Outflow from traffic jam in a nondeterministic two lanes system

Marcus Rickert*

April 20, 1995

Abstract

Vehicles accelerating from a high density region in a two lanes system selforganize to a throughput which is equivalent to the the maximum throughput of a closed system with periodic boundary conditions. This holds for both a symmetric and asymmetric set of rules defining the exchange and interactions of the two lanes.

*e-mail: mr@mi.uni-koeln.de

1 Introduction

In recent times cellular automata based simulations of traffic flow have gained considerable importance. By extending the range of rules from nearest neighbours to a range of 5 grid sites and introducing 6 descrete velocities $0\ldots5$ Nagel and Schreckenberg [2] have found a striking resemblence of simulation and realistic traffic behaviour. However, up to now only the model with $v_{max} = 1$ seems to be accessible for an analytic solution which has been found by Schadschneider and Schreckenberg [3].

A typical result of traffic flow simulations is a fundamental diagram of the average throughput versus the density of vehicles for a system of finite size with periodic boundary conditions. It usually contains a characteristic peak of throughput q_{max} at a critical density ϱ_c. In the single lane case Nagel has shown that vehicles released from a jam of high density $\varrho > \varrho_c$ exhibit a selforganizing behaviour [1]. By counting the vehicles that leave the system containing the traffic jam they have found that the outflow is equal to the maximum throughput of the one with periodic boundary conditions.

In this paper I have investigated the outflow behaviour in a two lanes systems. The model consists of two lanes parallel to each other with a set of rules defining the interactions of the lanes and the exchange of vehicles between the two lanes. On each lane the known single lane rules are applied. As in the single lane model the outflow again selforganizes to the maximum throughput both for the symmetric and the asymmetric rules.

2 Description of the single lane model

The system consists of one dimensional grid of sites in where each site can either be empty or occupied by a vehicle of velocity $0..v_{max}$. The velocity is equivalent to the number of sites that a vehicles advances in one time step[1]. Vehicles move only in one direction. The rules of the single lane update are those described by Nagel and Schreckenberg in []. They are listed below for the convenience of the reader. The index i denotes the number of a vehicle, $x(i)$ its position, $v(i)$ its current velocity, $v_d(i)$ its maximum speed, $pred(i)$ the number of the preceding[2] vehicle, and $gap(i) := x(pred(i)) - x(i) - 1$ the width of the gap to the predecessor. At the beginning of each time step the rules are applied to all vehicles simultaneously (parallel update). Then the vehicles are advanced according to their new velocites.

1. if $v(i) \neq v_{max}$ then $v(i) \leftarrow v(i) + 1$
2. if $v(i) > gap(i)$ then $v(i) \leftarrow gap(i)$
3. if $v(i) > 0$ and $rand < p_d(i)$ then $v(i) \leftarrow v(i) - 1$ (**SL3**)

The first rule represents a linear acceleration until the vehicle has reached its maximum velocity. The second rule ensures that vehicles having predecessors in their way slow down in order not to run into them. In the last rule a random value is used to decelerate a vehicle with a certain probability. This results in a free flow average velocity of $v_{max} - p_d$ (for $p_d \neq 1$).

3 Description of the two lanes model

The extension from one lane to two lanes brings in two new aspects:

[1] provided that there are no obstacles ahead

[2] *A precedes B* in this context means that A is followed by B

- Different 'types' of vehicles are introduced by defining different maximum velocities called *desired velocities*. In this paper two different desired velocities are used: $v_{d1} = 5$ for 'normal' vehicles which can be regarded as passenger cars and $v_{d2} = 3$ for trucks. Note that other characteristics of trucks, like e.g. a slower acceleration and a lower jam density, are not taken into account.
- Rules are introduced that define the impact of the lanes on one another and the exchange of vehicles. There are two sets of rules: one is symmetric the other is asymmetric. The basic idea for the symmetric set was to model real life traffic in the U.S. which has relaxed legislation for passing on the right lane leading to a more or less symmetric use of the lanes. The asymetric rules can be seen as an example for a country which has relatively strict laws as far as passing is concerned (e.g. Germany) which results in asymmetric behaviour.

One update cycle of the two lanes simulation consists of three phases:

1. Each vehicle checks whether it changes lanes in this time step.
2. The lane changing is executed.
3. Each vehicle is advanced according to the single lane update rules (**UP3**).

Each of the phases is applied simultaneously to *all* vehicles to accurately define a parallel update. Thus the decision of each vehicle whether to change lanes in time step n is only dependant on the configuation of vehicles which is available on the transition from time step $n-1$ to n.

3.1 Symmetric rules

Let $v_{hope}(i) = \min(v(i)+1, v_d(i))$ be the velocity that the vehicle i would have if it remained in its current lane and only the first of the three single lane rules (of section 2) were obeyed. The vehicle at position i changes to the other lane if

1. $v_{hope}(i) > gap(i)$
2. $site(x(i) - v_{max}) \dots site(x(i) + v_{hope}(i))$ of the other lane are vacant

The first rule expresses the wish of a driver to change lanes in case he cannot keep his velocity in the current lane. The second rule ensures that **a)** no other vehicle approaching from behind is blocked by the manoeuvre[3] and **b)** the velocity on the other lane will actually be increased by at least one unit.

3.2 Asymmetric rules

Let v_{hope} be defined as above. A vehicle on the left lane changes to the right lane if

1. $gap(i) > 2v_{hope}(i)$
2. $site(x(i) - v_{max}) \dots site(x(i) + v_{hope}(i))$ of the other lane are vacant

The first rule makes vehicles stay on the left lane until there is a gap which is twice as large as their current velocity. It represents drivers that are eager to pass even though there might be long queue ahead preventing them from immediate advancement. The second rule is equivalent to the symmetric case.

In one update cycle of the asymmetric rules the phase UP3 of section 3 consists of three sub phases:

[3] at least this is true for the next timestep

- For all vehicles on the right lane do the following step: let $v_s(i) = min(v(i) + 1, v_d(i), gap(i))$ be the new velocity of the vehicle. If $v_s(i)$ is greater than the velocity of the preceding vehicle on the left lane and this predecessor is at most $v_s(i)$ sites ahead [4] then the velocity of i will be adapted to the velocity of that predecessor. Thus no vehicle on the right lane can pass a vehicle on the left lane[5].
- Apply rule SL3 (see 2) and update all vehicles on the right lane.
- Update all vehicles on the left lane according to the usual single lane rules.

4 Implementation and performance issues

The symmetric simulation has been implemented for both a vector computer system (NEC SX3) and two parallel computer systems (Parsytec, Intel-IPSC 860 and Intel-Paragon). The algorithm for the vector computer uses bit coding to process 64 system sites at once. It vectorizes to aproximately 99.0 % since the implementation techniques required for bit coding have an intrinsic parallel structure. Computational speed reaches 180 MUPS[6] for the outflow simulation and 204 MUPS for periodic boundary conditions on the NEC. The asymmetric simulation is restricted to the parallel computer systems due the enormous effort to code conditional branches in bit coding. Since the performance of the vectorcomputer was considerably degraded by omitting bit coding this implementation was not further persued.

