

Testovania scenára true pre debug taktiku bez logov

Počiatočný stav bez pridaného a zapnutého Jima:

1. Heap memory:
 - 15 MB alokovanej
 - 4 MB využitej
2. Non - Heap memory
 - 15 MB alokovanej
 - 14,6 MB využitej
3. Garbage collection
 - 1/s
4. Garbage collection pauses
 - striedanie 0ms s 14ms
5. Classes
 - 2693 načítaných
6. Object allocation recording
 - 60 K/s
7. CPU usage telemetry
 - CPU TIME user + kernel 10-15% cca
 - CPU TIME kernel 4-7% cca
 - Time spent on GC 0-1%
8. Threads
 - 16 threadov z toho 11 daemon
 - hodnota state sa nepredvídateľne menila (blocked, waiting, waiting in native, runnable)

Stav počas spustenej taktiky:

1. Heap memory:
 - 15 MB alokovanej
 - 11 MB využitej
2. Non - Heap memory
 - 17 MB alokovanej
 - 16,1 MB využitej
3. Garbage collection
 - 15/s
4. Garbage collection pauses
 - priemer 4-7 ms s maximálnymi hodnotami až po 41 ms
5. Classes
 - 2909 načítaných
6. Object allocation recording
 - 200-250 K/s
7. CPU usage telemetry
 - CPU TIME user + kernel 40-65% cca
 - CPU TIME kernel 5-11% cca
 - Time spent on GC 4-9%
8. Threads
 - 18 threadov z toho 12 daemon
 - hodnota state sa nepredvídateľne menila (blocked, waiting, waiting in native, runnable)

Zistiť čo je

- used eden space
- used survivor space
- used tenured gen
- used perm gen
- use code cache

Hodnoty memory pre class:

Class	Objects	Shallow Size	Retained Size
char[]	12,786 (12%)	7,290,536 (46%)	≈ 7,290,536 (46%)
byte[]	3,963 (4%)	3,314,032 (21%)	≈ 3,314,032 (21%)
java.lang.Class	3,213 (3%)	1,376,304 (9%)	≈ 1,972,176 (12%)
java.lang.String	11,644 (11%)	279,456 (2%)	≈ 742,904 (5%)
int[]	780 (1%)	638,576 (4%)	≈ 638,576 (4%)
sk.fiit.robocup.library.geometry.Vector3D	6,926 (7%)	387,856 (2%)	≈ 387,856 (2%)
java.util.LinkedList\$Node	10,031 (9%)	240,744 (2%)	≈ 361,392 (2%)
java.util.HashMap	782 (1%)	37,536 (0%)	≈ 333,712 (2%)
java.lang.Object[]	7,538 (7%)	210,400 (1%)	≈ 329,472 (2%)
sk.fiit.jim.agent.models.PositionHistory	5,000 (5%)	320,000 (2%)	≈ 320,224 (2%)
java.util.HashMap\$Entry[]	687 (1%)	66,712 (0%)	≈ 313,040 (2%)
java.util.HashMap\$Entry	3,418 (3%)	82,032 (1%)	≈ 288,448 (2%)
short[]	4,745 (4%)	270,048 (2%)	≈ 270,048 (2%)
sk.fiit.jim.agent.models.AgentModel	4 (0%)	608 (0%)	≈ 193,456 (1%)
sk.fiit.jim.agent.models.WorldModel	4 (0%)	128 (0%)	≈ 193,200 (1%)
sk.fiit.testframework.communication.agent.AgentJim	1 (0%)	40 (0%)	≈ 193,120 (1%)
java.util.LinkedList	26 (0%)	624 (0%)	≈ 187,992 (1%)

Obr.1: Hodnoty memory pre jednotlivé classy

Class	Shallow size / one
byte[]	836
int[]	818
char[]	570
Class	428
PositionHistory	64
Vector3D	56
short[]	56

Tab.1: Hodnoty memory pre jednu jednotku classy

CPU hodnoty pre metódy:

Method	Time (ms)	Own Time (ms)
java.io.BufferedReader.readLine() BufferedReader.java	821,502 (62%)	821,502
com.intelij.rt.execution.application.AppMainV2\$1.run() AppMainV2.java	480,552 (36%)	480,552
sk.fiit.testframework.monitor.RobocupMonitor.mainLoop() RobocupMonitor.java	477,557 (36%)	0
sk.fiit.testframework.monitor.RobocupMonitor.receive() RobocupMonitor.java	477,557 (36%)	0
sk.fiit.testframework.monitor.RobocupMonitor.run() RobocupMonitor.java	477,557 (36%)	0
java.io.DataInputStream.readInt() DataInputStream.java	477,541 (36%)	477,541
sk.fiit.testframework.monitor.AgentMonitorThread.run() AgentMonitorThread.java	356,903 (27%)	0
java.io.ObjectInputStream.readObject() ObjectInputStream.java	12,453 (1%)	22,000
java.lang.reflect.Constructor.newInstance(Object[]) Constructor.java	3,421 (0%)	2,562
java.awt.EventQueueThread.run() EventDispatchThread.java	1,843 (0%)	156
sk.fiit.testframework.ui.GameView.paintComponent(Graphics) GameView.java	1,390 (0%)	0
sk.fiit.testframework.monitor.AgentMonitorMessage.parse(String, Object) AgentMonitorMessage.java	1,328 (0%)	0
sk.fiit.testframework.monitor.AgentMonitorMessage.init(Stack) AgentMonitorMessage.java	1,312 (0%)	15

Obr.2: CPU vyťaženie jednotlivých metód