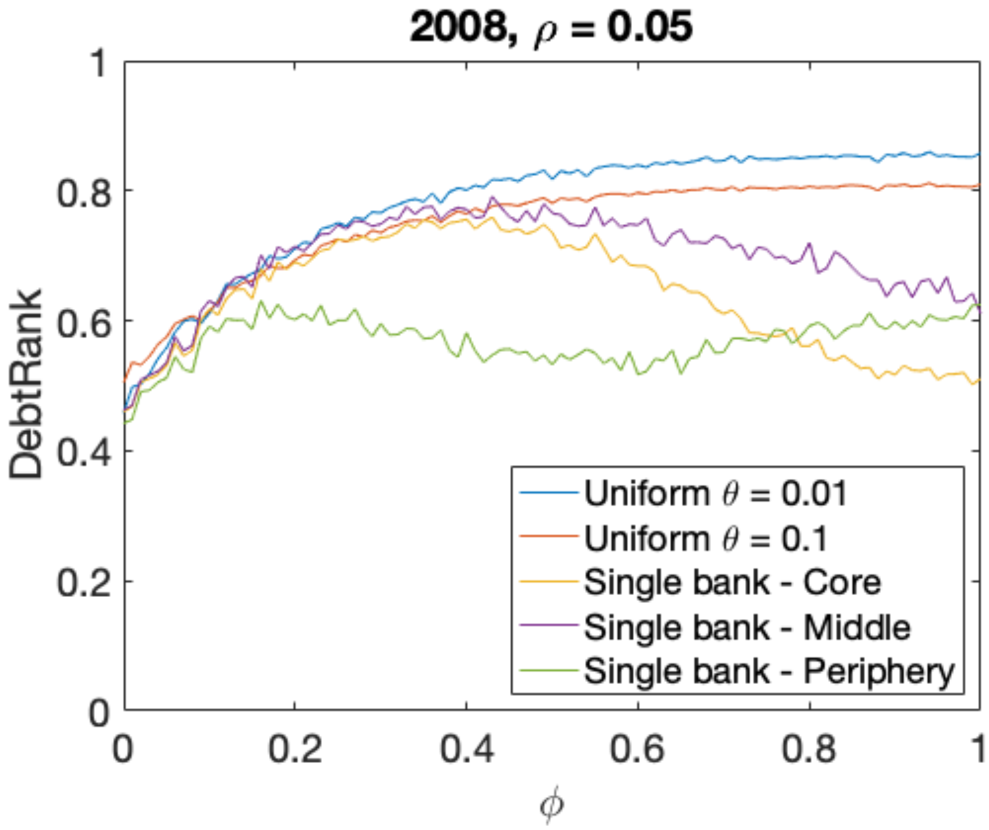
- For all figures here:
  - Y-axis is the value of DebtRank
  - X-avis is ϕ (the core-periphery parameter), where $\phi = 0$ ($\phi = 1$) means Erdos-Renyi (’pure’ Core Periphery’) structure.
  - ρ is the density of the network

