

# Topological Optimization of Fault-Tolerant Networks meeting Reliability Constraints

Sebastián Laborde

Tutor de Tesis

**Dr. Ing. Franco Robledo**

Director Académico

**Prof. Ing. Omar Viera**

Tesis de Maestría - Programa de Posgrado PEDECIBA Informática  
Instituto de Computación - Facultad de Ingeniería  
Universidad de la República

3 de febrero de 2021

## Motivación

- Los servicios de fibra al hogar (*Fiber-To-The-Home - FTTH*) tienen una gran penetración mundial brindando a los clientes finales transferencias de datos a altas velocidades.
- La cantidad de aplicaciones y servicios sobre internet ha crecido exponencialmente.
- Es mandatorio escalar de forma inteligente la infraestructura de red.
- Dado que el despliegue de la fibra optica es una importante inversión económica, el diseño topológico de las redes *FTTH* debe continuar considerandose.

## Objetivos I

- Problema de optimización combinatoria motivado por el diseño topológico de redes de comunicaciones con restricciones de confiabilidad.
- El objetivo es interconectar nodos distinguidos, llamados terminales, utilizando un nivel adecuado de redundancia y de forma simultanea, satisfacer las restricciones de confiabilidad.
- En el análisis de confiabilidad nos enfrentamos a fallas aleatorias en los componentes del sistema. Se considera el modelo como realista hostil, donde tanto nodos como enlaces pueden fallar.

## Objetivos II

- Encontrar una solución de costo mínimo que alcance un umbral de confiabilidad, donde tanto los nodos como los enlaces pueden fallar con probabilidades dadas.
- Entender el trade-off entre costo-confiabilidad, y como la confiabilidad aumenta naturalmente agregando niveles de redundancia entre los nodos terminales.
- Entender el impacto en la confiabilidad de las redes solución al aumentar o disminuir las probabilidades de falla elementales tanto en nodos como enlaces.
- Pertenece a la clase de problemas *NP-Hard*.
- Se propone una solución que resuelve el problema de forma apróximada con una metodología *GRASP/VND* para el problema de optimización y para el análisis de la confiabilidad el método *RVR*.

## Definición

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

## Definition (GSPNCHR)

Se considera el grafo no dirigido  $G = (V, E)$ , conjunto de nodos terminales  $T \subseteq V$ , costo de los enlaces  $\{c_{i,j}\}_{(i,j) \in E}$  y requerimientos de conectividad  $R = \{r_{i,j}\}_{i,j \in T}$ . Además, asumimos que tanto los enlaces como los nodos no terminales (Steiner) fallan con probabilidades  $P_E = \{p_e\}_{e \in E}$  y  $P_{V-T} = \{p_v\}_{v \in V-T}$ . Dado el umbral de confiabilidad  $p_{min}$ , el objetivo es construir una topología de costo mínimo  $G_S \subseteq G$  cumpliendo con los requerimientos de conectividad  $R$  y el umbral de confiabilidad:  $R_K(G_S) \geq p_{min}$ , siendo  $K = T$  el conjunto de nodos terminales.

## GSPNCHR

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

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

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

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

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

## GSPNCHR

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

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

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

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

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

## Resultados Teóricos

- Se demuestra la complejidad computacional del problema (*NP-hardness*).
- Se modela el problema para instancias particulares con  $r_{ij} = k, k \geq 2$ , tanto para nodo conectividad como para enlace conectividad.
- Se encuentran cotas inferiores para las formulaciones anteriores aplicando teoría de dualidad y relajación lagrangiana.
- Formulación de programación matemática del problema para las versiones  $k$  nodo-conectividad y  $k$  arista-conectividad aplicando los teoremas de BienStock y Monma.
- Se proponen algoritmos de orden polinomial para la resolución del problema en forma exacta de casos particulares, respecto a conectividad y topología de grafos.



## Trabajos Relacionados

- Se dedicó un capítulo entero donde se expone el relevamiento del estado del arte de la temática de esta tesis: Confiabilidad en redes, Diseño topológico de redes, GRASP, VND, RVR y otras como Monte Carlo Crudo.
- Son escasos los trabajos que abordan conjuntamente la optimización topológica de la red bajo restricciones de confiabilidad.
- GRASP, VND y RVR han sido utilizados con gran éxito en problemas de optimización combinatoria.

