30 August – 3 September 2021
INTERSPEECH 2021
BRNO | CZECHIA
Speech everywhere!

Reformulating DOVER-Lap Label Mapping as a Graph Partitioning Problem

Desh Raj and Sanjeev Khudanpur

Center for Language and Speech Processing
Johns Hopkins University, Baltimore, USA

September 1, 2021

CENTER FOR LANGUAGE AND SPEECH PROCESSING

Overview

- **Background & Motivation**:
  - What is speaker diarization?
  - Approaches for diarization: the need for ensembles
  - DOVER-Lap
- **The Label Mapping problem**
  - Graph formulation of the problem
  - DOVER-Lap's exponential time algorithm
  - New linear-time solution: the Hungarian algorithm
  - Randomized local search – approaching optimality
- **Experimental results**

Background & Motivation

Background

What is speaker diarization?

Task of “who spoke when”

Input: ***recording containing multiple speakers***

Output: ***homogeneous speaker segments***

Xavier Anguera Miro et al., “Speaker diarization: A review of recent research,” IEEE Transactions on Audio, Speech, and Language Processing, 2012.

Background

What is speaker diarization?

Task of “who spoke when”

Input: ***recording containing multiple speakers***

Number of speakers may be unknown

Overlapping speech may be present

Output: ***homogeneous speaker segments***

Motivation

Existing methods for diarization

Conventional methods

Spectral clustering (SC)
Agglomerative hierarchical clustering (AHC)
Variational Bayes (VBx)

- Clustering of small segment embeddings, such as i-vectors or x-vectors
- Optionally include overlap assignment

Daniel Garcia-Romero, David Snyder, Gregory Sell, Daniel Povey, and Alan McCree, “Speaker diarization using deep neural network embeddings,” ICASSP 2017.

Mireia Dîez, Lukas Burget, and Pavel Matejka, “Speaker diarization based on Bayesian HMM with eigenvoice priors,” Odyssey 2018.

Latane Bullock, Hervé Bredin, and L. Paola García-Perera, “Overlap-aware diarization: resegmentation using neural end-toend overlapped speech detection,” ICASSP 2020.

Motivation

Existing methods for diarization

New methods

Region proposal networks (RPN)

End-to-end neural diarization (EEND)

Target speaker voice activity detection (TS-VAD)

- Supervised training based systems, trained to directly predict segments.
- Includes overlap assignment by design

Zili Huang, Shinji Watanabe, Yusuke Fujita, Paola García, Yiwen Shao, Daniel Povey, and Sanjeev Khudanpur, “Speaker diarization with region proposal network,” ICASSP 2020.

Yusuke Fujita, Shinji Watanabe, Shota Horiguchi, Yawen Xue, and Kenji Nagamatsu, “End-to-end neural diarization: Reformulating speaker diarization as simple multi-label classification,” ArXiv.

Ivan Medennikov, et al., “Target speaker voice activity detection: a novel approach for multispeaker diarization in a dinner party scenario,” Interspeech 2020.

Machine learning tasks benefit from an ensemble of systems.

For example, ROVER is a popular combination method for ASR systems.

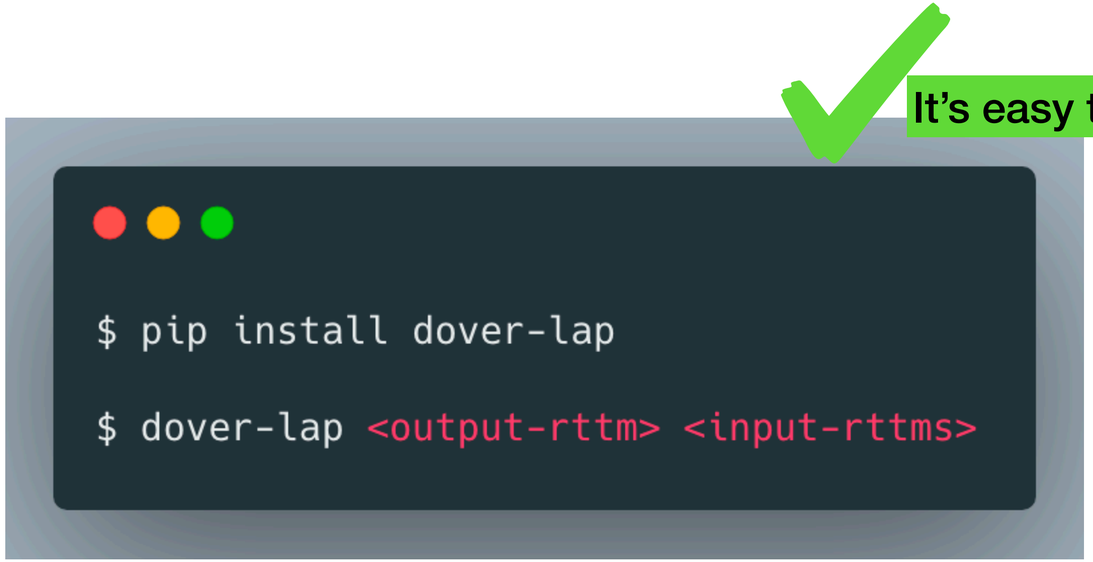
Jonathan G. Fiscus, “A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (ROVER),” IEEE ASRU 1997.

Our solution: DOVER-Lap

Top 2 teams used DOVER-Lap for system fusion in DIHARD-3

#1: USTC team combined clustering, separation-based, and TS-VAD systems

#2: Hitachi-JHU team combined VB-based and EEND-based systems



Wang, Y., et al. USTC-NELSLIP System Description for DIHARD-III Challenge. ArXiv, abs/2103.10661.

Horiguchi, S., et al. The Hitachi-JHU DIHARD III System: Competitive End-to-End Neural Diarization and X-Vector Clustering Systems Combined by DOVER-Lap. ArXiv, abs/2102.01363.

Our solution: DOVER-Lap

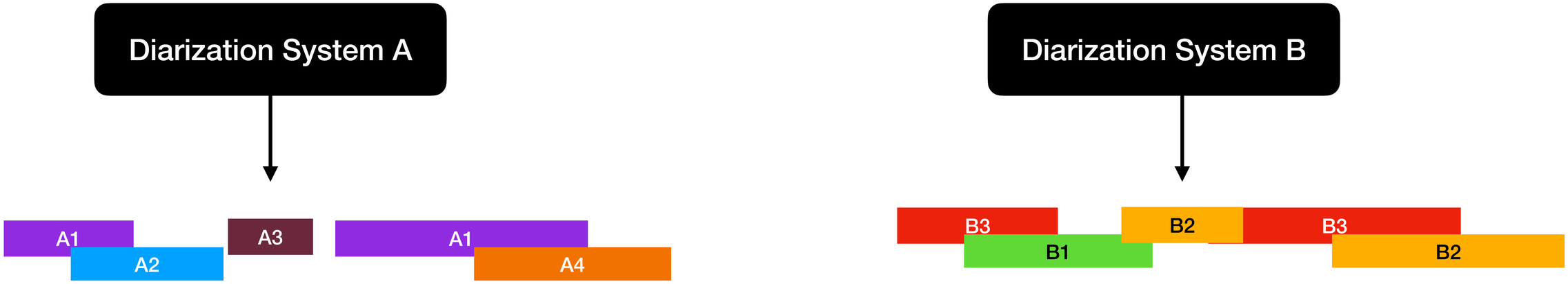
DOVER-Lap works in 2 stages

Label Mapping: map speaker labels to same label space

Label Voting: weighted majority voting among all hypotheses

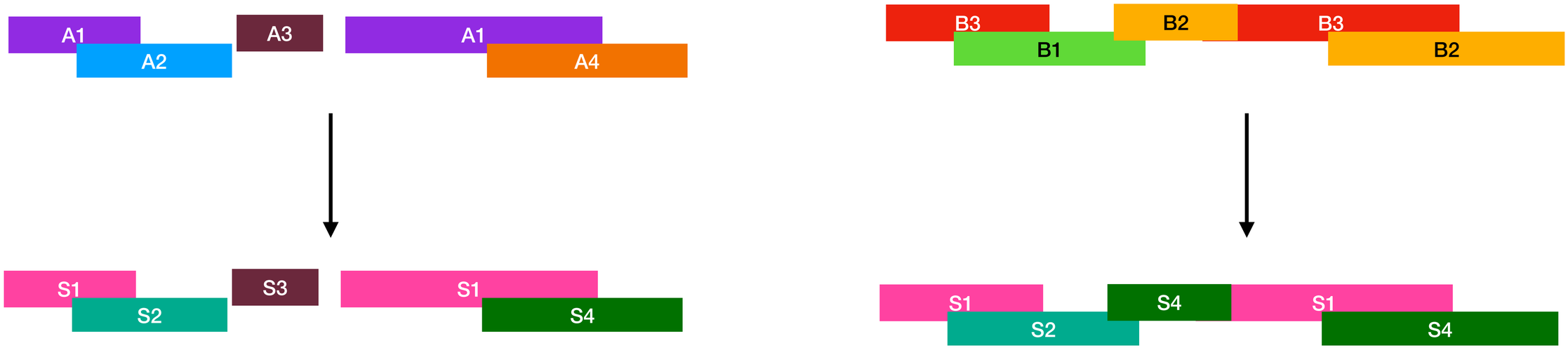
Raj et al., DOVER-Lap: A method for combining overlap-aware diarization outputs. IEEE SLT 2021.

DOVER-Lap Label Mapping



Diarization system outputs are **not** absolute speaker identities.

DOVER-Lap Label Mapping



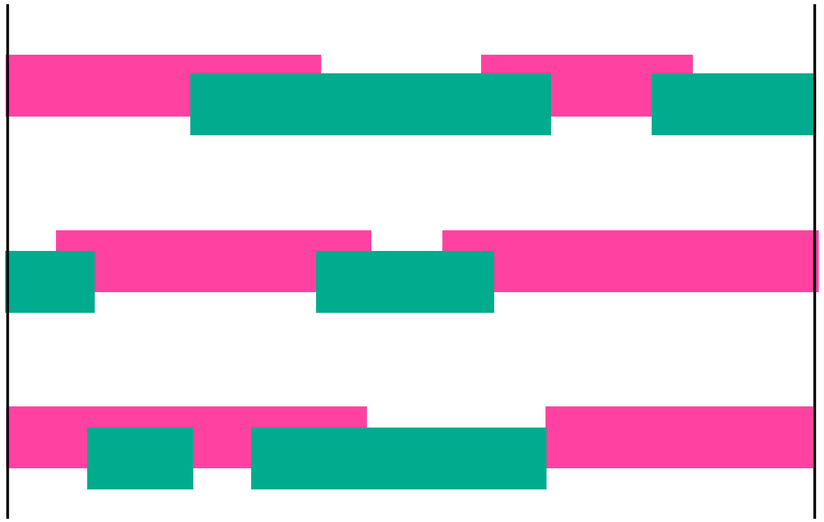
Need to map all hypotheses to **common label space**.

