Large Spurious Cycle in Global Static Analyses and Its Algorithmic Mitigation

Hakjoo Oh pronto@ropas.snu.ac.kr

School of Computer Science and Engineering Seoul National University Korea

Dec 14, 2009 Asian Symposium on Programming Language and Systems

4.1 First Stop Abstraction : Value
$$V_p = 2^{k} - 2^{k} - 1^{k} (1 \le 1)$$

First, we abstract the concrete parsing domain V_p ($\delta \neq 10 \ y/(1 \le 10 \land ret)$)
 $2^{k'}$ by establishing the Calibrate Parsing
 $2^{k'}$ by establishing the Calibrate Parsing
 $p \in 4$. First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} - \delta$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} = 0$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} = 0$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} = 0$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} = 0$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 2^{k'} = 0$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 1$ First Step Abstraction : Value $V_p = 0$
 $p = k^{k'} = 1$ First Step Abstraction : Value $V_p = 1$ First Step Abstraction : Value V_p

the sether ansist stick concludes that the program in any six concludes that the program may generate syntactically



4.1 First Step Abstraction : Value
$$V_{\rho} = 2^{k} - 2^{k}$$

First, we abstract the concrete parsing domain V_{ν} (of $V_{\nu}^{2} \supseteq \mathcal{Y}(\underline{i}_{\nu} = 10 \land \text{ret})$
 2^{k} by establishing high $(\nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu} = 10 \land \text{ret})$
 2^{k} by establishing high $(\nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu} = 10 \land \text{ret})$
for the code fragments $x = a_{\nu} a_{\nu} a_{\nu} D_{\nu} \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu} = 10 \land \text{ret})$
 $p = 1$ $\mathcal{Y}(\underline{i}_{\nu} = \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu} = 10 \land \text{ret})$
 $p = 1$ $\mathcal{Y}(\underline{i}_{\nu} = \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu} = 10 \land \text{ret})$
 $p = prove (p_{new} p) \otimes \nabla \mathcal{Y}(\underline{i}_{\nu}) \otimes \nabla \mathcal{Y}(\underline$

UNIVERSITY

1.900 110
bison-1.875
proftpd-1.3.1
apache-2.2.2

S =

cs is appear from the Eolleptilis an e Whitogramminifgthey are Course are hely whitogramminify they are

the set he parsa stack concludes that 1

410/832(49%) 940/1,096(85%) 1,364/2,075(66%)

12,558/18,110(69%) 35,386/41,062(86%) 71,719/95,179(75%)

k-limiting is not much effective to mitigate the problem.



	Program	Basic-blocks in the largest cycle		
-	spell-1.0	751/782 <mark>(95%)</mark>		
$[t]_{\hat{\sigma}}^{1}\sigma =$	gzip-1.2.4a	5,988/6,271 <mark>(95%)</mark>		
$parse(p, x), y) \qquad [f_1, f_2]_{\hat{P}}^1 \sigma =$	sed-4.0.8	14,559/14,976 <mark>(97%)</mark>		
$[[,e]]_{\hat{P}}^{1}\sigma =$	tar-1.13	10,194/10,800(94%)		
where Parse_action :	wget-1.9	15,249/16,544(92%)		
anticsextension of $parse_actic$	bison-1.875	12,558/18,110(69%)		
established as follows. 16 has to 100C = A	proftpd-1.3.1	35,386/41,062(86%)		
(p, c) erated solved conform to 2	apache-2.2.2	71,719/95,179(75%)		
$p \in P.parse(p_{r}c) = f(p) \} \frac{18}{8}$	•			









cs is the from the collecting second where using the theory of theory of theory of the theory of the



