Tracing the Sources of Belief Contestation in Policy Debates

Philip Leifeld Tim Henrichsen

This version: July 31, 2023

Abstract

Political actors agree or disagree with other actors on policy beliefs. When aggregated into a policy subsystem, advocacy coalitions with distinct belief systems emerge from actors' individual policy belief portfolios. Discourse network analysis measures coalitions by considering actors' stated beliefs. But policy beliefs differ in how important they are in strucuring coalitions. To understand the ideational "glue" that binds coalitions together or keeps them apart in any given subsystem, we must identify the joint subset of beliefs that is structurally most important for the coalition structure. We call this subset the backbone of a policy debate and distinguish it from its complement, the set of redundant beliefs. To identify the backbone and redundant set, we introduce a penalized spectral loss function and a custom simulated annealing algorithm to identify the backbone and redundant belief sets by combinatorial optimization. The approach is illustrated using the discourse network of German pension politics.

Paper presented at the workshop "Measuring, Modeling and Mitigating Opinion Polarization and Political Cleavage (MMM)", ETH Zurich, 13–15 September 2023.

1 Introduction

When a complex problem enters the political agenda, actors begin to discuss possible solutions to the problem (Béland 2005). Interest groups try to influence the policy process into directions that help their business or mission. Legislators and agencies of the executive government join in to pursue policy goals, ideological goals, and electoral goals (Strom 1990) when they discuss policy solutions. Scientists and experts seek attention and impact through their contributions to policy debates (Oliver and Cairney 2019). These different kinds of actors, with their different principals and incentives, engage in a "framing contest" (Kaplan 2008) and collectively shape the direction of policymaking in any given policy subsystem through *policy debates*.

Discourse network analysis is a methodology for the empirical analysis of policy debates (Leifeld 2017). It combines category-based qualitative content analysis with social network analysis to operationalize framing contests as complex, dynamic networks. Discourse networks link actors to other actors through their shared agreement on concepts. Concepts can include policy beliefs, arguments, or solution concepts for policy problems, depending on the theory that is operationalized. Discourse network analysis has been employed to measure advocacy coalitions (e. g., Leifeld 2013), a central concept in the advocacy coalition framework (Jenkins-Smith and Sabatier 1994), and discourse coalitions (e. g., Leifeld and Haunss 2012), a central concept in argumentative discourse analysis (Hajer 1995).

The *advocacy coalition framework*, a prominent theoretical approach in policy studies, posits that policy subsystems are structured around competing advocacy coalitions (Jenkins-Smith and Sabatier 1994). Advocacy coalitions are an analytical construct. They do not possess any fixed institutional boundaries, formal memberships, or agency. It is the researcher who identifies a set of actors as an advocacy coalition by distinguishing the belief system they share from the belief systems of other coalitions in a policy subsystem (Jenkins-Smith et al. 1991). Each coalition is defined around deep core beliefs, policy core beliefs, and secondary aspects—levels of policy beliefs that can be ordered from fundamental to applied beliefs (Jenkins-Smith and Sabatier 1994).

To measure advocacy coalitions around shared belief systems, applications of discourse network analysis have focused on a single belief level: *policy core beliefs*. Most researchers opt to examine policy core beliefs because these mid-level beliefs "glue" actors together in their advocacy coalitions (Kukkonen et al. 2017). Researchers who aim to identify advocacy coalitions therefore usually develop a coding system of maximally orthogonal policy beliefs to cover all dimensions in which actors could differ with regard to key aspects of the policy problem. They collect text data, such as newspaper articles, social media posts, or Congressional testimony, and annotate them by highlighting "statements." A statement is a text fragment where an actor speaks about a concept in a positive or negative way. After coding the statements in a text corpus, the set of statements can be transformed into network data. There are several types of networks that are interesting for policy analysis. The most common type of discourse network is an *actor congruence network*, in which actors are linked to other actors through their shared policy core beliefs. Clusters in the actor congruence network are interpreted as advocacy coalitions, noting that the more recent literature on advocacy coalitions has added other network relations, such as coordination and information exchange, to complement policy belief systems (Weible et al. 2020).

However, not all policy core beliefs are created equal. This aspect has been undertheorized in the literature on advocacy coalitions and discourse networks. First, there is variation in how ideologically extreme or moderate a concept is and whether it is used by extreme or moderate actors within advocacy coalitions. Leifeld et al. (2022) have suggested a scaling approach based on item response theory to measure how extreme or moderate actors and concepts are in advocacy coalitions. But how extreme or moderate concepts are is usually not taken into account when constructing actor congruence networks to measure advocacy coalitions. Second, some concepts may be more general (tending toward deep core beliefs) while other concepts may be more specific (tending toward secondary aspects). The boundaries between the three levels of policy beliefs are fluent, with some policy core beliefs resembling secondary aspects more than deep core beliefs and vice-versa. Hence, there is heterogeneity both within and across the three belief levels. Policy beliefs, after all, form a *system*, rather than a collection of independent traits. Third, concepts may be more important or less important for structuring the discourse network or subsystem into coalitions. Some concepts may be divisive and cause polarization across coalitions, whereas other concepts may be *redundant*.

There are different ways in which a concept can be redundant. A concept can be redundant if all actors agree to it, hence providing no information about coalition membership. A concept can also be redundant if there is no clear pattern in how the concept aligns with the coalition structure of the discourse network, thus adding merely random "noise" to the compartmentalization into coalitions. Or a concept can be redundant if it promotes the same coalition structure as the bulk of other concepts, thereby adding nothing to the compartmentalization of the collective belief system of the policy subsystem. In contrast, a concept is not redundant if it structures the discourse network into clear coalitions or supports the internal cohesion of at least one coalition, if it adds something to this structure that has not been added by the bulk of other concepts, and if it does not "go against the grain" by contradicting the structure imposed by the bulk of other concepts. In other words, a concept is not redundant if it forms part of the *backbone* of the discourse network, contributing to its coalition structure.

Researchers have dealt with the heterogeneity among concepts primarily in two ways (Leifeld 2017): First, they have focused on actor congruence networks, which collapse the concepts into similarity weights between actors as if the concepts were independent, orthogonal, and equally important, thereby largely ignoring the heterogeneity. Second, they have focused on *concept congruence networks*, which explicitly examine the interdependencies between concepts by projecting the number of actors who share any two concepts into similarity weights for the concepts, thereby shifting attention away from actors and toward the contents of a policy debate.

In the present article, we suggest ways to pinpoint the policy belief backbone of a policy subsystem and to distinguish it systematically from redundant and peripheral policy beliefs. Doing so has two main advantages.

The first advantage is a substantive one. The advocacy coalition framework posits that policy beliefs are arranged in a collective *belief system*. Treating policy beliefs as a system requires understanding which beliefs play which role in the system. To understand how policy subsystems are structured, it is, for example, important to know around which subset of policy beliefs polarization takes place. For example, Fisher et al. (2013) investigate around which concepts polarization in the US climate policy debate takes place. But they examine one concept at a time. Polarization may be jointly caused by a subset, or subsystem, or policy beliefs that need to be taken into account together. Tracing which concepts jointly form the backbone of contestation in a policy subsystem means measuring how central some policy beliefs are jointly to a policy debate.

Identifying the sources of contestation will permit a more accurate description of coalitions and subsystem dynamics, which will in turn inform theory. If progress is to be made in the measurement of *policy learning within and across coalitions*, the central theoretical element in the advocacy coalition framework hypothesized to cause structural change in policy subsystems (Weible et al. 2020), then it will be imperative to know which policy beliefs are sources of contestation and which policy beliefs are peripheral at any given point in time. Understanding which concepts jointly form the backbone of the coalition structure in a policy subsystem also means learning which policy beliefs are redundant. Redundant concepts may be unimportant, or they may not *yet* be important and may thus be relevant for policy learning. If a belief or set of beliefs is structurally important, it is *central* to the debate. Hence backbones and redundancy can serve as a distinct network centrality measure for concepts, making it unnecessary to misapply existing network centrality measures to concepts in discourse networks. Thus, we need to develop ways to measure backbones of policy subsystems by identifying redundant concepts.