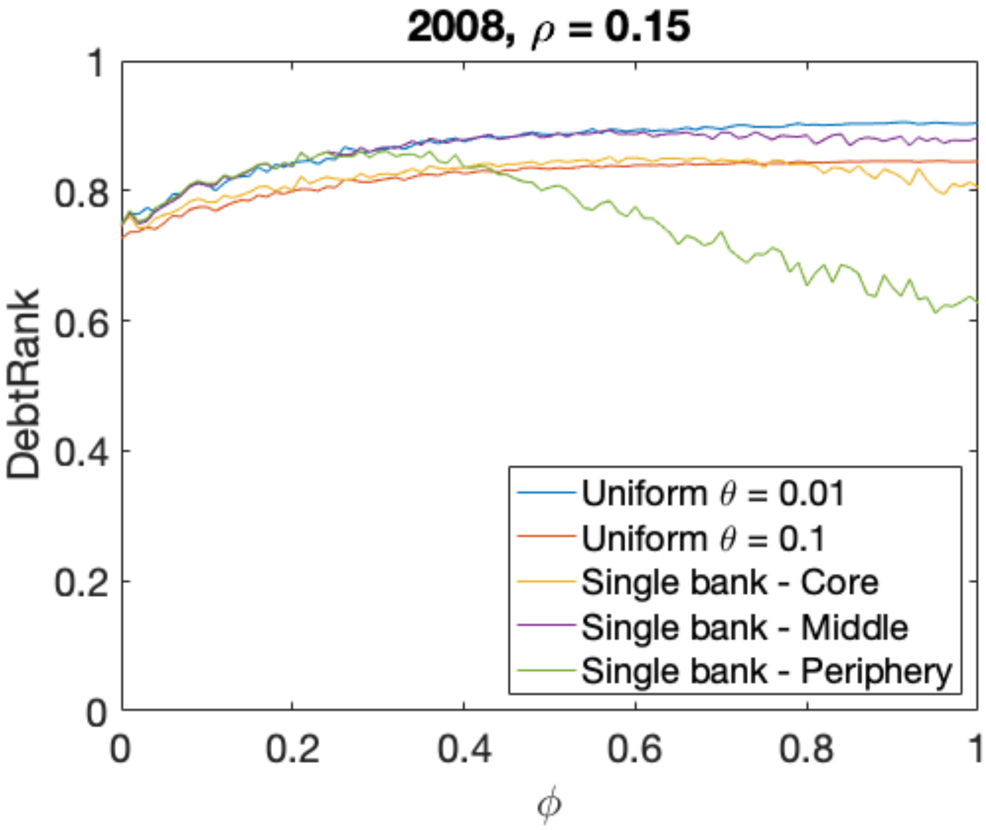
- I consider two different types of shock:
  - Uniform shock, where all banks are given similar amount of shock (θ)
  - Single bank shock. I look at top-10 banks in the core, middle, and periphery. I define those according to the number of link (in and out). In one simulation, only 1 bank is given a shock. For each group, DebtRank then is calculated as the average of 10 simulations (because there are 10 banks for each group).

- Unless stated otherwise, I calculate DebtRank by subtracting the value of the initial shock, and until the algorithm reachs convergence.

Pg. 3 Description

- The figures are similar with those in the paper (Fig. 3), but I change slightly the value of θ and ρ.
- The fundamental differences between those in here and in the paper:
  - I run the simulations with new (correct) network generation codes





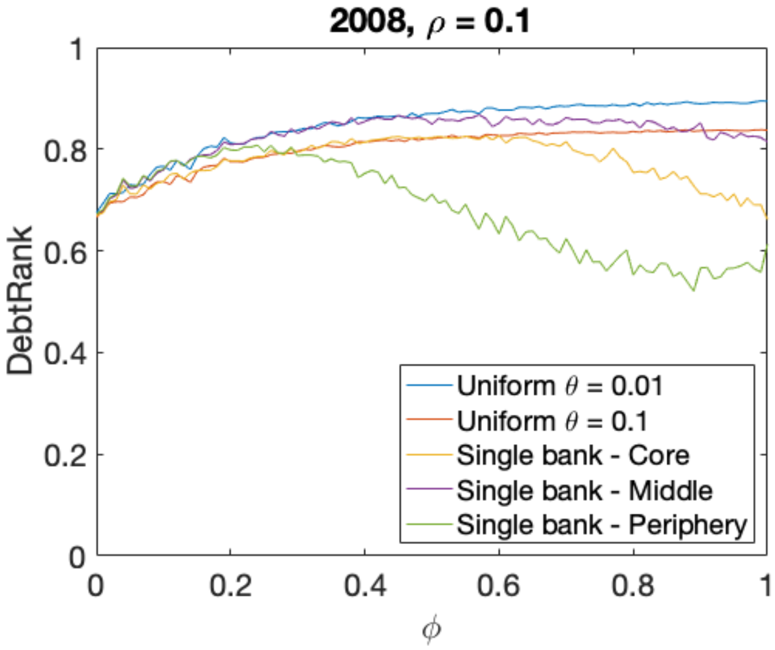
DebtRank as a function of ϕ (core-periphery parameter),
for different type of shock and ρ (density)

What (I think) I have seen:

