

Moving window approach for detecting change points in relational event data

Applied Data Science, MSc
Graduate School of Natural Sciences, Utrecht University
July 1, 2022

Author: Sterre Rood (6545475)
First supervisor: Dr. Mahdi Shafiee Kamalabad
Second supervisor: Prof. dr. ir. Vincent Buskens



Abstract

Recently the Relational Event Model (REM) has become a popular modeling tool for analyzing the social network over time. The REM is a flexible model that can incorporate exogenous and endogenous statistics. In the REM, the assumption that the effects in a REM are constant over the entire event history is restrictive. In real life, there are many situations in which the effects can change over time, where social interaction behavior can vary greatly between periods. For example, critical situations can fundamentally alter the interaction between people. This study looks at whether a moving window approach can detect change points. Change points are points in time where the effects of interactions change greatly. To find possible change points with a moving window approach, data from NASA's unsuccessful mission to the moon is used: Apollo 13. Additionally, the Bayes Factor is applied to statistically test whether there is evidence for some of the change points found by the moving window approach.

1 Introduction

With the increasing availability of social network data, there is a growing interest in how networks evolve. A social network is a network that reflects the social structure made up of individuals in society, their relationships among the individuals and other social interactions among individuals (Peng et al., 2018). Social networks can widely vary from one another. For example, social networks can be networks of friendships, they can cover communication between people, and social networks can be formed online. Social interaction is an essential element of social networks. In simple terms, social interaction is an event which changes the behavior of two or more individuals. Through social interactions, we construct relationships, trade information, and fulfill our essential desires for social belongingness (Meijerink-Bosman et al., 2022). Social interaction can be studied between two individuals or among larger groups.

Social interactions are driven by complex mechanisms where each interaction is not only affected by the characteristics of individuals or the environmental context but also by history of interactions (Butts, 2008). There are various approaches regarding what drives social interactions; this paper focuses on the REM.

The REM is a relatively new approach to studying social interaction in social networks. The REM was developed in the social sciences by Butts (2008) to investigate the timing of events in interactions, such as conversations or communications. It is different from other methods for social network analysis, in the sense that it models the event dynamics directly, examining how the past relational events influence future relational events (Foucault Welles et al., 2014). In contrast, other methods lose the temporal aspect of the data by aggregating the events (Tranmer et al., 2015). The REM thus directly analyzes relational event data without aggregating the events. Therefore, it is suitable for studying how social interaction evolves. The REM models *who* are involved in the interaction and *when* the interaction takes place. As a result, the model can incorporate cognitive, behavioral, and social processes.

Various studies have been done on social interaction, including the REM. Foucault Welles et al. (2014) studied online communication networks using relational event network analysis. Using an online chat network from the virtual world *Second Life*, they found different communication patterns among nonfriends and friends within the network. Another study illustrated the empirical application of REM in the context of a free/open-source software project to explain the level of effort produced by contributors to the project (Quintane et al., 2014). Pilny et al (2016) studied group interaction patterns. They concluded that the REM helps

understand group members' different patterns during interaction. The study of Meijerink-Bosman et al. (2022) illustrates that the REM can highlight the effect of personality and personal and interpersonal characteristics on how first-year students interact in a situation where the first-year students are unfamiliar to each other. For example, they illustrate that students who interacted more frequently in the past are likely to interact more frequently in the future. Furthermore, they illustrate that there is a strong preference for same-age group interaction.

These studies all show that the REM is helpful in the study of social interaction in social networks. However, the REM assumes that the effects on social interaction are constant over the study period. Yet, it makes more sense that these effects in real-world situations change over time. For example, consider an emergency; in such situations, the interaction between people will most likely change compared to before the emergency. The REM cannot discover these changes in interaction. Therefore, this study considers the moving window approach (Mulder & Leenders, 2019), in addition to the REM, to study social network dynamics. In this approach, a window of a specific length slides over the relational event sequence. In each slice, the REM is implemented to the subset of events that falls within the window. The moving window approach might enable us to see how the effects of social interaction change over time (Meijerink-Bosman et al., 2022).

A few studies have implemented the REM with a moving window approach. Mulder & Leenders (2019) illustrated a moving window approach by analyzing streams of email messages between employees. They showed that exogenous effects substantially change over time. Exogenous effects refer to variables outside the relational event history; think of age and geographical location. Whereas endogenous statistics refer to characteristics of past interaction; think of the volume of past interactions. For example, they found that similarity in geographic location affects the interaction process differently over a year.

Meijerink-Bosman et al. (2022) illustrates how the basic REM could be extended with a moving window approach to study how the effects of social interaction processes change over time. The data consists of face-to-face and digitally mediated interactions among first-year students. They expect to see some changes in how first-years develop their new persona as a student and how first-years interact. The study shows that older first-year students do not seek each other out during the first week. After this first week, older first-year students have a preference towards connecting with each other. Furthermore, the study shows that first-year students tend to interact more actively during the weekdays, compared to the weekends.

The studies using a moving window approach combined with the REM show that the effects of interaction change over time. Thus, the assumption of the REM that the effects of the interaction are constant does not always hold. These studies' focus was to understand social interaction dynamics better. To elaborate on this, this study looks at change points in social interaction. Change points are points in time where the effects of the interaction change greatly (Shafiee Kamalabad et al., 2018; Shafiee Kamalabad & Grzegorzcyk, 2019). For example, when people are communicating, and they end up in an argument, the manner of the communication may change. The ability of a moving window approach to detect such change points, time points where the strength of the effects strongly changes, is what this study focuses on. When a moving window approach can detect change points, a statistical test is applied to see if there is evidence for some of the change points found by the moving window approach. It is tested whether the effect before the change point is different from the effect after the change point. Whether a moving window approach can detect such change points in relational event data has not been applied before to current knowledge. Thus, this study goes beyond better understanding the dynamics of social interaction, as in previous studies. The research question follows: *is a moving window approach able to detect change points in relational event data?*

Relational event data based on NASA's mission to the moon is used to answer this question. Critical situations in this failed mission to the moon occurred in which communication patterns fluctuated greatly.

The paper is structured as follows. The next section discusses the Apollo data and discusses how the data was prepared. Section [3](#) introduces the different methods. In section [4](#), the results are presented. The paper ends with a conclusion and discussion in section [5](#).