The second advantage of measuring backbones and redundant concepts is a methodological one. Redundant concepts are *outliers*, given the remaining network structure. They can be outliers either because they are substantively less contested than other concepts or because they have been miscoded during the annotation phase of the research.

For example, if a concept captures the same meaning as another concept, the two should be merged to reduce redundancy. If redundant concepts can be reliably detected in discourse networks, doing so will improve the annotation and quality of discourse network datasets.

If a concept elicits agreement by all actors or by actors from different coalitions without a clear pattern, it can be a sign that the concept may have been ill-specified and should be checked for usefulness. The goal of the annotation is to produce data following a good coding scheme. Coding schemes are models of the data, and as models they should be *parsimonious, yet complete*. A coding scheme is parsimonious, yet complete if it contains a small number of distinct concepts that jointly capture maximal information about the structure of the policy subsystem when applied to the text data. Identifying the backbone of a discourse network by removing redundant concepts is, therefore, not only an important substantive task, but will also serve to improve the quality of the data by removing outliers and flagging redundant concepts in the coding scheme, hence leading to a better annotation model.

In the present article, we will make two contributions: First, we will introduce a method for the identification of backbones and redundant concepts in discourse networks. The method utilizes a custom combinatorial optimization approach to find the minimal subset of concepts that reproduces the full discourse network without a significant loss of information. It partitions the set of concepts into two disjoint subsets: a backbone set and a redundant set. The method can be adjusted to account for different levels of granularity to penalize more or less heavily for keeping many concepts in the backbone set. Second, we will illustrate this method using a discourse network dataset on German pension politics in the year 2000, when the debate around pension reform was most contested in Germany. Doing so will present the concept backbone and concept redundancy as two interrelated features of the collective belief system in a policy subsystem. The two features are theoretically distinct from other features of policy subsystems, such as advocacy coalitions and brokers, and they serve to delineate policy core beliefs more clearly than arbitrary distinctions of belief layers.

2 Models and Methods

2.1 Construction of Discourse Networks

In discourse network analysis, we model the relationships among actor vertices $a_i \in A$ and concept vertices $c_j \in C$ via an edge qualifier $q_k \in Q$. The qualifier indicates agreement ($k = 1$) or disagreement ($k = 2$) by actor i regarding concept j. The number of statements by actor a_i about concept c_j with qualifier q_k is saved in cell x_{ijk} of a cuboid tensor $\mathbf{X} \in \mathbb{N}_0^{|A| \times |C| \times 2}$. In the present application, we remove duplicate statements by letting $x_{ijk} = 1$ if at least one statement involving this actor–concept–qualifier combination was present empirically and $x_{ijk} = 0$ otherwise, $\forall i, j, k$. We transform the tensor data into a weighted adjacency matrix $\mathbf{Y}^{|A| \times |A|} \in \mathbb{R}_0^+$ with edge weights $y_{ii'}$ representing shared agreement among actors i and i' over concepts in excess of disagreement over concepts, normalized by activity:

$$y_{ii'} = \max \left(0, \frac{\sum_{j=1}^{n} \sum_k x_{ijk} x_{i'jk} - \sum_{j=1}^{n} \sum_k x_{ijk} x_{i'jk'}}{\frac{1}{2} \left(\sum_{j=1}^{n} \sum_k x_{ijk} + \sum_{j=1}^{n} \sum_k x_{i'jk} \right)} \right) \tag{1}$$

The first summation in the numerator is known as an *actor congruence network*, the second summation as an *actor conflict network*, and the denominator as *average activity normalization* in the literature on discourse network analysis (Leifeld 2017). The actor congruence network connects actors through shared concept–qualifier combinations. The actor conflict network connects actors through qualifier disagreements over shared concepts. The denominator normalizes the edge weight by the average total activity of both actors to prevent a core–periphery structure due to the presence of highly active actors (Leifeld 2017).

Subtracting the normalized conflict network from the normalized congruence network yields a normalized belief similarity measure among actors in excess of conflict between the same actors, with positive values indicating more agreement than disagreement between them and negative values indicating more disagreement than agreement. We discard negative values to facilitate network visualization. We call this method of constructing discourse networks the *subtract method*.

2.2 Backbone and Redundant Concepts

The aim of this research is to partition concept set C into backbone $B \subset C$ and redundant set $R \subset C$ such that $R \cup B = C$, $R \cap B = \emptyset$, and to partition $\mathbf{X} \rightarrow \left\{\mathbf{X}^B \in \mathbb{N}_0^{|A| \times |B| \times 2}, \mathbf{X}^R \in \mathbb{N}_0^{|A| \times |R| \times 2}\right\}$ and subsequently map $\mathbf{X}^B \rightarrow \mathbf{Y}^B$ and $\mathbf{X}^R \rightarrow \mathbf{Y}^R$ to yield actor networks for the backbone and the redundant concept set.

The full network defined by matrix $\mathbf{Y}$ typically displays a modular topology with multiple communities, or clusters, representing advocacy coalitions. When we reduce $\mathbf{Y}$ to $\mathbf{Y}^B$, two conflicting goals need to be balanced: We want to retain as much as possible of the community structure, and we want to do so using as few concepts as possible. This implies that we want to discard as many concepts as possible into the redundant set without incurring much loss of information in the network topology of $\mathbf{Y}^B$, yielding a redundant network $\mathbf{Y}^R$ with as much useless information as possible. To achieve this, we minimize a topological distance function between the original network and the backbone network and penalize the solution using an exponential decay function applied to the number of concepts retained in B. We will describe both criteria in turn.

2.3 Euclidean Spectral Distances as a Topological Loss Function

The backbone is supposed to be a good model of the original network. Hence it should preserve its logical topology as much as possible. We can achieve this by minimizing a distance measure between $\mathbf{Y}$ and $\mathbf{Y}^B$ that reflects differences in their logical topology.

There are many distance measures for networks one could choose from, some of which compare two networks with maximal detail while others compare abstractions of two networks. A distance measure that would capture differences between networks in maximal detail is the *graph edit distance* (Gao et al. 2010). For two networks with equal node sets, it measures the absolute difference in edge weights, summed over all elements of $\mathbf{Y}$. An abstract distance measure, which compares based on a simplifying function of the network, is the *difference in maximal modularity* (Newman 2006) between $\mathbf{Y}$ and $\mathbf{Y}^B$. It measures how strongly the two networks differ in their tendencies to contain communities. For our purposes, however, the graph edit distance does not sufficiently take the community structure into account, and the modularity distance does not sufficiently consider the location and connectedness of communities.

