Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Lecture with Computer Exercises:
Modelling and Simulating Social Systems with MATLAB

Project Report

Disaster Spreading

Patrik Kessler & Francesco Ferrara

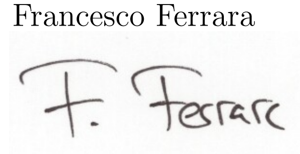
Zurich
May 2010

Eigenständigkeitserklärung

Hiermit erkläre ich, dass ich diese Gruppenarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen-Hilsmittel verwenden habe, und alle Stellen, die wörtlich oder sinngemäss aus veröffentlichen Schriften entnommen wurden, als solche kenntlich gemacht habe. Darüber hinaus erkläre ich, dass diese Gruppenarbeit nicht, auch nicht auszugsweise, bereits für andere Prüfung ausgefertigt wurde.

Patrik Kessler

Francesco Ferrara

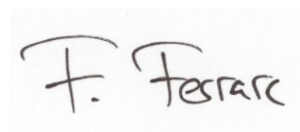


Agreement for free-download

We hereby agree to make our source code for this project freely available for download from the web pages of the SOMS chair. Furthermore, we assure that all source code is written by ourselves and is not violating any copyright restrictions.

Patrik Kessler

Francesco Ferrara



Contents

1 Introduction and Motivations

Disasters have big influence on todays society, as most infrastructures, organizations and communication systems in modern societies are supported by large and complex networks, there is big interest in being able to simulate and predict their outcome.
Many disasters show characteristic scenarios, where one strong initial event triggers a failure avalanche, which spreads in a cascade-like manner within a network and has finally a large impact on the system or at least parts of it.[5]
The goal of this work is to determine the influence of damage on the recovery process, an extension to the model used in [5]. The extension reduces the ability of regenerating as it gets more damaged.

2 Description of the Model

2.1 Basic Model[2, 3, 5]

A network consists of system components, represented by nodes. The nodes are connected with one another by so called vertexes through which damage propagates. This vertexes represent functional or structural dependencies between the components as written in [5]. The damage is modeled continuously. For $x_i = 0$, the component is working properly. If a damage or a perturbation of the node occurs, x_i rises. A node is modeled to fail almost instantly if the sum of the external and internal disturbances exceeds a particular threshold as seen in [2, 3, 5]. It is assumed to have nonlinear dynamics, described in (1).

$$\frac{\mathrm{d}x_i}{\mathrm{dt}} = -\frac{x_i}{\tau} + \Theta\left(\sum_{j\neq i} \frac{M_{ij}x_j(t-t_{ij})}{f(O_i)}\mathrm{e}^{-\beta t_{ij}/\tau}\right) + \xi_i(t) \tag{1}$$

The parameters of the dynamics are described below:

- $-\frac{x_i}{\tau}$ is the ability of the component to heal. $\frac{1}{\tau}$ is the recovery rate.
- M_{ij} weights the damage coming from another node. It represents the dependency between the components.
- t_{ij} is a time delay. The damage needs time to be propagated from one component to another.
- $\mathrm{e}^{-\beta t_{ij}/\tau}$ is a damping factor. Since the damage needs time for the propagation, it means that it has to travel a distance. During this distance, some damage is lost.
- $f(O_i)$ is the out-degree of the node i. It determines the internal robustness of a node (e.g. a node with many connections has to be more robust than others since it gets more damage than a node with less connections).
- $\xi_i(t)$ is an uniformly distributed internal disturbance.

The function $\Theta(x)$ is a nonlinear function that represents the damage dynamics of the component induced by its external disturbances. It has a behavior that resembles a sigmoid function. Important is that $\Theta(0) = 0$. These dynamics are plotted for different values of α and a fixed θ_i in Figure 1.

$$\Theta(x) = \frac{1-\mathrm{e}^{-\alpha x}}{1+\mathrm{e}^{-\alpha(x-\theta_i)}} \tag{2}$$

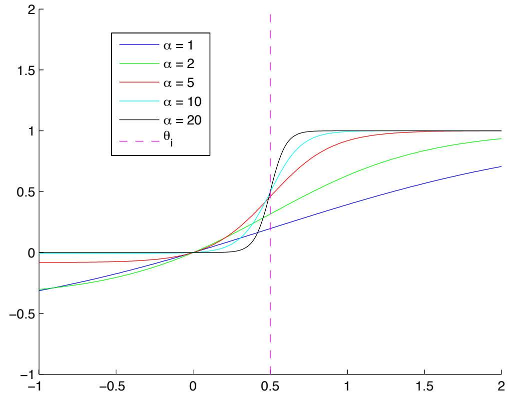


Figure 1: Sigmoid behavior of $\Theta(x)$.

The out-degree function is described by

$$f(O_i) = \frac{aO_i}{1 + bO_i} \tag{3}$$

where a and b are fit parameters.

If the input to a node is a pulse, the node will be first damaged. Then, the damage is propagated through the neighboring nodes and simultaneously, the node recovers as soon as the input goes back to zero. As long as there are internal disturbances, the state of the node will never be zero.
If the disturbance is constant, the state of the node will reach a constant value. With the propagation, as soon as a steady-state is achieved, every node will be at a constant value.
In the last two statements, a network with one-directional vertexes and no loops is assumed.

2.2 Extended Model

Generally, if a component is damaged excessively, it breaks. As soon as it is broken, the ability to recover disappears.