e - (I - I0 / Iec) $EQ(\beta) = No$ if $\beta \not\equiv \lambda \land (\mu \models \beta \oplus$ **4.1** First Step Abstraction : Value $V_{\hat{P}} = 2^{\hat{P}} \rightarrow 2^{\hat{P}}$ First, we abstract the concrete parsing domain V_P to $V_{\hat{P}} \stackrel{10}{=} 2^{P} (i = 10 \land \texttt{ret})$ 2^{P} by establishing the Galipisconnection $V_{P} \xrightarrow{\gamma} V_{P_{i}} \xrightarrow{\gamma} V_{P_{i$ for the code fragments x and p The P. $\bigcup \{f \in \mathcal{P} \text{ stract Parsing}$ ation x.y is computed as ρ ϵ $\gamma = \lambda F. \bigcup Figst, we go stract the concrete parsing domain <math>V_P$ to $V_{\hat{P}} = 2^P$ $(p_{init}, x.y)$ $\alpha = \lambda F.\lambda P. \bigcup \{f(p) \mid f \in F\}$ $\sigma \in Env_{\hat{P}} = Var \to V_{\hat{P}}$ $= parse(p_{init}, y) \text{ not } parse(p_{y}, y).$ se stack after parsing y from the initial $Env_{\hat{P}} \rightarrow V_{\hat{P}} \qquad \gamma = \lambda F. \bigcup \{S \mid \alpha(S) \subseteq f\}.$ ρ We cannot directly compute $p from p_x Env_p$ Then we derive the strategy and s for this domain as follows S_{0} ove concatenation problem elegantly. stack transition functions for $\sigma(x)$ y. Then we have $x \ e_1 \ e_2]_{\hat{P}}^0 \sigma = [\![e_2]\!]_{\hat{P}}^0 (\sigma[x \mapsto [\![e_1]\!]_{\hat{P}}^0 \sigma]\!]_{\hat{P}}^1 \in Env_{\hat{P}} \to V_{\hat{P}}$ $p.parse(p,x) \quad [[\texttt{or} \ e_1 \ e_2]]_{\hat{P}}^0 \sigma = [[e_1]]_{\hat{P}}^0 \sigma \cup [[e_2]]_{\hat{P}}^0 \sigma \quad [[x]]_{\hat{P}}^0 = \sigma(x)$ $w^{p.parse(p, y)}_{\hat{P}} = x \ e_1 \ e_2 \ e_3 \end{bmatrix}_{\hat{P}}^0 \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^0 (\sigma [x_{\text{[let } x \ e_1 \ e_2]}]_{\hat{P}}^0 \sigma = \llbracket e_2 \rrbracket_{\hat{P}}^0 (\sigma [x \mapsto \llbracket e_1 \rrbracket_{\hat{P}}^0 \sigma])$ isition functions f_x and f_y , we confix the function $f_{x,y}$ as follows. $\begin{bmatrix} f \end{bmatrix}_{\hat{P}}^0 \sigma = \begin{bmatrix} f \end{bmatrix}_{\hat{P}}^1 \sigma \begin{bmatrix} f \end{bmatrix}_{\hat{P}}^0 \sigma = \begin{bmatrix} f \end{bmatrix}_{\hat{P}}^1 \sigma \begin{bmatrix} f \end{bmatrix}_{\hat{P}}^0 \sigma = \begin{bmatrix} f \end{bmatrix}_{\hat{P$ $\llbracket t \rrbracket_{\hat{P}}^{1} \sigma = \lambda \underline{P}.Parse_action(\underline{P}, t) \quad fix \ \lambda k. \llbracket e_1 \rrbracket_{\hat{P}}^{0} \sigma \cup \llbracket e_2 \rrbracket_{\hat{P}}^{0} (\sigma[x \mapsto k])])$ (p, x.y)parse(p, x), y $parse(p,x),y) = f_{p} = f_{p$ $[\![,e]\!]_{\hat{P}}^1 \sigma = [\![e]\!]_{\hat{P}}^0 \sigma$ $[\![f_1.f_2]\!]^1_{\hat{P}}\sigma = [\![f_2]\!]^1_{\hat{P}}\sigma \circ [\![f_1]\!]^1_{\hat{P}}\sigma$ where $Parse_action$: ² Details On a gorithm, correctness, and where $Parse_action$: ² Software $Parse_action$ is the natural set pde to $P \rightarrow B$ the Galois connection of extension of partse with $Propartse Action \in Prime (sec)$ established as follows 16 6 9 8 - 10 Provide action A and A a urse auction the tRespar€ Prime (sec) (p, d) erated bedes conform to the granner and the granner of the second 20 eratedesedes conform to the grant may for the given program 82 we $p \in P.parse(p_{rc}) = f(p)$ 1810 $S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init}^{\text{vpr}} \} : 2^{P}$ 0.128 1.08 2.90.055cs where from the Eolle etils an empty sonment of the any the propagation of the son of NATIONAL







