2023/1/17 01:35 README

运行指南

环境需求

需要Ubuntu系统并安装docker

(下面使用机器为Ubuntu20.04系统, 16GB内存, docker版本20.10.21)

docker容器内布局

```
|-- bench
   |-- *.bc
                      # Bitcode we will analyse.
    |-- bench.sh
                      # Benchmarking scripts -- run analyses
   I-- table.awk
                      #
                          and produce tables.
|-- gbe
                      # Example code we will produce analysable bitcode from.
                      # SVF source and build tree.
`-- svf
    |-- knl-build
                      # Build for Knights Landing microarchitecture.
                    # SVF binary (similarly for the following two builds).
       |-- bin/wpa
   |-- haswell-build # Build for Haswell microarchitecture.
    |-- base-build
                      # More portable AMD64/x86-64 build.
    `-- * (remainder) # SVF source.
```

如何运行

- 1.到原文地址下载文件 compacting-points-to-sets.tar.gz
- 2.使用docker运行,在文件目录下打开终端,先后运行如下指令
- (1) docker load < compacting-points-to-sets.tar.gz</pre>
- (2) docker run -it compacting-points-to-sets bash

如图:

ah@Nitro:~\$ docker load < compacting-points-to-sets.tar.gz Loaded image: compacting-points-to-sets:latest

```
ah@Nitro:~$ docker run -it compacting-points-to-sets bash root@62b3b6aa8da7:~#
```

尝试如下命令可以运行一个简单的例子:

- (1) cd \$HOME/bench
- (2) ./bench.sh \$HOME/svf/base-build/bin/wpa 1 1 1 dhcpcd.bc

如图开始运行

```
root@62b3b6aa8da7:~# cd $HOME/bench
root@62b3b6aa8da7:~/bench# ./bench.sh $HOME/svf/base-build/bin/wpa 1 1 1 dhcpcd
bc
==== Clustering Benchmark =====
 = args =
    # runs
                  : /root/svf/base-build/bin/wpa
    svf binary
    time limit
                  : 1 hours
    memory limit : 1 gigabytes
    bitcode files : dhcpcd.bc (1)
    raw data dir : craw_1673887641
  = start =
    = run #1 of 1 =
     = dhcpcd.bc (file #1 of 1) =
        = running: /root/svf/base-build/bin/wpa -fspta -fs-time-limit=3600 -op
t-svfg=false -marked-clocks-only -node-alloc-strat=dense -staged-pt-type=sbv d
hcpcd.bc
```

指令(2)拆解:

./bench.sh SVF_BIN NUM_RUNS TIME_LIMIT MEM_LIMIT BITCODE...

SVF_BIN默认 \$HOME/svf/base-build/bin/wpa 即可

NUM_RUNS为执行分析的轮次

TIME_LIMIT为运行时间限制,以小时为单位,超出时间会跑出OOT(超时)的结果

MEM_LIMIT为内存限制,以GB为单位,超出时间会跑出OOM(超内存)的结果

BITCODE...为要运行的文件列表,如我要跑dhcpcd.bc和gawk.bc文件,则可以是 ./bench.sh \$HOME/svf/base-build/bin/wpa 1 1 1 dhcpcd.bc gawk.bc

2023/1/17 01:35 README

想要跑出论文中类似的结果, 我们使用了如下指令

./bench.sh \$HOME/svf/base-build/bin/wpa 1 2 15 dhcpcd.bc gawk.bc keepassxc.bc lynx.bc mutt.bc ruby.bc table.awk time.txt tmp.txt xpdf. 因为我们仅使用了15GB内存,所以很多结果OOM了,如下:

Benchmark	Theoretical	Orginal	Single	Complete	Average	Reduction
dhcpcd.bc	3317195	24726024	*4991412*	6635746	6635082	4.95
gawk.bc	58007460	429843180	*82989102*	132528508	99502900	5.18
bash.bc	26586881	289532162	*42914256*	42914700	53173774	6.75
mutt.bc	51298142	490532984	*102662924*	145767658	160026830	4.78
lynx.bc	133664618	965029716	267599228	319144056	*215831960*	4.47
xpdf.bc				1000		7.7
and the same of th				**		
ruby.bc					74381880	5.81
	13770856	315331336	*54312908*	74407698	/4301000	3.01
eepassxc.bc	13770856	315331336	*54312908*	74407098	Geo. mean	5.27
eepassxc.bc		315331336 Orginal	*54312908* Single	Complete	(大学)(大学)	5.27
eepassxc.bc ABLE 5: Requir	ed words for CBV Theoretical	Orginal	Single	Complete	Geo. mean Average	5.27
eepassxc.bc ABLE 5: Requir Benchmark dhcpcd.bc	ed words for CBV				Geo. mean	5.27 Reductio
eepassxc.bc ABLE 5: Requir Benchmark	ed words for CBV Theoretical 3317195	Orginal 23911464	Single *4961417*	Complete 6605816	Geo. mean Average 5784923	5.27 Reductio 4.82 5.19
eepassxc.bc ABLE 5: Requir Benchmark dhcpcd.bc gawk.bc	ed words for CBV Theoretical 3317195 58007460	Orginal 23911464 429739626	Single *4961417* *82783110*	Complete 6605816 140588641	Average 5784023 148836214	5.27 Reductio 4.82 5.19 9.30
ABLE 5: Requir Benchmark dhcpcd.bc gawk.bc bash.bc	ed words for CBV Theoretical 3317195 58007460 26586881	Orginal 23911464 429739626 295168815	Single *4961417* *82783110* *31731607*	Complete 6605816 140588641 36861568	Average 5784923 148836214 47120912	12/52/67
ABLE 5: Requir Benchmark dhcpcd.bc gawk.bc bash.bc mutt.bc lynx.bc xpdf.bc	ed words for CBV Theoretical 3317195 58007460 26586881 51298142	Orginal 23911464 429739626 295168815 548971337	Single *4961417* *82783110* *31731607* *87213543*	Complete 6605816 140588641 36861568 260457927	Average 5784023 148836214 47120912 259746461	5.27 Reductio 4.82 5.19 9.30 6.29
ABLE 5: Requir Benchmark dhcpcd.bc gawk.bc bash.bc mutt.bc lynx.bc	ed words for CBV Theoretical 3317195 58007460 2058081 51298142 133664618	Orginal 23911464 429739626 295168815 548971337 1015676938	Single *4961417* *82783110* *31731607* *87213543* *237113529*	Complete 6605816 140588641 36861568 260457927 289849510	Average 5784023 148836214 47120912 259746461 302122259	5.27 Reductio 4.82 5.19 9.30 6.29 4.28