The idea was to restrict the recovery of a component, as soon as it is damaged. As a first approach, the regeneration has been modeled to be piecewise continuous (see equation (4) and (5)).

$$\frac{\mathrm{d}x_i}{\mathrm{dt}} = -\frac{x_i}{\tau^\star} + \Theta\left(\sum_{j\neq i} \frac{M_{ij}x_j(t-t_{ij})}{f(O_i)} \mathrm{e}^{-\beta t_{ij}/\tau}\right) + \xi_i(t) \tag{4}$$

where

$$\tau^\star = \begin{cases} \frac{1}{\tau} & \text{if } x_i < \theta^\star_{i1} \\ \frac{1}{2\tau} & \text{if } x_i < \theta^\star_{i2} \\ \vdots & \quad\vdots \\ \frac{1}{n\tau} & \text{if } x_i < \theta^\star_{in} \end{cases} \tag{5}$$

$\theta_{ik}, k = 1, 2, \ldots, n$ are individual thresholds that determine when the recovering rate changes.
This approach was discarded, because it is simpler to have a continuous function for these dynamics.

Finally, the recovery dynamics have been modeled with sigmoid function that is similar to $\Theta(x)$ (see Equation (6)). Using this function, the recovery will stop abruptly as soon as one particular threshold is reached or smoothly, depending on the choice of the parameter α^{reg}.

$$\frac{\mathrm{d}x_i}{\mathrm{dt}} = -\frac{x_i}{\tau} \underbrace{(1-\Theta^{reg}(x_i))}_{\bar{\Theta}(x)} + \Theta\left(\sum_{j\neq i} \frac{M_{ij}x_j(t-t_{ij})}{f(O_i)} \mathrm{e}^{-\beta t_{ij}/\tau}\right) + \xi_i(t) \tag{6}$$

where

$$\Theta^{reg}(x) = \frac{1-\mathrm{e}^{-\alpha^{reg}x}}{1+\mathrm{e}^{-\alpha^{reg}(x-\theta_i^{reg})}} \tag{7}$$

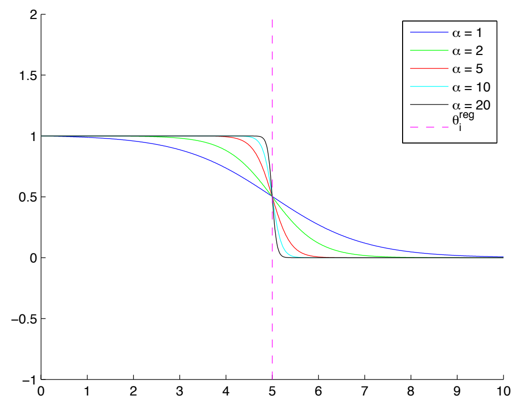


Figure 2: $\bar{\Theta}(x) = 1 - \Theta^{reg}(x)$

3 Implementation

3.1 Simulink

The model has been implemented with Simulink, a powerful MATLAB toolbox. Simulink is a graphical environment where dynamical systems can be first implemented in an intuitive way and then simulated. A model is built by connecting various elements that are represented by blocks. The signals respectively the dynamical parts of the systems are represented by arrows. This results in a so called block diagram. One important block used in this model is the so called integrator (see Figure 3). As the name says, it integrates a signal using one of the ODE solver of MATLAB.

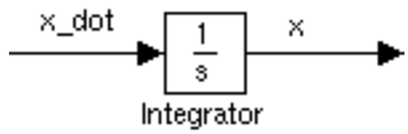


Figure 3: Integrator

The model can either be simulated in the graphical interface of Simulink itself or it can be controlled by a MATLAB code that is able to run it and which can evaluate the resulting data automatically.

3.2 Disaster Spreading in Simulink

The model has been implemented as seen in Figure 4, 5, 6 and 8. Simulink has the ability to mask block diagrams, e.g. the model in Figure 5 can be represented by the block seen in 4. It follows that components can be simply connected to one another without making the network model look too chaotic. The functions $\Theta(x)$ as well as $f(O_i)$ and the argument of $\Theta(x)$, $\sum\limits_{j \neq i} \frac{M_{ij} x_j (t - t_{ij})}{f(O_i)} \mathrm{e}^{-\beta t_{ij}/\tau}$, have been masked too for a better overview. The red ovals in the models represent the inputs and outputs of the respective mask.

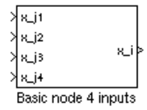


Figure 4: Single component/node.

