

Supplementary

1 The pseudo codes

Algorithm 1: CreateFollowingNetwork

input : A time series set $\mathcal{Q} = \{Q_1, \dots, Q_n\}$
 and a threshold σ
output: A $n \times n$ adjacency matrix E

$E_{i,j} = 0, \forall i, j \in \{1, \dots, n\};$
for $i \leftarrow 1$ **to** n **do**
 for $j \leftarrow i + 1$ **to** n **do**
 $U \leftarrow Q_i$ and $W \leftarrow Q_j;$
 $P_{U,W} \leftarrow DTW(U, W);$
 if $s(P_{U,W}) \geq \sigma$ **then**
 $E_{j,i} = |s(P_{U,W})|;$
 else if $s(P_{U,W}) \leq -\sigma$ **then**
 $E_{i,j} = |s(P_{U,W})|;$
 else
 $E_{i,j} = 0;$
 end
 end
end

2 Coordination measure

Given a set of time series \mathcal{U} , a set of clusters $\mathcal{C} = \{H_1, \dots, H_n\}$ such that $\bigcup_k H_k = \mathcal{U}$, we define a cluster membership indicator $\delta_{i,j} = 1$ if time series U_i and U_j belong to the similar cluster, otherwise it is zero. **The average coordination measure** Ψ of a set of clusters \mathcal{C} is defined as follows:

$$(2.1) \quad \Psi(\mathcal{C}) = \frac{\sum_{U_i, U_j \in \mathcal{U}, U_i \neq U_j} \text{sim}_{\max}(U_i, U_j) \delta_{i,j}}{\sum_{U_i, U_j \in \mathcal{U}, U_i \neq U_j} \delta_{i,j}}.$$

Note that $\Psi \in [0, 1]$. If Ψ is close to one, then all the time series within the same cluster are highly similar, with some time delay. This implies there exists a high degree of coordination within each clusters in this case. On the contrary, $\Psi \approx 0$ implies no coordination, on average.

THEOREM 2.1. *Given a set of time series \mathcal{U} containing*

Algorithm 2: CreateDyFollowingNetwork

input : A time series set \mathcal{U} , ω , δ , and σ
output: A $n \times n \times t^*$ adjacency matrix E^* .

$K \leftarrow (t^* - \omega) / \delta;$
for $i \leftarrow 1$ **to** K **do**
 / current time interval */*
 $w(i) = [(i - 1) \times \delta, (i - 1) \times \delta + \omega];$
 / SubTimeSeries($U, w(i)$) returns all sub time series in U within the interval $w(i)$ */*
 $Q \leftarrow \text{SubTimeSeries}(U, w(i));$
 $E \leftarrow \text{CreateFollowingNetwork}(Q, \sigma);$
 / Set all edges within the time interval $[(i - 1) \times \delta, i \times \delta]$ to be similar */*
 $E_{t \in [(i-1) \times \delta, i \times \delta]}^* \leftarrow E;$
end
 $Q \leftarrow \text{SubTimeSeries}(U, [K \times \delta, t^*]);$
 $E \leftarrow \text{CreateFollowingNetwork}(Q, \sigma);$
 $E_{t \in [K \times \delta, t^*]}^* \leftarrow E;$

a set of σ -faction $\mathcal{F} = \{F_1, \dots, F_n\}$ where $\bigcup_{F_i \in \mathcal{F}} F_i = \mathcal{U}$, then, for all possible sets of clusters, \mathcal{F} maximizes the average coordination measure Ψ .

Proof. Reminding that for all pairs U_i, U_j within any similar faction F , $\text{sim}_{\max}(U_i, U_j) \geq \sigma$. Hence, $\Psi(\mathcal{F}) \geq \sigma$.