Hence we opt for a middle ground, which focuses on the number and sizes of communities in the network and actor vertices' relations to these communities in order to construct the backbone: We compute the *Euclidean spectral distance* between the eigenvalues of the Laplacian matrices of $\mathbf{Y}$ and $\mathbf{Y}^{B^*}$, where B^* is a candidate set for B during optimization. The Euclidean spectral distance essentially compares the two networks by computing the dissimilarity between their cluster topologies. The Laplacian matrix $\mathbf{L}$ of $\mathbf{Y}$ is given by

$$\mathbf{L} = \mathbf{D} - \mathbf{Y}, \tag{2}$$

where $\mathbf{D}$ is the degree matrix of $\mathbf{Y}$, containing the row sums of $\mathbf{Y}$ on the diagonal and zeros elsewhere (Shore and Lubin 2015). The spectrum of L is the ordered set of eigenvalues, $\boldsymbol{\lambda}$, satisfying $0 = \lambda_1 \leq \lambda_2 \ldots \leq \lambda_{|A|}$ and $\sum_i \lambda_{i=1}^{|A|} = \sum_{i=1}^{|A|} \sum_{i'=1}^{|A|} y_{ii'}$. Following Shore and Lubin (2015), who employed Euclidean spectral distances in a different context, we standardize $\boldsymbol{\lambda}$ as a fraction of the largest eigenvalue to achieve comparability between the eigenvalues of $\mathbf{Y}$ and $\mathbf{Y}^{B^*}$:

$$\forall i : \hat{\lambda}_i = \frac{\lambda_i}{\sum_{l=1}^{|A|} \lambda_l} \tag{3}$$

The loss function we employ (without penalty, i. e., $p = 0$) is then the Euclidean distance between the normalized spectrum of the Laplacian matrix of $\mathbf{Y}$ and the normalized spectrum of the Laplacian matrix of $\mathbf{Y}^{B^*}$:

$$\ell\left(\mathbf{Y}, \mathbf{Y}^{B^*} | p = 0\right) = \sqrt{\sum_{i=1}^{|A|} \left(\hat{\lambda}_i^{\mathbf{Y}} - \hat{\lambda}_i^{\mathbf{Y}^B}\right)^2} \tag{4}$$

We determine the set of backbone concepts B by minimizing the loss function between the two networks. The candidate set of concepts B^* that minimizes the Euclidean spectral distance becomes the backbone set B.

$$B = \underset{B^* \subseteq C}{\arg\min}\, \ell(\mathbf{Y}, \mathbf{Y}^{B^*}, p) \tag{5}$$

However, a modification of this minimization is required because the set that minimizes the topological loss is the complete set and larger sets tend to incur smaller losses. Hence, a penalty p for the cardinality of the candidate backbone set of concepts, $|B^*|$, is required as a second criterion to be included in the optimization.

2.4 Penalty for Backbone Cardinality

Typically, $|C|$ is in the range of several dozen concepts. A penalty for the loss function is required to prevent $|B^*|$ to approach $|C|$ because only a parsimonious model can be useful. Hence we weight the loss function by an exponential decay for the share of concepts involved in the proposed backbone solution, with parameter p set by the researcher depending on how important a parsimonious model is relative to spectral loss minimization:

$$\ell\left(\mathbf{Y}, \mathbf{Y}^{B^*}, p\right) = \sqrt{\sum_{i=1}^{|A|} \left(\hat{\lambda}_i^{\mathbf{Y}} - \hat{\lambda}_i^{\mathbf{Y}^{B^*}}\right)^2} e^{-p\frac{|B^*|}{|C|}}. \tag{6}$$

Panel B in Figure 1 illustrates the exponential weighting factor resulting from different settings of penalty p from 0 to 12 for a given $|C| = 57$. Candidate solutions with almost as many concepts as $|C|$ are penalized heavily while lower numbers of concepts in the candidate set are penalized less. The penalty is switched off when $p = 0$, as indicated by the vertical line in Panel B.

2.5 Combinatorial Optimization using Simulated Annealing

The loss function in Equation 6 is minimized as per Equation 5 to find the backbone set. However, this minimization task is a combinatorial optimization problem with a large search space, requiring an iterative algorithm. It is insufficient to compute the loss incurred by removing each individual concept and then aggregate into a set because

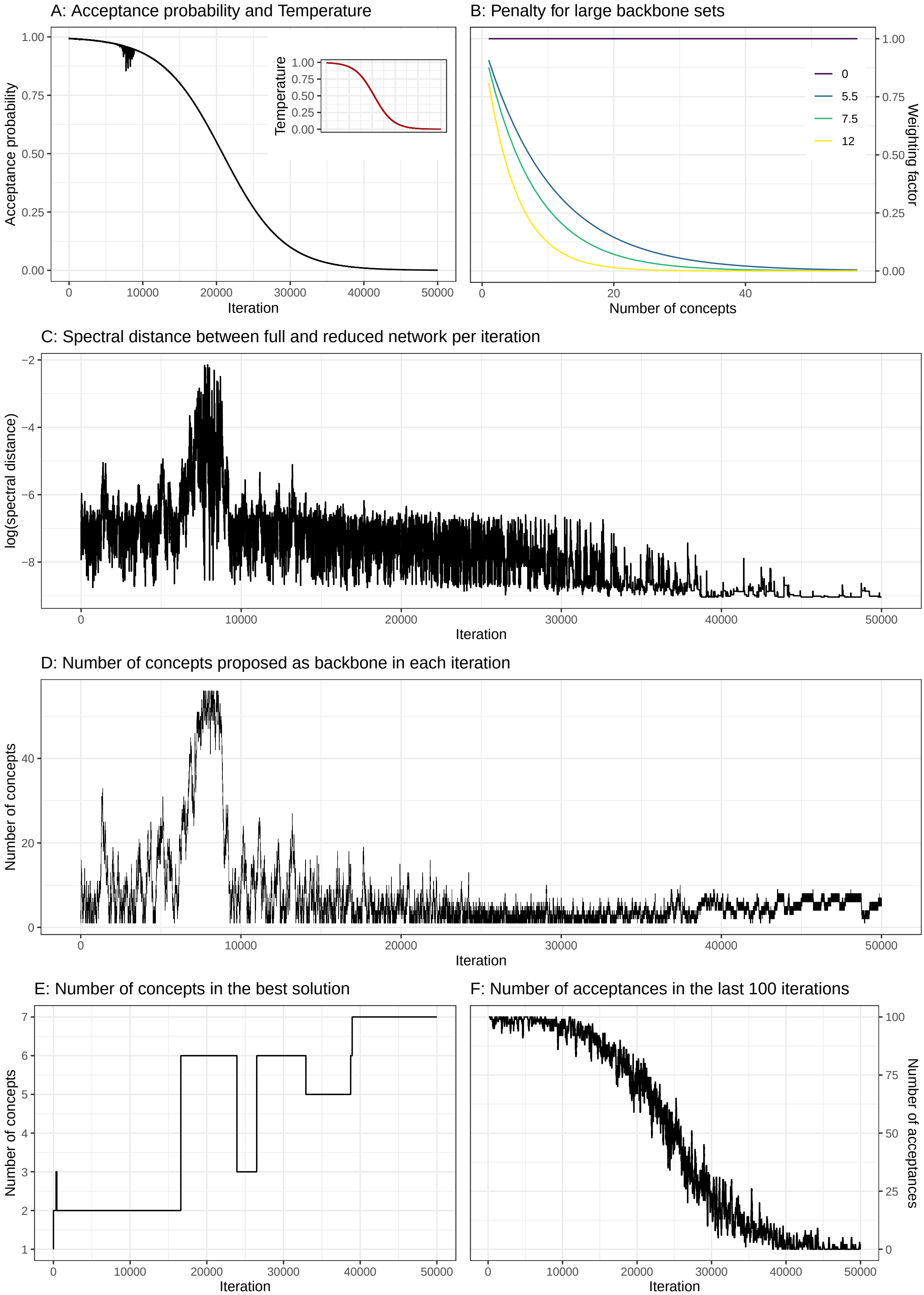

Figure 1: Summary of the optimization process for the case study.

the concepts may interact in how much they contribute to the structure of the network. With, say, $|C| = 57$, the search space comprises $2^{57} = 1.44 \times 10^{17}$ possible solutions.

To find the optimal solution that minimizes the weighted spectral loss, we employ a custom simulated annealing algorithm shown in Algorithm 1. `R` code with an implementation of this algorithm is listed in the appendix. The algorithm proceeds over a set number T of iterations t (e.g., $T = 50,000$, determined by the user). It starts with a backbone set (to be optimized) with a random concept $B = \{c_j\}$, where $c_j \sim U(C)$, and gets the opportunity to replace the set by a better candidate set in each iteration. Before $t = 1$, a copy of B is saved as B^*_0.

