Grid.updateNodes: loopa igenom alla noder, anropa node.update

Node.update: this.moveMessages, this.checkRequest

Node.moveMessages() messageQueue.first().move(), messageQueue.first().update

Message.move: currentNode.removeFirstElement(), currNode=nextNode, nextNode.addMessageToQueue(this), jumps++, path.add(currNode)

Request.update(): if(!hasMovedMax): nextNode=findNextNode(), else: currNode.removeFirstElement()

Agent.update: if jumps==maxJumps: currNode.removeFirstElement, else: updateOwnTable() currNode.compareTable(routingTable) nextNode=findNextNode()

Agent.findNextNode: checks where to go next based on where has been and currnodes neighs

Request.findNextNode: checks where to go next based on: 1. currNode has info on happening, 2: random

Node.checkRequest: if(has request):

if(timeSinceRequest<=max):

if(currentRequest.hasReturn):

printMessage,

remove request,   
else:

timeSinceRequest++;

else if(!sentTwice):

sentTwice=true, timeSinceRequest=0, currReqeust=createNewRequest(),

else:

currRequest=null, timesincerequest=0;