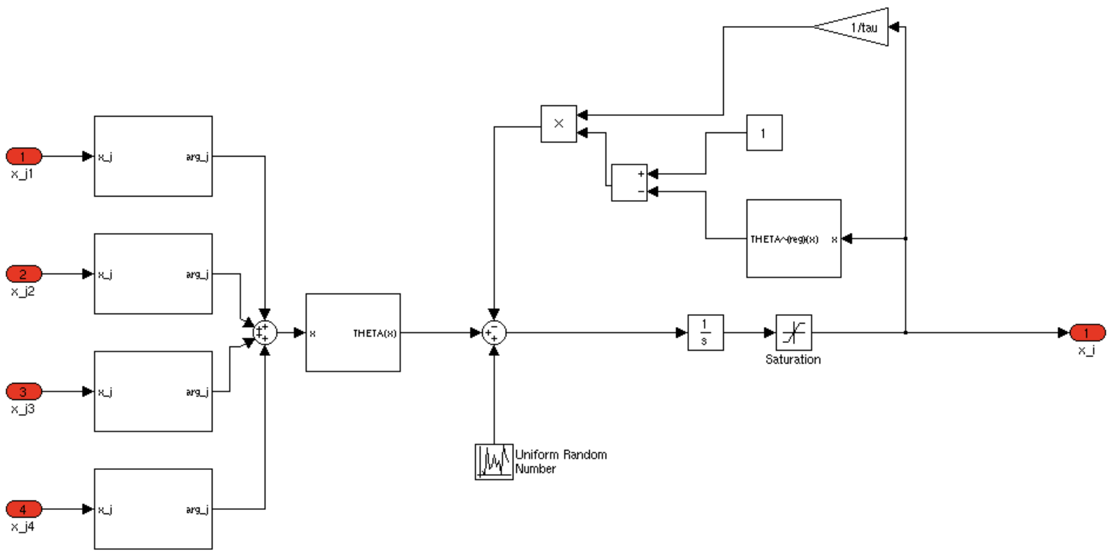


Figure 5: The dynamics of the node. The number generator stands for the internal disturbance while the saturation limits the damage on $(-\infty, 1]$. This is so that the damage doesn't get to infinity when a component is destroyed.

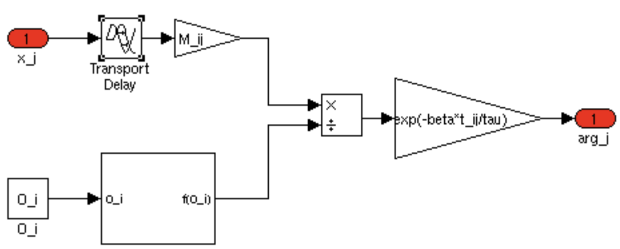


Figure 6: Weighting and delay of the inputs. The transport delay represents the time shift t_{ij} of the incoming disturbance (only because of $x(t - t_{ij})$ and not the damping part).

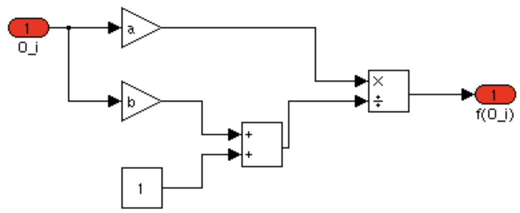


Figure 7: The function of out-degree $f(O_i)$.

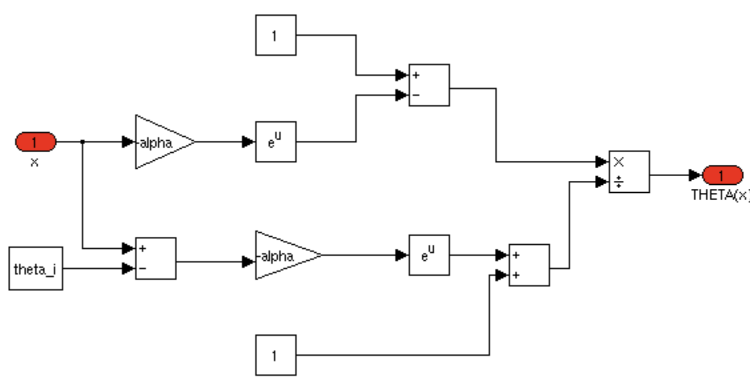


Figure 8: The sigmoid $\Theta(x)$.

4 Simulation Results and Discussion

For purposes of illustration, the internal disturbances have been removed in the main simulations. Results of simulations with internal noise will be shown in Section 4.4.

4.1 Generation of internal parameters

A component is characterized by many unknown parameters. Some simulations have been made to find out a suitable average parameter set. In these simulations, a single component has been first initialized with random parameters and then fed in with a pulse input

$$u(t) = \begin{cases} 0 & \text{if } t < 2 \\ 1 & \text{if } 2 \leq t < 6 \\ 0 & \text{if } t \geq 6 \end{cases} \tag{8}$$

for different parameter variations. In each simulation, only one parameter has been varied and the component state of each variation has been compared in a plot.

In Figure 9, it can be seen that by increasing θ_i, both the damage and the duration of the recovery get smaller. This was expected, because by increasing this internal threshold, the component is supposed to be able to take more damage until it fails. Being more robust leads also to the ability to recover faster. From Figure 9, it might seem that at a threshold $\theta_i \approx 0.6$, the component is not able to recover anymore. This is not the case because it is not located in a loop other than the internal recovery loop itself [5]. It just needs much more time to recover.

The variation of the other parameters yielded similar results, except for t_{ij}. By increasing the time that the disturbance needs to reach the component, it can be seen that also the damage that gets to it is smaller (see Figure 10). From Section 2.1, this was also an expected result.

Similar simulations have been performed to find suitable regeneration parameters.

4.2 Simulations

Four different network structures have been simulated to examine their behavior:

- a line
- a circle

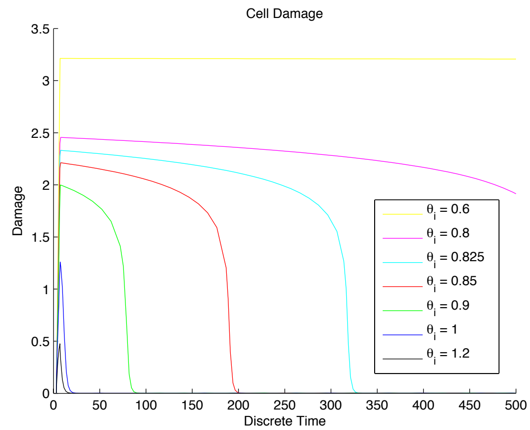


Figure 9: Variation of θ_i in the simulation for the parameter generation.

