Goldratt's "Dice Game" and the Five Focusing Steps

A simulation for learning purposes.

© Jürgen Kanz, March 2021

```
In[ ]:= ClearAll["Global`"]
```

1. Introduction

The Dice Game was introduced in one of the best-selling business books of all time, THE GOAL, authored by Dr. Eli Goldratt.

The WSU website created by James Holt gives very clear instructions on how to run your own dice game in a group, including all the details for each iteration and what the expected results should be (https://public.wsu.edu/~engrmgmt/holt/em530/Docs/DiceGames.htm).

Meantime several Excel sheets (like https://www.researchgate.net/publication/235301922_An_Excel-based_dice_game_An_integrative_learning_activity_in_operations_management) and special commercial apps (like https://www.goldrattsdicegame.com/) have been developed to simulate the Dice Game.

This notebook is slightly different because we simulate five different scenarios to highlight the interdependencies within the line. The scenario sequence is as follows:

- Balanced Line

- Constrained Line

- TOC - DBR with a focus on the Rope

- TLS

- Full TOC

For each of the scenarios, we analyze the statistics regarding the roll of dices, the queues between the work stations or work centers (called "stations" in the text), and the financial figures according to the Throughput Accounting.

1.1 Rules of the simulation

1. We have unlimited raw material available.

2. We can ship and sell what we produce.

3. Apart from the constrained station, we have two dices that represent the available capacity at each station. The constrained station has only one dice.

4. For comparison reasons, we save the rolled dice points from the *Balanced Line* scenario and re-use them in all other scenarios apart from the constraint dice result.

1.2 Rolling the dices

Please be aware that in case you have one dice and you roll it for example 100,000 times you get an almost Uniform distribution.

In[ ]:= `oneDice = Table[RandomInteger[{1, 6}], {i, 1, 100 000}];`
`Histogram[oneDice]`

Out[ ]=
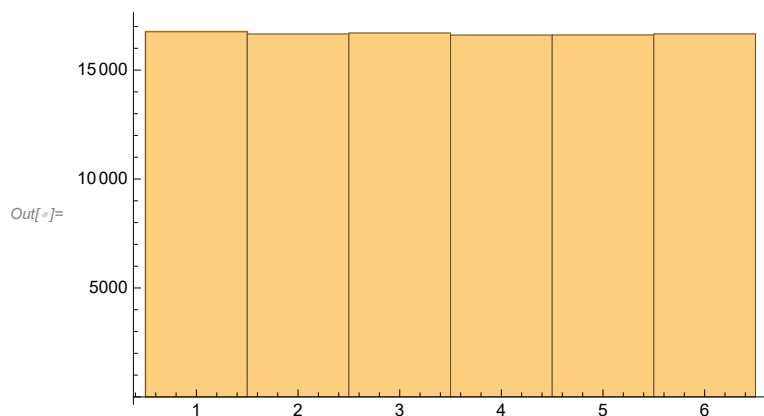
Average

In[ ]:= `averageOneDice = N[Mean[oneDice]]`

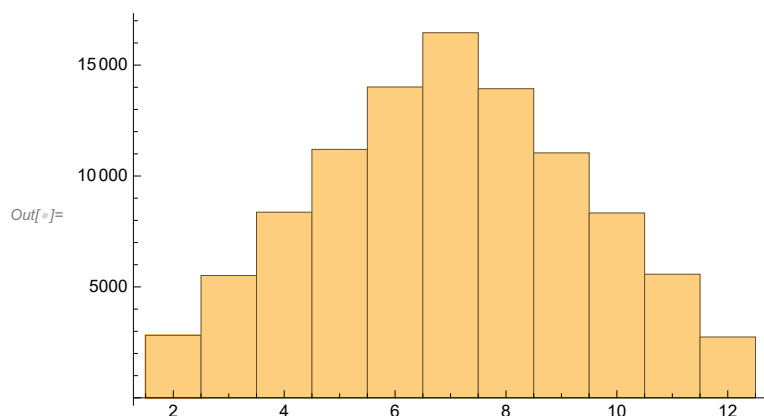*Out[ ]=* `3.49646`

Standard Deviation

In[ ]:= `stdOneDice = N[StandardDeviation[oneDice]]`

Out[ ]= `1.70896`

If you have two dices and you throw them 100,000 times you get a Normal (Gaussian) distribution.