2 Data

The relational event data in this study comes from NASA's unsuccessful mission to the moon, Apollo 13. The data contains voice-loop data from the seventh crewed mission in the Apollo space program. The mission of Apollo 13 would be to conduct a geological survey of the moon's surface. The space flight and moon landing should not have been difficult because of Apollo's experience. On April 11, 1970, the craft was launched. After over two days, an oxygen tank explosion in the service module forced the crew to abandon all thoughts of reaching the moon. What followed was to find a way to bring the astronauts back home safely (Wikipedia contributors, 2022). The communication patterns among the crew and the communication patterns between the crew and mission control remained steady until the accident. Because it turns out that change in communication patterns occur in highly critical situations (van den Oever & Schraagen, 2021), it is likely that communication patterns after the accident were different from those before the accident. This change in communication patterns might be important to overcome the never-before-seen situation. Therefore, the Apollo 13 data is interesting to identify any possible change points.

2.1 Data collection

The data is collected from two text files, 'air-ground-loop.compact.txt' and 'flight-director-loop.compact.txt'. 'air-ground-loop.compact.txt' is the transcript of the air-ground loop, and 'flight-director-loop.compact.txt' is the transcript of the flight-director loop. Furthermore, data from a website containing the Apollo Flight Journal covering the flight from launch to splashdown is collected. Together, these three sources form the data of Apollo 13.

The tasks of collecting the data have been divided in this study. However, in this paper, both processes will be described for completeness of understanding.

The two text files are accessible through [GitHub](#). These two data files are available for download and could thus be collected without difficulty. However, with the Apollo Flight Journal data from the [website](#), it took a little more to collect the data.

The data from the Apollo Flight Journal website consists of more than twenty web pages of communication transcripts between the air team and the ground team of Apollo 13. Python is used for the web scraping of this data. The URL is saved in a list for each webpage, which is then looped through. Using the package 'beautifulsoup4', the data of the communication transcripts is taken from the URLs. Two empty lists are created to store the data. In one list, all the messages are stored. In another list, all the times and names are stored. The preferred way

to scrape data from a webpage is by using the class of the items. However, the messages do not belong to any class.

For this reason, Regex patterns are created to scrape the messages. These Regex patterns take all the messages, times, and names from the webpages. With all the data collected, the data needs to be cleaned up before it can be used.

2.2 Data cleaning

As with collecting the data, the tasks for cleaning the data are also divided. The data cleaning is divided into two parts: cleaning the text files and cleaning the web scraped data. In this paper, both processes will be described for completeness of understanding.

2.2.1 Text files

The two original text files are two quite unstructured files. Generally, each line represents an event with the time, sender, and message. Each line separates the time, sender, and message with multiple, irregular whitespaces. Furthermore, random new lines quite often appear within and between events.

The first part of the data cleaning is done in Python. Loading the files in Python led to some trouble due to the irregular, random whitespaces. For this reason, the text files are first opened in excel. In excel, three columns are created: time, sender, and message. This data is stored as a CSV file. This solved the problem with irregular whitespaces. The CSV file can now be loaded into Python, with a semicolon as delimiter.

In addition to the irregular whitespaces, there are irregular new lines between and within various events. The irregular new lines *between* events had to do with different senders talking simultaneously. It is easy to recognize when senders are talking interchangeably. As said before, each event consists of the time, the sender, and the message. If there is a line with values for time, sender, and message, this points to one event.

In most cases, the subsequent line has different values for the time, sender, and message and thus points to a different event. However, if the succeeding row has a missing value for time but a value for sender and message, this indicates that multiple senders are talking simultaneously. For clarity, see Table [1](#). In the case of relational event data, there can only be one event per time. Thus, a problem has arisen here. The rows where multiple senders are talking simultaneously are manually checked to solve this problem. Then the missing value for the time was replaced with the time in the previous line. This leads each sender to have a time-based value. Note that this leads to duplicate values for the times in the data, in addition to the

duplicate values for the times that are already in the data. These duplicate values are not allowed in relational event data.

Before dealing with duplicate times, the time structure was converted from a hh:mm:ss format to seconds. Converting to seconds is done using a function which first splits and separately saves the hours, minutes, and seconds. Seconds are calculated by multiplying the hours times 3600 plus multiplying the minutes by 60 plus the seconds. This was followed by using a for loop to replace all times with seconds.

After all the values are converted to seconds, the duplicates must be handled. This is done through two for loops. In short, the first for loop goes through all the column 'time' values. The current time is stored in a list starting from the first time. If the current time is already in the list, the time plus one second is stored in the list. In the second for loop, the original times are replaced by the times from the list. With the help of indexes, the times from the list replace the original times in the data. Because a time sometimes occurs more than twice, some duplicates remain after running the for loops once. For example, $T1 = T2 = T3$. In this case, T2 and T3 both get one second added. This leads to T2 and T3 still being duplicates. Running it one more time will add one more second to T3. To solve this, the two for loops are run multiple times, which ensures that the data contains no duplicates.

Another problem with the irregular new lines is that sometimes new lines occur *within* one event. This causes one message to be on different lines. This can be recognized when lines have a missing value for both time and sender. This means that a sender's message is spread across several lines. For clarity, see Table [2](#). To get the message in one line with the corresponding time and sender, the rows belonging to the same sender, time and message are merged.

Finally, there was one off loop in the time, sender, and messages. This has been removed as it does not provide any relevant information. All these steps were done separately for both text files. These two files were merged and checked one more time for duplicates.

Table 1*Part of the unstructured data. Senders are talking interchangeably*

Time	Sender	Message
55:53:07	CMP	Okay.
	CAPCOM	- for looking at the Comet Bennett, if you need it.
56:13:36	LMP	What buses?
	CAPCOM	We'd like - on MAIN A, we'd like Charlie 1, 2, 3, and 4. Also Bravo 3 and 4 on MAIN A.

Note. The receivers have not been added yet. The receivers will be added to the structured data.**Table 2***Part of the unstructured data. A sender's message is spread across several lines*

Time	Sender	Message
55:49:25	CMP	Okay, Jack. During the TV we were AUTO TRACK, NARROW you
		BEAM WIDTH, and the PRIMARY ELECTRONICS. And we had a
		good lockup. Just after we started the maneuver, I was able to lock
		up and get real good signal strength, and it just seemed that right
		there at about 239 \rightarrow ∞ in yaw, that the signal strength would just drop
off and yaw would go to 0 and pitch would go to 90.		

Note. The receivers have not been added yet. The receivers will be added to the structured data.

2.2.2 Web scraped data

The data collection section discussed that Regex patterns were used to scrape the data from the webpages. However, these Regex patterns have also scraped data that is irrelevant to this study. Therefore, the data needs to be cleaned.

First, the messages that start with 'T-' are removed. These messages took place before the launch of the craft. Since in this study only the events starting from the launch are of interest, these prelaunch messages are removed.

