01 Oct 2017

Establishing Rules for Self-Organizing Systems-Of-Systems

David M. Curry

Cihan H. Dagli
Missouri University of Science and Technology, dagli@mst.edu

Available online at www.sciencedirect.com
ScienceDirect
Procedia Computer Science 114 (2017) 14–18
Procedia Computer Science
www.elsevier.com/locate/procedia

Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems, CAS
October 30 – November 1, 2017, Chicago, Illinois, USA

Establishing Rules for Self-Organizing Systems-of-Systems

David M. Curry[a,]*, Cihan H. Dagli[a]

[a]*Department of Engineering Management and Systems Engineering,*
Missouri University of Science and Technology
Rolla, Missouri 65409-0370, USA

Abstract

Self-organizing systems-of-systems offer the possibility of autonomously adapting to new circumstances and tasking. This could significantly benefit large endeavors such as smart cities and national defense by increasing the probability that new situations are expediently handled. Complex self-organizing behaviors can be produced by a large set of individual agents all following the same simple set of rules. While biological rule sets have application in achieving human goals, other rules sets may be necessary as these goals are not necessarily mirrored in nature. To this end, a set of system, rather than biologically, inspired rules is introduced and an agent-based model is used to simulate and analyze the behavior produced with various parameters. Agents represent systems and their decisions are defined by the given rule set and parameters. The environment provides a variety of time-critical missions on an ongoing basis. The effectiveness of a particular rule or set of rules is measured by a set of key performance metrics such as the rate at which missions achieve their required capabilities within a given deadline and the average time required to do so. Different rules will be compared using this criterion along with an assessment of their ability to demonstrate beneficial self-organizing behavior.

Peer-review under responsibility of the scientific committee of the Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems.

Keywords: System of Systems; SoS; Self-Organizing; Self-Organization; Agent-Based Modeling; ABM; Emergence

* Corresponding author. Tel.: +1-573-341-4572.
E-mail address: dmcfh3@mst.edu

Peer-review under responsibility of the scientific committee of the Complex Adaptive Systems Conference with Theme: Engineering Cyber Physical Systems.
10.1016/j.procs.2017.09.058

1. Introduction

Self-organizing behaviors are ubiquitous in nature, providing a large measure of stability despite widely varying and unpredictable circumstances [1]. Self-organizing behavior is differentiated from other behaviors in that it is the product of individuals making decision based on local information without external direction. This gives it resilience and scalability because the behavior is encoded in the individual and not dependent upon any particular external entity. While the benefit is obvious, the rules governing individual behavior are not.

Very simple rules have been discovered that produce complex behavior mimicking actual behavior observed in natural systems. One example of this is Schelling's Segregation Model [2] where population segregation is modeled used a single rule. This paper will define four rules for that individual systems will follow in the hopes that an effective self-organizing behavior can be demonstrated for a system-of-systems problem.

This problem has two classes of entities: systems and missions. Missions require a minimum set of capabilities in order to be carried out and have a deadline for acquiring these capabilities before the mission is failed. There are various types of missions, each requiring a different set of capabilities. Systems provide the capabilities required by missions. There are multiple types of systems, each providing a different set of capabilities, although they all move at the same speed. Simulations using various weightings of the four rules will then be carried out and the results analyzed. The parameters of the rule set are the weights given to each rule in the set.

2. Approach

An appropriate approach for this type of problem is spatial agent-based modeling [3]. NetLogo [4] was chosen as the modeling software because of its ease of use and excellent reputation. The missions have the following states: inactive, activated, go, and failed. Missions in the inactive state do nothing until activated at random. In the activated state, they signal their capability needs, location, deadline, and committed systems. If the required capability mix is achieved, then the mission transitions to go and then back to inactive after the mission is complete. If the deadline expires before these capabilities are acquired, the mission fails and transitions back to inactive. The systems have four states as well: idle, enroute, committed, and engaged. Systems are idle when they have not decided on a mission. An enroute system has decided on a mission and is moving towards it. A committed system is one that has reached its mission and is can no longer change missions until this one finishes or fails. An engaged system is one that is participating in a mission that is now in a go state. The four rules that systems use for their decision making, that decision being which mission to choose, are:

- Mission popularity (measures likelihood of mission gaining required assets),
- Distance to mission (measures time commitment required to reach a mission),
- Contribution to mission (measures degree to which a mission can make use of a system), and
- Urgency of mission (measures time remaining before mission failure).

