

Continuité de service sur Ethernet Industriel

Sylvain Kubler, Jean-Philippe Georges, Éric Rondeau
`Sylvain.Kubler@cran.uhp-nancy.fr`



Nancy-Université

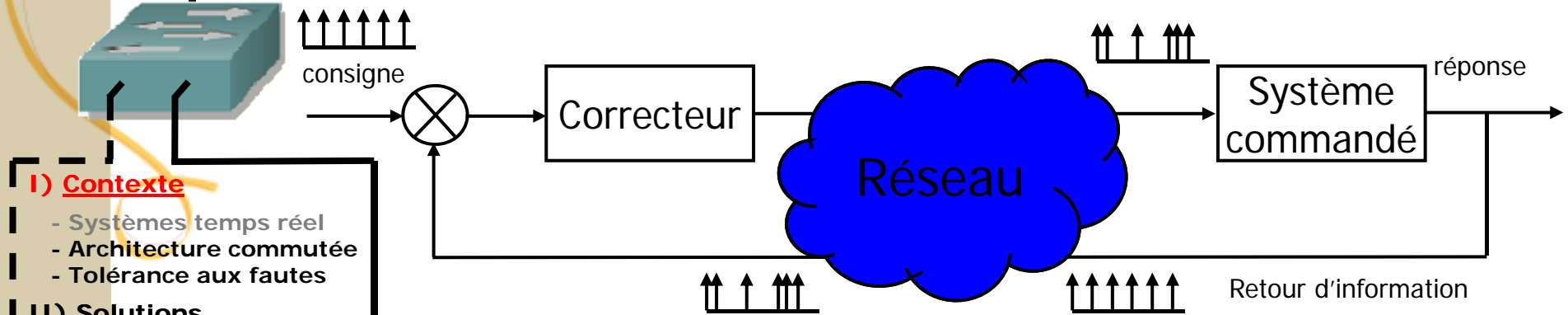


Sommaire

- I) Contexte et positionnement de l'étude**
- II) Solutions**
- III) Formalisation du problème**
- IV) Algorithmes d'optimisation**
- V) Cas d'étude (OPNET)**

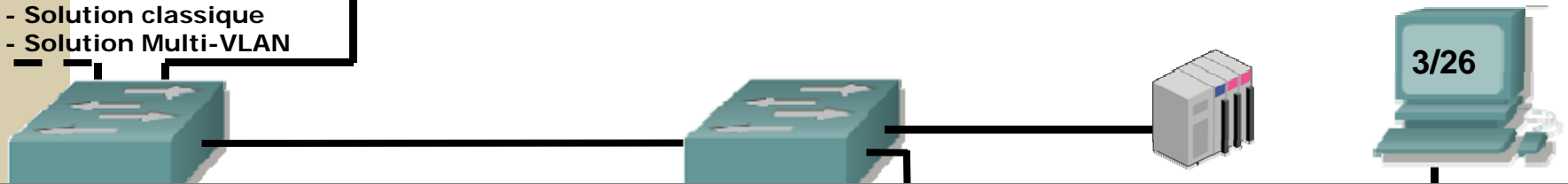
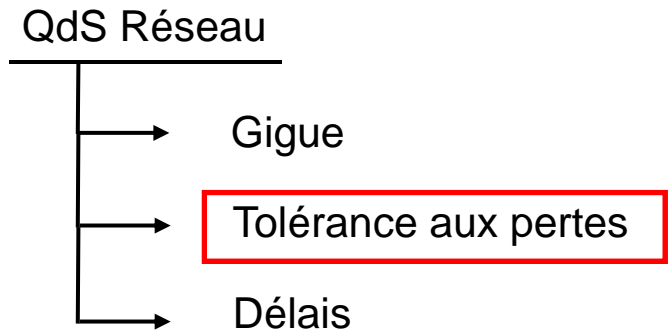
Continuité de service sur Ethernet

Contexte de l'étude : Systèmes Temps réel



Système Contrôlé en Réseau (SCR)

- I) Contexte**
 - Systèmes temps réel
 - Architecture commutée
 - Tolérance aux fautes
- II) Solutions**
 - Idée principale
 - Etat de l'art
 - Solution proposée
 - Comment réaliser ?
- III) Formalisation**
 - Chemins indépendants
 - Chemins non indépendants
- IV) Algorithmes optimisation**
 - Codage éléments réseau
 - Complexité
 - AG
- V) Cas d'étude**
 - Solution classique
 - Solution Multi-VLAN



Continuité de service sur Ethernet

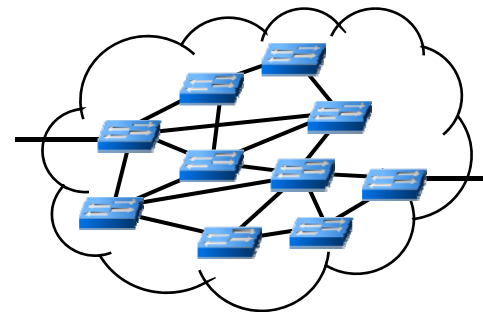
Architecture commutée

Architecture commutée (Wifi, Ethernet)

Full-duplex

Full-segmenté

Conserver les standards "Ethernet"



I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

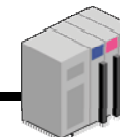
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

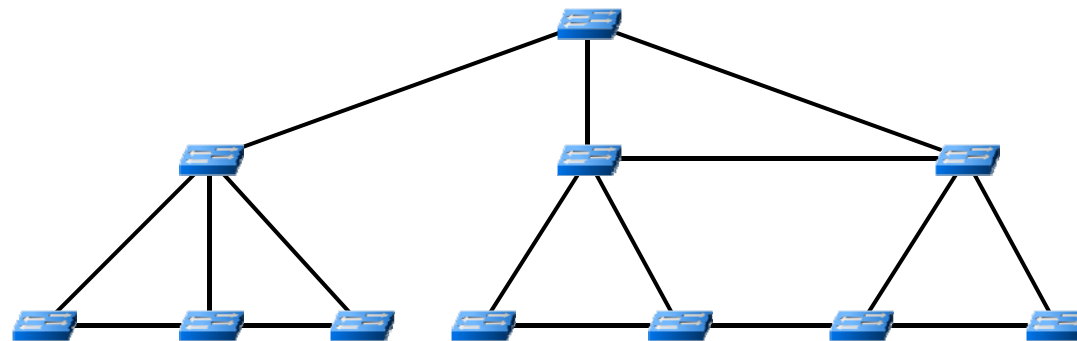


Continuité de service sur Ethernet

Tolérance aux fautes sur "Ethernet "

Protocole Spanning Tree (STP, IEEE 802.1d)

- Supprimer les boucles : Évite le trafic à l'infini



Arbre couvrant

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

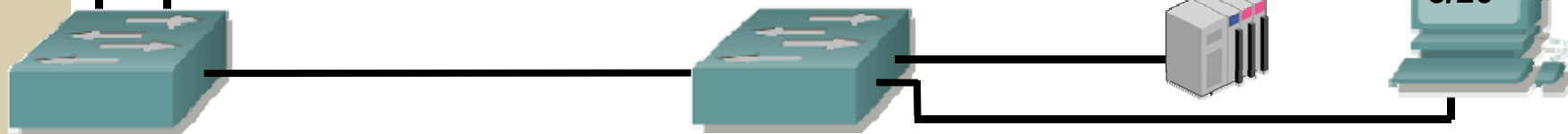
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN



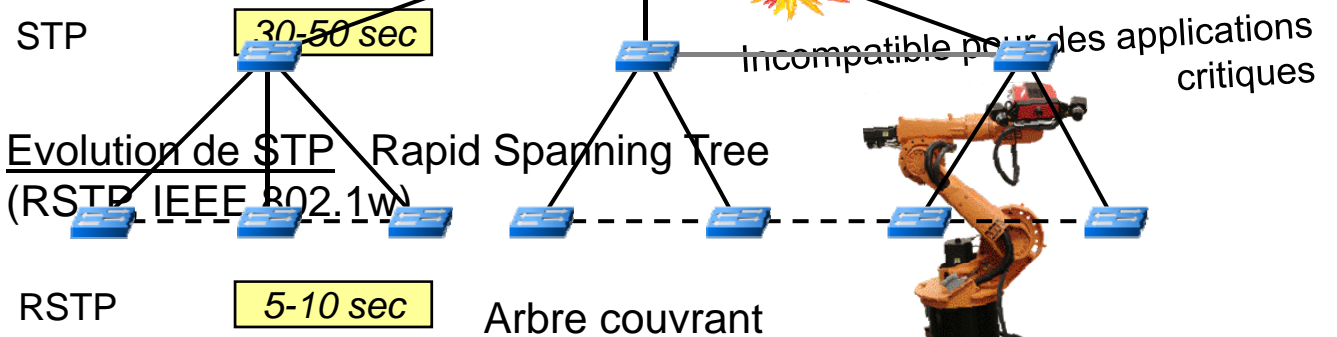
Continuité de service sur Ethernet

Tolérance aux fautes sur "Ethernet "

Protocole Spanning Tree (STP, IEEE 802.1d)

- Surveillance de l'état des équipements
Détecter et Reconfigurer si défaillance d'équipement