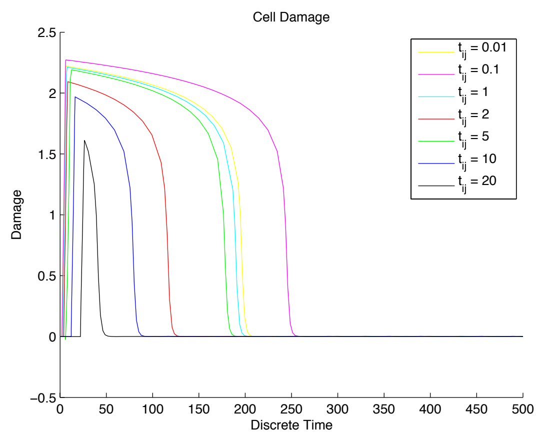


Figure 10: Variation of t_{ij} in the simulation for the parameter generation.

- a grid
- a "random" network

The goal was to find a threshold θ_i^{reg} where the network structure would collapse. The parameters used for the simulation can be found in Appendix A.

4.3 Line

Each component of this network has the same internal parameters, including connection weights and time delay. As it can be seen in Figure 11, the vertices of the network are bidirectional, which means that there exist loops. The external disturbance is fed into the first node. While decreasing θ_i^{reg}, the component receive more

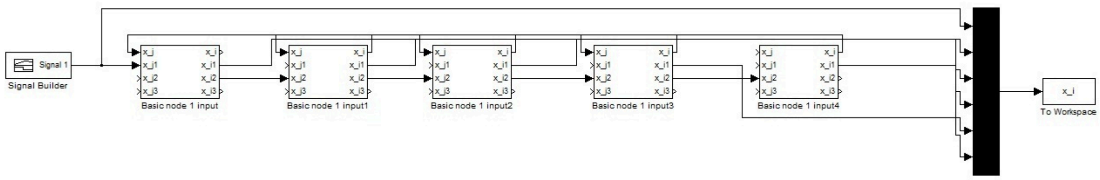


Figure 11: A straight line of network components.

damage and their recovery becomes slower(see Figure 12). The system collapses, if θ_i^{reg} falls below 0.62 (see Figure 13). This means that at some point, the component loses his recovery abilities and gets destroyed right away. The same occurs if the external disturbance is too high. The reason why the whole network collapses are the bidirectional vertices. If there were no loops, it would not matter how high the external disturbance would be. Given enough time, the whole network would recover.

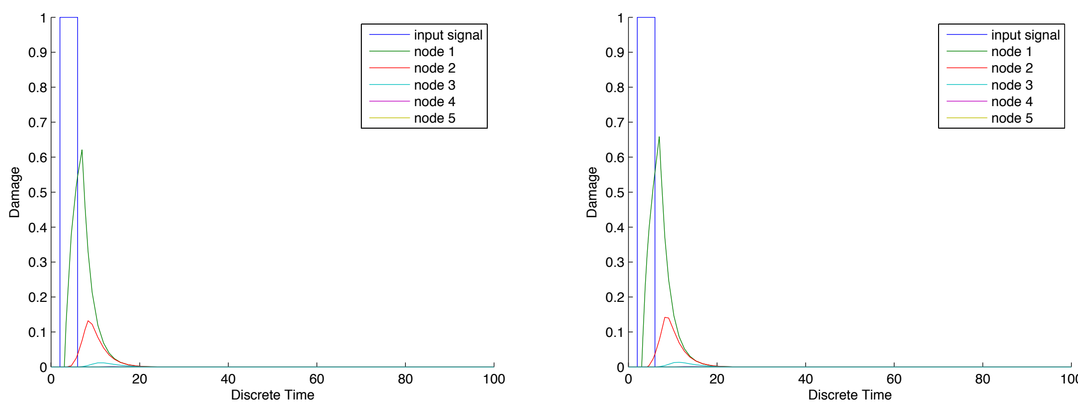


Figure 12: Without θ_i^{reg} (l.) and $\theta_i^{reg} = 0.8$ (r.).

4.4 Circle

This network structure consists of four identical components connected like a line, but the first and the last component are connected, too. The vertexes are bidirectional. This time, there is noise acting as an internal disturbance in each component. The external disturbance occurs in one component.

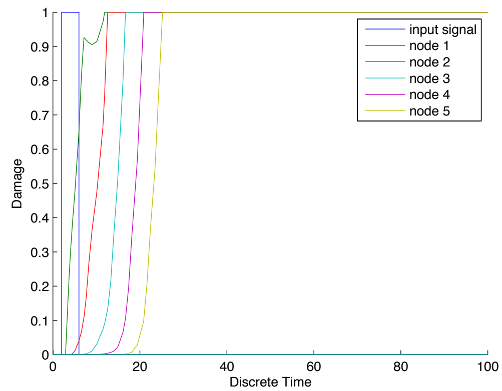


Figure 13: The line network collapses for $\theta_i^{reg} \approx 0.6$.

