Automatically predicting coalition formation in the Dutch House of Representatives

M. E. (Maarten) San Giorgi

A thesis for the Master Artificial Intelligence

First supervisor
dr. T. (Tim) Baarslag
Second supervisor
prof. dr. M. M. (Mehdi) Dastani

Department of Information and Computing Sciences
Utrecht University
The Netherlands
July 8, 2022

Abstract In this work, a model is presented to predict the outcome in the coalition formation in the Dutch House of Representatives. Using coalition formation theories and utility theory, a model is created which outputs a coalition, based on political party preferences and a seat distribution. We measure the results of the model and its robustness, by comparing them with actual coalition formations from the past. Furthermore, an implementation is presented in the form of a user-friendly web-application which interactively calculates coalitions based on different variables and data sets, and gives insight in the process.

Contents

1 Introduction

1.1 Background

In the Netherlands, formations of the government can take months. The last formations in 2017 and 2021 took respectively 225 and 278 days. During this time, the former cabinet (caretaker cabinet) was still in charge. This means only minor or urgent issues can be decided upon. Meanwhile, other possibly important and conflicting topics are set aside, until a new cabinet is formed. It has been argued before that the process of forming a cabinet could be improved in multiple ways, like changing the role of mandators or removing power from a king or queen [1]. These long formation processes can cause incomprehension and even frustration among voters, because it can be hard to understand how their votes lead to a certain cabinet. A coalition prediction model could speed up and give insight to the process of forming a government.

However, the duration of a coalition formation is caused by many factors. In [2], the authors looked at multiple variables which could influence this duration (such as previous defeat or number of parties). Furthermore, their research shows that Dutch coalition formations are one of the most time consuming processes of Europe. We cannot expect that a prediction model can cover all factors influencing the formation of a government and thus fully overcome long formation processes.

In the last 20 years, the participating parties in elections have more than doubled (from 16 to 37). The amount of parties that got at least one seat in the House of Representatives has gone from 10 to 17. One can reason that more parties would lead to harder coalition formations. The amount of possible coalitions to consider and investigate becomes larger, which makes the time to form one even longer.

Not only is it interesting to fasten the process of coalition formation, but ideally an optimal coalition is found with an optimal coalition agreement. A coalition is optimal when the ideologies of the parties in it lie as close as possible to each other and a coalition agreement can be made which satisfies the parties in it the most. With a large spectrum of parties and ideologies, an optimal coalition is hard to determine. A close approximation of an optimal coalition is needed to avoid early cabinet terminations. In [3], the authors show that the most significant reason why parties terminate cabinets is when they expect policy gains in a succeeding cabinet. It would thus be valuable to have insight in a coalition formation if a certain party could gain more profit in forming another coalition.

Not only could a prediction model have influence on the formation process after an election, but it could have an influence on the elections as well. Research has shown that election polls already have a certain influence on voting behaviour in the Netherlands [4]. This influence could grow when a coalition prediction model could show which coalitions are likely, given the election polls. In the past, election polls already have been the cause of tactical voting [5, 6], where voters do not vote for their most preferred political party. Reasons for tactical voting can be disliking other political parties or having a 'waste vote' according to polls, because your most preferred party probably won't be in the cabinet. Recently (June 2022), two parties (PvdA and GroenLinks) in the Dutch politics decided to form one front in the Senate after next elections. This implies that not only voters, but even parties are willing to put aside differences to gain more influence in the Dutch parliament.

In order to give a considered and well-thought vote, voters need as much information as possible about political parties and their differences. A double-axis political spectrum is a common way to show the differences between parties. However, the axes of such a political graph do differ per country and change over time [7, 8, 9]. In the Netherlands, Krouwel designed a political spectrum for the Election Compass [10], which is now commonly used throughout the country. The horizontal axis is designed to indicate the difference in money-based themes like economics, healthcare and taxes. The vertical axis shows how progressive or conservative political parties are on themes like climate control and migration. The placement of political parties for the elections of 2021 can be seen in figure 1. Such a graph already gives good insight in showing the proximity between political parties, yet other axes might represent this proximity better. A revised version of the political spectrum could thus give even more accurate information for the voters.

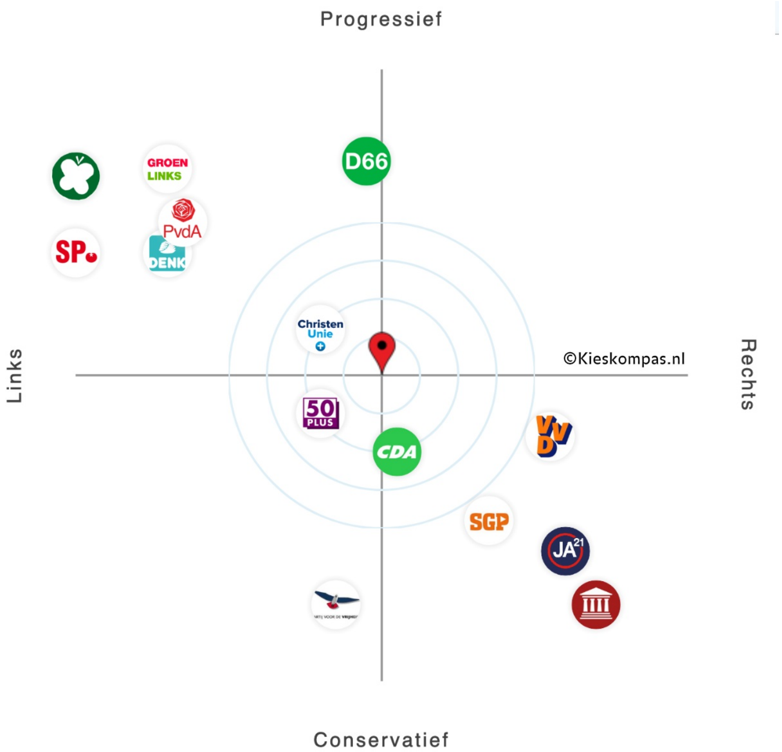

Figure 1: Political spectrum of the Dutch political parties, based on the elections for the House of Representatives of 2021 [11]

Studies in coalition formation descend from game theory and started with simple concepts, such as the three-person games [12, 13]. Expanding these theories, with notions of coalition structures, which formalize the notion of the core and stability, the field of coalition formation got wider. These theories were tested in existing political systems to see if they worked (more on this in section 2). In this work, we take a new direction in which we include concepts as ideologies and utility.

The goal of this research is to develop a model which predicts the best feasible coalitions and gives insight why these are the best coalitions. These predictions are based on a given seat distribution and the ideologies of political parties. A given seat distribution can be based on the outcome of elections, polls for future elections or (theoretical) experimental data. The ideologies of parties can be obtained by looking at promises before elections, overall party programs or actual voting behaviour in the House of Representatives. In this study, we only look at the Dutch political system, to narrow the scope of the research.

We start with formulating the research questions in the next section. In section 2 we briefly discuss previous work in the field of coalition formation and political coalitions. By doing this, we get a good of what is already done in this field, which concepts we can use and what problems other scientists found. A model for coalition prediction is presented in section 3, which we will use in the rest of this work. In section 4 we analyze the results from the experiments we perform on the model. Then, we show the implementation of the model in section 5. We conclude this work with a discussion, where we will show the limitations of this research and discuss possible future work.

1.2 Research questions

Based on the sketched problems from the previous section, the main research question is formulated as:

> *How can we predict the outcome in the coalition formation of the Dutch House of Representatives?*

To answer this question, we need to divide this problem into multiple side questions.

1.2.1 How to model coalition formation

To predict the outcome of a coalition formation, we first need to create a model which outputs one or more coalitions, based on a certain input. We thus can state the first side question:

> *How do we model the coalition formation in the Dutch House of Representatives?*

Modelling coalition formation is far from trivial. The main two factors in such a model are ideology and size [14]. Members in a coalition formation would ideally want to form a coalition with other members which are similar in their ideologies. However, a certain size must be reached to create power for decision control. Therefore, we must find a method to compare ideologies based on party preferences and find a balance between ideology and size.

1.2.2 How to measure robustness

To have a good prediction model, it is important that the outcomes are consistently accurate. This means that when we change chunks of data or leave pieces of data out the predictions of the model won't change drastically. In computer science, this property is called robustness [15]. Therefore we state the second side question:

> *How do we measure the robustness of the model?*

The challenge here lies in the way we measure robustness and in what manner changes in data and settings should give different outcomes or not. We elaborate more on this in section 4. If we can prove that our model is robust, we can give users of the model a certainty for the quality of the results.

1.2.3 How to provide insight

To better understand *why* the model generates certain outcomes, we need to look for a way to make the formation process more transparent. This way, we can make the coalition prediction model useful for third parties, like voters or even politicians. With the use of polls and current ideologies of parties, the model can show why certain coalitions are more likely than others. Thus, we formulate the last side question:

How can we provide insight in the feasibility of different coalitions?

When we can show why certain parties could work well together or why parties are better of in other coalitions, the model will have greater use for voters. Insight in feasible coalitions can then influence voting behaviour (e.g. tactical voting). Furthermore, it can make voters better understand real coalition formations and why the Dutch formations require such a long period of time.

2 Related work

In this section we discuss existing works related to our research. We look into coalition theories, utility theories and coalition formation in politics. This helps us in finding existing concepts to use in our work and in where we need our own contributions in answering the research questions.

2.1 Coalition theories

The field of coalition formation finds its origin in game theory [12]. Caplow [16] introduced a theory based on a game of the triad, where six types of situations were described in which a coalition could be formed. These situations were based on the power of members, which is similar to the problem in this research (the power of members can be translated to the amount of seats of a political party in the House of Representatives). Gamson [17] extended this topic even more and brought up a theory including a minimal winning and cheapest winning coalition, similar to the theories of Caplow and Riker [13]. A winning coalition has enough power to control a decision. A minimal winning coalition is one where removing any member will make the coalition not winning anymore, whereas a cheapest winning coalition has an extra restriction that the amount of power must lie closest to the decision point. Such theories form a solid foundation on how to model our coalition formation, yet only size of coalitions matters in these theories and not ideologies or other preferences.

In 1970, Axelrod [18] proposed the notion of connected coalitions, which are ideologically connected coalitions. This idea was extended further from a uni-dimensional ideology to an n-dimensional ideology [19, 20]. Combining these ideas with the minimum winning coalitions from [17], the concept of minimum connected winning coalitions was created. This concept already offers a nice basic solution for the problems of size and ideology described in section 1.2.1. Other methods show that members in a coalition formation can be connected as well with preference models for other members in the formation [21], instead of preferences for ideologies. A political party might agree with many ideologies of another party, but dislike the party itself due to other factors (e.g. VVD and FVD).

