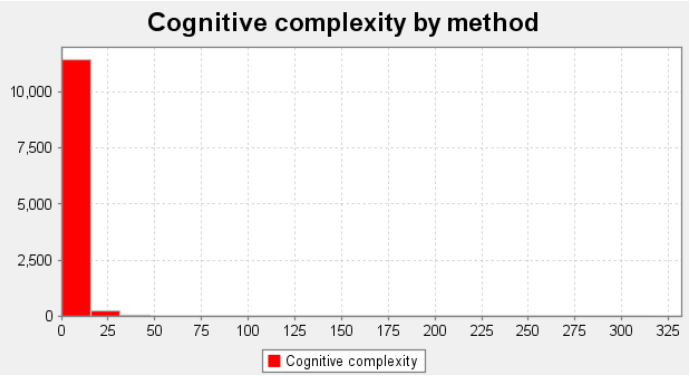
Complexity metrics

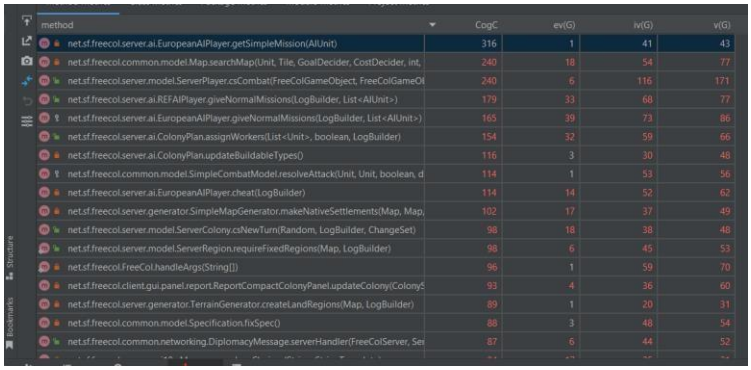
As complexidades métricas são ferramentas que nos ajudam avaliar a complexidade de um projeto e a identificar partes do código que podem ser difíceis de entender.

Method metrics

A **complexidade cognitiva** (CogC) avalia o quão difícil é entender uma unidade de código, ou seja, mede a carga cognitiva que um programador precisa para compreender uma unidade de código. Quanto menor for a CogC melhor, para que o código seja de fácil compreensão.

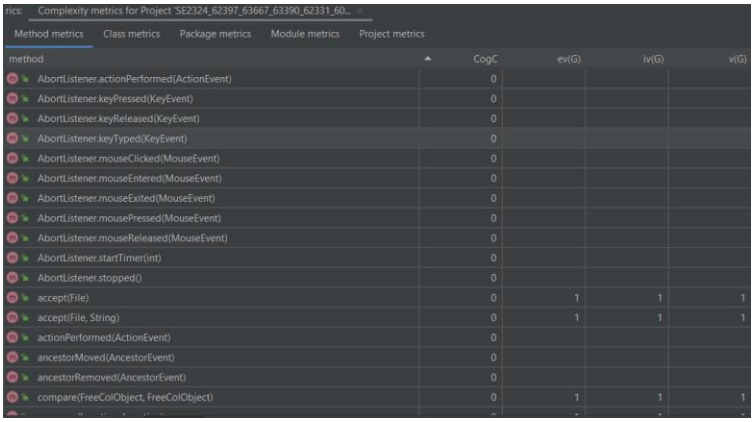


No nosso projeto, conseguimos analisar que existe um método (net/sf/freecol/server/ai/EuropeanAIPlayer.getSimpleMission(AIUnit)) com CogC igual a 316, sendo este o pior caso. Este valor é extremamente alto e, na maioria dos casos, é um indicador de que este método é extramente complexo e difícil de entender.



method	CogC	ev(G)	iv(G)	v(G)
net.sf.freecol.server.ai.EuropeanAIPlayer.getSimpleMission(AIUnit)	316	1	41	43
net.sf.freecol.common.model.Map.searchMap(Unit, Tile, GoalDecider, CostDecider, int,	240	18	54	77
net.sf.freecol.server.model.ServerPlayer.csCombat(FreeColGameObject, FreeColGameOl	240	6	116	171
net.sf.freecol.server.ai.REFAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	179	33	68	77
net.sf.freecol.server.ai.EuropeanAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	165	39	73	86
net.sf.freecol.server.ai.ColonyPlan.assignWorkers(List<Unit>, boolean, LogBuilder)	154	32	59	66
net.sf.freecol.server.ai.ColonyPlan.updateBuildableTypes()	116	3	30	48
net.sf.freecol.common.model.SimpleCombatModel.resolveAttack(Unit, Unit, boolean, d	114	1	53	56
net.sf.freecol.server.ai.EuropeanAIPlayer.cheat(LogBuilder)	114	14	52	62
net.sf.freecol.server.generator.SimpleMapGenerator.makeNativeSettlements(Map, Map,	102	17	37	49
net.sf.freecol.server.model.ServerColony.csNewTurn(Random, LogBuilder, ChangeSet)	98	18	38	48
net.sf.freecol.server.model.ServerRegion.requireFixedRegions(Map, LogBuilder)	98	6	45	53
net.sf.freecol.FreeCol.handleArgs(String[])	96	1	59	70
net.sf.freecol.client.gui.panel.report.ReportCompactColonyPanel.updateColony(ColonyS	93	4	36	60
net.sf.freecol.server.generator.TerrainGenerator.createLandRegions(Map, LogBuilder)	89	1	20	31
net.sf.freecol.common.model.Specification.fixSpec()	88	3	48	54
net.sf.freecol.common.networking.DiplomacyMessage.serverHandler(FreeColServer, Ser	87	6	44	52