Second, the two lists containing messages, times and names need to be stored as columns in a data frame. Therefore, the two lists need to be of equal length. However, this was not the case, meaning that there is redundant data in the lists. After reviewing the web pages and the data, it turned out to be data that was not related to the conversations. After removing this data,

both lists are of equal length. Both lists are stored in a data frame. The columns are named 'message' and 'timename'. The values in the column 'timename' consist of the times and the names. Based on whitespaces, the values in the column are split over a maximum of four columns. When the name of one actor consists of several parts, the '_' sign is used to bring it together as one name. The original 'message' and 'timename' columns ended up being split into three columns: one column containing the time, one column containing the messages, and one containing the sender. These columns are all in one data frame

Third, the times' hh:mm:ss format has been converted to seconds. The number of hours is multiplied by 3600. The number of minutes is multiplied by 60. This is added together along with the remaining seconds.

Fourth, the web scraped data contains the actors' names as senders. These names have been converted to the appropriate abbreviations to align with the two text files. The abbreviations come from a [GitHub](#) page. A glossary with the descriptions and meanings of the abbreviations is available here. For example, the name 'Liebergot' has been converted to 'EECOM'. Not all names were represented in this glossary. These names have been looked up on the Internet and have been replaced with the appropriate abbreviations. Some names with the corresponding abbreviations were also found in [NASA's descriptions](#).

Finally, there were some gaps in time when no one knew what was happening. These gaps in time are called long common breaks. These long common breaks had to be taken care of. Before the long common breaks could be taken care of, a manual search through the web pages had to be carried out to find the locations of these long common breaks. For each event after a long common break, the index was stored. Removing the long common breaks was broadly done in four steps.

The first step was to calculate the duration of the long common break and extract this duration from each time after the long common break. The duration of the long common break was calculated by taking the difference in time between the event after the long common break (the index) and the event before the long common break. The second step was to take care of the gap that remains between the event before the long common break and the event after the long common break (the index). This is taken care of by replacing the time of the event after the long common break (the index) by the time of the event before the long common break. The third step was calculating the mean difference in time for the five events before the long common break. This mean difference was added to the time of the event after the long common break (the index). The fourth step was to ensure the times after the long common break were in

sequential order again. For this reason, the mean difference was also added to the times after the long common break. This keeps the order of events, and the times correct.

This process was repeated for each long common break using a for loop. This yielded new values for the times unaffected by the long common breaks. For greater clarity, an example of the process of removing a long common break is shown in Table 3.

Table 3

Process of removing the long common breaks using a moving average technique

Original times	First step	Second step	Third step	Fourth step
910	910	910	910	910
934	934	934	934	934
942	942	942	942	942
944	944	944	944	944
952	952	952	952	952
955	955	955	955	955
<i>1299</i>	<i>1299</i>	955	964	<i>964</i>
1302	958	958	958	967
1304	960	960	960	969

Note. Times are in seconds. The times in italics represent the index of the long common break. Changes are represented in bold times.

2.2.3 Complete data by merging the text files and web scraped data

At this point, there are three separate data frames. These three data frames contain the columns time, sender, and message. One critical column missing in this relational event data is the receiver column. In most cases, the sender and receiver talk alternately. Starting from this pattern, the receivers are added to each data frame using a for loop. However, this alternate speaking does not apply everywhere. Furthermore, a few times the receiver did not answer until a few lines later. This meant that the receivers had to be checked manually because a structured pattern was absent.

When, during the manual check of the receivers, it was seen that there were multiple receivers, the receiver that responds (first) is designated as the receiver. In addition, there was no sender in a few cases, but there were question marks. First, a check was made to see if these events might have occurred in one of the other data sources. Unfortunately, all these events

without a sender did not appear in it. As a result, a guess was made about who would be the best fitting sender. After all the receivers have been added to the three data frames, the data frames are merged. This could easily be done because they all contained the same four columns: time, sender, message, and receiver. Finally, one more check was done for duplicates in this merged data.

2.3 Data pre-processing

For the REM, an edgelist is required as data. This edgelist can be at least a three-column data frame or a matrix. In this edgelist, each row must contain the time, sender and receiver. These should refer to for the edgelist as ‘time’, ‘actor1’, and ‘actor2’. The senders and receivers must be represented as integers from 1 to n. Because our data consists of names of senders and receivers, all senders and receivers were converted from strings to numerical values. For example, "CAPCOM" is converted to "2". The messages are irrelevant to the model, so this column is dropped. This results in a data frame with times, senders (in numeric values), and receivers (in numeric values). This data frame is the definitive data for the REM. The data frame's first six and last six events are shown in Table 4 and 5.

Table 4

First six events in the edgelist

Time	Actor1	Actor2
1	2	1
2	1	2
3	7	1
5	1	7
10	2	1
11	1	7

Table 5

Last six events in the edgelist

Time	Actor1	Actor2
198654.1	24	2
198660.1	24	26
198693.1	26	24
198709.1	20	24
198774.1	20	24
199080.1	24	20

2.4 Ethical considerations

To the best of my knowledge, the data have no ethical concerns. The data were publicly available and do not contain any sensitive information unrelated to the communication during NASA’s mission.

3 Methodology

3.1 Relational event model

As said before, the REM is suitable for studying social interaction in social networks. The REM models *who* are involved in the social interaction and *when* the social interaction takes place.

The central element in the REM is the relational event. A relational event is defined as "a discrete event generated by a social actor and directed towards one or more targets" (Butts, 2008). For example, astronaut X sends a message to astronaut Y, at a specific time point T. The composition of a relational event is made up of the actor who sends the message (the sender), the actor to whom the message is directed (the receiver) and the time of the message. The sender, receiver, and time is the minimum information a REM requires. A single relational event is defined as a tuple containing the sender, the receiver and the time of the event $a = (s,r,t)$. In short, relational events are interactions between people at specific times (Butts, 2008).

The set $A_t = \{a_i : \tau(a_i) \leq t\}$ consist of all events taken on or before time t . The interest of the REM is in modeling the probability of relational events occurring at a certain point in time. The core of the REM is the rate parameter λ . This rate parameter determines when the next social interaction will take place. Besides that, it determines which actors are involved in this social interaction. To apply this rate parameter, defining the risk set $R(t)$ is necessary. This risk set contains all the events that can potentially occur at time t . This risk set often consists of all possible directed or undirected pairs (s, r) of actors (Meijerink-Bosman et al., 2022). The probability of an event to occur between sender s and receiver r at time t is given in equation 1, which follows a multinomial distribution. The probability of an event to occur is equal to the rate parameter of the event (sender s , receiver r) relative to the rate parameters of all the events in the risk set at time t . Thus, common events have higher rates, and less common events have lower rates.

$$P((s, r)|t) = \frac{\lambda(s,r,t)}{\sum_{s',r'} \lambda(s',r',t)}, \quad (1)$$

