Verybench Complexity Report

Verybench 2.1.1.0 Report

C/C++ Code Complexity

idelatorre, 25/04/2014 12:06:18

Table of Contents

1. Snapshot Summaries

1.1 Project and Snapshot Info

Project Info

Project Title:	bitwalker
Project Author:	idelatorre
Project Type:	C/C++ Metrics
Project Location:	C:\Users\idelatorre.BIO-SQS\Documents

Snapshot Info

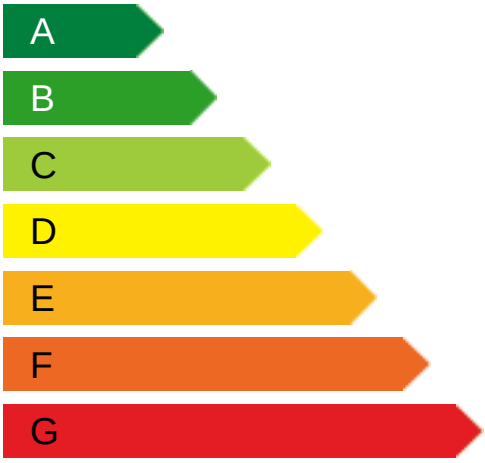
Latest Snapshot:	Bitwalker_snapshot
Snapshot Author:	idelatorre
Analysis Date:	25/04/2014 10:16:33
Source Files:	1
MI Preference:	MI
Notice Leading Comments:	Yes
Assembly File Extensions:	asm,as,s,uc
Assembly Comment Character:	;
Assembly Id Addon Characters:	.@
B Correction Factor:	1
No Comment Warnings Below:	1

File Scope Alarm Limits	Lower Limit	Higher Limit
c% (Comment Ratio)	30	75
ECC (McCabe Cyclomatic Complexity)	1	100
V (Program Volume)	100	8000
LOCpro (Program Lines of Code)	4	400
B (Estimated Number of Bugs)	0	2
MI (Maintainability Index)	65	171

Function Scope Alarm Limits	Lower Limit	Higher Limit
c% (Comment Ratio)	30	75
v(G) (McCabe Cyclomatic Complexity)	1	15
V (Program Volume)	20	1000
LOCpro (Program Lines of Code)	4	40
Preferred Maintainability Index	65	171

1.2 System Scope Summary

Code Quality Rating



	(0-5)	
7 >	(6-10)	B: Code could be better but still OK.
	(11-15)	
	(16-20)	
	(21-25)	
	(26-30)	
	(31-41)	