The first rule, mission popularity, is the number of systems assigned to each mission that are in a committed state which is then normalized by dividing by the total number of systems. The second rule, distance to mission, is the Euclidean distance from the agent to the mission rendezvous point normalized by dividing by the longest distance possible. The third rule, contribution to mission, is the number of capabilities supplied by the system that match those required by the mission and is normalized by dividing by the number of capabilities possible. The fourth, and final, rule is mission urgency which is defined as the time left until a mission's deadline expires and is normalized by dividing by the maximum mission deadline.

There is a stochastic element involved as well. A system in the idle state will decide to pursue a mission that it has decided upon with the probability given in the accept rate. A system can decide to pursue a different mission while enroute with a probability defined as the reconsider rate. The stochastic variables are intended to prevent similar systems from always choosing the same mission and becoming redundant. Finally, the weights applied to each rule are restricted to the range $[-1, +1]$, which allows a rules to act as its own antithesis with a negative weight or to be eliminated when its weight is zero. These weights, along with accept and reconsider rates, are the only parameters of the rule set and the weights are known as the rule affinities. When only one rule is non-zero, the magnitude of its weight is irrelevant. When all weights are zero, the behavior is determined entirely by the stochastic variables. When more than one rule is active (non-zero), their relative influence is determined by their assigned weights (affinities).

3. Experimental Setup

A total of eight capabilities are available. The capabilities are generic and not named, the only important quality about them is that each one is unique. There are twenty-eight types of systems, each one providing two of the eight possible capabilities. There are fifty-six types of missions, each one requiring five of the eight capabilities. This means that depending on how well the capabilities of the committed systems match the required capabilities, each mission will require between three and five systems to succeed (although many more may become committed due to inefficient rules).

Two set of trials will be performed. The first set of trials will consist of sixteen combinations of rules where the accept rate is kept at 0.1 and the reconsider rate is kept at 0.03. These rates were chosen because they worked relatively well in preliminary trials across a wide range of rules. The second set of trials consists of eleven combinations of different accept rates and reconsider rates using the best rule set found in the first set of trials. The experiments were carried out using NetLogo to provide the simulation and to collect the pertinent statistics. The experimental setup performing a typical run is shown in Fig. 1.

