STSM Report

December 24, 2016

1 STSM Title

Improving preparedness of networks to disaster based failures

2 STSM Applicant

Dr. Deepak Mehta, United Technologies Research Centre, Ireland.

3 Host

Prof. Amaro Fernandes de Sousa, Institute of Telecommunications, University of Aveiro

4 Period

04/12/2016 to 10/12/2016

5 Working group

WG1: Large-scale natural disasters

6 Purpose of the STSM

When a disaster based failure occurs, it is important not only to recover as quickly as possible the network in the failure areas but also to minimize the dis-connectivity between network nodes outside the disaster areas. The purpose of the mission was to study the variants of the following problems:

1. The Critical Node Detection (CND) problem is defined as the set of nodes that, if removed from the network, maximally decreases the network connectivity.
2. The Robust Node Detection (RND) problem is defined as a set of nodes that, if made resilient, the impact of the dis-connectivity among the network due to failure of any non-robust nodes is minimised.

7 Description of the work

We discussed a MIP model of the CND problem, and an exact and a heuristic approaches developed by Amaro and his team. I presented some insights about constraint programming (CP), the CP model of CND, and worked on a lot of a CP approach for solving CND.

Following this we discussed how CND can be adapted to geographically correlated networks. In the context of geographically correlated network, when a disaster occurs failure of one or more nodes might imply failure of one or more nodes. For the CND problem this means that for finding k critical nodes some possible combination of nodes might not be valid because of the dependency between the failure of nodes. In general there might be many types of dependencies, but we restricted our attention to the following:

1. For a given pair of nodes depending on the distance between them the nodes that are in between would also fail.

2. For a given pair of nodes depending on the distance between them not only the nodes that are in between them would fail but also zero or more nodes are not in-between could fail if they rely on the links that are going through the region.

3. The above is a pessimistic assumption, which can be relaxed by considering the failure of the links that in-between a given pair of nodes depending on the distance between them. Of course a node that is not in between can still fail if all the links connected to it fails.

We discussed the extensions of the MIP and CP models to capture these dependencies.

A big chunk of time was then spent on the RND problem. The RND problem seems very challenging not only for solving but also for modelling. Intuitively, if we compute all node-sets of size k and compute the impact of their removal from the network, we can arrange these sets in the decreasing order of dis-connectivity that they cause. RND problem can then be viewed as a variant of hitting set problem, where we want to finding a set nodes of a given cardinality that maximise the coverage of the sets in a lexicographic order. As we do not know how many sets of nodes would be covered by the optimal solution of RND, the modelling of the problem using polynomial space itself is challenging. Notice that computing the first set in the order is already a NP-complete task.

After several rounds of attempts and discussions, the belief is that this problem is in higher complexity class than NP. However, this remains to be proven, and this work is in progress. We therefore investigated decomposition based approaches using both MIP and CP, and developed models for the subproblems.

8 Description of the Approaches

It is desirable to have a very efficient approach for solving CND as it will not only allow to solve the variants of CND efficiently but it can also help to efficiently solve

RND problem. Therefore, during my visit I also spent some time on developing a CP based approach for the following CP model:

8.1 CP Model for CND

Let $G = \langle V, E \rangle$ be an undirected input graph. Let k be the number of nodes that can be removed from V.

8.1.1 Variables

- A Boolean variable x_i is used for each node V. If $x_i = 0$ then it denotes that i is the critical node.
- A graph variable S is used to denote the survivable reachable graph when k nodes are removed from the input graph G. The lower bound of S is an empty graph and the upper bound contains all the nodes and all the pairs of nodes. The domain of S is any graph restricted by the above lower and upper bounds.
- Let $C \in [0, (|V| \cdot (|V| - 1))/2]$ be an integer variable that denotes the number of connections between pairs of nodes.

8.1.2 Constraints

- A Boolean variable x_i is true if and only if the node i is in the graph S.

$$\forall_{i \in V} : x_i = node\_bool(S, i)$$

- The number of nodes in the graph S is constrained:

$$nb\_nodes(S, |V| - k)$$

$$\sum_{i \in V} x_i = |V| - k$$

- If any pair of nodes in E are included in graph S then they must be connected

$$\forall_{(i,j) \in E} x_i \times x_j = edge\_bool(S, (i, j))$$

- We also enforce the transitivity property on the graph.

$$transitivity(S) \tag{1}$$

- The number of connections between the survivable nodes is equal to the number of edges in the graph S:

$$nb\_edges(S, C)$$

- The objective is to maximise the value of C.