Figure 16 illustrates how the network recovery abilities decay with the lowering

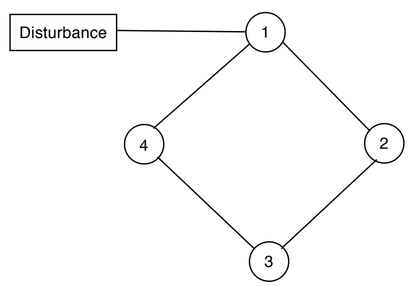


Figure 14: The circle network.

of θ_i^{reg}. The network fails when $\theta_i^{reg} \approx 0.2$. It does not fail anymore for the same θ_i^{reg}, if the internal disturbance is removed (see Figure 17). The internal disturbance lowers the robustness of a component. If a network with such a circle structure is big enough, the simulation results would be like the outcome of the line network simulation.

4.5 Grid

The Simulink Model of a network with a 5x5 grid structure can be seen in Figure 18. All nodes have the same parameters and are connected bidirectionally to their direct neighbor. If a neighbor does not exist, the respective disturbance will be zero. Again, by decreasing θ_i^{reg}, the systems recovery gets worse (see Figure 19). The system collapses, if $\theta_i^{reg} < 0.73$, as it can be seen in Figure 20.

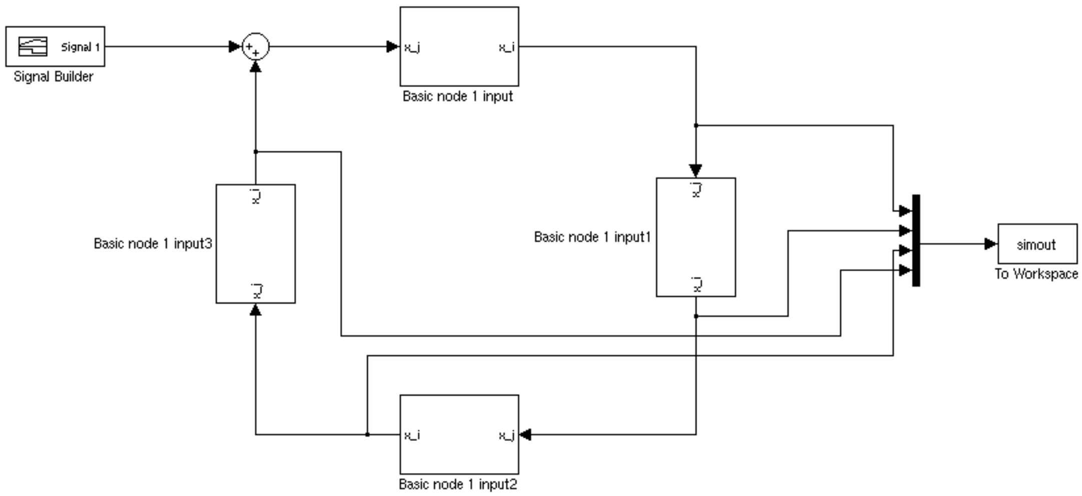


Figure 15: The Simulink model of the circle network.

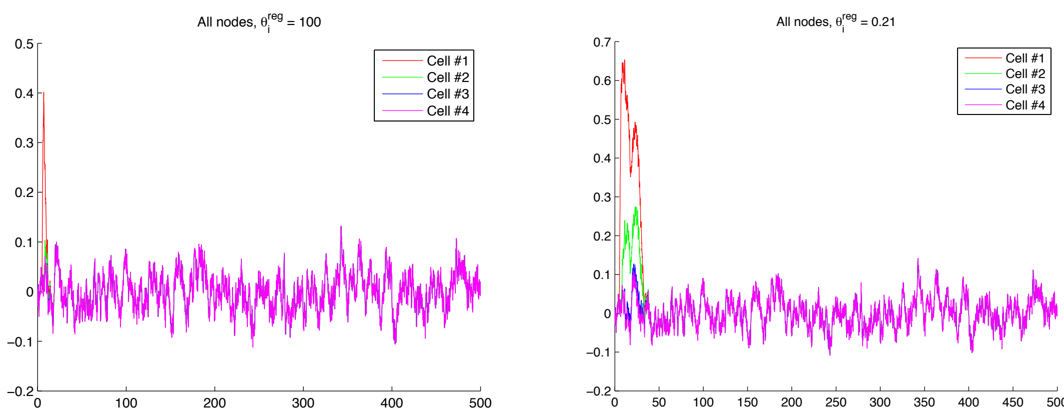


Figure 16: Simulation of the circle network. $\theta_i^{reg} = 100$ (l). and $\theta_i^{reg} = 0.21$ (r.).

The θ_i^{reg}, where the System collapses is larger than the one from the line structure, because there's a higher connectivity between the nodes, which leads to higher damage propagation.