According to Kelley [22], there are a lot more factors to hold into account when creating a model for coalition formation. Interesting factors for our research are

- the size and divisibility of the reward from a winning coalition (i.e. which party gains what from a coalition)
- the size of a coalition to win (is a majority needed or can we create a minority cabinet?)

- ideological similarities and differences among the participants (i.e. connected coalitions)
- the maximizing behaviour of the participants (how can parties gain as much for themselves in a coalition. We will see more on this when we discuss the stability of coalitions)

These factors are similar to the problems described in section 1.2.1 and we can thus use concepts when we create our model.

In further research, authors did not try to find only one coalition, but multiple coalitions for a set of participants. These coalitions together form a coalition structure (CS) [23, 24]. Rahwan et al. [25] created a survey, in which they discussed different methods for this problem in different settings, like optimization and reward division. Elkind et al. [26] extended the formation of coalition structures, by searching for non-empty CS-cores (based on the original cores as in [27, 28]) in weighted voting games. To define how certain coalition structures are better than others, Davis et al. [29] did a study about pay-off configurations and objections against other players in a cooperative game. In this work, we do not go further into coalition structures, because this concept is not relevant for the formation of one coalition in the House of Representatives (the coalition structure always exists of two coalitions, i.e. the ruling coalition and the opposition).

Another concept in coalition formation is that of stable coalitions [25, 30]. A coalition is stable when no member prefers another coalition. Brams et al. [31] present two processes to find (semi-) stable coalitions based on preference profiles of members. Stability offers a concept to give insight in coalition formations. It shows whether political parties might want to leave one coalition for another. This might come back in a future version of the implementation, as described in section 5, but we do not focus on this subfield.

Because we want to create a model which gives insight as well, we need to look into prediction models. In the field of Artificial Intelligence and Computing Sciences, prediction models are not uncommon. Using machine learning techniques with large amounts of data, prediction models are already widely implemented [32]. The disadvantage of using machine learning models is that the process between the data input and the prediction output is extremely hard to comprehend, due to the many layers of the model. To get insight in the prediction process, another method must be found. Therefore we look into utility theory.

2.2 Utility theory

A common way to express preferences (or ideologies) is with utility functions and values [33, 34, 35, 36] (similar to payoff vectors in [29]). Using such utility functions, we can search for parties with similar ideologies. As described in [35], utility is that which represents preferences. For example, a person can have preferences for the concept of *fruit*, where this person really likes apples and hates bananas. Utility can be ordinal [37], which represents these preferences as $apple > banana$ in a set of fruits. The preferences in types of fruit can also be represented with a number. For example, we give the apple a value of 0.9 and the banana a value of 0.3, where we still see that the apple is more favored than the banana. By doing this, we make the utility cardinal [38], i.e. represented by a numerical value.

Preferences of political parties can be modeled the same way. Instead of the concept of *fruit* we can use a political statement, like *investing more money in climate control*. Apples and bananas are replaced by other concepts, like ‘agree’ or ‘disagree’. When we give these concepts a value, we can use them in a model with utility functions.

Because of the fact that there is more than 1 ideology in politics, we can use the concept of multi-attribute utility theory (MAUT) [39, 40]. This helps us in obtaining a single utility value for a set of ideologies. For example, when we have categories *legalize drugs* and *investing more money in climate control*, a person can have utility values 0.3 and 0.5 for the individual variables of ’neutral’ and ’disagree’. We can combine these individual values into 1 value. One way to do this is by taking the average of the two. We can also add weights to the individual values, to make one more important than another, as in [41]. The formula to calculate such a utility value u for a set of individual variables X in certain categories would then be:

$$u(X) = \sum_{x \in X} \frac{1}{|X|} w(x) u(x)$$

Where, $w(x)$ is a weight for the category of x and $u(x)$ is the utility value of a single variable, like ’disagree’. MAUT offers a way to search for connected coalitions in a n-dimensional ideology space, as described in the previous section. The challenge here lies in how to translate the political ideologies to numerical values.

An alternative for MAUT is analytical hierarchy process (AHP) [42], which is another decision making method for groups. However, this method is more dependent on the weights and hierarchy of elements in the analysis. MAUT is a better method when we don’t know the preferences of parties exactly for sure. Because we still need to find a way to translate political ideologies to numerical values, we don’t know for sure if we represent these ideologies correctly. Furthermore, with a large space of alternatives (e.g. possible coalition agreements), MAUT is seen as a more suiting method [43].

2.3 Coalition formation in politics

One of multiple researches in coalition formation for political situations has been done by Dhillion [44]. She made a difference between pre-electoral and post-electoral coalition formations, and studies how and why political parties form coalitions. In this work, pre-electoral formations are irrelevant, but theories for post-electoral situations can help in modelling coalition formation. A more specific research was done by Steunenberg [45], who studied coalition formations in municipalities in the Netherlands. This empirical study was based on former local Dutch elections and shows that the dominated minimum number principle [12] has more effect in Dutch coalition formations than looking at political preferences. This means that the biggest party in a winning coalition cannot be changed with another party to keep the coalition winning and the coalition consists of a minimum number of parties needed to be a winning coalition. However, this research dates from 1992 and might be outdated, due to change in politics and culture. Furthermore, the author only looks at ideologies of parties in one dimension, while there are way more ideologies in which parties can differ (e.g. climate and immigration, instead of only economics).

On a higher level, coalition formation theories have been tested in European contexts [14, 19, 46, 47]. Practice showed the different importance of ideology and size in forming coalitions in parliaments. According to the results, the coalition in the Dutch parliament is more based on political ideologies rather than on the size of the coalition. Some coalition theories (e.g. Leiserson's [48]) show better results on existing parliaments than other theories. The main difference between these studies and our research is that instead of predicting a coalition in a parliament, the authors tested existing coalition theories on already formed coalitions. Based on these researches we can defend the choices we make in what manner we model size and ideology in the formation of a coalition in the Dutch House of Representatives.

In [14], multiple coalition theories are discussed from Taylor and Laver, Browne and de Swaan. According to the theory of Browne, the property of *minimum parties* is most important in coalition formations. This means that coalitions should have as few parties in it as possible. The theories of Taylor-Laver and de Swaan implied that the property of minimum connected winning coalitions should be considered most important in formations, where there can't be more parties in a coalition than the necessary amount to win. Furthermore, they already exclude parties from a coalition if their size is already below a certain threshold (1% to 2,5%). We can use these theories to see where our model stands between these existing works.

3 Coalition prediction model

In this section, we start working out the model of predicting the best coalition. Note that the model can use different methods to predict the best coalition and that we test the model with these different methods to analyse the difference in results. We start with making some assumptions to limit the factors weighing in coalition formations. After the assumptions, we set some definitions for the basis of the model. From section 3.3 until section 3.6 we walk through the model and the different ways to predict a coalition.

In section 5, we discuss the implementation of the model in the form of a web-application, called *De Coalitiewijzer*. This application can be used by voters, politicians and all others who are interested in the way coalitions are formed.

3.1 General assumptions

Because we cannot fully model real political coalition formations, we need to make some assumptions. To begin, we do not take political preferences of parties for other parties into account, as in [31]. One can reason that when parties have very different ideologies, their preference in forming a coalition with each other is probably low. However, recent coalition formations have shown that parties which lie relatively close to one another still can rule out each other (In the Dutch coalition formation of 2021, the VVD ruled out the FVD in a very early stage, while these parties lie quite close to each other). However, this is a very subjective matter and there exists no reliable data about personal preferences. Therefore, we don't take this into account.

Furthermore, we assume that preferences of parties for political issues are transparent and honest. This means that a political party would not lie about its preferences to gain a better position in the formation of the coalition.

Another assumption we make is that political parties always look for a majority in the House of Representatives. This means that a minority government will not be accepted. In the Netherlands, it happened twice that a minority cabinet was formed. Nevertheless, in both cases the cabinet had unofficial support from an extra party, which would support the government in most cases. A side note must be made that theory shows that minority governments are quite likely under certain circumstances, where the largest party is large and central in its ideologies [47], but we do not take this into account.

3.2 Theoretical background

Let $N = \{1, 2, ..., n\}$ be a set of political parties. Each party has a certain amount of power in the House of Representatives, which is defined as $w(i)$, the weight of party i. In practice, the power of a party is equal to the amount of seats a party holds (or is predicted to hold). Any subset of N is a coalition. $\mathcal{C} := \{ C \mid C \subseteq N \}$ is the powerset of political parties, where $\mathcal{C}$ contains all the possible coalitions that can be formed. Like a party, a coalition C has a weight as well, which is defined as $w(C) = \sum_{j \in C} w(j)$.

A *feasible winning coalition* is a coalition in $\mathcal{C}$, which has enough power to exceed a certain threshold to control a decision [17]. We define this as $w(C) \geq \theta$, where θ is the minimum proportion of power needed to control the decision.

Example 1. *Let* $N = \{1, 2, 3\}$ *be political parties with* $w(1) = 30$*,* $w(2) = 20$ *and* $w(3) = 10$*. The collection of possible coalitions is* $\mathcal{C} = \{\{1, 2, 3\}, \{1, 2\}, \{2, 3\}, \{1, 3\}, \{1\}, \{2\}, \{3\}, \emptyset\}$*. We define our threshold for a winning coalition as* $\theta = 40$*. A feasible winning coalition could thus be* $A = \{1, 3\}$ *with* $w(A) = 40$*. Figure 2 visualizes this situation.*

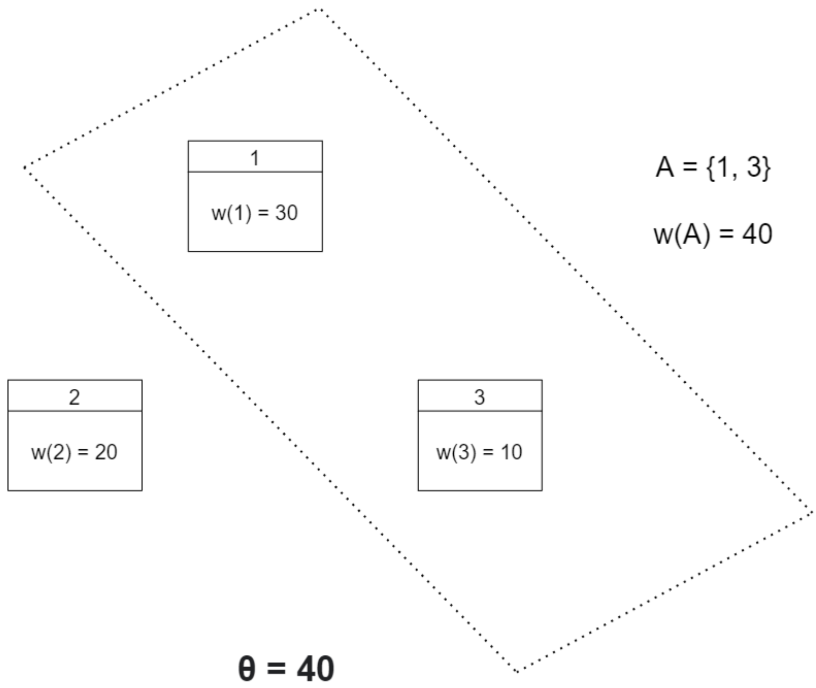

Figure 2: A feasible winning coalition $A = \{1, 3\}$

Let $I = \{1, 2, ..., i\}$ be a set of issues and V be a set of values $[0, 1] \in \mathbb{R}$. An issue corresponds with a political ideology about a certain subject (e.g. "The Netherlands need to build more windmills") and a value represents to what extent a party can agree of disagree with an issue. Let $\Omega = \{\omega_1, ..., \omega_n\}$ be the Cartesian product $V_1 \times ... \times V_{|I|}$, which contains all the possible outcomes. Each value $\omega_i \in \Omega$ represents a possible coalition agreement, based on a certain value of V for each issue in I. An outcome is an $|I|$-tuple $\omega_i = (v_1, v_2, ..., v_n)$, where the index of a value represents the corresponding issue in I.

