


# 정적 함수 호출 제거를 통한 분석 시간 개선

허진영

서울대학교 프로그래밍 연구실

# 동 기

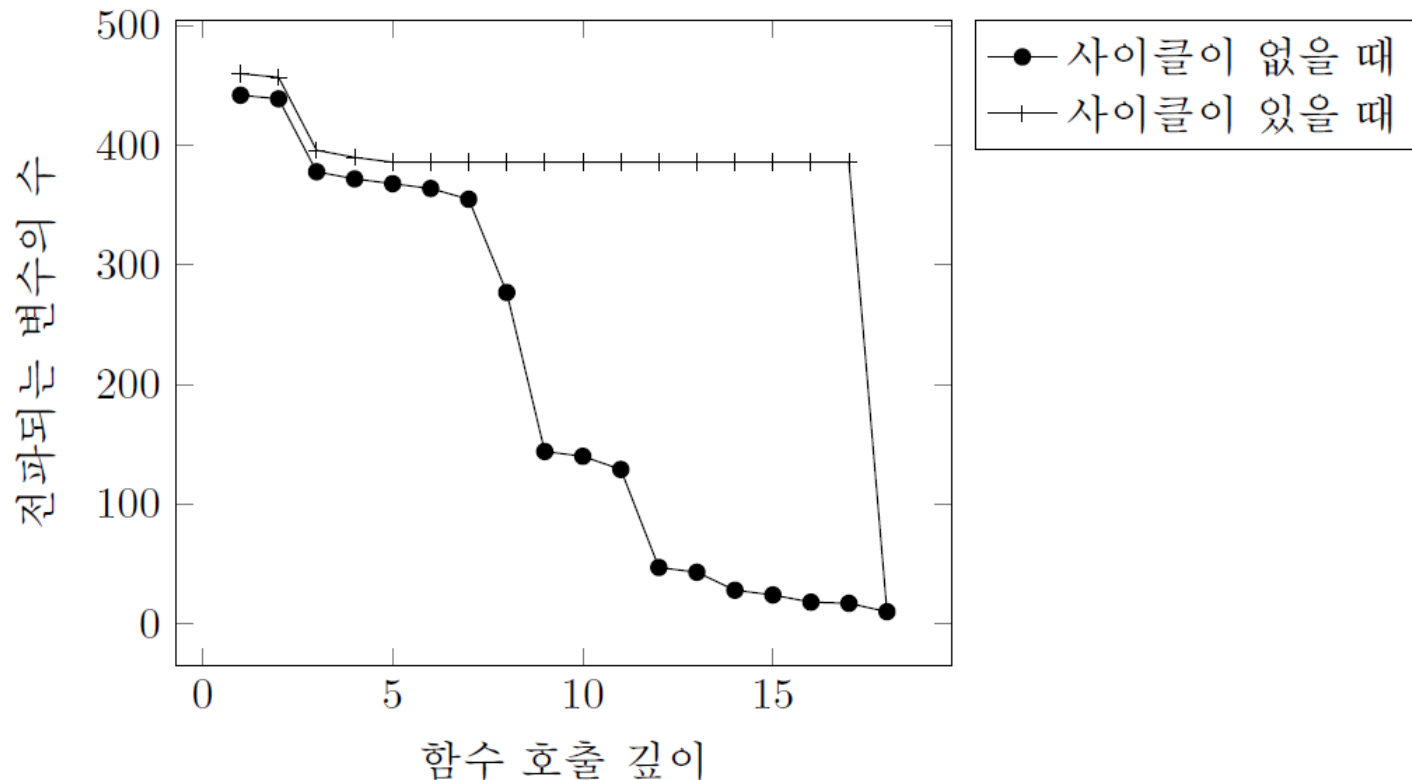
-  Sparrow  
The Early Bird  
– 요약 해석과 스파스 프레임워크에 기반  
– 100만 라인의 C 프로그램 분석
- 하지만, 일부 C 프로그램에서 성능 저하  
– 정적 함수 호출 그래프에 사이클이 있는 경우

# 분석 속도 비교

Programs	LOC	maxSCC	DU(s)	Main(s)	Total(s)
tar-1.13	20K	13	30	10	40
less-382	23K	46	87	38	125
nano-2.3.2	25K	36	145	56	201
make-3.76.1	27K	61	65	19	84
wget-1.9	35K	13	59	8	67
screen-4.0.2	45K	65	348	37	385
a2ps-4.14	64K	9	87	13	100
xboard-4.7.0	67K	89	620	320	940
mutt-1.4.2.3	87K	155	1,354	1,080	2,434
sendmail-8.13.6	130K	68	3,746	1,860	5,606
vim60	227K	1,306	25,412	45,879	71,291
bash-4.2	302K	435	4,746	1,680	6,426