Por outro lado, podemos analisar que há bastantes métodos com CogC igual a 0 que, geralmente significa que o método é extremamente simples e de fácil compreensão. Normalmente, os métodos que têm CogC igual a 0 são métodos que executam uma única tarefa simples.

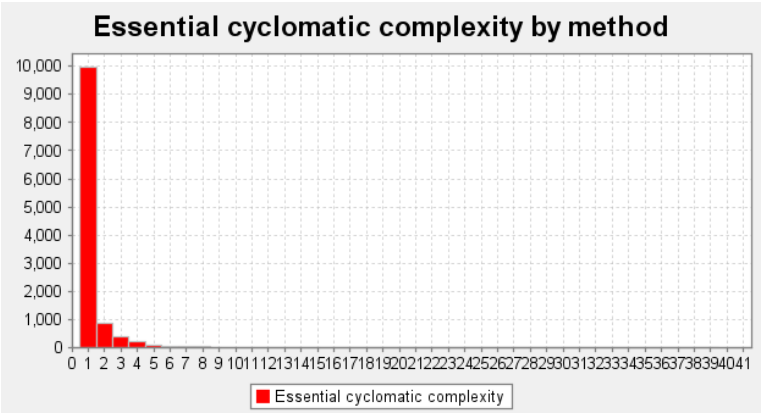


Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

method	CogC	ev(G)	iv(G)	v(G)
AbortListener.actionPerformed(ActionEvent)	0			
AbortListener.keyPressed(KeyEvent)	0			
AbortListener.keyReleased(KeyEvent)	0			
AbortListener.keyTyped(KeyEvent)	0			
AbortListener.mouseClicked(MouseEvent)	0			
AbortListener.mouseEntered(MouseEvent)	0			
AbortListener.mouseExited(MouseEvent)	0			
AbortListener.mousePressed(MouseEvent)	0			
AbortListener.mouseReleased(MouseEvent)	0			
AbortListener.startTimer(int)	0			
AbortListener.stopped()	0			
accept(File)	0	1	1	1
accept(File, String)	0	1	1	1
actionPerformed(ActionEvent)	0			
ancestorMoved(AncestorEvent)	0			
ancestorRemoved(AncestorEvent)	0			
compare(FreeColObject, FreeColObject)	0	1	1	1

A **complexidade essencial** (ev(G)) é uma métrica que se concentra na complexidade estrutural do código e a qualidade do código. A ev(G) é uma extensão da métrica de complexidade ciclomática que calcula a quantidade de caminhos independentes que podem ser percorridos durante a execução desse método. Esta métrica tem em consideração todas as formas possíveis de controlo de fluxo, como loops, estruturas de decisão condicional (como if-else, switch-case, etc).



No nosso trabalho, podemos analisar que existe um método (net/sf/freecol/server/ai/EuropeanAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)) com ev(G) igual a 39 e podemos deduzir que é relativamente alta. Pode ser um indicador de que o método em questão é demasiado complexo, tornando assim o código mais difícil de entender e testar.

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

method	CogC	ev(G)	iv(G)	v(G)
net.sf.freecol.server.ai.EuropeanAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	165	39	73	86
net.sf.freecol.server.ai.REFAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	179	33	68	77
net.sf.freecol.server.ai.ColonyPlan.assignWorkers(List<Unit>, boolean, LogBuilder)	154	32	59	66
net.sf.freecol.server.ai.mission.Mission.travelToTarget(Location, CostDecider, LogBuilde	63	26	28	35
net.sf.freecol.server.ai.mission.PioneeringMission.doMission(LogBuilder)	51	25	32	40
net.sf.freecol.server.ai.mission.BuildColonyMission.doMission(LogBuilder)	76	22	35	38
net.sf.freecol.server.model.ServerPlayer.checkForDeath()	42	21	22	32
net.sf.freecol.server.ai.EuropeanAIPlayer.buyUnitsInEurope(LogBuilder)	42	20	15	27
net.sf.freecol.common.model.Map.searchMap(Unit, Tile, GoalDecider, CostDecider, int,	240	18	54	77
net.sf.freecol.common.model.Scope.equals(Object)	32	18	6	18
net.sf.freecol.server.model.ServerColony.csNewTurn(Random, LogBuilder, ChangeSet)	98	18	38	48
net.sf.freecol.client.gui.option.OptionUI.getOptionUI(GUI, Option, boolean)	19	17	15	17
net.sf.freecol.common.i18n.Messages.replaceChoices(String, StringTemplate)	84	17	25	31
net.sf.freecol.common.model.Stance.getTensionModifier(Stance)	9	17	5	26
net.sf.freecol.common.networking.ClaimLandMessage.serverHandler(FreeColServer, Ser	43	17	21	24
net.sf.freecol.common.resources.ResourceMapping.preload(PreloadController)	24	17	9	17
net.sf.freecol.server.ai.mission.TransportMission.tryCargo(Cargo, LogBuilder)	51	17	24	31