Example 2. *Let* $I = \{1, 2\}$ *be a set of issues and* $V = \{0, 0.25, 0.5, 0.75, 1\}$ *be a set of values. The set of outcomes is* $\Omega = V \times V$*. A possible outcome in* Ω *could thus be* $\omega_i = (0.25, 0.5)$*. Figure 3 visualizes this situation.*

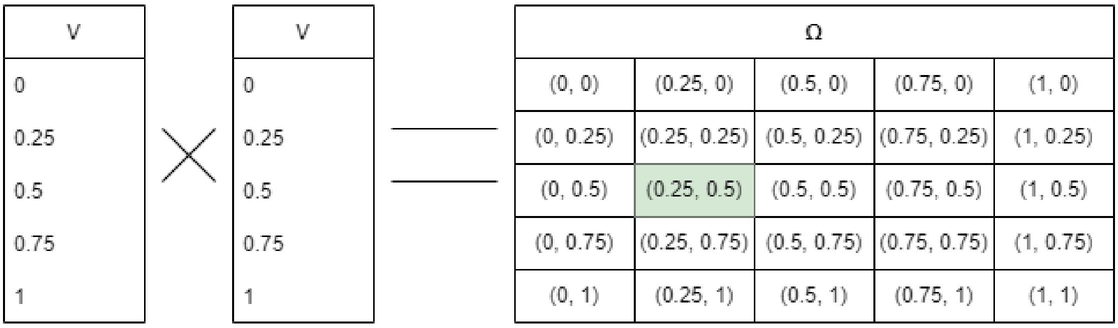

Figure 3: The Cartesian product $\Omega = V_1 \times ... \times V_{|I|}$ with a possible value of $\omega_i = (0.25, 0.5)$

3.3 Utility

To measure if one coalition is better than another, we use the concept of utility functions and values from section 2.2. Coalitions have utility values for outcomes in Ω, based on utility values of individual political parties for a certain outcome in Ω.

The utility of a coalition is conditional, where non-winning coalitions get a utility value of 0. Feasible winning coalition formations get a utility value $u_C(\Omega)$ based on all possible outcomes of Ω, as in [49, 50]. We propose methods to define $u_C(\Omega)$ in the next sections. First, we start at the bottom and discuss how we process the preferences of the political parties. These preferences can then lead to utility values for parties, which consequently lead to utility values for coalitions. A coalition C is better than coalition C' if and only if $u_C(\Omega) > u_{C'}(\Omega)$. The best possible coalition C^* would then be

$$C^* = \underset{C \in \mathcal{C}}{\operatorname{argmax}} \begin{cases} u_C(\Omega) & \text{if } w(C) \geq \theta \\ 0 & \text{otherwise} \end{cases}$$

3.4 Party preferences

3.4.1 Experimental data

To find out how we can best predict C^*, we will experiment on our proposed model, using different utility functions and different data (see section 3.7.3). The results from these experiments show which methods or data are best to use. For these experiments we use the data of Kieskompas (Election Compass) [11] and the data of Stemwijzer. These data sets contain issues where political parties have given their preference about, before an election. These preferences are based on initial promises that the parties make for their voters and could differ from what the parties would actually vote for in the House of Representatives.

Therefore, we also want to look at actual issues which have been discussed in the House of Representatives to see what parties actually have voted for. In this case, we can use the data of the Dutch Parliamentary Voting Dataset [51] (which is partly retrieved from the Dutch House of Representatives API). A disadvantage of this set is that in practice, parties of an existing coalition already have voted similarly for most of the issues. This causes a high similarity in issues for these parties, which would bias coalition formation based on this data. Furthermore, issues in this set tend to overlap a lot, because issues are frequently divided into multiple (overlapping) sub-issues. Therefore, we expect that the results from this data set aren't useful and thus we reject this data set.

The preferences of the political parties are defined in different ways per data set. For example, a party could have the preferences 'agree', 'neutral' and 'disagree' in the Stemwijzer set, while in the Kieskompas set the options 'fully agree' and 'fully disagree' are available as well. Figure 4 shows an example of a data entry in the Stemwijzer data set. In the section below, multiple possible translations are proposed to process these different data types to a general value set V.

```
{
    "id" : 2,
    "title" : "Gratis kinderopvang",
    "description" : "Kinderopvang moet gratis worden.",
    "eens" : ["CDA", "D66", "GroenLinks", "SP"],
    "geen_van_beide" : ["CU", "DENK"],
    "oneens" : ["VVD", "PVV", "SGP", "FVD", "JA21"]
}
```

Figure 4: Example entry of an issue in the Stemwijzer data set. Options are to agree, be neutral or to disagree. Note that some parties are left out in order to compress the figure.

To define the seat distribution $w(i)$ from section 3.2, we use actual data from elections and polls for next elections. Figure 5 shows an example of a seat distribution based on the elections of 2021.

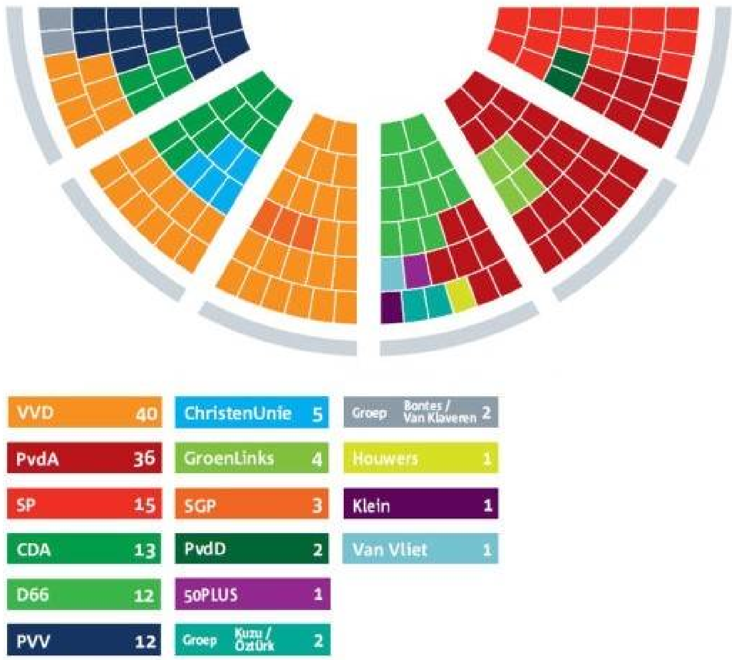

Figure 5: Seat distribution of the Dutch House of Representatives, based on the elections of 2021. *Source: www.Staten-Generaal.nl*

3.4.2 Translation of data sets

In the data sets from the previous section, parties give their preference for one of the options. Therefore, we assign a preference value $p_j(i)$ for each party $j \in N$ over each issue $i \in I$, which is a value we assign to the given option. For example, when party j fully agrees with issue i, then we can set $p_j(i) = 1$. This variable should not be seen as a certain score, but rather as a numerical value with which we can compute the utility of a party.

We propose multiple translation functions to translate the data from the data sets to preference value sets. All value sets only have values between 0 and 1. Translations can be done as following:

Options	linear	semi-linear
(A, D)	(1, 0)	(1, 0)
(A, N, D)	(1, 0.5, 0)	(1, 0.5, 0)
(FA, A, N, D, FD)	(1, 0.75, 0.5, 0.25, 0)	(1, 1, 0.5, 0, 0)

Table 1: Different ways to translate the raw data. (FA = Fully Agree, A = Agree, N = Neutral, D = Disagree, FD = Fully Disagree)

The difference between linear distribution and the semi-linear distribution is only relevant when there is a certain grade in how far parties agree or disagree with an option. It is hard to express in numbers what the exact difference is between 'Fully agree' and 'agree'. For this reason, we test with both linear and semi-linear distribution.

Political parties value certain issues higher than others. However, we assume that all issues are valued as equal and we thus assign no weights to issues. This is mainly due to lack of useful data.

3.5 Party Utility functions

With these value sets, we can calculate utility values for our parties with a certain value of an issue, based on their preferences. We present multiple methods to calculate such a utility value.

The first method to calculate the utility of a party for a given issue I_i is linear and based on the Euclidean distance, as in [52], and the MAE (Mean Absolute Error), as in [53]. The utility function is defined as

$$u_j^{MAE}(\omega_k) = \frac{1}{|I|} \cdot \sum_{i \in I} 1 - |\omega_k(i) - p_j(i)|$$

The second utility function is based on the RSME (Root Squared Mean Error) method [53]. This method is more popular among statisticians than MAE. The utility function using RSME is:

$$u_j^{RSME}(\omega_k) = 1 - \sqrt{\frac{\sum_{i \in I}(\omega_k(i) - p_j(i))^2}{|I|}}$$

The third way to calculate the utility for a party is to translate preferences to binary values (as in [54]), which gives a utility value of 1 when the party somehow agrees (fully or normal) with the given value from the value set, and otherwise a value of 0:

$$u_j^{BV}(\omega_k) = \frac{1}{|I|} \cdot \sum_{i \in I} \begin{cases} 1 & \text{if } \omega_k(i) \geq 0.5 \text{ and } p_j(i) \geq 0.5 \\ 1 & \text{if } \omega_k(i) \leq 0.5 \text{ and } p_j(i) \leq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

This third method ignores the subjective difference between 'Fully agree' and 'agree'. Furthermore, it assumes that a neutral value for an issue can work with both proponents and opponents and thus is given a value of 1. The method thus only assigns a utility of 0 when the preference of a party is in full contrast with the value of $\omega_k(i)$.