A three-stage random sampling process determines in each iteration how to determine a new candidate set B^*_t. At the first sampling stage, three actions are eligible to be selected with uniform probability, adding a concept $c_j \notin B^*$ to B^* (available if $|B^*| < |C| - 1$), removing a concept $c_j \in B^*$ from B^* (available if $|B^*| > 1$), and replacing a concept $c_j \in B^*$ by a random concept $c_{j'} \notin B^*$ (available if $0 < |B^*| < |C|$). If the randomly drawn action involves the addition of a concept (through the addition or replacement action), at the second sampling stage a random concept $c_{j'}$ from $C \setminus B^*$ is selected for addition with random probability. If the randomly drawn action involves replacement, at the third stage a random concept from B^* is sampled for deletion and replacement by the concept drawn at the second sampling stage. The outcome of this sampling process at iteration t is translated into a modified backbone candidate set B^*_t, on the basis of B^*_{t-1} and the modification determined by the sampling.

After creating B^*_t, the penalized loss function from Equation 6 is applied to establish whether the current or previous candidate has a smaller loss compared to the full network. If the new candidate improves the solution, it is retained and becomes the new B^*_{t-1} at $t := t + 1$. If its loss is not smaller than the previous candidate solution, it is accepted only sometimes.

Algorithm 1: Simulated Annealing

Input: $\mathbf{X}, p, T$

Result: Optimal backbone B and redundant concept set R

```
X → Y ;                                          /* Map data into full network */
c_j ~ U(C) ;
B ← B*_0 ← {c_j} ;                      /* Start with a single random concept */
t ← 1 ;                                             /* Start first iteration */
while t ≤ T do
    S ← {add; remove; swap} ;            /* Sample space of possible actions */
    if |B*_t| < 2 then
        S ← S \ {remove} ;                     /* Never remove all concepts */
    end
    if |B*_t| > |C − 2| then
        S ← S \ {add} ;                           /* Never add all concepts */
    end
    s ← U(S) ;                                        /* Pick action randomly */
    if s = {add} then
        c_j ~ U(C \ B*_t) ;
        B*_t ← B*_t ∩ {c_j} ;                         /* Add a random concept */
    end
    if s = {remove} then
        c_j ~ U(B*_t) ;
        B*_t ← B*_t \ {c_j} ;                      /* Remove a random concept */
    end
    if s = {swap} then
        c_j ~ U(B*_t) ;
        c_j' ~ U(C \ B*_t) ;
        B*_t ← {c_j'} ∪ B*_t \ {c_j} ;            /* Swap out random concepts */
    end
    X → X^{B*_t} → Y^{B*_t} ;         /* Map data into current backbone network */
    if ℓ(Y, Y^{B*_t}, p) < ℓ(Y, Y^{B*_{t−1}}, p) then
        B*_t ← B*_t ;                                     /* Accept candidate */
        if ℓ(Y, Y^{B*_t}, p) ≤ ℓ(Y, Y^B, p) then
            B ← B*_t ;                        /* Save globally optimal solution */
        end
    else
        δ_t = 1 − 1 / (1 + e^{−(−5 + (12/T) t)}) ;          /* Compute temperature */
        r ~ U_[0,1] ;                   /* Sample random number between 0 and 1 */
        if r < e^{−(ℓ(Y, Y^{B*_t}, p) − ℓ(Y, Y^{B*_{t−1}}, p))} × δ_t then
            B*_t ← B*_t ;                             /* Accept candidate sometimes */
        else
            B*_t ← B*_{t−1} ;       /* Reject candidate; use previous candidate */
        end
    end
end
R ← C \ B ;                 /* Create redundant set as complement of backbone */
```

$$B_t^* = \begin{cases} B_t^* & \text{if } \ell(\mathbf{Y}, \mathbf{Y}^{B_t^*}, p) < \ell(\mathbf{Y}, \mathbf{Y}^{B_{t-1}^*}, p) \\ B_t^* & \text{else if } r < e^{-\left(\ell(\mathbf{Y}, \mathbf{Y}^{B_t^*}, p) - \ell(\mathbf{Y}, \mathbf{Y}^{B_{t-1}^*}, p)\right)} \times \delta_t, \text{ with } r \sim U_{[0,\,1]} \\ B_{t-1}^* & \text{else.} \end{cases} \tag{7}$$

If the first condition does not hold, the exponentiated improvement in the backbone candidate set from $t-1$ to t is multiplied by the current temperature of the system. If the product is not greater than a uniform random number between 0 and 1, the current candidate B_t^* is also accepted, and B_{t-1}^* is retained otherwise.

The temperature δ_t follows cooling schedule

$$\delta_t = 1 - \frac{1}{1 + e^{-\left(-5 + \frac{12}{T} t\right)}}, \tag{8}$$

based on the inverse logit function, which is identical to both the logistic function and the cumulative distribution function of the logistic function with scale parameter $= 1$ and shape parameter $= 0$,

$$\text{logit}^{-1}(\alpha) = \text{logistic}(\alpha) = \frac{1}{1 + e^{-\alpha}} = \frac{e^{\alpha}}{1 + e^{\alpha}}, \tag{9}$$

with α being in the range between -5 at the first iteration and $+7$ at the last iteration to produce a flipped S curve with additional time for fine-tuning at the end. As $7 - (-5) = 12$, $\frac{12}{T}$ expresses the increase of each Δt on a linear scale towards $+7$ at the end of the linear scale. Hence $-5 + \frac{12}{T} t$ evenly spaces out the iterations over the range $[-5, 7]$.

The cooling schedule with the temperature as a function of t is shown in the inset of Panel A in Figure 1. The temperature scalar initially guarantees a high acceptance rate (under the second case in Equation 7) because the initial backbone solution starts with an empty set and needs time to explore the space freely. As the temperature cools after some iterations, the optimization gradually admits fewer acceptances that would increase the spectral loss until it hones in on a more localized search sub-space in which only hill climbing is permitted.

Finally, at each t in which a candidate solution is accepted, the accepted solution will be retained as the globally optimal backbone B if it minimizes the weighted loss at least as well as the previously assumed globally optimal backbone.

$$B = \begin{cases} B_t^* & \text{if } \ell(\mathbf{Y}, \mathbf{Y}^{B_t^*}, p) \leq \ell(\mathbf{Y}, \mathbf{Y}^{B}, p) \\ B & \text{if } \ell(\mathbf{Y}, \mathbf{Y}^{B_t^*}, p) > \ell(\mathbf{Y}, \mathbf{Y}^{B}, p) \end{cases} \tag{10}$$

This simulated annealing approach combines breadth (at high temperatures) with depth (at low temperatures) and therefore effectively finds an approximately optimal solution. While there is no guarantee that B is the globally optimal solution, repetition with empirical examples showed that the same solution is found almost each time the algorithm is run. The user chooses parameters p and T. Running time increases linearly with increasing T and is invariant to p. Larger T is advised with larger $|C|$. The choice of p guides how large the backbone is approximately. As p increases, smaller backbones are created and fewer concepts end up in R. Experience has shown that smaller backbones tend to be nested in larger backbones as p is changed.

3 Case Study: German Pension Politics

We illustrate the method using the discourse network of German pension politics in the year 2000. Figure 2 shows the actor network without scientific actors and actors without a clear actor type, and with advocacy coalitions identified as communities in the blue hyperplanes, as visualized in Leifeld (2013).

