Automatic Generation of Opaque Constants Based on the K-Clique Problem for Resilient Data Obfuscation

Roberto Fellin

Master of Science in Computer Science
University of Trento

21 Febbraio 2018

Outline

Outline

Introduzione

Obiettivo:

- Generare un **opaque costant** con le seguenti proprietà:
  - Req1: l'offuscazione non deve cambiare il comportamento del programma
  - Req2: deve essere difficile decodificare il valore della costante dal programma
  - Req3: deve essere facile costruire la costante opaca
  - Req4: deve essere veloce computare la costante a run time
- Confrontare questo metodo con quello di 3SAT

Outline

3SAT

Generare una formula 3SAT:

- non soddisfacibile
- difficile: *n* variabili, *4.3*n* clauses

Riduzione

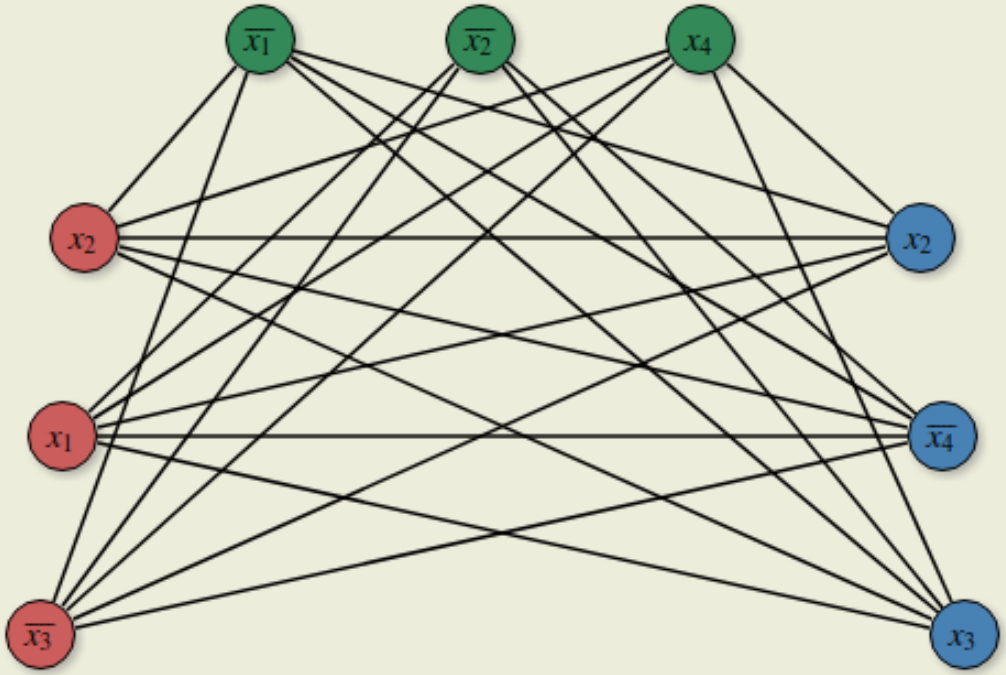
Data la formula ϕ con k clauses, generare il grafo G(V,E) come segue:

- V: un nodo per ogni letterale delle k clauses
- E: arco tra nodi di clause diverse e con lettterale non opposto (es. x $\neg$x)

La formula ϕ è soddisfacibile iff G contiene un k-clique

Esempio

$\Phi = (x_2 + x_1 + \overline{x_3}) \cdot (\overline{x_1} + \overline{x_2} + x_4) \cdot (x_2 + \overline{x_4} + x_3)$



Dimostrazione

- K-Clique $\rightarrow$ 3SAT:
  - supponiamo di avere il grafo G con un clique di k
  - allora avrò esattamente un nodo per cluster (cluster inteso come i 3 nodi dellla clause)
  - questi nodi sono tutti collegati, quindi posso attribuire a tutti TRUE
  - allora ϕ soddisfacibile
- 3SAT $\rightarrow$ K-Clique:
  - supponiamo di avere un assegnamento per ϕ
  - seleziono i nodi corrispondenti nel grafo a cui ho assegano TRUE
  - formeranno un clique perchè c'è un arco tra di loro visto che non sono della stessa clause e sono simultaneamente vere
  - allora G ha un k-clique