The graph below visualizes the effects of the proposed utility functions on linear value set $V = \{0, 0.25, 0.5, 0.75, 1\}$, with $I = \{1\}$ and $p_j(1) = 0.25$.

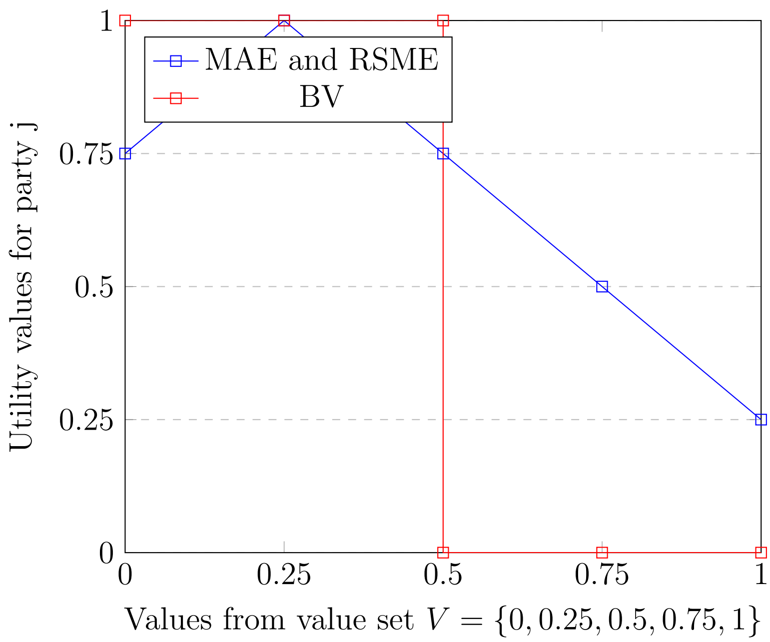

Figure 6: Example utility graph for utility functions for $I = \{1\}$ and $p_j(1) = 0.25$. Because we plot for one issue, the difference between MAE and RSME can not be seen.

3.6 Coalition Utility functions

For our experiments, we use different utility functions for coalitions to test which we can use best to make predictions. The first utility function is simply the average of the utilities of the political parties. This function is defined as

$$u_C^{avg}(\Omega) = \max_{\omega_j \in \Omega} \sum_{i \in C} \frac{1}{|C|} \cdot u_i(\omega_j)$$

The second utility function is the weighted average of the utilities of the political parties (as in [21, 55]), using a normalized weight $w'(n)$ based on $w(n)$ from section 3.2, where $w'(n) = \frac{w(n)}{w(C)}$. This function is defined as

$$u_C^{wavg}(\Omega) = \max_{\omega_j \in \Omega} \sum_{i \in C} w'(i) \cdot u_i(\omega_j)$$

These functions find utility values for the best possible $\omega_j \in \Omega$. This ω_j thus represents the best connection between the parties in C, as in [19].

3.7 Methods

3.7.1 The algorithm

Combining the methods from previous sections result in an algorithm to predict coalitions. We use a set of political parties N which have at least 1 seat in the House of Representatives. As noted before, the model is mainly based on minimum winning connected coalitions from [19]. We already defined in section 3.3 that a coalition must be feasible winning to get a utility value higher than 0. In the algorithm, this is translated to a restriction on the coalitions in $\mathcal{C}$. The algorithm predicts coalitions by returning the coalitions with the highest utility value.

The final equation we get from these methods is:

$$C^* = \operatorname*{argmax}_{\{C \in \mathcal{C} | w(C) > \theta\}} u_C(\Omega)$$

which we can extend to:

$$C^* = \operatorname*{argmax}_{\{C \in \mathcal{C} | w(C) > \theta\}} \max_{\omega \in \Omega} \sum_{x \in C} W_c \cdot u_x(\omega)$$

where W_c is the weight according to the coalition utility method we use. This can even be further extended to:

$$C^* = \operatorname*{argmax}_{\{C \in \mathcal{C} | w(C) > \theta\}} \max_{\omega \in \Omega} \sum_{x \in C} W_c \cdot W_p \cdot \sum_{i \in \omega} U_{xi}$$

where W_p is the weight according to the party utility function (which always is equal to $\frac{1}{|I|}$) and U_{xi} is the utility of a party x for a single issue i. Finally, we can rewrite this as:

$$C^* = \underset{\{C \in \mathcal{C} | w(C) > \theta\}}{\text{argmax}} \quad \max_{\omega \in \Omega} \quad \sum_{x \in C} \sum_{i \in \omega} W_c \cdot W_p \cdot U_{xi}$$

The algorithm takes $2^{|N|} \cdot |V|^{|I|}$ utility calculations to predict the best coalition in the worst case. This is because we might need to calculate it for all possible coalitions in $\mathcal{C}$ ($2^{|N|}$) and all possible coalition agreements ($|V|^{|I|}$). For example, with $|N| = 10, |V| = 5, |I| = 30$, the process needs approximately 10^{23} utility calculations. Therefore we need to optimize this algorithm to speed up this process to return an outcome within seconds, while maintaining its performance.

3.7.2 Optimization

Instead of considering all possible coalition agreements in Ω, we can search for the best value $v \in V_i$ of each issue $i \in I$ for coalition C. This way, we reduce $|V|^{|I|}$ to $|V| \cdot |I|$ calculations. We can do this because of the fact that each issue is independent from another and finding each best value for an individual issue will result in an optimal coalition agreement. Taking the example from above, the calculations are reduced from approximately 10^{20} to 150. We can formulate the new equation to find C^* as:

$$C^* = \underset{\{C \in \mathcal{C} | w(C) > \theta\}}{\text{argmax}} \quad \sum_{i \in I} \max_{v \in V_i} \quad \sum_{x \in C} W_c \cdot W_p \cdot U_{xv}$$

However, after this optimization, we still have $2^{|N|}$ possible coalitions to consider. The first step we took is to prune this set to only coalitions which are feasible winning coalitions ($w(C) \geq \theta$). Then, we apply the concept of minimal winning coalitions, which are coalitions without redundant parties to achieve a majority. Finally, we add a restriction to the maximum amount of parties allowed in a coalition. In practice, these steps can reduce $2^{16} = 65536$ coalition to 96 with a maximum amount of parties of 5. This maximum amount is reasonable because of the fact that in practice, there has never been a cabinet with more than 5 parties in the Dutch parliament. In conclusion, we have optimized the algorithm from 10^{23} utility calculations to $150 \cdot 96 \approx 10^4$.

3.7.3 Experiments

We run our experiments on the model as described in section 3 with the data sets from section 3.4.1, namely Stemwijzer 2021 and Kieskompas 2021. The seat distributions we will use are those of 2017 and 2021. For the experiments, we use different values for our variables and different utility functions. We begin with setting $\theta = 76$, because a majority is always required in this coalition formation, as described in section 3.1. We run the experiments while changing the following settings:

- A restriction that **the largest party in the House of Representatives must be in the coalition**. This is based on real political coalition formation in the Netherlands, where the largest party takes initiative in forming a coalition. A side note must be made that there are cases when the largest party did not end up in the cabinet, as in 1977 [2].
- A restriction whether to **consider the influence of the seat distribution of the Senate**. Political parties usually want to create a majority in the Senate as well, when forming a coalition. In the past this happened several times (1945, 1948, 1952, 1956, 1959, 1963, 1965, 1972, 1981, 1998), which shows that it is a factor in the Dutch coalition formations. Therefore, we need to experiment with taking a majority in the Senate into account. This can be done by setting a new threshold $\theta' = 38$, which restricts the coalition formation to gain a majority in the Senate, which has a total of 75 seats.
- A **maximum amount of parties allowed in the coalition**. This setting is mainly because of optimization reasons. The lower amount of parties amount, the faster the model runs. However, we need to test what the impact is of this setting and if the results won't differ too much.

Furthermore, we test with different preference models and utility functions as described in the previous sections:

- **Value sets** (section 3.4). We test the model using linear and semi-linear value sets.
- **Party utility functions** (section 3.5). We test the model using the MAE function, the RSME function and the BV utility function.
- **Coalition utility functions** (section 3.6). We test the model using the average and the weighted average.

Running experiments for all possible combinations of these methods, settings and data sets gives us the predictions of the model in all these situations. Comparing these results shows how certain methods might give completely different outcomes or very similar ones. Interpreting these outcomes by looking at the data and real coalition formation answers research question 1 on how to model a coalition formation (section 1.2.1).

3.8 Results

3.8.1 Evaluating correctness

We start the development of the model with creating basic test-cases, which we can reuse to check the correctness of the model when improving it. All of the results are as expected, except for one. The party utility function BV from section 3.5 showed a major flaw. In the set Ω, there always exists a ω_i containing only values of 0.5. Because of the fact it assumes that parties would agree with a neutral opinion, Method BV gives this ω_i always a utility of 1. Therefore, each possible coalition in $\mathcal{C}$ will obtain a coalition utility of 1 with ω_i, which leads to the conclusion that all feasible winning coalitions are the best possible coalition. We find these results meaningless and thus we scrap the method BV from the experiments.

When scrapping method BV, only RSME and MAE remain in the proposed party utility functions. However, after the optimization as described in section 3.7.2, we can prove that the outcomes of RSME and MAE are always the same. This is due to the fact that parameter ω_k always has length 1 after replacing $u_C(\Omega)$ for $u_C(i, v)$. As illustrated in figure 6, we can see that the results of the methods are the same with $|I| = 1$. Therefore, running experiments only with MAE will be sufficient.

3.8.2 Model Behaviour

Looking at the coalition utility functions from section 3.6, we notice interesting results. Using the weighted average for coalitions with a highly dominating party in size, larger differences in ideologies will matter less. For example, When comparing coalitions $C1 = \{$*VVD (75), BBB (1)*$\}$ and $C2 = \{$*VVD (75), JA21 (1)*$\}$, we can see that $u_{C2} > u_{C1}$. However, when we change $C2 = \{$*VVD (75), JA21 (5)*$\}$, the results alter to $u_{C1} > u_{C2}$. This happens because of the fact that the utility value of a coalition is less based on the largest party when giving a smaller party more seats. It makes sense that a party with more seats has a higher influence, yet it doesn't make sense that with small margins a less favorable party matches better with the largest party.

In section 3.7.2, we reduced the amount of considered coalitions by using the minimal winning coalition concept. This reduction makes the algorithm faster, but for the model behaviour we need to look if this optimization has a big influence on the results. Logically speaking, adding another party to an already feasible winning coalition should not make the coalition better. It only adds more disunity to a coalition.