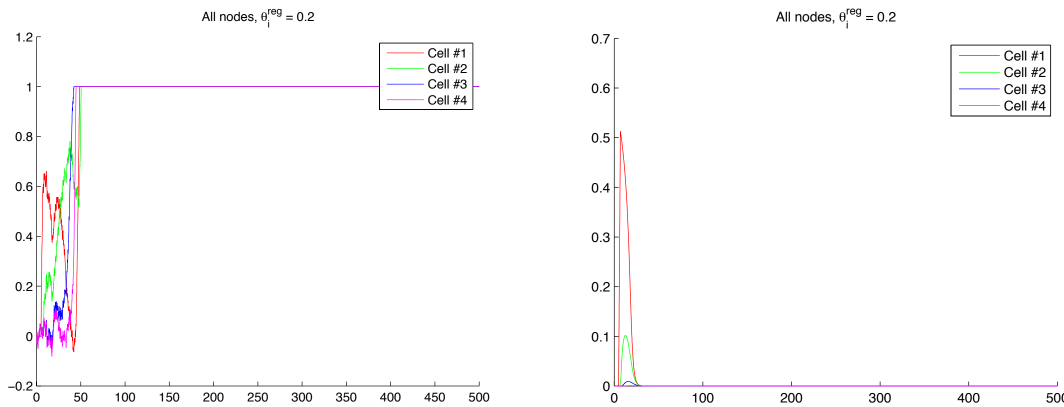


Figure 17: The circle network fails at $\theta_i^{reg} \approx 0.2$ (l.). For the same θ_i^{reg}, it does not fail, if there is no internal disturbance (r.).

Figure 18: Grid network.

4.6 "Random" Network

The Simulink model of the "random" network shown in Figure 22 can be seen in Figure 23. All the connections are bidirectional.In the first simulation, the internal

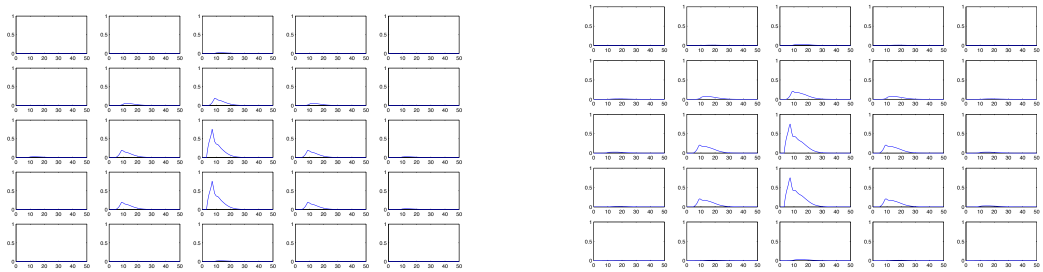


Figure 19: Without θ_i^{reg} (l.) and $\theta_i^{reg} = 0.73$ (r.).

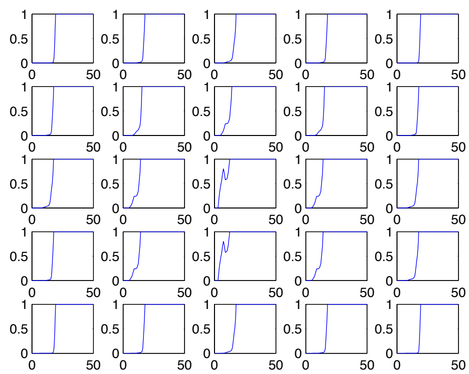


Figure 20: $\theta_i^{reg} = 0.72$

parameter are the same for every component. A second simulation has been performed where each component has a different disturbance weight M_{ij} and out-degree O_i.

By lowering θ_i^{reg}, the network loses recovery abilities (see Figure 24). As it can be seen in Figure 25, the network collapses as soon the threshold $\theta_i^{reg} \approx 0.205$. This means that even if there are loops it still remains quite robust compared to the grid network. This is somehow expected. Even if there are loops, the connectivity between the components is much lower than the connectivity of the grid network.

If the network components are different from one another, the robustness of the network depends on how robust the single components are to external and to internal disturbances and how strong the damage is propagated. With the parameters used in Appendix A the network is extremely robust, as it can be seen in Figure 26.

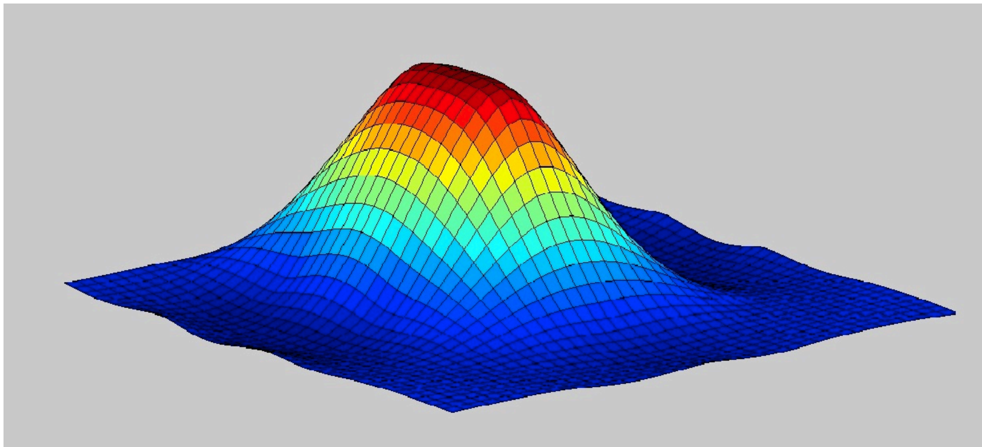


Figure 21: Other visualization of the grid network simulation.