In[ ]:= `twoDices = Table[Total[RandomInteger[{1, 6}, 2]], {i, 1, 100 000}];`

In[ ]:= `Histogram[twoDices]`

Out[ ]=

Average

In[●]:= `averageTwoDices = N[Mean[twoDices]]`

Out[●]= `6.9936`

Standard Deviation

In[●]:= `stdTwoDices = N[StandardDeviation[twoDices]]`

Out[●]= `2.41588`

1.3 Throughput Accounting

With the Input data and the simulation results, the Throughput Accounting figures can be determined:

- Throughput = (Selling Price - Total Variable Cost) * Total Quantity Shipped
- Inventory = (Total Ending WIP * Total Variable Cost) + Investment
- Net Profit = Throughput - Operating Expenses
- ROI = Net Profit / Inventory
- Customer Service = Total Quantity Shipped / Market Demand
- Productivity = Throughput / Operating Expenses
- Inventory Turnover = Total Quantity Shipped / Ending WIP

1.4 The Five Focusing Steps

The Five Focusing Steps belong to the core of TOC - Theory of Constraints. It is a focusing process that helps to manage the constraint(s) in a system.

1. Identify the Constraint
2. Exploit the Constraint
3. Subordinate Everything Else to the Constraint
4. Elevate the Constraint
5. Prevent Inertia from Becoming the Constraint

In this simulation we consider all steps during the Dice Game.

---

2. Input of data

INPUT: number of time units in the observed period. This can be seconds, minutes, hours, or days.

In[●]:= `timeUnits = 100;`

INPUT: amount of work-in-process in each queue at the beginning of the period

In[●]:= `startWipLevel = 15;`

INPUT : Total Variable Cost per produced unit [$]

In[●]:= `tvc = 200;`

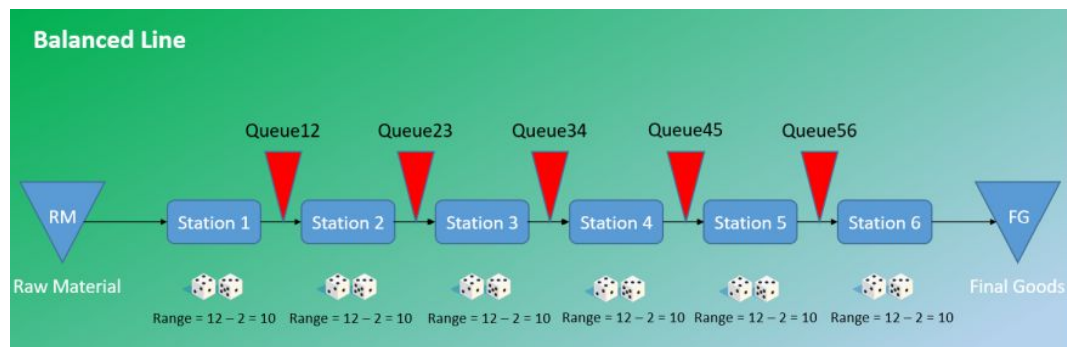INPUT : Selling Price per shipped unit [$]

In[ ]:= `sp = 400;`

INPUT: Operating Expenses (oe) for the period. OE is typically the summation of Period Expense + Carrying Cost + Improvement Initiative Cost [$]

In[ ]:= `oe = 30 000;`

INPUT: Needed Investment (invest) in the period in order to fulfill the order(s). [$]

In[ ]:= `invest = 20 000;`

---

3. Calculation for Balanced Line

The Balanced Line is a typical manufacturing line. Each station has two dices that represent the available capacity of the station. The statistical range per station is ten. Which means that each dice can roll one to six points. The points are added for each station in every round. The number of rounds depends on the variable *timeUnits*. What *station 1* has rolled will be forwarded to *station 2* in the form of material. For more details about the process, the reader should study the material as suggested in section 1. Introduction.