Germany around the turn of the millennium was (and still is at the time of writing) characterized by population aging. Population aging is a risk for a public pension system if the pension system ties pension levels in a given year to the volume of pension contributions by the working population in the same year or shortly before. Such a system is called a *pay-as-you-go system*. Pay-as-you-go systems are subject to financial pressures when the number of pension recipients grows while the number of workers who pay into

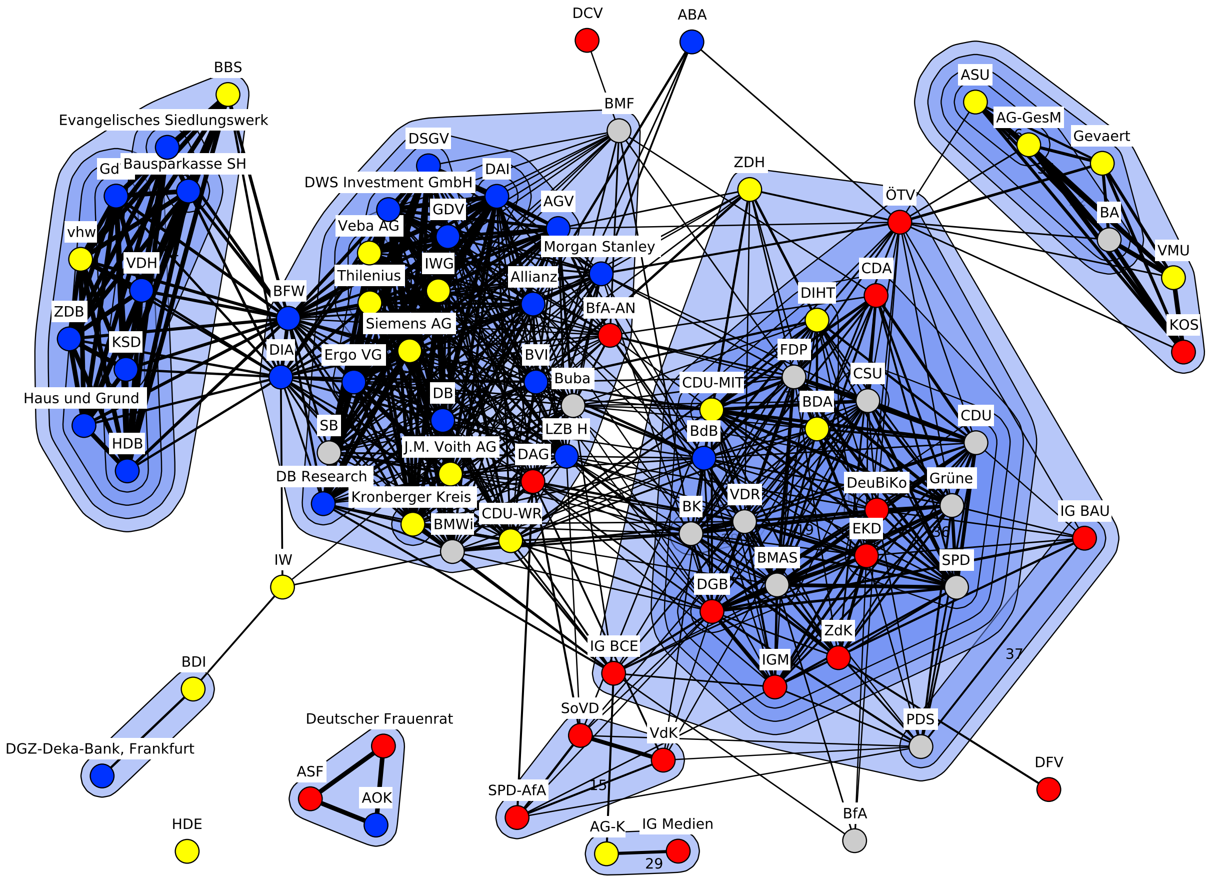

Figure 2: The discourse network of German pension politics in the year 2000, with actor labels and blue hyperplanes indicating advocacy coalitions identified using the Girvan-Newman edge betweenness algorithm (Girvan and Newman 2002). Diagram created using the software *visone* (Baur et al. 2001).

the pension system decreases, even when there is a financial buffer that can correct for random fluctuations.

The German pension system in the year 2000 was a pay-as-you-go pension system and was increasingly put under pressure by demographic developments, before it was eventually reformed in 2001. The problem pressure emerging from this situation led to a heated policy debate around how to solve the problem. There was widespread consensus that something had to be done to put the pension system on more sustainable financial footing, but actors had different stakes in the reform process and revealed different, and often contradictory, normative policy core beliefs. A more detailed analysis over several years was presented by Leifeld (2013). To illustrate the methods developed here, we chose the year 2000 as the most polarized time period with relatively clear-cut advocacy coalitions.

The discourse network data were manually collected using the software *Discourse Network Analyzer* (DNA). DNA was written for the purpose of the original study. The statements by political actors about 68 policy beliefs, or concepts, 57 of which were actively used in 2000, were found in newspaper coverage in one of the largest and highest-quality German newspapers (with a slight right-leaning ideological slant). The coding scheme was developed iteratively by two researchers, and disagreements in the manual annotation were discussed and resolved in order to develop a maximally valid system of orthogonal concepts representing the different policy core beliefs held by actors in the debate. The goal in this article is twofold: to probe the coding scheme for redundant concepts and to find out which policy beliefs were central in structuring the debate into coalitions.

In the year 2000, two main advocacy coalitions were in conflict. The first one can be found in the lower right corner of Figure 2. This coalition was composed of several political parties and governmental actors (in gray color), social groups and trade unions (in red), and a few peak employers' associations (in yellow). In the literature on German pension politics, this advocacy coalition was called the "hegemonic policy community" because it fought for retaining the status quo of the pension system and making only minor tweaks to make the pay-as-you-go system more sustainable. This coalition tried to fend off the perceived threat of privatization of the system into a capital cover system, in which workers save for their own future pensions, or a multi-pillar system.

Members of this coalition instead offered solutions that would save money within the existing system by removing elements from the pension system that were incompatible with the intended logic of the system as an insurance. Members of this coalition also argued in favor of measures that would directly influence demographic change, for example by producing more children who could pay into the pension system or through immigration.

The other large advocacy coalition can be found on the left of the previous coalition in Figure 2. It contains many banks and insurance companies (in blue color), who had an incentive to encourage privatization of the pension system. Banks and insurance companies would sell the products that would take the place of public pensions if the system was privatized, such as life insurances, stock market investment, pension funds, or specialized products. This coalition also included employers' and industry associations, who also had an interest in cutting back the public pension system to save the employers' contribution to pensions and shift the burden onto the worker through private investment. On the left of this coalition, Figure 2 shows a sub-coalition, which promoted closely related ideas but encouraged different products, such as home ownership, and was hence somewhat separated from the larger coalition.

Figure 3 shows all concepts present in this policy debate in 2000, with their frequencies separated by agreement/disagreement qualifier. This kind of diagram was first used in the context of discourse network analysis in Leifeld and Haunss (2012) to examine which concepts are most important and controversial. Here, we add to this goal by identifying not merely individual concepts, but sets of concepts that are jointly important in structuring the policy subsystem.

Many of the concepts toward the lower end of Figure 3 were more actively used in other years of the debate. As the year 2001 saw a partial privatization through the introduction of a multi-pillar pension system, the solution concept "Partial transition to a private capital cover system" was the most prominent policy belief. It was used almost exclusively in an affirmative way and mostly by members of the coalition that wanted to change the pension system, creating a common identity in this coalition.

More details on the case can be found in Leifeld (2013). Below, we apply the methods developed in this article to the discourse network shown in Figures 2 and 3 to partition the concepts shown in the barplot diagram into a backbone and a redundant set.

4 Results

Figure 1 on page 11 shows convergence diagnostics and other details of the combinatorial optimization using simulated annealing for the pension case. A penalty of $p = 7.5$ was employed for this optimization run. A relatively large number of $T = 50,000$ iterations

Figure 3: Concepts in the full network, ordered by prominence, with positive and negative qualifier, partitioned into backbone and redundant set.

was used, but shorter chains should approximate the solution well. Only one optimization run is shown here to illustrate the procedure.

Panel A shows the acceptance probability from Equation 7, second case, over the duration of the iterations. Due to the S-shaped temperature scalar, the acceptance probability declines in a very similar way as the temperature cools down. However, while the temperature is still high, there comes a point when the search space has been somewhat explored and backbone solutions with *many* concepts are attempted, around Iterations

8,500–9,000. Because these large solutions are so heavily penalized by the exponential decay weighting and spectral loss, the acceptances during this time go down to some extent despite the high temperature. The algorithm effectively finds a compromise between exploration and goal-directed optimization.

