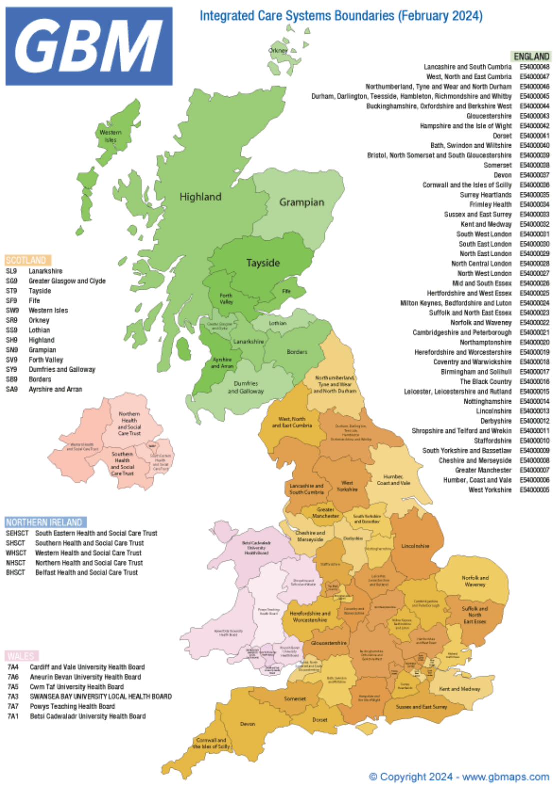
Local Beam Search Donor Networks

Lluís F Collell

Breu introducció:

This project aims to distribute 40 hospitals into 4 donor networks by optimizing two key factors: geographic proximity and similarity of population profiles. To achieve this, a Local Beam Search algorithm is combined with a Bayesian network that quantifies the probability that two centers share critical clinical cases. The algorithm explores assignments through hospital swaps and shifts, producing a network configuration that reduces logistical distances and strengthens synergies between hospitals with similar medical needs.


GBM
Integrated Care Systems Boundaries (February 2024)
ENGLAND
Lancashire and South Cumbria E54000048
West, North and East Cumbria E54000047
Northumberland, Tyne and Wear and North Durham E54000046
Durham, Darlington, Teesside, Hambleton, Richmondshire and Whitby E54000045
Buckinghamshire, Oxfordshire and Berkshire West E54000044
Gloucestershire E54000043
Hampshire and the Isle of Wight E54000042
Dorset E54000041
Bath, Swindon and Wiltshire E54000040
Bristol, North Somerset and South Gloucestershire E54000039
Somerset E54000038
Devon E54000037
Cornwall and the Isles of Scilly E54000036
Surrey Heartlands E54000035
Frimley Health E54000034
Sussex and East Surrey E54000033
Kent and Medway E54000032
South West London E54000031
South East London E54000030
North East London E54000029
North Central London E54000028
North West London E54000027
Mid and South Essex E54000026
Hertfordshire and West Essex E54000025
Milton Keynes, Bedfordshire and Luton E54000024
Suffolk and North East Essex E54000023
Norfolk and Waveney E54000022
Cambridgeshire and Peterborough E54000021
Northamptonshire E54000020
Herefordshire and Worcestershire E54000019
Coventry and Warwickshire E54000018
Birmingham and Solihull E54000017
The Black Country E54000016
Leicester, Leicestershire and Rutland E54000015
Nottinghamshire E54000014
Lincolnshire E54000013
Derbyshire E54000012
Shropshire and Telford and Wrekin E54000011
Staffordshire E54000010
South Yorkshire and Bassetlaw E54000009
Cheshire and Merseyside E54000008
Greater Manchester E54000007
Humber, Coast and Vale E54000006
West Yorkshire E54000005
SCOTLAND
SL9 Lanarkshire
SG9 Greater Glasgow and Clyde
ST9 Tayside
SF9 Fife
SW9 Western Isles
SR9 Orkney
SS9 Lothian
SH9 Highland
SN9 Grampian
SV9 Forth Valley
SY9 Dumfries and Galloway
SB9 Borders
SA9 Ayrshire and Arran
NORTHERN IRELAND
SEHSCT South Eastern Health and Social Care Trust
SHSCT Southern Health and Social Care Trust
WHSCT Western Health and Social Care Trust
NHSCT Northern Health and Social Care Trust
BHSCT Belfast Health and Social Care Trust
WALES
7A4 Cardiff and Vale University Health Board
7A6 Aneurin Bevan University Health Board
7A5 Cwm Taf University Health Board
7A3 SWANSEA BAY UNIVERSITY LOCAL HEALTH BOARD
7A7 Powys Teaching Health Board
7A1 Betsi Cadwaladr University Health Board
Orkney
Western Isles
Highland
Grampian
Tayside
Forth Valley
Fife
Lothian
Lanarkshire
Borders
Ayrshire and Arran
Dumfries and Galloway
Northern Health and Social Care Trust
Southern Health and Social Care Trust
Northumberland, Tyne and Wear and North Durham
West, North and East Cumbria
Lancashire and South Cumbria
West Yorkshire
Humber, Coast and Vale
Greater Manchester
Cheshire and Merseyside
Lincolnshire
Norfolk and Waveney
Suffolk and North East Essex
Herefordshire and Worcestershire
Gloucestershire
Somerset
Devon
Dorset
Cornwall and the Isles of Scilly
Kent and Medway
Sussex and East Surrey
© Copyright 2024 - www.gbmaps.com

Map of the United Kingdom's Integrated Care Systems (ICS) (February 2024).
ICS are healthcare regions that group hospitals, primary care, and social services to coordinate resources and planning within the NHS. Each colored area shows the boundaries of an ICS, including its official code and administrative headquarters. Source: © GBMaps 2024.