3.1 Calculation part

3.2 Rolling dice statistics

How many points have been rolled in each station? What are the statistical results when we take the results overall *timeUnits*?

	station 1	station 2	station 3	station 4	station 5	station 6
Min	2.0	2.0	2.0	2.0	2.0	2.0
Mean	6.91	6.95	6.73	6.58	6.91	6.96
Median	7.0	7.0	7.0	7.0	7.0	7.0
StandardDeviation	2.531	2.45104	2.43234	2.37487	2.78196	2.59728
Variance	6.40596	6.00758	5.91626	5.64	7.73929	6.74586
Max	12.0	11.0	12.0	12.0	12.0	12.0

The results are in line with the outcome of the two dices in section 1.2 Rolling the dices.

3.3 Queue statistics

The *Queue statistics* are different compared to the *Rolling dice statistics.* Here we see material among two stations as timelines overall *timeUnits*.
Isn't it obvious that our system belongs to the group of dynamic systems? It is recognizable that we have a strong interdependence caused by the variance of each roll of dices among the stations.

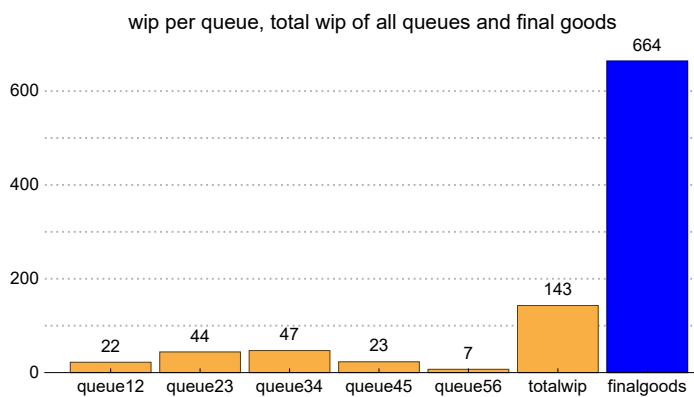*In[ ]:=* **BL1QueueGraph**

Out[ ]//TableForm=
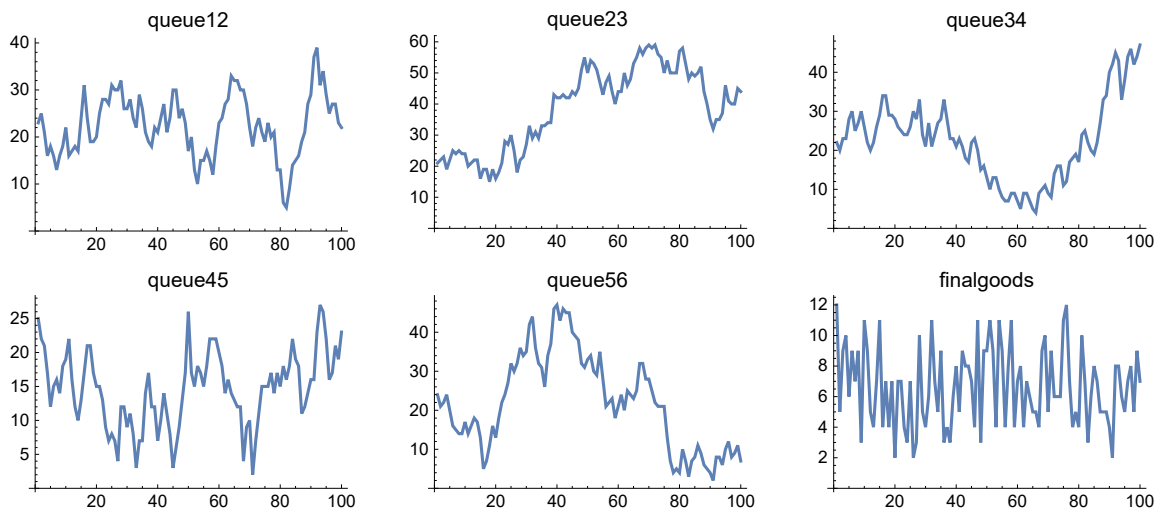
	queue 12	queue 23	queue 34	queue 45	queue 56	final goods
Min	5.0	15.0	4.0	2.0	2.0	2.0
Mean	22.44	38.67	22.4	14.64	21.9	6.64
Median	22.5	42.0	22.5	15.0	21.5	7.0
StandardDeviation	6.56178	13.091	10.4185	5.47634	12.0968	2.56046
Variance	43.057	171.375	108.545	29.9903	146.333	6.55596
Max	39.0	59.0	47.0	27.0	47.0	12.0