## Publicación

- 1 A GRASP/VND Heuristic for the Generalized Steiner Problem with Node-Connectivity Constraints and Hostile Reliability to be published in the Proceedings of the 8th International Conference on Variable Neighborhood Search (ICVNS March 2021). Khalifa University, Abu Dhabi, U.A.E. The article will be published by Springer in the Lecture Notes in Computer Science (LNCS) series.

# GRASP/VND

- Utilizamos una metodología GRASP/VND.
- GRASP y VND son metaheurísticas bien conocidas que se han utilizado con éxito para resolver muchos problemas difíciles de optimización combinatoria.
- GRASP es un poderoso proceso de arranque múltiple que opera en dos fases. Se construye una solución factible en una primera fase, cuyo vecindario luego se explora en la fase de búsqueda local.
- VND explora varias estructuras de vecindad en un orden determinista. Su éxito se basa en el simple hecho de que las diferentes estructuras de vecindad no suelen tener el mismo mínimo local. Así, la solución resultante es simultáneamente una solución localmente óptima en todas las estructuras vecinas.

## Network Design

---

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

---

```
1:  $i \leftarrow 0$ ;  $P \leftarrow \emptyset$ ;  $sol \leftarrow \emptyset$ 
2: while  $i < iter$  do
3:    $\bar{g} \leftarrow Construction(G_B, P, k)$ 
4:    $g_{sol} \leftarrow VND(\bar{g}, P)$ 
5:    $reliability \leftarrow RVR(g_{sol}, P_E, P_{V-T}, simiter)$ 
6:   if  $reliability > p_{min}$  then
7:      $sol \leftarrow sol \cup \{g_{sol}\}$ 
8:   end if
9: end while
10: return  $sol$ 
```

---

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

```

1:  $g_{sol} \leftarrow (S_D^{(l)}, \emptyset)$ ;  $m_{i,j} \leftarrow r_{i,j}$ ;  $P_{i,j} \leftarrow \emptyset, \forall i, j \in S_D^{(l)}$ ;  $A_{i,j} \leftarrow 0, \forall i, j \in S_D^{(l)}$ 
2: while  $\exists m_{i,j} > 0 : A_{i,j} < MAXATTEMPTS$  do
3:    $(i, j) \leftarrow ChooseRandom(S_D^{(l)} : m_{i,j} > 0)$ 
4:    $\bar{G} \leftarrow G_B \setminus P_{i,j}$ 
5:   for all  $(u, v) \in E(\bar{G})$  do
6:      $\bar{c}_{u,v} \leftarrow c_{u,v} \times 1_{\{(u,v) \notin g_{sol}\}}$ 
7:   end for
8:    $L_p \leftarrow KSP(k, i, j, \bar{G}, \bar{C})$ 
9:   if  $L_p = \emptyset$  then
10:     $A_{i,j} \leftarrow A_{i,j} + 1$ ;  $P_{i,j} \leftarrow \emptyset$ ;  $m_{i,j} \leftarrow r_{i,j}$ 
11:   else
12:     $p \leftarrow SelectRandom(L_p)$ ;  $g_{sol} \leftarrow g_{sol} \cup \{p\}$ 
13:     $P_{i,j} \leftarrow P_{i,j} \cup \{p\}$ ;  $m_{i,j} \leftarrow m_{i,j} - 1$ 
14:     $(P, M) \leftarrow GeneralUpdateMatrix(g_{sol}, P, M, p, i, j)$ 
15:   end if
16: end while
17: return ( $g_{sol}, P$ )

```

## VND

El objetivo es combinar una rica diversidad de vecindades para obtener una solución óptima local para cada vecindario. Se considera una implementación clásica de VND, ordenando las respectivas búsquedas locales después de la fase de construcción. Se consideran tres estructuras de vecindad para construir VND.