Outline

RQ_1: tempo per verificare 3SAT

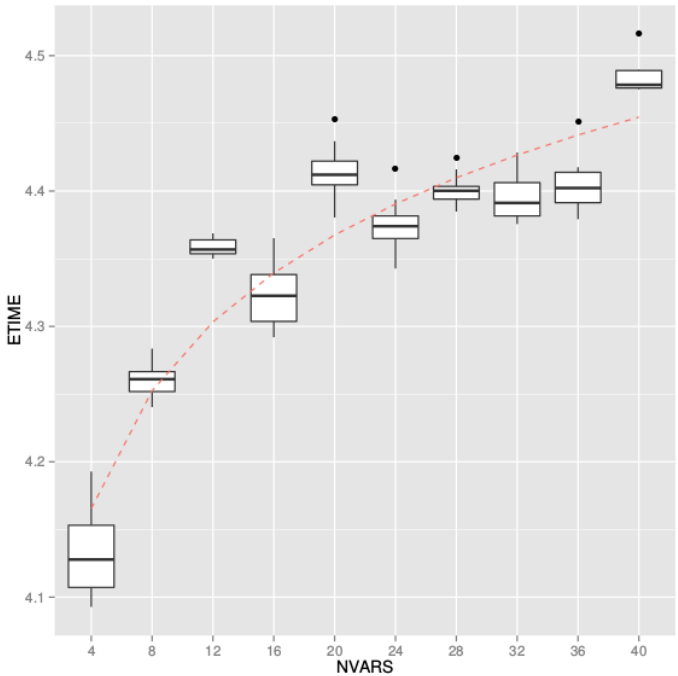
Seguono i test utilizzando SMT solver Yices

NVARS	PSAT	MEAN(ETIME)	SD(ETIME)
50	0.60	0.00013	0.00019
100	0.51	0.00176	0.00143
150	0.49	0.01134	0.00645
200	0.45	0.09320	0.06800
250	0.37	1.09245	0.74141
300	0.37	25.42116	26.68942
350	0.30	828.21444	837.64963

ne segue che con meno di 200 variabili, il tempo è trascurabile

RQ_2: tempo per computare il valore della costante opaca a runtime con il codice offuscato

Segue l'overhead a runtime utilizzando 1000000 costanti opache



l'aumento è logaritmico e segue la seguente formula:

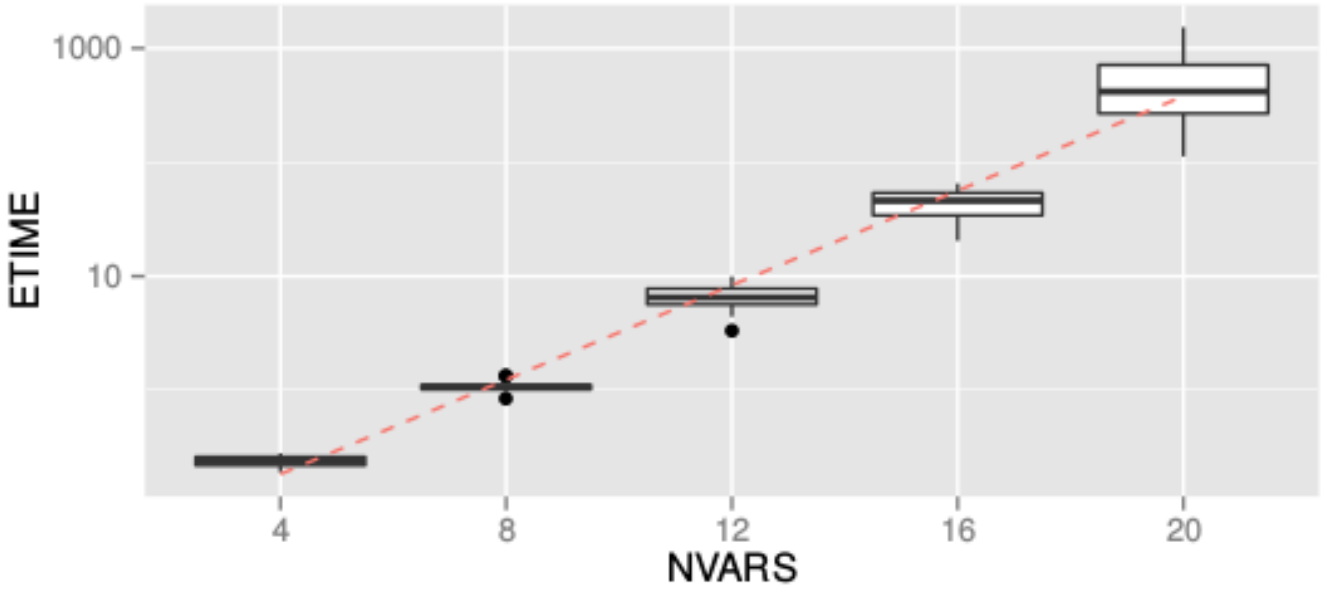
$$ETIME = 3.99 + 1.12 * log(NVARS) \quad (1)$$