Problem representation

1. State encoding

Each state represents an assignment of the 40 hospitals to 4 networks. This is encoded as a list of length 40, where each position indicates the network to which a hospital belongs.

Each state s is modeled as a vector of length N=40:

$s = [s_0, s_1, \ldots, s_{39}]$, s_i pertany {0, 1, 2, 3}

This encoding is:

- **Compact** (one integer per hospital).
- **Easy to mutate** (swap, shift, random reassignment).
- **Compatible with any search structure;** the state is a NumPy array, allowing thousands of states to be evaluated per second.

2. Set of possible states

The number of different assignments is:

$$|S| = G^N = 4^{40} \approx 1{,}2\times10^{24},$$

This enormous state space makes exhaustive search infeasible and justifies the use of heuristics such as Local Beam Search.

3. Initial objective function

The objective is to minimize the average intra-network distance.

We precompute the Euclidean distance matrix D (40 × 40) between all hospitals so that each evaluation runs in O(N).

For a network g with Ng hospitals, the intra-network distance is defined accordingly.

$$\overline{dist}(\sigma, g) = \frac{2}{N_g \cdot (N_g - 1)} \sum_{i=1}^{N_g - 1} \sum_{j=i}^{N_g} dist\left(\sigma(g)_i, \sigma(g)_j\right)$$

The global objective function is the weighted average:

$$\min_{\sigma} \frac{1}{N} \sum_{g=1}^{G} N_g \cdot \overline{dist}(\sigma, g)$$

Local search

1. General scheme

We start from a Local Beam Search (LBS) with beam size B.

At each iteration:

- Generate neighbors from all states in the beam.
- Evaluate each neighbor using the objective function f(σ) (Step 1).
- Select the best B states to form the new beam.

Stopping criterion: the algorithm stops when there is no improvement over Δ consecutive iterations or when a maximum of K iterations is reached.

2. Creation of the initial beam

(Initialization with random assignments.)

3. Neighbor generation

We selected the *swap* and *shift* methods. Both perform small changes to the current states (local search) while allowing exploration of a wide variety of configurations.

- **Swap** uses four nested loops to exchange all values of each row with all values of every other row.
- **Shift**, using three nested loops, moves all values of one row to the end of all other rows.

4. Evaluation and selection

Implemented in **Step 3: Neighbor generation / Transition action selection**.

Explanation:

We create a list containing all newly generated neighboring states, sort it by average distance, and finally return only the first 5 states (those with the smallest distance values).

5. Beam local search

Stopping criterion based on state quality

We define a fixed number of iterations. In addition, we implement a stopping condition that checks whether there has been an improvement in the objective function of the current beam compared to the previous one. If no improvement is detected, the search stops.

A single non-improvement is sufficient to stop the algorithm because beam local search follows a Hill Climbing strategy; therefore, a non-improvement indicates that a local optimum has been reached.

Time complexity analysis

Summary of time complexity

- Distance matrix construction: O(N2)
- Average distance computation per state: O(N2/G)
- Neighbor generation: O(B · N2)
- Beam assignment: O(B · N2)
- Beam local search: O(K · B · N2)

Where:

- N: number of hospitals
- G: number of networks
- B: beam size
- K: number of iterations

If B increases, more states are generated and computation time increases.
If K increases, more iterations are performed and computation time scales linearly.

Comparison with other search algorithms

We compare our method with a Beam Search variant where iterations are applied to the same beam and the best results are not accumulated across iterations: **Fixed Beam Search (FBS)** or **Static Beam Search**.

Compared to LBS, FBS provides much higher solution diversity because it starts from multiple initial beams. This allows it to explore many more states, resulting in significantly higher computation time.

LBS, on the other hand, tends to converge faster to good solutions but has a higher risk of getting stuck in local minima. FBS, although slower, is more likely to find better solutions.

Bibliography and appendices

Python libraries used

- **numpy**: efficient numerical computation and vector operations; essential for generating and manipulating the distance matrix and managing states as arrays.

- itertools (standard): generation of combinations and permutations (combinations, permutations) to enumerate all swap and shift actions between hospitals.

- **random** (standard): selection and initialization of random states within the beam; seed control to ensure reproducibility.

- **time** (standard): execution time measurement for analyzing time complexity with respect to B and K.

- **matplotlib** (optional, for report figures): visualization of cost convergence and final spatial distribution of hospitals on a point map.

Note: no external graph libraries were required; Bayesian structures were implemented manually using dict-based logic and NumPy.

Information sources and applied criteria

- Official NumPy documentation (numpy.org): array manipulation and vectorized operations (.mean, boolean indexing) to optimize objective evaluation.

- Python Standard Library documentation (docs.python.org): use of itertools for neighbor generation and random for random initial states.

- **Russell & Norvig – *Artificial Intelligence: A Modern Approach* (3rd ed.)**:

    - Chapters 4–5: local and heuristic search (beam search, state space, stopping criteria).

    - Chapters 13–14: foundations of Bayesian network inference (variable elimination and sampling).

- **Koller & Friedman – *Probabilistic Graphical Models* (MIT Press, 2009)**: reference for factor definition, product operations, and marginalization used in the custom variable elimination implementation.

- AI course notes – University of Girona (2024/25): slides on search heuristics, formal definition of the objective function, and structure of the criticalBN and matchBN Bayesian networks.

- Moodle teaching materials (bn.py, inferencia.py, assignment PDF): provided skeleton code defining factor interfaces and the rejection sampling algorithm used as a starting point.

- **Articles and tutorials on Local Beam Search**: best practices on beam diversity, neighbor selection, and comparisons with hill climbing and genetic algorithms.

- **Matplotlib tutorials (matplotlib.org)**: generation of supporting figures included in the report, such as cost curves and hospital location maps.