However, in many cases it does make the utility of a coalition higher. When the utility of a single party for the coalition agreement of a minimal winning coalition is higher than the utility of that coalition, a new coalition including that single party will have a higher utility than the current coalition (e.g. VVD and D66 have a majority in seats and a overall utility of 0.8. Adding FVD with a utility of 0.85 will make the overall utility better and thus the coalition as well).

When we look at the setting *Largest party in coalition*, we observe that this setting is redundant in most cases. Because of the fact the number of possible coalitions is already limited amount due to θ, in practice there are little coalitions possible without the largest party. However, when we allow more parties in a coalition, more coalitions can be formed without the largest party. In these cases, it might be useful to apply this requirement.

Requiring a majority in the Senate is a setting which influence strongly depends on the seat distribution in the Senate. In the case of the elections of 2021, there were little coalitions possible when requiring a majority in the Senate. In practice, the coalition that was formed that year did not have a majority in the Senate, which makes it harder using this setting consistently. However, in for the elections of 2017, this setting did help eliminate some coalitions by rejecting them based on a minority in the Senate.

The setting *Maximum allowed parties in coalition* has a large influence on the behaviour of the model. For example, when lowering this setting from a maximum of 5 parties to a maximum of 4, it eliminates approximately 75% of the feasible coalitions. However, overall the model ranks coalitions of 5 parties higher than coalitions of 4 parties. Lowering the maximum amount of parties generally thus removes the best coalitions.

Noticeable is the fact that we run experiments with the seat distribution of 2017, while we don't have any issue sets available from those elections. This is not ideal, yet we can assume that the ideologies of parties won't change significantly over a time period of one election. When we compare the behaviour of the model for the seat distributions of 2017 and 2021, we don't see any remarkable differences in outcomes. Therefore, we still find the results for the seat distribution of 2017 useful.

3.9 Conclusions

We presented a model with different methods and settings, by combining existing concepts and functions. Furthermore, we added settings based on political coalition formation in practice. After evaluating the correctness of the model and observing the behaviour of the model, we can cross out some methods. Despite some minor flaws in the model, the overall performance is quite good.

In section 1.2.1, we stated that we have to find a balance between ideology and size to create a good prediction model. A good method to make choices in how to value ideology and size is the behaviour of the coalition utility functions. We showed that size is way more important in the weighted average than in the average method. According to the work of Franklin et al. [14], we should valuate ideology higher than size and thus choose for the average method as the most suitable for the Dutch coalition formations in the House of Representatives.

Evaluating other methods and settings based on both size and ideology is harder. We can't draw conclusions about the size of a coalition based on the way we model value sets. However, we can interpret the behaviour of the maximum amount of parties in a coalition. When increasing this maximum, it is more likely that coalitions with more parties in it are ranked higher than coalitions with smaller amount of coalitions. This means that more (smaller) parties can unite better based on ideology than (larger) parties based on size. Again, this is in line with the findings of Franklin et al.

When we compare our model with the existing theories of Browne, Taylor-Laver and de Swaan (see section 2), we notice that it follows theories of Taylor-Laver and de Swaan most. According to the theory of Browne, the property of minimum parties should be considered the best property to follow. However, our model tends to put more parties in a coalition to find more connected coalitions. Therefore, our model is most in line with the minimum connected winning coalitions from Taylor-Laver and de Swaan.

In the next section, we show how different methods relate to each other and how much results differ when changing a method.

4 Measuring robustness

In section 1.2.2, we stated the research question on how to measure robustness. To do this, we need to analyze the results from section 3.7.3. To have a good analysis, we need to look into robustness theorems [56, 57] in order to do a correct analysis. Options in robustness analysis can vary from Root Mean Squared Error of Prediction (RMSEP) to the signal-to-noise ratio (SNR) to Jones' method [58]. SNR is a frequently used method, yet we can't define what we specify as noise in our data. A small party with only one seat can be noise, but could be in the best possible coalition as well. This makes SNR unsuitable for our analysis. Jones' method is based on different types of noise as well, and thus makes this method also unsuitable. We find RMSEP the most suitable method for our data. With this method, we can compare the influence of different settings and methods on the results, by measuring how much changing a setting will affect the results.

Usually, robustness analysis is done knowing the true prediction. In our case, we can use the formed coalition after an election. However, we do not know whether the coalitions formed after elections are the best ones or not. Nevertheless, we can assume that those coalitions are among the top of possible coalitions and thus will be good predictions. However, if the model predicts those coalitions as second best, the prediction will be labeled as false, while it might be a good prediction in a space of hundreds possible coalitions. We thus should not label predictions as true or false, but as a distance from the truth.

Furthermore, we will follow the ideas of Orzack and Sober [59] about robustness analysis. They argue that a robustness analysis for multiple models should be done by looking at their properties and identify robust properties. Using our proposed methods and settings, we can see the variations of our model as multiple models. These methods and settings can then be seen as properties of the model and we can analyze the robustness of those properties using the RMSEP analysis.

4.1 RMSEP analysis

We use the formula from [60], where we have p test cases. For each case, we have a result y_i and $\hat{y}_i$ for the different settings with which we test. For the results, we look at the actual coalition of the elections of that year and at what rank our model places that coalition. For example, the coalition of 2021 was *VVD, D66, CDA, CU*. In test case X, this coalition is ranked as fifth best with setting x and seventh with setting $\hat{x}$, so $y_i = 5$ and $\hat{y}_i = 7$. The formula is as following:

$$RMSEP = \sqrt{\frac{1}{p}\sum_{i=1}^{p}(y_i - \hat{y}_i)^2}$$

Table 2 shows the results of the RMSEP analysis. In section 3.8 we stated that we only use one party utility function, namely MAE. Furthermore, we stated in section 3.7.3 that we will test with the setting that a majority in the Senate is required. However, the formed coalition after the elections of 2021 did not have a majority in the Senate, which makes the results of those test not usable. Therefore we do not have an RMSEP analysis for these settings.

Setting	Option 1	Option 2	RSMEP
Issue set	Stemwijzer	Kieskompas	28,42
Seat distribution	TK2021	TK2017	8,79
Coalition utility function	Average	Weighted average	12,21
Value set	Linear	Semi-linear	1,39
Largest party in coalition	Yes	No	2,76
Maximum parties allowed in coalition	4	5	29,35

Table 2: Results from RMSEP analysis, with $p = 32$. TK stands for the elections of the House of Representatives that year. In both 2021 and 2017, the same coalition was formed, namely *VVD, D66, CDA, CU*

One of the largest RMSEP values comes from the *issue set*. A large RMSEP value for the issue set means that the two data sets are quite different. We observe that the set Kieskompas overall ranks the actual coalition higher than the set Stemwijzer. This does not imply that one is better than the other, but that it does matter a lot which data set a user chooses.

Looking at the other data related setting, *the seat distribution*, we notice a much smaller RMSEP value. This means that a fluctuation in seats has less influence than a fluctuation in issues. However, the RSMEP value might be higher or lower when using other seat distributions, where the difference in seats might be larger or smaller.

The RMSEP value of the *coalition utility function* is relatively high. This is mainly caused by test cases with many coalitions as outcome. In these cases, the *Weighted average* will rank the actual coalition higher than the *Average* method. We can show this by calculating the RMSEP value for the coalition utility function for a maximum of 4 parties in a coalition (RMSEP = 1,70) and for a maximum of 5 parties (RMSEP = 17,18). Thus a larger amount of coalitions will result in less robustness for variation in the coalition utility function.

The analysis for the *value set* shows that it matters little which option we choose. This is partly due to the fact that *Linear* and *Semi-linear* result in same outcomes when testing it with the Stemwijzer data set. Nevertheless, this setting is quite robust.

The setting *Largest party in coalition* has a small RMSEP value as well. This means that forcing the largest party in the best coalitions does not change the results much. The test cases showed that creating a majority without the largest party wasn't always possible with a restriction on the maximum amount of parties allowed.

With the analysis of the coalition utility functions, it is not a big surprise that the setting for *Maximum parties allowed in coalition* isn't robust at all. In section 3.8, we have already seen that adding an extra party to the coalition can result in higher utility values. The consequence of this behaviour can thus be seen in the RMSEP value. By allowing more parties in a (minimal winning) coalition, the model can find more possible coalitions which utility values lies somewhat higher than coalitions with less parties in it.

We can now compare these results with results from existing literature. In section 2 we have seen the work of Franklin et. al. [14] who studied coalition formations in different parliamentary democracies. They stated that in Dutch coalitions, ideology is weighted more than the size of a coalition. This statement is partially in line with the behaviour of our model. When we allow more parties in a coalition, the model ranks coalitions with more parties in it higher than coalitions with less parties. As explained before, this behaviour comes from finding higher utility values for coalitions with more parties. However, these coalitions don't come close to actual formed cabinets, so it is hard to obtain robust conclusions from this comparison.

4.2 Conclusions

In conclusion, we measured the robustness of our model using an RMSEP analysis. We found this method the most fitting one for our data, due to the variations in the model. the results from this analysis show that our model is not robust enough when changing certain settings. The outcomes vary a lot, which is mostly due to the amount of maximum parties allowed. This implies that the model should be revised in a way that the amount of parties in a coalition should have less influence on the model. For example, a coalition utility function could be altered by multiplying with a variable $\alpha = \frac{1}{|C|}$, which punishes larger coalitions.

As stated in section 1.2.2, if we can prove that our model is robust, we can give users of the model a certainty for the quality of the results. The RMSEP analysis has shown us that our model isn't fully robust. Therefore, we can't guarantee that with new data, our model will behave similarly as in the behaviour described in section 3.8.2. We thus can't give certainty for the quality of the results to the users. What we can do is make clear why the model behaves the way it does and why certain outcomes are given. We will discuss this in the next section.

5 Giving insight in coalition formations

To use the model in practice, we implement it in a web-application, called *De Coalitiewijzer*. The purpose of this application is to set an interactive environment where users can change settings to output possible coalitions. The application always shows the best coalitions based on the settings. For example, a user can select different data sets, create a personal seat distribution, add a higher restriction for the threshold and even select a utility function. This way, the user can experiment and find out what happens to a coalition when certain settings change.

The goal of this implementation is to give users insight in the process of coalition formation. The application visualizes the outcomes and shows how the results can be altered by using different settings or data sets. We can thus answer the third research question from section 1.2.3 using this approach.

5.1 Design and functionalities

Figure 7 shows the design of the web-application. At the top left side of the application, users can change settings with respect to the seat distribution and the data set for issues. A default seat distribution can be selected, based on previous elections (e.g. elections of 2017) or future polls. Our application downloads the latest polls from the Peilingwijzer [61] on a daily basis. The Peilingwijzer uses different sources to create polls as accurately as possible. Using the data from this source, people can use De Coalitiewijzer with the most recent seat distributions. Using previous elections is good to validate and explain how coalitions are formed, but polls can actually help voters in the process of choosing a party to vote for and thus stimulate tactical voting.