RQ_3: elasticità delle costanti opache con metodo 3SAT

- Valutiamo l'elasticità come il tempo necessario per rompere l'offuscasione
- Per rompere l'offuscazione intendiamo capire che non attraverserà mai alcuni path
- questo perchè la formula è sempre non soddisfacibile
- l'attaccante quindi deve risolvere 3SAT

RQ_3: elasticità delle costanti opache con metodo 3SAT

Segue il grafico del tempo impiegato da un tool che fa esecuzione simbolica (KLEE) per analizzare il programma

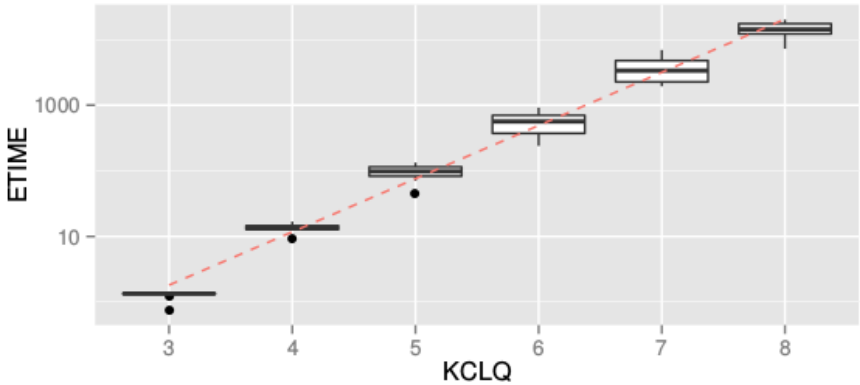


l'aumento è esponenziale e segue la seguente formula per analizzare un bit della costante:

$$ETIME = e^{-3.60+0.48*NVARS} \quad (2)$$

RQ_3: elasticità delle costanti opache con metodo K-Clique

- utilizzando lo stesso metodo (KLEE), con NVARS=4, si crea un 17-Clique con 51 nodi
- dopo 9 giorni analisi non era ancora completa
- esperimenti con meno di 51 nodi, generando random un grafo



l'aumento è esponenziale e segue la seguente formula per analizzare un bit della costante, dove KCLQ è la dimensione del clique:

$$ETIME = e^{-4.99+1.86*KCLQ} \quad (3)$$

Possiamo notare il passaggio esponenziale tra risolvere 3SAT e K-Clique

Outline

Requisiti

- Req1: comportamento del programma invariato: perchè si fa il controllo che la formula sia non soddisfacibile
- Req2: problema difficile da risolvere a run time: RQ4 mostra che il tempo per risolvere K-Clique è esponenziale
- Req3: costante facile da construire: dagli esperimenti, ill tool impiega millisecondi per generare il codice per il K-Clique
- Req4: veloce computare il valore a run time: da RQ2, tempo logaritmico

Outline

Conclusioni

- Costante opaca generata tramite K-Clique soddisfa i requisiti
- Tentativo di tornare indietro alla formula 3SAT iniziale dal K-Clique:
  - nella letteratura [1] la riduzuione K-Clique $\leq_p$ 3SAT è la seguente:
    - K-Clique (n nodi) $\leq_p$ Subgraph isomorphism problem (un grafo da n e uno da k nodi): aggiungere un grafo completo di k nodi
    - Subgraph isomorphism problem $\leq_p$ SAT: aggiungere una variabile per ogni combinazione di nodi: n*k variabili
    - SAT $\leq_p$ 3SAT: con almeno n*k variabili
  - questa riduzione non porta allo stesso problema 3SAT

Outline

Riflessioni

- esiste una qualche procedura per tornare indietro alla formula 3SAT iniziale?
- quanto è resistente questa offuscazione ad analisi dinamica?
- è possibili individuare il pattern dell'offuscazione e capire quindi che la formula è sempre non soddisfacibile?