Unclustered			Clustered		CDV/CDV	PV/CPV
SBV	BV [s]	CBV [s]	SBV	CBV	2BA\2BA	BV/CBV
54.85	51.03	63.92	46.13	46.74	1.19x	1.09
879.08	732.79	731.83	624.72	601.62	1.41x	1.22
204.30	165.37	166.57	140.13	140.62	1.46x	1.18
496.33	438.60	427.53	370.20	359.54	1.34x	1.22
MOO	MOO	MOO	1532.27	1560.07	++X	
MOO	MOO	MOO	MOO	MOO	x	
MOO	MOO	MOO	MOO	MOO	x	
688.88	MOO	MOO	515.54	511.04	1.34x	
				Geo. mean	1.34x	1.18
ory						
	[5] 54.85 879.08 204.30 496.33 00M 00M 00M 688.88	[5] [5] 54.85 51.03 879.08 732.79 204.30 165.37 496.33 438.60 00M 00M 00M 00M 00M 00M 00M 00M 00M 0	[S]	[5] [6] 13 24 13 24 13 24 13 24 12 13 370 20 </td <td>[s] [s] [s] [s] 54.85 51.03 63.92 46.13 46.74 879.08 732.79 731.83 624.72 601.62 204.30 165.37 166.57 140.13 140.62 496.33 438.60 427.53 370.20 359.54 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 688.88 00M 00M 515.54 511.04 Geo. mean</td> <td>[s] [s] [s] [s] 54.85 51.03 63.92 46.13 46.74 1.19x 879.08 732.79 731.83 624.72 601.62 1.41x 204.30 165.37 166.57 140.13 140.62 1.46x 496.33 438.60 427.53 370.20 359.54 1.34x 00M 00M 00M 00M 00M x 00M 00M 00M 00M x 00M 00M 00M 00M x 688.88 00M 00M 515.54 511.04 1.34x Geo. mean 1.34x</td>	[s] [s] [s] [s] 54.85 51.03 63.92 46.13 46.74 879.08 732.79 731.83 624.72 601.62 204.30 165.37 166.57 140.13 140.62 496.33 438.60 427.53 370.20 359.54 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 00M 688.88 00M 00M 515.54 511.04 Geo. mean	[s] [s] [s] [s] 54.85 51.03 63.92 46.13 46.74 1.19x 879.08 732.79 731.83 624.72 601.62 1.41x 204.30 165.37 166.57 140.13 140.62 1.46x 496.33 438.60 427.53 370.20 359.54 1.34x 00M 00M 00M 00M 00M x 00M 00M 00M 00M x 00M 00M 00M 00M x 688.88 00M 00M 515.54 511.04 1.34x Geo. mean 1.34x

Benchmark	Unclustered			Clustered		cov/cov	BV/CBV
вепсинат к	SBV [GB]	BV [GB]	CBV [GB]	SBV [GB]	CBV [GB]	SBV/SBV	ву/сву
dhcpcd.bc	1.20	0.92	0.91	0.74	0.68	1.62x	1.34
gawk.bc	12.76	8.00	7.79	4.63	3.67	2.75x	2.18
bash.bc	9.00	4.93	5.06	3.23	2.66	2.79x	1.85
mutt.bc	14.28	11.67	11.45	5.47	4.56	2.61x	2.56
lynx.bc	MOO	MOO	MOO	11.52	9.37	>=1.30x	>=1.60)
xpdf.bc	MOO	MOO	MOO	MOO	MOO	x)
ruby.bc	MOO	MOO	MOO	MOO	MOO	x	>
keepassxc.bc	12.41	MOO	MOO	6.30	6.21	1.97x	>=2.41)
					Geo. mean	>=2.09x	>=1.94>

原论文使用了100GB的机器,所以能跑出来的结果更多。