The rate parameter depends on sender, receiver, and past event history. In addition, the rate parameter can be flexibly deployed to include a broader range of endogenous and exogenous predictors. These predictors are referred to as statistics (Butts, 2008). Given the social context, these endogenous and exogenous statistics represent behavioral and cognitive mechanisms that lead actors to interact in events more frequently than others (Pilny et al., 2016). Exogenous statistics refer to any variable outside the relational event history itself; think of age and

geographical distance. Endogenous statistics refer to characteristics of past interaction; think of the number of messages an actor has received or the number of messages an actor has sent (Meijerink-Bosman et al., 2022). Some statistics that can be included in the rate parameter are inertia, reciprocity, and introversion. For example, inertia means that the more messages the capsule communicator sent to the crew in the past, the higher the likelihood of the capsule communicator sending a message to the crew in the future. The function for the rate parameter is given in equation 2. Where λ is the rate parameter which is estimated for each event (sender s , receiver r) at time t . This rate parameter λ is a log-linear function of the statistics. The model parameter b_i denotes the magnitude of the effect of the statistics x_i on the rate parameter.

When a realized relational event is observed, a sender talks to a receiver, the probability for an event to occur may change; this is because the occurrence of the particular event might have changed the state of the world. Consequently, the risk set and the statistics are updated to reflect the new network structure. Therefore, the rate parameter is modelled as a function of historical information, in addition to some other statistics (Pilny et al., 2016; Butts & Marcum, 2017).

$$\log \lambda (s,r,t) = b_1 x_1(s,r,t) + b_2 x_2(s,r,t) + b_3 x_3(s,r,t) + \dots \quad (2)$$

Important assumptions for the REM are that events should be temporally ordered, and that the onset of the observation period should be exogenously determined. The onset is the time from where a series of relational events are observed. This onset of the series of relational events should be chosen by the researcher or by a random external event (e.g., exogenously). In this study, the specific time points for the relational events are defined in seconds relative to the onset of the observation period. Additionally, the rate parameter is a piecewise constant function. This means that the rate parameter is assumed to remain constant, with changes occurring either when an endogenous event is realized or at exogenous events (Butts & Marcum, 2017).

The REM can handle continuous time or ordinal time. Where the order in which the events occur is known, but the exact timing is unknown, the term ordinal time is used. When the exact timing between the events is known, the term exact or interval time is used. In the first case, the ordinal likelihood is used, whereas the full likelihood is used in the latter. The Apollo 13 data consists of data where the exact timing of events is known. Thus the full likelihood will be used.

3.2 Model specification

Before a REM can be implemented, it must be decided which statistics will be included. A statistic in a REM can be endogenous and exogenous. As discussed before, exogenous statistics refer to any variable outside the relational event history itself and endogenous statistics refer to characteristics of past interaction. Because of the lack of exogenous statistics available for the Apollo 13 data, only endogenous statistics are considered in this study.

Inertia. Previous research shows that individuals who have interacted often in the past are to be expected to continue interacting in the future. For this reason, FrPSndSnd (the fraction of v 's past actions directed to v' affects v 's future rate of sending to v') will be included. This statistic considers the tendency for actor a to send a message to actor b , based on a 's prior messages sent to b .

Reciprocity. Previous research shows that individuals who have received messages from some sender are also more likely to return messages to that sender in the future. For this reason, FrRecSnd (fraction of v 's past receipt of actions from v' affects v 's future rate of sending to v') says something about reciprocity. This statistic considers the tendency for an actor a to send a message to actor b , based on b 's prior messages sent to a . Both inertia and reciprocity treat the previous contacts as a predictor of future contacts.

Participation shifts. There are a lot of possible participation shifts that can occur during a conversation. There are turn receiving, turn claiming, turn usurping, and turn continuing participations shifts. In the case of Apollo 13, a turn receiving participation shift is especially relevant. The turn receiving participation shift considered here is the AB-BA participation shift. The AB-BA participation shift captures the tendency for B to call A, given that A has just called B. In the AB-BA participation shift, the receiver of the initial event is a potential sender, and the receiver of the initial event is also the sender of the subsequent event. In the Apollo 13 data, it appears that this turn-receiving effect is quite common. In most of the events, the sender and receiver talk alternately. For this reason, it is important to include this turn-receiving effect.

Preferential attachment. The preferential attachment mechanism can be explained as a few actors doing most of the communication. The actors who have the most speaking time are the most attractive targets for others to communicate with. Butts (2008) argued that decisions about potential communication targets might be highly uncertain when events happen in a turbulent environment. For example, at the time of an incident, one may not be sure who can respond. In such an incident, it is natural to use earlier communication patterns as a predictor of current availability: those involved in earlier communication are more likely to be present and able to respond than those with no earlier communicative activity. This phenomenon is an

example of preferential attachment (Butts, 2008). Looking at Apollo 13, the capsule communicator connects the ground team and the astronauts. This is because the ground team cannot talk directly to the astronauts and vice versa. All communication goes through the capsule communicator. For this reason, the capsule communicator speaks a lot. Therefore, the capsule communicator might more likely be chosen as a communication target. Preferential attachment can be investigated by including statistics concerning normalized total degree (Butts & Marcum, 2017). The statistics that say something about preferential attachment are *NTDegSnd* and *NTDegRec*.

Shared partners. It is important to consider the potential impact of shared partners on interactions. For instance, two actors who have contacted the same third parties may be more or less likely to contact one another directly; this is referred to as an outbound shared partner effect (Butts, 2008). Looking at Apollo 13, it may be the case that the astronauts (e.g., CDR and LMP) communicated with the capsule communicator. This may have caused the astronauts to communicate more or communicate less with each other in the future. Similarly, we can imagine the effect of having been contacted by the same third party, which is referred to as an inbound shared partner effect. Looking at Apollo 13, this could be that the astronauts both received messages from the capsule communicator. In this case, the capsule communicator is a shared partner for the astronauts. This may make the astronauts more or less likely to communicate with each other in the future. The statistics that measure the effect of shared partners are *OSPSnd* and *ISPSnd*.

Of course, there are many more statistics that may be relevant. However, in this study, only seven statistics are considered. These seven statistics cover inertia, reciprocity, turn-receiving participation shift, preferential attachment, and shared partners. The ‘remstats’ package is used to compute the statistics for the model. The remstats package provides each object necessary to estimate the REM with the ‘relevent’ package.

3.3 Moving window approach

As it was discussed, this study considers the moving window approach. This is because social interaction effects change over time rather than being constant. This study aims to see whether a moving-window approach can detect change points in the relational event data. Through this moving window approach, it is possible to see if and how the coefficients of the statistics change over time.