Panel C shows the penalized Euclidean spectral distance between the full network and the current backbone candidate in each iteration. The distances are plotted on a log scale to make smaller distances more easily visible. Indeed, the spectral loss is high during those iterations in which large solutions are explored. Toward the final iterations, the algorithm hones in on good solutions with a small loss.

Panel D displays the number of concepts in the candidate backbone set of each iteration. Indeed, the iterations with the greatest spectral loss are characterized by very large combinations of concepts. Further toward the end, the number of concepts declines before it goes back up and stabilizes at around seven.

In Panel E, the number of concepts in B, the globally best backbone candidate up to the respective iteration, is plotted. The size of the best solution jumps up and down several times, but there is no further change in roughly the last 10,000 iterations.

Finally, Panel F serves as a convergence criterion. It shows the number of accepted candidate backbones among the last 100 iterations, similar to a moving average. The curve follows the temperature and indicates barely any acceptances in the last quintile of the iterations.

Figure 4 shows the resulting networks based on the full set of concepts (first column), the backbone set of concepts (second column), and the redundant set of concepts (third column), as identified by the simulated annealing algorithm. Nodes are colored according to coalition membership. Coalitions were identified using a clustering algorithm called partitioning around medoids (PAM, Kaufman and Rousseeuw 1990), which was used for its simplicity and consistently good results with high modularity scores when compared to a number of other community detection and clustering algorithms applied to this dataset. PAM was applied to the full network, and cluster memberships based on the full network

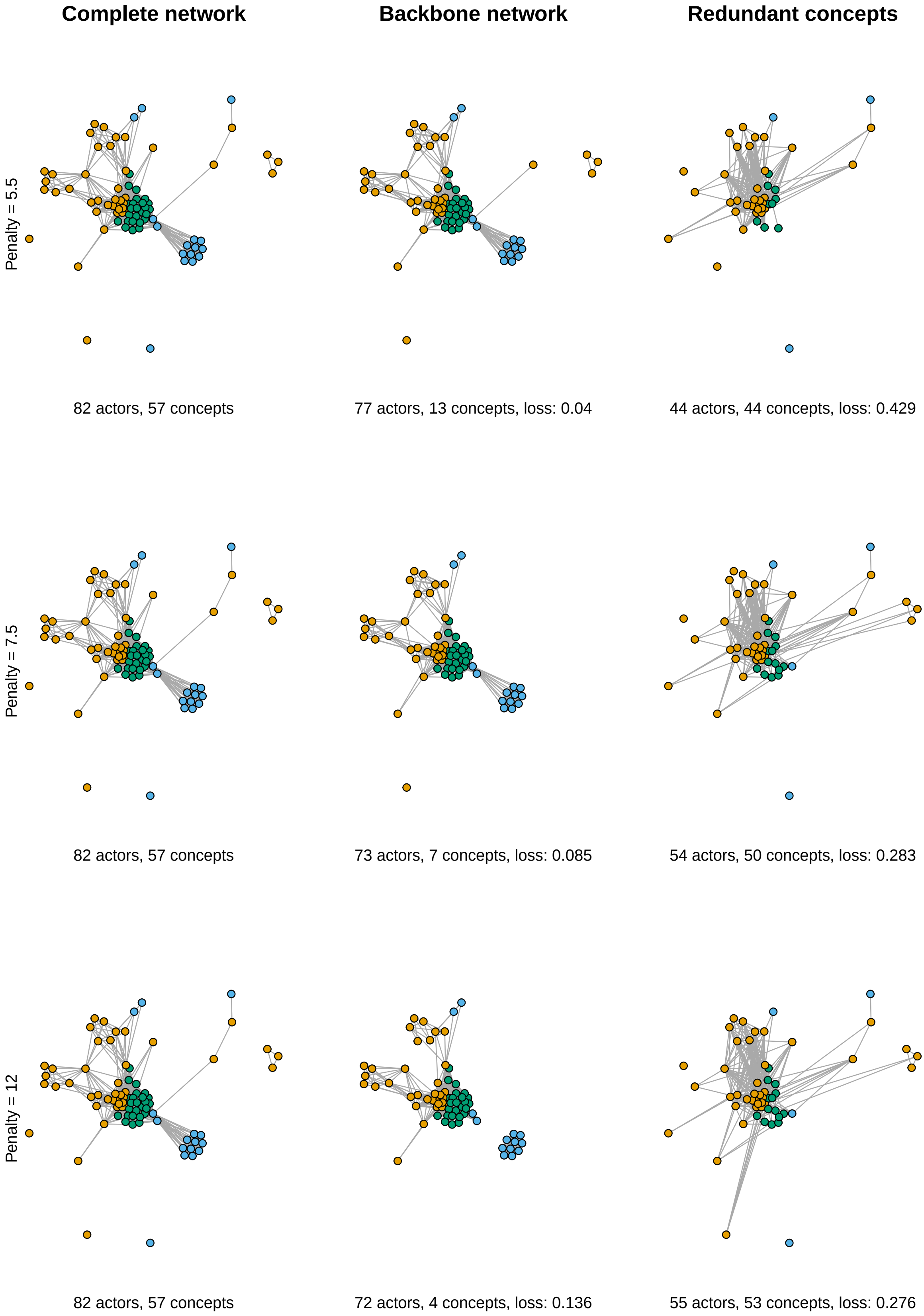

Figure 4: Original network, backbone network, and redundant concept network side by side with different penalties, leading to different backbone set sizes.

are shown in all three networks. Node coordinates are based on the full network and fixed in each row of the figure for comparability.

The second row corresponds to the optimization run shown in Figure 1 with a penalty of $p = 7.5$ and yields a backbone of seven concepts, while 50 concepts were discarded into the redundant set. The network on the left shows the full network with all 57 concepts, similar to Figure 2, but using the subtract method from Equation 1 while Figure 2 was based only on normalized actor congruence, leading to some minor differences.

The backbone network with only seven concepts looks remarkably similar, both with regard to visual details and the unpenalized spectral loss, vis-a-vis the full network, of only 0.085. The backbone captures all key features of the policy subsystem, including the location and most members of each of the three clusters. The subdivisions inside the orange cluster (the status-quo-oriented policy community) are retained in the backbone. Mainly some peripheral actor vertices are lost because their beliefs are no longer represented. The network based on the redundant set of the remaining 50 concepts duplicates much of the information that is already contained in the backbone, and it adds some peripheral vertices who do not seem to add much information in terms of coalition structure. At 0.283, the spectral loss of the redundant set compared to the full network is much higher than the loss of the backbone, indicating that it is much more dissimilar structurally. The 50 concepts from the redundant set can be removed or recoded without much loss of information; they are *redundant*, at least if we consider only this particular year of the debate.

Which seven concepts made it into the backbone set? The color coding in Figure 3 shows that some of the most frequently used concepts are part of the backbone, including the "partial transition to a private capital cover system," which was indeed the most central element of the debate in this particular year. It was the most frequently used concept and was mostly located in the coalition of banks, insurance companies, and employers' associations, holding this coalition together like "glue."

The other coalition was held together by a mix of socially minded concepts, such as "minimum pension," an opposition to pension cuts and curtailing early retirement and to increasing employees' share of contributions, fertility-oriented concepts like linking the contribution or pension level to the number of children, and suggesting cuts to the existing pension system to avert larger changes toward privatization, for example in favor of curtailing early retirement. There was some disagreement among members of this coalition, leading to sub-coalitions. The subcoalitions among the orange nodes in the full network in Figure 4 can be explained by the diversity of partially contradictory concepts prevalent in this coalition. This is supported by the presence of both positive and negative mentions of these policy beliefs in Figure 3.