As for the parallel computer systems the implementation is of course technically sequential but *logically* parallel in order to be compared to vector version. The only aspect by which the two versions differ is the way the random numbers are generated, since normal generation techniques cannot be applied to vectorizing algorithms. All simulations on the Parsytec were made using a 256-nodes partition with a canonical circular topology. With 1024 sites per node and boundaries comprising only 20 sites the problem can easily transferred onto a parallel computer system without suffering too great a decrease in performance due to the communications overhead. All simulations on the Paragon were made using a 32/64-nodes partition with 4096 sites per node.

5 Simulations

The simulations were done for the two set of rules and three different ratios of slow vehicles $RatioS = 0.0$, $RatioS = 0.05$, and $RatioS = 0.15$.

5.1 Outflow

At the beginning of the simulation the whole system ($N = 2^{18}$) was filled with a given ratio of slow ($v_{max} = 3$) and fast ($v_{max} = 5$) vehicles. Usually vehicles arriving during the first $N/20$ timesteps were ignored. Then in each time step the open boundary region in the direction of motion was searched for vehicles. The velocites of these vehicles leaving the system ($v_{boundary}(j)$ for time step j) were accumulated. Afterwards all vehicles in the boundary region were deleted. For the plot the following formula was used

$$q(i) = \frac{1}{2iN} \sum_{j=1}^{i} v_{boundary}(j).$$

[4] if there is a vehicle right *beside* it the gap is regarded as zero

[5] if the corresponding predeccessor is not forced to decelerate due to a predecessor

[6] one MUPS is equivalent to the update of one million sites per second

5.2 Fundamental diagrams for periodic boundary conditions

The fundamental diagrams were computed for a set of densities in the region of interest with system sizes $N = 2^{17}$ over $T = 2^{17}$ time steps. At the beginning of each run approximately $N/20$ time steps were ignored before statistical accumulation was started and statistical data was gathered only every 10th time step.

6 Results

One set of results consists of two parts: one diagram depicting the accumulated averaged outflow versus time (steps) and one diagram depicting a small excerpt from the fundamental diagram for throughput in dependence of density. See []. The following table contains an overview of the numerical results:

System	CPN	S/A	RatioS	bound. cond.	Size	Steps	MUPS	time [h]	Throughput
NEC	1	S	0.00	outflow	2^{18}	-	178	0.31	0.341 ± 0.06
NEC	1	S	0.00	peridodic	2^{18}	2^{17}	204	0.93	0.341 ± 0.001
NEC	1	S	0.05	outflow	2^{18}	-	179	0.34	0.317 ± 0.07
NEC	1	S	0.05	peridodic	2^{18}	2^{17}	203	0.94	0.317 ± 0.002
NEC	1	S	0.15	outflow	2^{18}	-	179	0.34	0.313 ± 0.05
NEC	1	S	0.15	peridodic	2^{18}	2^{17}	204	0.93	0.313 ± 0.001
Parsytec	256	S	0.00	outflow	2^{18}	-	-	-	0.342
Paragon	32	S	0.00	periodic	2^{17}	2^{16}	22.1	2.61	0.343
Parsytec	256	S	0.05	outflow	2^{18}	-		-	0.317
Paragon	32	S	0.05	periodic	2^{17}	2^{16}	19.7	2.93	0.318
Parsytec	256	S	0.15	outflow	2^{18}	-		-	0.313
Paragon	32	S	0.15	periodic	2^{17}	2^{16}	19.5	2.96	0.314
Parsytec	256	A	0.0	outflow	2^{18}	-		-	0.257
Paragon	32	A	0.0	periodic	2^{17}	2^{15}	19.3	1.63	(0.255)
Parsytec	256	A	0.05	outflow	2^{18}	-		-	0.247
Paragon	32	A	0.05	periodic	2^{17}	2^{15}	21.0	1.50	(0.248)
Parsytec	256	A	0.15	outflow	2^{18}	-		-	0.238
Paragon	32	A	0.15	periodic	2^{17}	2^{15}	20.6	1.53	(!0.242!)

7 Conclusion

8 Acknowledgements

I thank A. Bachem for supply of computing facilities, K. Nagel, T. Pfenning, S. Rasmussen and M. Schreckenberg for help and discussions. Computing time on the Parsytec GCel-3 of the Zentrum f"ur Paralleles Rechnen K"oln, on the Paragon XP/S 5 and XP/S 10 of the Zentralinstitut f"ur Angewandte Mathematik (ZAM) of the Forschungszentrum J"ulich and on the NEC-SX 3 of the Regionales Rechenzentrum K"oln is gratefully acknowledged. I further thank all persons in charge of maintaining the above mentioned machines, especially M. Wottawa and S.Esser (K"oln) as well as W.Vermoehlen and R.Knecht (J"ulich).