Reminder: The table here shows material units/pieces in each queue. Not dice points!

3.4 Comparison of a) available and utilized capacities and b) last run WIP plus Final goods

In[ ]:= `Row[{plot1, plot2}, "*        *"]`

Out[ ]=

available versus utilized capacity per workstation

[Bar chart comparing available capacity (purple) and utilized capacity (blue) per workstation:
- station 1: 691., 691.
- station 2: 695., 692.
- station 3: 673., 673.
- station 4: 658., 652.
- station 5: 691., 649.
- station 6: 696., 664.]

■ available capacity   ■ utilized capacity

*   *

wip per queue, total wip of all queues and final goods

[Bar chart:
- queue12: 22
- queue23: 44
- queue34: 47
- queue45: 23
- queue56: 7
- totalwip: 143
- finalgoods: 664]
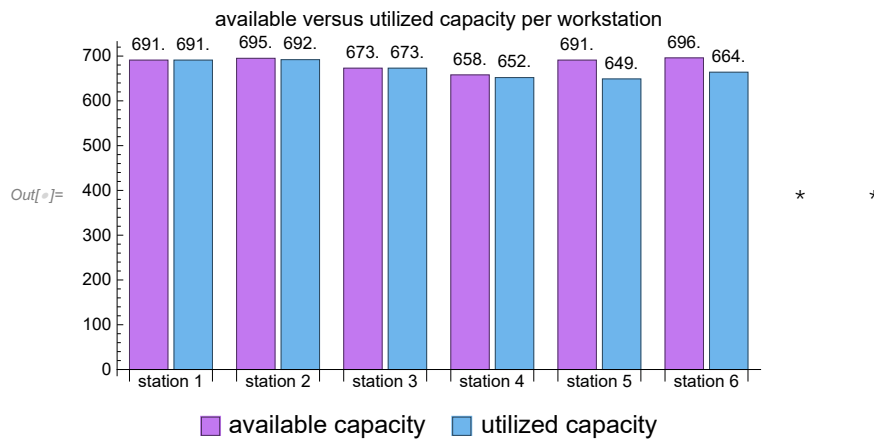
A *Balanced Line* leads to an almost 100% utilization of the available capacity. At the end of our game, we have only 143 pieces in working-in-process (Ending WIP), but we have sold and shipped 664 final goods. An excellent result!

Let us have a look at the financial figures.

3.5 Throughput Accounting figures

	Balanced Line
Throughput	132 800.
Inventory	48 600.0
Operating Expenses	30 000.0
Net Profit	102 800.
ROI	2.115%
Customer Service	100%
Productivity	4.42667
Inventory Turnover	4.64336

The financial figures of Throughput Accounting confirm the excellent result we have achieved so far. Please be aware that we can sell and ship all final goods independent from the scenario, hence the *Customer Service* will always be on a level of 100%.

Therefore it is not a wonder that companies strive for balanced lines. But is such a solution realistic?

According to Dr. Eliyahu Goldratt there is always at least one constraint that limits the flow inside a system. Typical reasons are capacities of machines and/or workers are insufficient, Murphy's Law comes into play *"Anything that can go wrong, will go wrong"*, or any other thinkable disturbance can limit the flow. In such a case we face a physical bottleneck or in other contexts called a constraint. Our line becomes unbalanced. In the next section, we will analyze a *Constrained Line*.

