A GRASP/VND Heuristic for the Generalized Steiner Problem with Node-Connectivity Constraints and Hostile Reliability

Sebastián Laborde, Franco Robledo, Pablo Romero, and Omar Viera

Facultad de Ingeniería - Universidad de la República
Montevideo - Uruguay

23 de marzo de 2021

Motivation

- Fiber-To-The-Home (FTTH) services have a large penetration throughout the world, and provides high data rates to the final customers.
- The number of applications and services on the internet has grown exponentially.
- A smart augmentation of the physical network is mandatory.
- Since the deployment of fiber optics is an important economical investment, the topological network design of *FTTH* networks should be revisited.

Objectives

- The goal is to interconnect distinguished nodes, called terminals, using large level of redundancy, and simultaneously, meeting large reliable constraints.
- Find a minimum cost solution that achieves a reliability threshold, where both nodes and links can fail with given probabilities.
- Understand the impact on the reliability of the solution networks, by increasing or decreasing the basic reliability of both nodes and links.
- Understand the cost-reliability trade-off, and how the reliability is naturally increased adding levels of redundancy between distinguished terminals.

Generalized Steiner Problem with Node-Connectivity Constraints and Hostile Reliability (GSPNCHR)

Definition (GSPNCHR)

Consider a simple undirected graph $G = (V, E)$, terminal-set $T \subseteq V$, link-costs $\{c_{i,j}\}_{(i,j) \in E}$ and connectivity requirements $R = \{r_{i,j}\}_{i,j \in T}$. Further, we assume that both links and non-terminal (Steiner) nodes fail with respective probabilities $P_E = \{p_e\}_{e \in E}$ and $P_{V-T} = \{p_v\}_{v \in V-T}$. Given a reliability threshold p_{min}, the goal is to build a minimum-cost topology $G_S \subseteq G$ meeting both the connectivity requirements R and the reliability threshold: $R_K(G_S) \geq p_{min}$, being $K = T$ the terminal-set.

GSPNCHR

$$\text{mín} \sum_{(i,j)\in E} c_{i,j} x_{i,j}$$

$$s.t. x_{ij} \geq y^{u,v}_{(i,j)} + y^{u,v}_{(j,i)} \forall (i,j) \in E, \forall u,v \in T, u \neq v \quad (1)$$

$$\sum_{(u,i)\in E} y^{u,v}_{(u,i)} \geq r_{u,v} \ \forall\, u,v \in T,\ u \neq v \quad (2)$$

$$\sum_{(j,v)\in E} y^{u,v}_{(j,v)} \geq r_{u,v} \ \forall\, u,v \in T,\ u \neq v \quad (3)$$

$$\sum_{(i,p)\in I(p)} y^{u,v}_{(i,p)} - \sum_{(p,j)\in I(p)} y^{u,v}_{(p,j)} \geq 0,\ \forall p \in V - \{u,v\},\ \forall u,v \in T,\ u \neq v \quad (4)$$

GSPNCHR

$$\sum_{(s,i)\in E} x_{s,i} \leq M\hat{x}_s,\ \forall s \in V - T \tag{5}$$

$$R_K(G_S(\{x_{ij}\})) \geq p_{min} \tag{6}$$

$$x_{(i,j)} \in \{0,1\}\ \forall (i,j) \in E \tag{7}$$

$$\hat{x}_i \in \{0,1\}\ \forall i \in V - T \tag{8}$$

$$y^{u,v}_{(i,j)} \in \{0,1\}\ \forall (i,j) \in E,\ \forall u,v \in T,\ u \neq v \tag{9}$$

GRASP/VND

- GRASP and VND are well known metaheuristics that have been successfully used to solve many hard combinatorial optimization problems
- GRASP is a powerful multi-start process which operates in two phases. A feasible solution is built in a rst phase, whose neighborhood is then explored in the Local Search Phase.
- VND explores several neighborhood structures in a deterministic order. Its success is based on the simple fact that different neighborhood structures do not usually have the same local minimum.

