

Optimal refugees allocation

Hackathon “Code for refugees”, 19-20 March 2022

During this hackathon

Our goal: lay out algorithm to advise on “optimal” allocation of refugees in tents/rooms within a camp.

- !! Requires info about each refugee: database management as e.g. in “challenge 2: unified biography”. Not covered here.
- No front-end etc. at present.
- We focused on defining and structuring the optimisation algorithm. “Brain” of the system.

Case 1: new batch of refugees to allocate

Workflow:

1. Refugees registration.
2. Aggregation of families (pre-existing and new refugees).
3. Check current camp status.
4. Compute allocation for new batch (+ family)
trying to maximise refugees' needs satisfaction.
5. Update camp status.

Case 2: camp or refugee(s) change condition

Examples:

- ▶ Refugee(s) has an issue and asks to leave a specific allocation.
- ▶ Refugee(s) has an preference and asks to join a specific allocation.
- ▶ Change in refugee medical/etc. condition.
- ▶ Camp structure changes (tends restructured...)

Workflow:

1. Update database,
from either camp operators or occupants through app.
2. Select occupants to move (create new batch)
either manually or through optimization procedure.
3. Compute allocation for new batch (as in case 1).
4. Update camp status.

Case 3: refugees leave the camp

Workflow:

1. Camp status update by camp operators.
2. If desired, compute new allocations (go to case 2).

List of needs

code4refugees.org/2022/03/18/sfida-1-alloggio-nei-campi

Primary needs:

- ▶ Keep family together.
- ▶ Provide support to people with handicaps etc.

Other needs:

- ▶ Cultural
- ▶ Religious
- ▶ Social

Entities definition

Refugee:

- ▶ Identity (name, internal ID, ...)
- ▶ Medical or handicap support needs
(can be split into several properties if relevant).
- ▶ Language / Nationality
- ▶ Religion / cultural group
- ▶ List of family members
- ▶ Other relevant features (to be added where needed)
- ▶ Allocation preferences
(for flexibility, free parameter set to account factors not included in the programme)

Entities definition

Tent:

- ▶ ID
- ▶ Medical or handicap support provided
- ▶ Accommodation availability
- ▶ List of current occupants
- ▶ Number of occupants per nationality
- ▶ Number of occupants per cultural group
- ▶ ...

Perhaps increase granularity and similarly define accommodation distribution in each tent.

Optimiser definition

Input:

- ▶ List of tents.
- ▶ List of refugees, already allocated and/or to be allocated

Output:

- ▶ Optimal refugees allocation.

Parametric and highly flexible algorithm:

- ▶ Can add/remove needs, to adapt to specific situation.
- ▶ Can manage and alter priority of each need, choosing the cost function parameters.

Cost function defined within matrix mathematical formalism.

Cost function definition

$$\max_X F = \text{tr}(X \cdot (w_m \cdot S_{SM}^T + w_r \cdot S_R^T + w_n \cdot S_N^T)) + \phi(X, S_F)$$

x_{ij} are X element.

$$x_{ij} = \begin{cases} 1 & \text{if refugee } i \text{ allocated in tent } j \\ 0 & \text{otherwise} \end{cases}$$

Such that

$$x_{ij} \in [0,1], \forall i \in [1..n], \forall j \in [1..m]$$

$$\sum_j x_{ij} = 1, \forall i \in [1..n]$$

$$\sum_i x_{ij} < d_j, \forall j \in [1..m]$$

Cost function definition

- ▶ S_{SM} : "medical service satisfaction". Shows satisfaction score when placing each refugee in any possible tent (greater score when need of each occupant is met).
- ▶ S_N : "nationality satisfaction". Shows satisfaction score when placing each refugee in any possible tent (greater score for greater homogeneity).
- ▶ S_R : "religion/cultural group satisfaction". Analogous to S_N .
- ▶ ϕ : describes correlations between family members (greater score when less members are separated)

Best model depends on how a real refugee camp works!
Interaction with on-field experts valuable.

How to approach the problem?

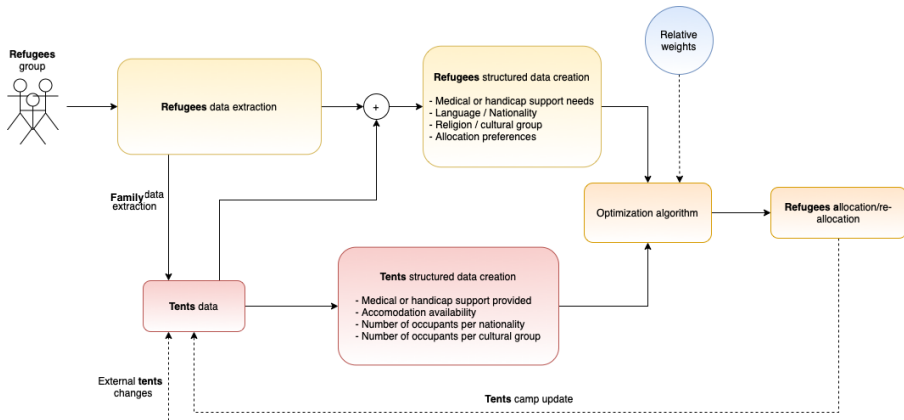
General solution:

- ▶ Definition of a cost function.
- ▶ Minimization over refugees allocation.

Features:

- ▶ For suitable (simple) cost functions, optimal solution easily computed.
(Sufficient to solve the real problem?)
- ▶ Can be made arbitrarily generic/complicated:
still solvable for small enough configuration space
(brute-force, random sampling...).

Block scheme



Call for action

- ▶ We have laid out the core of the system
- ▶ No user-experience and operator interfaces have been thought
- ▶ Need of integration with an mobile app or web app