Before possible change points can be investigated, the optimal length of the window and the optimal overlap of the window should be determined. Because these are not known yet, the

first objective of this study is to find the optimal length of the window and the optimal overlap of the window. This paper focuses on the optimal overlap of a window, whereas the other paper focuses on the optimal length of a window.

Since the optimal length of the window is not yet known, three fixed window lengths are specified: 5 hours, 12 hours, and 24 hours. The 5-hour window is a narrow window length, the 12-hour window is a medium window length, and the 24-hours window is a wide window length. The effect estimation will be more sensitive to each point in time for smaller window lengths than narrow ones. For this reason, a narrow, medium, and wide window length is chosen. The overlap between windows determines the smoothness of the estimation of the effects. A greater overlap between windows leads to greater smoothness, whereas a smaller overlap between windows leads to less smoothness (Meijerink-Bosman et al., 2022). To accommodate different degrees of smoothness, six different overlaps are considered: 10%, 25%, 33%, 50%, 66% and 75%. For example, the 25% overlap indicates that the window is moved such that it overlaps with 25% of events of the previous window but also contains 75% new events. Since one of the focuses of this model is to find the optimal overlap, more overlaps than window lengths were explored. It is important to note that there should be enough events within each window. If there are not enough events in a window, the estimation of the effects may become unstable (Mulder & Leenders, 2019).

The REM is fitted on the subset of relational events that took place in the first period of a window. Then the window is moved according to the overlap, such that it partly overlaps with the previous window but also contains a new subset of relational events. Subsequently, the REM is fitted to this new subset of relational events. These steps are repeated until the last period of the relational events. In each step fitting the model, the coefficients of the statistics are saved in a separate data frame for each window length with a specific overlap.

To choose the optimal overlap for the moving windows, the Bayesian Information Criterion (BIC) is used. The BIC is generally used for model selection, where the model with the lowest BIC is considered the best. The REM will be implemented for each window with a specific length and a specific overlap. For example, the 5-hour window with a 50% overlap can consist of twelve different windows. In that case, the REM is implemented on all twelve windows. For each of these windows, the BIC of the model will be saved in addition to the coefficients for each statistic. In this example, this will result in a data frame with 12 rows, where the columns contain the statistics and the BIC of the model. This process is performed for each combination of window length and window overlap. To determine which window overlap is optimal, the

average BIC is considered. This results in a BIC for, say, a 5-hour moving window with 25% overlap. A BIC for a 5-hour moving window with 50% overlap. A BIC for a 12-hour moving window with 25%, etcetera. The moving window that eventually will be used is chosen according to the lowest BIC.

3.4 Bayes Factor for change point detection

This study looks at how the coefficients for each statistic change over time. At time points where the coefficients of the statistics strongly change, there might be a change point. When such a change point is visible in a moving window approach, evidence for a change point is gathered through the Bayes Factor. The Bayes Factor is the relative evidence for one hypothesis over another hypothesis. The Bayes Factor is defined as the ratio of the posterior odds to its prior odds. The Bayes Factor is given in equation 3 (Kass & Raftery, 1995).

$$BF_{1,0} = \frac{p(D|H_1)}{p(D|H_0)} \quad (3)$$

This study should evaluate whether the data provides more support in favor of the change point than against the change point. To prove that there is a change point, the coefficient of a specific statistic (β_i) before the change point at time t must be different from the coefficient of a specific statistic (β_i) after the change point at time t . These two coefficients will be obtained by fitting the REM to the data before and after the change point. Two hypotheses follow (Shafiee Kamalabad et al., in review):

H_0 β_i before the change point at time $t = \beta_i$ after the change point at time t

H_1 β_i before the change point at time $t \neq \beta_i$ after the change point at time t

How to approximate the Bayes Factor in this study is given in equation 4. Where $N(0|m, \Sigma)$ denotes the normal density with mean m , and covariance matrix Σ . $\hat{\zeta}$ denotes β_i before the change point at time t minus β_i after the change point at time t . The prior is approximated with a normal distribution with a mean of zero and the covariance matrix $\Sigma^{\hat{\zeta}}$ multiplied by the number of events n . The posterior is approximated with a normal distribution with the maximum likelihood estimation $\hat{\zeta}$ and the covariance matrix $\Sigma^{\hat{\zeta}}$ (Shafiee Kamalabad et al., in review).

$$BF_{1,0} = \frac{N(0|0, n\hat{\Sigma}_{\zeta})}{N(0|\hat{\zeta}, \hat{\Sigma}_{\zeta})}, \quad (4)$$

As mentioned before, statistics are calculated using the 'remstats' package. Taking reciprocity as an example, first the effects for reciprocity for each time point and each potential edges are computed. Usually, this array, with the effects for reciprocity, is the feed to the REM to get the coefficient. This array of effects needs to be adjusted to two other arrays to calculate Bayes Factor. One array with the effects for reciprocity before the change point, and the effects replaced with zeros after the change point. And one array with the effects for reciprocity after the change point, and the effects replaced with zeros before the change point. These two arrays are stored in one large array with an array of the baseline effects (intercept). This large array thus consists of one array with the baseline (intercept), one with zeros after the change point and one with zeros before the change point. Then, a REM is run with these three arrays as statistics. The coefficients of reciprocity before and after the change point are stored, along with the covariance matrix. With this information, the logarithm of the Bayes Factor can be computed.

When the $\log(\text{BF}_{1,0})$ falls between 1 and 3, one speaks of convincing evidence in favor of H_1 . When the $\log(\text{BF}_{1,0})$ falls between 3 and 5, one speaks of strong evidence in favor of H_1 . When the $\log(\text{BF}_{1,0})$ is greater than or equal to 5, one speaks of very strong evidence in favor of H_1 .

4 Results

4.1 Results REM

First, the REM is implemented on the whole data. Second, the moving windows are applied to the REM. Then, a logarithm of the Bayes Factor is computed to check whether there is evidence that a moving window correctly suggest change points. Finally, it is seen whether these change points can be linked to actual events.

The results of the REM implemented on the whole data are shown in Table [6](#). All the seven statistics are significant ($p < 0.05$). It is important to mention that these effects are significant under the assumption that they are constant over time.

The positive effect of inertia suggests that actors in Apollo 13 were more likely to send messages to other actors with whom they have communicated more in the past. The negative effect of reciprocity suggests that actors are less likely to send messages to recipients from whom they previously received many messages. It should be mentioned that these effects are very small, however, they are significant. The positive effect of the participation shift suggests that actor B has the tendency to send a message to A, given that A just sent a message to B. This shows that the actors in Apollo 13 are inclined to talk back and forth. It is interesting that reciprocity has a negative effect, while the participation shift has a positive effect. It may be that reciprocity is related to the participation shift. This might be because the participation shift is an immediate form of reciprocity, whereas reciprocity is a more distant effect. The positive effects of preferential attachment suggest that the actors who have been involved in earlier communication are more likely to be chosen as communication targets. For example, the capsule communicator does a lot of the communication. Thus, the capsule communicator is likelier to be chosen as a communication target. The positive effect of outbound shared partners suggests that two actors who have contacted the same third party are more likely to contact one another directly. Finally, the negative effect of inbound shared partners suggests that two actors who both have been contacted by the same third party are less likely to contact one another directly.