System Scope Metrics

Files	Functions	LOCphy	LOCpro	LOCcom	LOCbl
1	7	109	58	33	23

Semicolons	ECC	MI	MIcw	MIwoc	c%
30	7	135	31	104	30%

File Scope Alarms

B	V	LOCpro	c%	ECC	MI	**a%**
0	0	0	0	0	0	**0 (0%)**

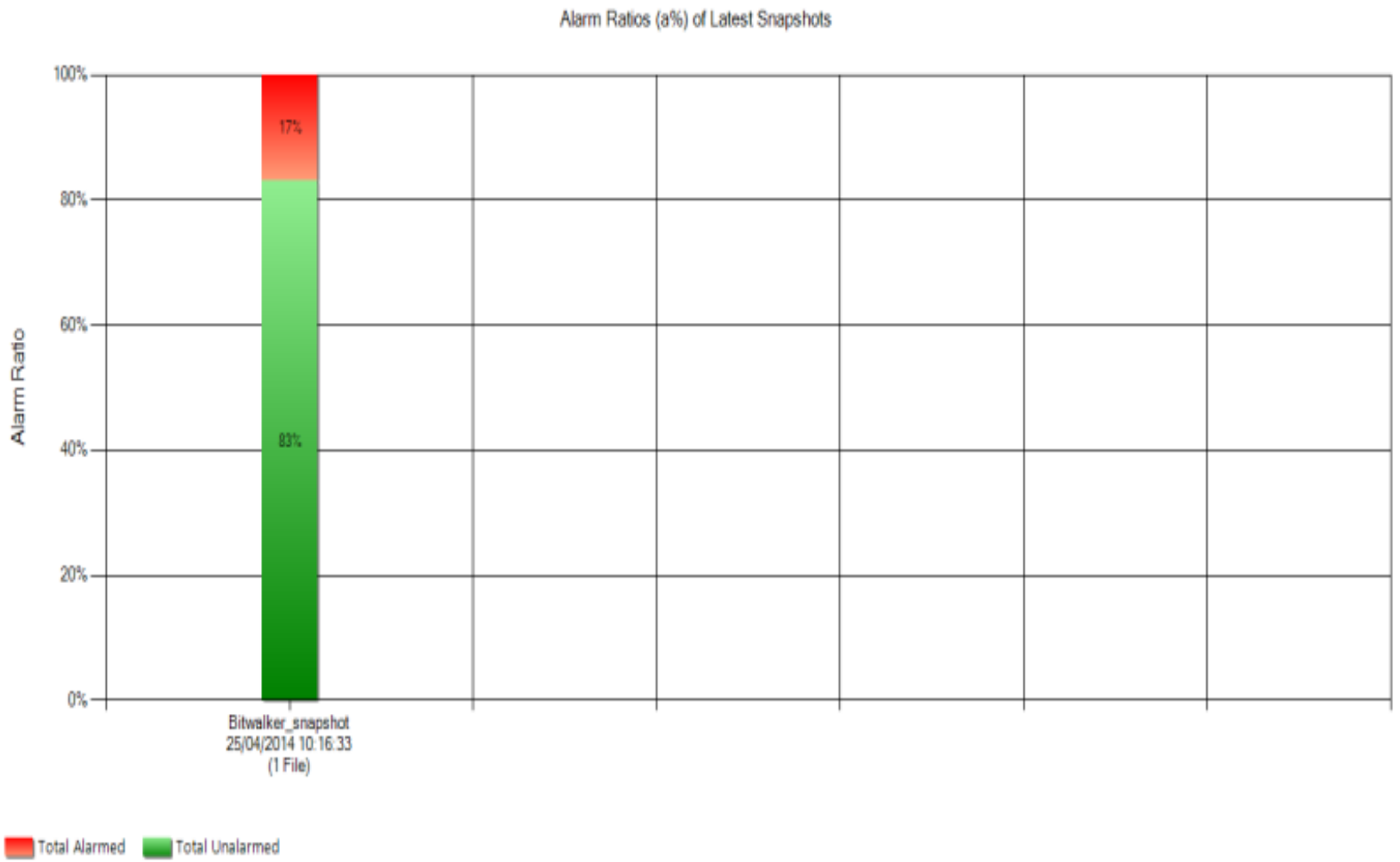
Max. Alarms: 6

Function Scope Alarms

LOCpro	V	c%	ECC	MI	**a%**
0	0	7	0	0	**7 (20%)**

Max. Alarms: 35

1.3 Snapshots Overview



2. File Scope Summaries

2.1 File Scope Metrics (Tables)

Alarming Metrics in File Scope

File Name	Functions	Alarms	a%	c%	LOCpro	V	B	ECC	MI
Bitwalker.c	7	0	0%	30%	58	2332.232	1.024	7	134

Non-Alarming Metrics in File Scope

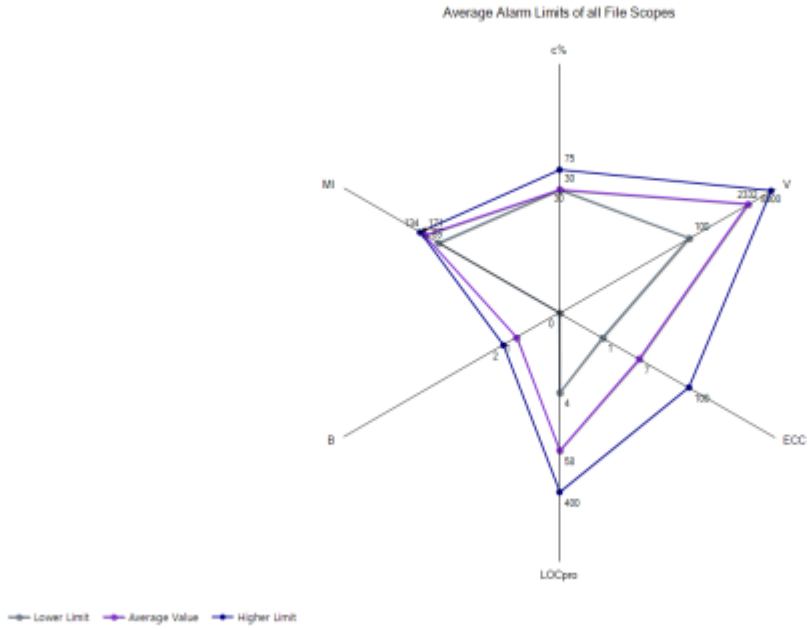
File Name	LOCbl	LOCphy	LOCcom	MaxND	MIwoc	MIcw	E	T	D	L
Bitwalker.c	23	109	33	2	96	38	170167.578	02:37:33	72.963	0.014

Non-Alarming Metrics in File Scope

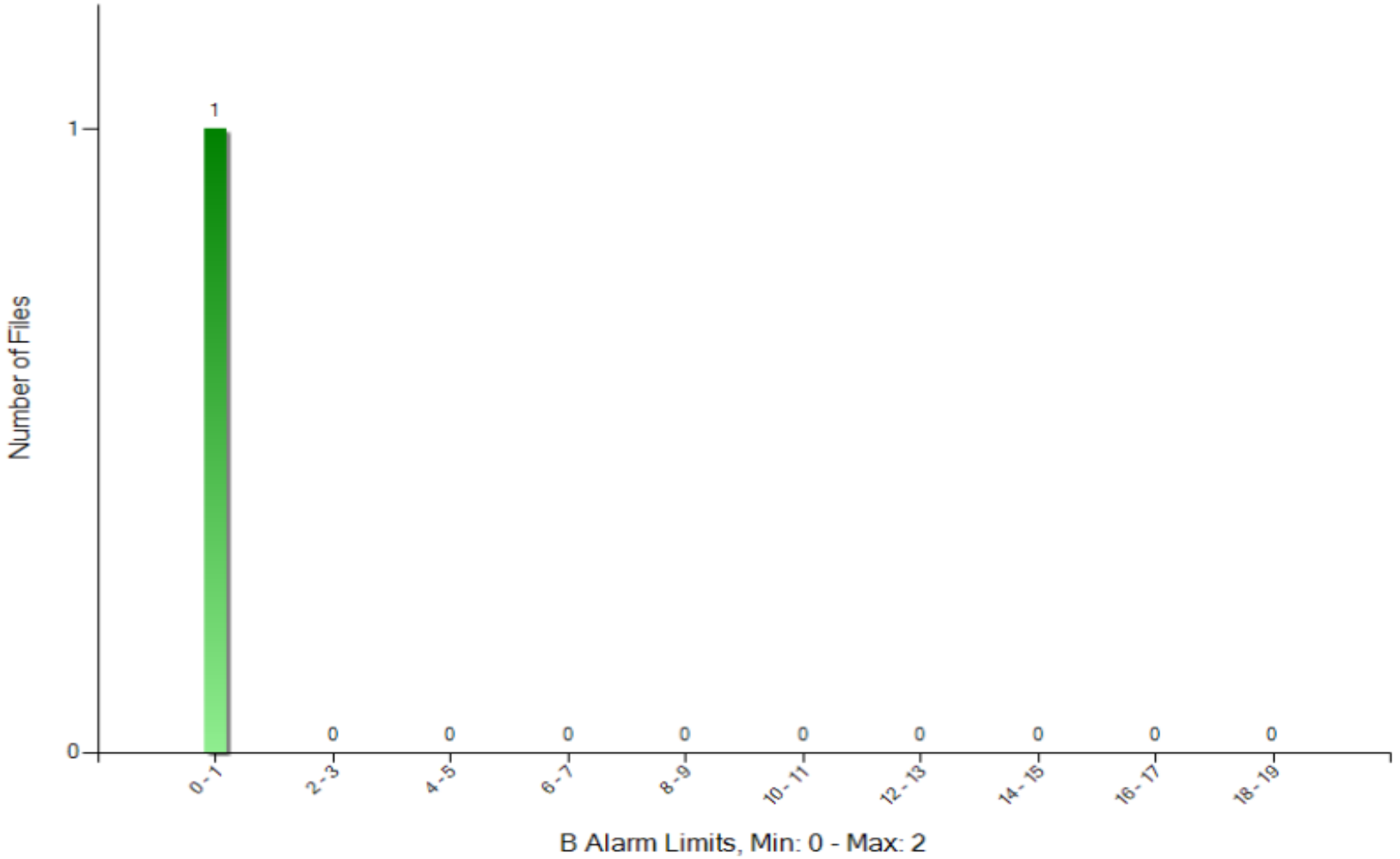
File Name	n	n1	n2	N	N1	N2
Bitwalker.c	72	31	41	378	185	193