DOVER-Lap Label voting

Hypothesis A

Hypothesis B

Hypothesis C

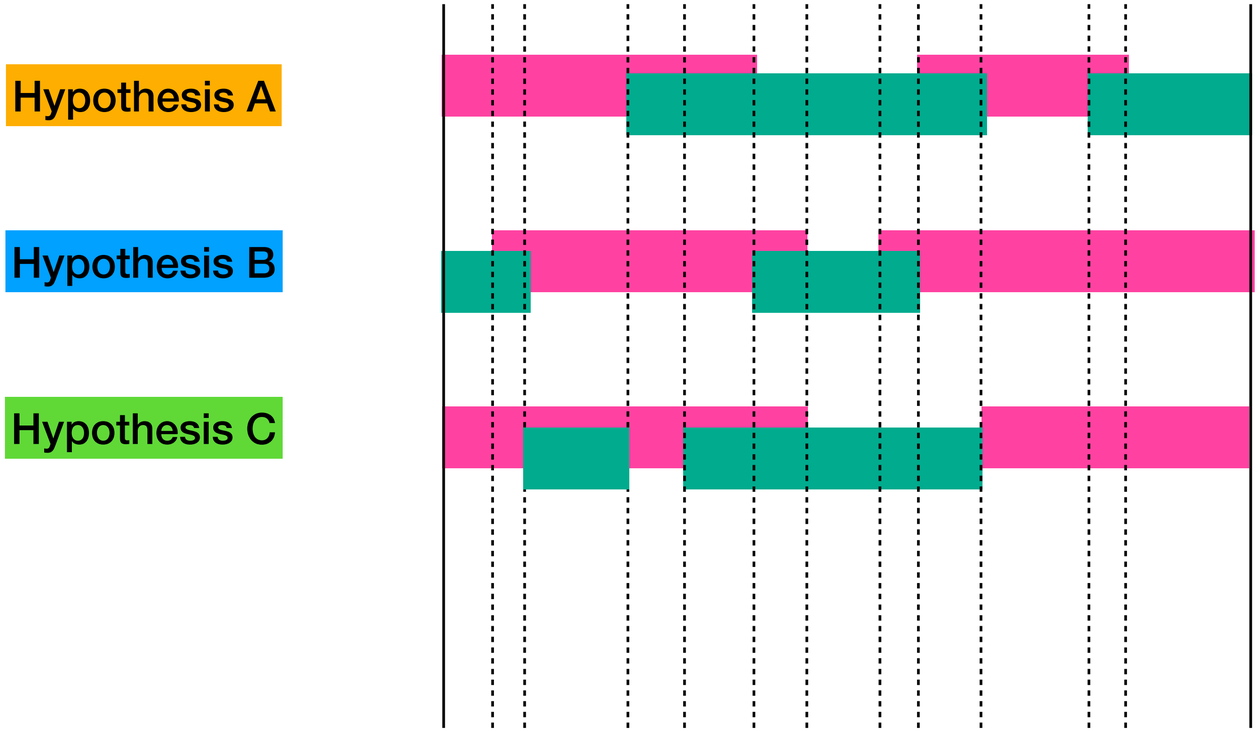


Speaker 1

Speaker 2

DOVER-Lap Label voting

Divide into regions (no speaker change within region)