4. Constrained Line
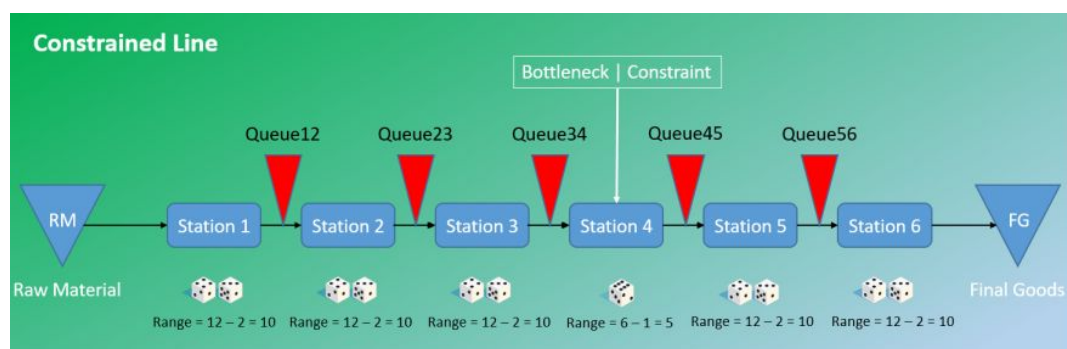
For this simulation scenario we reduce the available capacity of *station 4*. We only use one dice. "**Station 4**" is now our *constraint* also for the following scenarios. This also means that we have achieved the first step of the *Five Focusing Steps*.

To have a good comparison with the *Balanced Line*, we re-use the rolled dice points of the remaining stations. The material exchange from station to station has to be re-calculated.

4.1 Calculation part

4.2 Rolling dice statistics

	station 1	station 2	station 3	station 4	station 5	station 6
Min	2.0	2.0	2.0	1.0	2.0	2.0
Mean	6.91	6.95	6.73	3.41	6.91	6.96
Median	7.0	7.0	7.0	3.0	7.0	7.0
StandardDeviation	2.531	2.45104	2.43234	1.72969	2.78196	2.59728
Variance	6.40596	6.00758	5.91626	2.99182	7.73929	6.74586
Max	12.0	11.0	12.0	6.0	12.0	12.0

We can recognize the change of *station 4* here compared to all other stations. Please compare the results with section 1.2 Rolling the dices.

4.3 Queue statistics
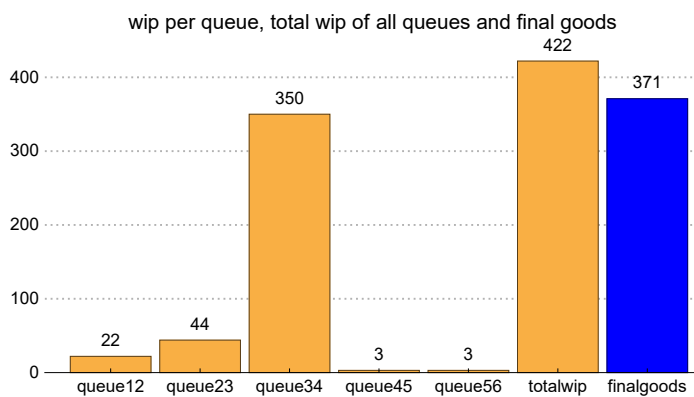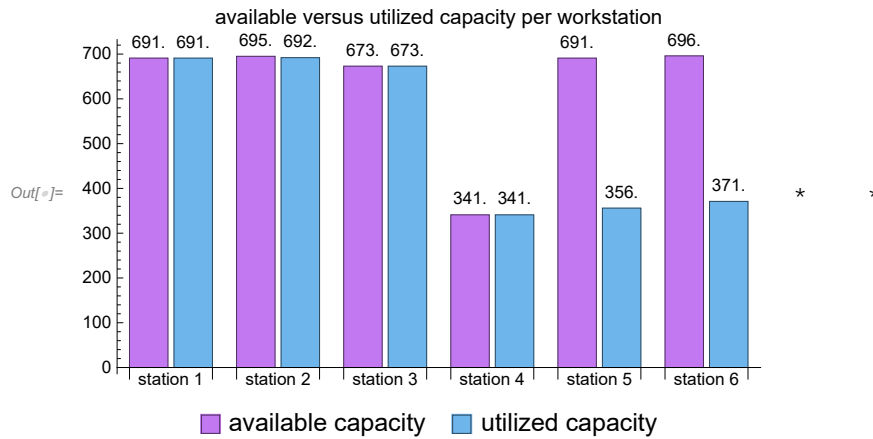
In[ ]:= **CL1QueueGraph**

