NC State University

Department of Electrical and Computer Engineering

ECE 466/566 Spring 2022

Project #3: Loop Invariant Code Motion

by

Himwantsai Satyabhargava Alapati

(halapat@ncsu.edu)

NCSU Honor Pledge: "I have neither given nor received unauthorized aid on this project."

Student's electronic signature: Himwantsai Satyabhargava Alapati

Course number: 566

1. Describe your implementation of LICM in pseudocode including any bonus optimizations. Also, give pertinent details about any statistical counters you collect and report.

Ans: The Implementation of LICM follows the pseudo code provided in the project document, some of the APIs are used from dominance.c and loop.c. There weren't any extra optimizations performed.

Below are the stats for the counters that are collected using stats.py for 3 different settings i.e.,

MLICM: (Mem2Reg and LICM Passes), MCLICM: (Mem2Reg, CSE and LICM Passes) and LICM.

COUNTERS	**LICM**	**MLICM**	**MCLICM**
LICMBasic	826	4292	2402
Functions	1900	1900	1900
Stores	24955	6383	6346
NumLoops	1309	1309	1309
LICMLoadHoist	442	58	56
NumLoopsNoStoreWithLoad	7	725	726
NumLoopsNoStore	7	782	783
NumLoopsWithCall	767	767	767
Loads	66941	18512	16444
NumLoopsNoload	1309	1309	1309
Instructions	200751	128290	115250

2. How many instructions were moved using your implementation of Loop Invariant Code Motion?

Ans: LICMBasic and LICMLoad Hoist counters were used to keep the track of the Instructions (not load and store) that are loop Invariant and Load Instructions which can be moved to preheader respectively. From the above stats in Q1 the Instructions that were moved using LICM pass are 1268 (826+442).

3. How many instructions were moved per loop on average? What happens to the average when you precede LICM with other optimizations, like mem-to-reg and CSE?

Ans: Total Number of loops observed during the testing are 1309.
So, the average Instructions moved per loop using LICM pass is **0.97** (1268 / 1309).
When we use the Mem2Reg and LICM optimization the average is **3.3** (4350 / 1309).
Upon using the Mem2Reg, CSE and LCIM optimization together the average is **1.87** (2458 / 1309).

As observed the average is best during when Mem2Reg and LICM optimizations are performed together.

4. 566 only or 466 bonus: How many load instructions were moved versus non-load instructions?

Ans:

→ 442 load Instructions and 826 non-load and non- store instructions were moved using LICM pass.
→ 58 Load Instructions and 4292 non-load and non-store instructions were moved using Mem2Reg and LICM optimizations.
→ 56 Load Instructions and 2402 non-load and non-store instructions were moved using Mem2Reg, CSE and LICM optimizations.

Below are the stats collected using timing.py and using fullstats.py.

	Full Stats: Instructions		
Benchmarks	**LICM**	**MLICM**	**MCLICM**
adpcm	419	249	239
arm	787	434	376
basicmath	594	353	314
bh	3285	2043	1859
bitcount	665	444	423
crc32	145	83	83
dijkstra	322	235	216
em3d	1233	697	625
fft	742	455	390
hanoi	96	53	51
hello	4	2	2
kmp	559	383	324
l2lat	97	61	60
patricia	1079	741	616
qsort	148	103	92
sha	661	425	375
smatrix	316	236	199
sql	176969	113493	102763
susan	12630	7800	6243

Timing		
LICM	**MLICM**	**MCLICM**
1.81	0.44	0.44
0	0	0
0.08	0.07	0.08
0.98	0.47	0.41
0.15	0.04	0.04
0.12	0.08	0.06
0.04	0.05	0.04
0.33	0.23	0.24
0.05	0.04	0.04
1.58	1.36	1.32
-	-	-
0.13	0.13	0.14
0.01	0.01	0.01
0.05	0.11	0.06
0.02	0.03	0.03
0.02	0.01	0.01
3.83	3.8	3.78
0	0	0
0.67	0.15	0.15