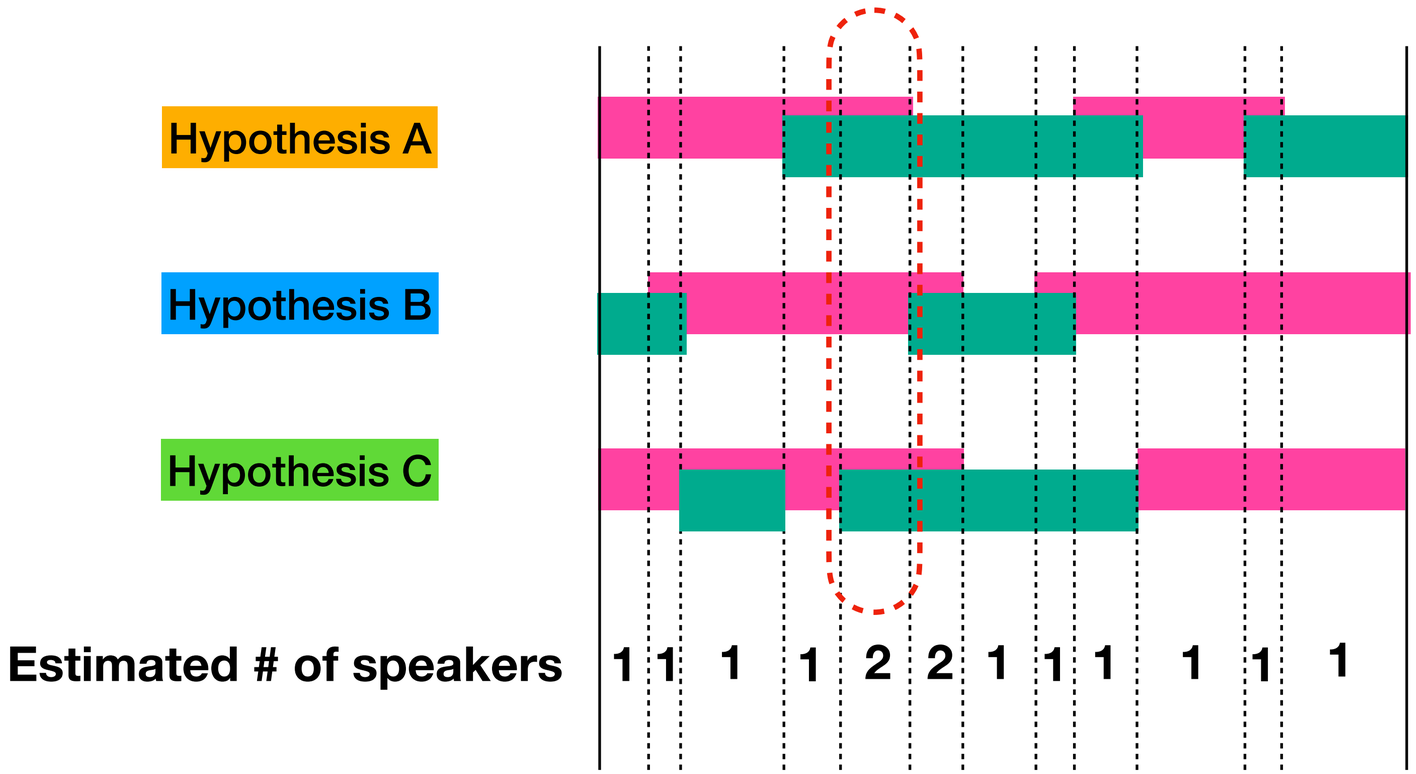


Speaker 1

Speaker 2

DOVER-Lap Label voting

Estimate number of speakers in each region

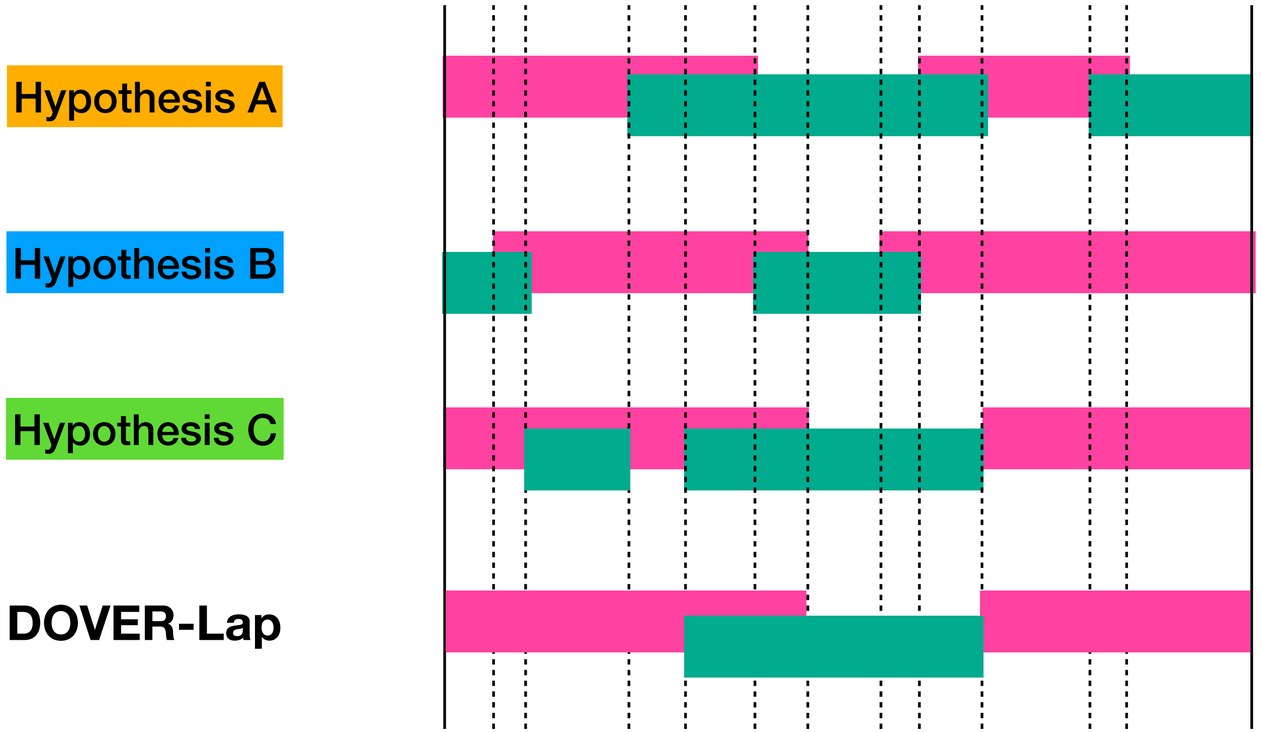


 # speakers = weighted mean of # speakers in hypotheses

Weights -> obtained by ranking hypotheses by **total cost**

DOVER-Lap Label voting

Assign highest weighted N speakers in each region



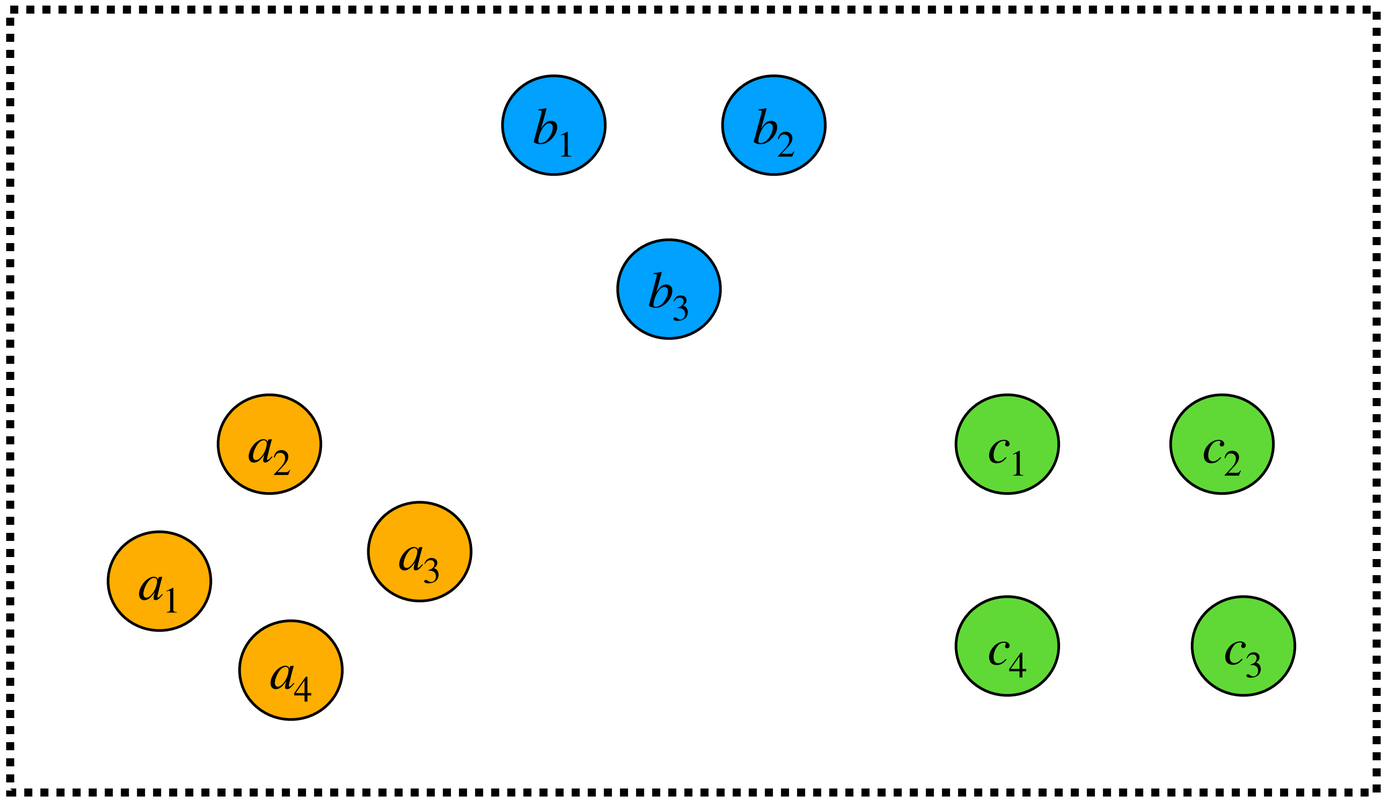
Speaker 1

Speaker 2

The Label Mapping Problem

Label Mapping

A graphical view



Hypothesis A

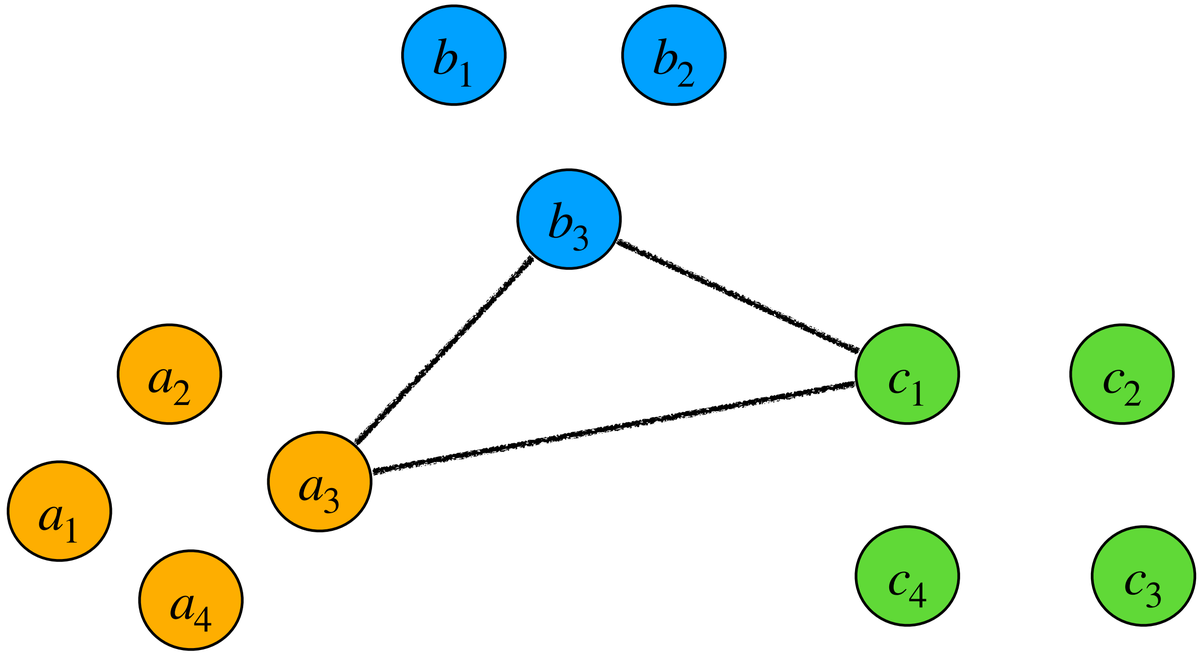
Hypothesis B

Hypothesis C

Each speaker in every hypothesis is a node.

Label Mapping

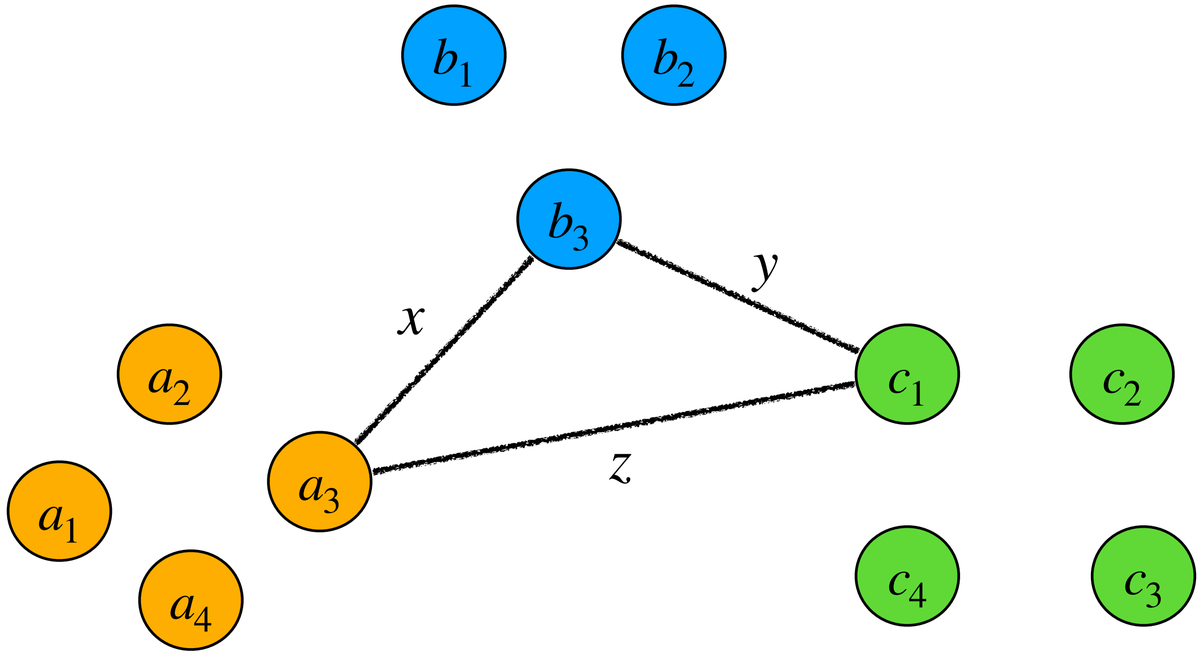
A graphical view



Edge between **every 2 nodes** not from the same hypotheses.

Label Mapping

A graphical view

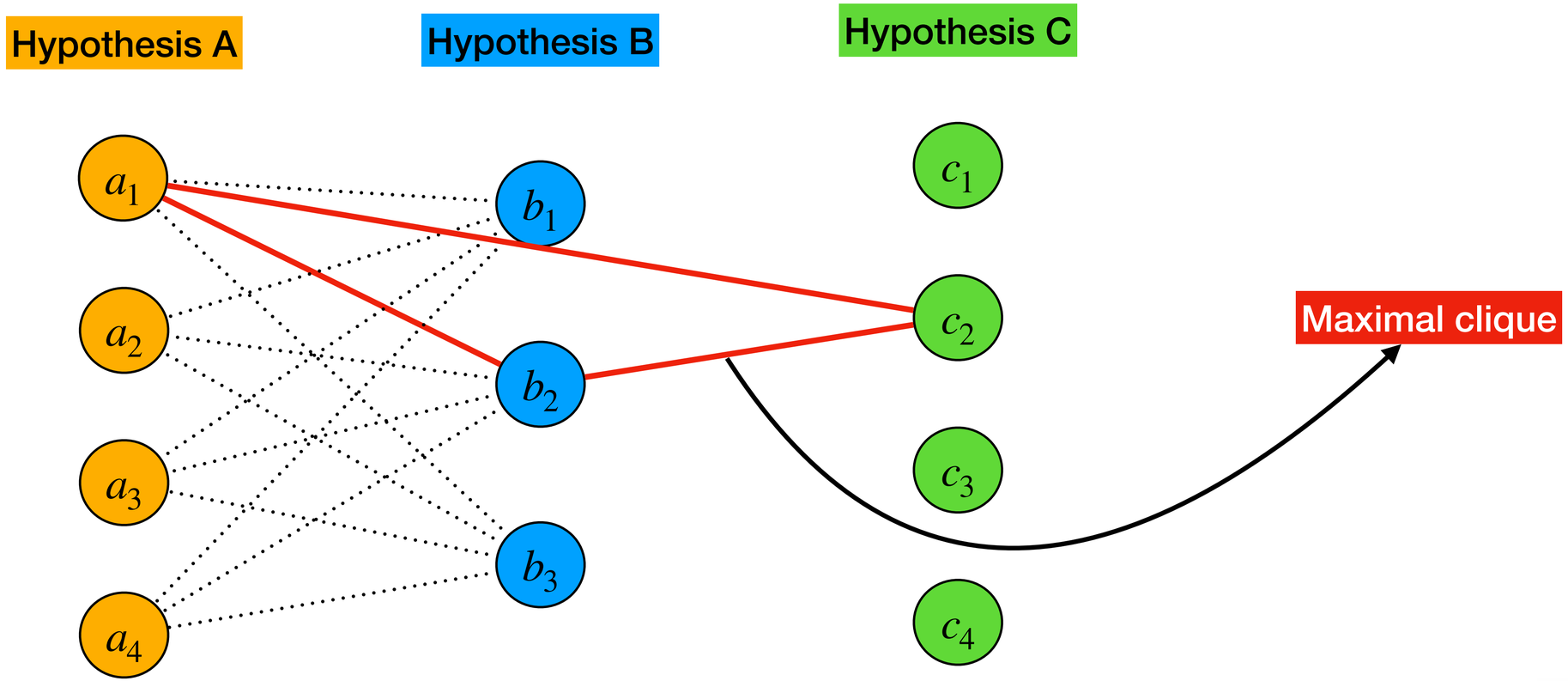


$$x = \frac{\Delta(a_3) \cap \Delta(b_3)}{\Delta(a_3) \cup \Delta(b_3)}$$

Edge weight denotes **similarity** of speaker labels – computed as **relative overlap** between active regions.

Label Mapping

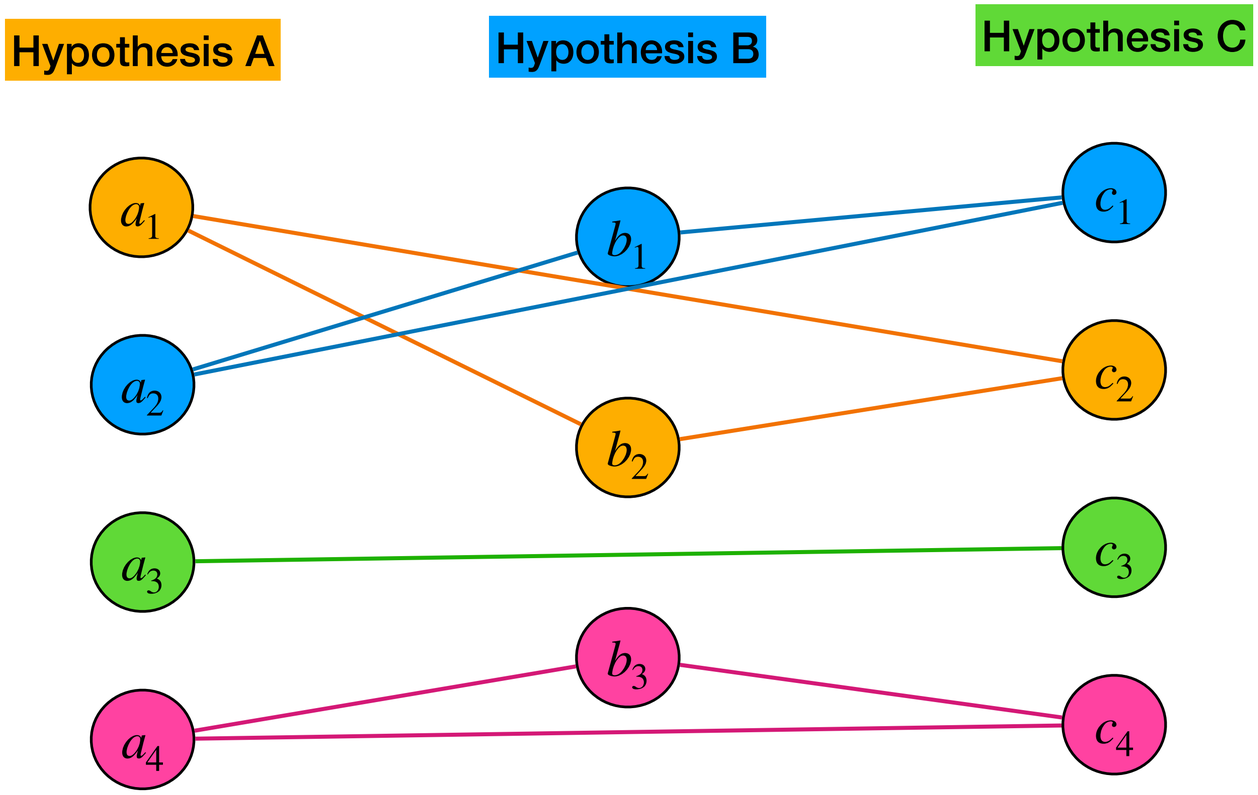
Instance of a graph partitioning problem