On the bottom right side, users can create a personal distribution as well, with a restriction that the total amount of seats must be a fixed size (according to the the Dutch House of Representatives, which is 150 seats). A 'random distribution' button can be pressed to randomly distribute seats. The goal of this 'random' button is to make the application more experimental for users. However, many times the web-application can't find any possible coalitions when using this functionality, due to the restriction of a maximum amount of seats allowed in a coalition (more on this later). Therefore, we need a few large parties to form the basis of a coalition, which a random distribution doesn't offer most of the time.

Figure 7: Screenshot of *Coalitiewijzer.nl*

Above these manual seat distribution settings users can adjust more settings, which will influence the calculations of the model. A user can select different utility functions as described in section 3 and give restrictions (section 3.7.3) to the formation process. We have set the default values of the web-application in a way that it gives results within seconds and that the results are most intuitive for users. The basic settings are defined as following:

- The largest party must be in coalition (**True** or **False**, default = False)
- A majority in the Senate is required (**True** or **False**, default = False)
- A minimal winning coalition is required (**True** or **False**, default = False)
- Minimum amount of seats required (default = 76)
- Maximum amount of parties allowed in coalition (default = 4)

We create a button which opens a window where users can change more advanced settings, namely the utility functions. We hide these functions behind a button, because these settings are harder to comprehend for users and should only be used by users who know what they mean. Again, we choose default values based on speed and intuitiveness. The advanced settings are defined as following:

- Main utility method (**Median/mode** or **Linear**, default = Median/mode)
- Coalition utility method (**Average** or **Weighted average, default = Average**

- Party utility method (**MAE/RSME**, default = MAE/RSME)
- Value sets (**Linear** or **Semi-linear**, default = Linear)

Note that we added a similar functionality to the (linear) model we use in this research, namely Median/mode. This method is even faster than the presented optimization in section 3.7.2, but we can't guarantee its correctness. However, the results are very similar and thus we use this method in practice.

The method Median/mode searches for the median in the highest preferences of the parties in a coalition for a single issue. For example, when we have three parties of which two have the highest preference 'agree' and one has 'disagree', the median is 'agree' and thus we take this value for that issue in our coalition agreement. When there is a tie, we take the mode of these options. Instead of $|V| \cdot |I|$ calculations for a coalition, we downgrade this even further to $|I|$ calculations.

In the middle of the web-application, all the feasible coalitions will always be shown with the current settings, sorted by utility value. The outcomes can be expanded (figure 8), where users can see more information about a certain result, like the individual utility per party and the coalition agreement which came with it. The utility values are translated to percentages, to make it easier to understand for users. Using colors corresponding to political parties, users can see at first glance which parties stand out in the top coalitions.

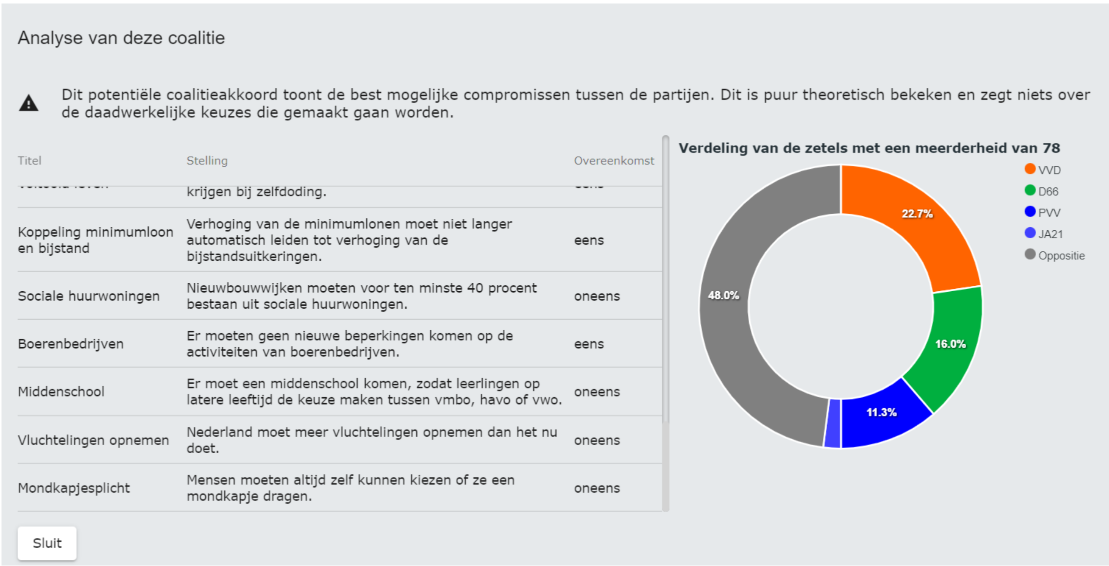

Figure 8: A generated coalition agreement and a donut chart for a given coalition

To give more insight in the disunity of political parties, we reconstruct the political spectrum (as shown in section 1) in the bottom left side of the web-application. We take all the issues from a data set and select all the preferences $p_j(i)$ of political parties for all the issues from the set, as a matrix of size $|I| \times |N|$. Then we use Principal Component Analysis (PCA) [62] to reduce all $|I|$ preferences for one party to just two values. With these values, we can plot all the parties in a scatter plot and visualize the division of the parties. Figure 9 show how the generated graph (left) looks like in comparison with the manually made graph by Kieskompas (right). It shows quite some resemblance when looking at certain clusters of parties (e.g. SGP, JA21 and FVD).

PCA linearly transforms n-dimensional data to a new coordinate system, which in our case is 2-dimensional. This method is widely used in all kinds of research fields, like neurosciences and biology. PCA can be sensitive to outliers in data. However, all the values in our data lie between 0 and 1. Therefore no preprocessing is needed. Because PCA is especially suitable for predictive models, we reject other methods, like Independent Component Analysis (ICA) and Network Component Analysis (NCA).

Figure 9: Comparison of Kieskompas graph from figure 1 and rendered graph using PCA for issues from Kieskompas

To make this plot interactive, we add the functionality to click on the parties in the plot to add them to a coalition, as can bee seen in figure 9. These parties appear in a bar underneath the scatter plot and it gets filled when adding more and more parties. The utility of this coalition is shown as well. After clicking on a party in the scatter plot, the best next k options for an additional party in the coalition are highlighted. These k options are based on the utility value for the current selected coalition, added with the new potential party. Unfortunately, creating a coalition while always choosing a new party in the top k options can lead to sub-optimal results. This is because of the fact that the k options don't hold θ into account and thus can lead to local optima instead of global optima.

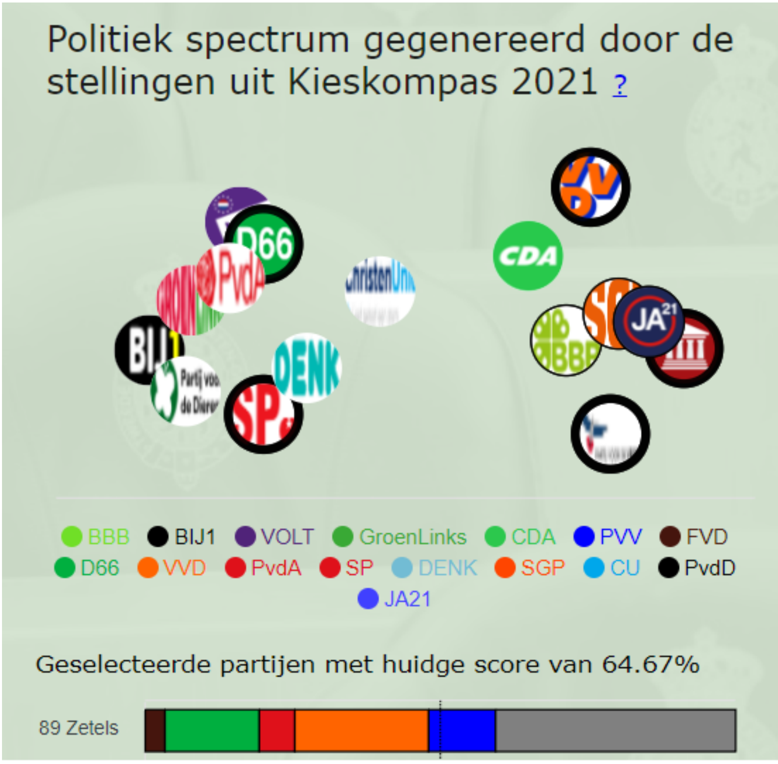

Figure 10: The functionality of the web-application to create a personal coalition by selecting different parties

When we calculate all possible coalitions of the current selected coalition C in combination with each party $n \notin C$, we thus calculate the marginal contribution for each party n, i.e. $U_C - U_{C \cup n}$. Because this value would not mean much to users of the application, we do not use it any further than calculating the best next k options.

The graph is extremely helpful in understanding the results given by our web-application. At first glance, users already see the disunity between parties on the graph. Users can trace the steps of a coalition formation by adding parties to a coalition and observing the change in utility value.

At last, we create a link to a second page, which gives more information about the web-application and its underlying model. In a more simplistic Dutch usage of words we explain how the model works, to make it more understandable for people without much knowledge about Computing Sciences. On this page, a feedback form can be found. With the feedback from users, we can improve the web-application even more and adjust it to the needs of users.

Before we started with designing the web-application, we made some requirements for the appearance. First of all, the theme should not relate to existing political parties and thus remain neutral. Secondly, the web-application should have a calm radiation. We thus chose for a light and calm theme with different shades of green.

5.2 Process of development and architecture

The web-application is developed using Agile methods [63]. These methods are widely used and very popular. First, we categorized the functionalities (as elaborated in the previous section) of the web-application in 'must have', 'should have', 'could have' and 'won't have'. We started with building a *version 0*, which contained all the 'must haves'. From there on, we started adding 'should haves' for further versions. The first three versions were described as following:

- *Version 0* must contain the options to select a data set of issues and a seat distribution. It should always show a best coalition. This version is sufficient to run basic test cases.
- *Version 1* should have the page containing information about the model. Users should be able to create an own seat distribution and add restrictions on the model. This version is sufficient to run all experiments for the model.
- *Version 2* should have the political graph in which users can select parties to be in the coalition.

The web-application is built using the back-end framework Django [64] (Python) and the front-end framework Angular [65] (Typescript, html, scss). The architecture is a client/server architecture. The server side runs on a Ubuntu machine with 4GB RAM and 2 CPU cores. Using REST-API calls, the client requests data and results from the server. The server can calculate a ranking of coalitions within seconds. However, when allowing more parties in a coalition, the model becomes drastically slower.