Network Design

Alg 1 $sol = NetworkDesign(G_B, iter, k, p_{min}, P_E, P_{V-T}, simiter)$

$i \leftarrow 0;\ P \leftarrow \emptyset;\ sol \leftarrow \emptyset$
while $i < iter$ **do**
    $\overline{g} \leftarrow Construction(G_B, P, k)$
    $g_{sol} \leftarrow VND(\overline{g}, P)$
    $reliability \leftarrow RVR(g_{sol}, P_E, P_{V-T}, simiter)$
    if $reliability > p_{min}$ **then**
        $sol \leftarrow sol \cup \{g_{sol}\}$
    end if
end while
return sol

Alg 2 $(sol, P) = Construction(G_B, C, R, k)$

$g_{sol} \leftarrow (S_D^{(l)}, \emptyset)$; $m_{i,j} \leftarrow r_{i,j}$; $P_{i,j} \leftarrow \emptyset, \forall i,j \in S_D^{(l)}$; $A_{i,j} \leftarrow 0, \forall i,j \in S_D^{(l)}$
while $\exists\, m_{i,j} > 0 : A_{i,j} < MAXATTEMPTS$ **do**
  $(i,j) \leftarrow ChooseRandom(S_D^{(l)} : m_{i,j} > 0)$
  $\overline{G} \leftarrow G_B \setminus P_{i,j}$
  for all $(u,v) \in E(\overline{G})$ **do**
    $\overline{c}_{u,v} \leftarrow c_{u,v} \times 1_{\{(u,v) \notin g_{sol}\}}$
  end for
  $L_p \leftarrow KSP(k, i, j, \overline{G}, \overline{C})$
  if $L_p = \emptyset$ **then**
    $A_{i,j} \leftarrow A_{i,j} + 1$; $P_{i,j} \leftarrow \emptyset$; $m_{i,j} \leftarrow r_{i,j}$
  else
    $p \leftarrow SelectRandom(L_p)$; $g_{sol} \leftarrow g_{sol} \cup \{p\}$
    $P_{i,j} \leftarrow P_{i,j} \cup \{p\}$; $m_{i,j} \leftarrow m_{i,j} - 1$
    $(P, M) \leftarrow GeneralUpdateMatrix(g_{sol}, P, M, p, i, j)$
  end if
end while
return (g_{sol}, P)

Local Search Phase

VND

The goal is to combine a rich diversity of neighborhoods in order to obtain an output that is locally optimum solution for every feasible neighborhood. Here, we consider three neighborhood structures to build a VND.

1. SwapKeyPathLocalSearch
2. KeyPathLocalSearch
3. KeyTreeLocalSearch

Alg 3 $g_{sol} = SwapKeyPathLocalSearch(G_B, C, g_{sol}, P)$

```
improve ← TRUE
while improve do
    improve ← FALSE
    K(g_sol) ← {p_1, ..., p_h} /* Key-path decomposition of g_sol */
    while not improve and ∃ key-paths not analyzed do
        p ← (K(g_sol)) /* Path not analyzed yet */
        (g_sol, improve) ← FindSubstituteKeyPath(g_sol, p, P)
    end while
end while
return g_sol
```

KeyPathLocalSearch

Alg 4 $g_{sol} = KeyPathLocalSearch(G_B, C, g_{sol})$

$improve \leftarrow TRUE$
while $improve$ **do**
    $improve \leftarrow FALSE$
    $K(g_{sol}) \leftarrow \{p_1, \dots, p_h\}$ /* Key-path decomposition of g_{sol} */
    while not $improve$ **and** $\exists$ **key-paths not analyzed do**
        $p \leftarrow (K(g_{sol}))$ /* Path not analyzed yet, with extremes u and v */
        $\hat{\mu} \leftarrow < NODES(p) \cup S_D \setminus NODES(g_{sol}) >$ /* Induced subgraph $\hat{\mu}$ */
        $\hat{p} \leftarrow Dijkstra(u, v, \hat{\mu})$
        if $COST(\hat{p}) < COST(p)$ **then**
            $g_{sol} \leftarrow \{g_{sol} \setminus p\} \cup \{\hat{p}\}$
            $improve \leftarrow TRUE$
        end if
    end while
end while