Case 1: let $H, J \in \mathcal{F}$, if we modify \mathcal{F} by exchanging any time series $U_H \in H$ with $U_J \in J$ and call it \mathcal{C} , then we have:

$$\Psi(\mathcal{F}) - \Psi(\mathcal{C}) = \frac{S + S'}{\sum_{U_i, U_j \in \mathcal{U}, U_i \neq U_j} \delta_{i,j}}.$$

$$S = \sum_{U_i \in H \setminus \{U_H\}} \left(\text{sim}_{\max}(U_i, U_H) - \text{sim}_{\max}(U_i, U_J) \right)$$

$$S' = \sum_{U_i \in J \setminus \{U_J\}} \left(\text{sim}_{\max}(U_i, U_J) - \text{sim}_{\max}(U_i, U_H) \right)$$

For any $U_i \in H$, $\text{sim}_{\max}(U_i, U_H) \geq \sigma$ since $U_H \in H$.

Algorithm 3: FindFactionsAndInitiators

input : An adjacency matrix E^* of dynamic network
output : A time series of faction sets \mathcal{F}^* , and a time series of initiator sets \mathcal{L}^*

```
for  $i \leftarrow 1$  to  $t^*$  do
    /* Get a matrix at time  $t=i$  */
     $E \leftarrow E_{t=i}^*$ ;
    /* FindInitiators( $E$ ) returns all nodes which have zero outgoing degree */
     $\mathcal{L} \leftarrow \text{FindInitiators}(E)$ ;
     $\mathcal{F} = \emptyset$ ;
    for  $l \in \mathcal{L}$  do
        /* FindReachNodeFrom( $E, l$ ) returns all nodes which have any directed path to  $l$  */
         $F_l \leftarrow \text{FindReachNodeFrom}(E, l)$ ;
         $\mathcal{F} = \mathcal{F} \cup \{F_l\}$ 
    end
     $\mathcal{F}_{t=i}^* = \mathcal{F}$  and  $\mathcal{L}_{t=i}^* = \mathcal{L}$ 
end
```

In contrast, because $U_J \notin H$, then $\text{sim}_{\max}(U_i, U_J) < \sigma$, which implies $S > 0$. $S' > 0$ for a similar reason. Therefore, $\Psi(\mathcal{F}) - \Psi(\mathcal{C}) > 0$.

Case 2: if we create \mathcal{C} from \mathcal{F} by spiting a cluster $H \in \mathcal{F}$ to be $H_1 \subset H$ and $H_2 = H \setminus H_1$, then we have:

$$\Psi(\mathcal{F}) - \Psi(\mathcal{C}) = \frac{\sum_{U_i \in H_1, U_j \in H_2} \text{sim}_{\max}(U_i, U_j)}{|H_1||H_2|} \geq \sigma.$$

Case 3: we create \mathcal{C} from \mathcal{F} by merging any cluster $H \in \mathcal{F}$ with any $J \in \mathcal{F}$ such that $H \neq J$ to be H' . So, let

$$\Psi(\mathcal{F}) = \frac{X_{\mathcal{F}}}{S_{\mathcal{F}}} \geq \sigma,$$

then

$$\Psi(\mathcal{C}) = \frac{X_{\mathcal{F}} + \sum_{U_i \in H, U_j \in J} \text{sim}_{\max}(U_i, U_j)}{S_{\mathcal{F}} + |H||J|}.$$

By merging H and J , we introduce pairs of time series across H and J to Equation 2.1 such that $\text{sim}_{\max}(U_i, U_j) < \sigma$ since these pairs are not belong to the same faction. These pairs decrease the average of $X_{\mathcal{F}}$, which implies $\Psi(\mathcal{F}) > \Psi(\mathcal{C})$.

Since we shown that no matter how we edit \mathcal{F} , the average coordination measure Ψ cannot increase, therefore, \mathcal{F} maximizes the average coordination measure.

3 Time complexity

Let n be a number of time series, ω be a time window, δ be a shifting factor (we use $\delta = 0.1\omega$), and t^* be a total length of time series. By deploying DTW Sakoe Chiba band technique [2] setting δ as a band limitation, the time complexity of computing a following network is $\mathcal{O}(n^2 \times \omega \times \delta)$. Since we need warping paths, not a distance, the upper/lower bounds tricks which are used to speed up DTW found in the time series literature cannot be applied here. The number of following networks we need to compute is $\frac{t^*}{\delta}$. In total, the time complexity of our framework is $\mathcal{O}(n^2 \times \omega \times t^*)$. Additionally, we might explore k candidates of ω in order to find the optimal ω . Since k is a constant, the asymptotic time complexity of our framework also remains the same. This expensive cost is unavoidable and it makes our framework hard to be a scalable framework.