Table 6*Results relational event model*

Statistic	MLE	Standard error	<i>p</i>-value
Inertia	0.0403	0.0145	0.006
Reciprocity	-0.0318	0.0146	0.029
Participation shift	5.1977	0.0309	0.000
Preferential attachment 1	0.5676	0.0086	0.000
Preferential attachment 2	0.5074	0.0089	0.000
Outbound shared partner effect	0.1706	0.0307	0.000
Inbound shared partner effect	-0.1605	0.0323	0.000

Note. All statistics are standardized except participation shift. Participation shift could not be standardized in the ‘remstats’ package.

4.2 Results moving window approach

The assumption that all effects are constant over time is dropped in the following part. The results of applying the moving windows to the REM are shown in Table 7. The window length and the average BIC are shown for each percentage of overlap. It turns out that a moving window overlap of 10% is the most optimal since the 5-hour window with a 10% overlap has the lowest BIC. Thus, this window is chosen to see if a moving window approach can detect possible change points.

Table 7*Results relational event model*

Window overlap	Window length	Average BIC
10%	5-hours	15160.80
	12-hours	38737.59
	24-hours	66640.48
25%	5-hours	15385.23
	12-hours	36296.45
	24-hours	71253.01

33%	5-hours	15746.59
	12-hours	36587.30
	24-hours	81020.42
50%	5-hours	15843.75
	12-hours	37889.49
	24-hours	79676.69
66%	5-hours	15661.60
	12-hours	38800.31
	24-hours	36587.30
75%	5-hours	15819.92
	12-hours	40003.68
	24-hours	76138.81

As mentioned in the methods, there must be enough events within each window. For the 5-hour window with 10% overlap, the number of events within each window is shown in Table 8. The smallest number of events within a window is 284. The largest number of events within a window is 4055. According to Schecter and Quintane (2020), at least 100 events per actor should be considered for small networks to detect weak effects with sufficient power. However, when a network reaches 20 to 50 actors, and sequences are longer than 1,000 events, the model will detect most effects of the statistics with confidence. Since the Apollo 13 network consists of more than 20 actors and there are at least 1000 events within almost every window, it can be argued that most of the statistics' effects can be detected confidently. The only window that is questionable is the last window with 284 events.

Table 8

Number of events in the five-hour moving window with 10% overlap

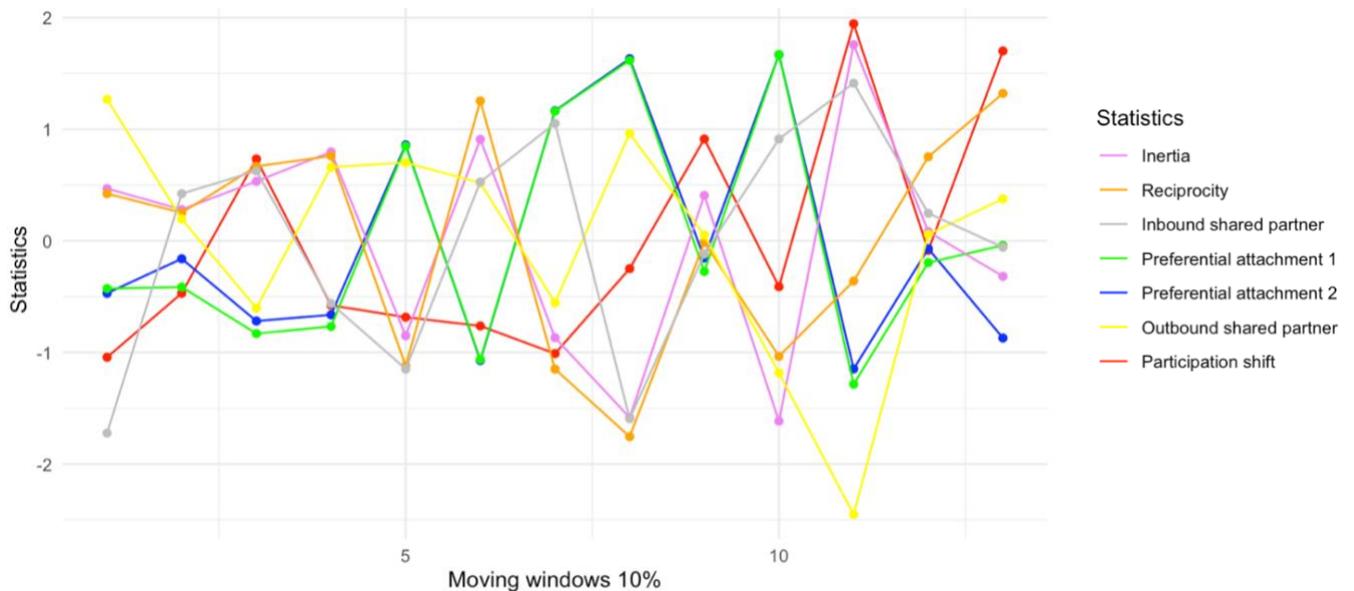
Window	Begin	End	Number of rows
1	0	18000	1628
2	16200	34200	1199
3	32400	50400	2636
4	48600	66600	4055
5	64800	82800	863
6	81000	99000	1092

7	97200	115200	958
8	113400	131400	1031
9	129600	147600	1118
10	145800	163800	945
11	162000	180000	1391
12	178200	196200	1189
13	194400	212400	284

The result of applying the moving window approach to the REM is shown in Figure 1. It is evident that communication patterns during Apollo’s mission changed over time. Regarding the change points, the biggest changes occur in the later windows. In the later windows, the effects of all statistics are experiencing great changes. For example, the outbound shared partner effect changes greatly in the eleventh window. The effects for preferential attachment change greatly in various windows like in the ninth, tenth, eleventh and twelfth window. The effect of reciprocity changes greatly in the eighth window. But also in the early windows, changes in the effects are visible.

Figure 1

Detecting change point(s)



Note. The y-axes show the change in the coefficients of the statistics. The x-axes show the moving windows and thus the entire event stream.

4.3 Results Bayes Factor for change point detection

To see if there is statistical evidence that this moving window can detect some change points, the logarithm of the Bayes Factor is computed. For the statistics preferential attachment, outbound shared partner, and reciprocity, it was tested whether there is evidence for change points. The results are summarized in Table 9. Table 9 shows the windows, the logarithm of the Bayes Factor, and the coefficients before and after the change point. Figure 2 shows the coefficient before and after the change points as a step function.