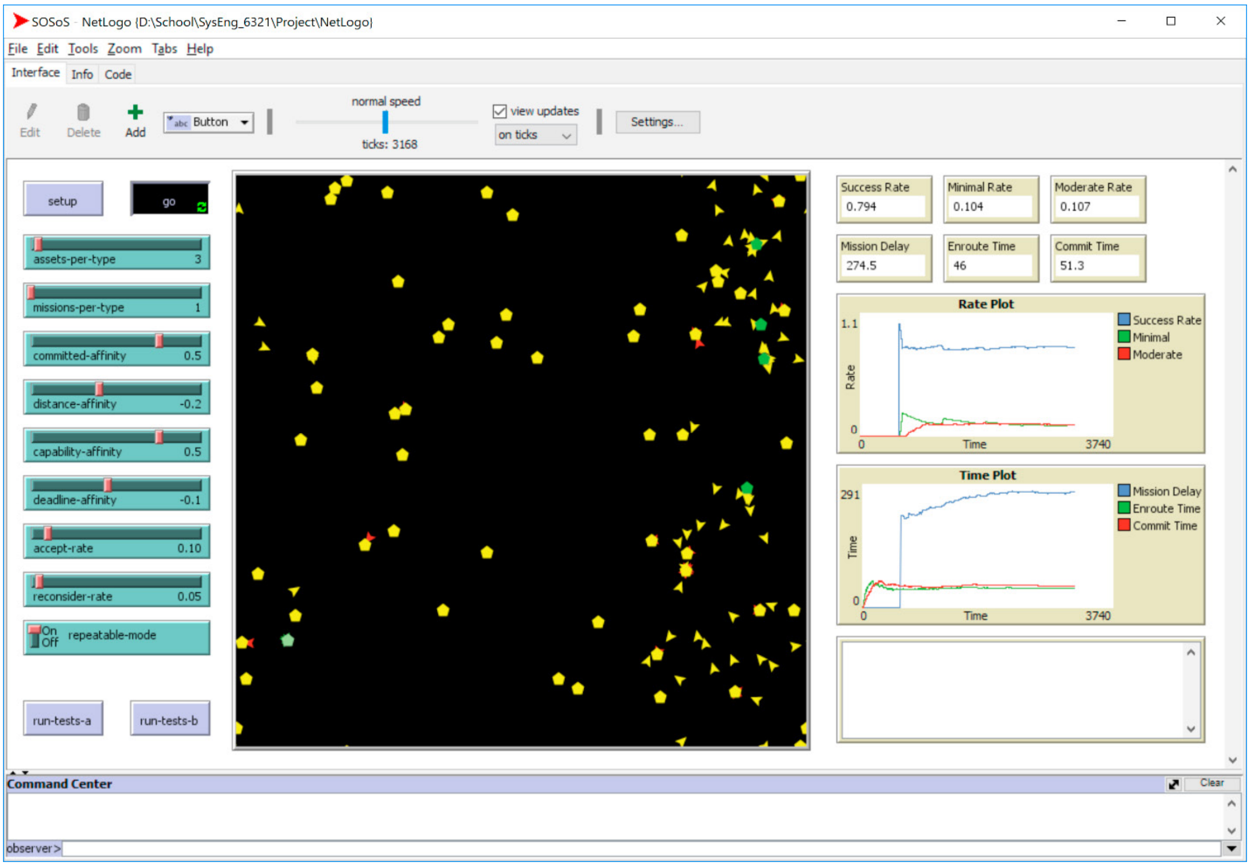

Fig. 1. NetLogo experimental setup. The sliders on the left allow the parameters to be adjusted while the monitors and graphs on the right display the relevant statistical properties. The center is a graphical representation showing the current state of the agents where pentagons represent missions and arrows represent the systems and their direction of travel.

4. Results

The results for the first set of trials is shown in Table 1. Since case fifteen had the highest success rate, it was used in the second set of trials. The results of the second set of trials are shown in Table 2.

Table 1. Differing rule sets with fixed accept and reconsider rates.

Case	Decision Variable				Key Performance Metric					
	Committed Affinity	Distance Affinity	Capability Affinity	Deadline Affinity	Success Rate	Coverage Neglected	Coverage Occasional	Mission Delay	Time Enroute	Time Committed
1	0	0	0	0	0.184	0.756	0.009	149.9	264.3	135.7
2	-0.5	0	0	0	0.006	0.935	0.000	371.8	409.9	352.4
3	0.5	0	0	0	0.426	0.465	0.048	58.2	34.1	9.0
4	0	-0.5	0	0	0.160	0.564	0.086	291.2	25.0	267.8
5	0	0.5	0	0	0.012	0.902	0.000	422.9	620.8	222.4
6	0	0	-0.5	0	0.000	0.000	0.000	0.0	148.4	348.8
7	0	0	0.5	0	0.033	0.715	0.000	379.7	265.6	339.0
8	0	0	0	-0.5	0.433	0.282	0.177	536.5	62.0	57.1
9	0	0	0	0.5	0.126	0.489	0.075	95.2	103.2	248.2
10	0.5	-0.2	0	0	0.514	0.384	0.056	92.0	24.2	15.9
11	0.5	0	0.5	0	0.604	0.257	0.095	138.9	45.4	25.5
12	0.5	0	0	-0.1	0.437	0.341	0.099	204.1	33.5	16.0
13	0.5	0	0	0.1	0.456	0.428	0.052	52.8	28.3	10.7
14	0.5	-0.2	0.5	0	0.735	0.113	0.118	177.8	40.0	45.9
15	0.5	-0.2	0.5	-0.1	0.770	0.060	0.123	292.2	46.4	51.2
16	0.5	-0.2	0.5	0.1	0.692	0.146	0.137	126.2	37.2	44.7

Table 2. Fixed rule set with differing accept and reconsider rates.