Systematic search can consume a lot of time. Therefore, a specialised search strategy was implemented:

- I implemented a restart approach where the search would restart after a given number of failures.
- In each iteration the solver is asked to find a better solution than the currently known best.
- Additionally, in order to make the approach complete, the limit on the number of failures was increased if many unsuccessful attempts were made and it would be decrease if many successful iteration.
- during search an adaptive heuristic is used to ranks the variables based on the constraints that are reason for the failure

This search strategy reduced the time for solving CND significantly.

8.2 Robust Node Detection CP Approach

As mentioned before an exact model for RND seems exponential in size. Therefore, during the visit, approaches for both MIP and CP were investigated where the list of the sets of nodes for possible simultaneous failures can increase iteratively

Given a partial sorted list of sets of nodes that can fail simultaneously, the idea is to compute a set of robust nodes that can cover that partial list maximally and also check whether there exists another set of nodes not covered by the robust nodes that if removed can result in dis-connecting the network more than the best known solution. If that happens the partial list is updated by adding the new set, and the process continues until the problem becomes unsatisfiable. The following is a CP model that was developed during the visit:

8.2.1 Input Parameters

- Let r be an upper bound on the number of robust nodes.
- Let k be an upper bound on the number of nodes that can fail simultaneously.
- Let $\mathcal{C} = \{C_1, \ldots, C_m\}$ be a subset of sets of nodes of size k that can fail simultaneously. Initially $m \geq r$ and it is ensured that there exist an hitting-set of size r to cover sets $C_1, \ldots, C_i$ such that $i \geq r$ and $i \leq m$.

8.2.2 Variables

- Let R_i be an integer variable associated with each set of nodes $C_i \in \mathcal{C}$. The domain of R_i is the set C_i, which denotes the set of possible robust nodes to cover the set C_i. Any value assigned to R_i would be one of the robust nodes.
- Let $rnodes$ be a set variable that denotes the set of robust nodes.

8.2.3 Constraints

All the constraints of the previous model should also be considered. Therefore, in the following I am only showing the new constraints.

- The number of robust nodes cannot be equal to r. This is enforced by using the following constraint
$$nvalues(r, \langle R_1, \ldots, R_m \rangle)$$
The constraint ensures that r is the number of distinct robust nodes assigned to the variables $R_1, \ldots, R_m$

- The set of robust nodes is the union of R_i variables:
$$int\_values\_union(\langle R_1, \ldots, R_m \rangle, rnodes)$$
This constraint ensures that $rnodes = \{R_1\} \cup \ldots \cup \{R_m\}$

- Let non_failed_nodes denotes the set of nodes included in the network
$$non\_failed\_nodes = nodes\_set(G)$$

- The set of robust nodes must be included in the graph
$$subseteq(rnodes, non\_failed\_nodes)$$

- For the purpose of modelling each variable R_i also considers a dummy value 0. Thus, the set C_i is only covered if a non-zero value is assigned to R_i
$$\forall_{C_i \in \mathcal{C}} : \quad covered_i = 1 \Rightarrow R_i > 0$$

- The partial list is sorted based on the number of pairs that are disconnected if a set of nodes fail. The following constraint enforces that the sets of nodes should covered in a lexicographic order:
$$lex\_chain\_lex\_eq(covered_m, \ldots, covered_1)$$

- If the set C_i is not covered then its cost is set to 0, otherwise it is the original cost. The following constraint enforces that $cost_i = table_i[covered_i]$
$$\forall_{C_i \in \mathcal{C}} : \quad element(cost_i, table_i, covered_i)$$

- The maximum cost is computed using the following constraint:
$$mcost = max(cost_1, \ldots, cost_m)$$

- The cost of the newly compute set of nodes that if removed must be less than or equal to cost based on the robust nodes:
$$connectivity \leq mcost$$

This model is currently being implemented in constraint programming solver CHOCO.

9 Future collaboration and foreseen publications

The plan is to submit a paper on RND in the DRCN conference, and a journal paper on CND by the end of February.