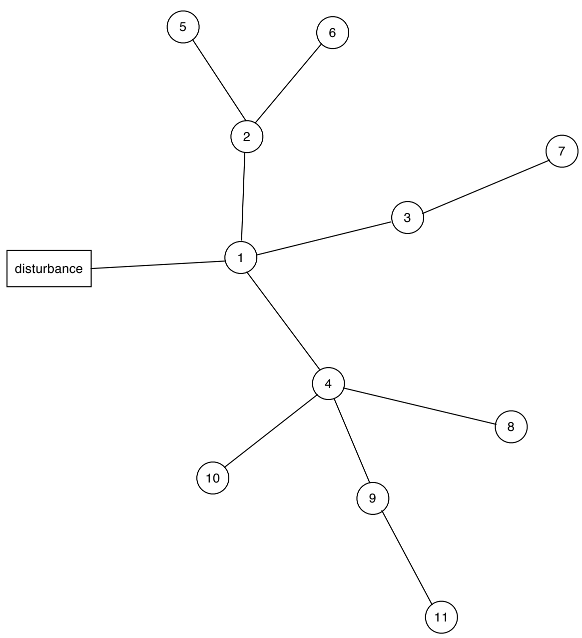


Figure 22: Symbolic representation of the “random” network.

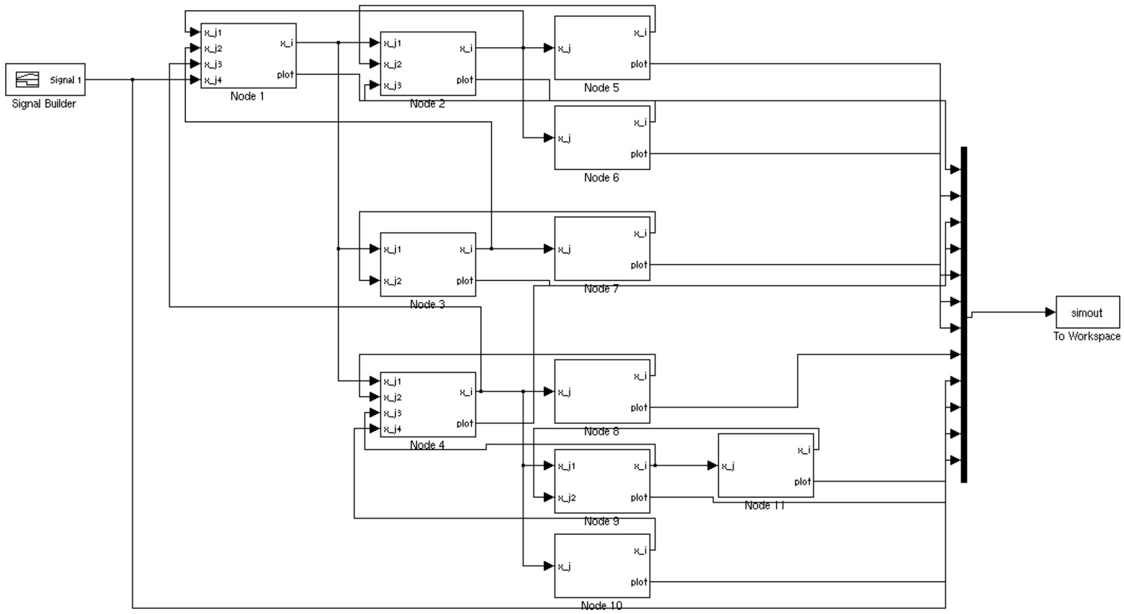


Figure 23: Simulink model of the "random" network.

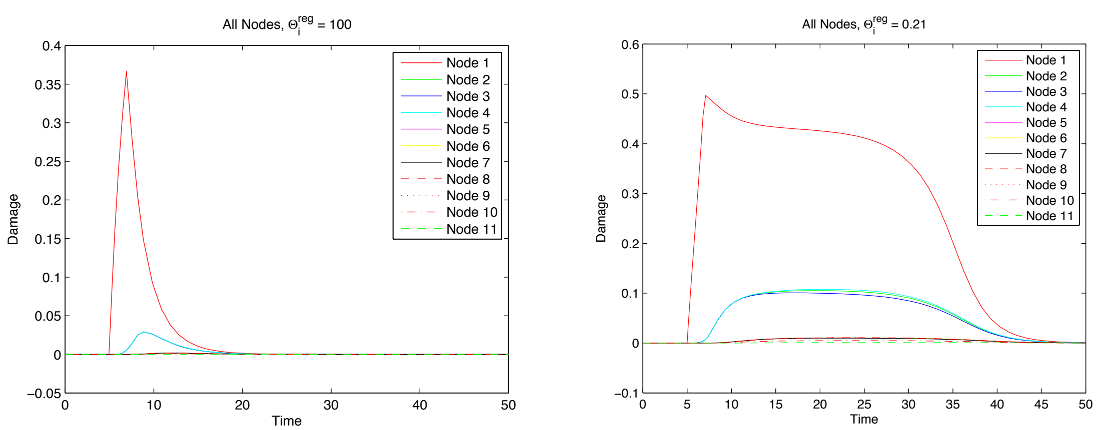


Figure 24: $\theta_i^{reg} = 100$ (the same, as the extension of the recovery would miss, l.) and $\theta_i^{reg} = 0.21$ (r.). Notice that for higher θ_i^{reg}, the damage is lower and the recovery faster.

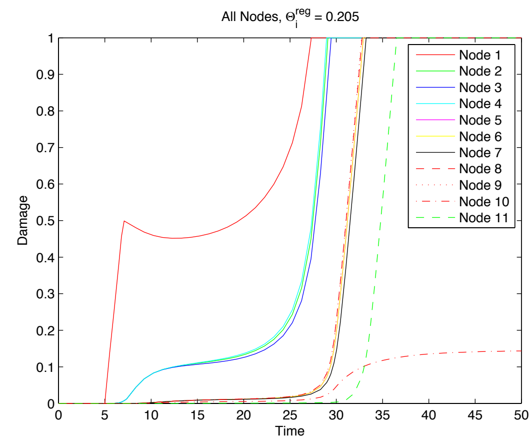


Figure 25: The "random" network fails.