2.2 File Scope Metrics (Charts)

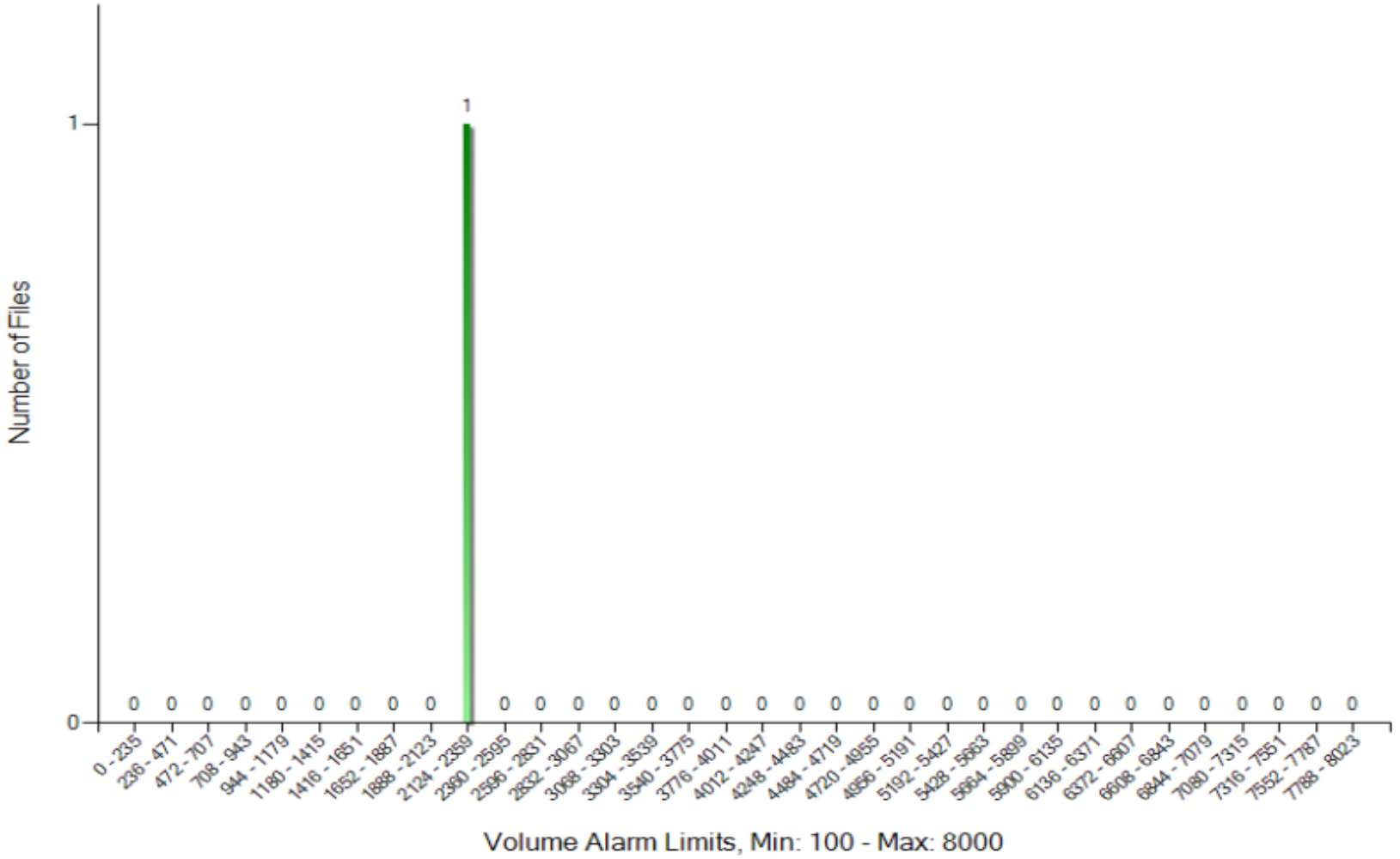
2.2.1 Alarm Limits Overview (File Scope)