5.3 Conclusions

In this section, we have shown how we give insight in coalition formations and the feasibility of different coalitions. By building a suitable web-application for users, we can show and visualize how and why certain parties form better coalitions than others and what influences this process. The web-application is available for everyone. We expect that the application is most useful during election times with the help of the data from the Peilingwijzer.

In section 3, we looked at the the ideology and size of coalitions. In the web-application, we implemented an accurate graph to visualize the difference in ideology between parties using the PCA algorithm. However, the effect of the size of parties in a coalition is less clear when using the application, due to the low robustness, as explained in section 4.

Earlier in this work, we stated that a coalition prediction model can influence voting behaviour, like increasing tactical votes in election times. Our application does not provide an extensive analysis on how a small change in seats will result in different coalitions, but it does give a good overall view on how different sizes of parties influence the top predicted coalitions. The results are clear enough for voters to give a general understanding of the influence of elections results on a coalition. Furthermore, the political graph helps in finding parties similar in ideology with their most preferred party. When one of those parties has a high chance of being in one of the best predicted coalitions, it might stimulate voters to make a tactical vote.

6 Discussion and Conclusions

In this last section we will discuss the results from our research. We show the limitations of our model and aspects which could have been done better. Furthermore, we elaborate on how this work could serve as an inspiration for future works. We conclude this research by briefly summarizing this work.

In section 1.2, we stated the main research question as how to predict the outcome in the coalition formation of the Dutch House of Representatives. We created a model which predicts coalitions based on political issues and preferences and on a seat distribution. We showed different methods to calculate the best coalitions and analyzed their performance and robustness. Furthermore, we built a web-application to give insight in the coalition formation process and results. Users can use this application to change settings and form own coalitions using a political graph. Functionalities are explained on a separate web-page, to make the model more understandable.

With this model and web-application we can help voters understand how coalitions are formed and why such formations can take a very long time. With the help of the political graphs, we can show users how closely parties lie to each other based on their ideologies. We might not overcome many issues in coalition formations, but our model can stimulate tactical voting in elections and forming one front between parties, like the PvdA and Groenlinks. Furthermore, looking at the individual utilities per party in a coalition, we can make risk assessments on which parties are the least content with the coalition and would thus be most likely to terminate a cabinet. Our model thus makes fine predictions and gives insight very well.

6.1 Limitations

6.1.1 The model

When we look at the performance of the model, we can start by evaluating the influence of the different settings on the results. With the right settings, the model can rank the actual coalition formed after an election quite high. However, changing some settings can also lead to very different results (see section 4). Especially allowing more parties in a coalition results in a large amount of possible coalitions, in which the actual coalition is ranked quite low. This doesn't make the model only inaccurate, it also makes giving insight in coalition formation harder. A possible solution for this problem is by decreasing the utility value of a coalition when a coalition gets larger in size.

In section 3 we proposed multiple methods to use in our model. For our analysis, we could not use all the methods due to incorrectness for some of them.

Therefore, it would have been valuable to have proposed more utility functions for our model. For example, for the coalition utility function, we could have simply looked at a function which takes the smallest utility value of the party in that coalition for a certain coalition agreement. This method punishes coalitions in which certain members are not content with the agreement at all.

Another limitation is the influence of the issue data set. It is remarkable that in each test case, the Kieskompas set ranked the actual coalition higher than the Stemwijzer set. We can't really say which data set is better, because we can't judge whether the actual coalition in the House of Representatives is a good option or not. However it is not desirable that these results differ so much.

According to theories from Taylor-Laver and de Swaan [14], the smallest parties (where their size is beneath a certain threshold) should be excluded from feasible coalitions. This property could have been used as an additional setting for our experiments. This could have excluded some more feasible coalitions to achieve different results. Such a settings would thus still be a nice feature for a future version of our web-application.

At the start of this research, we aimed for more than two data sets, divided in three different categories. The first category is long-term ideology of political parties. A data source for this category is the *Partijenwijzer* of ProDemos. Unfortunately, this tool has been offline during the time of this work. No other data sources were found, so we did not have any data in this category. The second category is that of ideology per election. We obtained data from Kieskompas and Stemwijzer, but many other instances failed to respond to our requests for data. The last category is that of actual voting data of parties in the House of Representatives. This data is available, yet unusable (see section 3.4.1). With more data, we could have made a better analysis and draw more solid conclusions.

As explained in section 3.7.2, we needed to optimize the algorithm to get results in a reasonable amount of time. Due to the immense duration of the original algorithm, we never obtained results using this method and thus we did not validate that our optimization gave similar results. Due to the mathematical correctness of the optimization, we do not expect that the results would have differed much.

6.1.2 Robustness and analysis

We have looked at the robustness of our model when changing the seat distribution. This gives us insight in how much influence large changes in the amount of seats has. However, we didn't look at smaller changes in data (e.g. changing just one or two seats per party). This could have told us in what matter minor shifts in seats would mean for our results. Furthermore, it could give users more insight in what changes need to be made to get different results.

Furthermore, more analysis on the prediction model could have been done to measure the accuracy of the model. The work of Debray et al. [66] show more methods to analyze prediction models. For example, an empirical study could have been done to validate the prediction model even better, where users of the model could have rated the predictions given by the model.

6.1.3 Implementation and insight

Not being able to veto parties in a coalition is another shortcoming of our model. During the actual formation process after an election, party leaders often state that they won't cooperate with another party. For example, in 2021 the VVD would not cooperate with the FVD and PVV, despite the fact that their ideologies lie relatively close to one another. Possessing and processing data with these veto's could cross out some more feasible winning coalitions.

However, not including this data keeps the model (and web-application) more generic. Other Coalition formation tools, like *www.coalitiechecker.nl* and *De Volkskrant coalitiewijzer* take statements from politicians and media into account, while our model purely looks at the preferences of the parties. Furthermore, this makes those models more biased, because larger parties get more attention from the media and thus have more influence on these other applications.

The results in the web-application could have been presented with even more details. In first instance, the idea existed to show in the presented coalition agreement (see figure 8) the preference of each individual party for each issue. This could clarify more to users why a certain agreement was made. This is a feature which can made in a future version of the web-application.

Furthermore, we described in section 2 how the concept of stability in coalitions can show if parties might wish to leave a coalition to form a better one. Showing such desires to form other coalitions would have fit in nicely in our web-application. This could have been a first step in modeling the behaviour of political parties in a coalition formation process. It also helps in evaluating failed coalitions from the past, by analyzing the stability of that coalition.

In section 3.7.2, we proposed some optimization methods to make the algorithm faster. We achieved large speed-ups, yet the time to calculate a coalition still is a bit too long for a user-friendly interface. It is a bit slow to work with in comparison with other web-applications. To increase the usability of the web-application, it eventually needs more speed-ups. This can be done by working with caches and storing more results in a database.

6.2 Future work

In this work, we have looked at a model which predicts the best coalitions in a vast space of possibilities. We have given insight in the process and results for users, by building a web-application. However, there are more approaches in coalition formation in the House of Representatives. One of these approaches is automated negotiation [67, 68]. After finding all possible coalition agreements for parties in a possible coalition, we can let agents negotiate for a certain outcome. We can calculate what the best agreement would be, but if we can simulate how political parties would form an agreement after negotiation, we could get even better predictions.

In section 2 we mentioned prediction models based on machine learning algorithms. This is another approach to predict coalitions. The downside of this method is that it is quite hard to understand why the model gives certain results. On the other hand, this approach can better handle larger amounts of data and process it.

As stated before, more work could be done with respect to the stability of coalitions. There is still much to be explored in this field. Applying existing concepts, like the Shapley value [69, 70], on our data and model could give more and even better understanding of coalition formations in the Dutch House of Representatives. The Shapley value takes the expected marginal contribution of a party for a certain coalition. This way, we can see what individual parties contribute to the utility of existing coalitions.

With our model, we created new possibilities for other fields of research. Studies in politics could use our model to analyze and interpret coalition formation processes in parliamentary democracies. If our model would be widely used, studies about tactical voting could be done as well. Furthermore, we could do analyses on how users would use the web-application and which features they use the most. This could help in improving the web-application and thus in making it more user-friendly.

6.3 Conclusions

We showed that we can predict coalitions to a certain degree. Due to many factors in real coalition formations, it is not always possible to give a good prediction. Therefore, we don't expect that our model is suited for real formations and thus won't solve the issue of long formation processes. However, the web-application *De Coalitiewijzer* can show why a certain prediction is given, which helps voters in understanding the results. Due to very useful insight in the formations and the latest polls from the Peilingwijzer, the web-application can really help in making tactical votes. Users of our model should keep in mind that the robustness of our model is not very high and that changing certain settings could lead to very different outcomes. This answers our research question and concludes this work.

Acknowledgements

I'd like to thank the people who have invested a lot of time in my research by reviewing drafts and giving feedback on the web-application. First of all, Tim Baarslag, my main supervisor, who had a lot of ideas and suggestions to set up and improve this research. Furthermore, thanks to Sho Cremers, Jos Zuijderwijk and Marlouke San Giorgi for reviewing my draft works.

References

[1] P. F. Maas, *Kabinetsformaties: 1959-1973.* 's-Gravenhage: Staatsuitgeverij, 1982.

[2] D. Diermeier and P. Van Roozendaal, "The duration of cabinet formation processes in western multi-party democracies," *British Journal of Political Science*, vol. 28, no. 4, pp. 609–626, 1998.

[3] B. Grofman and P. Van Roozendaal, "Modelling cabinet durability and termination," *British Journal of Political Science*, vol. 27, no. 3, pp. 419–451, 1997.

[4] G. A. Irwin, "According to the polls: The influence of opinion polls on expectations," *The Public Opinion Quarterly*, vol. 66, no. 1, pp. 92–104, 2002.

[5] G. Evans, J. Curtice, and P. Norris, "New labour, new tactical voting? the causes and consequences of tactical voting in the 1997 general election," *British Elections & Parties Review*, vol. 8, no. 1, pp. 65–79, 1998.

[6] S. D. Fisher, *Intuition Versus Formal Theory: Tactical Voting in England, 1987-1997.* ISPO, 2000.

[7] T. W. Bell, "The constitution as if consent mattered," *Chap. L. Rev.*, vol. 16, p. 269, 2012.

[8] B. P. Mitchell, *Eight ways to run the country: A new and revealing look at left and right.* Greenwood Publishing Group, 2007.

[9] V. Postrel, *The Future and Its Enemies: The Growing Conflict Over Creativity, Enterprise.* Simon and Schuster, 2011.

