Problem: assign rescuers to victims

• each victim may need more than one rescuer
• we want to minimise the distance that rescuers need to cover to reach their assigned victims.

Solution

• Establish the distance of every rescuer to its closest victim
• Select the closest k rescuers for every victim
• Disregard the matched victims & rescuers
• Finish if no additional victims have been saved in the last iteration

```
finish ← false;
until · agree · on finish do
  /* 1st Stage: */
  /* Establishing the distance to victims */
  D ← μZ.min₁(source, ⟨dist⟩Z);

  /* 2nd Stage: */
  /* Computing the rescuers paths */
  rescuers ← μZ.init ∪ ⟨⟨grad⟩⟩Z;

  /* 3rd Stage: Engaging rescuers */
  finish ← false;
  /* engaging the rescuers */
  engaged ← μZ.choose ∪ ⟨cograd⟩Z;
  /* updating victims and available rescuers */
  victim' ← victim;
  victim ← victim ∧ ¬saved;
  rescuer ← rescuer ∧ engaged ≠ Ø;
  /* determining termination */
  finish ← (victim’ == victim);

  /* 4th Stage: Checking success */
  if · agree · on ¬victim
    /* ended with success */
  else
    /* ended with failure */
```
Prototype Implementation

Randomly generated graph: 1000 landmarks, 5 victims and 10 rescuers.

Each victim can be reached by more than one rescuer, the closer one is selected.