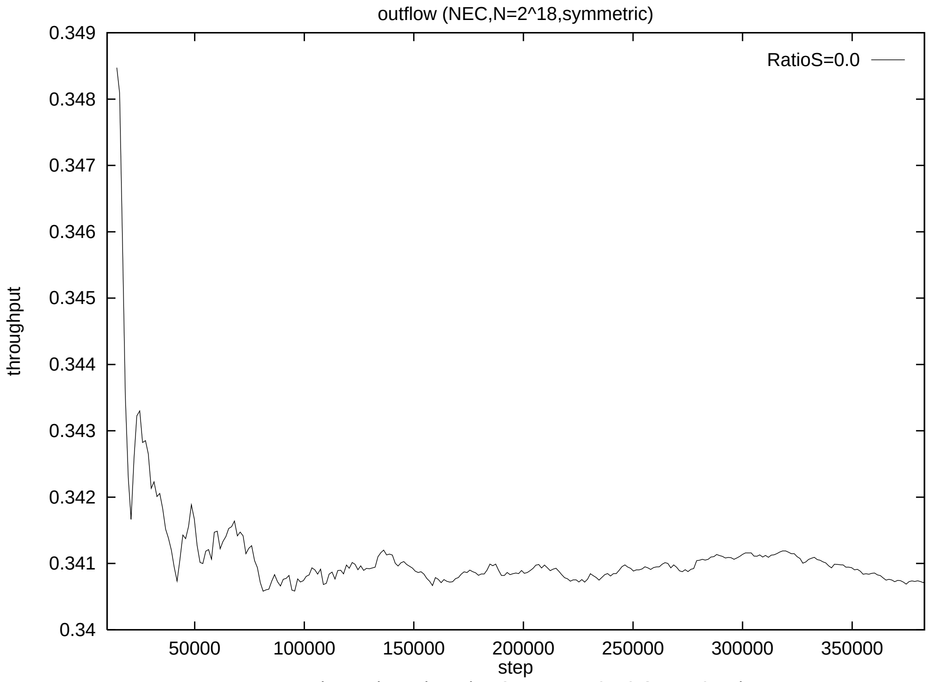
References

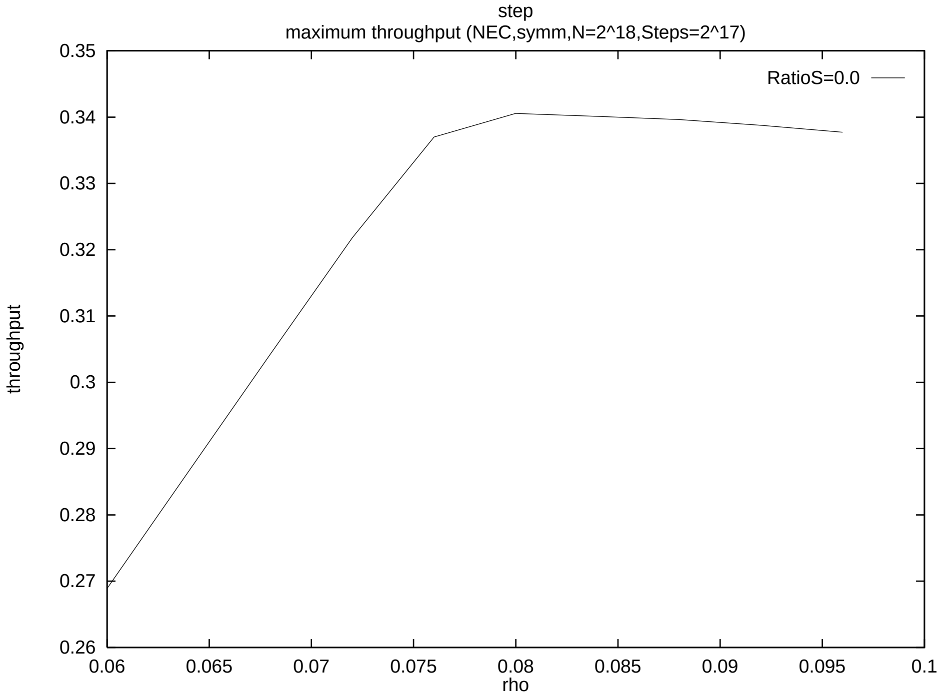
[1] K. Nagel. Life-times of simulated traffic jams. *Int. J. Mod. Physics*, C, in press.

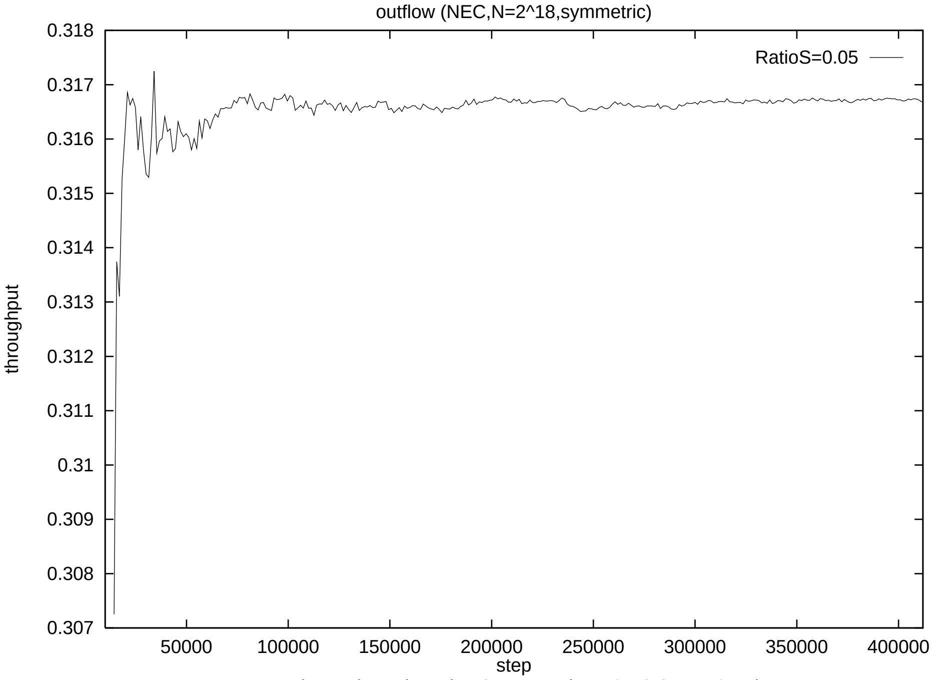
[2] K. Nagel and M. Schreckenberg. A cellular automaton model for freeway traffic. *J. Physique I*, 2:2221, 1992.

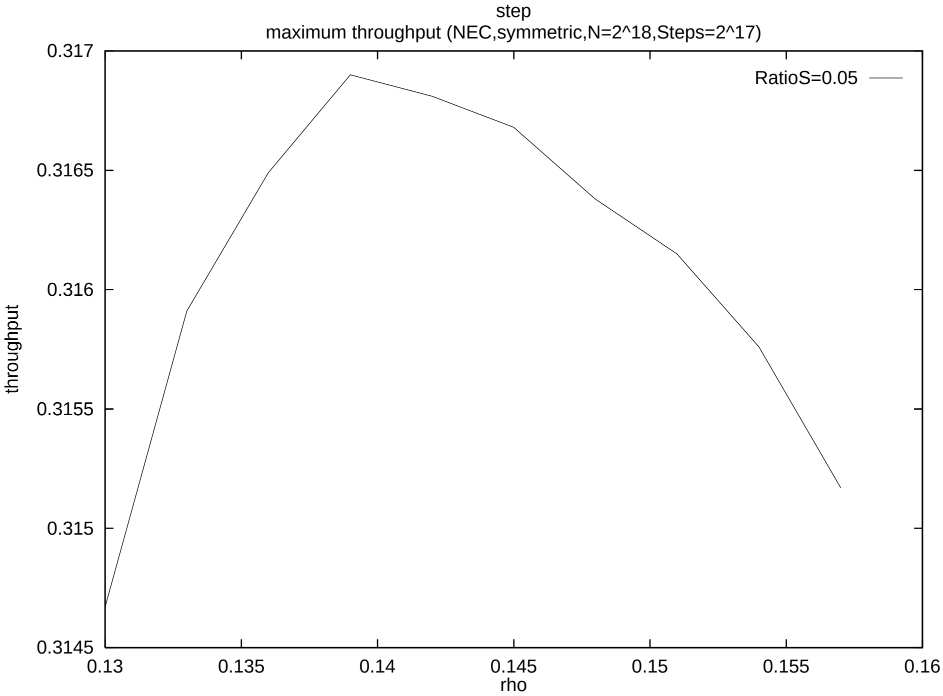
[3] A. Schadschneider and M. Schreckenberg. Cellular automaton models and traffic flow. *Journal of Physics A*, 26, 1993.