The annex to the privatization coalition in the form of home ownership can also be found in the backbone set, despite its relatively low number of statements, because this concept is necessary to maintain the coalition structure of the discourse network. While some of the frequently used concepts have made it into the backbone set, some of the highly used concepts, such as "increase contributions" or "occupational pensions" are in fact redundant and do not add much to the coalition structure of the policy subsystem. The majority of concepts that were infrequently used were weak predictors of network structure and ended up in the redundant set in this particular year of the debate.

These findings are consequential for both the annotation scheme and substantive insights. We can see around which concepts the debate was mostly structured. It becomes much clearer what the major conflicts are about and how coalitions are structured. The model is a helpful way to analyze advocacy coalitions and their belief systems. The redundant set also informs our coding by suggesting concepts to recode. If the whole debate were structured like in this particular year, some of the concepts should likely be dropped or recoded. For example, concepts related to specific ways of removing non-insurance elements from the pension system, such as "cut back invalidity or widows' pensions," opposition to "subsidies from the national budget," "earnings- and effort-based pensions," "include civil servants in the pension system," and a few other concepts are likely related to a super-concept that is part of the same approach as saving the pay-as-you-go pension system by curtailing early retirement as ways to cut back on expenses. However, a more detailed analysis at varying penalty levels can give a more differentiated picture and clarify how concepts are structurally nested in other concepts. We recommend an iterative approach with different parameter settings to develop an understanding of how concepts form a belief system.

In Figure 4, two additional parameter settings for the penalty are shown. The first row uses a lower penalty of $p = 5.5$ for comparison and yields a larger backbone set with 13 concepts. The spectral loss is slightly smaller, but visually hardly noticeable. The spectral loss of the set of 44 redundant concepts meanwhile increases because some of the more useful concepts have been moved into the backbone, rendering the redundant set even more redundant.

The last row in Figure 4 uses a higher penalty of $p = 12$ and yields a smaller backbone of only four concepts. This small backbone set is remarkably powerful as a model of the coalition structure in the policy subsystem and still contains the major faultlines, even makes divisions within the status-quo-oriented coalition of orange vertices more visible.

One could additionally use a very low penalty to start at the other extreme and first identify only the most redundant concepts to flag them for recoding and then progress iteratively to larger penalties.

5 Conclusion

In this contribution, we have presented a way to identify the backbone concept set of a discourse network and, at the same time, the set of redundant concepts that do not explain the community structure of the network. This is useful both from the perspective of improving the annotation scheme as a model of the data and for substantive analysis of empirical policy subsystems, advocacy coalition structure, and belief systems.

There are several future directions for this research. First, we encourage applications of this approach to empirical applications of discourse network analysis to policy subsystems and other kinds of discourse networks. Several dozen applications of discourse network analysis have been published to date, and the methods presented here should be incorporated into the toolbox of discourse network analysts.

Second, the work presented here opens up interesting ways to analyze discourse networks further. By applying this methodology to all possible penalty levels, it should be possible to create a dendrogram of how concepts cluster together in structural terms, i. e., how they are nested in other concepts in terms of how they structure a discourse network. This will greatly facilitate the systematic description of belief systems and will fill a real void in the current literature on belief systems.

Third, we will make these methods available as part of the DNA software to support their application and adoption.

Fourth, the methods presented here may be applicable not only to discourse networks and policymaking, but potentially to a larger set of two-mode networks, or bipartite graphs, in other application areas of network science. Many bipartite graphs in which the second vertex mode does not possess agency could be eligible for these methods. For example, one could use the approach to study which policy forums in an ecology of games (Lubell et al. 2010) are decisive or redundant for a complex, decentralized governance network, which social events or parties knit the social fabric of Florentine families (Padgett and Ansell 1993), or finding key researchers and publications that hold together a scientific network (Leifeld et al. 2017).

Fifth, a model selection criterion should be developed to automatically select the best penalty level. The elbow criterion and silhouette coefficient in cluster analysis could inspire such a method.

References

Baur, M., Benkert, M., Brandes, U., Cornelsen, S., Gaertler, M., Köpf, B., Lerner, J., and Wagner, D. (2001). Visone software for visual social network analysis. In *International Symposium on Graph Drawing*, pages 463–464. Springer.

Béland, D. (2005). Ideas and social policy: An institutionalist perspective. *Social Policy & Administration*, 39(1):1–18.

Fisher, D. R., Waggle, J., and Leifeld, P. (2013). Where does political polarization come from? locating polarization within the US climate change debate. *American Behavioral Scientist*, 57(1):70–92.

Gao, X., Xiao, B., Tao, D., and Li, X. (2010). A survey of graph edit distance. *Pattern Analysis and Applications*, 13(1):113–129.

Girvan, M. and Newman, M. E. J. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826.

Hajer, M. A. (1995). *The Politics of Environmental Discourse: Ecological Modernization and the Policy Process*. Oxford University Press, Oxford.

Jenkins-Smith, H. C. and Sabatier, P. A. (1994). Evaluating the advocacy coalition framework. *Journal of Public Policy*, 14(2):175–203.

Jenkins-Smith, H. C., St. Clair, G. K., and Woods, B. (1991). Explaining change in policy subsystems: Analysis of coalition stability and defection over time. *American Journal of Political Science*, 35(4):851–880.

Kaplan, S. (2008). Framing contests: Strategy making under uncertainty. *Organization Science*, 19(5):729–752.

Kaufman, L. and Rousseeuw, P. J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Wiley, Hoboken, NJ.

Kukkonen, A., Ylä-Anttila, T., and Broadbent, J. (2017). Advocacy coalitions, beliefs and climate change policy in the United States. *Public Administration*, 95(3):713–729.

Leifeld, P. (2013). Reconceptualizing major policy change in the advocacy coalition framework: A discourse network analysis of German pension politics. *Policy Studies Journal*, 41(1):169–198.

Leifeld, P. (2017). Discourse network analysis: Policy debates as dynamic networks. In Victor, J. N., Montgomery, A. H., and Lubell, M. N., editors, *The Oxford Handbook of Political Networks*, pages 301–326. Oxford University Press, Oxford.

Leifeld, P. and Haunss, S. (2012). Political discourse networks and the conflict over software patents in Europe. *European Journal of Political Research*, 51(3):382–409.

Leifeld, P., Henrichsen, T., Buckton, C., Fergie, G., and Hilton, S. (2022). Belief system alignment and cross-sectoral advocacy efforts in policy debates. *Journal of European Public Policy*, 29(8):1225–1248.

Leifeld, P., Wankmüller, S., Berger, V. T. Z., Ingold, K., and Steiner, C. (2017). Collaboration patterns in the German political science co-authorship network. *PloS ONE*, 12(4):e0174671.

Lubell, M., Henry, A. D., and McCoy, M. (2010). Collaborative institutions in an ecology of games. *American Journal of Political Science*, 54(2):287–300.

Newman, M. E. J. (2006). Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582.

Oliver, K. and Cairney, P. (2019). The dos and don'ts of influencing policy: A systematic review of advice to academics. *Palgrave Communications*, 5(1):1–11.

Padgett, J. F. and Ansell, C. K. (1993). Robust action and the rise of the medici, 1400-1434. *American Journal of Sociology*, 98(6):1259–1319.

Shore, J. and Lubin, B. (2015). Spectral goodness of fit for network models. *Social Networks*, 43:16–27.

Strom, K. (1990). A behavioral theory of competitive political parties. *American Journal of Political Science*, pages 565–598.

Weible, C. M., Ingold, K., Nohrstedt, D., Henry, A. D., and Jenkins-Smith, H. C. (2020). Sharpening advocacy coalitions. *Policy Studies Journal*, 48(4):1054–1081.