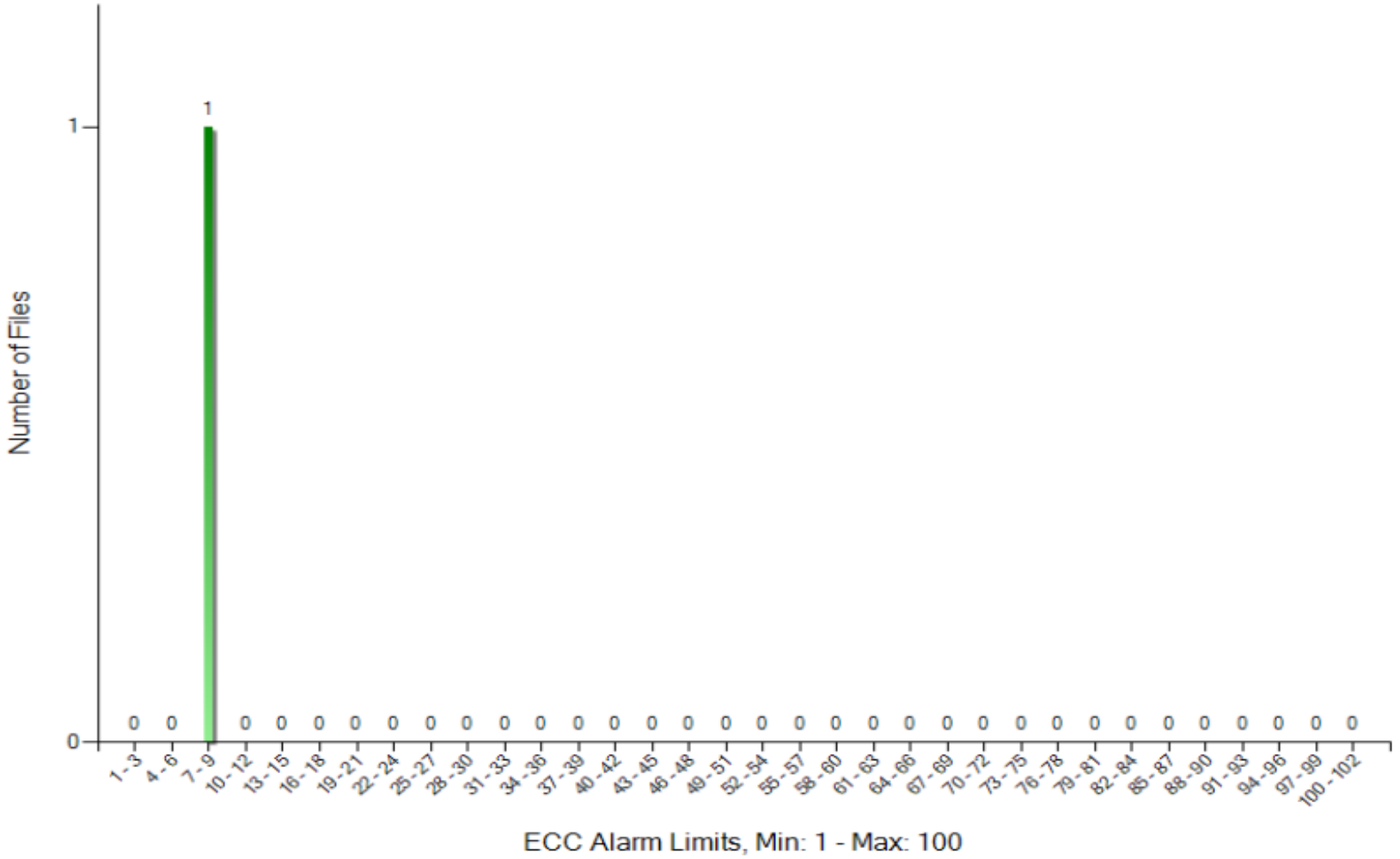
2.2.2 File Distribution to B (Estimated Number of Bugs)



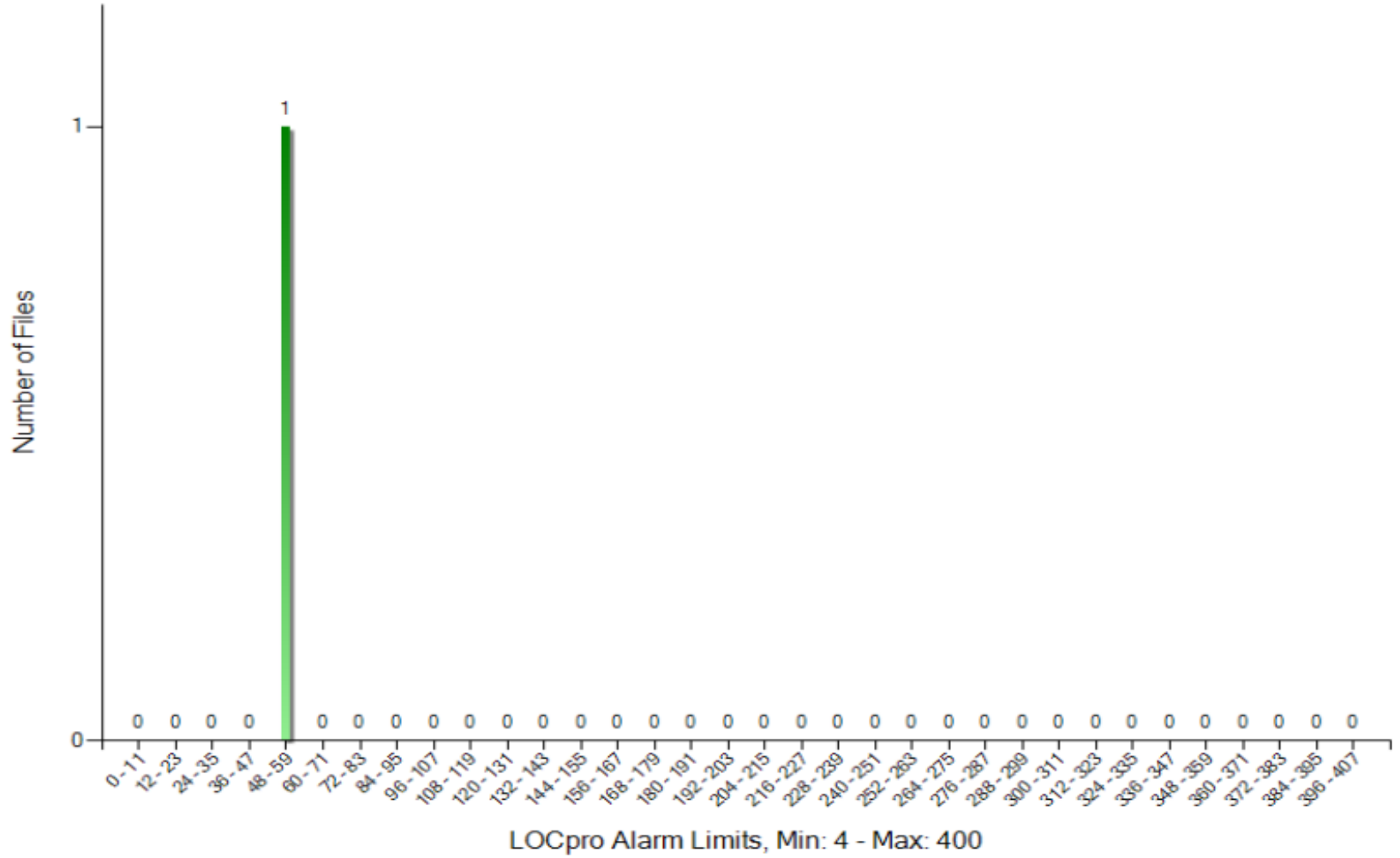
2.2.3 File Distribution to V (Program Volume)



