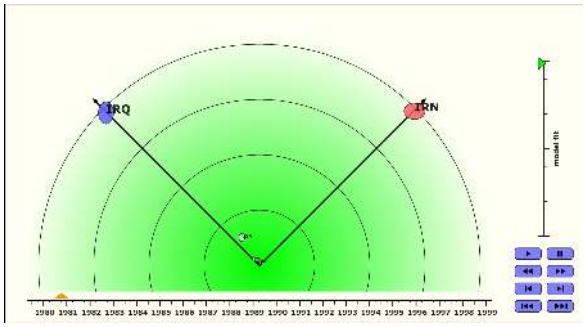

Figure 22: Iraqi invasion of Iran.

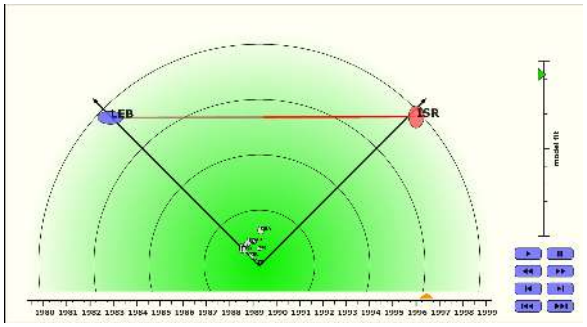

Figure 23: Conflict pair Lebanon vs. Israel.

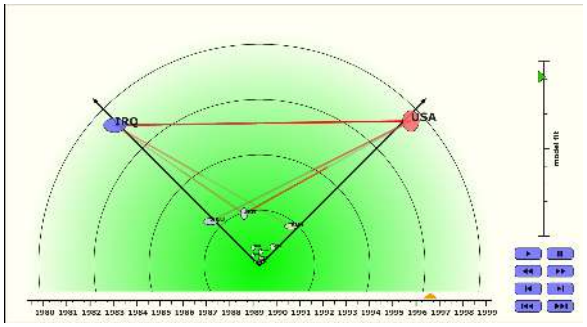

Figure 24: Conflict pair Iraq vs. USA.

7.2 Persian Gulf

The KEDS Gulf data set[7] contains 57,000 events for the states of the Gulf region and the Arabian peninsula for the period 15 April 1979 to 31 March 1999. The volatility plot, computed with $\delta = 0.2$, is shown in Fig. 21.

In September 1980, the Iraqi invasion of Iran (shown in Fig. 22) took place. The projection shown in Fig. 22 remains almost unchanged until 1987, although Iraq and Iran switch several times from being the main source to being the main target of the hostilities and back. The very flat line in the first half of the volatility plot (Fig. 21) is a quantitative measure for the continuity or pervasiveness of this conflict.

Starting from the end of 1991 a repeated switch between the two conflict pairs USA vs. Iraq on one hand and Israel vs. the Lebanon on the other can be observed. This interchange of the dominant conflict structure is also reflected in the volatility plot (Fig. 21) which shows many peaks. Figures 23 and 24 show one of these changes which took place in 1996. This oscillation is not an artifact of our method but can actually be found in the data. We calculated the aggregated negative weights of the undirected edges (`ISR,LEB`) and (`IRQ,USA`) and plotted them in Fig. 25. For example, the weight of (`ISR,LEB`) is significantly higher than the weight of (`IRQ,USA`) at the beginning of 1996. This priority becomes inverted in the second half of 1996, which is the cause for the change from Fig. 23 to Fig. 24.

8 Discussion

We have presented a method for designing scatterplots that represent the conflict structure embodied in event data, and a method for smoothly animating these scatterplots to highlight conflict dynamics. One of the main advantages of the proposed method is that it allows for a rigorous stability analysis and—on the above example data sets—actually proved to be stable. There are several interesting avenues for extending and improving our visualizations which will be addressed in further research.