Red clique denotes **(a1, b2, c2)** mapped to the **same label.**

Label Mapping

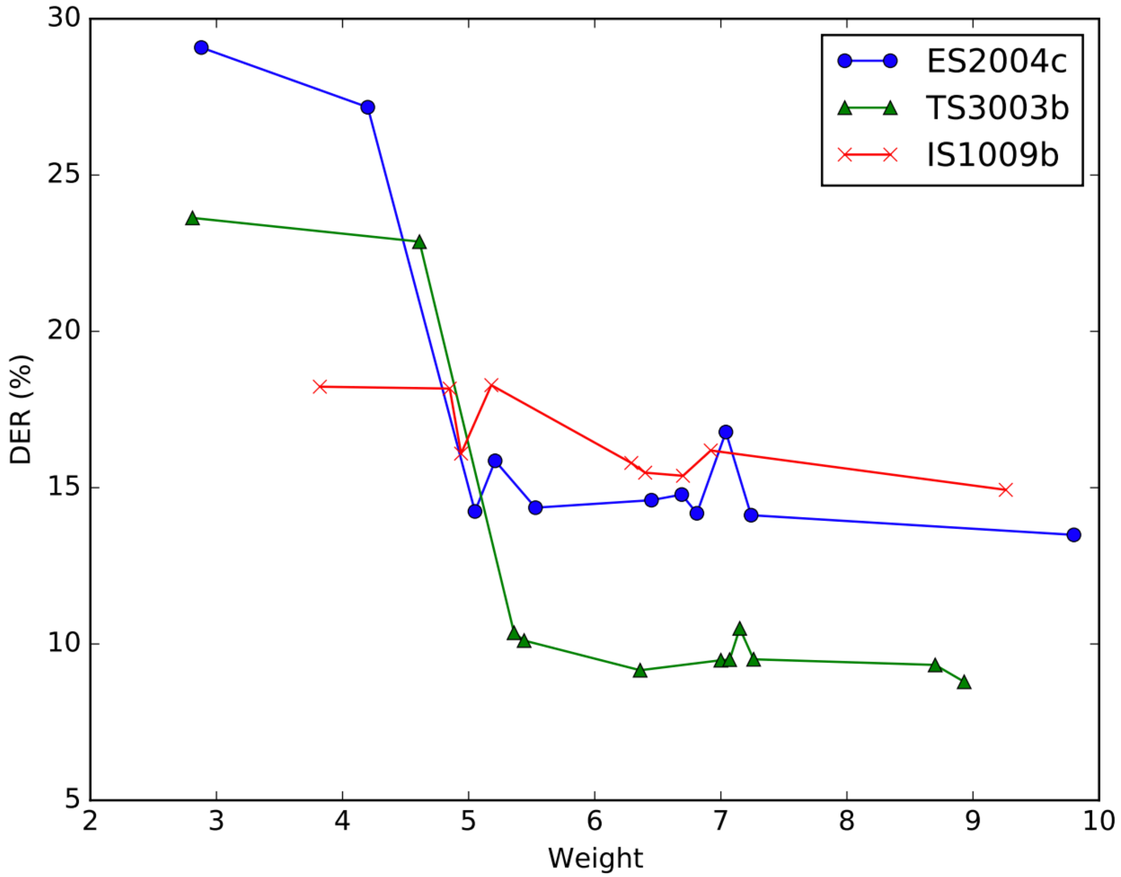
Instance of a graph partitioning problem



Partition the graph into a set of maximal cliques which **maximizes the sum of edge weights** within the cliques.

Label Mapping

Better solution loosely correlated with better DER



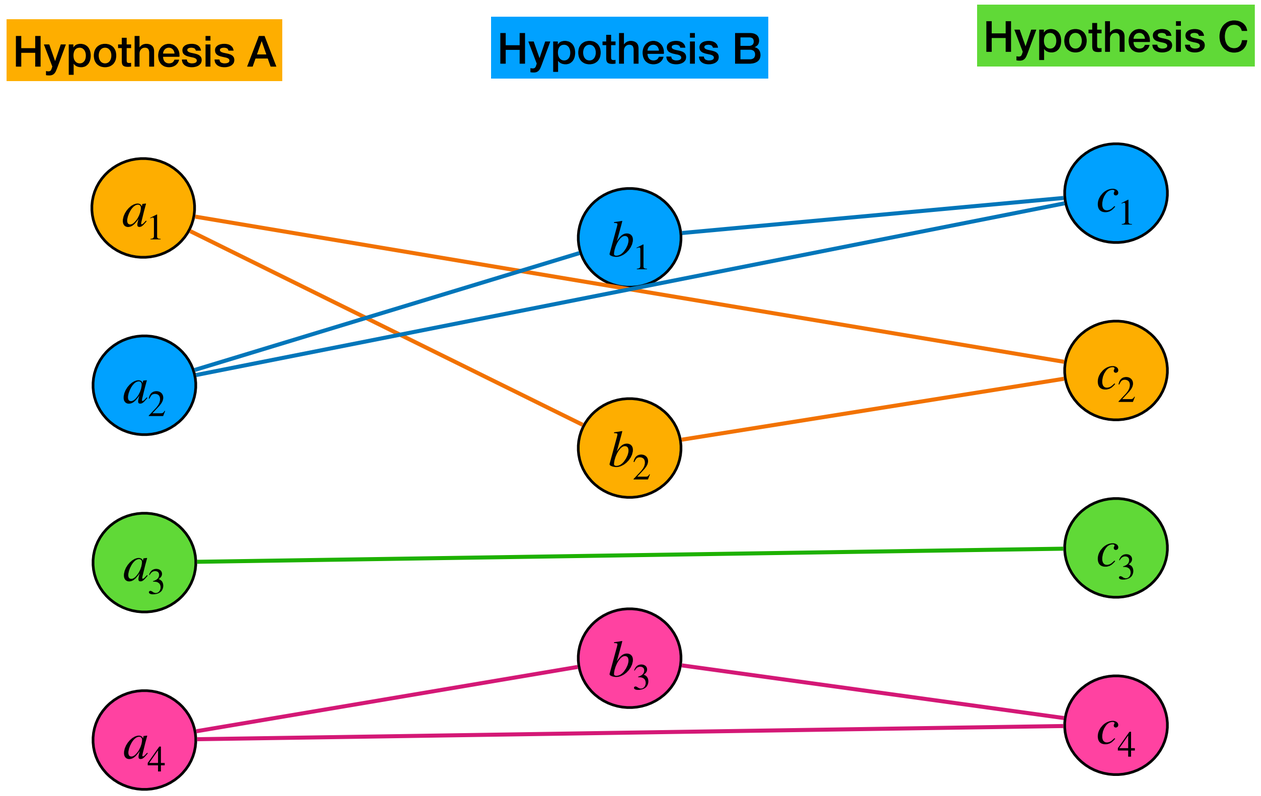
Recordings chosen from AMI evaluation set.

Higher weight → Lower DER

How does DOVER-Lap solve this problem?

DOVER-Lap algorithm

Greedy selection of maximal cliques

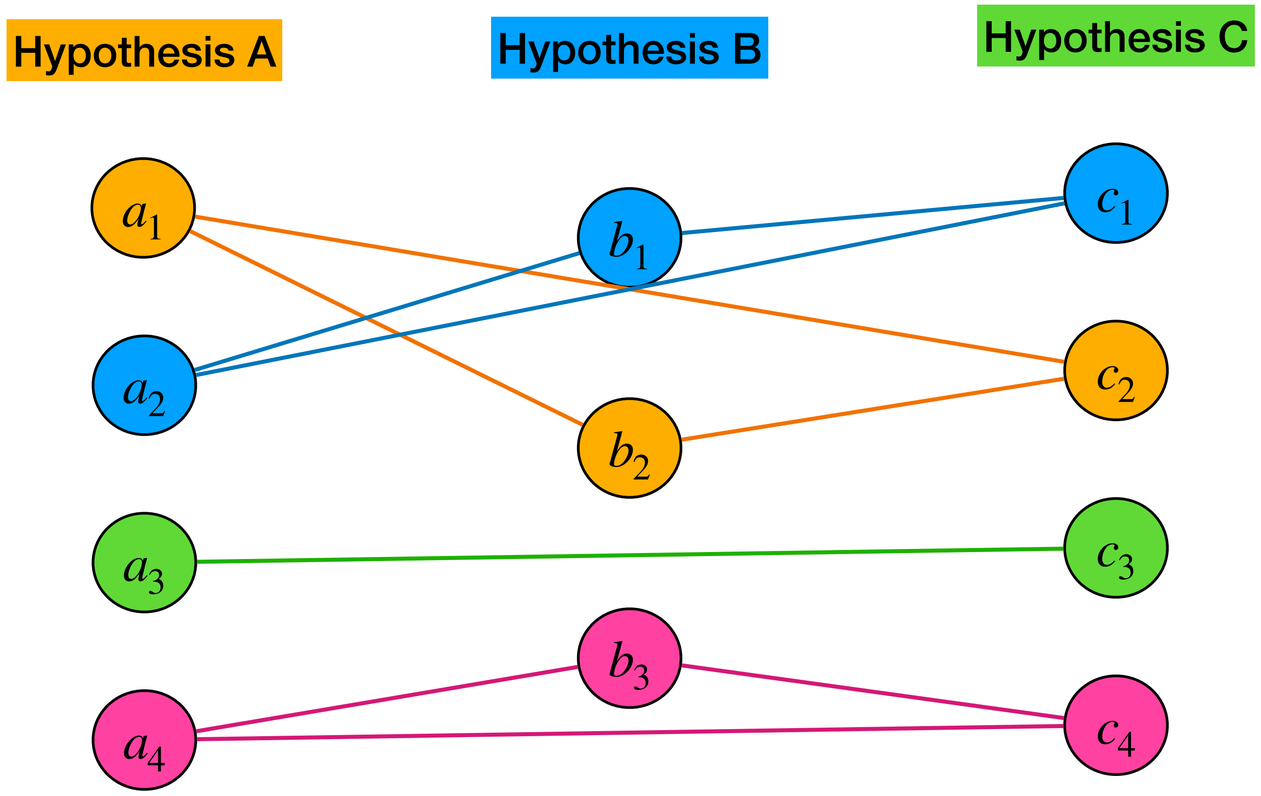


Step 1: Enumerate all maximal cliques and compute their edge weight sum.

Step 2: Greedily select in decreasing order until all nodes are covered.

DOVER-Lap algorithm

Exponential time!