- 1 SwapKeyPathLocalSearch
- 2 KeyPathLocalSearch
- 3 KeyTreeLocalSearch

## 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  $v$ ).

### Definition (Estructura de Vecindad para Swap Key-Paths)

Dado un key-path  $p \subseteq g_{sol}$ , una solución vecina para  $g_{sol}$  es  $\hat{g}_{sol} = \{g_{sol} \setminus p\} \cup \{m\}$ , siendo  $m$  el conjunto de nodos y enlaces que serán añadidos preservando la factibilidad de  $\hat{g}_{sol}$ .

---

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

---

```
1: improve  $\leftarrow$  TRUE
2: while improve do
3:   improve  $\leftarrow$  FALSE
4:    $K(g_{sol}) \leftarrow \{p_1, \dots, p_h\}$  /* Key-path decomposition of  $g_{sol}$  */
5:   while not improve and  $\exists$  key-paths not analyzed do
6:      $p \leftarrow (K(g_{sol}))$  /* Path not analyzed yet */
7:      $(g_{sol}, \textit{improve}) \leftarrow \text{FindSubstituteKeyPath}(g_{sol}, p, P)$ 
8:   end while
9: end while
10: return  $g_{sol}$ 
```

---



## Definition (Estructura de Vecindad para Key-Paths)

Dado un key-path  $p \in g_{sol}$ , una solución vecina es  $\hat{g}_{sol} = \{g_{sol} \setminus p\} \cup \{\hat{p}\}$ , donde  $\hat{p}$  es otro camino que conecta los extremos desde  $p$ . La vecindad de key-paths desde  $g_{sol}$  esta compuesta por la operación previa a los posibles miembros pertenecientes a  $K_{g_{sol}}$ .

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

```

1: improve  $\leftarrow$  TRUE
2: while improve do
3:   improve  $\leftarrow$  FALSE
4:    $K(g_{sol}) \leftarrow \{p_1, \dots, p_h\}$  /* Key-path decomposition of  $g_{sol}$  */
5:   while not improve and  $\exists$  key-paths not analyzed do
6:      $p \leftarrow (K(g_{sol}))$  /* Path not analyzed yet, with extremes  $u$  and  $v$  */
7:      $\hat{\mu} \leftarrow < \text{NODES}(p) \cup S_D \setminus \text{NODES}(g_{sol}) >$  /* Induced subgraph  $\hat{\mu}$  */
8:      $\hat{p} \leftarrow \text{Dijkstra}(u, v, \hat{\mu})$ 
9:     if  $\text{COST}(\hat{p}) < \text{COST}(p)$  then
10:       $g_{sol} \leftarrow \{g_{sol} \setminus p\} \cup \{\hat{p}\}$ 
11:      improve  $\leftarrow$  TRUE
12:    end if
13:  end while
14: end while
15: return  $g_{sol}$ 

```

## Definition (Estructura de Vecindad para Key-Tree)

Se considera el key-tree  $T_v \in g_{sol}$  con raíz key-node  $v$ . Un vecino de  $g_{sol}$  es  $\hat{g}_{sol} = \{g_{sol} \setminus T_v\} \cup \{T\}$ , siendo  $T$  otro árbol que reemplaza a  $T_v$  con hojas idénticas.

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

```

1: improve  $\leftarrow$  TRUE
2: while improve do
3:   improve  $\leftarrow$  FALSE
4:    $X \leftarrow \text{KeyNodes}(g_{sol})$  /* Key-nodes from  $g_{sol}$  */
5:    $\bar{S} \leftarrow S_D \setminus \text{NODES}(g_{sol})$ 
6:   while not improve and  $\exists$  key-nodes not analyzed do
7:      $v \leftarrow X$  /* Key-node not analyzed yet */
8:      $[g_{sol}, \text{improve}] \leftarrow \text{GeneralRecConnect}(G_B, C, g_{sol}, v, \bar{S})$ 
9:   end while
10: end while
11: return  $g_{sol}$ 

```

## RVR

- Metodo Recursivo de Reducción de Varianza.
- Objetivo reducir la red original en varias redes mas pequeñas de forma recursiva.
- Construir un estimador insesgado de la confiabilidad, con menor varianza que el Monte Carlo crudo.