[10] A. Arian, A. Krouwel, M. Pol, and R. Ventura, "The election compass: party profiling and voter attitudes," in *The Elections in Israel 2009*, pp. 275–298, Routledge, 2017.

[11] A. Krouwel, T. Etienne, and J. van Lindert, "Kieskompas party coding data for the 2021 dutch general," 2022. data retrieved from Kieskompas: Amsterdam.

[12] O. Morgenstern and J. Von Neumann, *Theory of games and economic behavior*. Princeton university press, 1953.

[13] W. H. Riker, *The theory of political coalitions*. Yale University Press, 1962.

[14] M. N. Franklin and T. T. Mackie, "Reassessing the importance of size and ideology for the formation of governing coalitions in parliamentary democracies," *American Journal of Political Science*, pp. 671–692, 1984.

[15] J. W. Baker, M. Schubert, and M. H. Faber, "On the assessment of robustness," *Structural Safety*, vol. 30, no. 3, pp. 253–267, 2008.

[16] T. Caplow, "A theory of coalitions in the triad," *American sociological review*, vol. 21, no. 4, pp. 489–493, 1956.

[17] W. A. Gamson, "A theory of coalition formation," *American sociological review*, pp. 373–382, 1961.

[18] R. M. Axelrod, "Conflict of interest: A theory of divergent goals with applications to politics.," 1970.

[19] B. Grofman, "A dynamic model of protocoalition formation in ideological n-space," *Behavioral Science*, vol. 27, no. 1, pp. 77–90, 1982.

[20] H. Rosenthal and W. Copline, "Voting and coalition models in election simulations," *Simulation in the study of politics*, 1968.

[21] S. Aknine, S. Pinson, and M. F. Shakun, "A multi-agent coalition formation method based on preference models," *Group Decision and Negotiation*, vol. 13, no. 6, pp. 513–538, 2004.

[22] E. W. Kelley, "Techniques of studying coalition formation," *Midwest Journal of Political Science*, pp. 62–84, 1968.

[23] R. J. Aumann and J. H. Dreze, "Cooperative games with coalition structures," *International Journal of game theory*, vol. 3, no. 4, pp. 217–237, 1974.

[24] J. Greenberg, "Coalition structures," *Handbook of game theory with economic applications*, vol. 2, pp. 1305–1337, 1994.

[25] T. Rahwan, T. P. Michalak, M. Wooldridge, and N. R. Jennings, "Coalition structure generation: A survey," *Artificial Intelligence*, vol. 229, pp. 139–174, 2015.

[26] E. Elkind, G. Chalkiadakis, and N. R. Jennings, "Coalition structures in weighted voting games.," in *ECAI*, vol. 8, pp. 393–397, 2008.

[27] T. Arnold and U. Schwalbe, "Dynamic coalition formation and the core," *Journal of economic behavior & organization*, vol. 49, no. 3, pp. 363–380, 2002.

[28] W. W. Sharkey, "Cooperative games with large cores," *International Journal of Game Theory*, vol. 11, no. 3-4, pp. 175–182, 1982.

[29] M. Davis, M. Maschler, and M. Shubik, *Existence of stable payoff configurations for cooperative games.* Princeton University Press, 2015.

[30] D. E. Loeb, "Stable winning coalitions," *Games of No Chance*, vol. 29, pp. 451–471, 1996.

[31] S. J. Brams, M. A. Jones, and D. M. Kilgour, "Forming stable coalitions: The process matters," *Public Choice*, vol. 125, no. 1, pp. 67–94, 2005.

[32] G. S. Collins and K. G. Moons, "Reporting of artificial intelligence prediction models," *The Lancet*, vol. 393, no. 10181, pp. 1577–1579, 2019.

[33] P. C. Fishburn, "Utility theory for decision making," tech. rep., Research analysis corp McLean VA, 1970.

[34] P. C. Fishburn, "Retrospective on the utility theory of von neumann and morgenstern," *Journal of Risk and Uncertainty*, vol. 2, no. 2, pp. 127–157, 1989.

[35] J. Broome, "Utility," *Economics & Philosophy*, vol. 7, no. 1, pp. 1–12, 1991.

[36] H. S. Houthakker, "Revealed preference and the utility function," *Economica*, vol. 17, no. 66, pp. 159–174, 1950.

[37] R. A. Pollak, "Additive von neumann-morgenstern utility functions," *Econometrica, Journal of the Econometric Society*, pp. 485–494, 1967.

[38] R. H. Strotz, "Cardinal utility," *The American Economic Review*, vol. 43, no. 2, pp. 384–397, 1953.

[39] D. Von Winterfeldt and G. W. Fischer, "Multi-attribute utility theory: models and assessment procedures," *Utility, probability, and human decision making*, pp. 47–85, 1975.

[40] J. R. S. C. Mateo, "Multi-attribute utility theory," in *Multi Criteria Analysis in the Renewable Energy Industry*, pp. 63–72, Springer, 2012.

[41] T. Hesterberg, "Weighted average importance sampling and defensive mixture distributions," *Technometrics*, vol. 37, no. 2, pp. 185–194, 1995.

[42] R. de FSM Russo and R. Camanho, "Criteria in ahp: a systematic review of literature," *Procedia Computer Science*, vol. 55, pp. 1123–1132, 2015.

[43] Z. A. Bukhsh, I. Stipanovic, A. Hartmann, and G. Klanker, "Evaluation and application of ahp, maut and electre iii for infrastructure management," in *6th International Symposium on Life-Cycle Civil Engineering, IALCCE 2018*, pp. 1467–1474, CRC Press/Balkema, 2019.

[44] A. Dhillon, "Political parties and coalition formation," *G Demange, G and M. Wooders, M.(Eds), Group formation in economics: Networks, Clubs and Coalitions, Cambridge University Press, New York*, pp. 289–311, 2005.

[45] B. Steunenberg, "Coalition theories: Empirical evidence for dutch municipalities," *European Journal of Political Research*, vol. 22, no. 3, pp. 245–278, 1992.

[46] E. C. Browne, "Testing theories of coalition formation in the european context," *Comparative Political Studies*, vol. 3, no. 4, pp. 391–412, 1971.

[47] C. Crombez, "Minority governments, minimal winning coalitions and surplus majorities in parliamentary systems," *European Journal of Political Research*, vol. 29, no. 1, pp. 1–29, 1996.

[48] M. Leiserson, "Factions and coalitions in one-party japan: An interpretation based on the theory of games," *American political science review*, vol. 62, no. 3, pp. 770–787, 1968.

[49] J. Bakker, A. Hammond, D. Bloembergen, and T. Baarslag, "Rlboa: A modular reinforcement learning framework for autonomous negotiating agents," in *Proceedings of the 18th international conference on autonomous agents and multiagent systems*, pp. 260–268, 2019.

[50] T. Baarslag and M. Kaisers, "The value of information in automated negotiation: A decision model for eliciting user preferences," in *Proceedings of the*

16th conference on autonomous agents and multiagent systems, pp. 391–400, 2017.

[51] T. Louwerse, S. Otjes, C. M. v. Vonno, and H. Van der Veer, "Dutch Parliamentary Voting Dataset," 2017.

[52] I. Sened, "A model of coalition formation: Theory and evidence," *The Journal of Politics*, vol. 58, no. 2, pp. 350–372, 1996.

[53] R. G. Pontius, O. Thontteh, and H. Chen, "Components of information for multiple resolution comparison between maps that share a real variable," *Environmental and Ecological Statistics*, vol. 15, no. 2, pp. 111–142, 2008.

[54] T. Baarslag, A. T. Alan, R. Gomer, M. Alam, C. Perera, E. H. Gerding, and M. Schraefel, "An automated negotiation agent for permission management," in *Proceedings of the 16th conference on autonomous agents and multiagent systems*, pp. 380–390, 2017.

[55] R. T. Marler and J. S. Arora, "The weighted sum method for multi-objective optimization: new insights," *Structural and multidisciplinary optimization*, vol. 41, no. 6, pp. 853–862, 2010.

[56] M. Weisberg, "Robustness analysis," *Philosophy of science*, vol. 73, no. 5, pp. 730–742, 2006.

[57] M. Zeaiter, J.-M. Roger, V. Bellon-Maurel, and D. Rutledge, "Robustness of models developed by multivariate calibration. part i: The assessment of robustness," *TrAC Trends in Analytical Chemistry*, vol. 23, no. 2, pp. 157–170, 2004.

[58] S. P. Jones, *Designs for minimizing the effect of environmental variables*. PhD thesis, The University of Wisconsin-Madison, 1990.

[59] S. H. Orzack and E. Sober, "A critical assessment of levins's the strategy of model building in population biology (1966)," *The Quarterly Review of Biology*, vol. 68, no. 4, pp. 533–546, 1993.

[60] A. S. Luna and J. S. de Gois, "Application of chemometric methods coupled with vibrational spectroscopy for the discrimination of plant cultivars and to predict physicochemical properties using r," in *Comprehensive analytical chemistry*, vol. 80, pp. 165–194, Elsevier, 2018.

[61] T. Louwerse, "Peilingen in het nieuws: van hypes naar trends," *Beleid En Maatschappij*, vol. 40, no. 2, pp. 176–179.

[62] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley interdisciplinary reviews: computational statistics*, vol. 2, no. 4, pp. 433–459, 2010.

[63] B. Meyer, "Agile," *The good, the hype and the ugly*, vol. 1, 2014.

[64] J. Forcier, P. Bissex, and W. J. Chun, *Python web development with Django*. Addison-Wesley Professional, 2008.

[65] S. bin Uzayr, N. Cloud, and T. Ambler, "Angular," in *JavaScript Frameworks for Modern Web Development*, pp. 209–223, Springer, 2019.

[66] T. P. Debray, J. A. Damen, K. I. Snell, J. Ensor, L. Hooft, J. B. Reitsma, R. D. Riley, and K. G. Moons, "A guide to systematic review and meta-analysis of prediction model performance," *bmj*, vol. 356, 2017.

[67] N. R. Jennings, P. Faratin, A. R. Lomuscio, S. Parsons, C. Sierra, and M. Wooldridge, "Automated negotiation: prospects, methods and challenges," *International Journal of Group Decision and Negotiation*, vol. 10, no. 2, pp. 199–215, 2001.

[68] T. Sandholm, "Agents in electronic commerce: Component technologies for automated negotiation and coalition formation," *Autonomous Agents and Multi-Agent Systems*, vol. 3, no. 1, pp. 73–96, 2000.

[69] E. Winter, "The shapley value," *Handbook of game theory with economic applications*, vol. 3, pp. 2025–2054, 2002.

[70] S. Hart, "Shapley value," in *Game theory*, pp. 210–216, Springer, 1989.