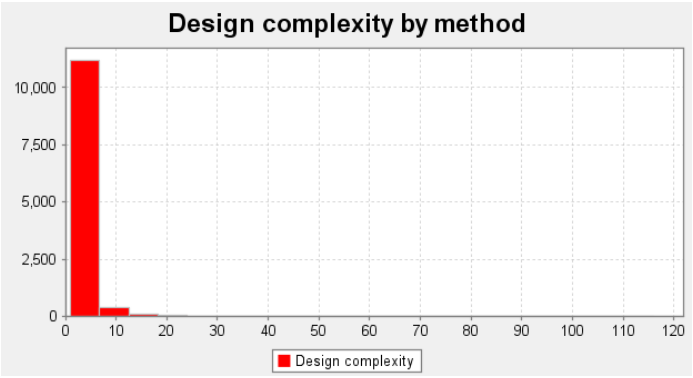
Por outro lado, podemos analisar que há alguns métodos que tem ev(G) igual a 0, ou seja, não existem caminhos independentes no método em questão, o que indica que o código é extremamente simples. Uma ev(G) igual a 0 é a métrica que representa a simplicidade máxima em um método.

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

method	CogC	ev(G)	iv(G)	v(G)
AbortListener.actionPerformed(ActionEvent)	0			
AbortListener.execute()	1			
AbortListener.keyPressed(KeyEvent)	0			
AbortListener.keyReleased(KeyEvent)	0			
AbortListener.keyTyped(KeyEvent)	0			
AbortListener.mouseClicked(MouseEvent)	0			
AbortListener.mouseEntered(MouseEvent)	0			
AbortListener.mouseExited(MouseEvent)	0			
AbortListener.mousePressed(MouseEvent)	0			
AbortListener.mouseReleased(MouseEvent)	0			
AbortListener.startTimer(int)	0			
AbortListener.stopped()	0			
actionPerformed(ActionEvent)	0			

A **complexidade de design** (iv(G)) é uma métrica qualitativa que considera a organização e a estrutura do código.



No nosso trabalho, podemos analisar que existe um método (net/sf/freecol/server/model/ServerPlayer.csCombat(FreeColGameObject, FreeColGameObject, List <CombatEffectType>, Random, ChangeSet)) com um iv(G) igual a 116. Podemos supor que este método tem um iv(G) bastante elevado, sendo um indicador na necessidade de rever o design do método para tentar simplificar o mesmo.

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

method	CogC	ev(G)	iv(G)	v(G)
net.sf.freecol.server.model.ServerPlayer.csCombat(FreeColGameObject, FreeColGameOI	240	6	116	171
net.sf.freecol.server.ai.EuropeanAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	165	39	73	86
net.sf.freecol.server.ai.REFAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	179	33	68	77
net.sf.freecol.FreeCol.handleArgs(String[])	96	1	59	70
net.sf.freecol.server.ai.ColonyPlan.assignWorkers(List<Unit>, boolean, LogBuilder)	154	32	59	66
net.sf.freecol.common.model.Map.searchMap(Unit, Tile, GoalDecider, CostDecider, int,	240	18	54	77
net.sf.freecol.common.model.SimpleCombatModel.resolveAttack(Unit, Unit, boolean, d	114	1	53	56
net.sf.freecol.server.ai.EuropeanAIPlayer.cheat(LogBuilder)	114	14	52	62
net.sf.freecol.common.model.Specification.fixSpec()	88	3	48	54
net.sf.freecol.server.model.ServerRegion.requireFixedRegions(Map, LogBuilder)	98	6	45	53
net.sf.freecol.common.networking.DiplomacyMessage.serverHandler(FreeColServer, Ser	87	6	44	52
net.sf.freecol.common.model.Player.getAllColonyValues(Tile)	81	13	43	54
net.sf.freecol.server.ai.EuropeanAIPlayer.getSimpleMission(AIUnit)	316	1	41	43