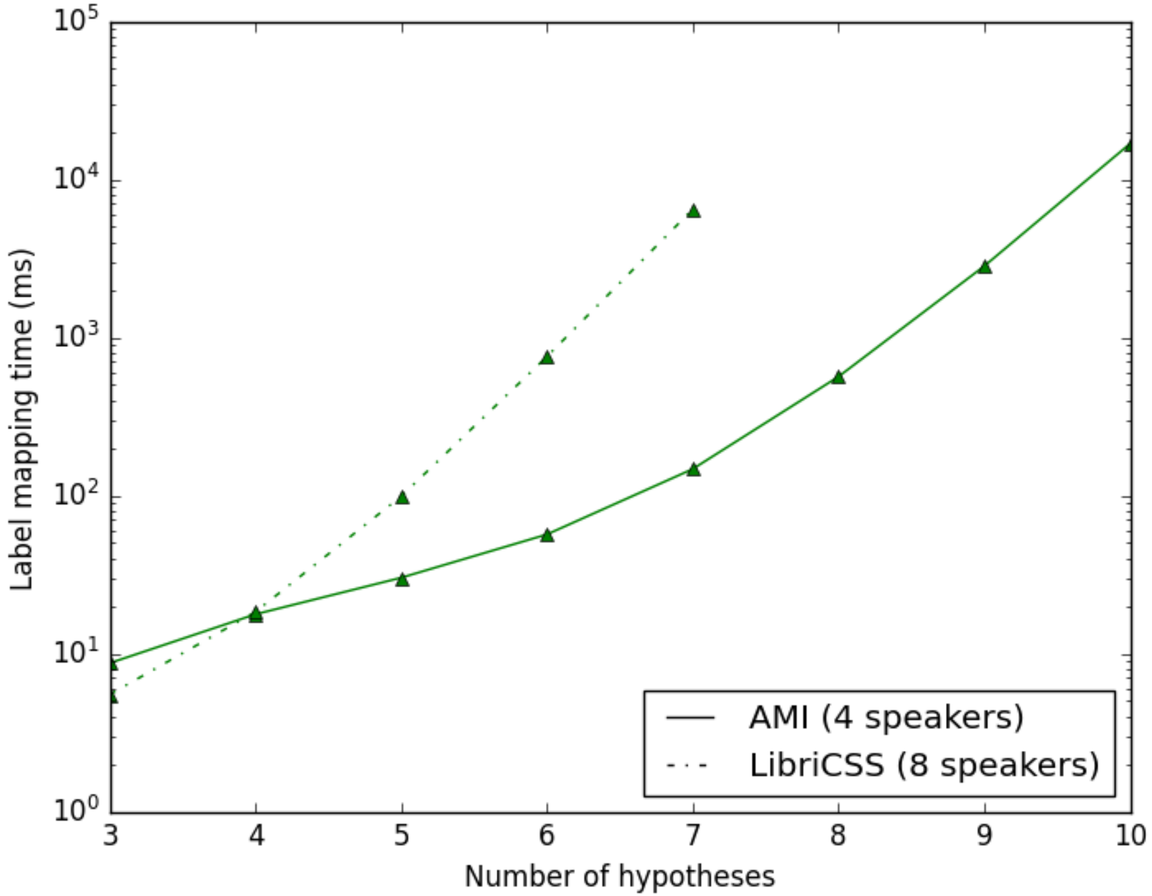


Step 1: Enumerate all maximal cliques and compute their edge weight sum.

→ There are **exponentially many** maximal cliques in the graph!

DOVER-Lap algorithm

Exponential time!



- Infeasible to perform system combination beyond a **small number** of hypotheses.
- Limit is reached faster for recordings with more speakers.

DOVER-Lap algorithm

But it works well for combining small number of systems

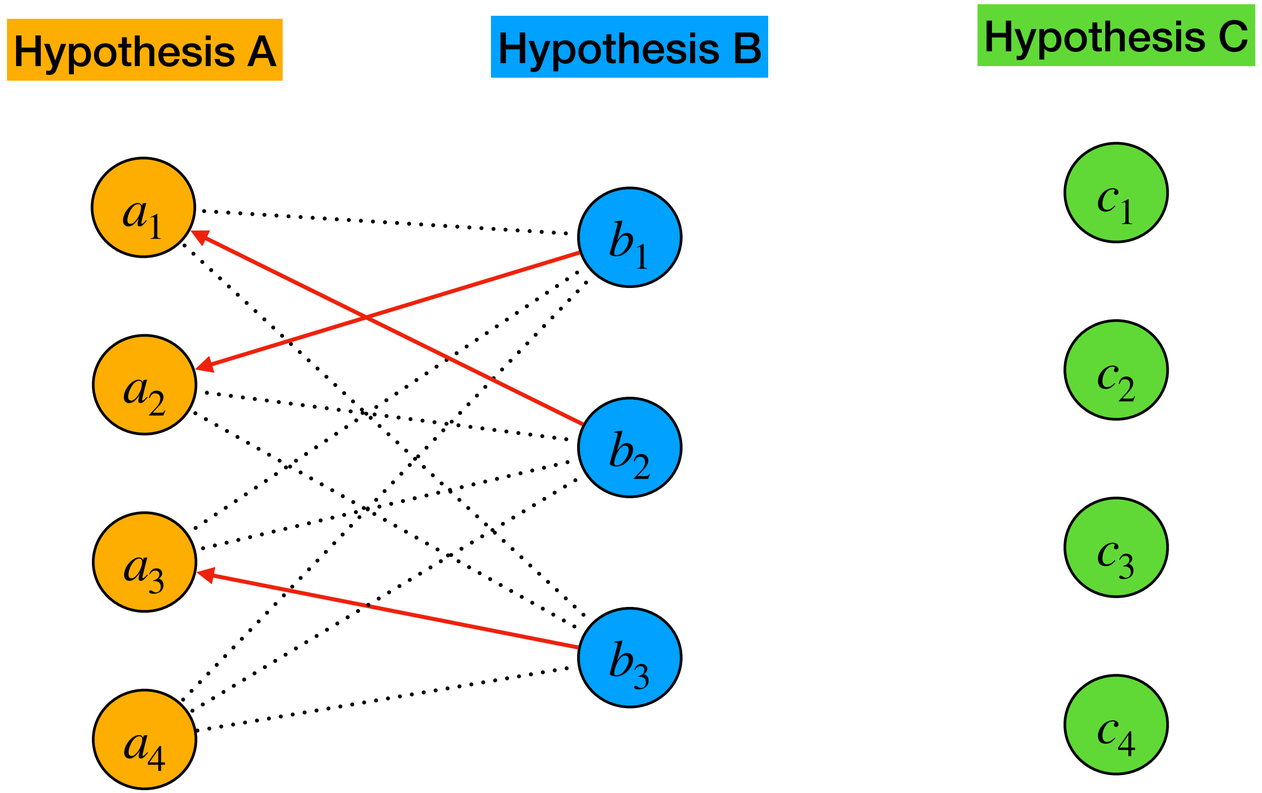
System	Spk. conf.	DER
Overlap-aware SC	10.1	23.6
VB-based overlap assignment	9.6	21.5
Region proposal network	8.3	25.5
Average	9.3	23.5
DOVER-Lap	**7.7**	**20.4**

Results on
AMI evaluation set

How can we avoid exponential complexity without sacrificing performance?

Preliminary: how DOVER works

Pair-wise incremental label mapping

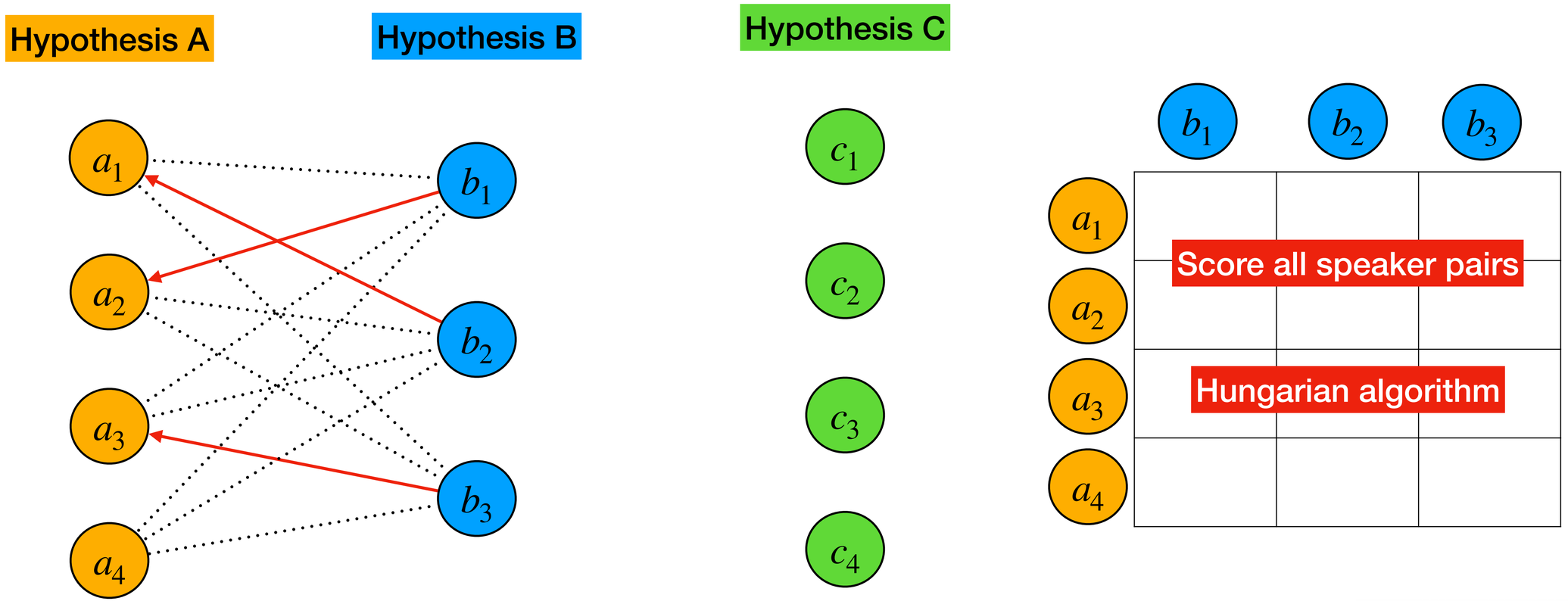


- Map hypothesis B to A using **Hungarian method**
- This same algorithm is used to map hypothesis to reference for **DER computation**

Andreas Stolcke and Takuya Yoshioka, “DOVER: A method for combining diarization outputs,” IEEE ASRU 2019.

Preliminary: how DOVER works

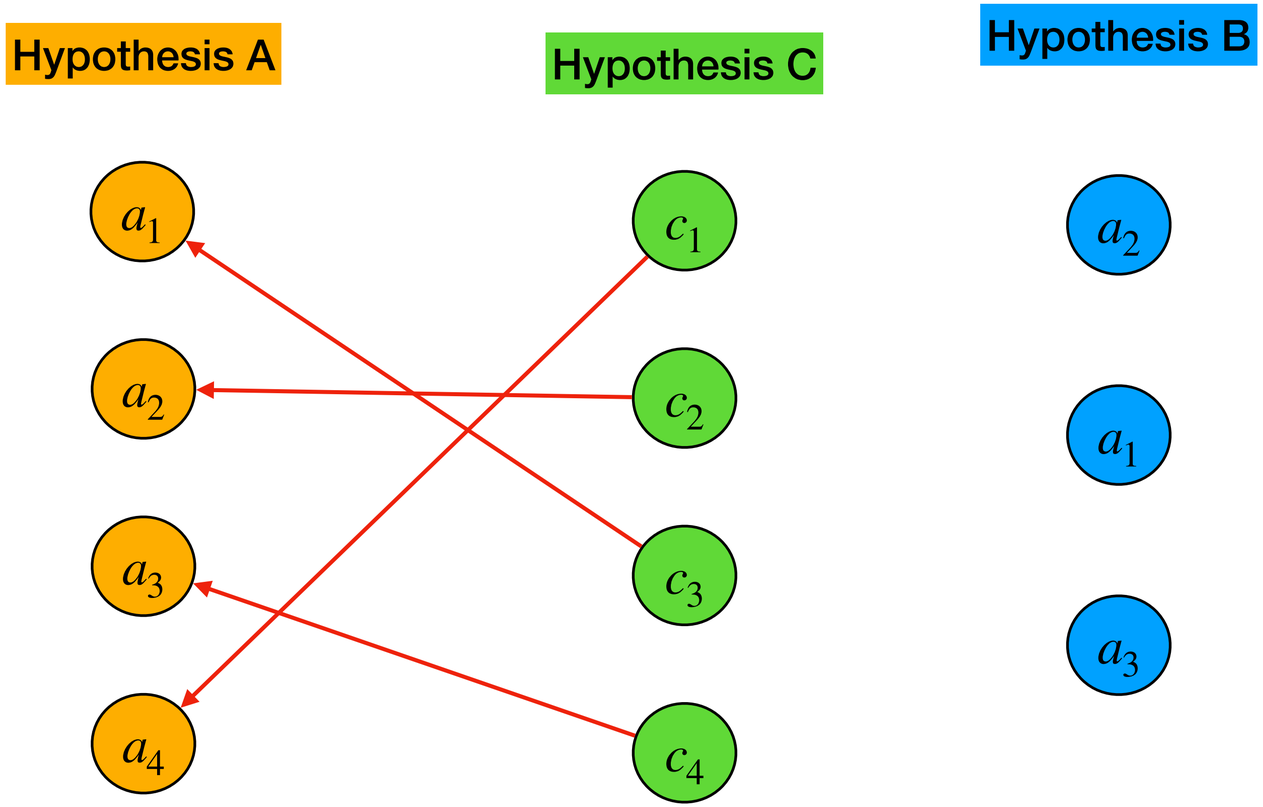
Pair-wise incremental label mapping



Andreas Stolcke and Takuya Yoshioka, “DOVER: A method for combining diarization outputs,” IEEE ASRU 2019.