Délai de détection + reconfiguration du réseau lors d'une défaillance



I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

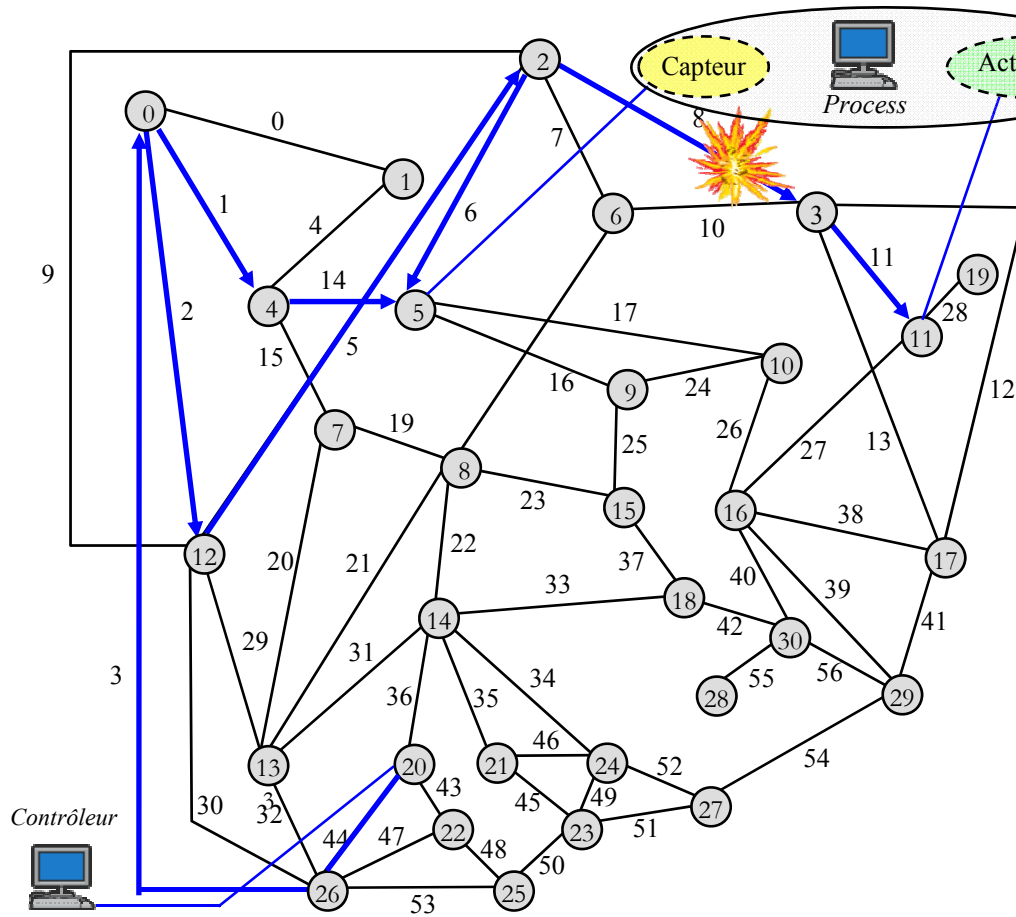
- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN



Etude de cas : plate-forme SCR



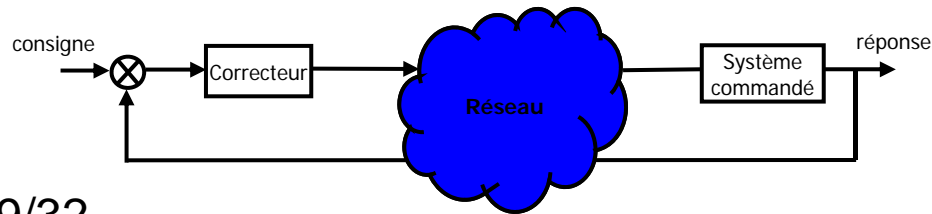
Solution classique (sans redondance de chemin)

Défaillance : lien 8

t = 115 sec

(x) Représente le commutateur x

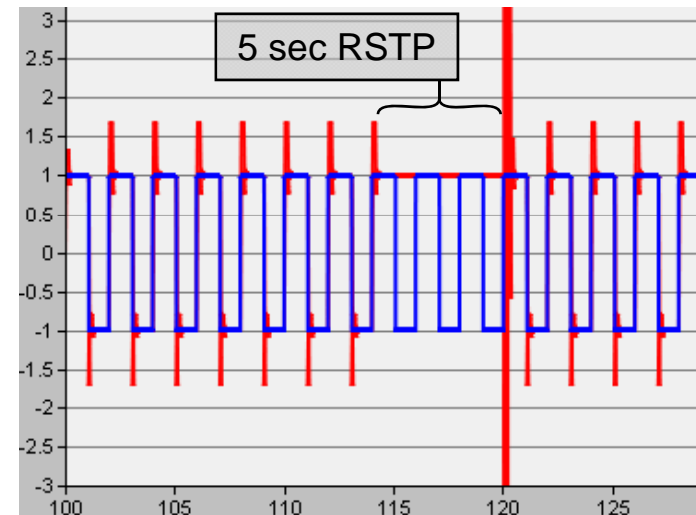
Arbre par défaut de STP



Boucle de régulation

Simulation OPNET

— Référence $U(t)$
— Réponse $Y_c(t)$



Continuité de service sur Ethernet

Solutions

Idée principale

Limiter l'indisponibilité engendrée par la défaillance d'un équipement
(temps détection + reconfiguration)



Conserver les standards "Ethernet"

Sans ajout

Sans modification



I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

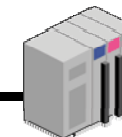
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN



Continuité de service sur Ethernet



Solutions
Etat de l'art

2 approches

Approche hors ligne

- Anticipation d'une défaillance

Limal (2009)

Approche en ligne

- Améliorer les détections et reconfigurations

Jayavelu et al., (2009)

Amélioration des performances de reconfiguration

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- *Etat de l'art*
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

Continuité de service sur Ethernet

Solutions
Etat de l'art

Solution 1 Limal S. (2009)

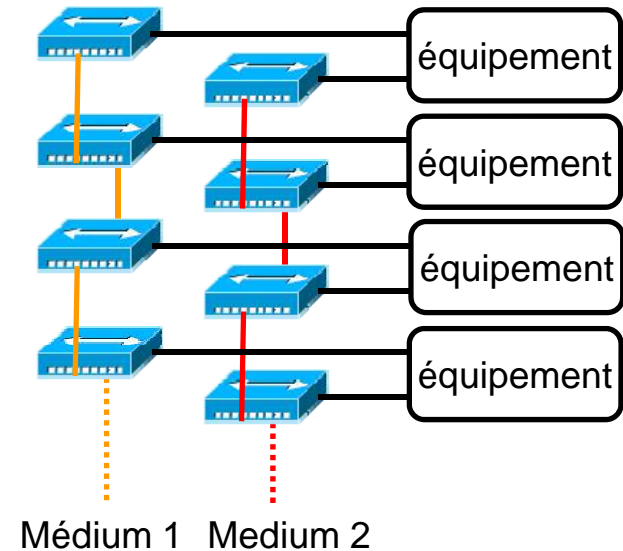
Redondance du médium

Duplication des informations

Particularités :

→ Architecture non commutée

→ Offrant des débits limités par rapport à des architectures commutées



- I) **Contexte**
 - Systèmes temps réel
 - Architecture commutée
 - Tolérance aux fautes
- II) **Solutions**
 - Idée principale
 - Etat de l'art
 - Solution proposée
 - Comment réaliser ?
- III) **Formalisation**
 - Chemins indépendants
 - Chemins non indépendants
- IV) **Algorithmes optimisation**
 - Codage éléments réseau
 - Complexité
 - AG
- V) **Cas d'étude**
 - Solution classique
 - Solution Multi-VLAN

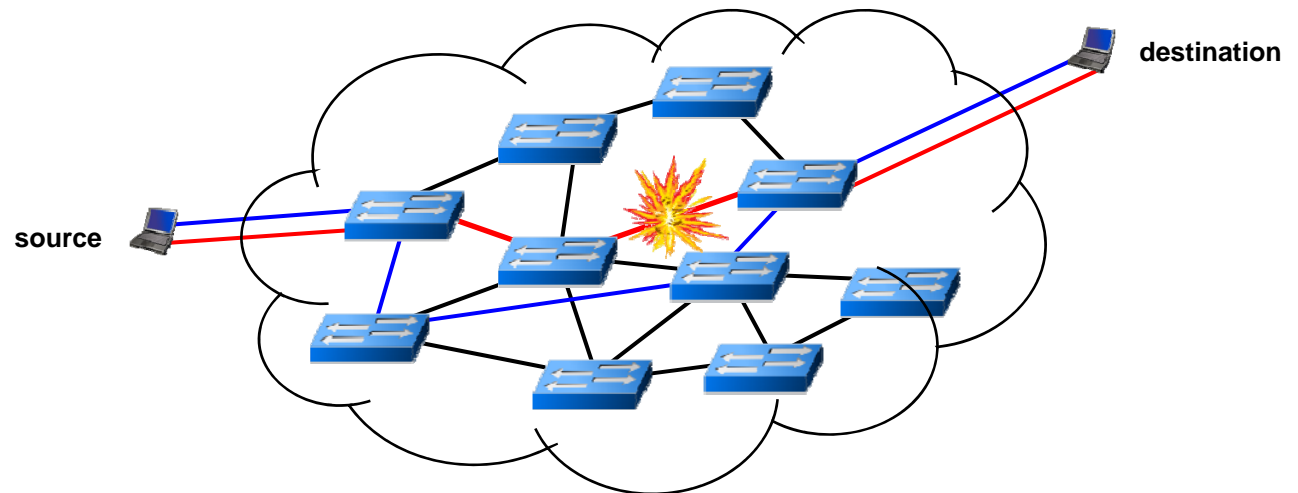
Continuité de service sur Ethernet

Solutions

Solution proposée

- Définir n chemins afin d'acheminer les informations
- Déterminer et Configurer "off-line"
- Méthode passive : Dupliquer les informations sur les n chemins
- Surcoût de Bande Passante / Filtrage en réception (numérotation de paquets)

