Soft μ-Calculus for Computational Fields

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

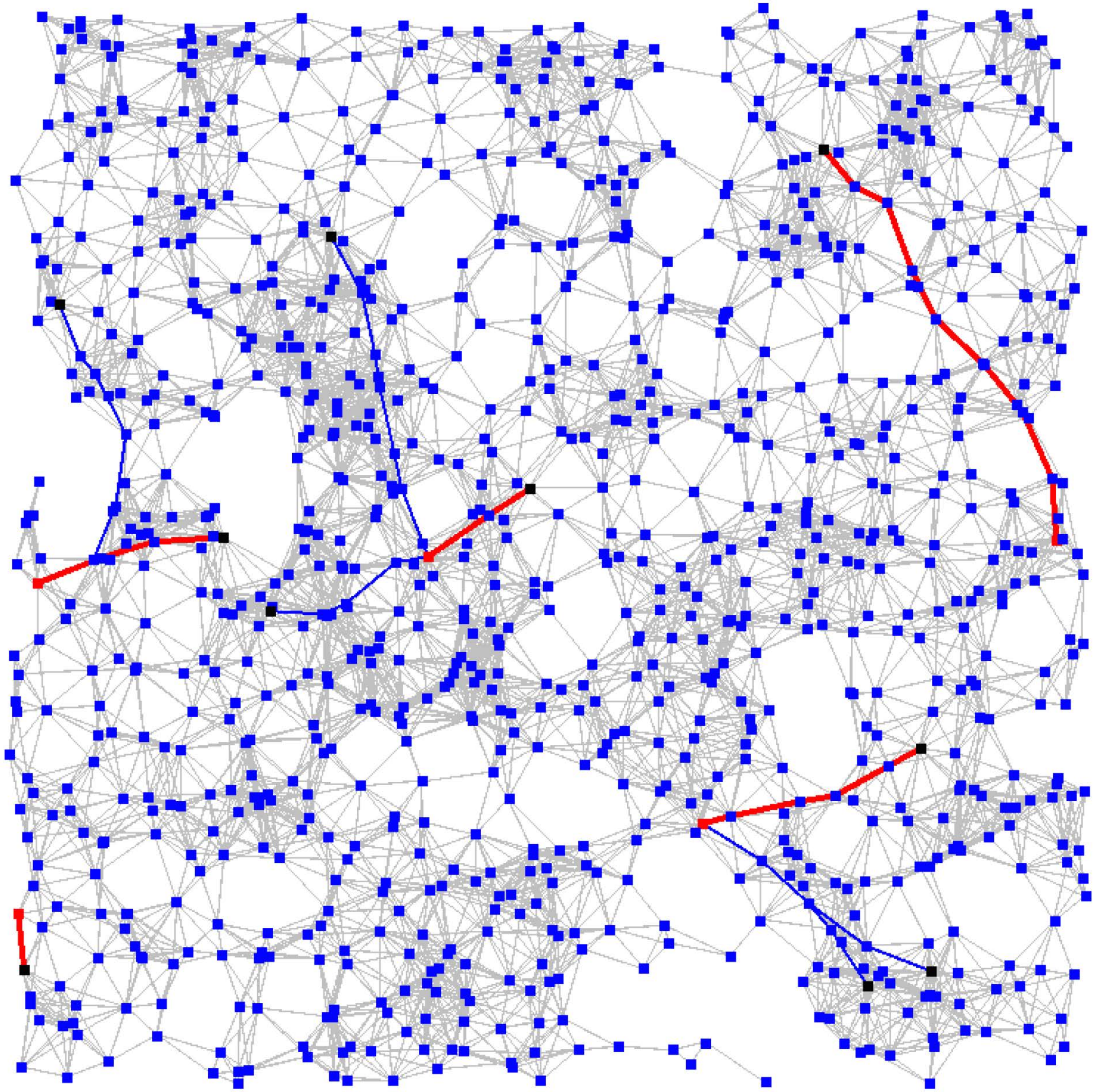
```
finish $\leftarrow$ $false$;
until · agree · on finish do
    /* 1st Stage: */
    /* Establishing the distance to victims */
    D $\leftarrow \mu Z.\mathrm{min}_1(\mathrm{source}, \langle \mathrm{dist} \rangle Z)$;

    /* 2nd Stage: */
    /* Computing the rescuers paths */
    rescuers $\leftarrow \mu Z.\mathrm{init} \cup \langle\langle \mathrm{grad} \rangle\rangle Z$;

    /* 3rd Stage: Engaging rescuers */
    finish $\leftarrow$ $false$;
    /* engaging the rescuers */
    engaged $\leftarrow \mu Z.\mathrm{choose} \cup \langle \mathrm{cograd} \rangle Z$;
    /* updating victims and available rescuers */
    victim' $\leftarrow$ victim;
    victim $\leftarrow$ victim $\wedge$ $\neg$saved;
    rescuer $\leftarrow$ rescuer $\wedge$ engaged $\neq \emptyset$;
    /* determining termination */
    finish $\leftarrow$ (victim$'$ $==$ victim);

/* 4th Stage: Checking success */
if · agree · on $\neg$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.