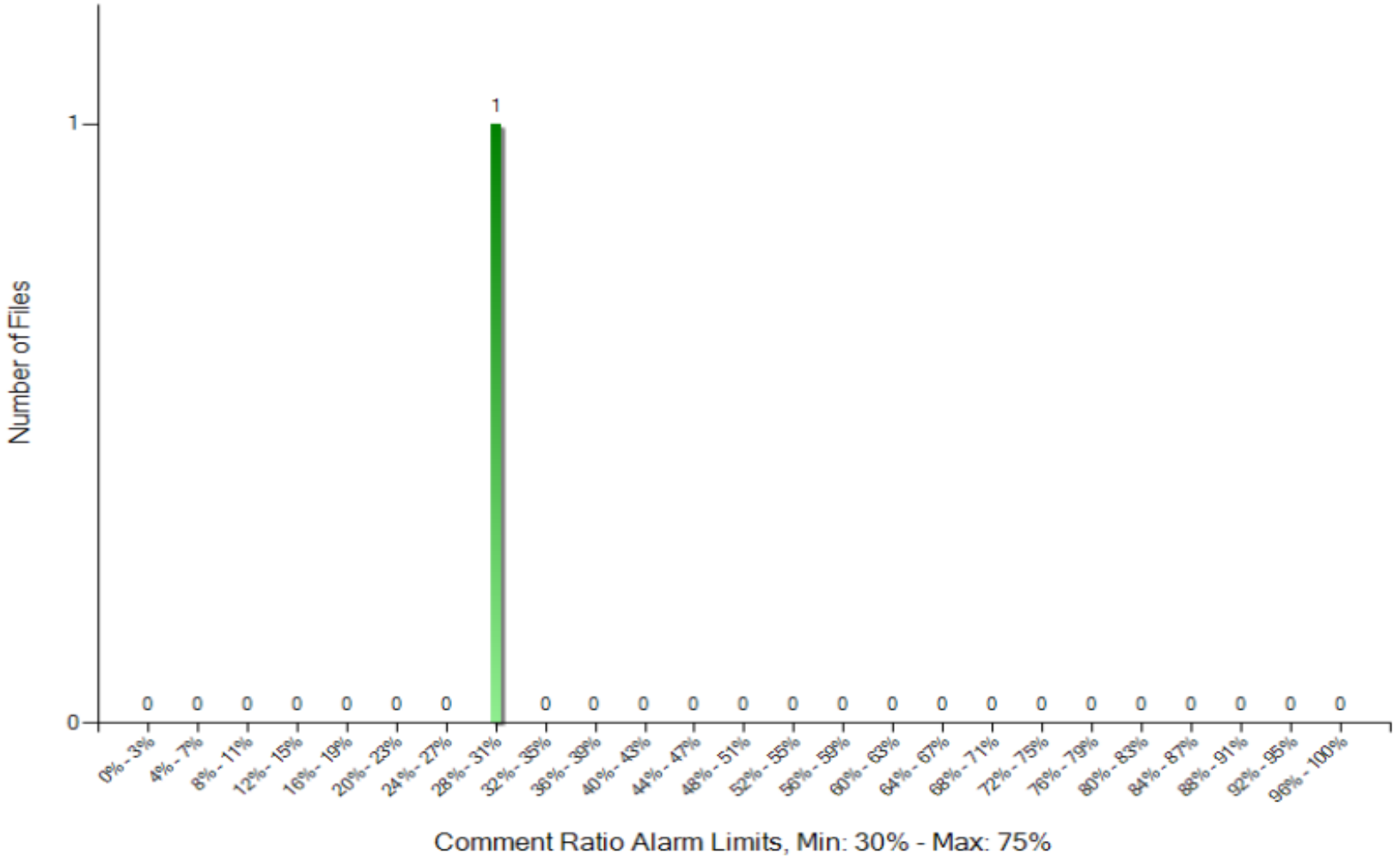
2.2.4 File Distribution to ECC (McCabe Cyclomatic Complexity)



2.2.5 File Distribution to LOCpro (Program Lines of Code)



2.2.6 File Distribution to c% (Comment Ratio)



3. Function Scope Summaries

3.1 Function Scope Metrics (Tables)

Alarming Metrics in Function Scope

Function Name	Alarms	a%	c%	LOCpro	V	ECC	MI
Bitwalker_Peek()	1	20%	25%	13	456.506	3	125
Bitwalker_Poke()	1	20%	25%	17	647.078	5	120
Bitwalker_IncrementalWalker_Init()	1	20%	0%	6	159.911	1	115
Bitwalker_IncrementalWalker_Peek_Next()	1	20%	14%	6	168.555	1	140
Bitwalker_IncrementalWalker_Peek_Finish()	1	20%	0%	4	41.513	1	129
Bitwalker_IncrementalWalker_Poke_Next()	1	20%	14%	6	193.262	1	140
Bitwalker_IncrementalWalker_Poke_Finish()	1	20%	0%	4	41.513	1	129

Non-Alarming Metrics in Function Scope

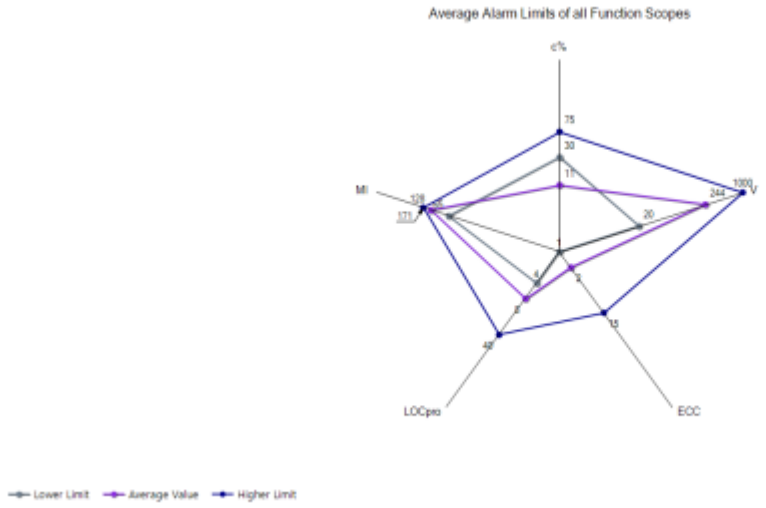
Function Name	LOCbl	LOCphy	LOCcom	MaxND	MIwoc	MIcw	B	E	T	D
Bitwalker_Peek()	4	20	5	2	90	35	0.166	11117.269	00:10:17	24.353
Bitwalker_Poke()	4	24	6	2	85	35	0.267	22715.846	00:21:01	35.105
Bitwalker_IncrementalWalker_Init()	0	6	0	1	115	0	0.036	1119.379	00:01:02	7.000
Bitwalker_IncrementalWalker_Peek_Next()	0	7	1	1	113	28	0.043	1448.042	00:01:20	8.591
Bitwalker_IncrementalWalker_Peek_Finish()	0	4	0	1	129	0	0.009	149.447	00:00:08	3.600
Bitwalker_IncrementalWalker_Poke_Next()	0	7	1	1	112	28	0.048	1739.358	00:01:36	9.000
Bitwalker_IncrementalWalker_Poke_Finish()	0	4	0	1	129	0	0.009	149.447	00:00:08	3.600