Preliminary: how DOVER works

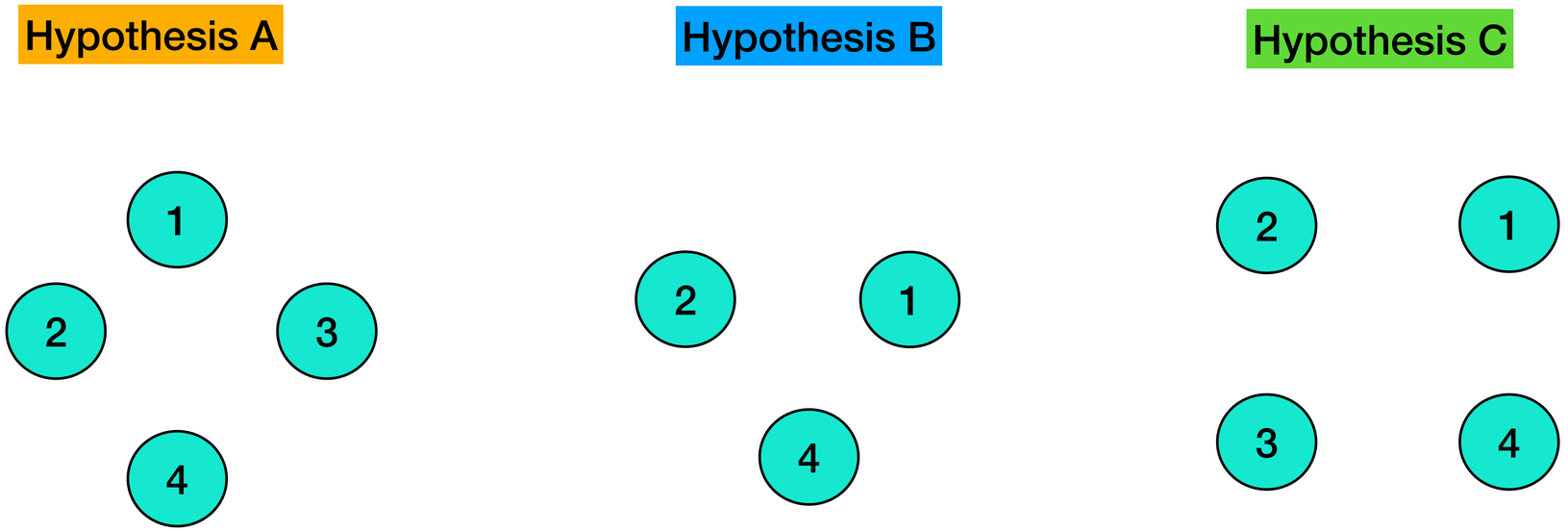
Pair-wise incremental label mapping



- Map hypothesis C to A using **Hungarian method**

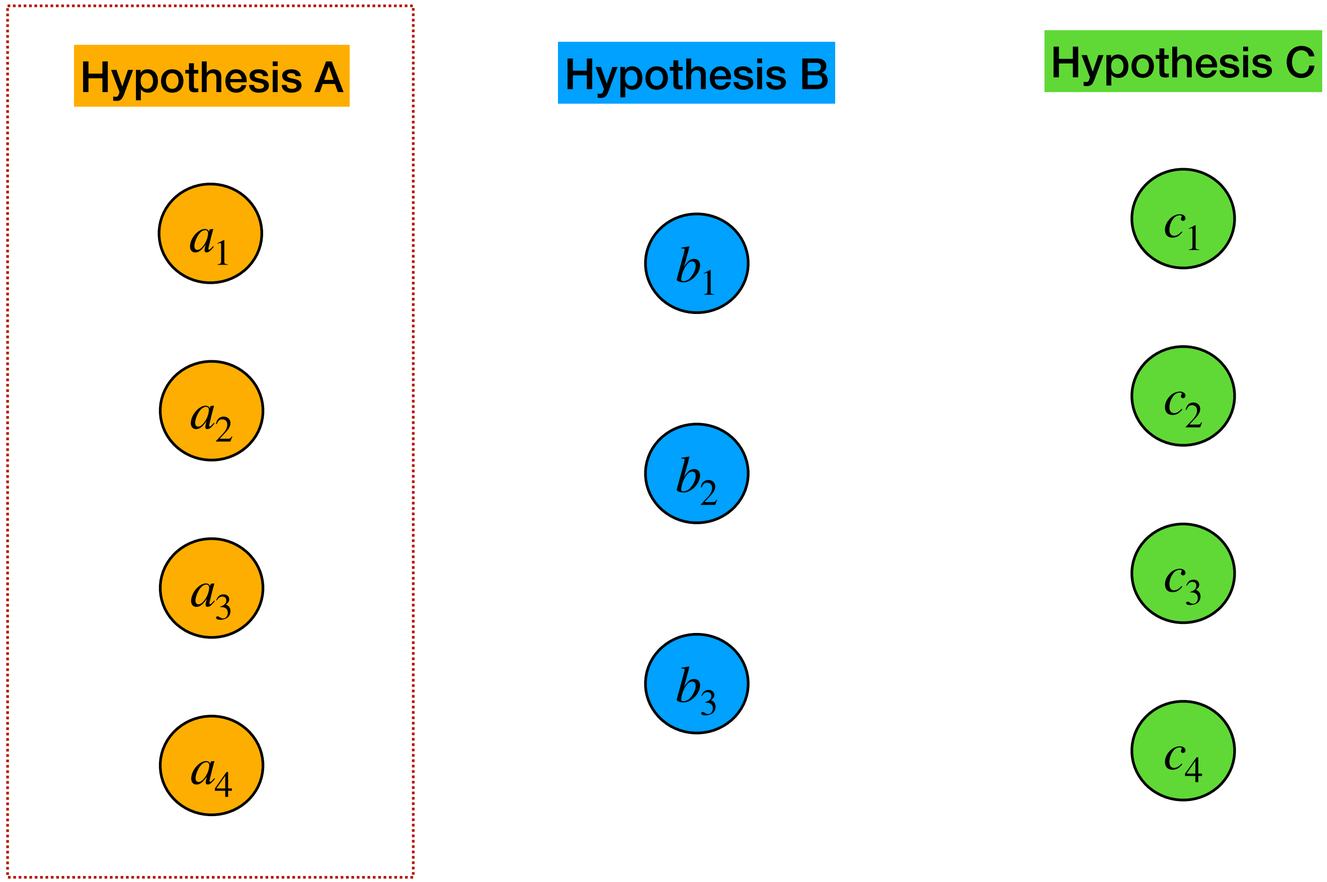
Preliminary: how DOVER works

Pair-wise incremental label mapping



Preliminary: how DOVER works

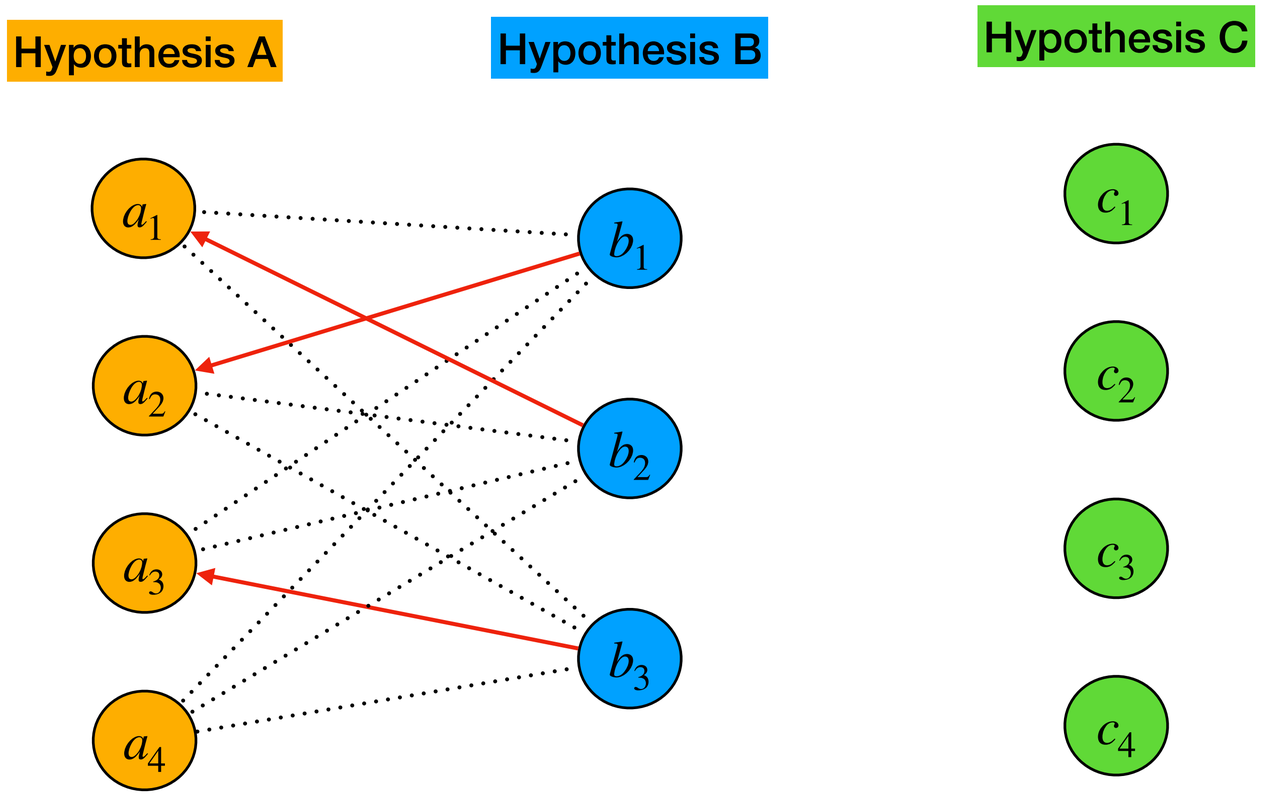
The “anchor” problem



- How to select the “**anchor**” hypothesis?
- Choose hypothesis with **lowest average DER** to all other hypothesis.

Proposed modification

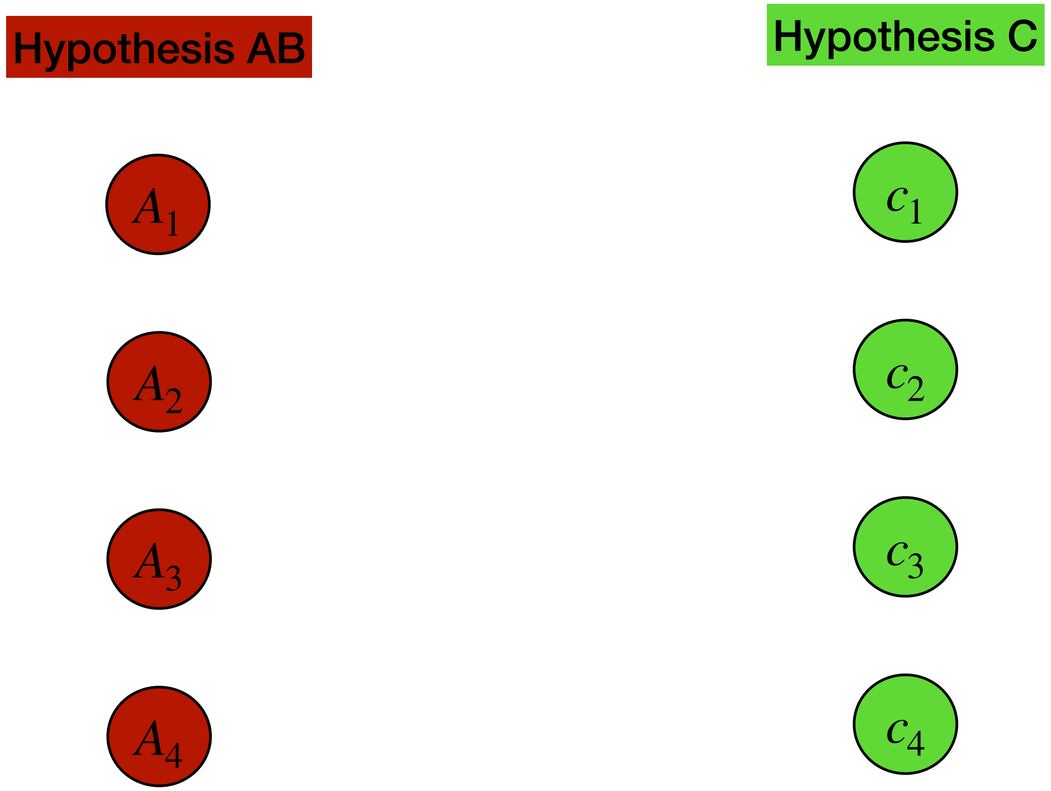
A graph “merge” operation



- After combining A and B, **update edge weights** to other hypotheses.