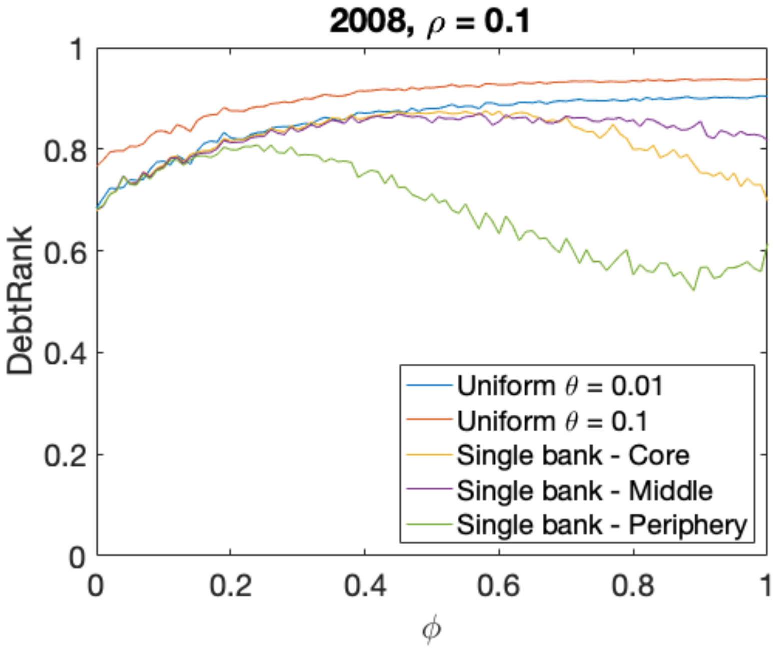
1. DebtRank increases in denser network
2. The various behavior of DebtRank for different type of shock (in term of a function of ϕ):
   1. Uniform shock: monotonically increases
   2. Single bank: non-monotonic.
3. As we have more core-periphery structure (ϕ closer to 1), the DebtRank from single-shock (core) is lower compared to single shock (middle) and uniform shock.

In the next figures, I want to look at:

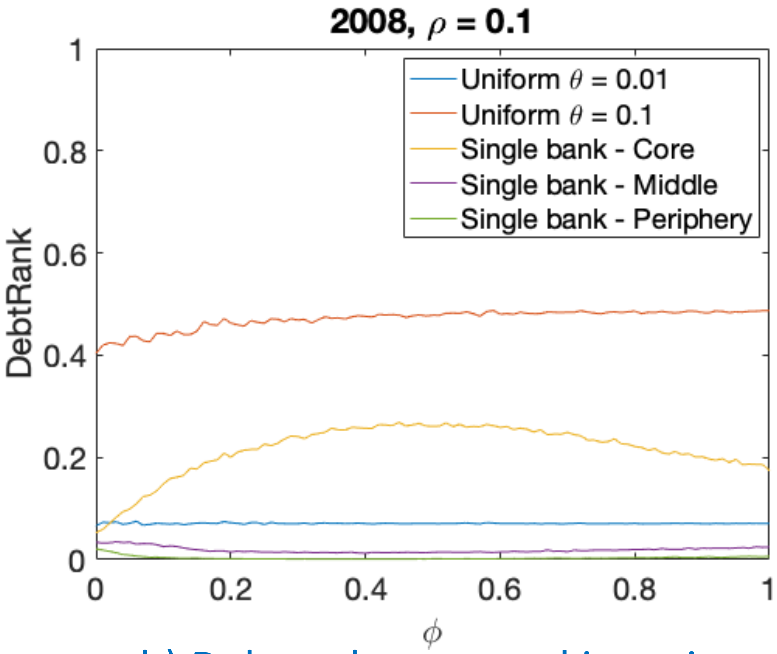
1. Debtrank without subtracting the initial shock (panel b)
2. Debtrank during the 2^{nd} iteration (panel c)
3. The number of iteration required until the algorithm converges (panel d)