Non-Alarming Metrics in Function Scope

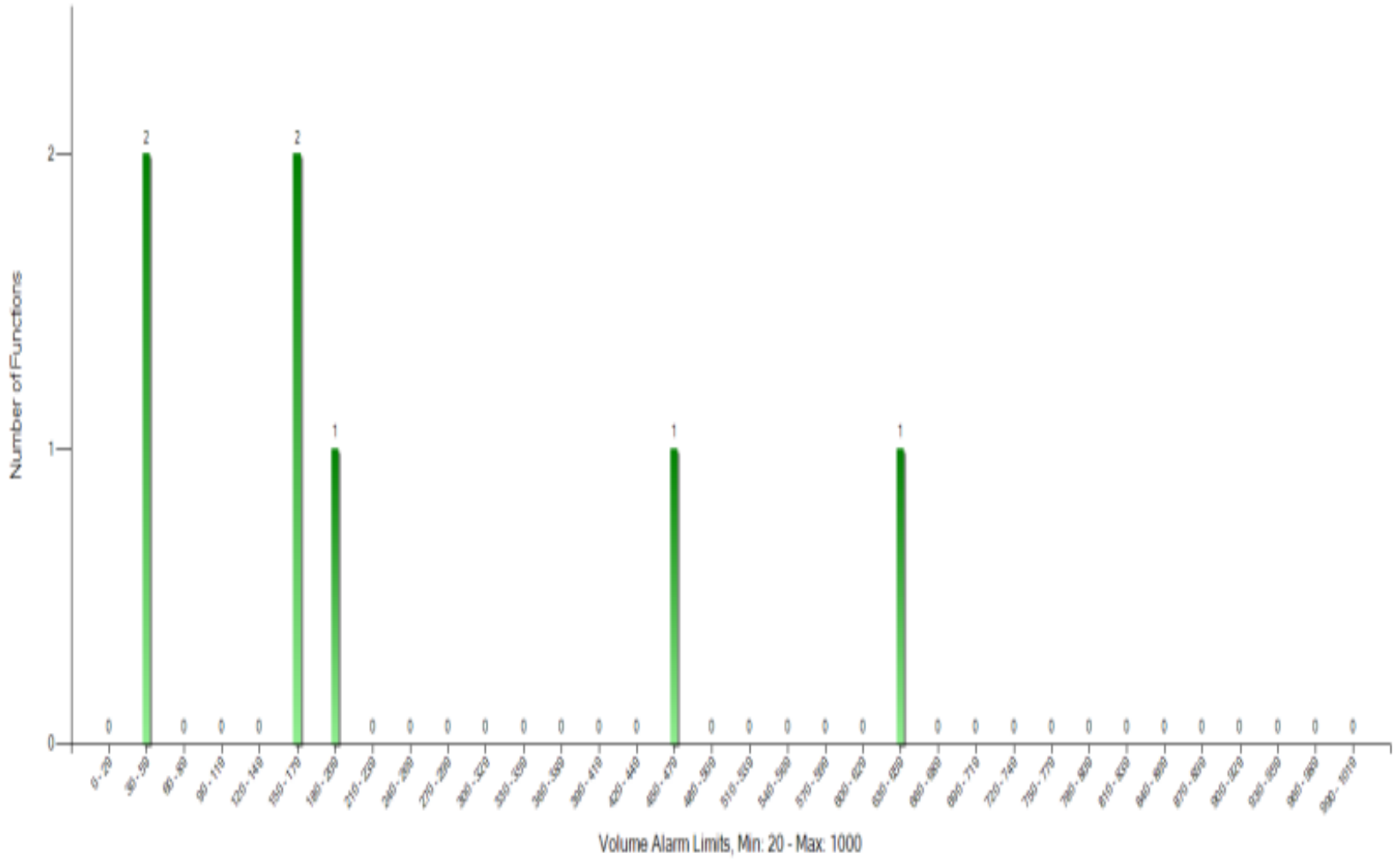
File Name	L	n	n1	n2	N	N1	N2
Bitwalker_Peek()	0.041	35	18	17	89	43	46
Bitwalker_Poke()	0.028	42	23	19	120	62	58
Bitwalker_IncrementalWalker_Init()	0.143	20	8	12	37	16	21
Bitwalker_IncrementalWalker_Peek_Next()	0.116	20	9	11	39	18	21
Bitwalker_IncrementalWalker_Peek_Finish()	0.278	11	6	5	12	6	6
Bitwalker_IncrementalWalker_Poke_Next()	0.111	21	9	12	44	20	24
Bitwalker_IncrementalWalker_Poke_Finish()	0.278	11	6	5	12	6	6