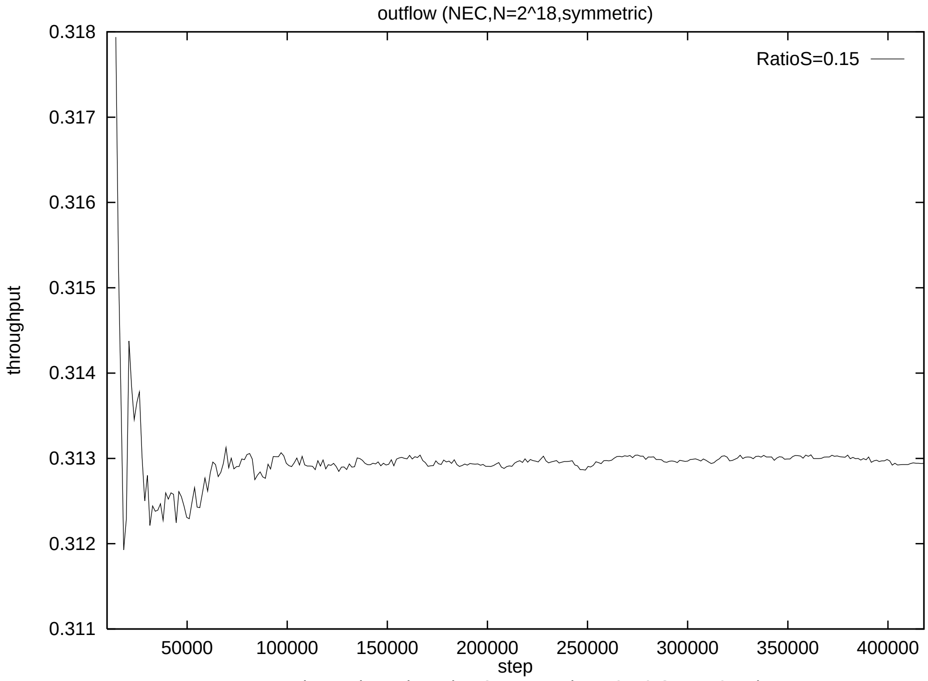
Figures

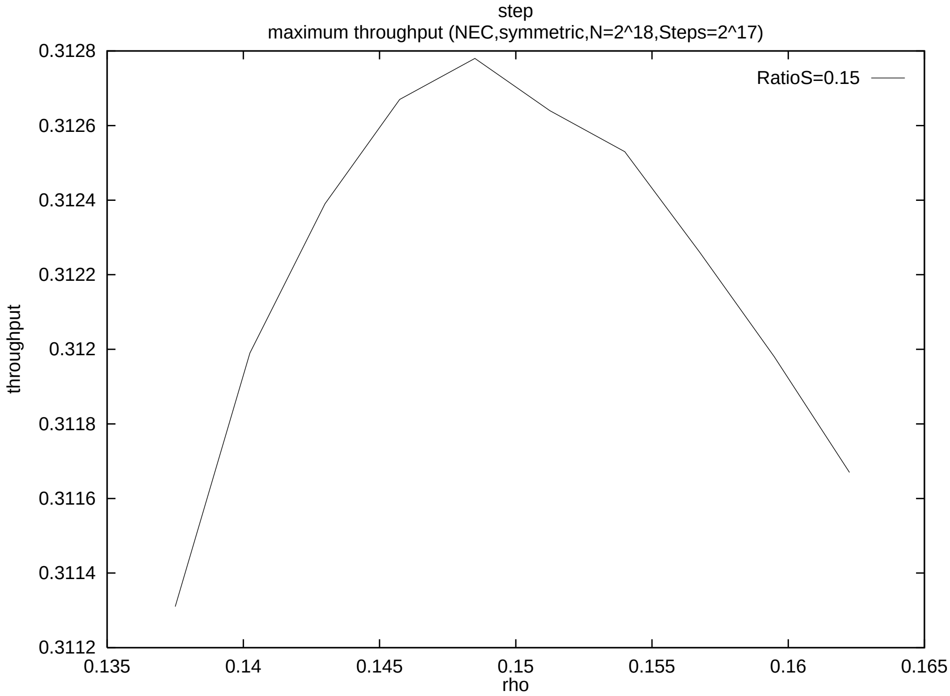


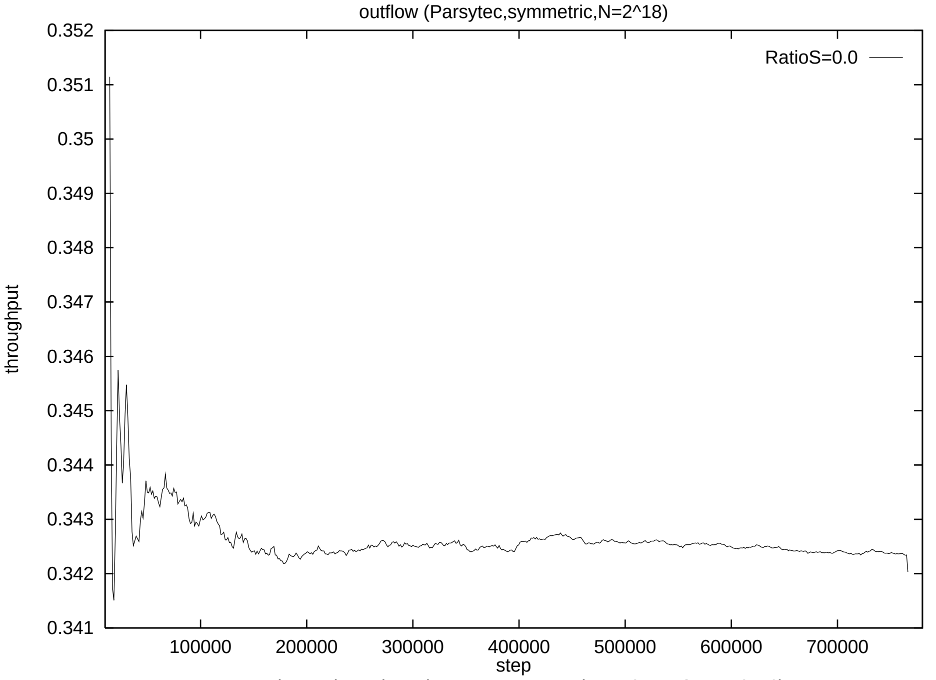


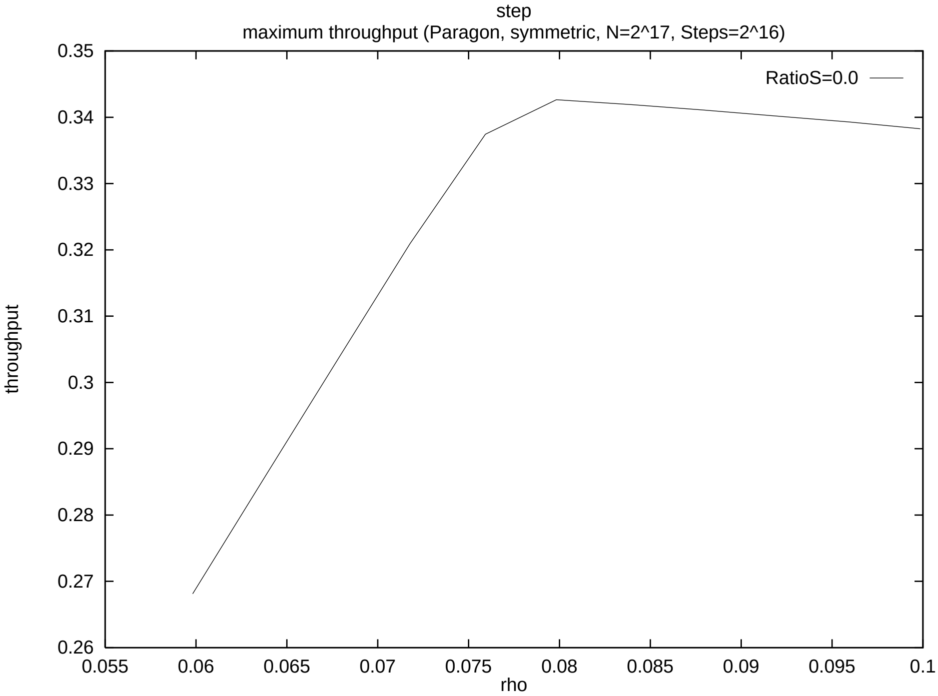

outflow (NEC,N=2^18,symmetric)
RatioS=0.05
throughput
step
0.318
0.317
0.316
0.315
0.314
0.313
0.312
0.311
0.31
0.309
0.308
0.307
50000
100000
150000
200000
250000
300000
350000
400000

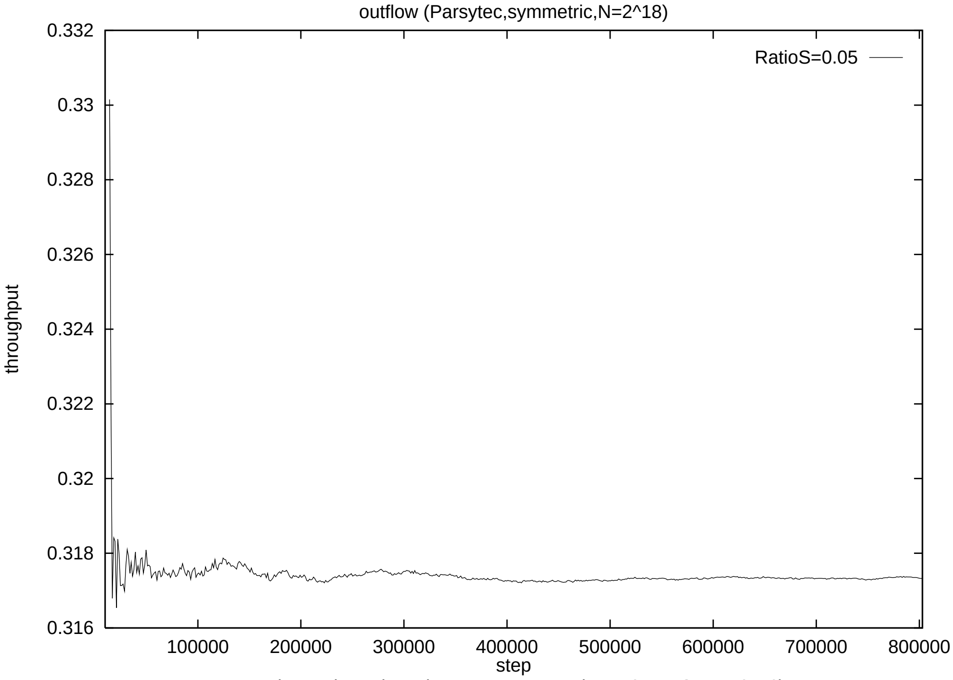

maximum throughput (NEC,symmetric,N=2^18,Steps=2^17)
RatioS=0.05
throughput
rho
0.317
0.3165
0.316
0.3155
0.315
0.3145
0.13
0.135
0.14
0.145
0.15
0.155
0.16

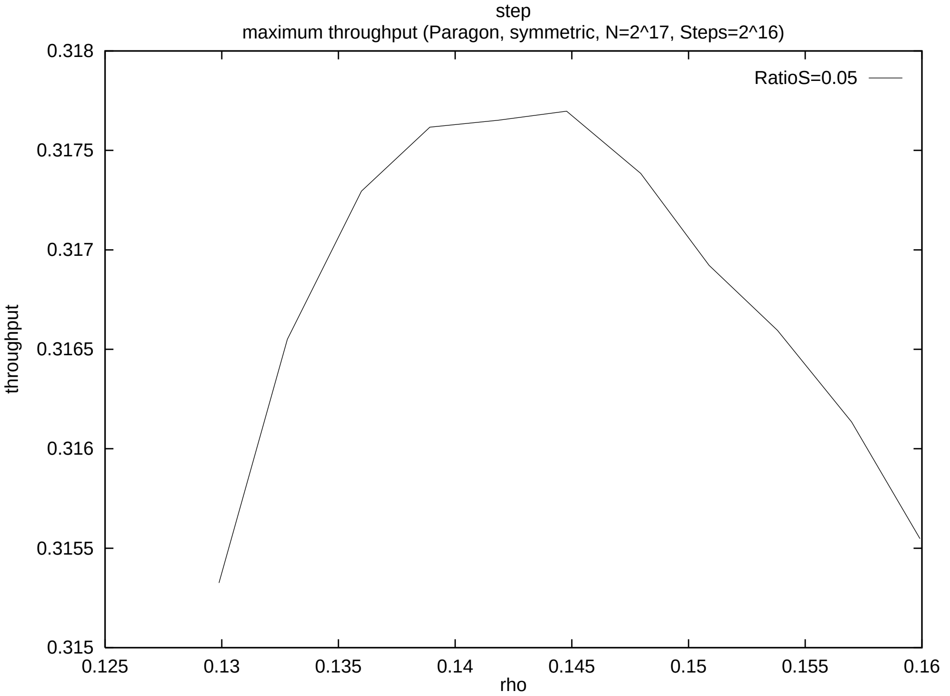

outflow (NEC,N=2^18,symmetric)
RatioS=0.15
throughput
step
0.318
0.317
0.316
0.315
0.314
0.313
0.312
0.311
50000
100000
150000
200000
250000
300000
350000
400000

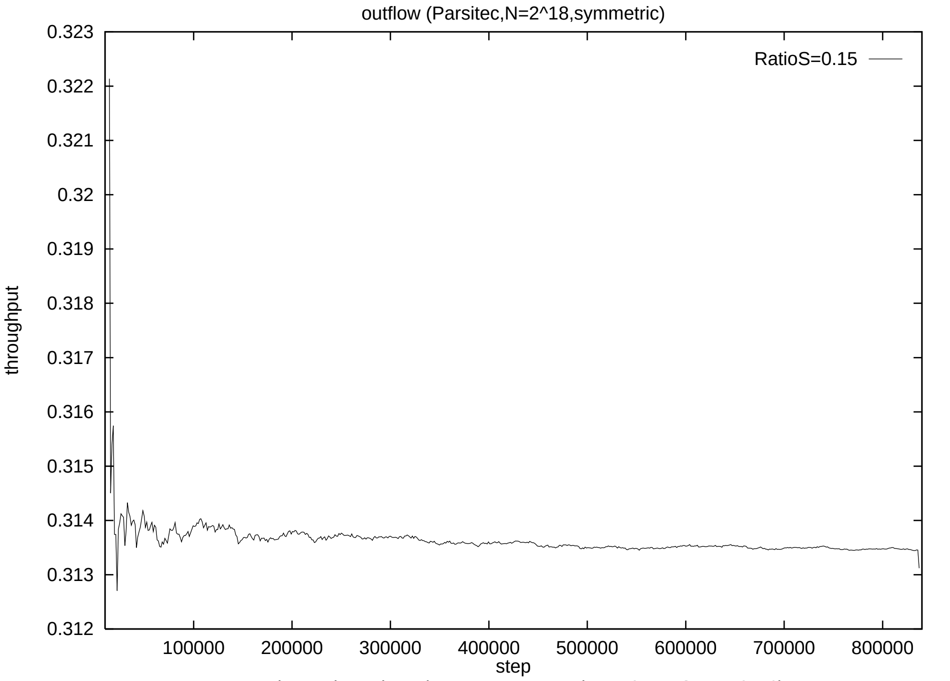

maximum throughput (NEC,symmetric,N=2^18,Steps=2^17)
RatioS=0.15
throughput
rho
0.3128
0.3126
0.3124
0.3122
0.312
0.3118
0.3116
0.3114
0.3112
0.135
0.14
0.145
0.15
0.155
0.16
0.165

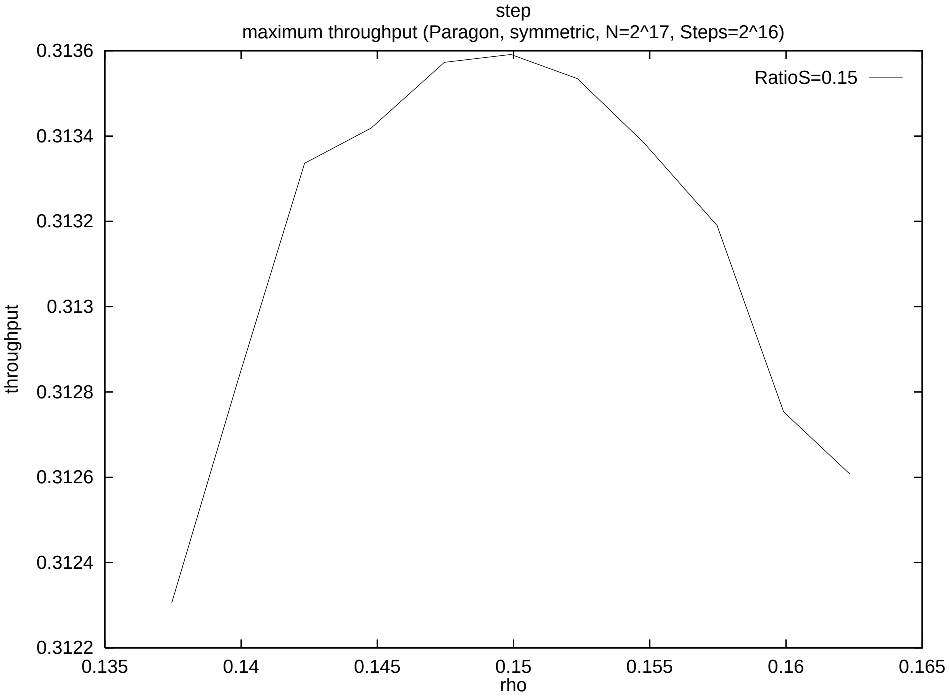

outflow (Parsytec,symmetric,N=2^18)
RatioS=0.0
throughput
step
0.352
0.351
0.35
0.349
0.348
0.347
0.346
0.345
0.344
0.343
0.342
0.341
100000
200000
300000
400000
500000
600000
700000

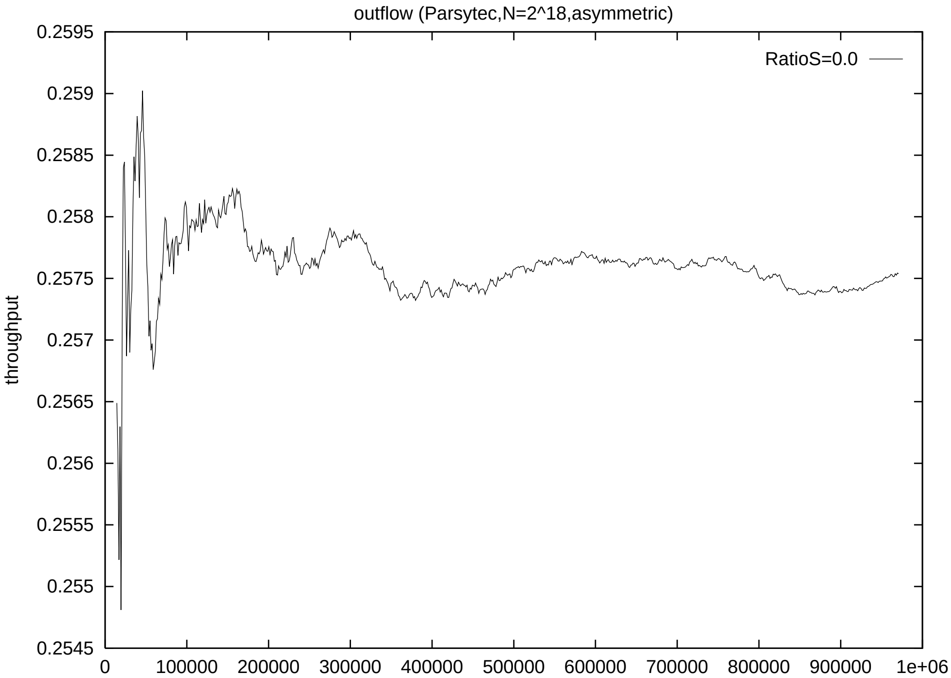

maximum throughput (Paragon, symmetric, N=2^17, Steps=2^16)
RatioS=0.0
throughput
rho
0.35
0.34
0.33
0.32
0.31
0.3
0.29
0.28
0.27
0.26
0.055
0.06
0.065
0.07
0.075
0.08
0.085
0.09
0.095
0.1

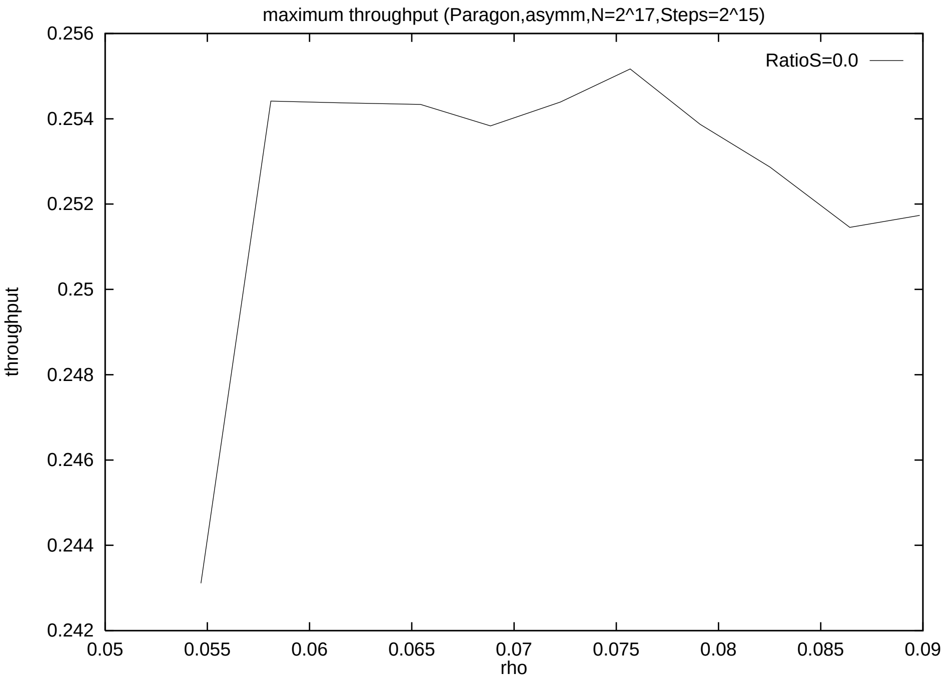

outflow (Parsytec,symmetric,N=2^18)
RatioS=0.05
throughput
step
0.332
0.33
0.328
0.326
0.324
0.322
0.32
0.318
0.316
100000
200000
300000
400000
500000
600000
700000
800000

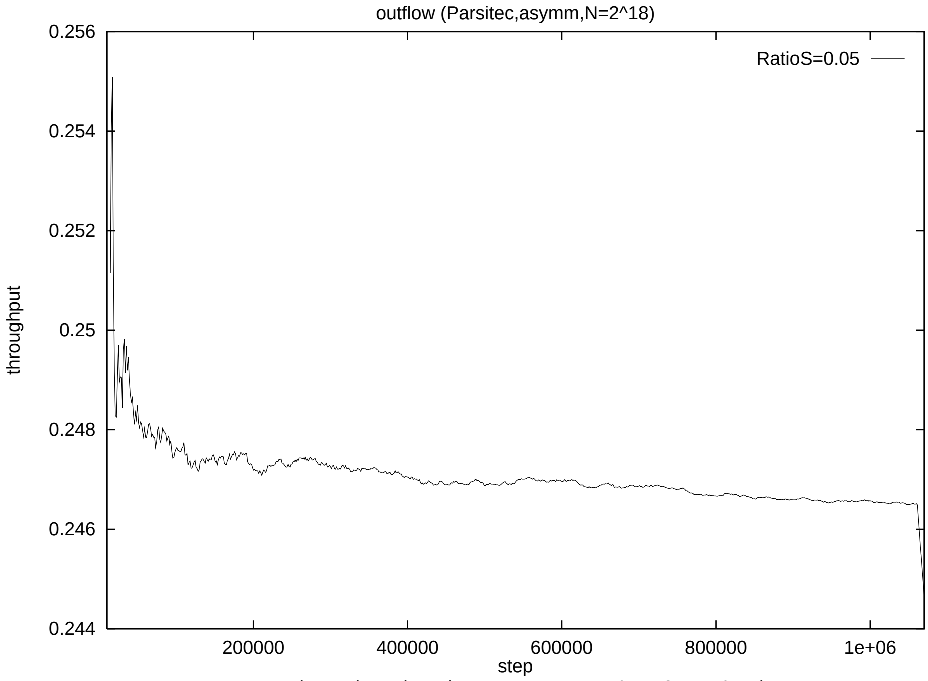

maximum throughput (Paragon, symmetric, N=2^17, Steps=2^16)
RatioS=0.05
throughput
rho
0.318
0.3175
0.317
0.3165
0.316
0.3155
0.315
0.125
0.13
0.135
0.14
0.145
0.15
0.155
0.16

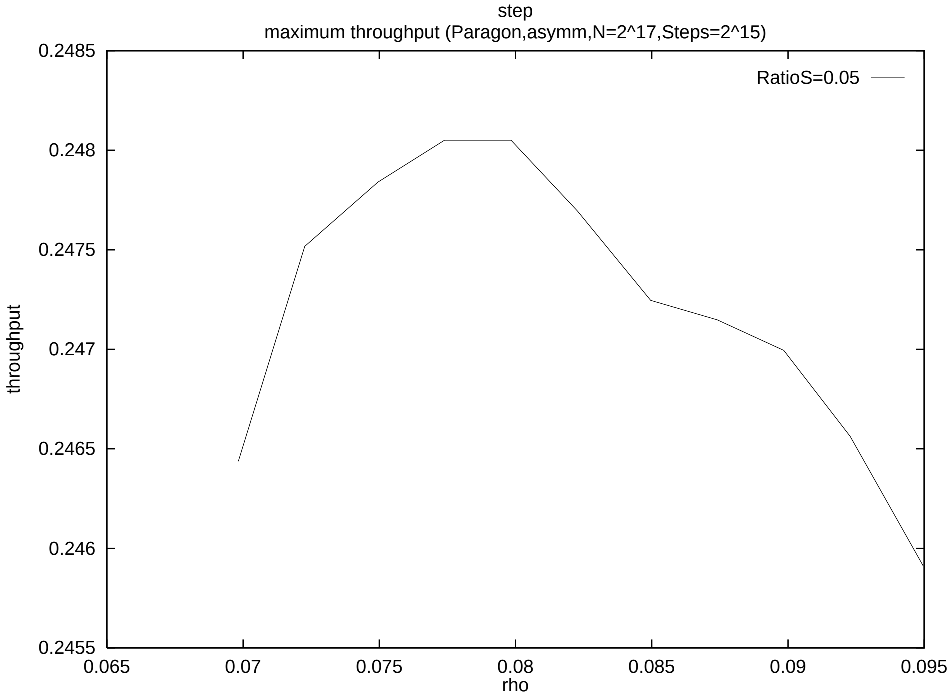

outflow (Parsitec,N=2^18,symmetric)
RatioS=0.15
throughput
step
0.323
0.322
0.321
0.32
0.319
0.318
0.317
0.316
0.315
0.314
0.313
0.312
100000
200000
300000
400000
500000
600000
700000
800000

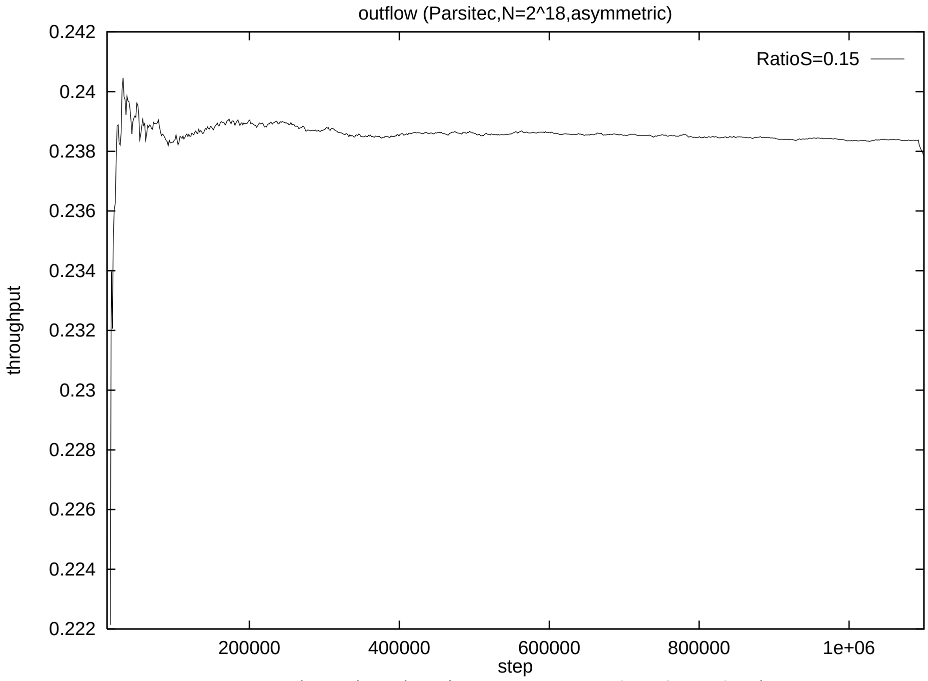

maximum throughput (Paragon, symmetric, N=2^17, Steps=2^16)
RatioS=0.15
throughput
rho
0.3136
0.3134
0.3132
0.313
0.3128
0.3126
0.3124
0.3122
0.135
0.14
0.145
0.15
0.155
0.16
0.165

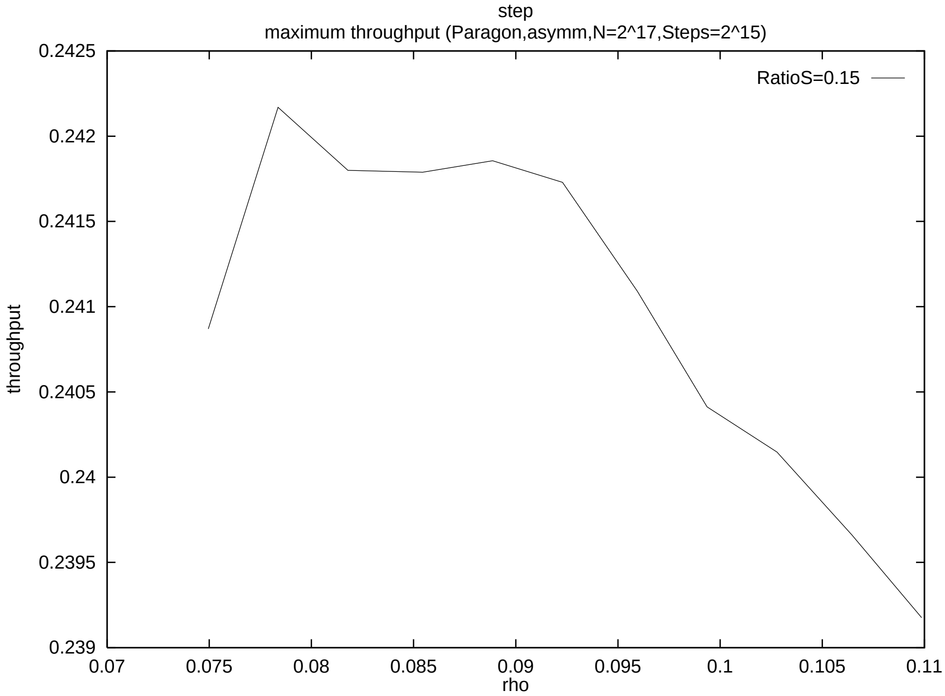

outflow (Parsytec,N=2^18,asymmetric)
RatioS=0.0
throughput
0.2595
0.259
0.2585
0.258
0.2575
0.257
0.2565
0.256
0.2555
0.255
0.2545
0
100000
200000
300000
400000
500000
600000
700000
800000
900000
1e+06


maximum throughput (Paragon,asymm,N=2^17,Steps=2^15)
RatioS=0.0
throughput
0.256
0.254
0.252
0.25
0.248
0.246
0.244
0.242
0.05
0.055
0.06
0.065
0.07
0.075
0.08
0.085
0.09
rho


outflow (Parsitec,asymm,N=2^18)
RatioS=0.05
throughput
step
0.256
0.254
0.252
0.25
0.248
0.246
0.244
200000
400000
600000
800000
1e+06


maximum throughput (Paragon,asymm,N=2^17,Steps=2^15)
RatioS=0.05
throughput
rho
0.2485
0.248
0.2475
0.247
0.2465
0.246
0.2455
0.065
0.07
0.075
0.08
0.085
0.09
0.095


outflow (Parsitec,N=2^18,asymmetric)
RatioS=0.15
throughput
step
0.242
0.24
0.238
0.236
0.234
0.232
0.23
0.228
0.226
0.224
0.222
200000
400000
600000
800000
1e+06


maximum throughput (Paragon,asymm,N=2^17,Steps=2^15)
RatioS=0.15
throughput
rho
0.2425
0.242
0.2415
0.241
0.2405
0.24
0.2395
0.239
0.07
0.075
0.08
0.085
0.09
0.095
0.1
0.105
0.11