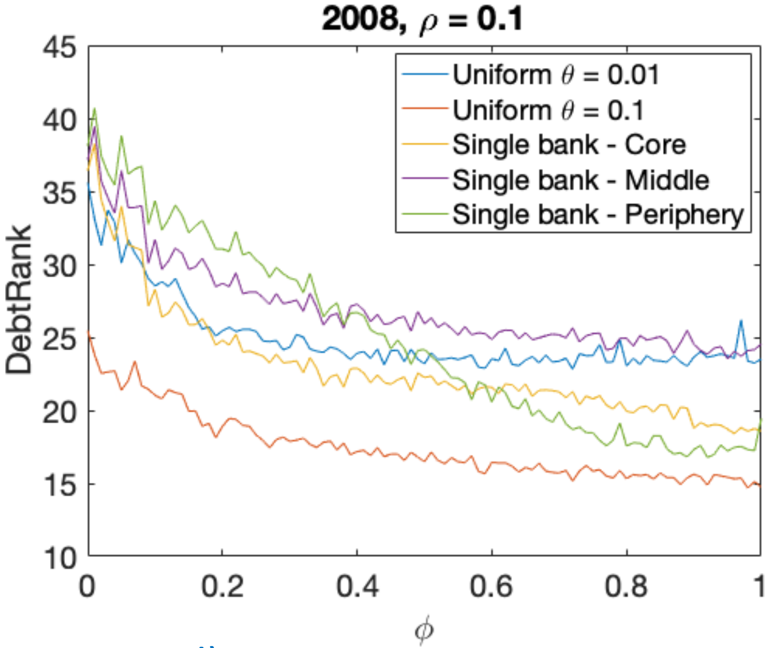
a) DebtRank



b) Debtrank with initial shock



b) Debtrank at second iteration

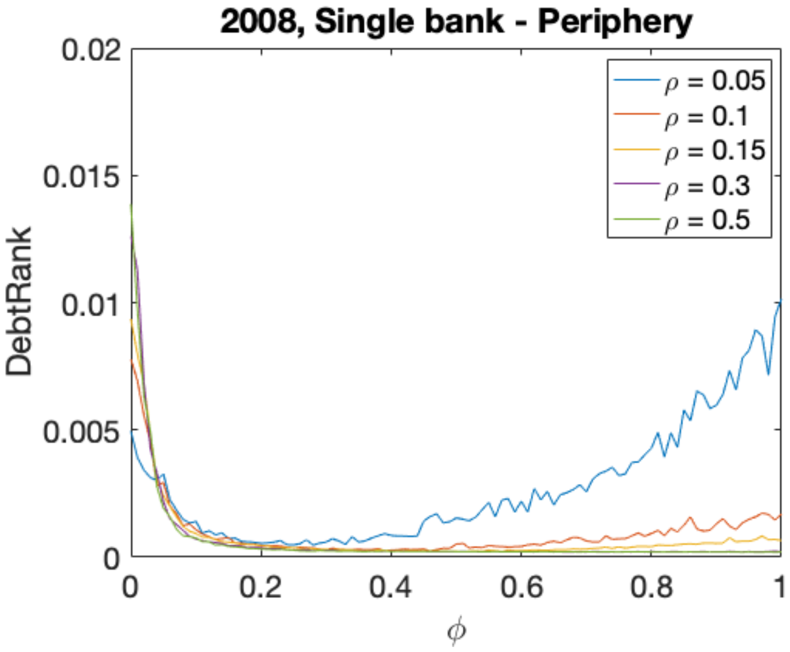


d) n_iteration to converge

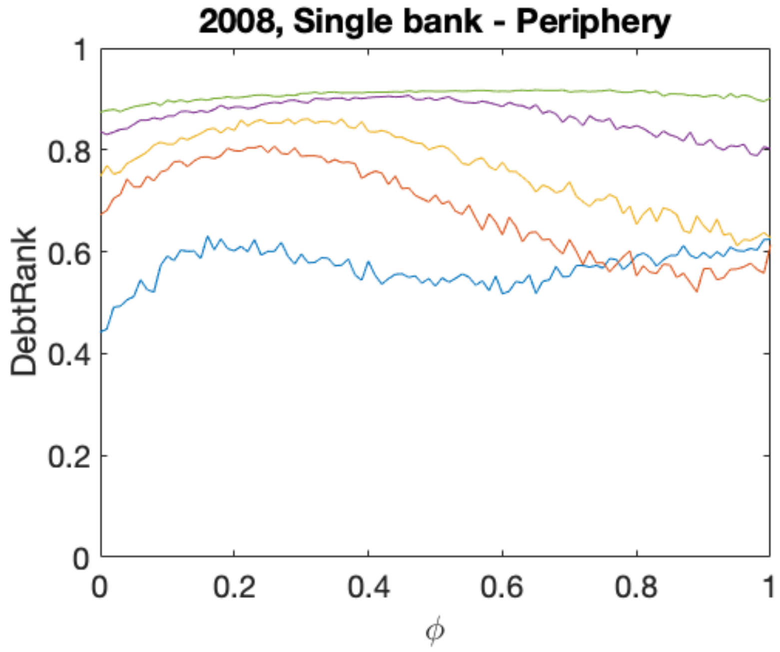
What (I think) I have seen:

1. The order of network robustness to the shock is different between 2nd iteration and final value (until the algorithm converges).
    1. For example, single-shock (bank) is the 2nd most fragile during the 2nd iteration, but it becomes the 2nd most robust during the final iteration.
2. Previously we have seen that, in some cases, the DebtRank is non-monotonic(as a function of ϕ). Here we see that the number of iteration to converge instead, always, monotonically decreases.

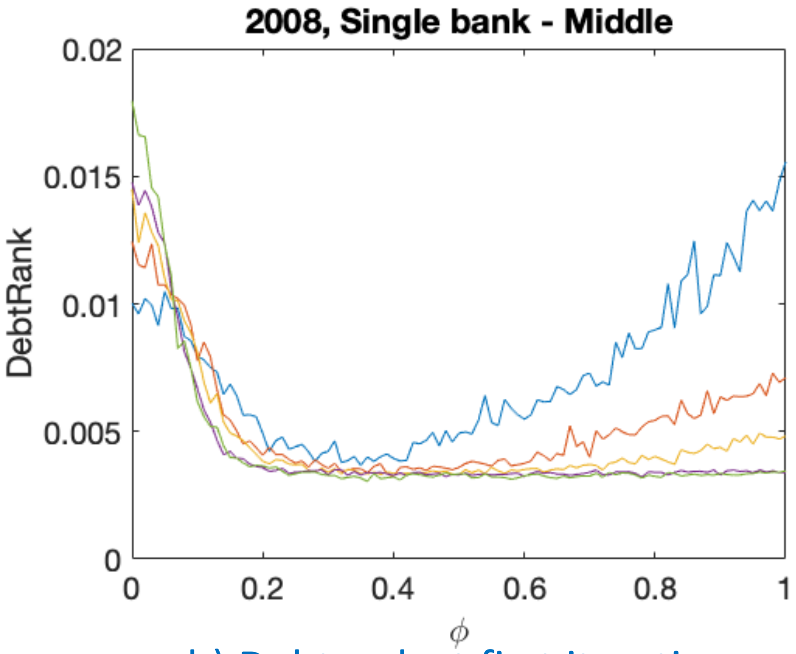
- Previously, we have seen that as the network becomes more fragile as it becomes denser.
- Here I want to check whether we see similar results during 1^{st} iteration.
- In the next page, you will see 4 different figures of DebtRank:
  i. Single shock on Periphery (Middle) banks in the top (bottom) panels
  ii. DebtRanks at the first iteration (the final step) in the left (right) panels