3.2 Function Scope Metrics (Charts)

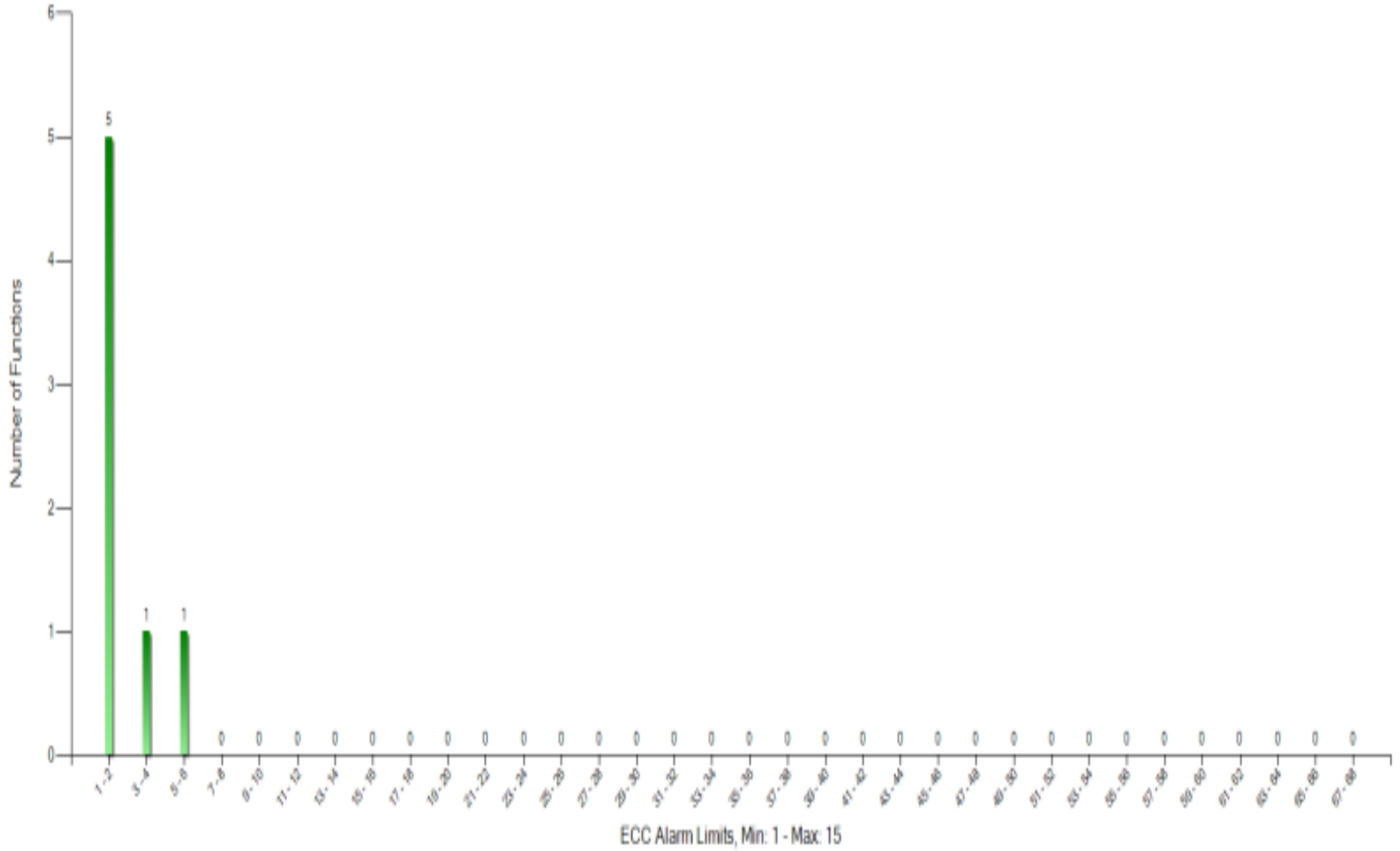
3.2.1 Alarm Limits Overview (Function Scope)



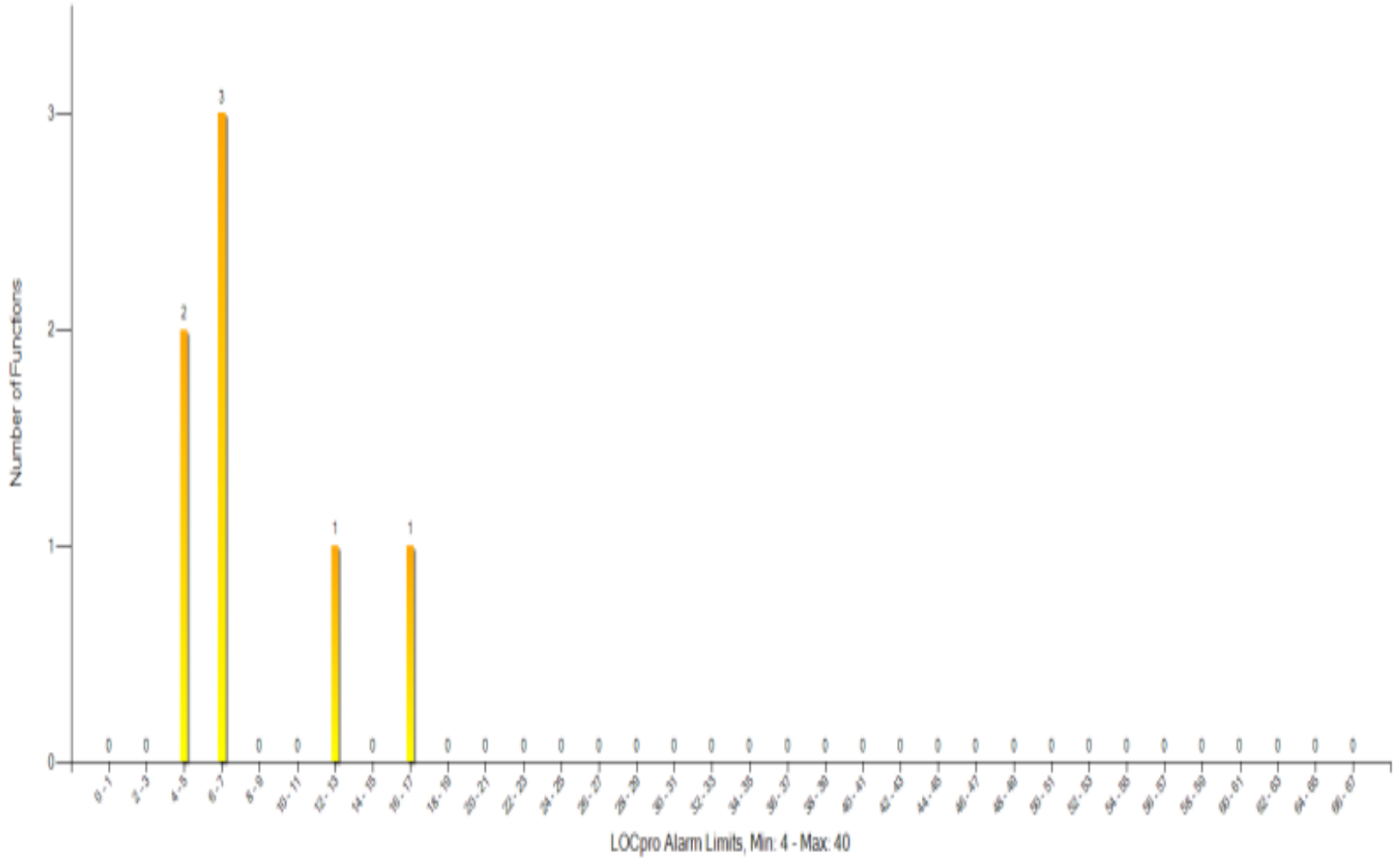
3.2.2 Function Distribution to V (Program Volume)