Appendix: R Code for Simulated Annealing

```r
# Load libraries
library("rDNA")
library("progress")
library("ggplot2")

# Seed seed for replicability
set.seed(123456)

# Initialize DNA
dna_init("dna-2.0-beta25.jar")

# Get pension data
conn <- dna_connection("Rente.dna")

# Load list of concepts
concepts <- dna_getAttributes(conn, variable = "concept")

# Filter concepts to only those used in 2000
nw_concepts <- dna_network(conn,
                           start.date = "01.01.2000",
                           stop.date = "31.12.2000")

concepts <- concepts[concepts$value %in% colnames(nw_concepts),]


#### Dataprep for spectral distance calculation of full network ####

nw_org <- dna_network(conn,
                      networkType = "onemode",
                      qualifierAggregation = "subtract",
                      normalization = "average",
                      start.date = "01.01.2000",
                      stop.date = "31.12.2000",
                      verbose = FALSE)

# Set all negative edge weights to zero
nw_org[nw_org < 0] <- 0

# Calculate actor degrees
deg <- rowSums(nw_org)

# Create copy of network matrix and set all values to zero
```

```
nw_deg <- nw_org
nw_deg[,] <- 0

# Insert actor degrees into matrix diagonal
diag(nw_deg) <- deg

# Compute Laplacian matrix
laplac <- nw_deg - nw_org

# Compute eigenvalues of Laplacian matrix
nw_eigen <- eigen(laplac)

# Normalize eigenvalues
nw_eigen_norm <- nw_eigen$values/sum(nw_eigen$values)


#### Dataprep for Simulated Annealing ####

# Set number of iterations
niterations <- 50000

# Set starting temperature
temperature <- 1 - plogis(seq(-5, 7, length.out = niterations))


### Create some data objects for monitoring

# Spectral distance history
esd_history <- numeric()

# Random starting values for best and current solution
esd_best <- esd_current <- 1

# Create vectors for candidate and best solutions
concepts_removed_best <- concepts_removed <- NULL

# Data.frame for acceptance history
acceptance_history <- data.frame(matrix(nrow = niterations, ncol = 2))
colnames(acceptance_history) <- c("acceptance", "iteration")

# Vector for number of best concept and general concept solutions
concepts_best <- concepts_general <- numeric()

# Set counter for number of acceptances
```

```r
counter <- rep(0, niterations)

pb <- progress_bar$new(
  format = "  Finished [:bar] :percent remaining: :eta",
  total = niterations, clear = FALSE, width = 60)


#### Simulated Annealing loop ####

for (i in 1:niterations) {

  ## Three different sampling options for every iteration:
  # 1.) remove a concept, 2.) add a concept, 3.) replace a concept

  if (i > 1) {
    if (length(concepts_removed) > 1) {
      # If current solution contains more than one concept, all three options
      # are possible
      sample_x <- sample(1:3, 1)
    } else {
      # If not, removing a concept is excluded from choice of options
      sample_x <- sample(2:3, 1)
    }
  } else {
    sample_x <- 2 # First iteration always adds a concept
  }

  # In case all but one concept are included in solution, the sampling option
  # is set to 1 (remove a concept)
  if (length(concepts_removed) == (nrow(concepts) - 1)) {
    sample_x <- 1
  }

  # Exclude already included concepts from list of concept options
  concepts_x <- concepts$value[!(concepts$value %in% concepts_removed)]

  if (sample_x == 1) {
    # Remove one concept
    concepts_removed2 <- concepts_removed[-(sample(1:length(concepts_removed), 1))]
  } else if (sample_x == 2) {
    # Add one concept
    concepts_removed2 <- c(concepts_removed, concepts_x[sample(1:length(concepts_x), 1)
    ])
  } else if (sample_x == 3) {
```

```
    # Replace one concept by another one
    concepts_removed2 <- concepts_removed[-(sample(1:length(concepts_removed), 1))]
    concepts_removed2 <- c(concepts_removed2, concepts_x[sample(1:length(concepts_x), 1)
    ])
  }

  # Extract network matrix (excluding removed concepts)
  nw <- dna_network(conn,
                    networkType = "onemode",
                    qualifierAggregation = "subtract",
                    normalization = "average",
                    start.date = "01.01.2000",
                    stop.date = "31.12.2000",
                    excludeValues = list("concept" = concepts_removed2),
                    invertValues = TRUE,
                    verbose = FALSE)

  # Spectral distance cannot be computed for networks with no ties
  # (i.e., single node or isolate networks). In this case, skip to the next
  # iteration
  if (max(nw) == 0) {
    # Save for monitoring
    esd_history <- c(esd_history, esd_current)
    concepts_general <- c(concepts_general, length(concepts_removed2))
    concepts_best <- c(concepts_best, length(concepts_removed_best))

    pb$tick()

    next
  }

  # Set all negative edge weights to zero
  nw[nw < 0] <- 0

  # Spectral distance requires reduced nw and complete nw to have identical
  # nodes
  if (!(identical(rownames(nw), rownames(nw_org)))) {

    # Extract missing actors
    actors_miss <- rownames(nw_org)[!(rownames(nw_org) %in% rownames(nw))]

    # Bind missing rows
    nw_miss_row <- matrix(data = 0, nrow = length(actors_miss), ncol = ncol(nw))
    rownames(nw_miss_row) <- actors_miss
```

```r
  nw <- rbind(nw, nw_miss_row)

  # Bind missing columns
  nw_miss_col <- matrix(data = 0, ncol = length(actors_miss), nrow = nrow(nw))
  colnames(nw_miss_col) <- actors_miss
  nw <- cbind(nw, nw_miss_col)

  # Order row- and colnames of reduced nw according to complete nw
  row_index <- numeric(length = nrow(nw))
  for (xx in 1:length(row_index)) {
    row_index[xx] <- which(rownames(nw_org)[xx] == rownames(nw))
  }

  nw <- nw[row_index,row_index]
}

### Calculate spectral distance of reduced nw ###

# Calculate actor degrees
deg <- rowSums(nw)

# Create copy of network matrix and set all values to zero
nw_deg <- nw
nw_deg[,] <- 0

# Insert actor degrees into the matrix diagonal
diag(nw_deg) <- deg

# Create laplacian matrix
laplac <- nw_deg - nw

# Compute eigenvalues
nw_eigen <- eigen(laplac)

# Normalize eigenvalues of reduced network
nw_eigen_norm2 <- nw_eigen$values/sum(nw_eigen$values)

# Calculate eigenvalue differences between complete and reduced nw
nw_euclid <- abs(nw_eigen_norm - nw_eigen_norm2)

# Calculate euclidean spectral distance
esd <- sqrt(sum(nw_euclid)^2)

# Calculate number of concepts remaining
```

```
concepts_remaining <- nrow(concepts) - length(concepts_removed2)

# Penalize spectral distance using an exponential decay
esd_candidate <- esd / (7.5 * (exp(-(7.5 * (length(concepts_removed2) / nrow(concepts)
  )))))

# Update if candidate is better than current score
if (esd_candidate <= esd_current) {
  esd_current <- esd_candidate

  # Update concept solution
  concepts_removed <- concepts_removed2

  # Set counter for number of acceptances
  counter[i] <- 1

  # Update global solution if candidate spectral distance is better
  if (esd_current <= esd_best) {
    esd_best <- esd_current
    concepts_removed_best <- concepts_removed
  }
} else {
  # Calculate difference between candidate and current spectral distance
  esd_diff <- esd_candidate - esd_current

  # Calculate acceptance
  acceptance <- exp(-esd_diff) * temperature[i]

  # Add acceptance value for monitoring
  acceptance_history$acceptance[i] <- acceptance
  acceptance_history$iteration[i] <- i

  # If acceptance higher than random number between 0 and 1 then update
  if (runif(1) < acceptance) {
    esd_current <- esd_candidate
    concepts_removed <- concepts_removed2

    # Set counter for number of acceptances
    counter[i] <- 1
  }
}

## Save for monitoring

```

```r
  # Spectral distance history
  esd_history <- c(esd_history, esd_current)

  # General concept solutions
  concepts_general <- c(concepts_general, length(concepts_removed2))

  # Best concept solutions
  concepts_best <- c(concepts_best, length(concepts_removed_best))

  pb$tick()

}
```