Exemple avec 1 chemin redondant (soient 2 chemins au total)



I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

Continuité de service sur Ethernet

Solutions

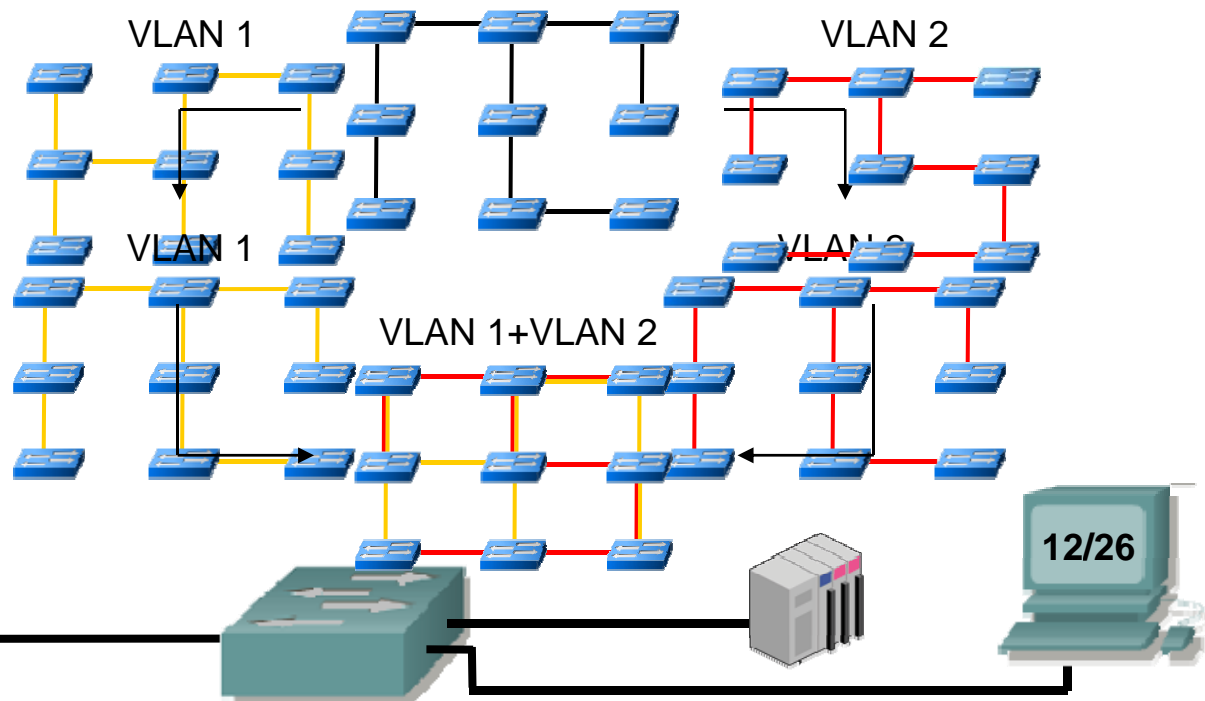
Comment configurer n arbres sur 1 réseau ?

Protocole Concept de réseau virtuel (IEEE 802.1q)

Aucune relation commune entre les protocoles STP et VLAN

VLAN s'appuient sur l'arbre défini par STP

Possible d'affecter un arbre par VLAN → MSTP (IEEE 802.1s)



I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

Continuité de service sur Ethernet

Formalisation du problème

Paire de chemins optimale

Chemins les plus disjoints possibles

Evitant des solutions aberrantes en terme de sauts

⇒ Minimiser la probabilité de défaillance du système global (réseau)

Etablir l'équation générale de la probabilité de défaillance du système globale (réseau)

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

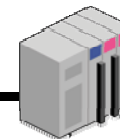
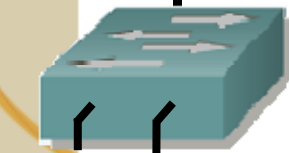
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

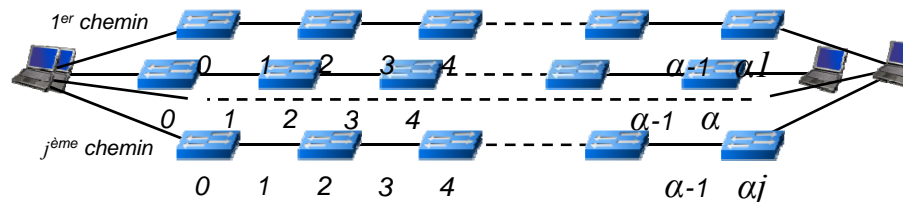
- Solution classique
- Solution Multi-VLAN



Continuité de service sur Ethernet

Formalisation du problème

- Analyse de la fiabilité pour des chemins indépendants



λ : taux de défaillance d'un équipement réseau

α_i : Nombre d'équipements composant le chemin i

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

$$P_{S1} = (1 - \lambda)^\alpha$$

$$P_{D1} = 1 - (1 - \lambda)^\alpha$$

$$P_{Dj} = \prod_{i=1}^j P_{Di}$$

Continuité de service sur Ethernet

Formalisation du problème

Modélisation probabiliste de l'application

Notre évaluation se fait par rapport à des niveaux SIL :
Norme IEC 61508

Mode à faible sollicitation

Mode continu

Toujours assurer la connectivité du système

Mode continu

Niveau SIL	Probabilité d'une défaillance dangereuse par heure
4	$\geq 10^{-9}$ à $< 10^{-8}$
3	$\geq 10^{-8}$ à $< 10^{-7}$
2	$\geq 10^{-7}$ à $< 10^{-6}$
1	$\geq 10^{-6}$ à $< 10^{-5}$

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

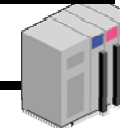
- Chemins indépendants
- Chemins non indépendants

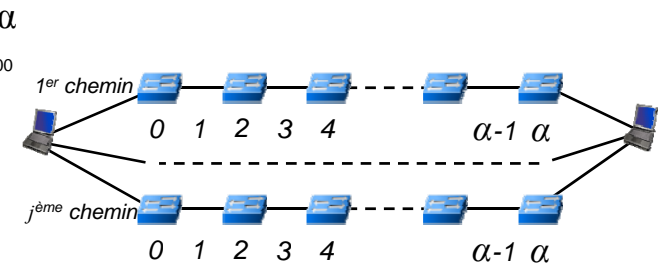
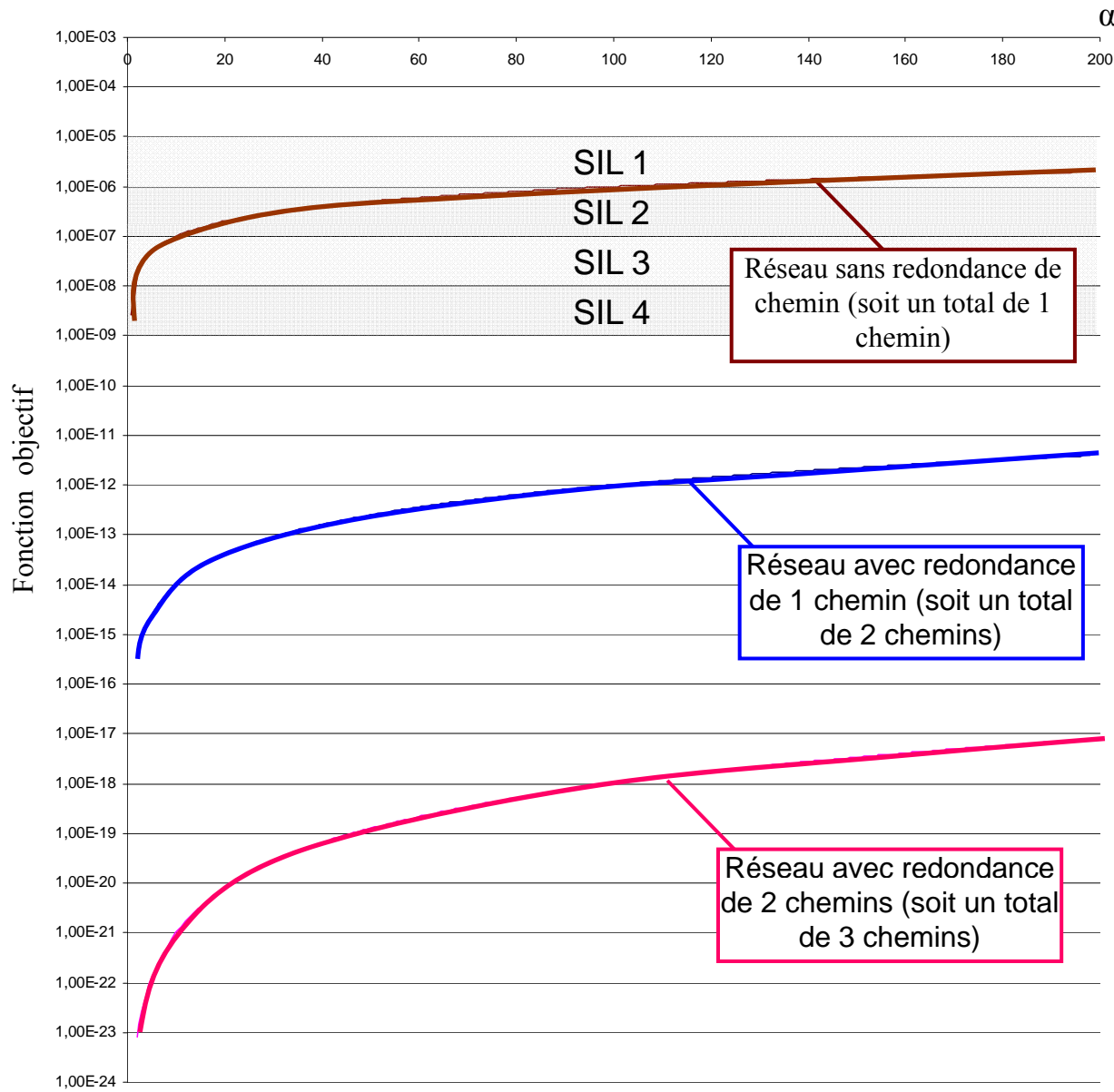
IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN





Deux observations

- La probabilité de fonctionnement du réseau se dégrade rapidement avec l'utilisation d'un chemin unique
- Un chemin redondant (soient 2 chemins au total) est suffisant à maintenir une probabilité inférieure à celles définies par SIL 4

$$P_{Dj} = \prod_{i=1}^j P_{Di}$$

Continuité de service sur Ethernet

Formalisation du problème

- Analyse de la fiabilité pour deux chemins



λ : Taux de défaillance d'un équipement réseau

α_i : Nombre d'équipements composant le chemin i , non communs à l'autre chemin

β : Nombre d'équipements communs aux chemins 1 et 2

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

$$Pd_{\alpha_2} = [1 - (1 - \lambda)^{\alpha_1}] [1 - (1 - \lambda)^{\alpha_2}]$$

$$Ps_{\alpha_2} = 1 - Pd_{\alpha_2}$$

$$Ps_{\beta} = (1 - \lambda)^{\beta}$$

Rappel

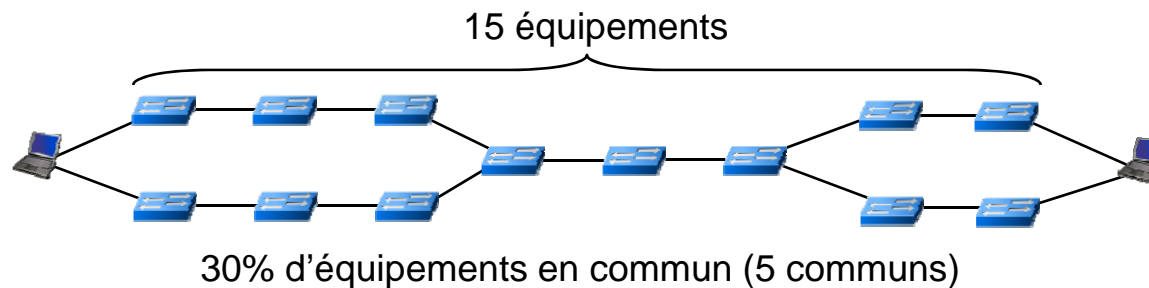
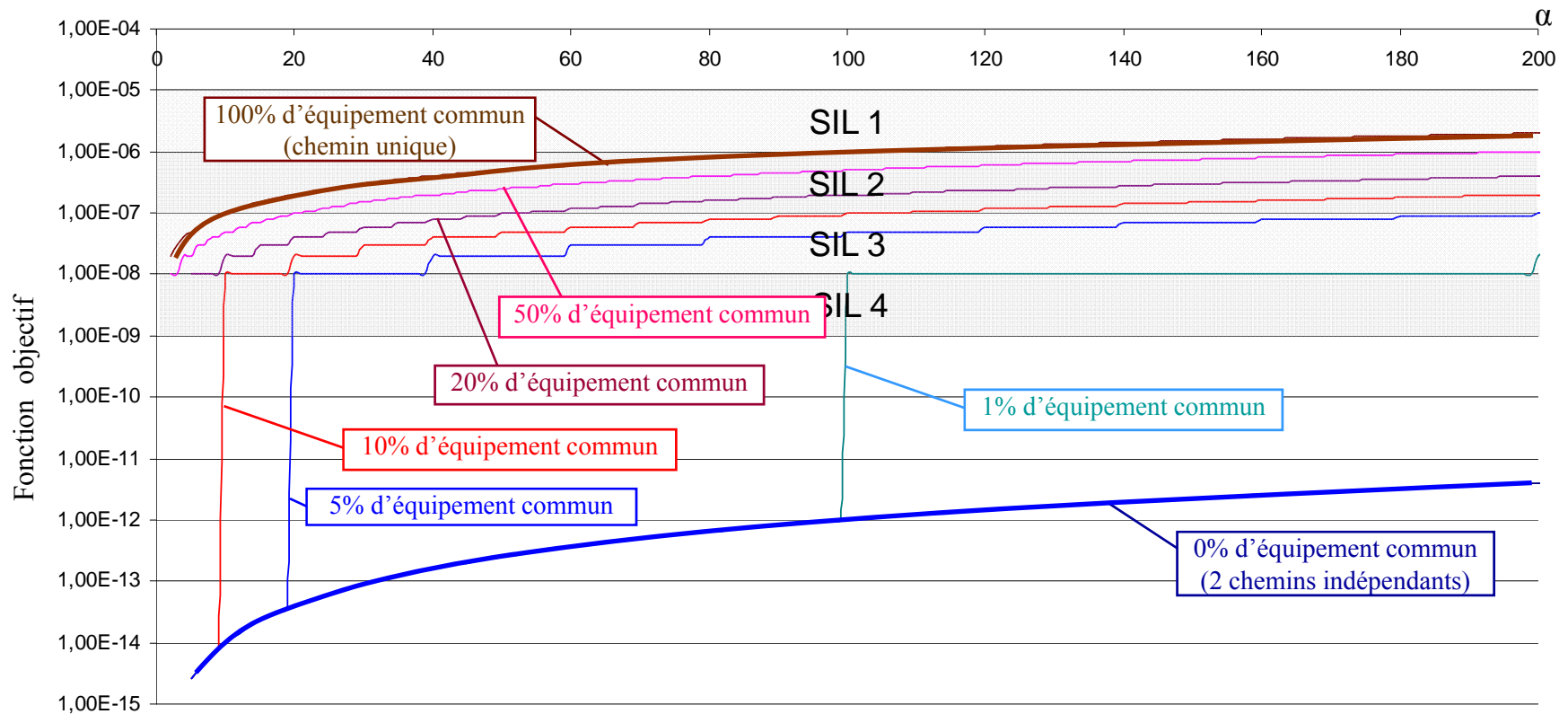
$$Pd_{ind_i} = \prod_{i=1}^j [1 - (1 - \lambda)^{\alpha_i}]$$

$$Ps = Ps_{\beta} \times Ps_{\alpha_2}$$

$$Pd = 1 - Ps$$

← Fonction Objectif

$$P_D = 1 - (1 - P_{D1} \cdot P_{D2})(1 - \lambda)^\beta$$



Observation

➤ Eviter les équipements en commun entre les 2 chemins OU utiliser des composants plus fiables

Continuité de service sur Ethernet

Algorithmes d'optimisation

Codages des éléments du réseau

Codage des graphes

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

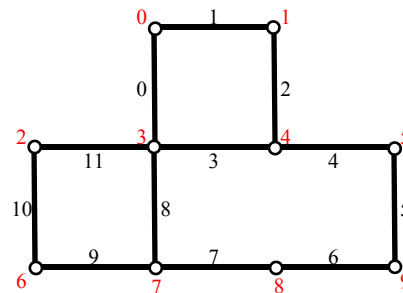
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

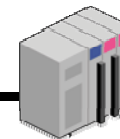
- Solution classique
- Solution Multi-VLAN



Graphe

	0	1	2	3	4	5	6	7	8	9	10	11
0	1	1	0	0	0	0	0	0	0	0	0	0
1	0	1	1	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	1	1
3	1	0	0	1	0	0	0	0	1	0	0	1
4	0	0	1	1	1	0	0	0	0	0	0	0
5	0	0	0	0	1	1	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	1	1	0
7	0	0	0	0	0	0	0	1	1	1	0	0
8	0	0	0	0	0	0	1	1	0	0	0	0
9	0	0	0	0	0	1	1	0	0	0	0	0

matrice d'incidence sommets-arêtes



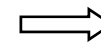
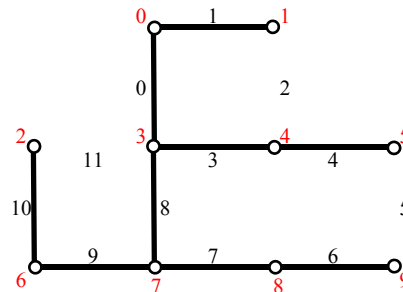
Continuité de service sur Ethernet

Algorithmes d'optimisation

Codages des éléments du réseau

Codage des arbres couvrants

Codage numérique des branches ouvertes
Radha B. (2003)



2 5 11

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

Continuité de service sur Ethernet

Algorithmes d'optimisation

Complexité

Déterminer les solutions optimales d'un problème

Méthodes exhaustives

- Problème d'explosion combinatoire

Méthodes heuristiques



méthodes constructives

- algorithmes glouton
- méthode Pilote

méthodes à recherche locale

- recuit simulé
- recherche Tabou

méthodes évolutionnaires

- algorithmes génétiques
- systèmes de fourmis

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

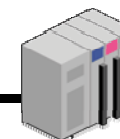
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN



Continuité de service sur Ethernet

Algorithmes d'optimisation

Algorithme génétique : Principe

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

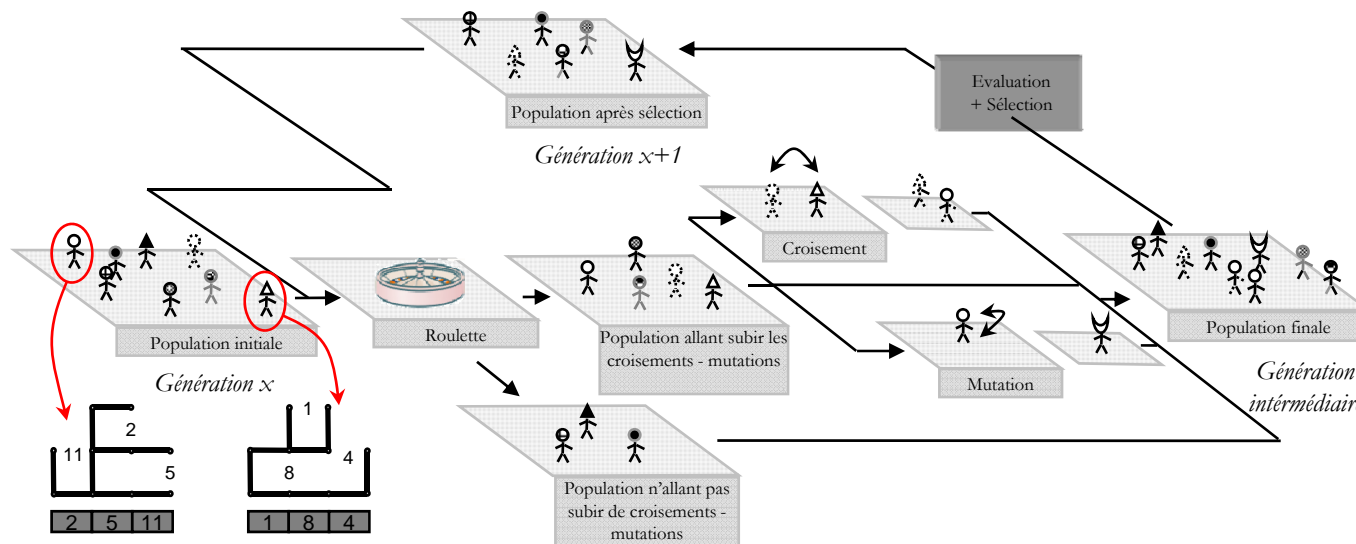
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

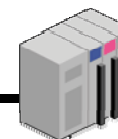
- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN



Principe général de la sélection naturelle



Continuité de service sur Ethernet

Algorithmes d'optimisation

Algorithme génétique : Opérateurs croisements/mutations

Technique de mutation

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

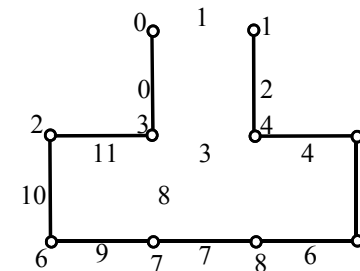
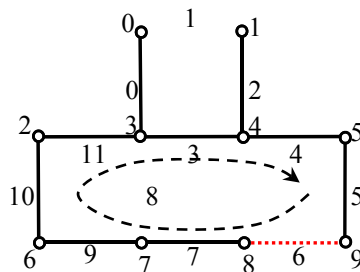
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

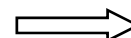
V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

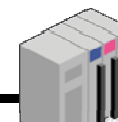


1 8 6

↑
Choix aléatoire



1 8 3



Codage du reseau

branche	Noeud	Noeud
branche 2	0	12
branche 3	0	26
branche 4	1	4
branche 5	2	12
branche 6	2	6
branche 7	2	6
branche 8	2	3
branche 9	2	12
branche 10	3	6
branche 11	3	11
branche 12	3	17
branche 13	3	17
branche 14	4	5
branche 15	4	7
branche 16	5	9
branche 17	5	10
branche 18	6	8
branche 19	7	8
branche 20	7	13
branche 21	8	13
branche 22	8	14
branche 23	8	15
branche 24	9	10
branche 25	9	15
branche 26	10	16
branche 27	11	16
branche 28	11	19
branche 29	12	13
branche 30	12	26
branche 31	13	14
branche 32	13	26
branche 33	14	18
branche 34	14	24
branche 35	14	21
branche 36	14	20
branche 37	15	18
branche 38	16	17
branche 39	16	29
branche 40	16	30
branche 41	17	29
branche 42	18	30
branche 43	20	22
branche 44	20	26
branche 45	21	23

Légende des couleurs

REGLAGE DES PARAMETRES

Noeud Controleur : Probabilité de panne du lien :
Indice compris entre 0 et 1

Noeud Actionneur : Nombre d'Individus initiaux :

Noeud Capteur : Critère d'arrêt :

Nombre d'itérations à réaliser :

Veuillez sélectionner le fichier :

INFORMATION

Les résultats vous sont donnés ci-dessous !!!

RESULTAT DE LA MEILLEURE PAIRE D'ARBRES

Le premier chemin est composé des branches suivantes :

Longueur du 1er chemin : 6 => Voici les branches successives : [5 6 44 30 8 11]

Le deuxième chemin est composé des branches suivantes :

Longueur du 2ème chemin : 7 => Voici les branches successives : [17 26 40 42 33 36 27]

Console

```

amcrtage des chromosomes initiaux
matrice des Individus selectionnes :
0 10

chromosome de l'individu l8 :
Chromosome de l8 : [3 4 9 10 13 14 19 20 21 22 24 25 27 32 36 37 38 39 40 43 45 46 48 49 52 54 56]

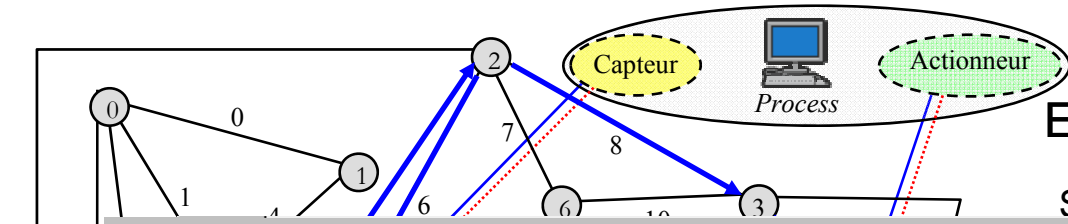
chromosome de l'individu l1_fin12a :
Chromosome de l1_fin12a : [2 4 9 10 12 13 14 15 22 19 21 24 25 39 11 32 49 45 44 23 38 29 47 46 48 53 52]

-----
Meilleure paire : 0 10
resultat de la meilleure paire : 4.218847493575595E-15
Longueur du chemin pour l0 : 6 => [5 6 44 30 8 11]
Longueur du chemin pour l10 : 7 => [17 26 40 42 33 36 27]
    
```

Contr

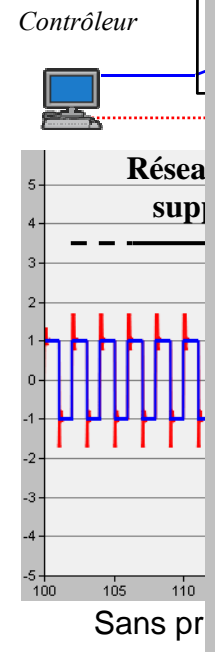
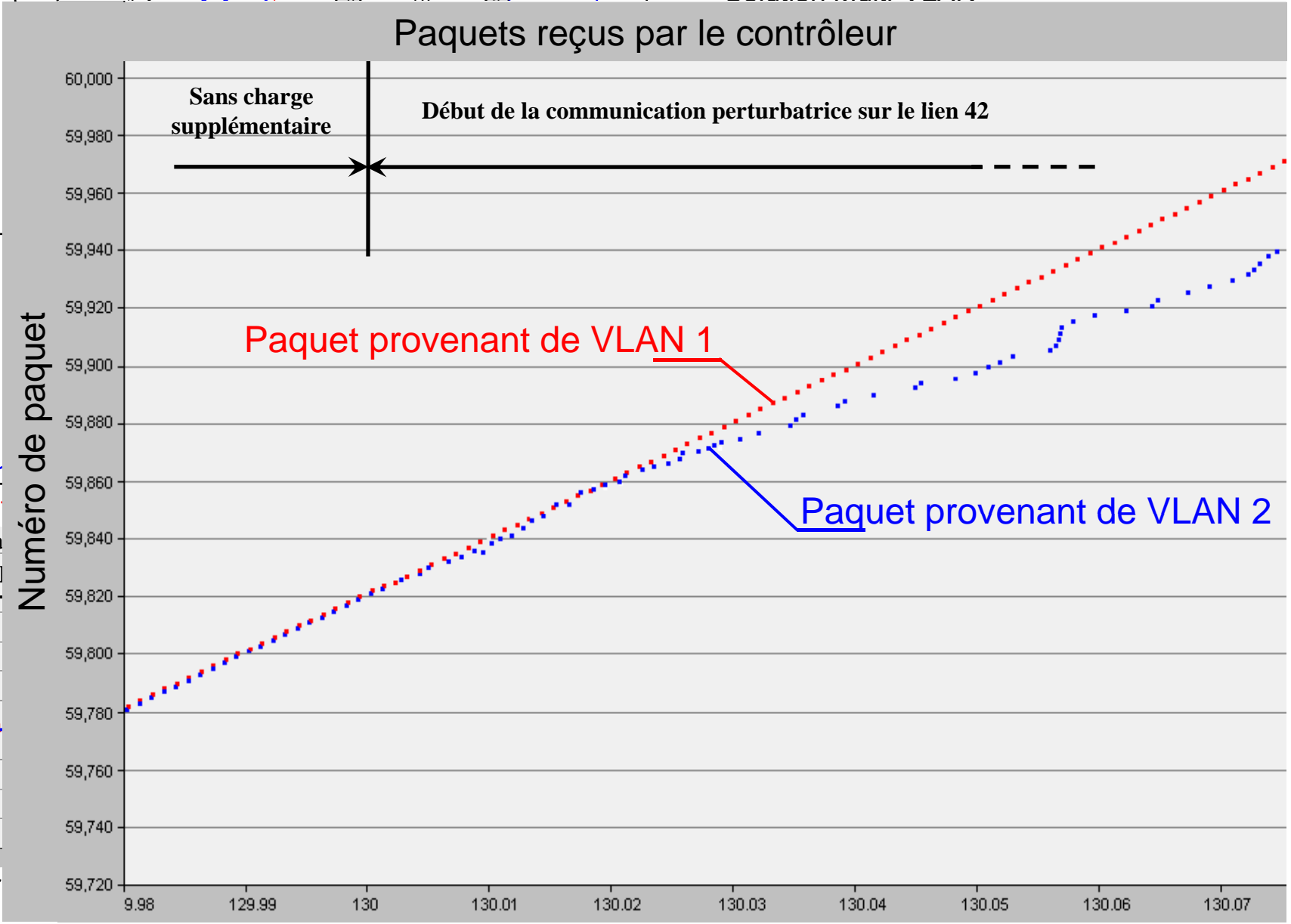
(t)
c(t)

2



Etude de cas : plate-forme SCR

Solution Multi-VLAN



Conclusion

Proposition d'une méthode permettant de réduire la probabilité de défaillance du réseau, tout en conservant les standards Ethernet



Paire d'arbres optimale (Multi-VLAN)



Programme (Algorithmes génétiques)

Perspectives :

Filtrage / Bande Passante

Prise en compte d'autres paramètres (probabilité de dysfonctionnement différente : liens/commutateurs, délais de bout-en-bout, etc.)

Continuité de service sur Ethernet Industriel

Éric Rondeau, Jean-Philippe Georges, Sylvain Kubler
Sylvain.Kubler@cran.uhp-nancy.fr

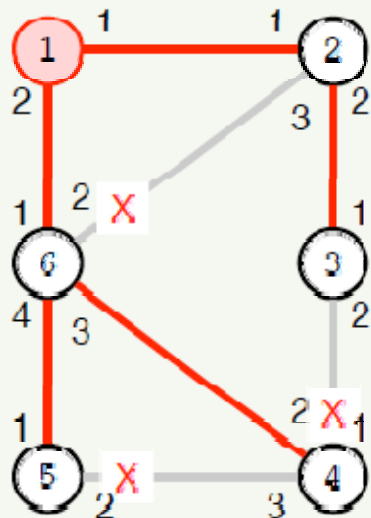


Nancy-Université

Spanning tree (Pohlman, 1992)

- Élection d'un pont racine : n° pont le plus petit
- Sélection du port désigné par port : port avec la distance vers la racine la plus petite et si égalité, n° de port le plus petit
- Sélection du pont désigné par segment : pont avec la distance vers la racine la plus petite et si égalité, n° de pont le plus petit
- Chaque pont maintient la meilleure BPDU reçue et la calculée, et cette dernière est diffusée sur chaque port (BPDU : <n° racine, coût, n° pont, n° port>)

Exemple (Arbre STP et Bellman-Ford?)



À la réception d'une BPDU,

- Si BPDU reçue > configuration calculée, alors pas de changement
- Si BPDU reçue < configuration calculée, alors la meilleure BPDU reçue est mise à jour et la calculée devient <n° pont le plus petit, coût=coût reçu+1, n° pont, x > ; le port par lequel la meilleure BPDU est reçue devient le port désigné et arrête d'émettre sa BPDU
- Si meilleure config. < meilleure BPDU reçue (via le port n) < config. calculée, alors inhiber le port n

Imposer son arbre

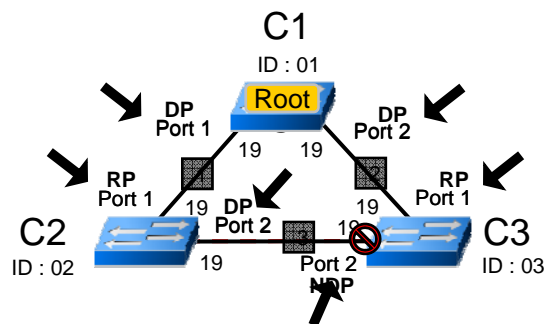
⇒ Agissant sur différents paramètres

Spanning Tree (STP, IEEE 802.1d)

- Supprimer les boucles

Plusieurs phases pour l'élection des liens à inhiber

- 1 - Choix du nœud racine (Root)
- 2 - Choix du port racine sur les nœuds non Root (RP)
- 3 - Choix du port désigné pour tous les segments (DP)



Coût de l'interface = 194196

Délais de Spanning Tree

Protocole Spanning Tree (STP, IEEE 802.1d)

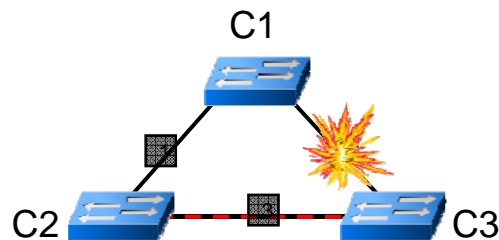
- Surveillance de l'état des équipements

Détecter et Reconfigurer si défaillance d'équipement

3 phases

- 20 sec L → Délai de non réception des BPDU : "Max-Age" (paquets HELLO)
- 15 sec L → Délai d'écoute pour chaque interface susceptible de changer de statut
- 15 sec L → Délai de reconfiguration de l'interface : "Forward Delay"

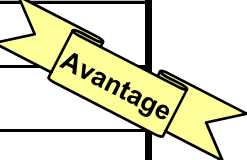
30-50 sec



Comparaison STP - RSTP

Evolution de STP Rapid Spanning Tree (RSTP, IEEE 802.1w)

STP (Statut du port)	RSTP (Statut du port)
Blocking	Discarding
Listening	Discarding
Learning	Learning
Forwarding	Forwarding



Temps de reconvergence (BPDUs "Max-Age")

STP **20-50 sec**

RSTP **6-40 sec**

Continuité de service sur Ethernet

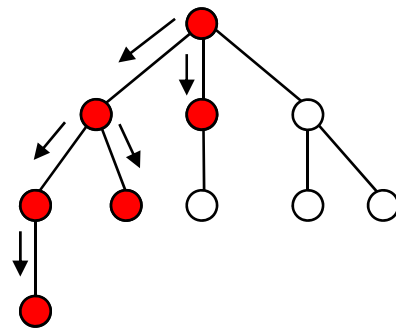
Algorithmes d'optimisation

Codages des éléments du réseau

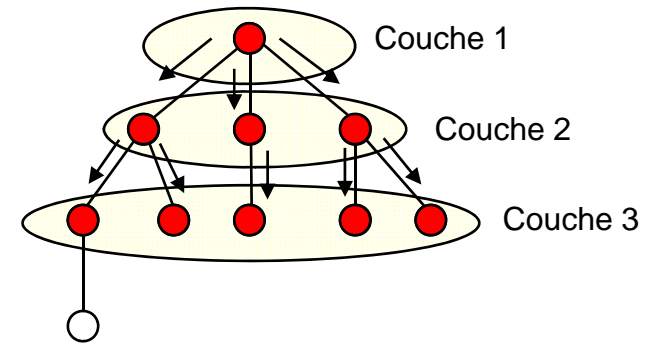
Algorithme de parcours

Parcours en profondeur

Parcours en largeur



Requiert peu de mémoire



Requiert beaucoup de mémoire

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

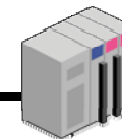
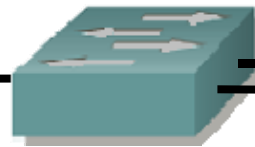
- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN



Continuité de service sur Ethernet

Algorithmes d'optimisation

Codages des éléments du réseau

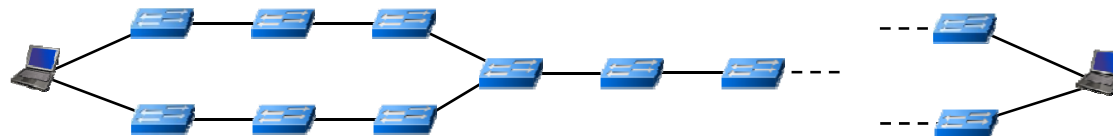
Algorithme de parcours en profondeur

Algorithme général	Le chemin entre deux nœuds
<pre>marque tous les sommets du graphe comme non visités parcours_graphe(graphe, noeud_start) { marque noeud_start comme visité ; pour tous les voisins Vi de noeud_start si Vi non visité parcours_graphe (graphe, Vi) fin si fin pour }</pre>	<pre>marque tous les sommets du graphe comme non visités chemin= { } recherche_chemin (graphe, noeud_start, noeud_fin) { marque noeud_start comme visité ; si noeud_start ≠ noeud_fin mettre à jour le chemin pour tous les voisins Vi de noeud_start si Vi non visité noeud_start = Vi recherche_chemin(graphe, noeud_start, noeud_fin) fin si fin pour fin si }</pre>

Continuité de service sur Ethernet

Formalisation du problème

- Analyse de la fiabilité pour deux chemins



λ : Taux de défaillance d'un équipement réseau

α_i : Nombre d'équipements composant le chemin i , non communs à l'autre chemin

β : Nombre d'équipements communs aux chemins 1 et 2

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

$$Pd_{\alpha_2} = [1 - (1 - \lambda)^{\alpha_1}] [1 - (1 - \lambda)^{\alpha_2}]$$

$$Ps_{\alpha_2} = 1 - [1 - (1 - \lambda)^{\alpha_1}] [1 - (1 - \lambda)^{\alpha_2}]$$

$$Ps_{\beta} = (1 - \lambda)^{\beta} \quad Pd = 1 - Ps$$

Rappel

$$Pd_{ind_i} = \prod_{i=1}^j [1 - (1 - \lambda)^{\alpha_i}]$$

Fonction Objectif

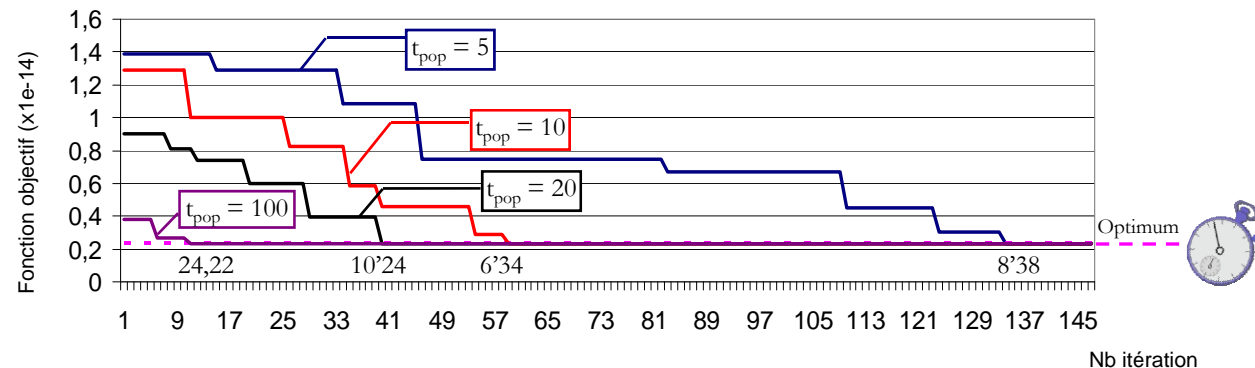
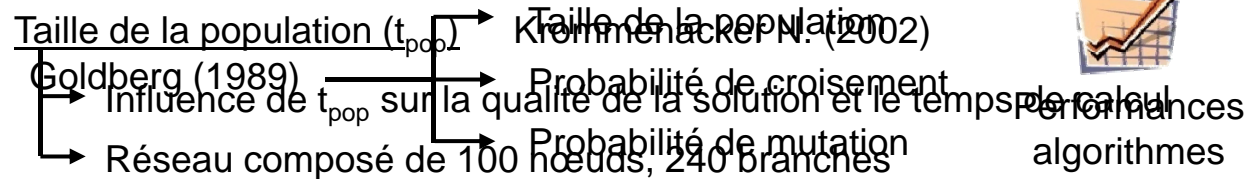
$$Ps = Ps_{\beta} * Ps_{\alpha_2}$$

$$Pd = 1 - [1 - [1 - (1 - \lambda)^{\alpha_1}] (1 - \lambda)^{\alpha_1} (1 - (1 - \lambda)^{\alpha_2})]^{\beta} [1 - \lambda]^{\beta}$$

Continuité de service sur Ethernet

Algorithmes d'optimisation

Algorithme génétique : Ajustement des paramètres



- $t_{pop} = 100$ Très peu de générations - temps prohibitif
- $t_{pop} = 20$ Temps de calcul divisé par 2½
- $t_{pop} = 5$ Converge avec 3 fois plus de générations - temps réduit par 3
- $t_{pop} = 10$ meilleur compromis : qualité de la solution - temps de calcul

36/26

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

Krommenacker, N. (2002). *Heuristiques de conception de topologies réseaux : application aux réseaux locaux industriels*. Thèse, Nancy.

Goldberg, D. (1989). *Genetic Algorithm in Search Optimization*. Addison Wesley

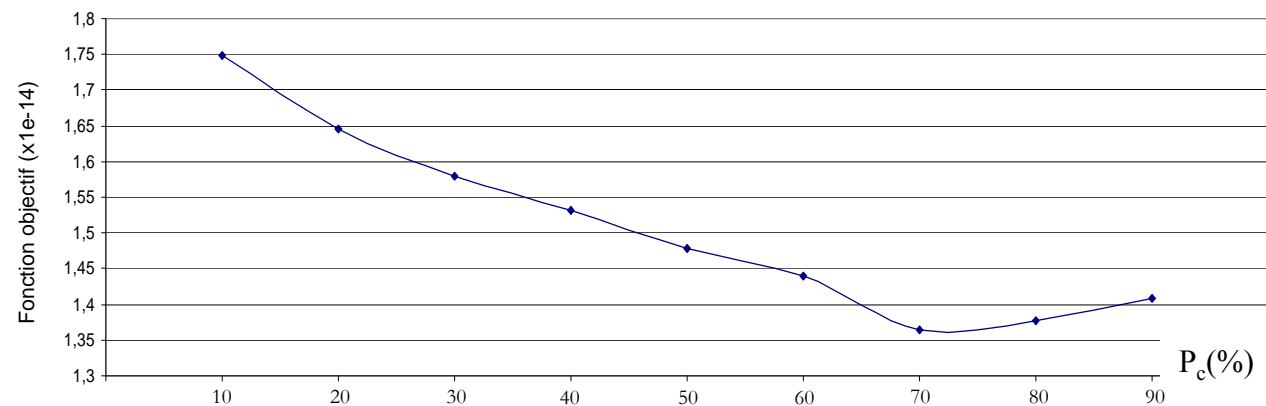
Continuité de service sur Ethernet

Algorithmes d'optimisation

Algorithme génétique : Ajustement des paramètres

Probabilité de croisement (P_c)

- Influence de P_c avec $t_{pop}=10$ et $P_m=40\%$ sur 150 générations
- Réseau composé de 100 nœuds, 240 branches
- Courbe de tendance obtenue à partir des valeurs moyennes sur les 100 graphes



$P_c=70\%$ permet d'obtenir la meilleure solution

37/26

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

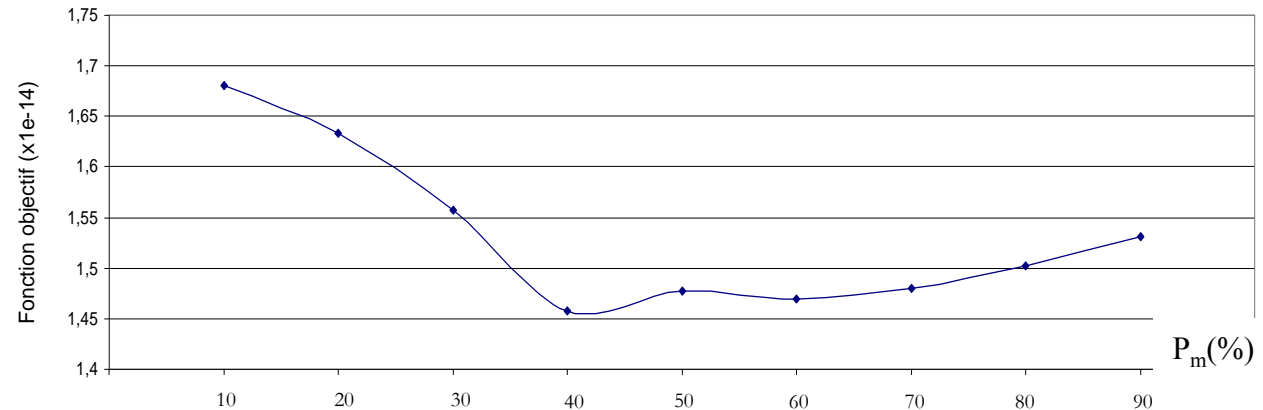
Continuité de service sur Ethernet

Algorithmes d'optimisation

Algorithme génétique : Ajustement des paramètres

Probabilité de mutation (P_m)

- Influence de P_m avec $t_{pop}=10$ et $P_c=70\%$ sur 150 générations
- Réseau composé de 100 nœuds, 240 branches
- Courbe de tendance obtenue à partir des valeurs moyennes sur les 100 graphes



$P_m=40\%$ permet d'obtenir la meilleure solution

38/26

I) Contexte

- Systèmes temps réel
- Architecture commutée
- Tolérance aux fautes

II) Solutions

- Idée principale
- Etat de l'art
- Solution proposée
- Comment réaliser ?