Proposed modification

A graph “merge” operation



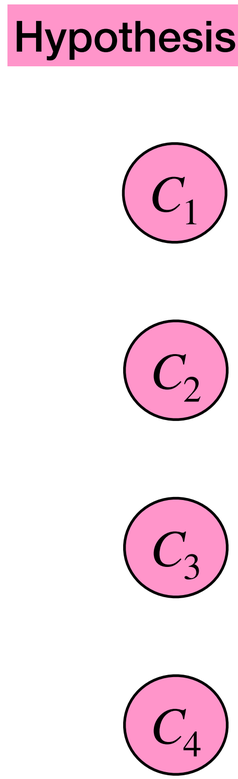
- We merge independent sets A and B, and create **new independent set AB** with union of their segments.

- For instance, if a_1 and b_2 were merged to create node A_1:

$$\Delta(A_1) = \Delta(a_1) \cup \Delta(b_2)$$

Proposed modification

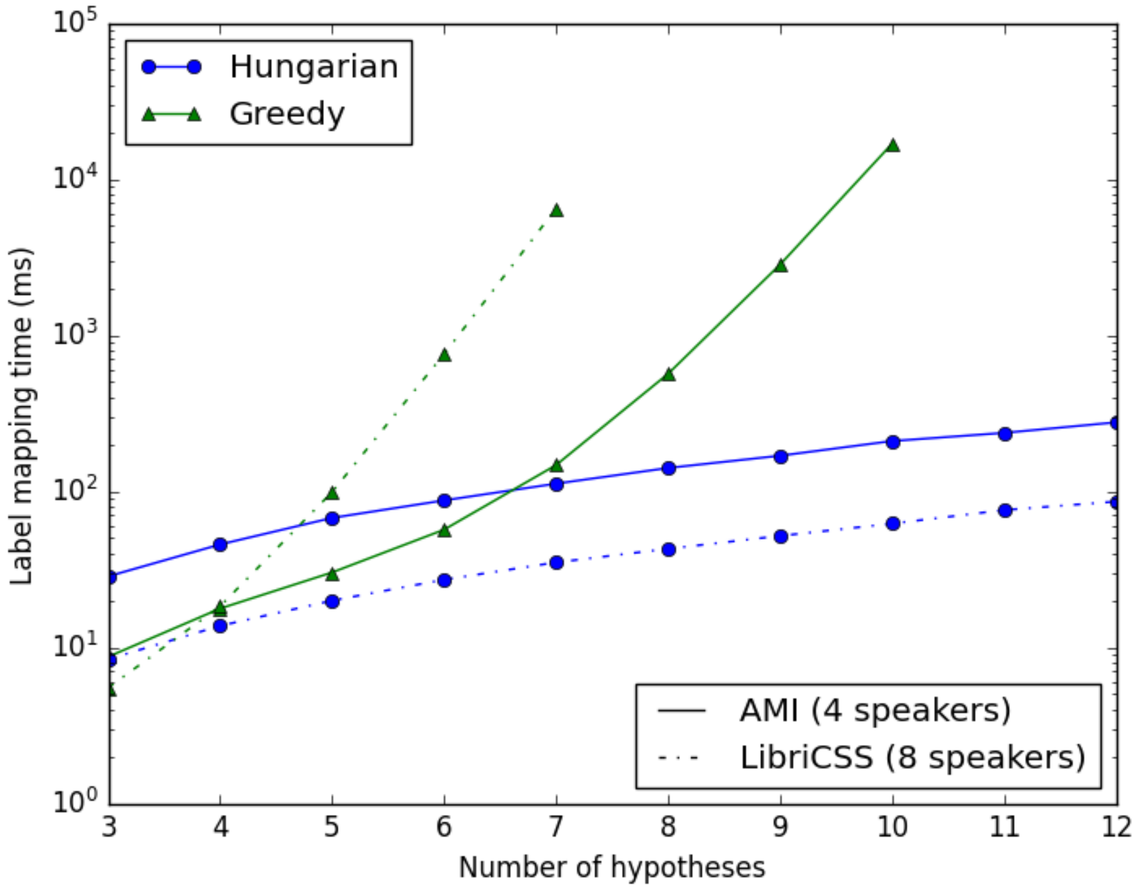
A graph “merge” operation



- Each pair-wise combination and “merge” is polynomial in speaker size.
- Need only *K* such combinations (for *K* hypotheses)

DOVER-Lap algorithm

Linear in number of input hypotheses!



- Can easily combine lots of hypotheses!
- Does not blow up even for large number of speakers.

Proposed modification

DER improvement over base DOVER algorithm

System	Spk. conf.	DER
Agglomerative hierarchical clustering	8.3	27.8
Spectral clustering	6.9	26.4
VBx	7.4	26.9
DOVER (base)	8.4	27.9
+ “merge” operation	**6.8**	**26.3**

Results on DIHARD-3

Does better than original DOVER-Lap

System	MS	FA	Spk. conf.	DER
TDNN x-vector + VBx	5.5	3.7	6.9	16.1
Res2Net x-vector + VBx	5.5	2.0	8.0	15.5
EEND-EDA	5.3	3.7	6.9	15.9
SC-EEND	5.7	1.3	5.8	12.9
VBx + EEND as post-processing	6.0	1.3	7.4	13.1
DOVER-Lap (greedy)	5.5	1.2	5.2	11.9
DOVER-Lap (Hungarian)	**5.5**	**1.2**	**4.9**	**11.6**

S. Horiguchi, et al. (2021). The **Hitachi-JHU DIHARD III System**: Competitive End-to-End Neural Diarization and X-Vector Clustering Systems Combined by DOVER-Lap. *ArXiv.*

Results on
Track-1 eval set

Proposed modification

Provable approximation bound for modified algorithm

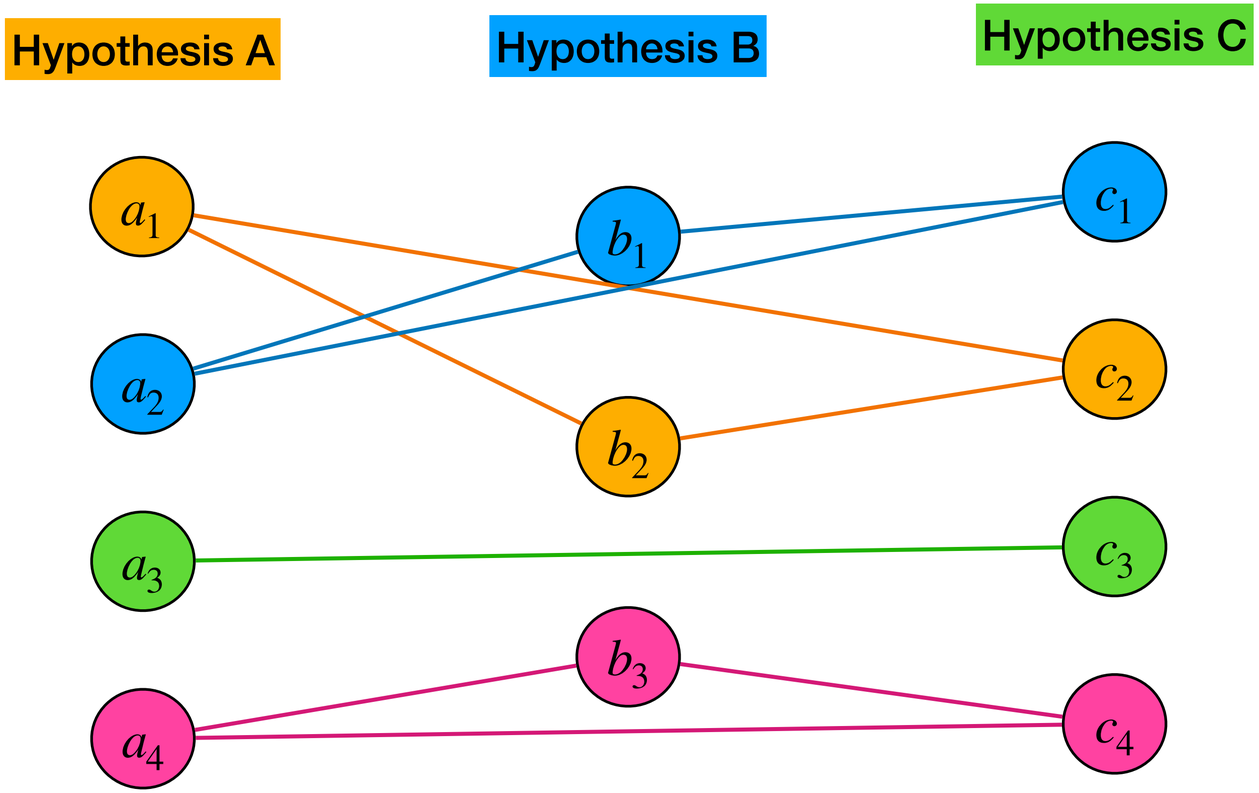
$$w(\Phi) \geq \frac{w(\Phi)^*}{C}$$

Maximum number of speakers in any hypothesis

Can we find approximation algorithms with tighter bounds?

Randomized local search

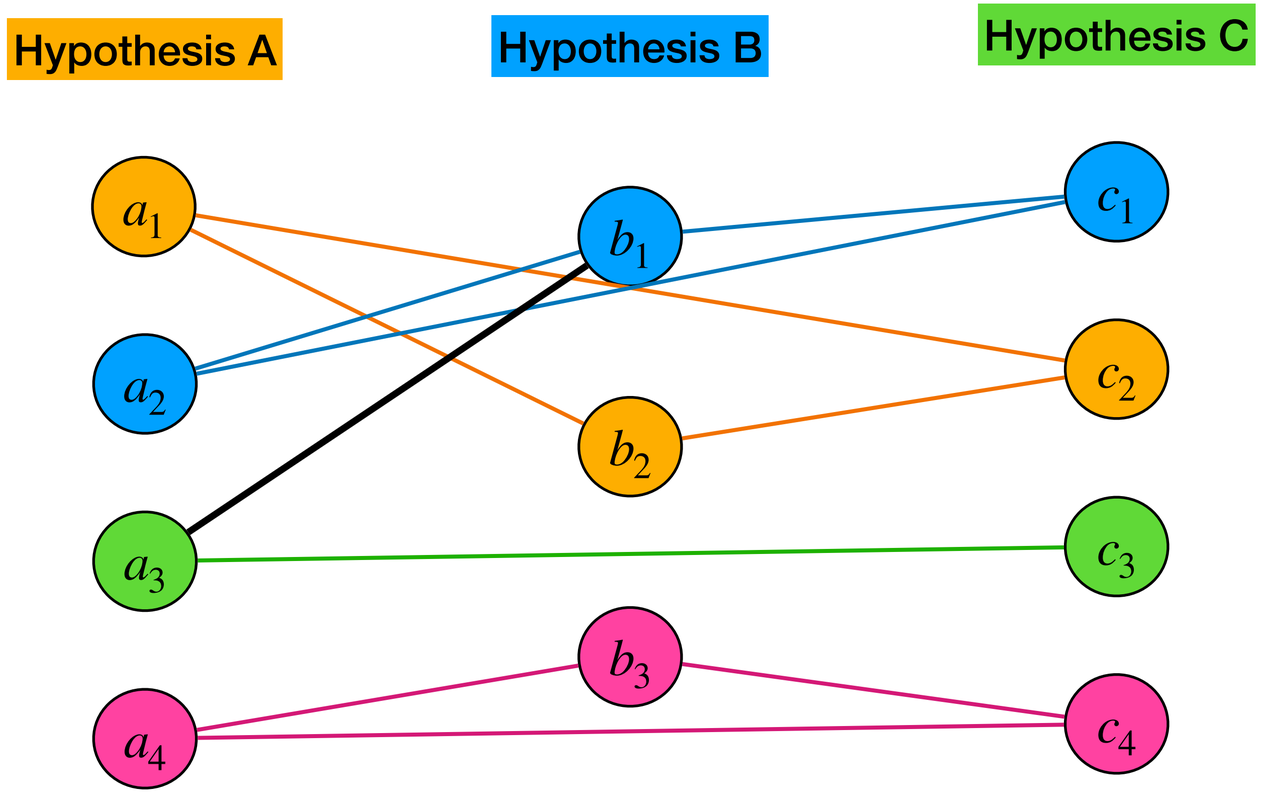
Near optimal solution with high probability



- Initialize with a random partition.