Case	Stochastic Variable		Key Performance Metric					
	Accept Rate	Reconsider Rate	Success Rate	Coverage Neglected	Coverage Occasional	Mission Delay	Time Enroute	Time Committed
1	1	0	0.449	0.047	0.123	294.9	67.2	9.0
2	1	1	0.406	0.044	0.240	363.5	27.6	15.6
3	0.05	0.01	0.709	0.044	0.163	327.9	62.7	83.3
4	0.05	0.03	0.751	0.040	0.165	297.6	50.6	59.9
5	0.05	0.05	0.769	0.043	0.120	289.8	46.2	51.0
6	0.1	0.01	0.712	0.051	0.173	325.1	62.0	82.3
7	0.1	0.03	0.759	0.045	0.141	296.6	50.8	58.9
8	0.1	0.05	0.770	0.060	0.123	292.2	46.4	51.2
9	0.15	0.01	0.707	0.045	0.181	328.1	62.6	82.2
10	0.15	0.03	0.757	0.066	0.144	297.1	50.6	58.3
11	0.15	0.05	0.761	0.060	0.121	288.9	46.1	51.5

The metrics used to evaluate the various rules were: success rate, coverage neglected, coverage occasional, mission delay, time enroute, and time committed. The success rate is simply the percentage of attempted missions that did not fail. Coverage neglected is a measure of the percentage of missions that received little to no commitment from systems while coverage occasional were those that received little to moderate commitment. Mission delay is the time elapsed between activation and go; failed missions do not count in this metric. Time enroute measures how long systems spend enroute to missions while time committed measures how long systems spend committed to a mission while waiting for the mission to achieve go. The primary metric is mission success, while the other metrics describe fair (coverage neglected and coverage occasional) and efficient (time enroute and time committed) the rules are.

When applying a single rule, there was a great variation in the success rate. The worst rule, which amounted to systems being attracted to missions that did not need their capabilities, had a success rate of zero. The best single rule (success rate of 0.433) was being most attracted to missions whose deadline was about to expire. The second-best rule (success rate of 0.426) was that of being attracted to the most popular mission. However, when combining rules, results were not as expected. When the two most successful single rules were combined, they were the worst of all the two rule combinations at 0.437. The best was 0.604 and occurred when popularity was combined with capability. The best overall rule set involved all four rules and achieved a success rate of 0.770. When looking at the rules with success rates over 0.5, only coverage neglected and mission delay correlated with the success rate. While coverage neglected correlated negatively with the success rate as might be expected, the mission delay correlated positively which was not intuitive. Changing the accept and reconsider rates showed that the results were not too sensitive provided that the rates were not deterministic (zero or one). When rates were zero or one, the success rate dropped substantially.

5. Conclusions

The mission success rate when systems chose missions purely at random was 0.184 while the best single rule produced a success rate of 0.433 and the best overall rule set produced a success rate of 0.770. This means that behavior created by this rule set when properly weighted significantly outperforms random behavior. This problem, as defined in the experimental setup is difficult to solve well as demonstrated by the low (under 20%) success rate of the random case. The high success rate of 77% achieved by the best weighting found in the test cases shows that this rule set produces effective self-organizing behavior in a systems-of-systems environment. However, it was also found that rules do not combine in obvious ways—sometimes a combination of relatively effective rules are less effective than a combination of effective and ineffective rules. Finally, the results showed that it was important to have a stochastic element to the decision making even with the best rules. The stochastic element creates a diversity of behavior limiting similar systems from all choosing the same missions and becoming redundant. However, too much randomness overrides the purpose of the rules and degrades performance. The effective range of the stochastic variables was found to be between 2% and 10%.

References

[1] Camazine, Scott, Jean-Louis Deneubourg, Nigel R. Franks, James Sneyd, Guy Theraulaz, and Eric Bonabeau. *Self-Organization in Biological Systems*. Princeton University Press, 2001.

[2] Schelling, Thomas C. (1971) "Dynamic Models of Segregation." *Journal of Mathematical Sociology* **1** (**2**): 143–186.

[3] Boccara, Nino. *Modeling Complex Systems*, 2nd ed., ser. Graduate Texts in Contemporary Physics. Springer, 2010.

[4] Wilensky, U. 1999. NetLogo. http://ccl.northwestern.edu/netlogo/. Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL.