YaDiff

Encore un outil de ~~diff de binaires~~ propagation de symboles

B. Amiaux, J. Bouetard, V. Comiti, F.Grelot, E.Renault, M. Tourneboeuf

Direction Générale de l'Armement / Maîtrise de l'Information

14 juin 2018

DGA

Introduction

- **Analyser des binaires complexes**
  - Systèmes embarqués, binaires lourds
  - Temps d'exécution raisonnable

DGA

Introduction

- **Analyser des binaires complexes**
  - Systèmes embarqués, binaires lourds
  - Temps d'exécution raisonnable
- **Propager le maximum d'information**
  - Version X.Y → W.Z
  - Linux → Windows
  - Lib → Exécutable

DGA

Introduction

- **Analyser des binaires complexes**
  - Systèmes embarqués, binaires lourds
  - Temps d'exécution raisonnable
- **Propager le maximum d'information**
  - Version X.Y → W.Z
  - Linux → Windows
  - Lib → Exécutable
- **Supporter plusieurs architectures**
  - ARM, x86, autre?
  - Options de compilation (optimisations, deboggage...)

DGA

État de l'art

Logiciel	Entrée	License	Plugiciel IDA	Méthode	Autres Limitations
BinDiff	ASM	Privative	Oui	Signature des fonctions, basics blocs, séquences	
Diaphora	ASM	Libre	Oui	Signature des fonctions	Lent
Gorille	CFG	Privative	Non	Isomorphisme de sous-arbres	Compare seulement le CFG
SMIT	ASM	Privative	Non	Graphe d'appel	Ignore les fonctions isolées
TurboDiff	ASM	Libre	Oui	Signature des fonctions	Pas beaucoup d'heuristiques Incompatible avec Linux
PatchDiff	ASM	Libre	Oui	Signature des fonctions	
DarunGrim	ASM	Libre	Oui	Signature des fonctions	Pas facile à installer
BitShred	Octets	Privative	Non	N-gram	Pas cross-archi
Fcatalog	Octets	Libre	Oui	N-gram	Pas cross-archi

DGA

État de l'art

Logiciel	Entrée	License	Plugiciel IDA	Méthode	Autres Limitations
BinDiff	ASM	Privative	Oui	Signature des fonctions, basics blocs, séquences	
Diaphora	ASM	Libre	Oui	Signature des fonctions	Lent
Gorille	CFG	Privative	Non	Isomorphisme de sous-arbres	Compare seulement le CFG
SMIT	ASM	Privative	Non	Graphe d'appel	Ignore les fonctions isolées
TurboDiff	ASM	Libre	Oui	Signature des fonctions	Pas beaucoup d'heuristiques Incompatible avec Linux
PatchDiff	ASM	Libre	Oui	Signature des fonctions	
DarunGrim	ASM	Libre	Oui	Signature des fonctions	Pas facile à installer
BitShred	Octets	Privative	Non	N-gram	Pas cross-archi
Fcatalog	Octets	Libre	Oui	N-gram	Pas cross-archi

- Pas de support des binaires lourds
- Pas de traitement de la sortie (autre qu'affichage)

DGA

État de l'art

Logiciel	Entrée	License	Plugiciel IDA	Méthode	Autres Limitations
BinDiff	ASM	Privative	Oui	Signature des fonctions, basics blocs, séquences	
Diaphora	ASM	Libre	Oui	Signature des fonctions	Lent
Gorille	CFG	Privative	Non	Isomorphisme de sous-arbres	Compare seulement le CFG
SMIT	ASM	Privative	Non	Graphe d'appel	Ignore les fonctions isolées
TurboDiff	ASM	Libre	Oui	Signature des fonctions	Pas beaucoup d'heuristiques Incompatible avec Linux
PatchDiff	ASM	Libre	Oui	Signature des fonctions	
DarunGrim	ASM	Libre	Oui	Signature des fonctions	Pas facile à installer
BitShred	Octets	Privative	Non	N-gram	Pas cross-archi
Fcatalog	Octets	Libre	Oui	N-gram	Pas cross-archi

- Pas de support des binaires lourds
- Pas de traitement de la sortie (autre qu'affichage)

Notre solution : YaDiff

DGA

Contenu

Orchestrateur d'algorithmes : YaDiff "legacy"

- Enchainement des algorithmes

Algorithme d'Intelligence Artificielle

- Vecteur de caractéristiques
- Apprentissage supervisé

DGA

Contenu

DGA

Définitions

- Objet
  - Concept introduit dans notre chaine d'outils
  - Abstrait le type d'un élément (fonctions, blocs basiques, données)
- Bloc basique
  - Séquence d'instructions
- Fonction
  - Fonctionnalité
  - Implémentation
- Référence croisée
  - Lien logique entre les objets

DGA

Signatures

- Associer un maximum d'objets de manière fiable
- Tolérant aux changements mineurs
- Temps d'exécution rapide

DGA

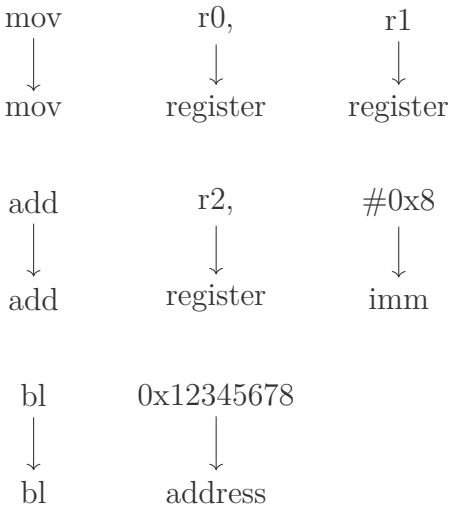
Génération des signatures (Octets invariants)

```
mov     r0,     r1

add     r2,     #0x8

bl      0x12345678
```

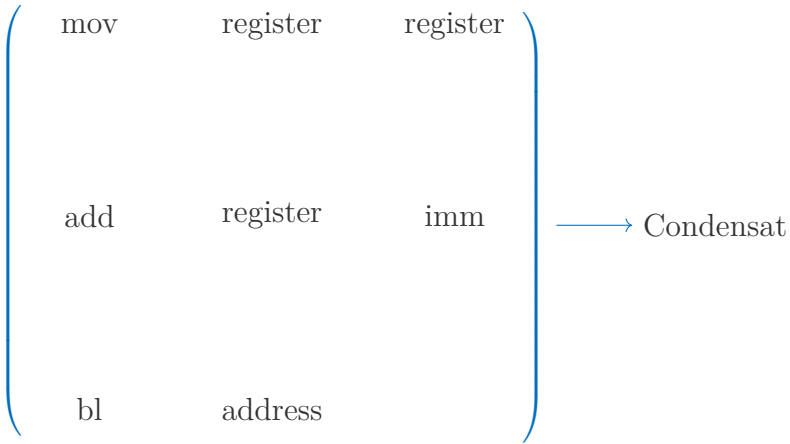
Génération des signatures (Octets invariants)



Génération des signatures (Octets invariants)

mov	register	register
add	register	imm
bl	address	

Génération des signatures (Octets invariants)



Algorithmes

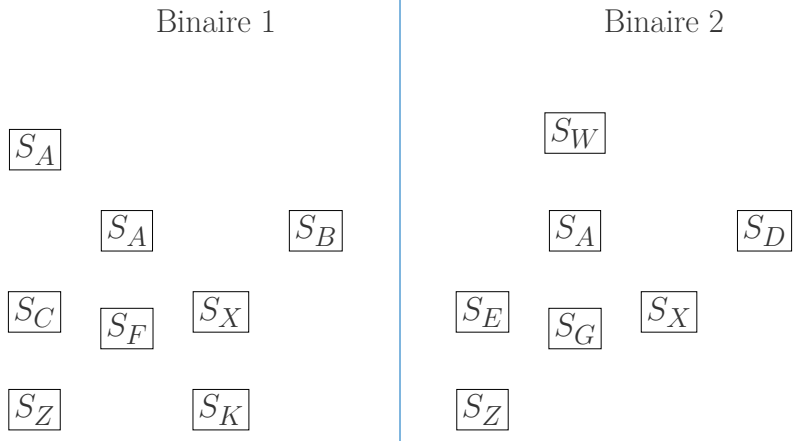
- Algo 1 : Association initiale

Algorithmes

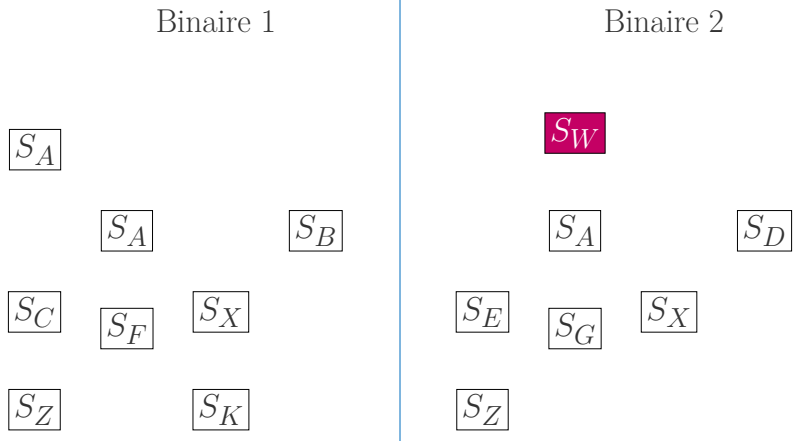
- Algo 1 : Association initiale
- Algo 2 : Propagation par références croisées montantes
- Algo 3 : Propagation par références croisées descendantes

DGA

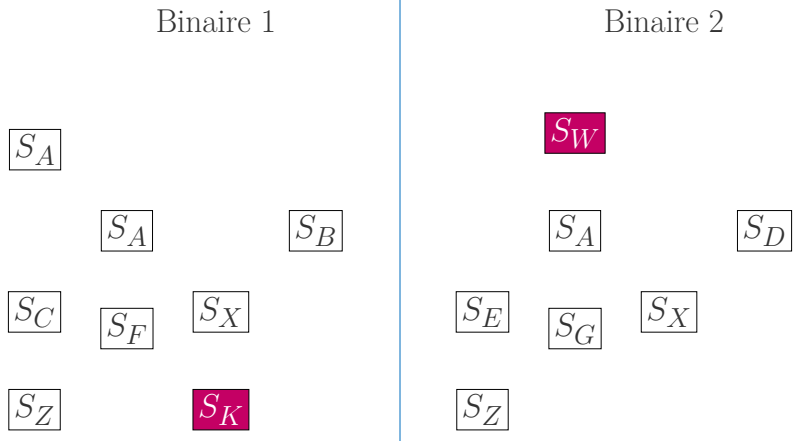
Algo 1 : Association initiale



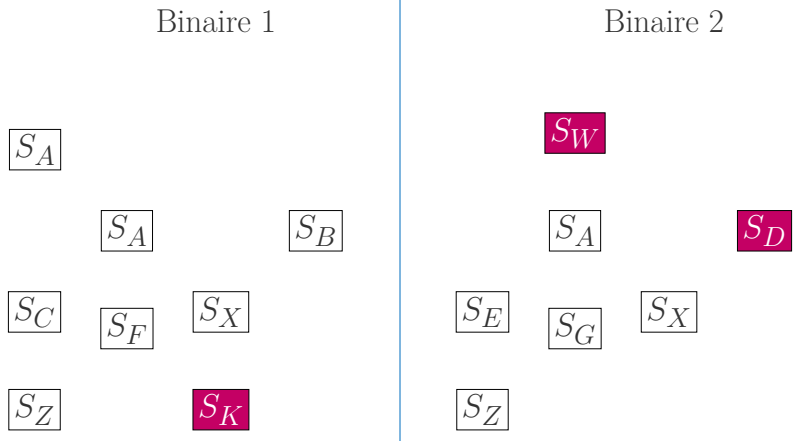
Algo 1 : Association initiale



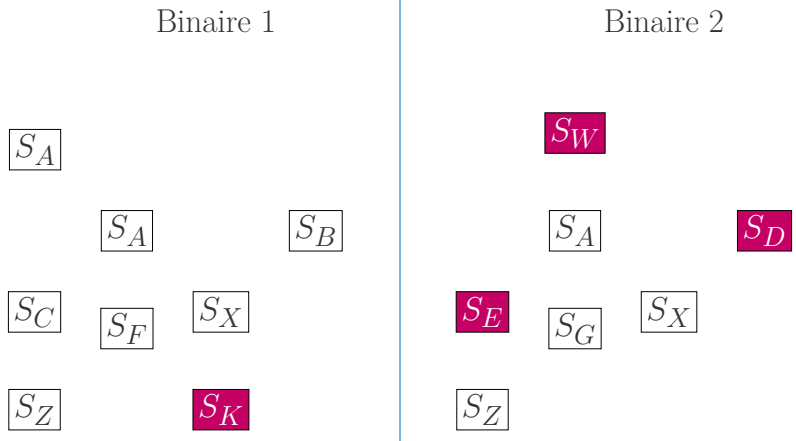
Algo 1 : Association initiale



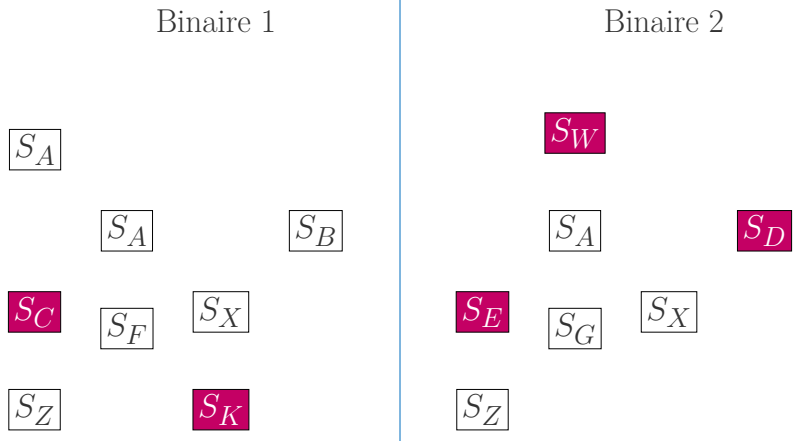
Algo 1 : Association initiale



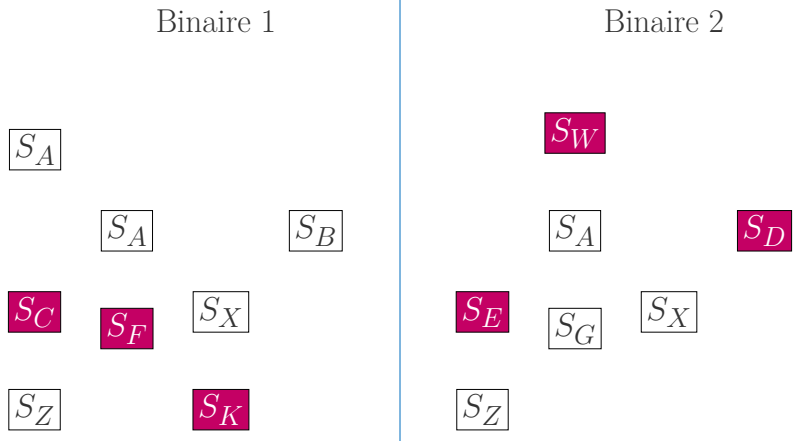
Algo 1 : Association initiale



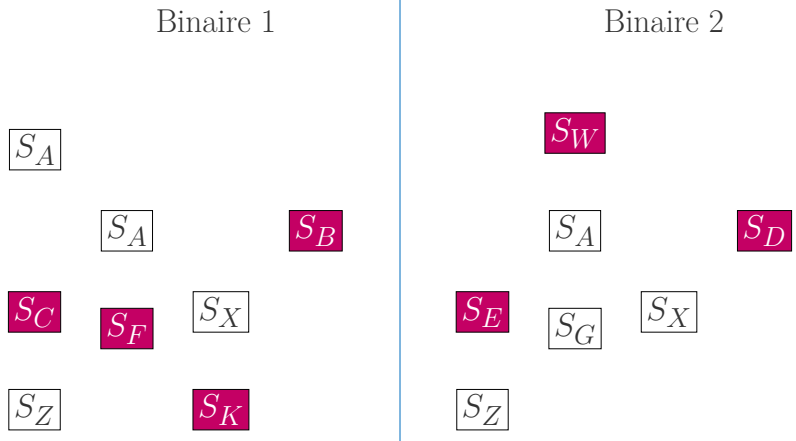
Algo 1 : Association initiale



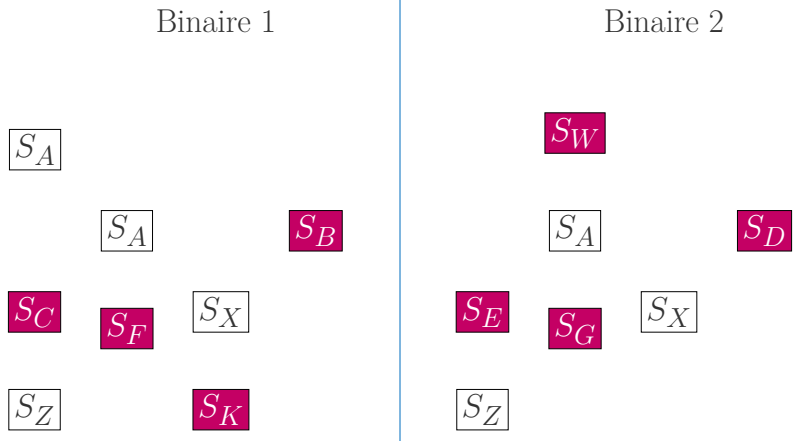
Algo 1 : Association initiale



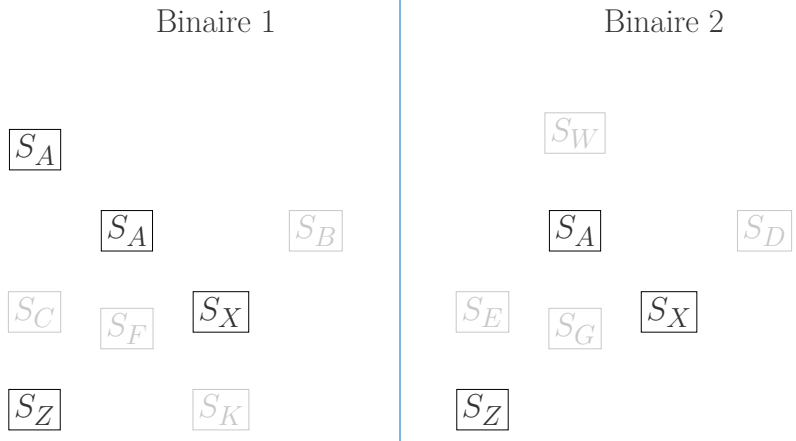
Algo 1 : Association initiale



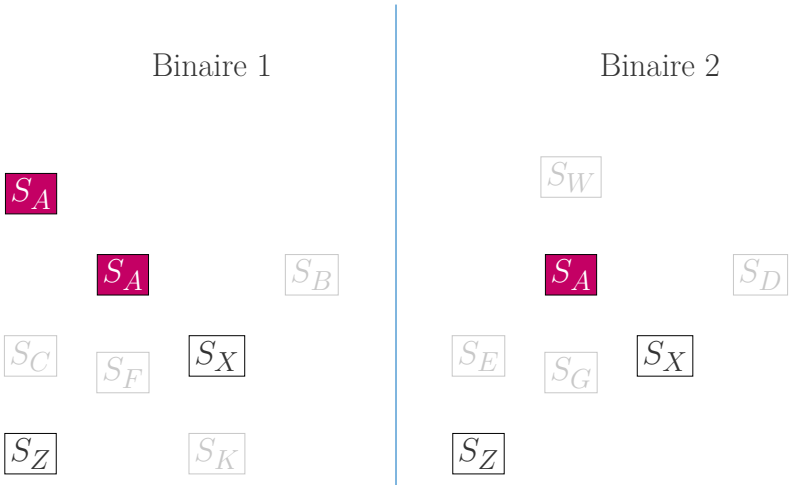
Algo 1 : Association initiale



Algo 1 : Association initiale

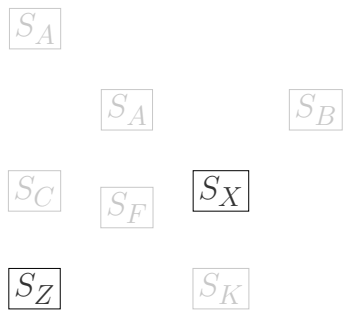


Algo 1 : Association initiale

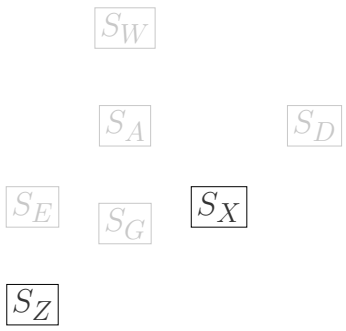


Algo 1 : Association initiale

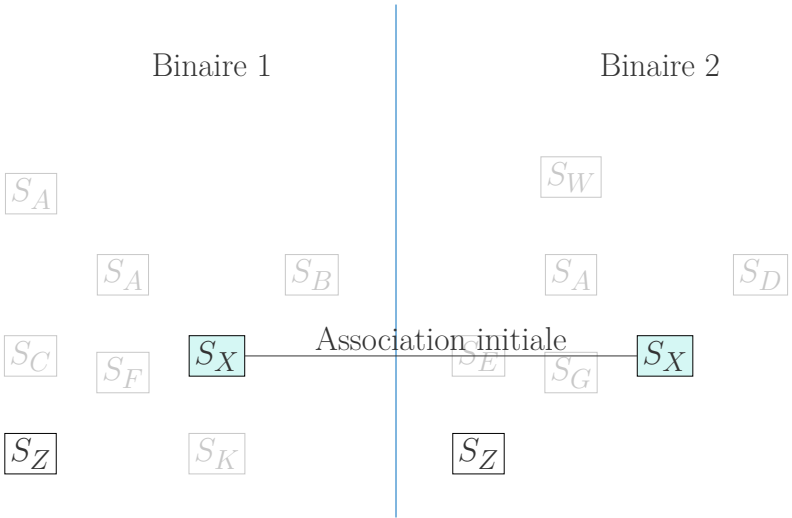
Binaire 1



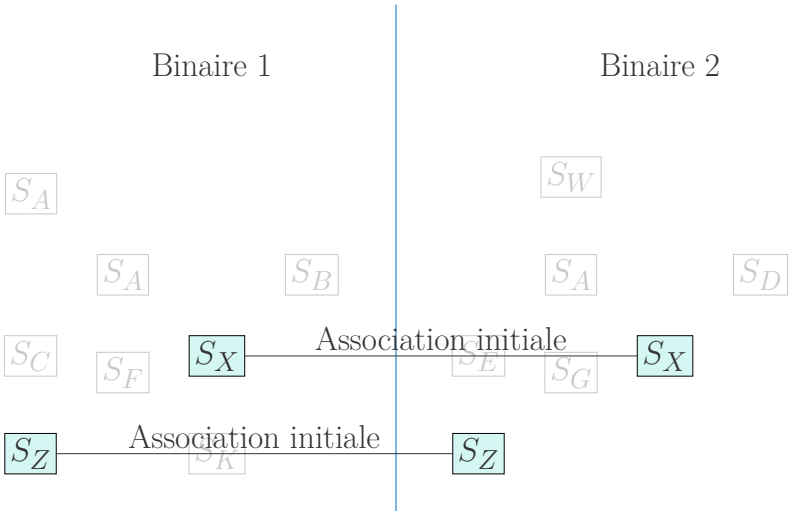
Binaire 2



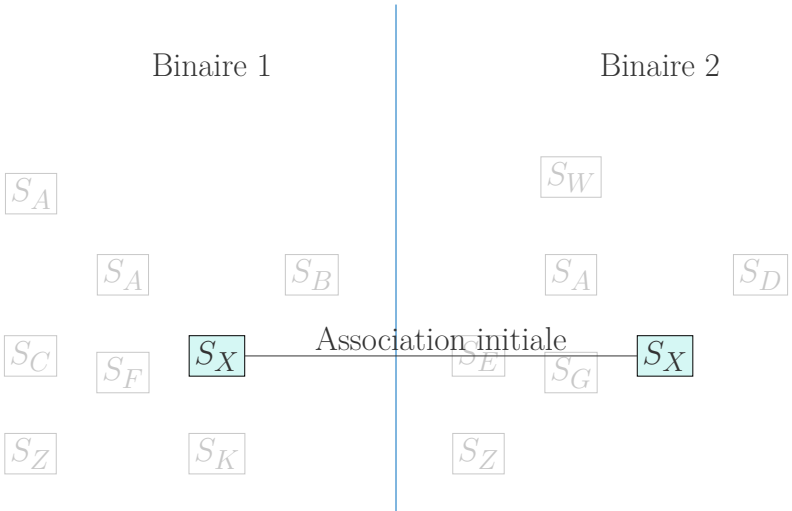
Algo 1 : Association initiale



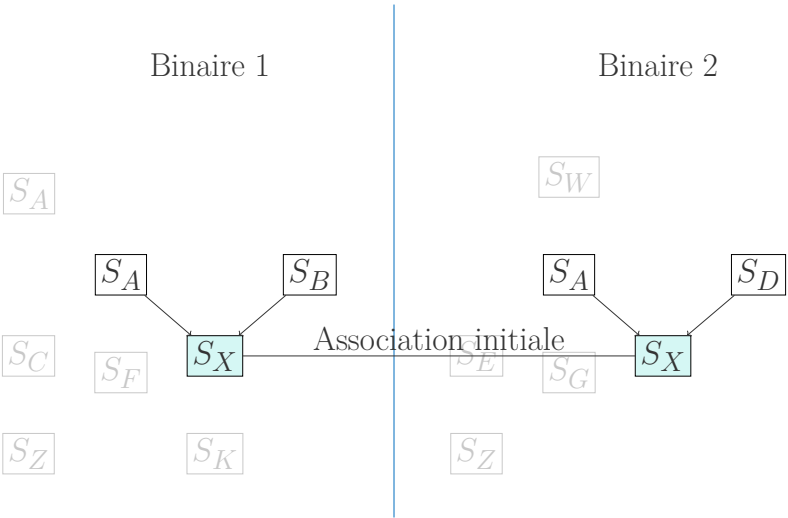
Algo 1 : Association initiale



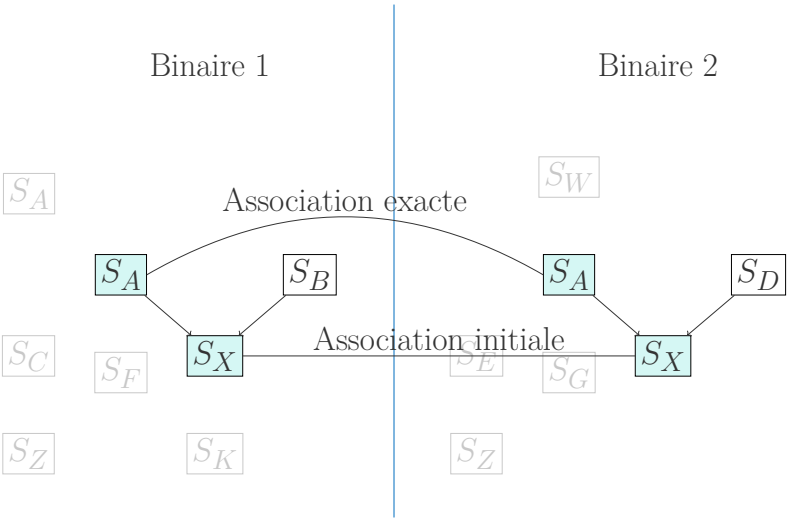
Algo 2 : Propagation montante



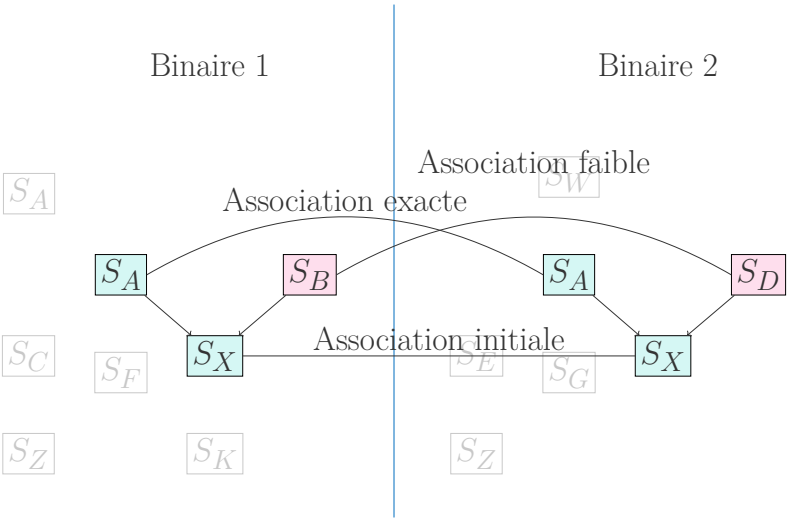
Algo 2 : Propagation montante



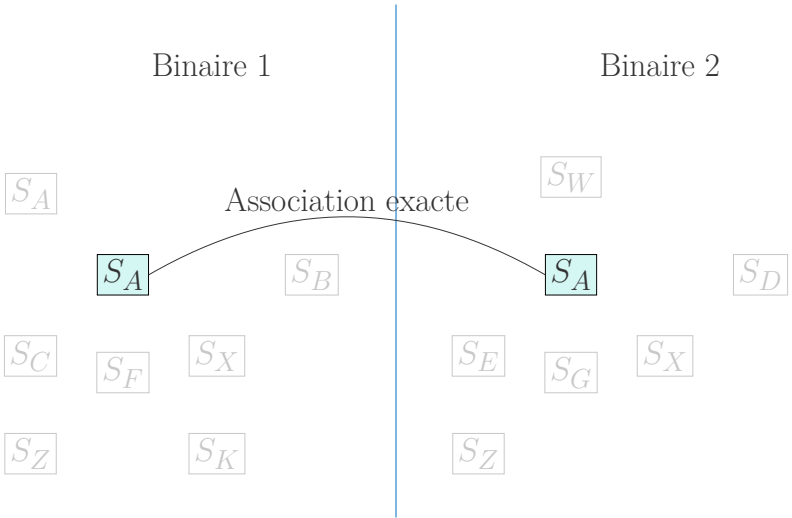
Algo 2 : Propagation montante



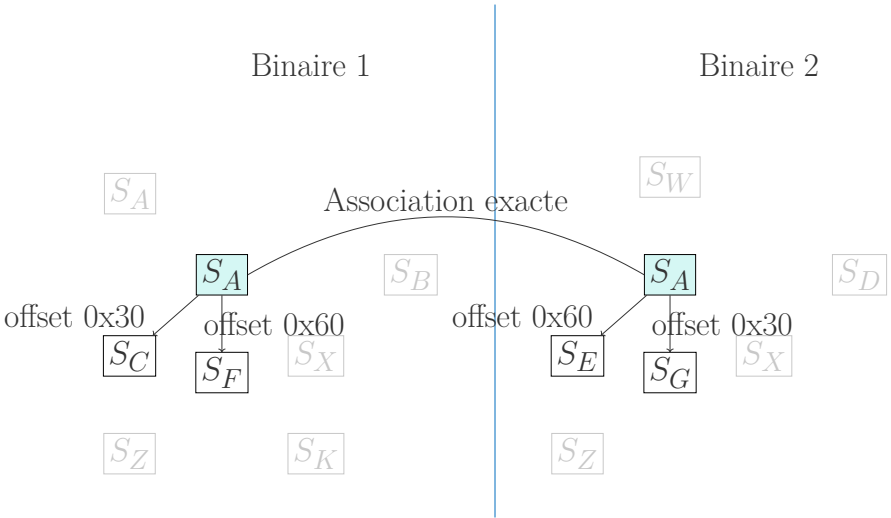
Algo 2 : Propagation montante



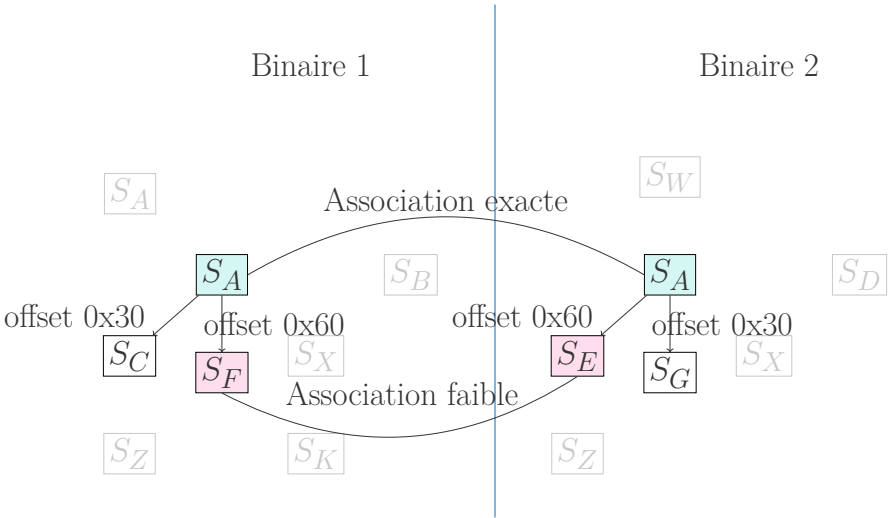
Algo 3 : Propagation descendante



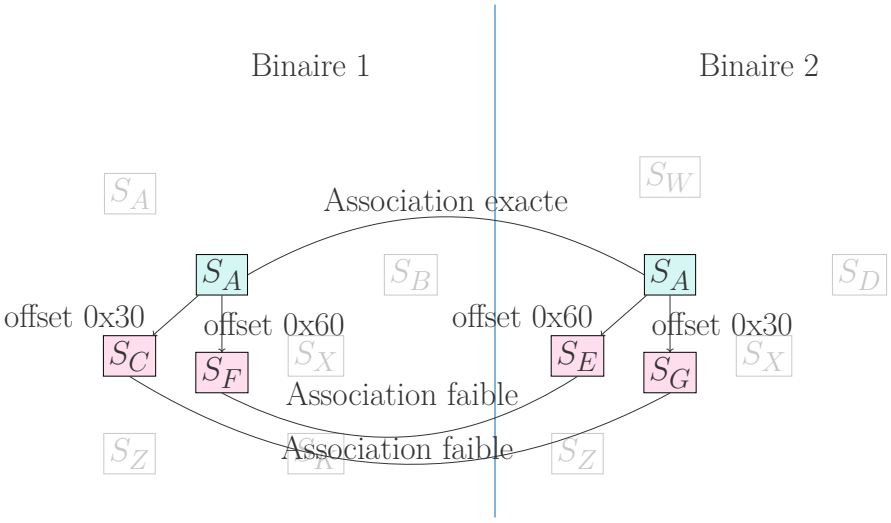
Algo 3 : Propagation descendante



Algo 3 : Propagation descendante



Algo 3 : Propagation descendante



Enchainement des algorithmes

```
Fonction association():
    algo1_association_initiale()
    répéter
        algo2_propagation_montante()
        algo3_propagation_descendante()
    tant que nouvelles associations ajoutées
```

DGA

Résultats et applications

- Grande confiance dans les résultats obtenus
- Utilisable sur des gros binaires
- Rapidité de propagation (e.g. Chromium ~2h)

DGA

Conclusion intermédiaire

- Répond à notre besoin
- Activement utilisé

Mais

- Pourrait obtenir un plus grand nombre de correspondances
- Ne supporte pas le multi-architecture
- Ne supporte pas les modifications importantes.

DGA

Contenu

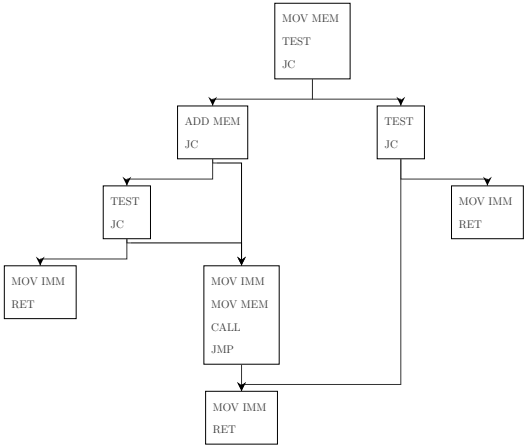
DGA

Vecteur de caractéristiques

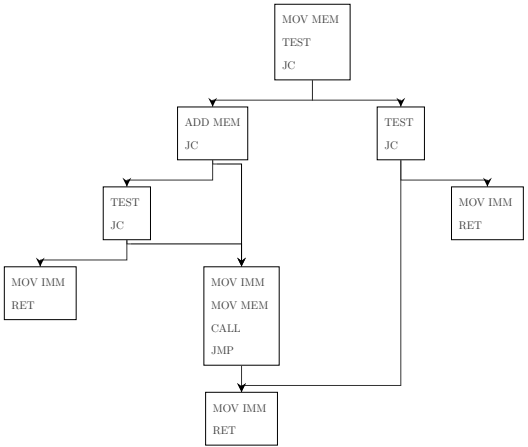
- Graphe de flot de contrôle
- Instructions
- Graphe d'appel

DGA

Vecteur de caractéristiques



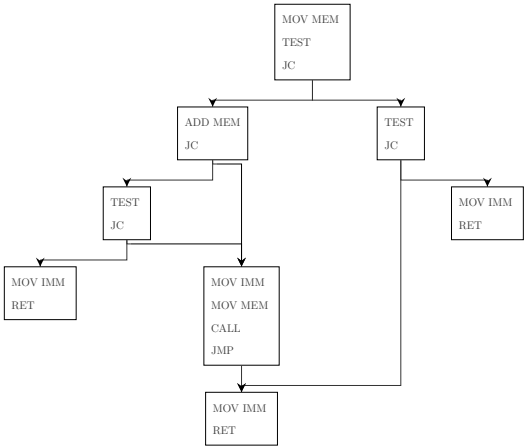
Vecteur de caractéristiques : Graphe de flot de contrôle



- Hauteur du graphe :
  minimale et maximale en nombre d'instructions et de blocs basiques
- Largeur maximale :
  le nombre maximum de basiques blocs à même distance de l'entrée
- Blocs basiques
- Arêtes
- Boucles
- ...

Vecteur de caractéristiques : Graphe de flot de contrôle

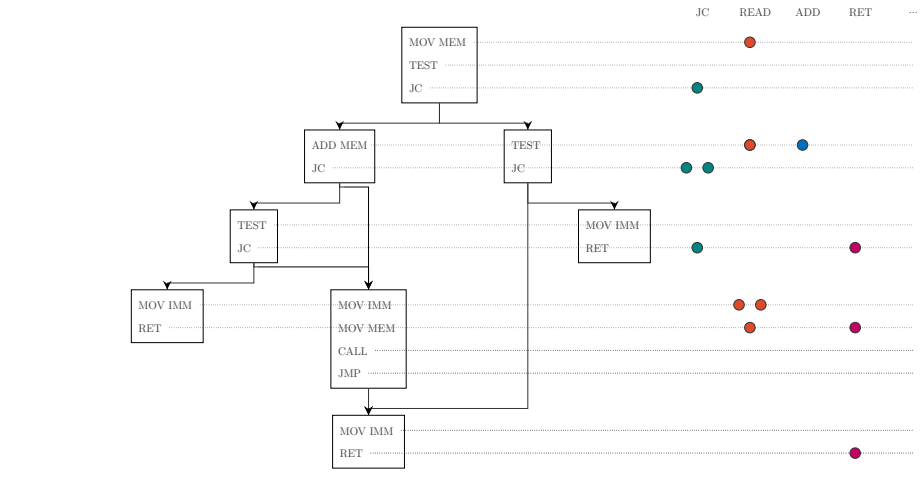
BB	Inst	Boucle	...	Xref



- Hauteur du graphe :
  minimale et maximale en nombre d'instructions et de blocs basiques
- Largeur maximale :
  le nombre maximum de basiques blocs à même distance de l'entrée
- Blocs basiques
- Arêtes
- Boucles
- ...

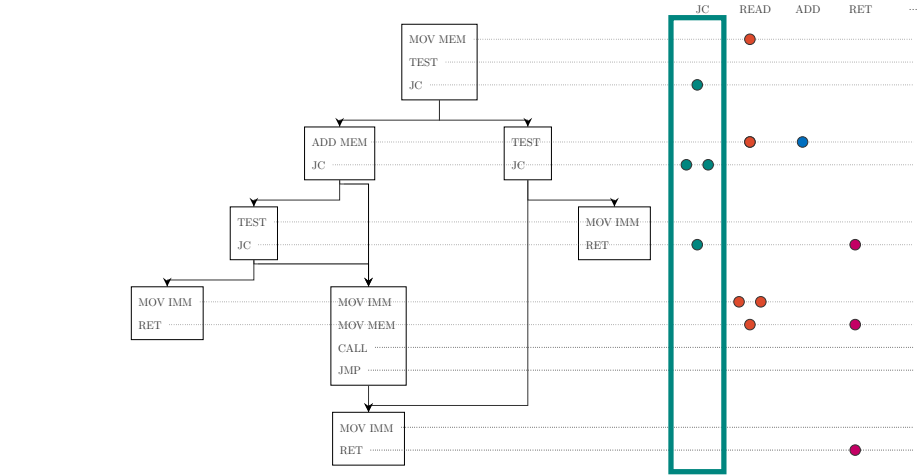
Vecteur de caractéristiques : Instructions

BB	Inst	Boucle	...	Xref



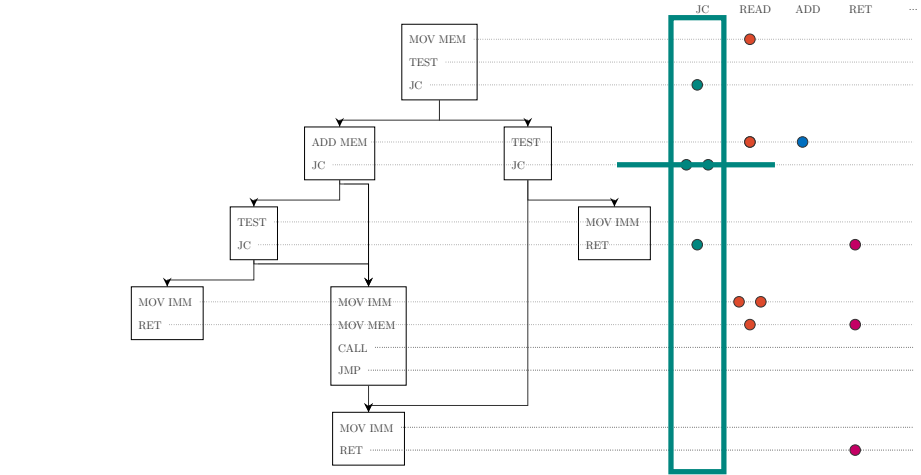
Vecteur de caractéristiques : Instructions

BB	Inst	Boucle	...	Xref



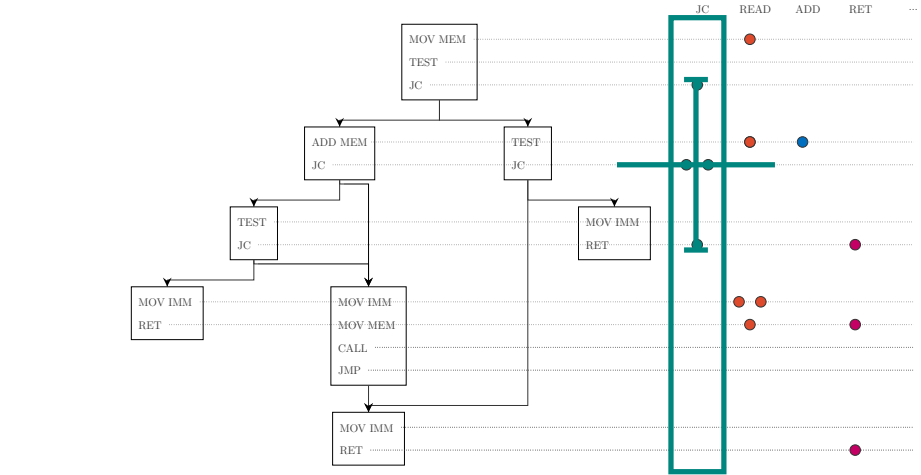
Vecteur de caractéristiques : Instructions

BB	Inst	Boucle	...	Xref



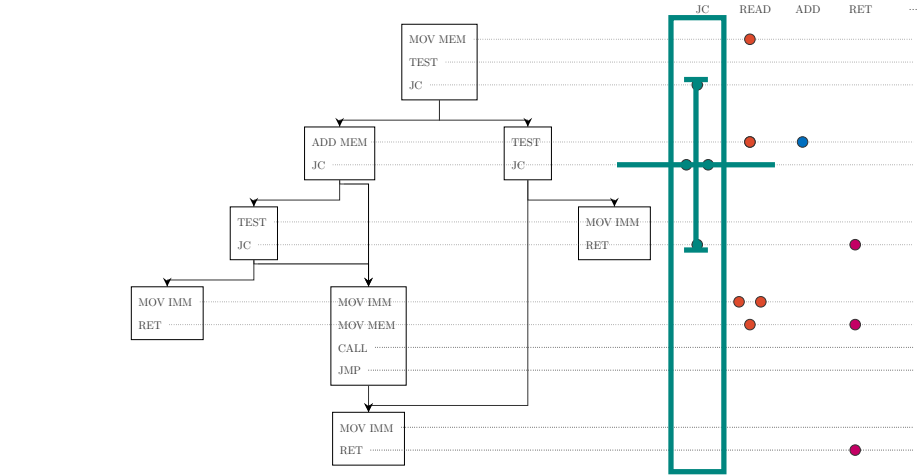
Vecteur de caractéristiques : Instructions

BB	Inst	Boucle	...	Xref



Vecteur de caractéristiques : Instructions

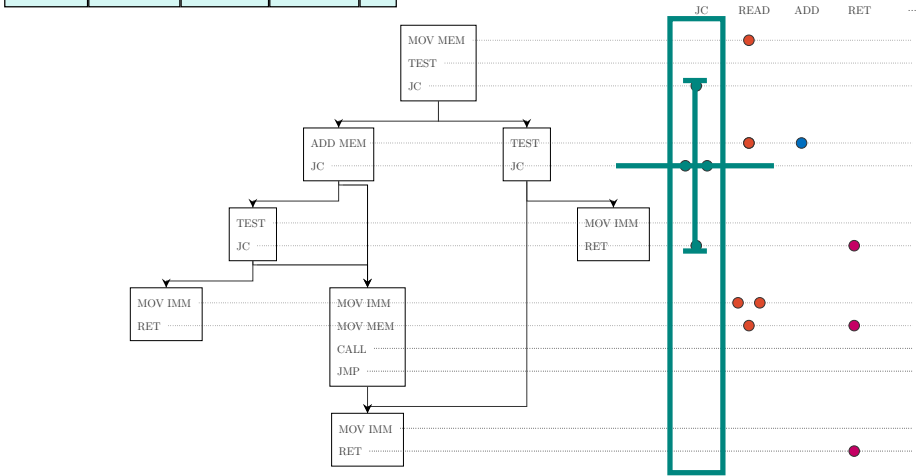
BB	Inst	Boucle	...	Xref



Vecteur de caractéristiques : Instructions

BB	Inst	Boucle	...	Xref

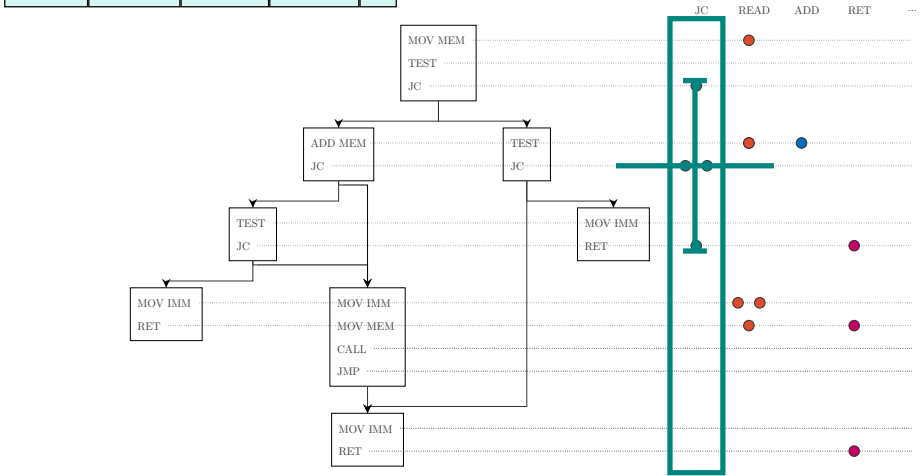
Nombre	Moyenne	Mediane	Variance	...



Vecteur de caractéristiques : Instructions

BB	Inst	Boucle	...	Xref	JC	READ	ADD	...	...	...	...	RET

Nombre	Moyenne	Mediane	Variance	...



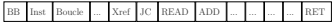
Vecteur de caractéristiques : Graphe d'appel

BB	Inst	Boucle	...	Xref	JC	READ	ADD	...	...	...	...	RET

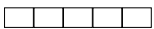
Vecteur de caractéristiques : Graphe d'appel

BB	Inst	Boucle	...	Xref	JC	READ	ADD	...	...	...	...	RET

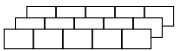
Vecteur de caractéristiques : Graphe d'appel



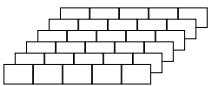
Vecteur de caractéristiques : Graphe d'appel



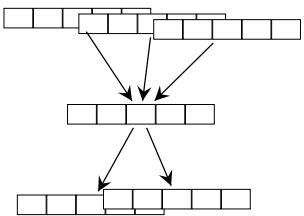
Vecteur de caractéristiques : Graphe d'appel



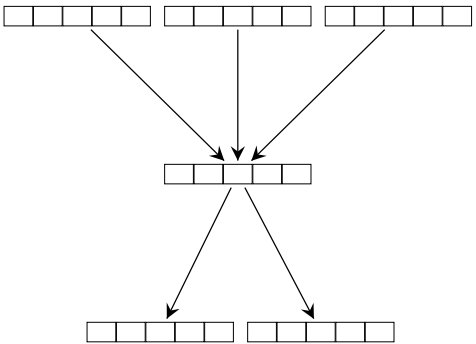
Vecteur de caractéristiques : Graphe d'appel



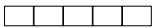
Vecteur de caractéristiques : Graphe d'appel

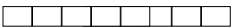


Vecteur de caractéristiques : Graphe d'appel

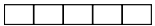


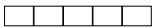
Vecteur de caractéristiques : Graphe d'appel



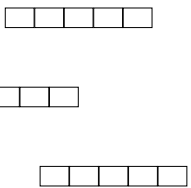


Vecteur de caractéristiques : Graphe d'appel

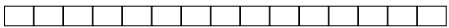




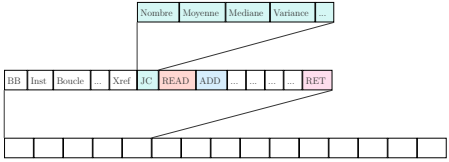
Vecteur de caractéristiques : Graphe d'appel



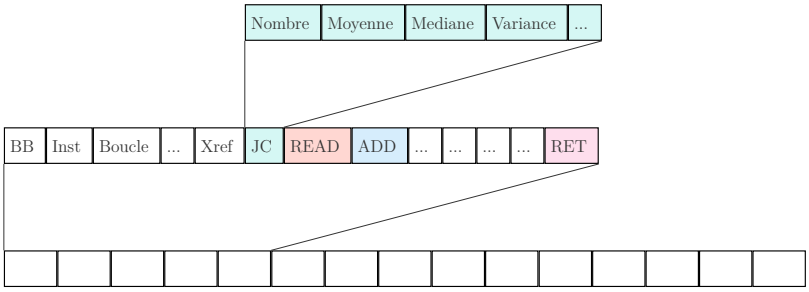
Vecteur de caractéristiques : Graphe d'appel



Vecteur de caractéristiques : Graphe d'appel



Vecteur de caractéristiques : Graphe d'appel



Distance entre vecteurs de caractéristiques

- Nombre de dimensions élevé
- Distances difficiles à définir
  - Quelques paramètres prépondérants
  - Certaines coordonnées sont liées
- Frontières floues

DGA

Définition

- Problème : étant donnés 2 vecteurs, représentent-ils la même fonction ?
- Solution : un réseau de neurones
- Méthode : apprentissage supervisé
- Données d'entrée : un ensemble de vecteurs de fonctions connues

DGA

Corpus d'apprentissage

- Suffisamment fourni
- Facilement accessible
- Représentatif des données analysées

DGA

Corpus d'apprentissage

- Suffisamment fourni
- Facilement accessible
- Représentatif des données analysées

Dépôt Linux debian

- Wheezy & Jessie & Stretch
- 400k fichiers
- 50M functions
- À faire évoluer une fois la démarche validée

DGA

Corpus d'apprentissage

Entraînement vs Validation

- Objectif : détecter et éviter le sur-apprentissage
- Exclusion d'une partie du corpus de l'entraînement
- Tests sur cette partie
- Isolation de 3 binaires : nfsd.ko, cc1plus, libxhtml.so
  - environ 100k fonctions

DGA

Extraction des données d'entrée

- Vecteur de caractéristiques + Architecture

Extraction des données d'entrée

- Vecteur de caractéristiques + Architecture

Normalisation

- Pour chaque coordonnée, distribution :
  - non équirépartie
  - potentiellement non bornée
  - différente

DGA

Extraction des données d'entrée

- Vecteur de caractéristiques + Architecture

Normalisation

- Pour chaque coordonnée, distribution :
  - non équirépartie
  - potentiellement non bornée
  - différente
- Normalisation par $\frac{2\times\log(1+X\text{-}min)}{\log(1+max\text{-}min)\text{-}1}$

DGA

Extraction des données d'entrée

- Vecteur de caractéristiques + Architecture

Normalisation

- Pour chaque coordonnée, distribution :
  - non équirépartie
  - potentiellement non bornée
  - différente
- Normalisation par $\frac{2\times\log(1+X-min)}{\log(1+max-min)-1}$
  - de 0 à 1
  - maximise la variance

DGA

Extraction des données d'entrée

- Pour chaque vecteur, on conserve le nom du binaire et le nom de la fonction
- Hypothèse sur le corpus :
  - Deux instances de fonctions sont sémantiquement identiques si les deux noms le sont (binaire+fonction)
  - Elles sont différentes sinon

DGA

Conception de l'algorithme d'apprentissage automatique

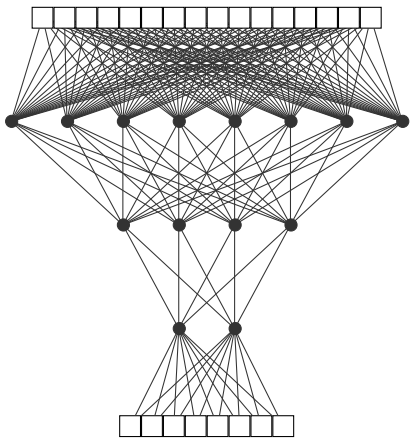
V=950

n1=512

n2=256

n3=128

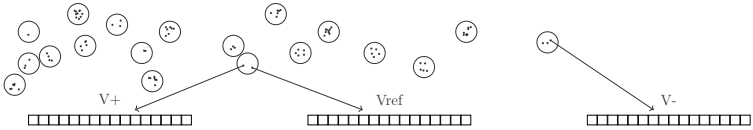
$Vp = 8$



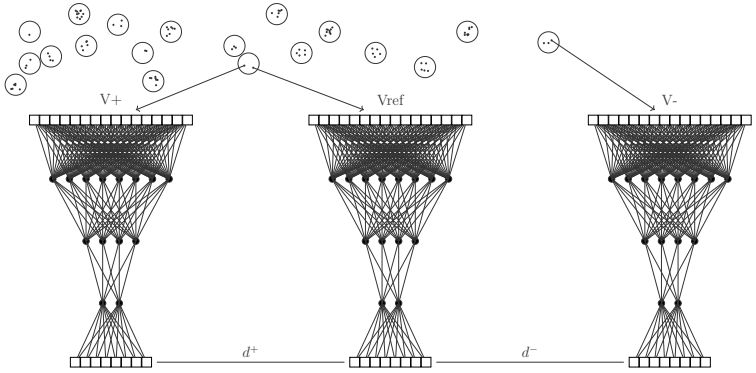
DGA

Apprentissage : fonctions étiquetées

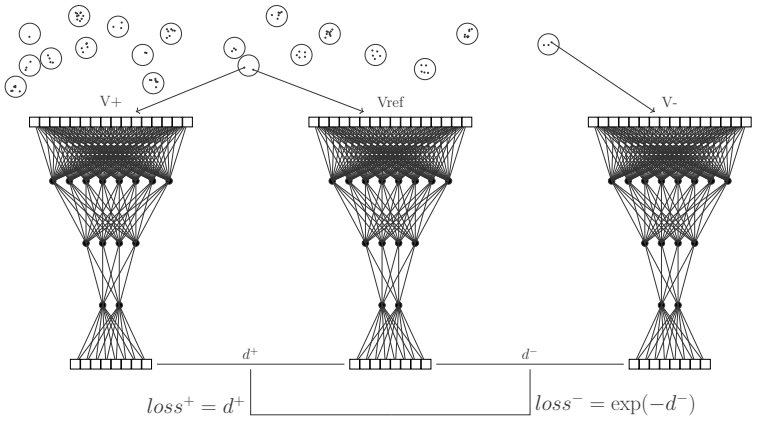
Apprentissage : + et -



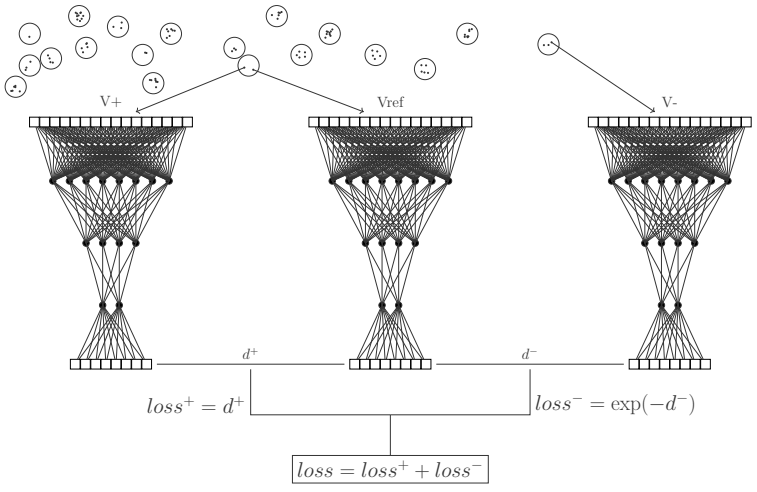
Apprentissage : réseau de projection



Apprentissage : fonction de coût



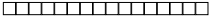
Apprentissage : fonction de coût



Exploitation

Base

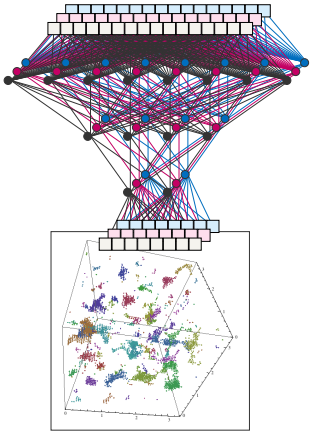
Fonction



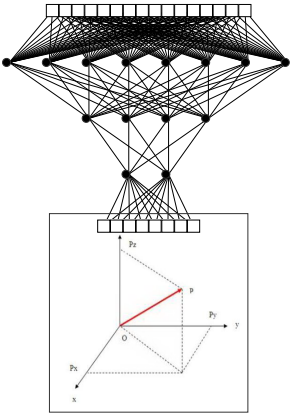
DGA

Exploitation

Base



Fonction



Résultats

Comparaison d'une fonction avec les autres versions du même binaire

```
[LADDER][EVAL][0.00s][0] metric: 477.96982
Ladder for [876864][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]: 933.70588 [17 sibblings], ranks in [6 - 1335]/1345
[ ][ 1079790][0.504219574][0.005551429][nfsd -- atomic_dec_and_spin_lock (Intel 80386 / 32 bits)]
[ ][  130338][0.752887033][0.009280259][nfsd -- __copy_from_user_ll (Intel 80386 / 32 bits)]
[ ][  533093][1.091241859][0.010920132][nfsd -- arm_copy_from_user (ARM / 32 bits)]
[ ][ 1079883][0.916691100][0.012009483][nfsd -- rt_spin_unlock (Intel 80386 / 32 bits)]
[ ][  129502][0.900925817][0.012271119][nfsd -- pv_lock_ops (Intel 80386 / 32 bits)]
[ ][ 1079784][1.071697658][0.014492417][nfsd --  mutex_lock (Intel 80386 / 32 bits)]
[X][  943605][0.810944381][0.014661974][nfsd -- remove_proc_entry (ARM / 32 bits)]
[ ][  900646][1.071307493][0.015965492][nfsd -- __csum_partial (PowerPC or cisco 4500 / 32 bits)]
[ ][ 1080171][0.892108948][0.016692763][nfsd -- rt_up_write (Intel 80386 / 32 bits)]
[ ][   70770][1.121043705][0.017173097][nfsd -- .rpc_shutdown_client (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   71397][1.062122340][0.018237174][nfsd -- .qword_addhex (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   71479][1.045865227][0.018292412][nfsd -- .wake_up_process (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][ 1079991][0.979788133][0.018413384][nfsd -- __mutex_do_init (Intel 80386 / 32 bits)]
[ ][   71277][1.046812001][0.018891755][nfsd -- .mnt_want_write_file (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   70696][1.044534709][0.019041903][nfsd -- .mnt_drop_write_file (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   71573][1.073005776][0.019122235][nfsd -- .rpc_unlink (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   70543][1.063754186][0.019183440][nfsd -- .vfs_rmdir (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   71261][1.081320947][0.019579004][nfsd -- .cache_check (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   70625][1.072801285][0.019587792][nfsd -- .locks_end_grace (64-bit PowerPC or cisco 7500 / 64 bits)]
[ ][   71483][1.093553917][0.019799624][nfsd -- .set_posix_acl (64-bit PowerPC or cisco 7500 / 64 bits)]
```

Résultats

Comparaison d'une fonction avec les autres versions du même binaire

```
[LADDER][EVAL][120.00s][337744] metric: 114.80121
Ladder for [876864][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]: 8.00000 (17 sibblings), ranks in [2 - 39]/1345
[ ][ 1079790][0.504219574][0.002655557][nfsd -- atomic_dec_and_spin_lock (Intel 80386 / 32 bits)]
[ ][  129536][1.134311844][0.002689022][nfsd -- set_fs (Intel 80386 / 32 bits)]
[X][  943605][0.810944381][0.002886003][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][  947306][2.049949634][0.005552270][nfsd -- remove_proc_entry (PowerPC or cisco 4500 / 32 bits)]
[X][  198972][5.925486433][0.005908569][nfsd -- remove_proc_entry (PowerPC or cisco 4500 / 32 bits)]
[X][  781199][3.443695263][0.006057259][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][   52439][1.088887903][0.006791204][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][  756194][5.790339007][0.010467720][nfsd -- remove_proc_entry (x86-64 / 64 bits)]
[ ][  781202][1.069555127][0.012558789][nfsd -- net_generic.part.1_0 (Intel 80386 / 32 bits)]
[ ][  781584][1.068393730][0.013884161][nfsd -- net_generic.part.3 (Intel 80386 / 32 bits)]
[ ][  781626][1.062813000][0.014329499][nfsd -- net_generic.part.6 (Intel 80386 / 32 bits)]
[ ][  781423][1.087630798][0.016591001][nfsd -- net_generic.part.1 (Intel 80386 / 32 bits)]
[X][  496018][1.709477965][0.017667634][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][  372374][5.332226244][0.018103490][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[ ][  781276][1.069777819][0.020596625][nfsd -- net_generic.part.1_1 (Intel 80386 / 32 bits)]
[ ][ 1079771][1.176402809][0.022174485][nfsd -- call_rcu (Intel 80386 / 32 bits)]
[X][ 1080196][5.569054868][0.022854714][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][ 1028625][4.934297144][0.022904934][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][  533481][1.394036595][0.023260947][nfsd -- remove_proc_entry (ARM / 32 bits)]
[ ][  888361][2.356059694][0.029136794][nfsd -- put_zone_device_page (x86-64 / 64 bits)]
```

Résultats

Comparaison d'une fonction avec les autres versions du même binaire

```
[LADDER][EVAL][540.00s][1608032] metric: 45.34632
Ladder for [876864][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]: 0.00000 (17 sibblings), ranks in [0 - 16]/1345
[X][  372374][5.332226244][0.001001099][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][  781199][3.443695263][0.001294585][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][  943605][0.810944381][0.002491688][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][ 1028625][4.934297144][0.002592712][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][  496018][1.709477965][0.003801425][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][ 1080196][5.569054868][0.004284129][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][   52439][1.088887903][0.007239995][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][  947306][2.049949634][0.009259175][nfsd -- remove_proc_entry (PowerPC or cisco 4500 / 32 bits)]
[X][  269148][6.218058688][0.012043204][nfsd -- remove_proc_entry (x86-64 / 64 bits)]
[X][  888060][5.620438449][0.013195468][nfsd -- remove_proc_entry (x86-64 / 64 bits)]
[X][  676419][3.700961476][0.015630072][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][  186528][5.473922680][0.015866069][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][  198972][5.925486433][0.017689172][nfsd -- remove_proc_entry (PowerPC or cisco 4500 / 32 bits)]
[X][  533481][1.394036595][0.022609925][nfsd -- remove_proc_entry (ARM / 32 bits)]
[X][  900676][7.413600353][0.024682559][nfsd -- remove_proc_entry (PowerPC or cisco 4500 / 32 bits)]
[X][  130054][6.355939734][0.031016290][nfsd -- remove_proc_entry (Intel 80386 / 32 bits)]
[X][  756194][5.790339007][0.044317406][nfsd -- remove_proc_entry (x86-64 / 64 bits)]
[ ][  781584][1.068393730][0.044861235][nfsd -- net_generic.part.3 (Intel 80386 / 32 bits)]
[ ][  877037][1.064036173][0.047644138][nfsd -- __be32_to_cpup.isra.7 (Intel 80386 / 32 bits)]
[ ][  129536][1.134311844][0.086903065][nfsd -- set_fs (Intel 80386 / 32 bits)]
```

Total

```
Fonction association():
    algo1_apprentissage_automatique()
    algo4_associations_initiales()
    répéter
        algo2_propagation_montante()
        algo3_propagation_descendante()
        pour chaque association de la liste faire
            si score ia élevé
                ajouter association à la liste point de pivot
    tant que nouvelles associations ajoutées
```

DGA

Travaux restants

- Vecteurs : ajouter d'autres paramètres discriminants
- Apprentissage : Gradient Boosting, ...
- Sortie : "function dating"

DGA

Conclusion

- Fonctionne
- Utilisé
- IA prometteur

NOTE : pour en savoir plus, lire le papier

DGA

One more thing

- https://github.com/DGA-MI-SSI/YaCo
- GPLv3

NOTE : pour en savoir plus, lire le papier

DGA

Questions ?

- Rendez-vous ce soir pour en discuter

DGA

Améliorations

- Intégrer l'IA
  - Dans la version open source sur Github