Randomized local search

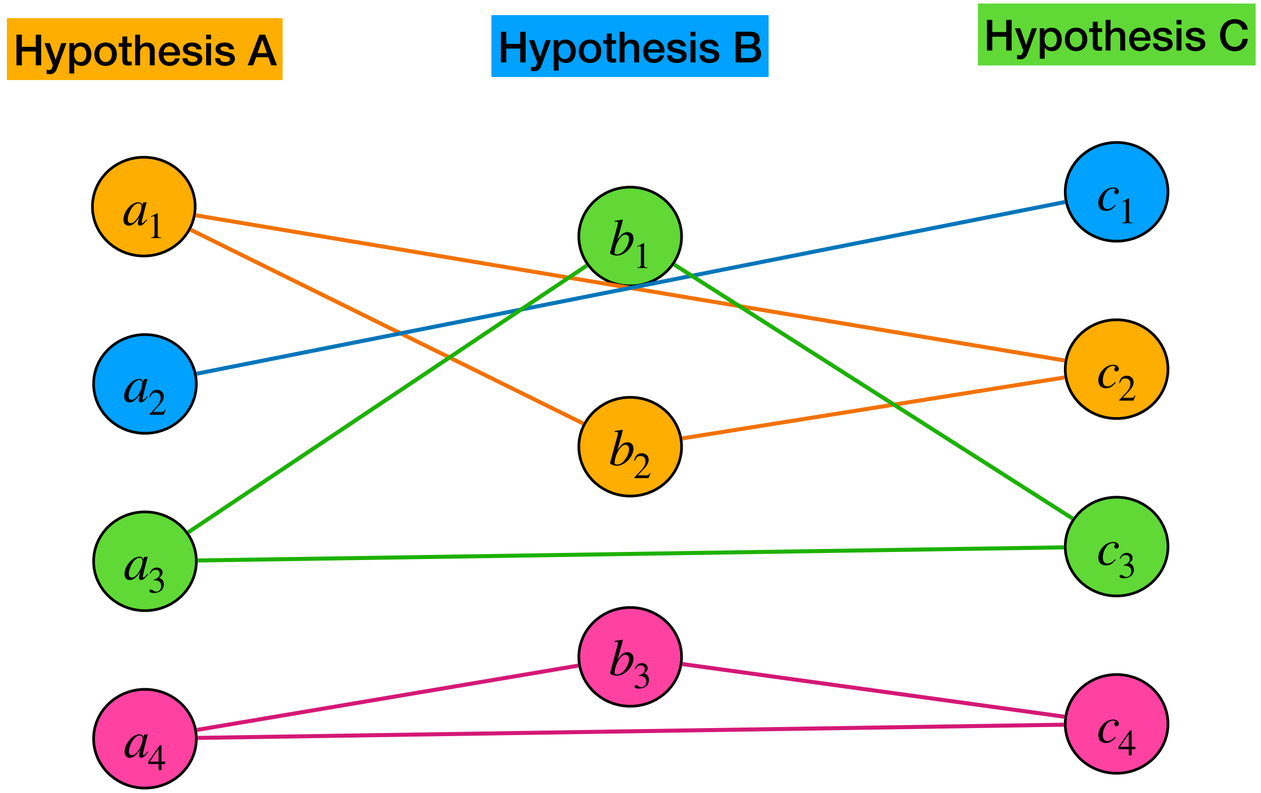
Near optimal solution with high probability



- Select an edge outside the partition based on probability distribution by edge weights.

Randomized local search

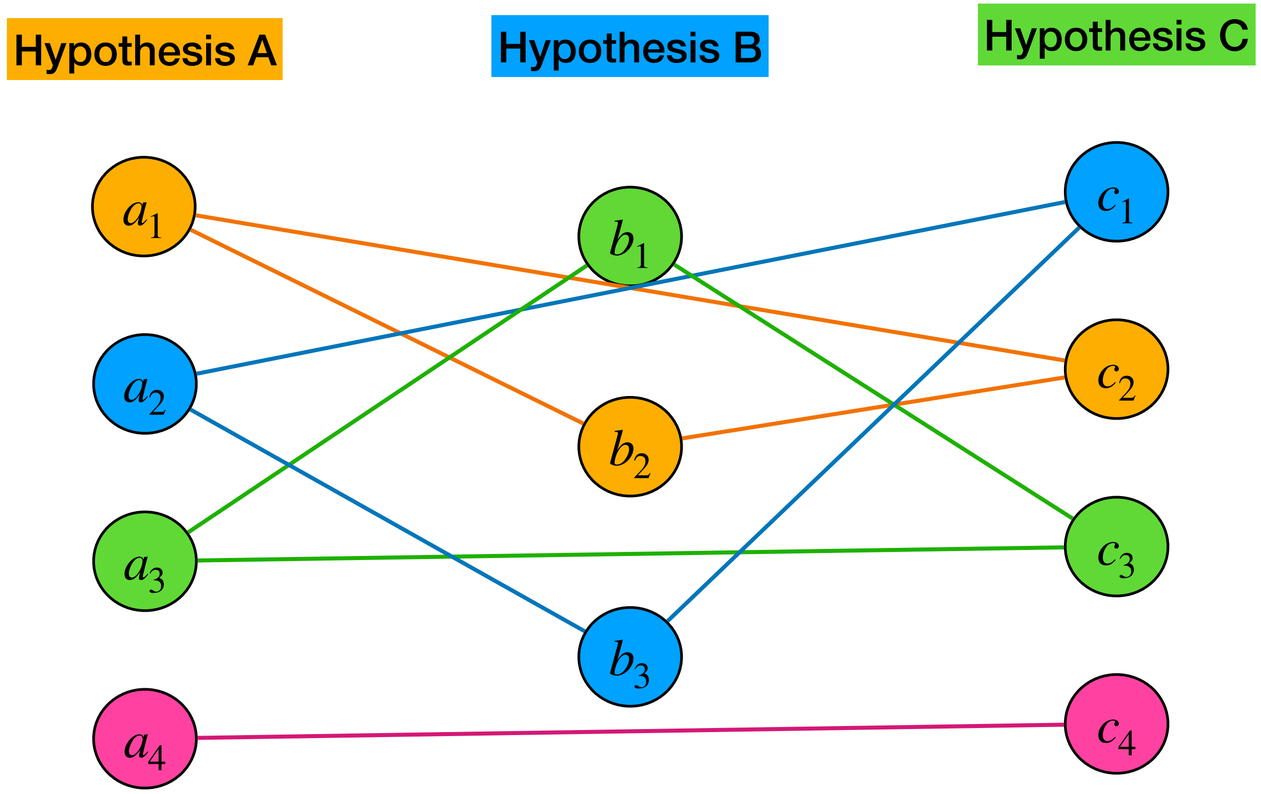
Near optimal solution with high probability



- Randomly swap one of the incident nodes with node in the clique.
- This increases the total edge weight inside the partition.

Randomized local search

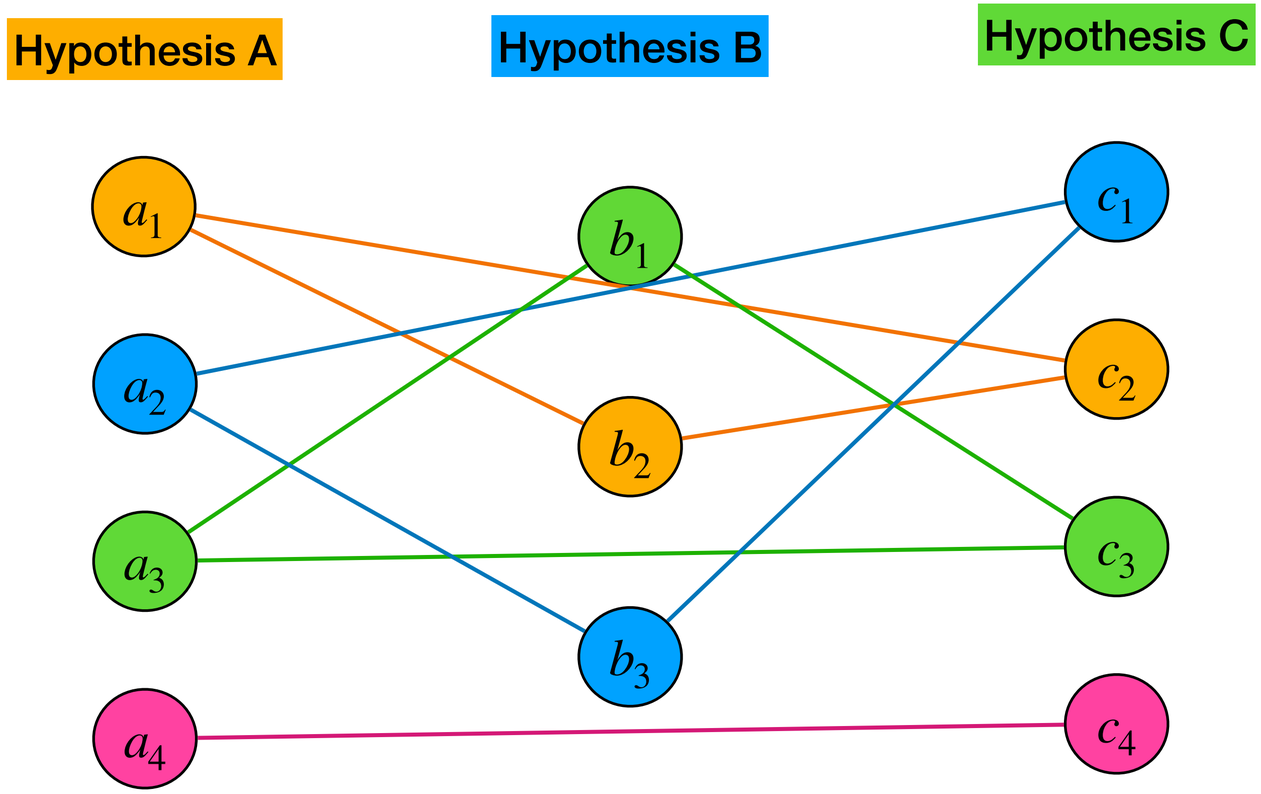
Near optimal solution with high probability



- Repeat for large number of steps.

Randomized local search

Near optimal solution with high probability



$$w(\Phi) \geq (1-\epsilon)w(\Phi^*)$$

with high probability

Results on AMI

DER improves over Hungarian method

System	DER
Overlap-aware SC	23.6
VB-based overlap assignment	21.5
Region proposal network	25.4
DOVER-Lap (Hungarian)	20.9
DOVER-Lap (Randomized Local Search)	**20.7**

The code is available!

All methods are still easy to use

```
$ dover-lap <output-rttm> <input-rttms> --label-mapping [greedy|hungarian|randomized]
```

Acknowledgements:

Shota Horiguchi (Hitachi), Mao-kui He (USTC)