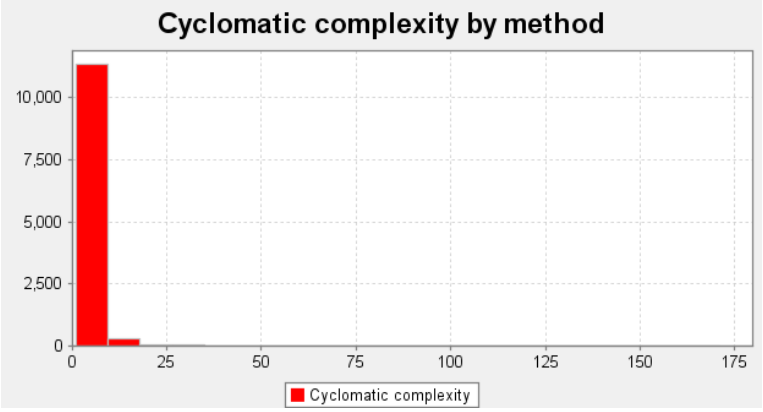
Por outro lado, podemos averiguar que existem alguns métodos com uma iv(G) igual a 0. Uma complexidade de design tão baixa pode oferecer vantagens em termos de simplicidade e legibilidade, mas é raro encontrar projetos inteiros com essa característica. Algumas vantagens de uma complexidade de design baixa são: a sua simplicidade extrema, o seu baixo risco de erros, a sua alta eficiência, a facilidade em realizar testes e a simplicidade do código faz com que a colaboração entre membro de equipa seja mais fácil.

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

method	CogC	ev(G)	iv(G)	v(G)
AbortListener.actionPerformed(ActionEvent)	0			
AbortListener.execute()	1			
AbortListener.keyPressed(KeyEvent)	0			
AbortListener.keyReleased(KeyEvent)	0			
AbortListener.keyTyped(KeyEvent)	0			
AbortListener.mouseClicked(MouseEvent)	0			
AbortListener.mouseEntered(MouseEvent)	0			
AbortListener.mouseExited(MouseEvent)	0			
AbortListener.mousePressed(MouseEvent)	0			
AbortListener.mouseReleased(MouseEvent)	0			
AbortListener.startTimer(int)	0			
AbortListener.stopped()	0			
actionPerformed(ActionEvent)	0			

A **complexidade ciclomática** (v(G)) avalia a complexidade do código tendo em consideração todos os elementos que afetam o fluxo do método. Em termos práticos, a complexidade ciclomática mede a quantidade de caminhos independentes que podem ser percorridos no código, isto é, cada estrutura condicional, cada loop e cada "switch-case" aumentam a complexidade ciclomática, pois introduz novos caminhos possíveis no método.



No nosso trabalho, podemos ver que existe um método (net/sf/freecol/server/model/ServerPlayer.csCombat(FreeColGameObject,FreeColGameObject, List <CombatEffectType>, Random, ChangeSet)) com uma v(G) igual a 171. Uma v(G) igual a 171 é extremamente alta e pode ser um indicador de que o código é extremamente complexo e difícil de compreender, testar e manter.

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

method	CogC	ev(G)	iv(G)	v(G)
net.sf.freecol.server.model.ServerPlayer.csCombat(FreeColGameObject, FreeColGameOI	240	6	116	171
net.sf.freecol.server.ai.EuropeanAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	165	39	73	86
net.sf.freecol.common.model.Map.searchMap(Unit, Tile, GoalDecider, CostDecider, int,	240	18	54	77
net.sf.freecol.server.ai.REFAIPlayer.giveNormalMissions(LogBuilder, List<AIUnit>)	179	33	68	77
net.sf.freecol.FreeCol.handleArgs(String[])	96	1	59	70
net.sf.freecol.server.ai.ColonyPlan.assignWorkers(List<Unit>, boolean, LogBuilder)	154	32	59	66
net.sf.freecol.server.ai.EuropeanAIPlayer.cheat(LogBuilder)	114	14	52	62
net.sf.freecol.client.control.InGameController.moveDirection(Unit, Direction, boolean)	54	1	23	60
net.sf.freecol.client.gui.panel.report.ReportCompactColonyPanel.updateColony(ColonyS	93	4	36	60
net.sf.freecol.common.model.SimpleCombatModel.resolveAttack(Unit, Unit, boolean, d	114	1	53	56
net.sf.freecol.common.model.Player.getAllColonyValues(Tile)	81	13	43	54
net.sf.freecol.common.model.Specification.fixSpec()	88	3	48	54
net.sf.freecol.server.model.ServerRegion.requireFixedRegions(Map, LogBuilder)	98	6	45	53

Por outro lado, podemos observar que existem bastantes métodos com complexidade ciclomática inferior a 10, ou seja esses métodos tem poucos caminhos independentes e oferecem bastantes vantagens.