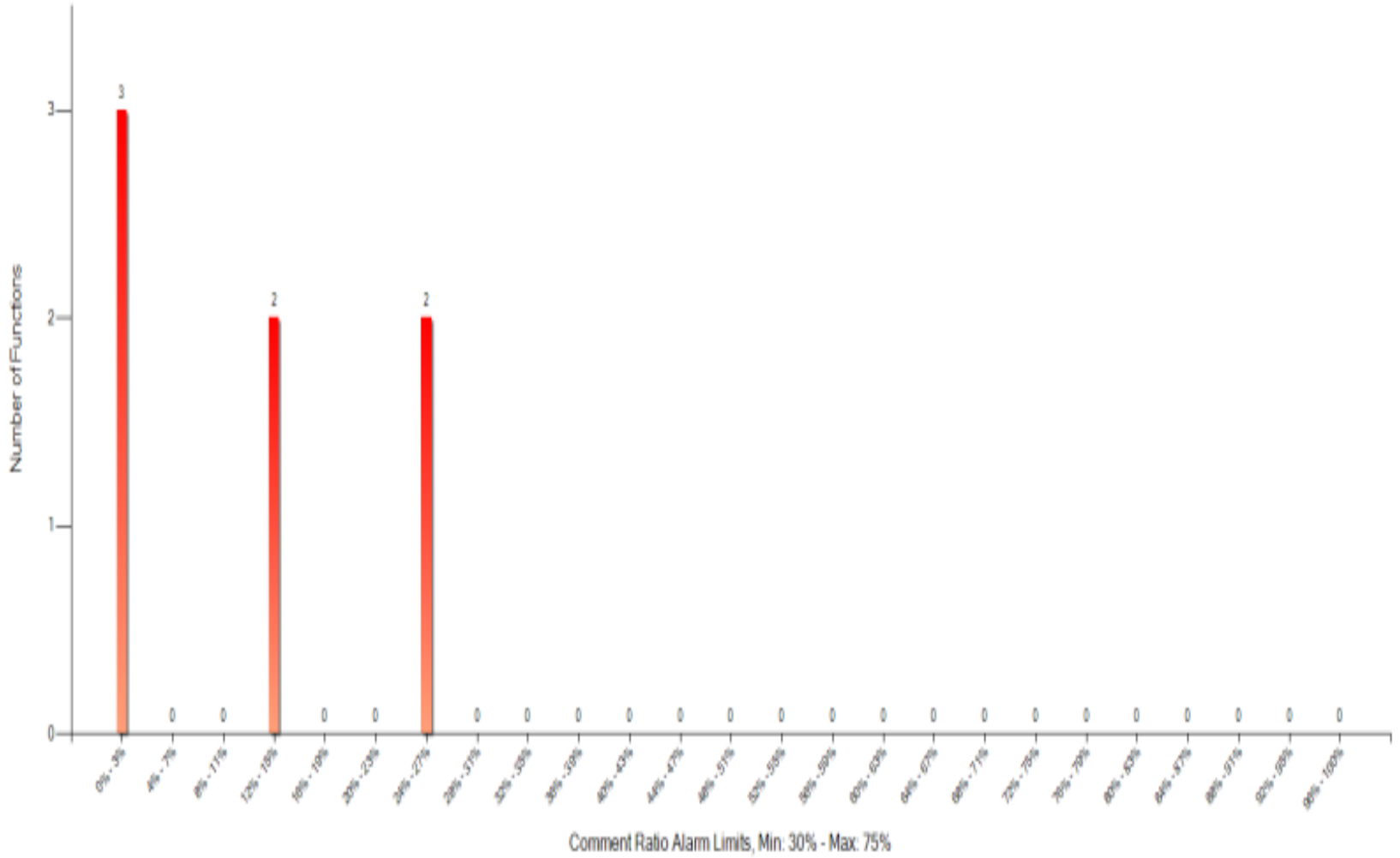
3.2.3 Function Distribution to ECC (McCabe Cyclomatic Complexity)



3.2.4 Function Distribution to LOCpro (Program Lines of Code)



3.2.5 Function Distribution to c% (Comment Ratio)



4. Appendix

4.1 Tool Info

This report was created with Verifysoft Verybench 2.1.1.0.
(Graphical Add-on for Testwell CMT++ 5.0)

4.2 Metrics Glossary

McCabe Cyclomatic Complexity

ECC	Extended complexity of a program's control flow.

Lines of Code

LOCphy	Physical lines of source code
LOCpro	Program lines of source code (these lines may also contain comments)
LOCcom	Comment lines of source code (these lines may also contain program code)
LOCbl	Blank lines of source code

Maintainability Indexes

MI	Maintainability Index
MIwoc	Maintainability Index without Comments
MIcw	Maintainability Index with Weighted Comments

Halstead

B	Estimated Number of Bugs
D	Difficulty Level (Error Proneness)
E	Effort to Implement
L	Program Level
N	Program Length
N1	Number of Operators
N2	Number of Operands
n	Vocabulary Size (unique operators and unique operands)
n1	Number of unique Operators
n2	Number of unique Operands
T	Implementation Time/Time to Understand
V	Size of the Implementation of an Algorithm (Volume)

Misc.

MaxND	Maximum Nested Depth of {}
c%	Comment Ratio: 100 * LOCcom / LOCphy
a%	Alarm Ratio: 100 * Alarms / MaxAlarms