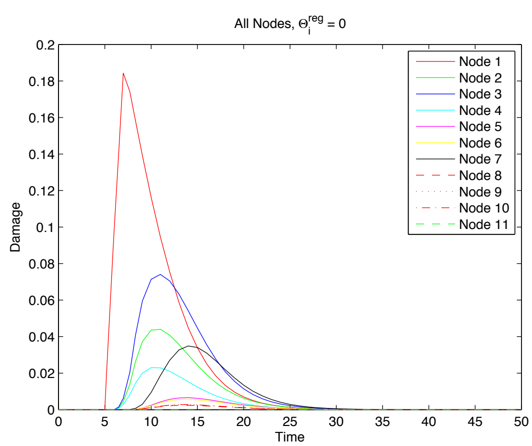


Figure 26: Simulation of the "random" network with components that differ from one another. Notice how robust the network is with $\theta_i^{reg} = 0$.

5 Summary and Outlook

As seen in the simulations, this model suits well to cases where a components recovery abilities decay with increasing damage.
However, the Simulink model is not suited for the construction of big networks, because the components have to be connected by hand. It can become very chaotic, if the connectivity in the model is big.
There is a potential in the simulation of grid networks where the components are only connected to their neighbors. As seen in Figure 21 of Section 4.5, The 5x5 grid network has been visualized with a finer resolution. It looks like the network would consist of much more components. If there is the possibility of obtaining the internal component parameters through the grid resolution (i.e. the number of nodes) or vice versa, the network can be built connecting a small number of components in Simulink and then it can be extended by adjusting the grid resolution to the desired network parameters. It would reduce complexity and would be computationally more efficient.

A Parameters

A.1 Line

a	=	1.95	b	=	1
t_{ij}	=	1	M_{ij}	=	0.7
O_i	=	0.95	α	=	5
α^{reg}	=	10	β	=	0.01
θ_i	=	0.85	τ	=	2

A.2 Circle

A.2.1 Simulation with Noise

a	=	1.95	b	=	1
t_{ij}	=	1	M_{ij}	=	0.7
O_i	=	0.95	α	=	5
α^{reg}	=	10	β	=	0.01
θ_i	=	0.9	τ	=	2
$noiseStdDev$	=	0.2			

A.2.2 Simulation without Noise

a	=	1.95	b	=	1
t_{ij}	=	1	M_{ij}	=	0.7
O_i	=	0.95	α	=	5
α^{reg}	=	10	β	=	0.01
θ_i	=	0.9	τ	=	2

A.3 Grid

a	=	1.95	b	=	1
t_{ij}	=	1	M_{ij}	=	0.7
O_i	=	0.95	α	=	5
α^{reg}	=	10	β	=	0.01
θ_i	=	0.85	τ	=	2

A.4 "Random" Network

A.4.1 Identical Components

a	=	1.95	b	=	1
t_{ij}	=	1	M_{ij}	=	0.7
O_i	=	0.95	α	=	5
α^{reg}	=	10	β	=	0.01
θ_i	=	0.9	τ	=	2

A.4.2 Different Components

a	=	1.95	b	=	1
t_{ij}	=	1	M_{11}	=	0.6
M_{12}	=	0.7	M_{13}	=	0.9
M_{14}	=	0.4	M_{21}	=	1.3
M_{25}	=	0.8	M_{26}	=	0.7
M_{31}	=	1.4	M_{37}	=	0.5
M_{41}	=	1.2	M_{48}	=	0.3
M_{49}	=	0.8	M_{410}	=	0.2
M_{52}	=	0.9	M_{62}	=	0.8
M_{73}	=	1.8	M_{84}	=	0.7
M_{94}	=	0.9	M_{911}	=	0.2
M_{104}	=	0.8	M_{119}	=	0.5
O_1	=	1.5	O_2	=	0.8
O_3	=	0.6	O_4	=	1.5
O_5	=	0.5	O_6	=	0.5
O_7	=	0.5	O_8	=	0.5
O_9	=	0.6	O_{10}	=	0.5
O_{11}	=	0.5	α	=	5
α^{reg}	=	10	β	=	0.01
θ_i	=	0.9	τ	=	2

References

[1] K. Peters,L. Buzna,D. Helbing: *Modelling of cascading effects and efficient response to disaster spreading in complex networks*, Int. J. Critical Infrastructures, Vol. 4, Nos. 1/2, 2008

[2] I. Simonsen,L. Buzna,K. Peters,S. Bornholdt,D. Helbing: *Transient Dynamics Increasing Network Vulnerability to Cascading Failures*, PHYSICAL REVIEW LETTERS 30 MAY 2008

[3] K. Peters,L. Buzna,D. Helbing,H. Ammoser,C. Khnert: *Efficient response to cascading disaster spreading*, PHYSICAL REVIEW E 75, (2007)

[4] C. Khnert,D. Helbing: *Assessing interaction networks with applications to catastrophe dynamics and disaster management*, Physica A 328 (2003)

[5] L. Buzna,K. Peters,D. Helbing: *Modelling the dynamics of disaster spreading in networks*, Physica A 363 (2006)