cs is the from the Ebileptific an empty isonment, The anvension parts ment. Then we compare by whito gramphiligthey are eccessive (onclude that the generated on clude that the generated code on to the set of the set of parts of the program may generate syntactically the set of parts of the program may generate syntactically the set of parts of the program may generate syntactically the set of parts of the program may generate syntactically the set of parts of the program may generate syntactically the set of the set of the program may generate syntactically the set of the program may generate syntactically size of the set of the s





the sethe parsal stickeon cludes that the program six concludes that the program may generate syntactically



 $c = (I = I0 \land Iec)$ $EQ(\beta) = No$ if $\beta \not\equiv \lambda \land (\mu \models \beta \oplus$ 4.1 First Step Abstraction : Value $V_{\hat{P}} = 2^{\hat{P}} \rightarrow 2^{\hat{P}}$ First, we abstract the concrete parsing domain V_P to $V_{\hat{P}} \stackrel{10}{=} 2^{P} (\underset{\rightarrow}{i = 10 \land ret})$ 2^{P} by establishing the Galipisconnection $V_{P} \xrightarrow{\gamma} V_{P_{i}} \xrightarrow{\gamma} V_{P_{i$ for the code fragments x and y_{λ} The P. $(p_{init}, x.y)$ $(p_{init}, x.y)$ $(p_{init}, x.y)$ $(p_{arse}(p_{init}, x), y)$ $(p_{arse}(p_{arse}(p_{init}, x), y)$ $(p_{arse}(p_{init}, x), y)$ $(p_{arse}(p_{init}, x), y)$ $(p_{arse}(p_{init}, x), y)$ $(p_{arse}(p_{init}, x), y)$ $(p_{arse}(p_{ar$ ϵ $(parse(p_{init}, x), y)$ $(parse(p_{init}, x), y)$ Then we derive the abstract semantics for this domain as follows. $\alpha = \lambda F. \lambda P. \bigcup \{ f(p) \mid f \in F \}$ $\sigma \in Env_{\hat{p}} = Var \rightarrow V_{\hat{p}}$ $= parse(p_{init}, y) \text{ not } rse(rRie cursive price vector vect$ $\rho \lor \epsilon$ Ve cannot directly compute of from p_x to remove the star good to the period as follows. $\sigma \in Env_{\hat{\mathcal{P}}} = Var \to V_{\hat{\mathcal{P}}}$ to the parse stack transition function stack transition functions for the first of p.parse(p,x) [or $e_1 e_2$] p with put considering other calls in it. $w^{p.parse(p)} [\overset{y)}{\mathbb{T}e} x \ e_1 \ e_2 \ e_3]]_{\hat{P}}^0 \sigma = [[e_3]]_{\hat{P}}^0 (\sigma [x_{[] \stackrel{i}{\mathbb{T}et} x \ e_1 \ e_2]]_{\hat{P}}^0 \sigma = [[e_2]]_{\hat{P}}^0 (\sigma [x \mapsto [[e_1]]_{\hat{P}}^0 \sigma])$ isition functions f_x and f_y , we confix $\lambda k [e_1]_{\hat{P}}^0 \sigma[b_x [e_2]_{\hat{P}}^0 \sigma[x [e_1]_{\hat{P}}^0]] = [e_2]_{\hat{P}}^0 \sigma[x [e_1]_{\hat{P}}^0]$ on function $f_{x,y}$ as follows. $\llbracket f \rrbracket_{\hat{P}}^{0} \sigma = \llbracket f \rrbracket_{\hat{P}}^{1} \sigma \quad \llbracket re \ x \ e_{1} \ e_{2} \ e_{3} \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{2} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{3} e_{3}] \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{3} \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_{1} e_{3} e_{3} e_{3} e_{3}]$ $\begin{array}{c} p, x.y) \\ parse(p, x), y) \\ (p, y)) \circ (\lambda p. parse(p, x), y) \\ (p, y)) \circ (\lambda p. parse(p, x), y) \\ (p, y)) \circ (\lambda p. parse(p, x), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (\lambda p. parse(p, y), y) \\ (p, y) \circ (p, y) \\ (p, y) \circ (p, y) \\ (p, y) \circ (p, y) \\ (p, y)$ where $Parse_action : 2^P \rightarrow Token \rightarrow \mathbb{R}^P_e$ by the *algural* set anticsextension of *parse_action*: where $Parse_action : 2^P \rightarrow Tok \ a 2^f$ is the natural set $pde to P \rightarrow B$, the Galois connection λt is a set of the factor of particles, the figure P in (sec)established as follows. (p, d) erated bedes conform to the grant and the grant of the second of 20 eratedesedes conform to the grammar for the given program 82.1420 $p \in P.parse(p_{rc}) = f(p)$ 1810 $S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init}^{vpr} \} : 2^{P} = \begin{bmatrix} 1.8^{18} & 10_{2} & 9 & 20.1 \\ 1.8^{18} & 10_{2} & 10_{2} & 10_{2} & 10_{2} \\ 1.8^{18} & 10_{2} & 10_{2} & 10_{2} & 10_{2} \\ 1.8^{18} & 10_{2} & 10_{2} & 10_{2} & 10_{2} & 10_{2} \\ 1.8^{18} & 10_{2} & 10_{2} & 10_{2} & 10_{2} & 10_{2} & 10_{2} & 10_{2} & 10_{2} \\ 1.8^{18} & 10_{2}$ 0.128 8 1.02.9 0.055cs where from the Eolle etils an empty sonment of the any the propagation of the son of