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

```

1: If  $terminals=1$ , return 0
2: Elseif  $\phi(G, K) = 1$ , return 1.
3: Else
4:    $D := GetKExtendedCut(G)$ 
5:    $Q_D := AllFailedProb(D)$ 
6:    $index := GetRandomItem(D)$ 
7:    $c := D[index]$ 
8:    $remove(G, D, index - 1)$ 
9:    $add(G, c)$ 
10:  return  $Q_D + (1 - Q_D) \times RVR(G)$ 
11: EndIf

```

## Conjunto de Casos de Prueba

- Se realizó un extenso estudio computacional utilizando el algoritmo *NetworkDesign*.
- Adaptamos instancias conocidas de la librería *TSPLIB* añadiendo probabilidades de falla en nodos y enlaces y los requerimientos de conectividad entre nodos.
- No hay casos benchmark para nuestro problema específico.
- Umbral de confiabilidad  $p_{min} = 0,8$ .
- Probabilidades de nodos y aristas  $p_v = 0,99$  y  $p_e = 0,95$  respectivamente.
- Número de iteraciones en *NetworkDesign* por cada instancia  $iter = 100$ , y número de replicaciones para el método *RVR*  $10^4$ .
- Comprender la sensibilidad de la solución a perturbaciones en confiabilidades elementales. Se utilizaron nueve combinaciones para  $p_v, p_e \in \{0,99, 0,97, 0,95\}$ .

## Cuadro: Conjunto de Casos de Prueba

<i>Problem</i>	<i>% T</i>	<i>%Rel</i>	<i>% Req</i>	<i>Iter_ND</i>	<i>Iter_RVR</i>	<i>#</i>
att48	20-35-50	99-95	100-0-0	100-100-100	10 <sup>4</sup>	3
berlin52	20-35-50	99-95	100-0-0	100-100-100	10 <sup>4</sup>	3
brazil58	20-35-50	99-95	100-0-0	100-100-100	10 <sup>4</sup>	3
ch150	20-35-50	99-95	100-0-0	100-100-100	10 <sup>4</sup>	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 <sup>4</sup>	3
gr202	20-35-50	99-95	100-0-0	100-100-100	10 <sup>4</sup>	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 <sup>4</sup>	3
rd400	20-35-50	99-95	100-0-0	50-50-50	10 <sup>4</sup>	3
berlin52(E)	20	99-90	65-25-10	100	10 <sup>4</sup>	1
att48(E)	35	99-90	65-25-10	100	10 <sup>4</sup>	1
brazil58(E)	50	99-90	65-25-10	100	10 <sup>4</sup>	1
rd100(E)	35	99-90	65-25-10	20	NA	1

## Cuadro: GRASP/VND Effectiveness

<i>Problem</i>	<i>% T</i>	<i>%IC</i>	<i>% IVND</i>	<i>CPU(s)</i>	$\bar{R}$	$\bar{Var}$
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

## Resultados IV

# Preguntas Claves

1

- ¿Cuántas redes factibles existen 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 mejor 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

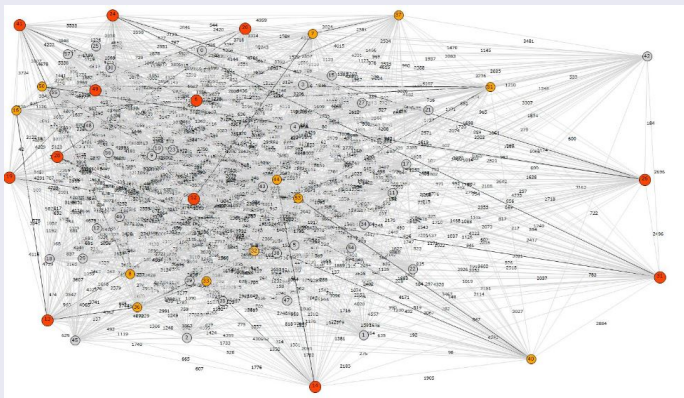
**Cuadro:** Solutions such that  $R \geq 0,98$  (case %99- %97)