Out[ ]//TableForm=
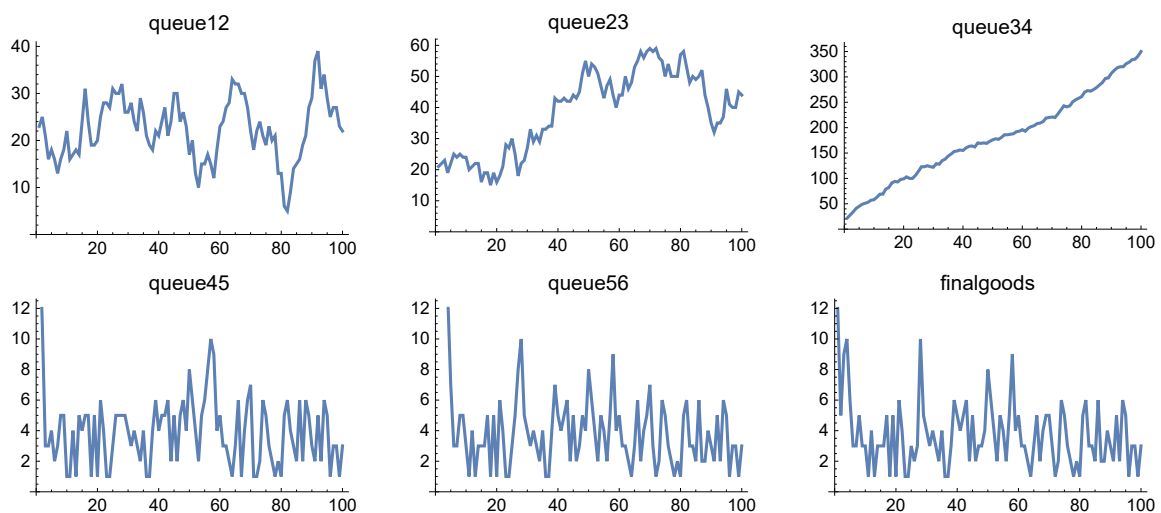
It is interesting to observe the tremendous almost linear growth of buffered material in queue 34. This queue is in front of the constraint.

We have to remind ourselves that the constrained station is now reacting with a Uniform probability distribution compared to all other stations with a Gaussian distribution.

	queue 12	queue 23	queue 34	queue 45	queue 56	final goods
Min	5.0	15.0	22.0	1.0	1.0	1.0
Mean	22.44	38.67	180.45	3.97	4.3	3.71
Median	22.5	42.0	176.5	4.0	3.5	3.0
StandardDeviation	6.56178	13.091	85.6205	2.48817	3.71048	2.16163
Variance	43.057	171.375	7330.88	6.19101	13.7677	4.67263
Max	39.0	59.0	350.0	17.0	24.0	12.0

4.4 Comparison of a) available and utilized capacities and b) last run WIP plus Final goods

In[ ]:= `Row[{plot1, plot2}, "*          *"]`

Out[ ]=

available versus utilized capacity per workstation

[Bar chart comparing available capacity (purple) and utilized capacity (blue) per station:
station 1: 691., 691.
station 2: 695., 692.
station 3: 673., 673.
station 4: 341., 341.
station 5: 691., 356.
station 6: 696., 371.]