4 Comparison method

From the best of our knowledge, there is no existing methods dealing with the FACTION INITIATOR INFERENCE PROBLEM. The closest method that we can compare against is the flock model [1, 3]. We compared our framework against Volatility Collective Behaviors Model [3], which has an assumption that all members in a similar group move toward the similar direction on a non-linear trajectory. Hence, we modified the FLOCK framework to make it work in our setting as a baseline of comparison. In stead of using DTW to build following networks, we created FLOCK following networks. According to the work in [1], the time series A follows the time series B at any time step t if the angle of their direction vector from time $t-1$ to t is less than the threshold β as well as B is in the front of A with respect to B 's direction, as well as A and B must have their distance less than the threshold γ . The FLOCK following networks are built for all time steps. The rest of FLOCK framework is similar to our framework. We set the FLOCK parameters such that it can perform the best.

5 Centrality measures in multi-faction datasets

In this section, we explore the use of centrality measures to infer faction initiators. We used 200 datasets from dictatorship models to conduct the analysis. For each dataset, we created a global static following network and used centrality measures on this network. In each dataset, we have 30 individuals and four of them are initiators.

The Jaccard similarity result between top-4 ranking individuals from centrality measures and the ground truth set of four initiators is in Table 1. PageRank and IN-Degree centrality perform well in dataset containing

Table 3: Support of four initiators being in the list of top-4 ranking individuals from centrality measures in 100 datasets containing merging/splitting coordination events.

Initiator's ID	Centrality methods		
	PageRank	IN-Degree	Closeness
ID1	1	1	1
ID2	0.29	0.47	0.20
ID3	1	1	0.83
ID4	0.28	0.19	0.08

Table 1: Jaccard similarity between top-4 ranking individuals from centrality measures and the ground truth set of four initiators in dictatorship model from 100 datasets.

Event types	Centrality methods		
	PageRank	IN-Degree	Closeness
Linear	0.85	0.84	0.54
Merge/split	0.64	0.67	0.53

Table 2: Support of four initiators being in the list of top-4 ranking individuals from centrality measures in 100 datasets containing linear coordination events.

Initiator's ID	Centrality methods		
	PageRank	IN-Degree	Closeness
ID1	1	1	0.92
ID2	1	1	0.66
ID3	1	1	0.36
ID4	0.39	0.37	0.20

the simple linear coordination events while closeness centrality performs the worst. This is because initiators suppose to have a higher number of followers than normal individuals, which implies the higher ranking w.r.t.

PageRank and In-Degree centrality. In contrast, initiators is not necessary being close to their followers in the network, which made closeness centrality perform poorly. For the datasets that contain merge/split-coordination events, because there is complicated dynamics of interaction among factions, the simple centrality measures fail to capture the true initiators.

Table 2 illustrates the result of supports of four initiators being in the list of top-4 ranking individuals from centrality measures in linear-coordination datasets. Similarly, PageRank and In-Degree centrality perform well, while closeness centrality perform poorly. For the merge/split-coordination datasets, Table 3 shows that all centrality measures perform poorly to infer ID2 and ID4 initiators while they perform well to include ID1 and ID3 in their top-4 ranking lists. This is because ID1 and ID3 spent significantly more time to lead their factions than ID2 and ID4.

In conclusion, these results emphasize the need of dynamic following network to deal with complicated faction initiator inference problem.

References

- [1] M. Andersson, J. Gudmundsson, P. Laube, and T. Wolle. Reporting leaders and followers among trajectories of moving point objects. *GeoInformatica*, 12(4):497–528, 2008.
- [2] H. Sakoe and S. Chiba. Dynamic programming algorithm optimization for spoken word recognition. *IEEE transactions on acoustics, speech, and signal processing*, 26(1):43–49, 1978.
- [3] T. E. Will. Flock leadership: Understanding and influencing emergent collective behavior. *The Leadership Quarterly*, 27(2):261–279, 2016.