For all three statistics, during a specific window, the logarithm of the Bayes Factor is greater than 5. This implies that there is very strong evidence in favor of a change point for each statistic. This shows that communication patterns and the interaction during NASA's mission change significantly at specific time points.

Table 9

Evidence for CP for different times and statistics in a 5-hour moving window with 10% overlap

Statistic	Window	Bayes Factor	β before CP	β after CP
Reciprocity	8 th	216.47	0.51	0.47
Preferential attachment	4 th	80.14	0.99	0.90
Preferential attachment	10 th	202.40	0.95	0.81
Outbound shared partners	3 rd	57.31	0.50	0.47
Outbound shared partners	11 th	248.44	0.49	0.41

Note. The Bayes Factor is logged.

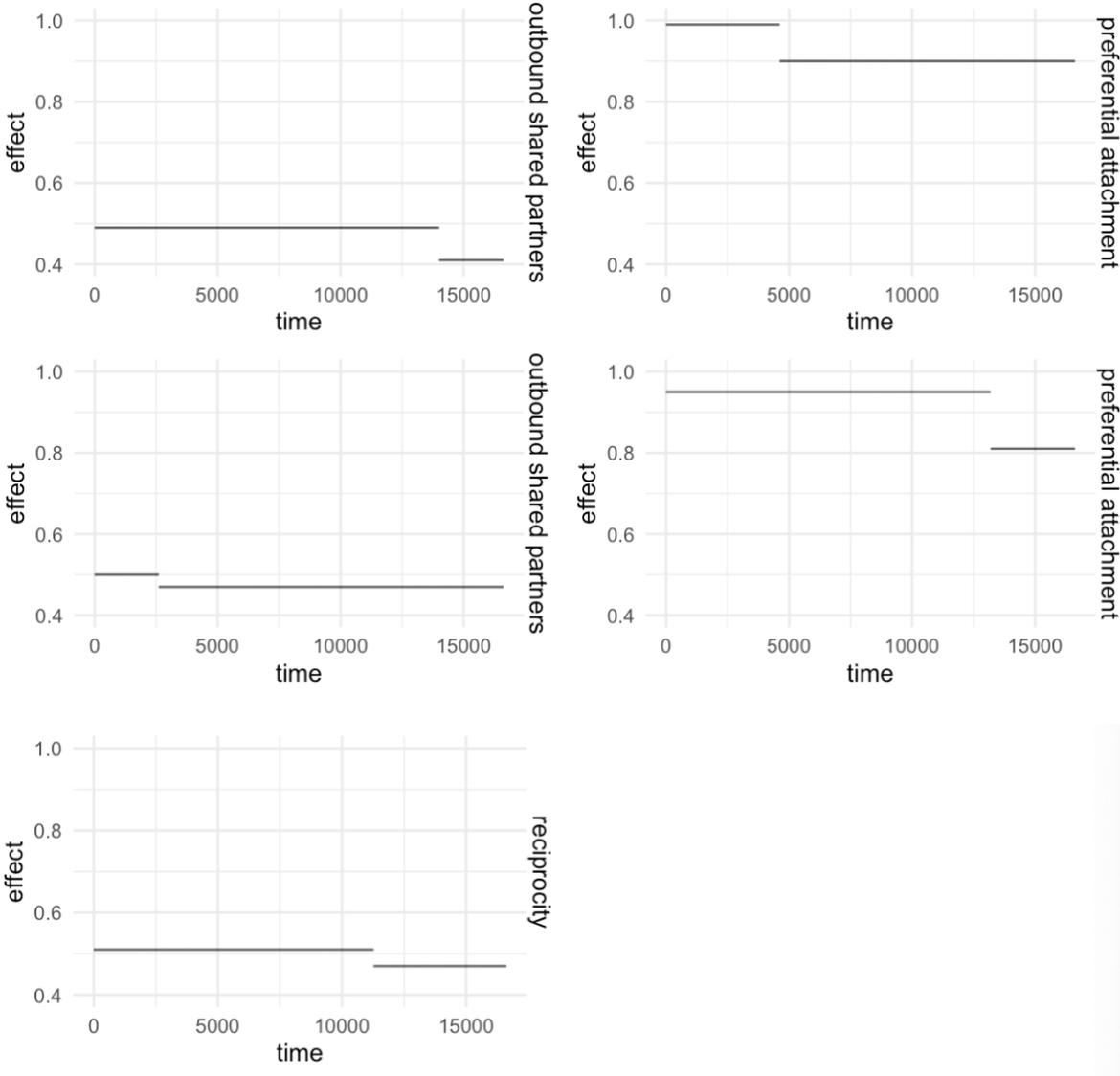
"Houston, we've had a problem," are the famous words spoken just after the explosion. One of the astronauts called mission control in Houston to report the problem 42424.8 seconds after the launch. This sentence marks the beginning of the problems experienced by Apollo 13. To overcome this unexpected situation, it is likely that communication patterns were changing. It is therefore interesting to see if some of the change points can be linked to this moment.

The event where the astronaut called mission control to report the problem occurs in the third window. The change point found for outbound shared partners in the third window can therefore to a certain extent be linked to the beginning Apollo's problems. The precise time of

the change point is not known, but it does suggest that there is a change point in outbound shared partners around the time of the explosion.

The fourth window begins not long after the astronaut called mission control to report the problem. The results show that there was a change point for preferential attachment in the fourth window. Again, the precise time point of the change point is not known, but it does suggest that communication patterns did change short after the explosion.

Figure 2
Change points as a step function for different statistics



Note. The y-axes show the effects of the statistic. The x-axes show the time in seconds and thus the entire event stream.

5 Conclusion and discussion

The research question this study sought to answer is: *is a moving window approach able to detect change points in relational event data?* We used the moving window approach for the investigation of the change points and we used the Bayes Factor for obtaining the evidence of the existence of change points detected by the moving window approach.

The results showed that a moving window approach reveals that Apollo 13's communication patterns changed greatly during the mission. Change points were visible after applying this moving window approach. Using a statistical test, it was checked whether some of these change points in the moving window were actual change points in Apollo's mission. The result of this statistical test, based on the Bayes Factor, confirmed that there is strong evidence in favor of the change points. This implies that a moving window approach is able to detect at least some of the change points. It should be mentioned that not all change points that popped up in the moving window were statistically tested. Therefore, it seems that a moving window approach can detect some change points, but it cannot be said with certainty that all change points are correctly detected.