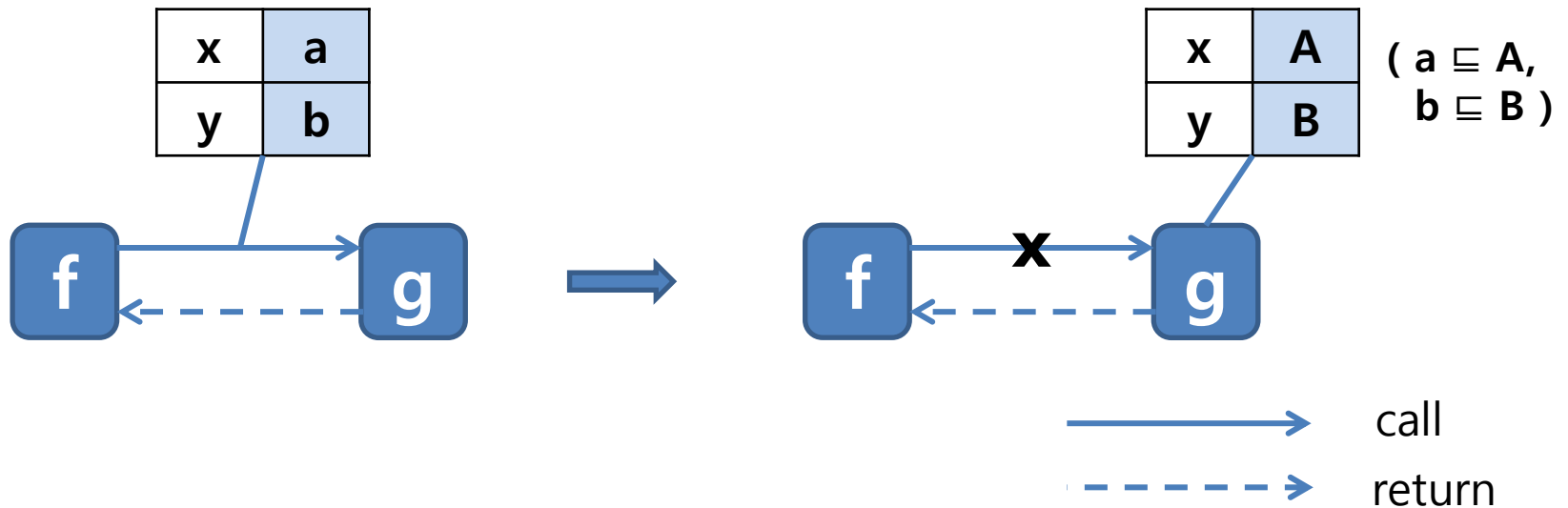
# 무엇이 문제인가?

- 함수 호출을 따라가면서 분석할 때 전달할 정보가 지나치게 많다.



# 개선 방안

- 모든 함수 호출 경로 제거 후 분석 진행
- 흐름을 고려하지 않는(flow-insensitive) 분석 결과 이용



# 실험 결과

프로그램	수정 전		수정 후		Speedup	알람 수(FI)
	분석시간	알람 수	분석시간	알람 수		
less-382	135	664	59	663	2.3 x	711
nano-2.3.2	201	2,464	93	2,447	2.2 x	2,574
make-3.76.1	84	1,556	43	1,767	1.9 x	2,113
screen-4.0.2	385	4,243	223	4,868	1.7 x	5,969
xboard-4.7.0	940	2,724	492	2,677	1.9 x	3,065
mutt-1.4.2.3	2,433	3,751	1,188	6,105	2.0 x	6,814
sendmail-8.13.6	5,606	6,156	2,779	6,665	2.0 x	7,457
vim60	71,291	13,499	21,425	13,572	3.3 x	14,324
bash-4.2	6,426	7,293	3,546	7,686	1.8 x	8,285

# 실험 결과 - 실제 알람

- sendmail의 일부 알려진 취약점\*에 대해 동일한 실험 수행

취약점	알람 수	수정 전	수정 후
CA-2003-07	28	28	28
CVE-1999-0131	3	2	2
CVE-1999-0206	3	3	3
CVE-1999-0047	10	10	10
CA-2003-12	3	3	3
CVE-2001-0653	1	1	1
CVE-2002-0906	2	2	2

\*Misha Zitser, Richard Lippmann, and Tim Leek. Testing static analysis tools using exploitable buffer overflows from open source code. SIGSOFT FSE, 2004

# 결론

- 전분석 결과를 이용해 함수 호출 제거 후 분석
  - 분석 시간 평균 2.1 배 단축
  - 알람 수는 기존과 비슷하거나 일부 증가