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

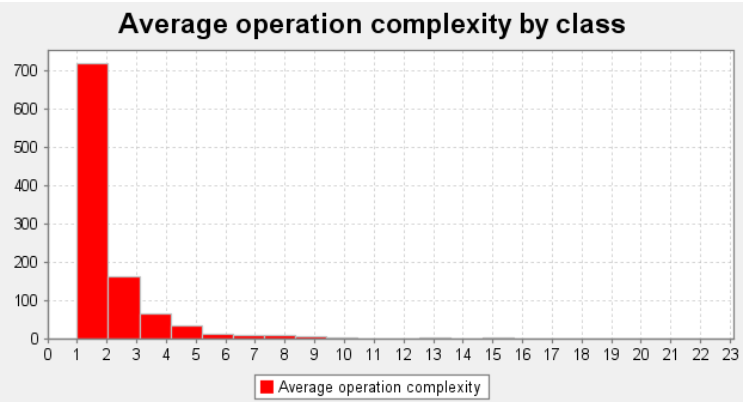
Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

method	CogC	ev(G)	iv(G)	v(G)
AbortListener.actionPerformed(ActionEvent)	0			
AbortListener.execute()	1			
AbortListener.keyPressed(KeyEvent)	0			
AbortListener.keyReleased(KeyEvent)	0			
AbortListener.keyTyped(KeyEvent)	0			
AbortListener.mouseClicked(MouseEvent)	0			
AbortListener.mouseEntered(MouseEvent)	0			
AbortListener.mouseExited(MouseEvent)	0			
AbortListener.mousePressed(MouseEvent)	0			
AbortListener.mouseReleased(MouseEvent)	0			
AbortListener.startTimer(int)	0			
AbortListener.stopped()	0			
actionPerformed(ActionEvent)	0			

Class metrics

As **métricas de uma classe** são métricas usadas para avaliar as características e a qualidade das classes num projeto/sistema.

A **complexidade média das operações** (OCavg) mede a complexidade das operações em relação à lógica contida em cada método.



No nosso trabalho, podemos analisar que existe uma classe(net/sf/freecol/tools/ForestMaker) com uma OCavg igual a 22.00. Uma classe com uma OCavg igual a 22.00 é relativamente alta e pode ser um sinal de que as operações na classe são, em média, complexas. Isto pode resultar em código que é mais difícil de entender, testar e manter.

rics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

class	OCavg	OCmax	WMC
net.sf.freecol.tools.ForestMaker	22.00	22	22
net.sf.freecol.common.resources.ResourceFactory	15.00	15	15
net.sf.freecol.server.ai.Cargo.CargoPlan	15.00	15	15
net.sf.freecol.tools.ColonizationMapReader	15.00	15	15
net.sf.freecol.server.generator.River.DirectionChange	14.00	14	14
net.sf.freecol.client.gui.tooltip.BuildingToolTip	13.50	26	27
net.sf.freecol.server.ai.REFAIPlayer	13.50	56	135
net.sf.freecol.client.gui.panel.colopedia.BuildingDetailPanel	9.67	22	29
net.sf.freecol.client.gui.panel.DragListener	9.50	18	19
net.sf.freecol.client.gui.panel.report.ReportCompactColonyPanel.ColonySummary	9.33	16	28
net.sf.freecol.server.generator.TerrainGenerator	9.07	30	136
net.sf.freecol.client.gui.panel.report.ReportContinentalCongressPanel	9.00	9	9
net.sf.freecol.client.gui.panel.report.ReportIndianPanel	9.00	15	18

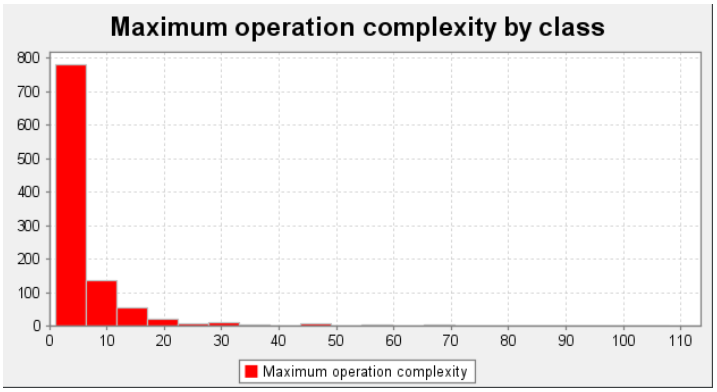
Por outro lado, podemos ver que existem bastantes classes com uma OCavg bastante baixa, até mesmo igual a 0. No entanto, é importante equilibrar a simplicidade com a necessidade de cumprir os requisitos do projeto. Em alguns casos, operações complexas podem ser justificadas se estiver a tratar de problemas complexos.

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

class	OCavg	OCmax	WMC
net.sf.freecol.client.gui.Canvas.ToolBoxFrame			0
net.sf.freecol.client.gui.FontLibrary.FontType			0
net.sf.freecol.client.gui.GUI.ViewMode			0
net.sf.freecol.client.gui.SwingGUI.PopupPosition			0
net.sf.freecol.client.gui.dialog.Flag.Alignment			0
net.sf.freecol.client.gui.dialog.Flag.UnionShape			0
net.sf.freecol.client.gui.dialog.FreeColDialog.DialogType			0
net.sf.freecol.client.gui.label.UnitLabel.UnitAction			0
net.sf.freecol.client.gui.mapviewer.MapViewer.BorderType			0
net.sf.freecol.client.gui.panel.EuropePanel.EuropeAction			0
net.sf.freecol.client.gui.panel.FindSettlementPanel.DisplayListOption			0
net.sf.freecol.client.gui.panel.FreeColButton.ButtonStyle			0
net.sf.freecol.client.gui.panel.InfoPanel.InfoPanelMode			0

A **complexidade máxima de operações** (OCmax) é a métrica que avalia a complexidade mais alta entre todos os métodos em um sistema.



Podemos observar que a classe com a OCmax mais alta é a classe (net/sf/freecol/server/model/ServerPlayer).

Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

class	OCavg	OCmax	WMC
net.sf.freecol.server.model.ServerPlayer	5.72	108	601
net.sf.freecol.server.ai.EuropeanAIPlayer	6.40	70	525
net.sf.freecol.common.model.Map	3.41	68	351
net.sf.freecol.FreeCol	3.04	63	207
net.sf.freecol.server.ai.ColonyPlan	8.38	59	218
net.sf.freecol.client.gui.panel.report.ReportCompactColonyPanel	8.07	58	121
net.sf.freecol.server.ai.REFAIPlayer	13.50	56	135
net.sf.freecol.common.model.Specification	2.49	48	359
net.sf.freecol.server.model.ServerRegion	4.50	47	72
net.sf.freecol.client.control.InGameController	4.62	45	786
net.sf.freecol.common.model.Player	1.97	45	536
net.sf.freecol.common.networking.DiplomacyMessage	5.08	45	66
net.sf.freecol.server.generator.SimpleMapGenerator	8.22	44	148

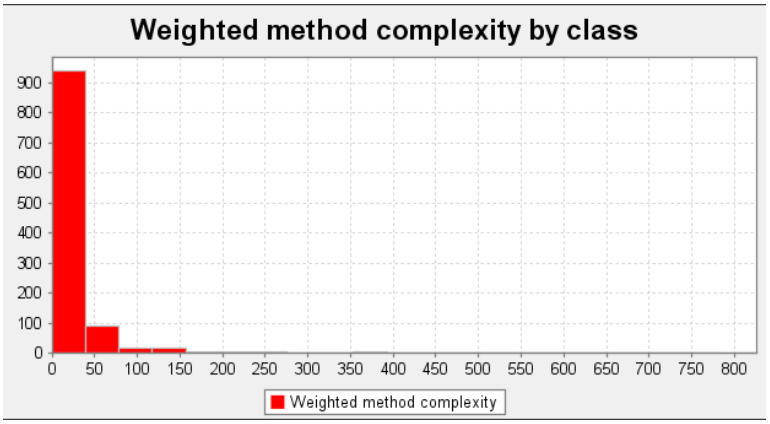
Por outro lado, podemos ver que existem bastantes classes onde a OCmax é bastante baixa.

trics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

class	OCavg	OCmax	WMC
net.sf.freecol.client.gui.Canvas.ToolBoxFrame			0
net.sf.freecol.client.gui.FontLibrary.FontType			0
net.sf.freecol.client.gui.GUI.ViewMode			0
net.sf.freecol.client.gui.SwingGUI.PopupPosition			0
net.sf.freecol.client.gui.dialog.Flag.Alignment			0
net.sf.freecol.client.gui.dialog.Flag.UnionShape			0
net.sf.freecol.client.gui.dialog.FreeColDialog.DialogType			0
net.sf.freecol.client.gui.label.UnitLabel.UnitAction			0
net.sf.freecol.client.gui.mapviewer.MapViewer.BorderType			0
net.sf.freecol.client.gui.panel.EuropePanel.EuropeAction			0
net.sf.freecol.client.gui.panel.FindSettlementPanel.DisplayListOption			0
net.sf.freecol.client.gui.panel.FreeColButton.ButtonStyle			0
net.sf.freecol.client.gui.panel.InfoPanel.InfoPanelMode			0

A **complexidade ponderada de métodos** (WMC) é uma métrica que avalia a complexidade de métodos ou operações em um sistema, atribuindo pesos diferentes aos diversos elementos encontrados em um método.