return g_{sol}

KeyTreeLocalSearch

Alg 5 $g_{sol} = KeyTreeLocalSearch(G_B, C, g_{sol})$

$improve \leftarrow TRUE$
while $improve$ **do**
  $improve \leftarrow FALSE$
  $X \leftarrow KeyNodes(g_{sol})$ /* Key-nodes from g_{sol} */
  $\overline{S} \leftarrow S_D \setminus NODES(g_{sol})$
  while not $improve$ **and** $\exists$ **key-nodes not analyzed do**
    $v \leftarrow X$ /* Key-node not analyzed yet */
    $[g_{sol}, improve] \leftarrow GeneralRecConnect(G_B, C, g_{sol}, v, \overline{S})$
  end while
end while
return g_{sol}

RVR

- Recursive Variance Reduction Method.
- Objective to reduce the original network into several smaller networks recursively.
- Construct an unbiased estimator of reliability, with less variance than the raw Monte Carlo.

Alg 6 $RVR(G, K, p_v, p_e)$

If *terminals*=1, **return** 0
Elseif $\phi(G, K) = 1$, **return** 1.
Else
$D := \mathit{GetKExtendedCut}(G)$
$Q_D := \mathit{AllFailedProb}(D)$
$\mathit{index} := \mathit{GetRandomItem}(D)$
$c := D[\mathit{index}]$
$\mathit{remove}(G, D, \mathit{index} - 1)$
$\mathit{add}(G, c)$
$\mathit{return}\ Q_D + (1 - Q_D) \times RVR(G)$
EndIf

Test Set

- An extensive computational study was conducted using the *NetworkDesign* algorithm.
- We adapt known instances of the *TSPLIB* library adapting them to our problem adding probabilities of failure in nodes and edges and the connectivity requirements between nodes.
- There are no benchmark cases for our specific problem.
- Reliability threshold $p_{min} = 0{,}8$.
- Node and edge reliabilities $p_v = 0{,}99$ y $p_e = 0{,}95$ respectively.
- Number of iterations in *NetworkDesign* for each instance $iter = 100$, and number of replications for the *RVR* 10^4 simulation method.
- We want to understand the sensitivity of the solution to disturbances in elementary reliabilities. Therefore, different values were used for the elementary reliabilities for both the Steiner nodes and links $p_v, p_e \in \{0{,}99, 0{,}97, 0{,}95\}$.

Numerical Results II

Cuadro: Test Set

Problem	*% T*	*%Rel*	*% Req*	*Iter_ND*	*Iter_RVR*	#
att48	20-35-50	99-95	100-0-0	100-100-100	10^4	3
berlin52	20-35-50	99-95	100-0-0	100-100-100	10^4	3
brazil58	20-35-50	99-95	100-0-0	100-100-100	10^4	3
ch150	20-35-50	99-95	100-0-0	100-100-100	10^4	3
d198	20-35-50	99-95	100-0-0	20-20-20	NA	3
eil51	20-35-50	99-95	100-0-0	100-100-100	10^4	3
gr202	20-35-50	99-95	100-0-0	100-100-100	10^4	3
kroA100	20-35-50	99-95	100-0-0	100-100-100	NA	3
kroB100	20-35-50	99-95	100-0-0	100-100-100	NA	3
kroB150	20-35-50	99-95	100-0-0	100-20-20	NA	3
kroB200	20-35-50	99-95	100-0-0	20-20-20	NA	3
lin105	20-35-50	99-95	100-0-0	100-100-100	NA	3
pr152	20-35-50	99-95	100-0-0	20-20-20	NA	3
tsp225	20-35-50	99-95	100-0-0	50-50-50	10^4	3
rd400	20-35-50	99-95	100-0-0	50-50-50	10^4	3
berlin52(E)	20	99-90	65-25-10	100	10^4	1
att48(E)	35	99-90	65-25-10	100	10^4	1
brazil58(E)	50	99-90	65-25-10	100	10^4	1
rd100(E)	35	99-90	65-25-10	20	NA	1