A straightforward extension is secondary analysis of our dynamic projections. For instance, it might be interesting to focus on particular actors and follow their trajectories in the animated scatterplot and relate it to their involvement profile. Sensitivity analysis may point to crucial events or actors that trigger major changes in the structure. Ultimately, combinations of these analyses could serve as early warning indicators.

User feedback. Due to the prototypical nature of our implementation we have not yet been able to perform a systematic evaluation of what users perceive in and conclude from the animations. However, we did discuss the animations with domain experts both

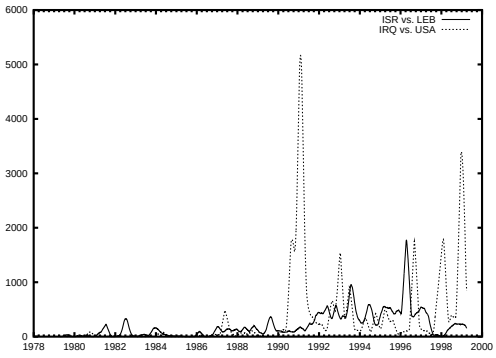

Figure 25: Aggregated weight of edges (`ISR,LEB`) and (`IRQ,USA`).

[7] `http://www.ku.edu/~keds/data.dir/gulf.html`

in regional politics and communication science. According to their feedback, the results of our analyses and the visualizations are indicative of many facts they were aware of, and even of interesting aspects that had escaped their attention up to then. Since all of them had at least some training in network analysis, future testing should include subjects who have no understanding of the inner workings of our approach.

Beyond bipolar conflict structures. The assumption of one dominant bilateral conflict is quite well satisfied in the example data for most time steps. However, in some situations (especially when generating the conflict networks for longer periods of time) this could be violated in two directions: Firstly, there could be more than two groups that are mutually in conflict (k-lateral conflicts). For instance, the situation in the Persian Gulf for the period from 1979 to 1999 (shown in Fig. 5) is best described as a triangle formed by the USA, Iraq, and Iran, since these actors have mutually strong negative ties. Secondly, the conflict network might contain several (mostly) independent major conflicts. An example is provided by the two conflict pairs (`ISR,LEB`) and (`IRQ,USA`), which are alternately visible in the Gulf video (see Figs. 23 and 24). (Note however, that the media rarely covers both conflicts with the same intensity, see Fig. 25. While it seems to be improbable that one pair stopped fighting whenever the others increased their hostilities, this is sometimes suggested by the media coverage.)

The analysis method based on structural similarities can be extended to cope with both situations. However, to generalize the complete visualization technique, several problems have to be solved. To detect k-lateral conflicts, a k-clique has to be used as quotient instead of the bipolar quotient, shown in Fig. 4. The eigenvalues of this quotient determine, according to Theorem 3, the eigenvectors to project on. Similarly, to handle independent bilateral conflicts, the union of several copies of the graph in Fig. 4 has to be used as quotient. Combinations of both (i. e., multiple k-lateral conflicts) are also possible.

The problems to be solved are the following. Firstly, one has to decide on the model (quotient) for the conflict structure. Moreover, this decision has to be made dynamically as the conflict network evolves. Secondly, the non-uniqueness of the direction of the eigenvectors (compare Sect. 4) becomes harder to resolve. In the bipolar case only one eigenvector (potentially) had to be reversed. In the more general case the correct combination of reversions has to be chosen. Finally, to visualize the projection matrix (which in the general case is a $k \times n$ matrix) it has to be mapped to two-dimensional space. Possibilities include parallel coordinates for the different conflicts, or displaying several adjacent frames on one screen. All three tasks require further work, to yield an applicable method.

As a different generalization, we may extend our data basis by also considering events with positive weights. Natural groups are identified by clustering actors based on cooperation, and these can be compared to or integrated with a partitioning based on hostility.

References

[1] James Allan, Rahul Gupta, and Vikas Khandelwal. Temporal summaries of news topics. In *Proc. ACM-SIGIR'01*, pages 10–18, 2001.