No nosso trabalho, podemos ver que existe uma classe (net/sf/freecol/client/control/InGameController) com a WMC mais alta.

trics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

class	OCavg	OCmax	WMC
net.sf.freecol.client.control.InGameController	4.62	45	786
net.sf.freecol.common.model.Unit	2.51	23	676
net.sf.freecol.server.model.ServerPlayer	5.72	108	601
net.sf.freecol.common.model.Player	1.97	45	536
net.sf.freecol.server.ai.EuropeanAIPlayer	6.40	70	525
net.sf.freecol.server.control.InGameController	4.42	30	398
net.sf.freecol.common.model.Colony	2.15	11	396
net.sf.freecol.client.gui.SwingGUI	1.86	18	371
net.sf.freecol.common.model.Tile	2.26	22	366
net.sf.freecol.common.model.Specification	2.49	48	359
net.sf.freecol.common.model.Map	3.41	68	351
net.sf.freecol.client.gui.ImageLibrary	1.59	16	272
net.sf.freecol.common.util.CollectionUtils	1.49	7	269

Podemos, por outro lado, observar que existem bastantes classes com uma WMC bastante reduzida.

trics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

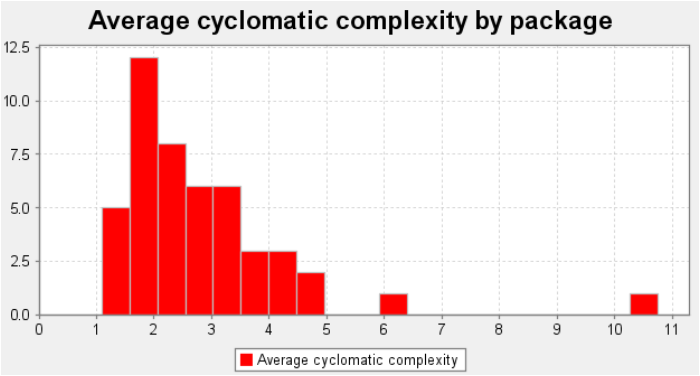
Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

class	OCavg	OCmax	WMC
net.sf.freecol.client.gui.Canvas.ToolBoxFrame			0
net.sf.freecol.client.gui.FontLibrary.FontType			0
net.sf.freecol.client.gui.GUI.ViewMode			0
net.sf.freecol.client.gui.SwingGUI.PopupPosition			0
net.sf.freecol.client.gui.dialog.Flag.Alignment			0
net.sf.freecol.client.gui.dialog.Flag.UnionShape			0
net.sf.freecol.client.gui.dialog.FreeColDialog.DialogType			0
net.sf.freecol.client.gui.label.UnitLabel.UnitAction			0
net.sf.freecol.client.gui.mapviewer.MapViewer.BorderType			0
net.sf.freecol.client.gui.panel.EuropePanel.EuropeAction			0
net.sf.freecol.client.gui.panel.FindSettlementPanel.DisplayListOption			0
net.sf.freecol.client.gui.panel.FreeColButton.ButtonStyle			0
net.sf.freecol.client.gui.panel.InfoPanel.InfoPanelMode			0

Package metrics

As **métricas de pacotes** são particularmente importantes em projetos de grande escala, onde a estrutura do sistema é complexa e a manutenção é crítica.

A **complexidade ciclomática média** (v(G)avg) é uma métrica que calcula a complexidade média de cada método num sistema, sendo o package com maior v(G)avg o net/sf/freecol/client/gui/tooltip.



trics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics

package	v(G)avg	v(G)tot
net.sf.freecol.client.gui.tooltip	10.75	43
net.sf.freecol.server.generator	6.01	463
net.sf.freecol.client.control	4.87	1,212
net.sf.freecol.server.model	4.52	1,577
net.sf.freecol.client.gui.panel.report	4.27	675
net.sf.freecol.tools	4.23	292
net.sf.freecol.common.model.pathfind	4.10	160
net.sf.freecol.server.ai	3.95	2,483
net.sf.freecol.client.gui.images	3.62	105
net.sf.freecol.server.control	3.51	541
net.sf.freecol	3.50	245
net.sf.freecol.common.logging	3.50	21
net.sf.freecol.common.model.producti	3.40	85

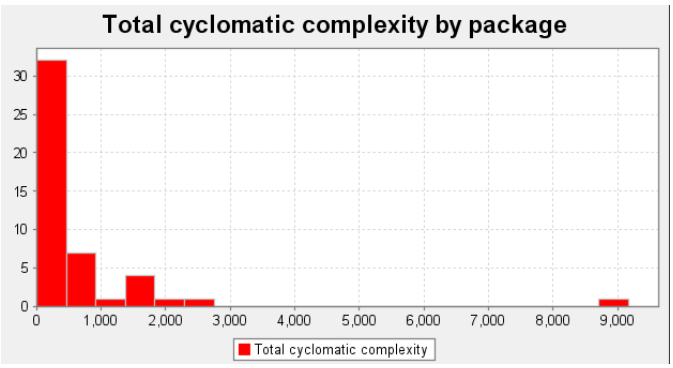
Por outro lado, podemos ver que o package com menor v(G)avg o net.sf.freecol/common.