III) Formalisation

- Chemins indépendants
- Chemins non indépendants

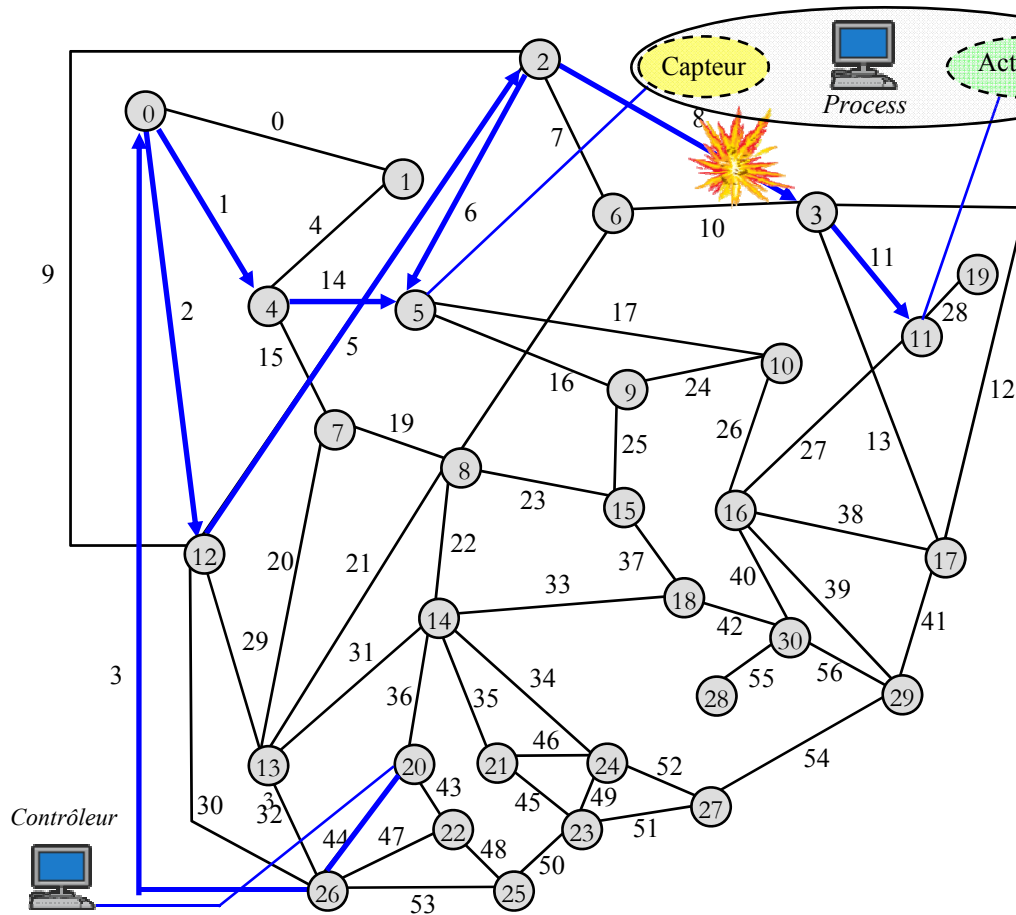
IV) Algorithmes optimisation

- Codage éléments réseau
- Complexité
- AG

V) Cas d'étude

- Solution classique
- Solution Multi-VLAN

Etude de cas : plate-forme SCR



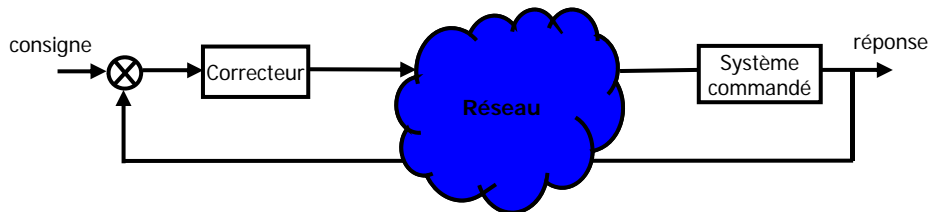
Solution classique (sans redondance de chemin)

Défaillance : lien 8

t = 115 sec

(x) Représente le commutateur x

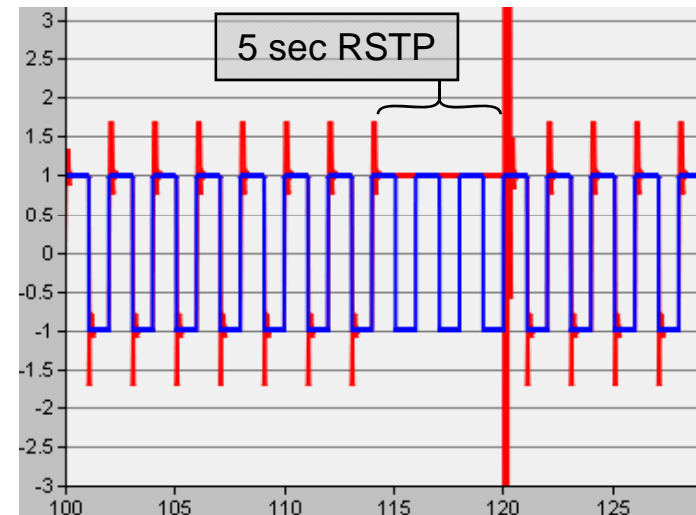
Arbre par défaut de STP



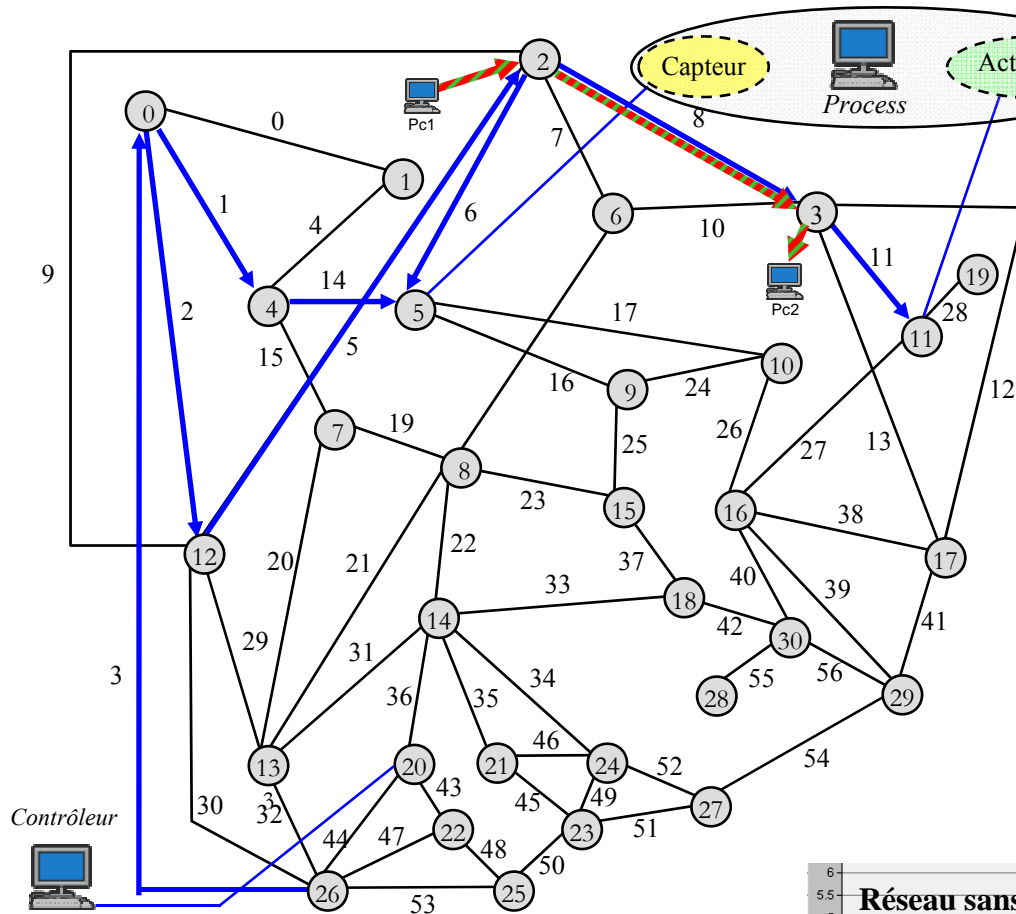
Boucle de régulation

Simulation OPNET

— Référence $U(t)$
— Réponse $Y_c(t)$



Etude de cas : plate-forme SCR



Solution classique (sans redondance de chemin)

Communication perturbatrice : lien 8

t = 115 sec

Délais avant charge du réseau

- Délais contrôleur-actionneur : 0,76 ms

Délais après charge du réseau

- Délais contrôleur-actionneur : 0,28 à 21 ms

(x) Représente le commutateur x

Arbre par défaut de STP

— Référence $U(t)$

— Réponse $Y_c(t)$