[2] James Allan, Ron Papka, and Victor Lavrenko. On-line new event detection and tracking. In *Proc. ACM-SIGIR'98*, pages 37–45, 1998.

[3] Clive Best, Erik Van der Groot, and Monica de Paola. Thematic indicators derived from world news reports. In *Proc. International Conference on Intelligence and Security Informatics (ISI 2005)*, pages 436–447, 2005.

[4] Ulrik Brandes and Thomas Erlebach, editors. *Network Analysis – Methodological Foundations*. Springer-Verlag, 2005.

[5] Ulrik Brandes, Daniel Fleischer, and Jürgen Lerner. Highlighting conflict dynamics in event data. In *Proc. IEEE Symp. Information Visualization (InfoVis '05)*, pages 103–110, 2005.

[6] Ulrik Brandes and Jürgen Lerner. Structural similarity in graphs. In *Proc. 15th Intl. Symp. Algorithms and Computation (ISAAC'04)*, pages 184–195, 2004.

[7] John M. Chambers, William S. Cleveland, Beat Kleiner, and Paul A. Tukey. *Graphical Methods for Data Analysis*. Wadsworth, 1983.

[8] William S. Cleveland and Robert McGill. The many faces of a scatterplot. *Journal of the American Statistical Association*, 79(388):807–822, 1984.

[9] Joshua S. Goldstein. A conflict-cooperation scale for WEIS international events data. *Journal of Conflict Resolution*, 36(2):369–385, 1992.

[10] Gene H. Golub and Charles F. van Loan. *Matrix Computations*. John Hopkins University Press, 1996.

[11] Valerie M. Hudson, Philip A. Schrodt, and Ray D. Whitmer. A new kind of social science: The path beyond current (IR) methodologies may lie beneath them. Paper presented at the Annual International Studies Association Conference, Montreal, 2004.

[12] Michael Kaufmann and Dorothea Wagner, editors. *Drawing Graphs*. Springer-Verlag, 2001.

[13] Michael Krivelevich and Van H. Vu. On the concentration of eigenvalues of random symmetric matrices. Technical Report MSR-TR-2000-60, Microsoft Research, 2000.

[14] Charles A. McClelland. World event/interaction survey codebook (icpsr 5211), 1976.

[15] Phillip A. Schrodt, Shannon G. Davis, and Judith L. Weddle. Political science: KEDS-a program for the machine coding of event data. *Social Science Computer Review*, 12(3):561–588, 1994.

[16] Ben Shneiderman. The eyes have it: A task by data type taxonomy for information visualizations. In *Proc. 1996 IEEE, Visual Languages*, 1996.

[17] G. W. Stewart and Ji-Guang Sun. *Matrix Perturbation Theory*. Academic Press, 1990.

[18] Thomas Widmer and Vera Tröger. Event data based network analysis. Paper presented at the 45th Annual Convention of the International Studies Association, Montréal, 2004.

[19] Pak Chung Wong, Harlan Foote, Dan Adams, Wendy Cowley, and Jim Thomas. Dynamic visualization of transient data streams. In *Proc. IEEE Symp. Information Visualization (InfoVis '03)*, pages 97–104, 2003.

[20] Kazuo Yamaguchi. *Event History Analysis*. Sage, 1991.

Ulrik Brandes is a full professor of computer science at the University of Konstanz. He graduated from RWTH Aachen in 1994, obtained his PhD and Habilitation from the University of Konstanz in 1999 and 2002, and was an associate professor at the University of Passau until 2003. His research interests include algorithmic graph theory, graph drawing, social network analysis, information visualization, and algorithm engineering.

Daniel Fleischer graduated from the University of Konstanz in 2002. Currently he is a research assistant at the Department of Computer and Information Science at the University of Konstanz. His research areas include algorithm engineering and information visualization.

Jürgen Lerner graduated from the University of Konstanz in 2002. Currently he is a research assistant at the Department of Computer and Information Science at the University of Konstanz. His research areas include the analysis of large data sets and information visualization.