trics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics

package	v(G)avg	v(G)tot
net.sf.freecol.common	1.10	11
net.sf.freecol.common.model.mission	1.38	73
net.sf.freecol.client.gui.video	1.50	12
net.sf.freecol.common.utils	1.50	6
net.sf.freecol.common.io.sza	1.56	25
net.sf.freecol.client.networking	1.58	19
net.sf.freecol.client.gui.action	1.60	404
net.sf.freecol.server.networking	1.71	36
net.sf.freecol.common.option	1.75	493
net.sf.freecol.client.gui.menu	1.75	63
net.sf.freecol.common.networking	1.82	1,797
net.sf.freecol.client.gui.option	1.88	244
net.sf.freecol.client.gui.plaf	1.92	236

A **complexidade ciclomática total** (v(G)tot) é uma métrica que avalia a complexidade global de um sistema, tendo em conta todos os métodos presentes. O package com uma v(G)tot maior é o package net/sf/freecol/common/model



trics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics

package	v(G)avg	v(G)tot
net.sf.freecol.common.model	2.25	9,171
net.sf.freecol.server.ai	3.95	2,483
net.sf.freecol.client.gui.panel	2.69	1,917
net.sf.freecol.common.networking	1.82	1,797
net.sf.freecol.client.gui	1.96	1,665
net.sf.freecol.server.model	4.52	1,577
net.sf.freecol.server.ai.mission	3.27	1,408
net.sf.freecol.client.control	4.87	1,212
net.sf.freecol.client.gui.dialog	2.96	802
net.sf.freecol.client.gui.panel.report	4.27	675
net.sf.freecol.client.gui.mapviewer	3.24	644
net.sf.freecol.common.util	2.05	583
net.sf.freecol.server.control	3.51	541

Podemos, por outro lado, ver que o package com menor v(G)tot é o package net/sf/freecol/common/utils. Sendo este package o de testes.

trics: Complexity metrics for Project 'SE2324_62397_63667_63390_62331_60...

Method metrics | Class metrics | Package metrics | Module metrics

package	v(G)avg	v(G)tot
net.sf.freecol.common.utils	1.50	6
net.sf.freecol.common	1.10	11
net.sf.freecol.client.gui.video	1.50	12
net.sf.freecol.client.networking	1.58	19
net.sf.freecol.common.logging	3.50	21
net.sf.freecol.common.io.sza	1.56	25
net.sf.freecol.server.networking	1.71	36
net.sf.freecol.client.gui.tooltip	10.75	43
net.sf.freecol.client.gui.animation	3.36	47
net.sf.freecol.metaserver	1.92	50
net.sf.freecol.common.metaserver	2.18	61
net.sf.freecol.client.gui.menu	1.75	63
net.sf.freecol.common.model.mission	1.38	73

Module Metrics

As **métricas de módulos** são métricas que avalias as características, a qualidade e a complexidade demódulos.

No nosso trabalho, podemos ver que o módulo com maior v(G)avg e v(G)tot é o módulo principal do projeto. Por outro lado, o módulo com menor v(G)avg e v(G)tot é o modulo de test.

Method metrics | Class metrics | Package metrics | Module metrics | Proj

module	v(G)avg	v(G)tot
SE2324_62397_63667_63390_62331_60253_	2.62	29,064
test	1.81	1,140
Total		**30,204**
Average	2.58	15,102.00

Project metrics

As **métricas do projeto** são métricas usadas para avaliar a qualidade e o desempenho do projeto.

O projeto apresenta uma v(G)avg de 2.58 que é relativamente baixa, ou seja, os métodos do projeto têm baixa complexidade em média. Já sobre a v(G)tot podemos ver que tem um valor de 30.204, o que significa que o sistema como um todo tem complexidade relativamente alta, o que pode tornar ocódigo mais difícil de gerir no seu todo.

Method metrics | Class metrics | Package metrics | Module metrics | Project metrics

project	v(G)avg	v(G)tot
project	2.58	30,204

Relação das complexidades métricas com os Code Smells

Long Method - A complexidade ciclomática e a complexidade cognitiva está relacionada a métodos muito longos, pois métodos com elevada complexidade ciclomática podem ter muitos caminhos independentes, tornando-os mais longos e difíceis de compreender e no caso da complexidade cognitiva, esta está relacionada com métodos longos pois se a carga cognitiva que um programador precisa para compreender uma unidade de código é maior, significa que este método não é simples compreensão.

Large Class - Além da complexidade ciclomática, a complexidade ponderada de métodos está tambémrelacionada a este code smell, já que uma grande quantidade de métodos em uma classe, normalmente indica que esta classe é grande.

Message Chain – A complexidade ponderada de métodos ajuda a identificar métodos que fazem varias chamadas de métodos encadeados numa única linha de código, já que este code smell envolve a chamada de métodos em série.