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

## Resultados VIII

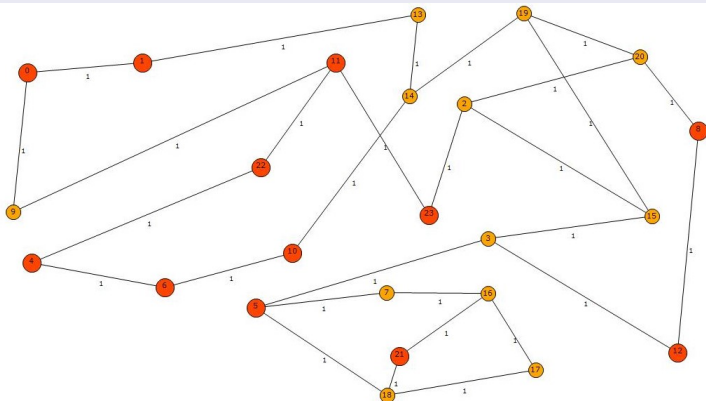
### Brazil58

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



## Brazil58 NetworkDesign Salida

Costo resultado 25106 (32 % de mejora con respecto a la de construcción) y confiabilidad 92 %.



# Conlusiones y Aportes de este Trabajo

- Estudiamos el diseño topológico de redes de alta confiabilidad.
- Nuestro objetivo es combinar aspectos puramente deterministas como la conectividad con modelos probabilísticos provenientes de la confiabilidad de la red.
- *Se introduce The Generalized Steiner Problem with Node-Connectivity Constraints and Hostile Reliability (GSPNCHR).*
- *Probamos formalmente que GSPNCHR pertenece a la clase NP-Hard.*
- *Como consecuencia se propone una metodología GRASP/VND.*

# Conlusiones y Aportes de este Trabajo

- Dado que la evaluación de confiabilidad para el modelo hostil también pertenece a la clase NP-Hard, Adoptamos una excelente método de estimación de confiabilidad puntual, conocida como Recursive Variance Reduction (RVR).
- La mejora proporcionada por la fase VNS después de la fase de Construcción oscila entre el 25,25 % y el 39,84 %.
- La confiabilidad promedio para todas las redes varía entre 82.31 % y 99.87 %.
- *Nuestros resultados resaltan que el modelo es más robusto bajo fallas en los nodos que bajo fallas en los enlaces.*
- *Se logró la publicación: A GRASP/VND Heuristic for the Generalized Steiner Problem with Node-Connectivity Constraints and Hostile Reliability, a publicarse en Volume 12559 of the Lecture Notes in Computer Science series.*

# Líneas de Trabajo Futuro

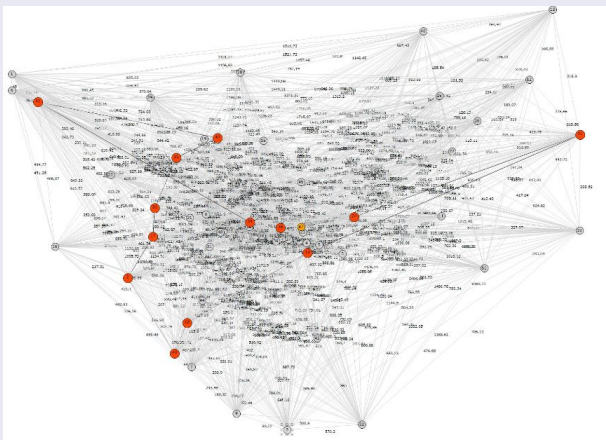
- La interacción entre el diseño de la red topológica y la confiabilidad de la red aún no se comprende bien. Aquí se propusieron algunas búsquedas locales, esencialmente utilizando reemplazos de key-path y key-tree, con el fin de reducir los costos y preservar la factibilidad.
- Una línea de investigación actual es introducir transformaciones que aumenten la confiabilidad.
- El desarrollo de búsquedas locales que aumenten la confiabilidad y reduzcan los costos enriquecería la solución actual.
- Otra posibilidad para el trabajo futuro es enriquecer el número de búsquedas locales y considerar transiciones probabilísticas entre ellas.
- Mejorar en lo posible la algoritmia, apta para computación de alta performance y reducir los tiempos de cómputo.





## 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 %.