■ available capacity   ■ utilized capacity

wip per queue, total wip of all queues and final goods

[Bar chart (orange bars, final goods in blue):
queue12: 22
queue23: 44
queue34: 350
queue45: 3
queue56: 3
totalwip: 422
finalgoods: 371]

Let us try a simple explanation. *Stations 1,2,3* know nothing about the constrained *station 4*, hence they produce as much as possible. The same is valid for *station 4* but on a lower level.
Stations 5 and 6 only get the lower output from *station 4.* Both stations are on a utilized capacity level of approx. 50%.
In this scenario we have much more work-in-process than as we have sold and shipped final goods.
If we take no action, queue 34 will "linearly" grow forever.

4.5 Comparison of Throughput Accounting figures

	Balanced Line	Constrained Line
Throughput	132 800.	74 200.0
Inventory	48 600.0	104 400.
Operating Expenses	30 000.0	30 000.0
Net Profit	102 800.	44 200.0
ROI	2.115%	0.9847%
Customer Service	100%	100%
Productivity	4.42667	2.47333
Inventory Turnover	4.64336	0.879147

5. TOC-DBR

TOC-DBR is an abbreviation that stands for TOC-Theory of Constraints with the application solution DBR Drum-Buffer-Rope.

Drum is the output of our constraint or to be more precisely: in operations, the schedule of the constraint.

Buffer is the protection against uncertainty for the constraint. In our case the material available in queue 34. In practice, it should be realized with active buffer management which is not integrated into this simulation.

Rope is a feedback loop or you can call it an information line.

In this section, we will make use of the **Rope** to study its impact on the flow. It is very simple because we inform *station 1* about the utilized capacity of *station 4* of the former round.
In the next round, *station 1* will only release the amount of material that was moved from *station 4* to *station 5* in the previous round.
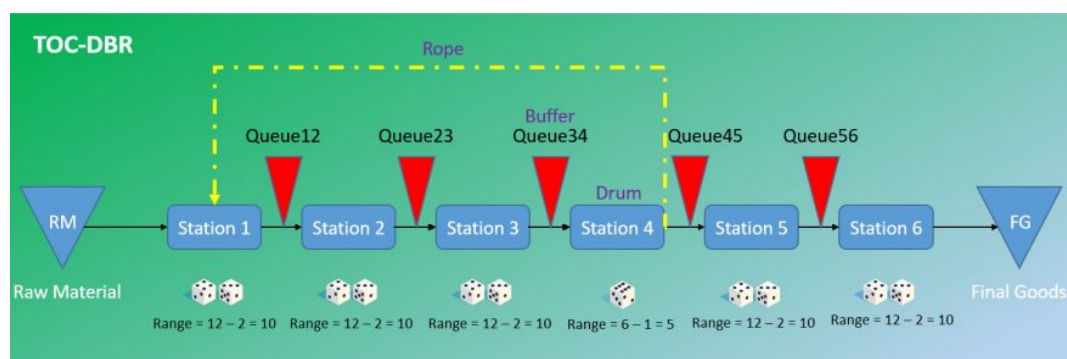
5.1 Calculation part

5.2 Rolling dice statistics

In this scenario, we re-use all rolled dice points, but we modify the program code to make use of the information via the rope.

	station 1	station 2	station 3	station 4	station 5	station 6
Min	2.0	2.0	2.0	1.0	2.0	2.0
Mean	6.91	6.95	6.73	3.41	6.91	6.96
Median	7.0	7.0	7.0	3.0	7.0	7.0
StandardDeviation	2.531	2.45104	2.43234	1.72969	2.78196	2.59728
Variance	6.40596	6.00758	5.91626	2.99182	7.73929	6.74586
Max	12.0	11.0	12.0	6.0	12.0	12.0