the set he parsa stickeon cludes that the program six concludes that the program may generate syntactically

cs is defined from the Ebileptific an empty reprisonment. The anvension parts ment. Then we compare Swith (onclude that the generated on clude that the generated on clude that the generated of the graph of the gr

	$EQ(\beta)$	$\beta = No$ if $\beta \not\equiv \lambda \land$	$(\mu \models \beta \oplus$
ret) b			
O _{ret})			
ments	0	C	
to $V_{\hat{P}} = 2^{P} \rightarrow$	ρ	C	
$\Rightarrow W_{i} \ll (b_{i=10} \wedge b_{r})$	et)		
	δ	0	F
No und/al			C
hain as follows: In the last	k VS. I	лээ $_k$	
Newsel			
Ivormai	k VS. I	RSS_{k+1}	
Program		#Basic-Blocks]
Program	LOC	#Basic-Blocks]
Program spell-1.0	LOC 2,213	#Basic-Blocks 782	MAGINE
Program spell-1.0 barcode-0.96	LOC 2,213 4,460	#Basic-Blocks 782 2,634	MAGINE.
Program spell-1.0 barcode-0.96 httptunnel-3.3	LOC 2,213 4,460 6,174	#Basic-Blocks 782 2,634 2,757	MAGINE.
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a	LOC 2,213 4,460 6,174 7,327	#Basic-Blocks 782 2,634 2,757 6,271	MAGINE.
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a jwhois-3.0.1	LOC 2,213 4,460 6,174 7,327 9,344	#Basic-Blocks 782 2,634 2,757 6,271 5,147	MAGINE:
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a jwhois-3.0.1 parser	LOC 2,213 4,460 6,174 7,327 9,344 10,900	#Basic-Blocks 782 2,634 2,757 6,271 5,147 9,298	
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a jwhois-3.0.1 parser bc-1.06	LOC 2,213 4,460 6,174 7,327 9,344 10,900 13,093	#Basic-Blocks 782 2,634 2,757 6,271 5,147 9,298 4,924	
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a jwhois-3.0.1 parser bc-1.06 less-290	LOC 2,213 4,460 6,174 7,327 9,344 10,900 13,093 18,449	#Basic-Blocks 782 2,634 2,757 6,271 5,147 9,298 4,924 7,754	
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a jwhois-3.0.1 parser bc-1.06 less-290 twolf	LOC 2,213 4,460 6,174 7,327 9,344 10,900 13,093 18,449 19,700	#Basic-Blocks 782 2,634 2,757 6,271 6,271 5,147 9,298 4,924 7,754 14,610	
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a jwhois-3.0.1 parser bc-1.06 less-290 twolf tar-1.13	LOC 2,213 4,460 6,174 7,327 9,344 10,900 13,093 18,449 19,700 20,258	#Basic-Blocks 782 2,634 2,757 6,271 6,271 5,147 9,298 4,924 7,754 14,610 10.800	
Program spell-1.0 barcode-0.96 httptunnel-3.3 gzip-1.2.4a jwhois-3.0.1 parser bc-1.06 less-290 twolf tar-1.13 make-3.76.1	LOC 2,213 4,460 6,174 7,327 9,344 10,900 13,093 18,449 19,700 20,258 27,304	<pre>#Basic-Blocks 782 2,634 2,757 6,271 6,271 5,147 9,298 4,924 7,754 14,610 10,800 11,061</pre>	
	ret) b_{ret}) Performants $b_{V_{\hat{P}}} = 2^{P} \rightarrow b_{ret}$ $b_{V_{\hat{P}}} = 2^{P} \rightarrow b_{ret}$ $b_{i=10} \wedge b_{ret}$ hain as follows: remained Normal	$p_{\text{ret}}(p)$ $p_{\text{ret}}(p$	$b_{ret}(\beta) = no \qquad n \beta \neq n \beta$ $b_{ret}(\beta) = no \qquad n \beta \neq n \beta$ $p = p = p \qquad \rho \qquad \epsilon$ $b_{ret}(b_{i} = p \rightarrow k) \qquad \rho \qquad \epsilon$ $b_{ret}(b_{ret}(b_{i} = p \rightarrow k) \qquad \rho \qquad \epsilon$ $b_{ret}(b_{ret}(b_{i} = p \rightarrow k) \qquad \rho \qquad \epsilon$ $b_{ret}(b_{ret}(b_{i} = p \rightarrow k) \qquad \epsilon$ $b_{ret}(b_{ret}(b_{ret}(b_{i} = p \rightarrow k) \qquad \epsilon$ $b_{ret}(b_{ret}(b_{ret}(b_{i} = p \rightarrow k) \qquad \epsilon$ $b_{ret}(b_{ret}(b_{ret}(b_{i} = p \rightarrow k) \qquad \epsilon$ $b_{ret}(b_{re$

cs where from the Eollegistics an empty environment of the anventor propartionment. Then we compare whore gramminifighery are eccessive and an clude that the egenterated nclude that the generated eccession in other ison programment. Then we compare the sether are stated on the programment of the program may generate syntactically

cs is defined from the collecting set on where by any province in the set of the set of