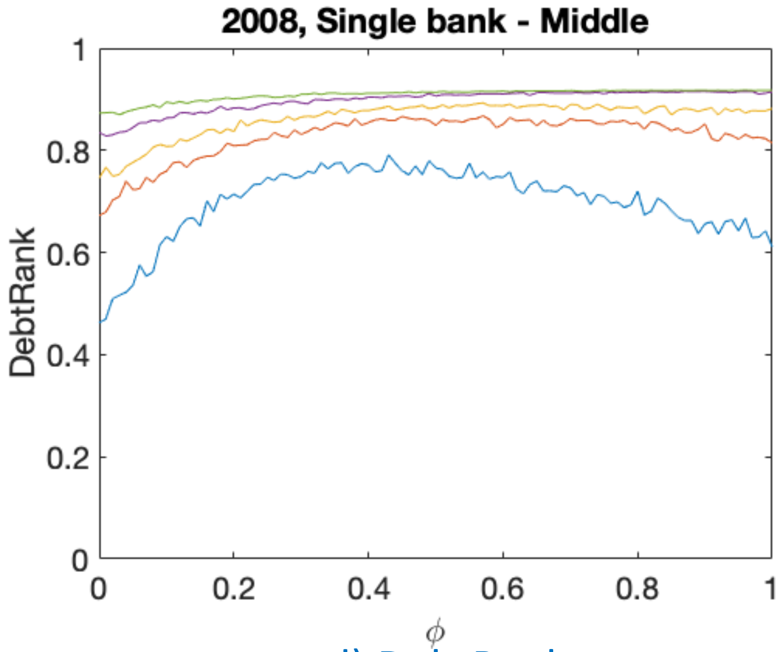
a) DebtRank at first iteration



b) Debtrank



b) Debtrank at first iteration



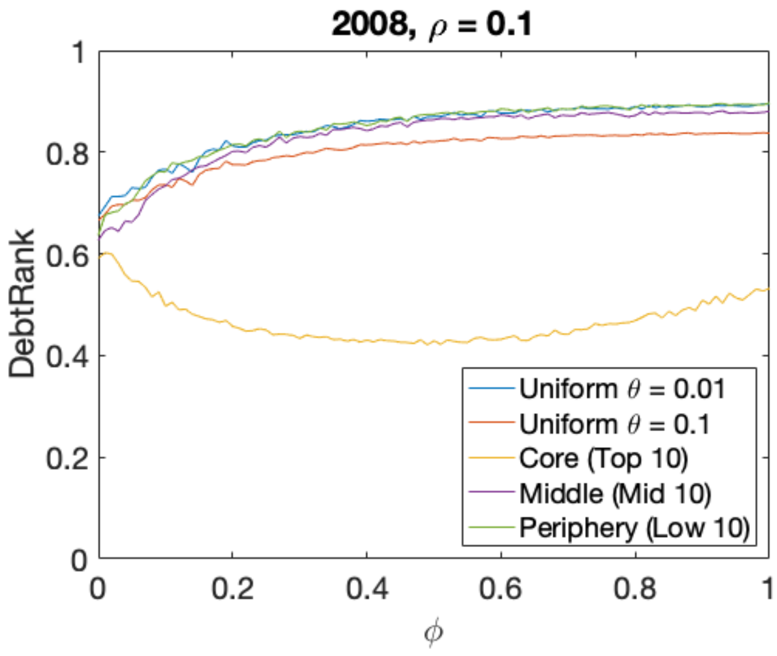
d) DebtRank

What (I think) I have seen:

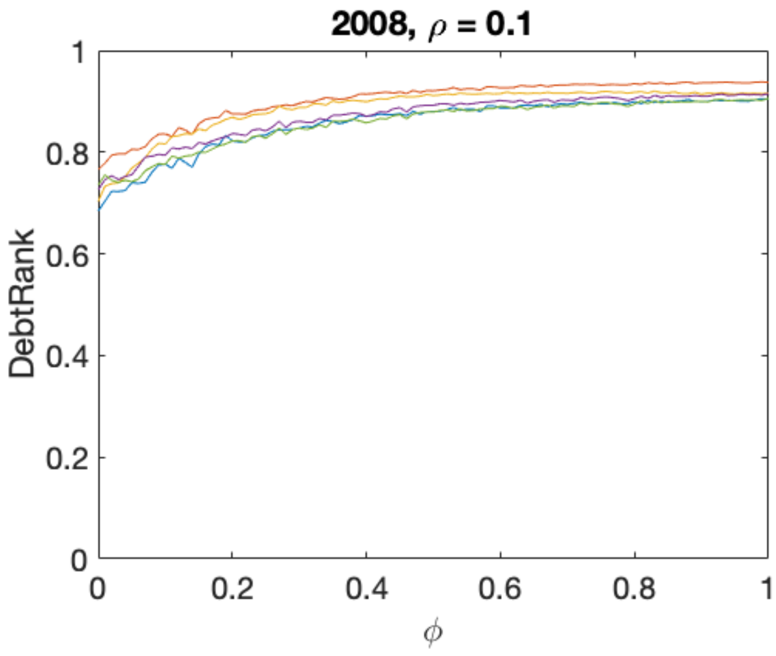
1. It turns out that for single-shock (Periphery and Middle banks), the network with lower density becomes more fragile during the 1st iteration. However, as the iteration continues until it converges, they become more robust compared to those with higher density.
2. I only observe this behavior for single shock on Periphery and Middle banks, but not for single shock on Core nor uniform shock.

In the next page, you will see four plots. They are similar with the figures in the pg. 6, but I shock all banks in the core (middle, and periphery) instead.

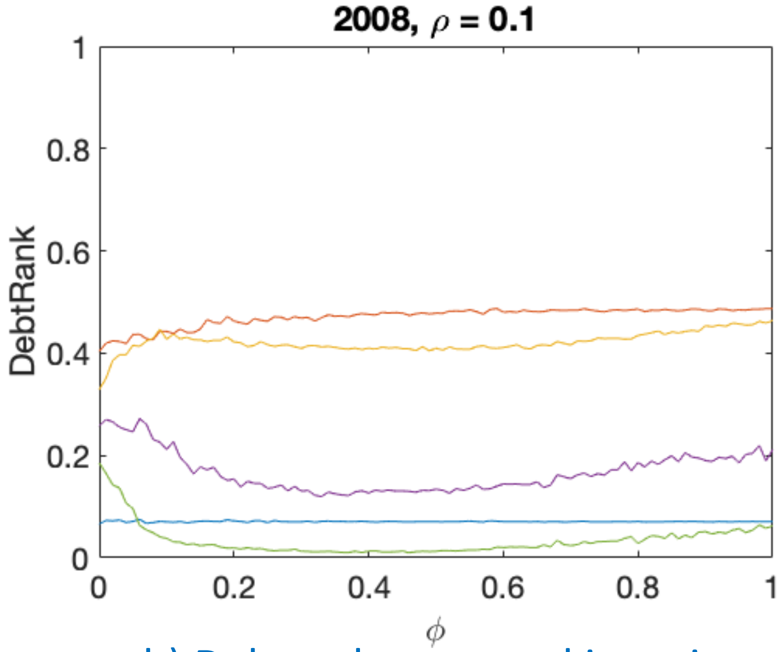
I find more or less similar results as before.



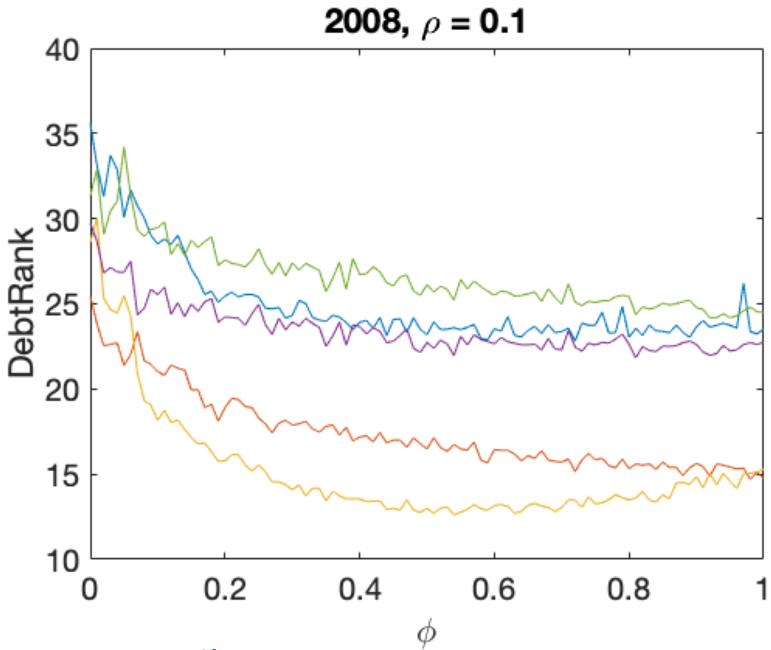
a) DebtRank



b) Debtrank with initial shock



b) Debtrank at second iteration



d) n_iteration to converge