The main advantage of this study is that the receivers were manually added to the data. The receivers were not known in advance, the only information available were the senders and the messages of Apollo's mission. The receivers were identified by looking at the senders and messages in previous events. It is worth mentioning that the communication between the ground team and the astronauts could only be done through the capsule communicator. This may affect specific effects such as preferential attachment. Suppose one of the astronauts was involved in many communications. In that case, this astronaut could still not be a target for an actor from the ground team because this communication always went through the capsule communicator.

It should be notified that this study has a few limitations. First, the statistics considered in the model are based on literature and expectations. As a result, relevant statistics may have been left out. In addition to leaving out potentially relevant statistics, it is also possible that statistics had to be taken out because of multicollinearity. It is plausible that in this study statistics are correlated. Multicollinearity may have caused the interesting and somewhat unexpected negative effect of reciprocity. For follow-up research, estimating which statistics are best to include in the model is important. This can be done, for example, using ridge regression or a subset selection method. However, this was beyond the scope of this study. Furthermore, most studies add at least nine to ten statistics.

Second, there is no method yet that finds the correct window length and window overlap for a REM. In this study, six different overlaps were compared. One of these six overlaps was chosen based on the Bayesian Information Criterion. For follow-up research, it may be relevant to find a method that can find the most optimal window overlap given the data. Also, it can be argued that the BICs cannot be compared due to the unequal number of events within each window. It is noticeable that the BICs for the narrow windows are a lot lower than the BICs for the wide windows. For follow-up research, it is important to investigate how many events and actors are needed to make the BICs comparable. Due to limited time, it was not possible to conduct this analysis for this study

Finally, a moving window approach causes the precise time of a change point not to be known, but only the window in which the change point falls. For follow-up research, it may be relevant to further investigate such a window in which a change point falls. For example, by creating a grid of time points, the precise time point of the change point may be discovered. This could lead to more accurate results. However, attention must be paid to ensure that there are still sufficient events and actors.

References

- Amati, V., Lomi, A., & Mascia, D. (2019). Some days are better than others: Examining time-specific variation in the structuring of interorganizational relations. *Social Networks*, 57, 18–33. <https://doi.org/10.1016/j.socnet.2018.10.001>
- Apollo 13 Flight Journal - Index Page*. (2020, April 6). Apollo Flight Journal. <https://history.nasa.gov/afj/ap13fj/index.html>
- Butts, C. T. (2008). 4. A Relational Event Framework for Social Action. *Sociological Methodology*, 38(1), 155–200. <https://doi.org/10.1111/j.1467-9531.2008.00203.x>
- Butts, C. T., & Marcum, C. S. (2017). A relational event approach to modeling behavioral dynamics. In *Group processes* (pp. 51-92). Springer, Cham.
- Foucault Welles, B., Vashevko, A., Bennett, N., & Contractor, N. (2014). Dynamic Models of Communication in an Online Friendship Network. *Communication Methods and Measures*, 8(4), 223–243. <https://doi.org/10.1080/19312458.2014.967843>
- Hewes, D. E., & Poole, M. S. (2012). The analysis of group interaction processes. In *Research methods for studying groups and teams* (pp. 358-385). Routledge.
- I. (2021, October 7). *GitHub - issa-tseng/apollo13rt: Apollo 13 in real-time. Flight director's loop, transcript, information, and more*. GitHub. <https://github.com/issa-tseng/apollo13rt>
- Kass, R. E., & Raftery, A. E. (1995). Bayes Factors. *Journal of the American Statistical Association*, 90(430), 773–795. <https://doi.org/10.1080/01621459.1995.10476572>
- Meijerink-Bosman, M., Back, M., Geukes, K., Leenders, R., & Mulder, J. (2022). Discovering trends of social interaction behavior over time: An introduction to relational event modeling. *Behavior Research Methods*. <https://doi.org/10.3758/s13428-022-01821-8>

- Mulder, J., & Leenders, R. T. (2019). Modeling the evolution of interaction behavior in social networks: A dynamic relational event approach for real-time analysis. *Chaos, Solitons & Fractals*, *119*, 73–85. <https://doi.org/10.1016/j.chaos.2018.11.027>
- Peng, S., Zhou, Y., Cao, L., Yu, S., Niu, J., & Jia, W. (2018). Influence analysis in social networks: A survey. *Journal of Network and Computer Applications*, *106*, 17–32. <https://doi.org/10.1016/j.jnca.2018.01.005>
- Pilny, A., Schechter, A., Poole, M. S., & Contractor, N. (2016). An illustration of the relational event model to analyze group interaction processes. *Group Dynamics: Theory, Research, and Practice*, *20*(3), 181–195. <https://doi.org/10.1037/gdn0000042>
- Quintane, E., Conaldi, G., Tonellato, M., & Lomi, A. (2014). Modeling Relational Events. *Organizational Research Methods*, *17*(1), 23–50. <https://doi.org/10.1177/1094428113517007>
- Shafiee Kamalabad, M. S., & Grzegorzczak, M. (2019). Non-homogeneous dynamic Bayesian networks with edge-wise sequentially coupled parameters. *Bioinformatics*. <https://doi.org/10.1093/bioinformatics/btz690>
- Shafiee Kamalabad, M., Heberle, A. M., Thedieck, K., & Grzegorzczak, M. (2018). Partially non-homogeneous dynamic Bayesian networks based on Bayesian regression models with partitioned design matrices. *Bioinformatics*, *35*(12), 2108–2117. <https://doi.org/10.1093/bioinformatics/bty917>
- Shafiee Kamalabad, M., Leenders, R., & Mulder, J. (in review). *What's the Point of Change? Change Point Detection in Relational Event Models*.
- Schechter, A., & Quintane, E. (2020). The Power, Accuracy, and Precision of the Relational Event Model. *Organizational Research Methods*, *24*(4), 802–829. <https://doi.org/10.1177/1094428120963830>

TilburgNetworkGroup. (2022, April 13). *GitHub - TilburgNetworkGroup/remstats: Computing statistics for relational event history data*. GitHub.
<https://github.com/TilburgNetworkGroup/remstats>

Tranmer, M., Marcum, C. S., Morton, F. B., Croft, D. P., & de Kort, S. R. (2015). Using the relational event model (REM) to investigate the temporal dynamics of animal social networks. *Animal Behaviour*, *101*, 99–105.
<https://doi.org/10.1016/j.anbehav.2014.12.005>

van den Oever, F., & Schraagen, J. M. (2021). Team Communication Patterns in Critical Situations. *Journal of Cognitive Engineering and Decision Making*, *15*(1), 28–51.
<https://doi.org/10.1177/1555343420986657>

Wikipedia contributors. (2022, February 6). *Apollo 13*. Wikipedia.
https://en.wikipedia.org/w/index.php?title=Apollo_13&oldid=1070171476