e - (I - I0 / Iec) $EQ(\beta) = No$ if $\beta \not\equiv \lambda \land (\mu \models \beta \oplus$ **4.1** First Step Abstraction : Value $V_{\hat{P}} = 2^{\hat{P}} \rightarrow 2^{\hat{P}}$ First, we abstract the concrete parsing domain V_P to $i_{\hat{V}_{\hat{P}}} \stackrel{10}{=} 2^{P} (i = 10 \land \text{ret})$ for the code fragments x and λ The P. If f betract Parsing $p \in 4.1$ First Step A straction : also f = 2 by ϵ $\alpha = \lambda F. \lambda P. \bigcup \{ f(p) \mid f \in F \}$ $\sigma \in Env_{\hat{P}} = Var \to V_{\hat{P}}$ $= parse(p_{init}, y) \text{ not } parse(p_{g}, y).$ se stack after parsing y from the initial $Env_{\hat{P}}$ $V_{\hat{P}}$ $Y_{\hat{P}}$ $Y_{\hat{P}$ $Y_{\hat{P}}$ $Y_{\hat{P}}$ $Y_{\hat{P}}$ $Y_{\hat{P}}$ $Y_{\hat{P}$ $Y_{\hat{P}}$ $Y_{\hat{P$ to the parse stack transition function function **nsitivity** $E_{periode} = E_{nv_{\hat{p}}} = E_$ y. Then we have $\|e_1\|_{\hat{P}}^0 \sigma = \|e_2\|_{\hat{P}}^0 (\sigma[x \mapsto \|e_1\|_{\hat{P}}^0 \sigma])^1 \in Env_{\hat{P}} \to V_{\hat{P}}$ $p_{p, parse(p, x)}$ [or $e \mathbf{e}_2$] $\stackrel{\circ}{P} \mathbf{A}$ simple $a_{lgo}(x)$ that mitigates the $p.parse(p_{p}) = x e_{1} e_{2} e_{3} \Big]_{\hat{P}}^{0} \sigma = \llbracket e_{3} \Big]_{\hat{P}}^{0} (\sigma \llbracket x \vdash e_{1} e_{2} \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{2} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \mapsto \llbracket e_{1} \rrbracket_{\hat{P}}^{0} \sigma])$ isotion functions f_{x} and f_{y} , we have $f_{y} = [e_{3} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \vdash e_{1} e_{2} \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{2} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \mapsto \llbracket e_{1} \rrbracket_{\hat{P}}^{0} \sigma])$ isotion functions f_{x} and f_{y} , we have $f_{y} = [e_{3} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \vdash e_{1} \rrbracket_{\hat{P}}^{0} \sigma]) = \llbracket e_{2} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \vdash e_{2} \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{2} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \vdash e_{1} \rrbracket_{\hat{P}}^{0} \sigma]) = \llbracket e_{2} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \vdash e_{2} \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_{2} \rrbracket_{\hat{P}}^{0} (\sigma \llbracket x \vdash e_{2} \rrbracket_{\hat{P}}^{0} \sigma)] = [e_{2} \rrbracket_{\hat{P}}^{0} \sigma]$ on function $f_{x.y}$ as follows. $\llbracket f \rrbracket_{\hat{P}}^{0} \sigma = \llbracket f \rrbracket_{\hat{P}}^{1} \sigma \quad \llbracket re \ x \ e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_2 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_3 \ e_3 \ e_3 \ e_3 \rrbracket_{\hat{P}}^{0} \sigma = \llbracket e_3 \rrbracket_{\hat{P}}^{0} (\sigma [x \mapsto x e_1 \ e_3 \ e_3$ $\begin{array}{l} p, x.y) \\ parse(p, x), y) \\ (p, y)) \circ (\lambda p. parse(p, x)) \\ (p, y)) \circ (\lambda p. parse(p, x)) \\ f_{1}(p, y) = 0 \\ f_{2}(p, x) \\ f_{2}(p, x)$ RSS_k where $Parse_action : 2^{P} \rightarrow Token \rightarrow \mathbb{P}_{e^{-action}}^{P}$ is the average set of parse_action is the parse is the parse_action is the parse_ac anticsextension of *parse_action*: $pde \text{ to } P \rightarrow B$, the Galois connection AF (λt for second for the field second for the field second secon RSS_{k+1} established as follows. $\frac{\text{hished as tollows}}{\text{ide-wastract semantic function of the set of the$ (p, d) erated bedes conform to the gran and the for the provident provident stranger of the st 20 erated sodes conform to the grammar for the given program 82 we $p \in P.parse(p_{rc}) = f(p)$ 1810 $S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init}^{\text{vpr}} \} : 2^{P} \xrightarrow{8} 7 \xrightarrow{14.5} 8.9 \xrightarrow{8.9} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init}^{\text{vpr}} \} : 2^{P} \xrightarrow{8} 7 \xrightarrow{14.5} 8.9 \xrightarrow{8.9} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{8} 7 \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 8.9 \xrightarrow{11.8} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \sigma_{0} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \widehat{P} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \widehat{P} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \widehat{P} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \widehat{P} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \widehat{P} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \widehat{P} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket e \rrbracket_{\hat{P}}^{0} \widehat{P} \{ p_{init} \} : 2^{P} \xrightarrow{14.5} 11.8 \\ S = \llbracket$ 0.128 1.08 2.90.055cs where from the Eolle etils an empty isonment The any information ment. Then we compare where gramming they are eccession of the they also gramming they are eccession of the they also gramming the second second

Thank you