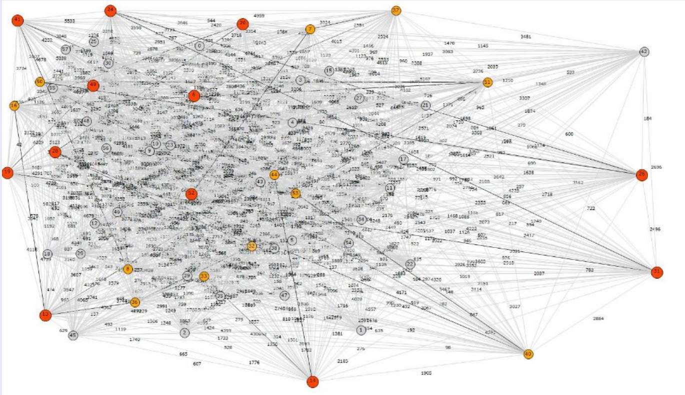
Numerical Results III

Cuadro: GRASP/VND Effectiveness

Problem	% *T*	%*IC*	% *IVND*	*CPU*(*s*)	$\overline{R}$	$\overline{Var}$
att48	20	99.27	34.61	11.466	0.967	7.608E-07
att48	35	98.6	36.83	29.769	0.943	3.448E-06
att48	50	98.22	37.1	65.904	0.927	5.322E-06
berlin52	20	98.98	30.55	30.605	0.937	3.294E-06
berlin52	35	99.06	33.93	33.433	0.938	3.19E-06
berlin52	50	98.02	33.48	106.945	0.907	6.487E-06
brazil58	20	98.92	31.96	62.377	0.885	6.722E-06
brazil58	35	99.25	39.45	68.891	0.86	8.347E-06
brazil58	50	98.75	35.26	103.553	0.91	7.093E-06
ch150	20	99.76	37.51	222.552	0.8559	1.029E-05
ch150	35	99.72	36.65	546.652	0.8803	9.033E-05
gr202	20	99.89	32.43	100.162	0.8231	1.224E-05
gr202	35	99.75	34.56	200.698	0.8414	1.11E-05
gr202	50	99.74	33.36	600.629	0.8303	1.279E-05
rd400	20	99.94	35.84	88.214	0.8094	14.22E-05
rd400	35	99.94	33.54	504.103	0.8537	11.89E-05
rd400	50	99.93	33.16	980.701	0.8643	11.51E-05
Average	35	99.28	34.72	220.980	0.884	3.28E-05

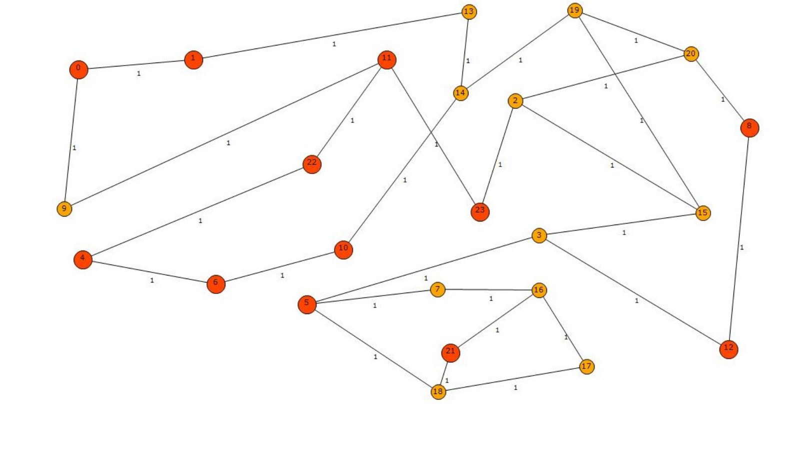
Brazil58

20 % terminals nodes, elemental reliability of Steiner nodes 99 %, link reliability 95 % and 100 % 2 node-disjoint paths. Red terminal nodes do not fail, orange and gray Steiner nodes.



Brazil58 NetworkDesign Output

Result cost 25106 (32 % improvement over construction) and reliability 92 %.



Contributions of this work

- We study the topological design of high reliability networks.
- Our goal is to combine purely deterministic aspects such as connectivity with probabilistic models derived from network reliability.
- The Generalized Steiner Problem with Node-Connectivity Constraints and Hostile Reliability (GSPNCHR) is introduced.
- We formally prove that GSPNCHR belongs to the NP-Hard class.
- As a consequence, a GRASP / VND methodology is proposed.

Conclusions

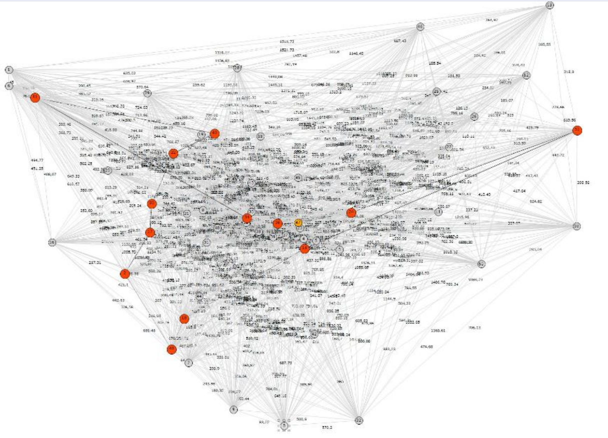
- Since the reliability assessment for the hostile model also belongs to the NP-Hard class, we adopt an excellent point reliability estimate, known as Recursive Variance Reduction (RVR). This method is unbiased, accurate, and has small variance, as the results show.
- The improvement provided by the VNS phase after Greedy Construction ranges from 25.25
- Average reliability for all networks varies between 82.31
- Our results highlight that the model is robust under non-terminal node-failures, rather than link-failures.

End

Thanks for your attention.

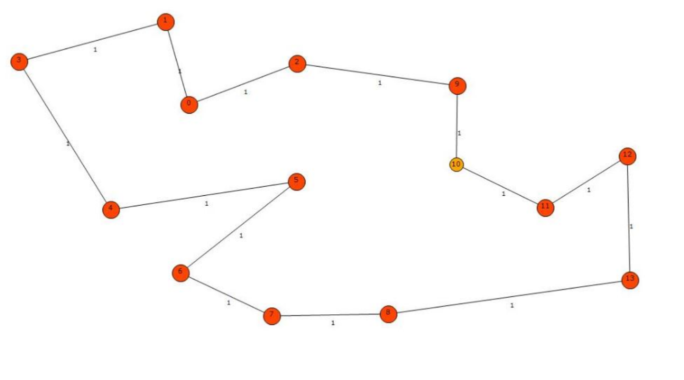
Berlin52

20 % de nodos terminales, confiabilidad elemental de nodos de Steiner 99 %, confiabilidad de enlaces 95 % y 100 % 2 caminos nodo-disjuntos. Nodos terminales rojos no fallan, nodos de Steiner naranjas y grises.



Berlin52 Salida NetworkDesign

Costo resultado 4534 (31 % de mejora con respecto a la de construcción) y confiabilidad 84 %.



Definition (key-node)

Un key-node v en una solución factible $v \in g_{sol}$ es un nodo de Steiner (no terminal) con grado mayor o igual a tres.

Definition (key-path)

Un key-path p en una solución factible $p \subseteq g_{sol}$ es un camino elemental donde todos los nodos intermedios no son terminales con grado dos en g_{sol}, y los nodos extremos son terminales o key-nodes.

Definition (key-tree)

Sea $v \in g_{sol}$ un key-node perteneciente a una solución factible g_{sol}. El key-tree asociado a v, denotado como T_v, es un árbol compuesto por todos los key-paths que se encuentran en un punto común (i.e., el key-node v).

Preguntas Claves

1

- ¿Cuántas redes factibles superan el umbral de confiabilidad, dado el modelo probabilístico completo (p_{min} : 0,98, P_E : 0,99, P_{V-T} : 0,99)?
- La cantidad de soluciones que cumplen con el umbral de confiabilidad es alta. Alcanza el 100 % en la mayoría de los casos.

2

- ¿Cuál es la sensibilidad del modelo con respecto a las confiabilidades elementales? ¿Es mejor aumentar la confiabilidad elemental de los enlaces, o la confiabilidad de los nodos Steiner, para cumplir con un umbral de confiabilidad exigente?
- El modelo es más sensible a fallas de enlaces que a fallas de nodos. Un aumento en la confiabilidad de los enlaces tiene un mayor impacto que un aumento correspondiente en la confiabilidad de los nodos.

3

- ¿Cuántas redes sobreviven en promedio, para cualquier modelo probabilístico dado? Comprender la sensibilidad del modelo con respecto a los requisitos de conectividad $r_{i,j} \in \{2, 3, 4\}$?
- Podemos apreciar que un aumento en los requisitos de conectividad de la red implica necesariamente un aumento correspondiente en el porcentaje de redes que cumplen con el umbral de confiabilidad, y viceversa.

Resultados V

Cuadro: Soluciones factibles con $R \geq 0{,}98$, $p_v = 0{,}99$ fijo y confiabilidad de enlaces variable.

Instance	$p_e = 0{,}99$	$p_e = 0{,}97$	$p_e = 0{,}95$
att48 T20	100	90	12
att48 T35	100	53	0
att48 T50	100	20	0
berlin52 T20	100	41	0
berlin52 T35	100	50	0
berlin52 T50	100	1	0
brazil58 T20	99	15	0
brazil58 T35	97	0	0
brazil58 T50	100	5	0
ch150 T20	100	0	0
ch150 T35	100	0	0
ch150 T50	100	0	0
gr202 T20	99	0	0
gr202 T35	100	0	0
gr202 T50	100	0	0
rd400 T20	100	0	0
rd400 T35	100	0	0
rd400 T50	100	0	0
Average	99.72	15.28	0.67

Resultados VI

Cuadro: Soluciones factibles con $R \geq 0{,}98$, $p_e = 0{,}99$ fijo y confiabilidad de nodos variable.

Instance	$p_v = 0{,}99$	$p_v = 0{,}97$	$p_v = 0{,}95$
att48 T20	100	100	99
att48 T35	100	98	96
att48 T50	100	100	99
berlin52 T20	100	100	80
berlin52 T35	100	99	93
berlin52 T50	100	100	100
brazil58 T20	99	59	41
brazil58 T35	97	43	9
brazil58 T50	100	99	81
ch150 T20	100	60	20
ch150 T35	100	98	76
ch150 T50	100	100	97
gr202 T20	99	80	30
gr202 T35	100	69	16
gr202 T50	100	100	76
rd400 T20	100	16	2
rd400 T35	100	98	80
rd400 T50	100	100	100
Average	99.72	84.39	66.39

Resultados VII

Cuadro: Soluciones con $R \geq 0{,}98$ ($p_v = 0{,}99$ - $p_e = 0{,}97$)

Rel %99- %97	% Feasible solutions with $R \geq 0{,}98$
att48 T20 (100-0-0)	90
att48 T20 (65-25-10)	100
att48 T20 (0-100-0)	100
att48 T20 (0-0-100)	100
eil51 T20 (100-0-0)	76
eil51 T20 (65-25-10)	100
eil51 T50 (100-0-0)	54
eil51 T50 (65-25-10)	100
berlin52 T20 (100-0-0)	41
berlin52 T20 (65-25-10)	100
brazil58 T50 (100-0-0)	5
brazil58 T50 (65-25-10)	100
kroA100 T35 (100-0-0)	0
kroA100 T35 (65-25-10)	100
kroB100 T20 (100-0-0)	3
kroB100 T20 (65-25-10)	100

Future Work

- The interplay between topological network design and network reliability is not yet well understood. Some local lookups were proposed here, essentially using key-path and key-tree replacements, in order to reduce costs and preserve feasibility.
- A current line of research is to introduce transformations that increase reliability.
- Developing local searches that increase reliability and reduce costs would enrich the current solution.
- Another possibility for future work is to enrich the number of local searches and consider probabilistic transitions between them.
- Improve as much as possible the algorithm, suitable for high-performance computing and thus reduce computing times.