5.3 Queue statistics

In[ ]:= **DBRQueueGraph**

Out[ ]//TableForm=
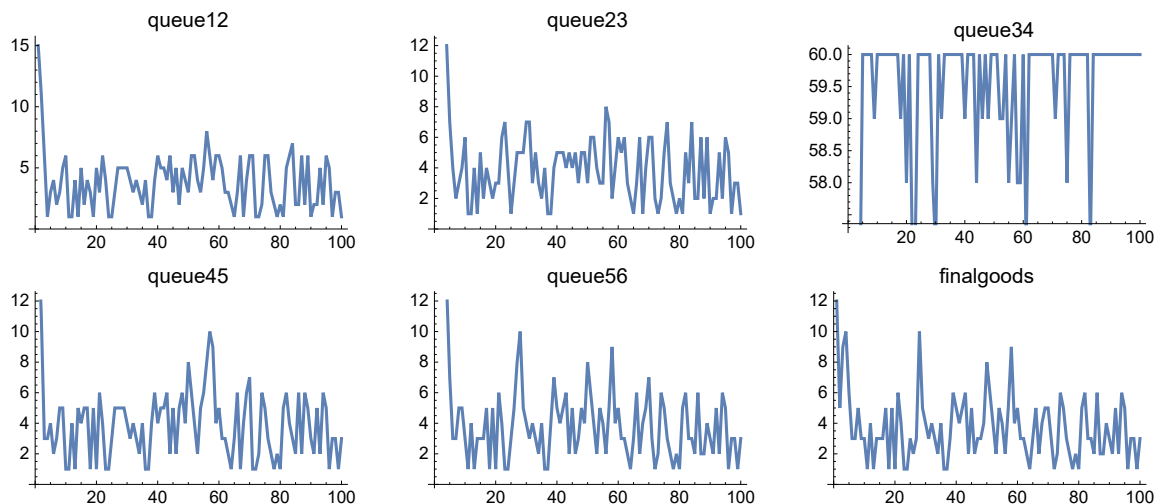
	queue 12	queue 23	queue 34	queue 45	queue 56	final goods
Min	1.0	1.0	37.0	1.0	1.0	1.0
Mean	3.81	4.34	59.06	3.97	4.3	3.71
Median	4.0	4.0	60.0	4.0	3.5	3.0
StandardDeviation	2.28608	3.5851	3.09715	2.48817	3.71048	2.16163
Variance	5.22616	12.8529	9.59232	6.19101	13.7677	4.67263
Max	15.0	22.0	60.0	17.0	24.0	12.0

5.4 Comparison of a) available and utilized capacities and b) last run WIP plus Final goods

In[ ]:= `Row[{plot1, plot2}, "*          *"]`

Out[ ]=

available versus utilized capacity per workstation

	station 1	station 2	station 3	station 4	station 5	station 6
available capacity	691.	695.	673.	341.	691.	696.
utilized capacity	346.	353.	368.	341.	356.	371.

■ available capacity   ■ utilized capacity

*   *

wip per queue, total wip of all queues and final goods

queue12	queue23	queue34	queue45	queue56	totalwip	finalgoods
1	1	60	3	3	68	371
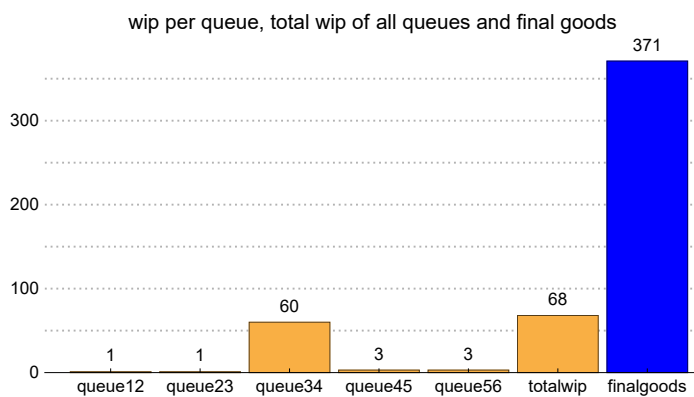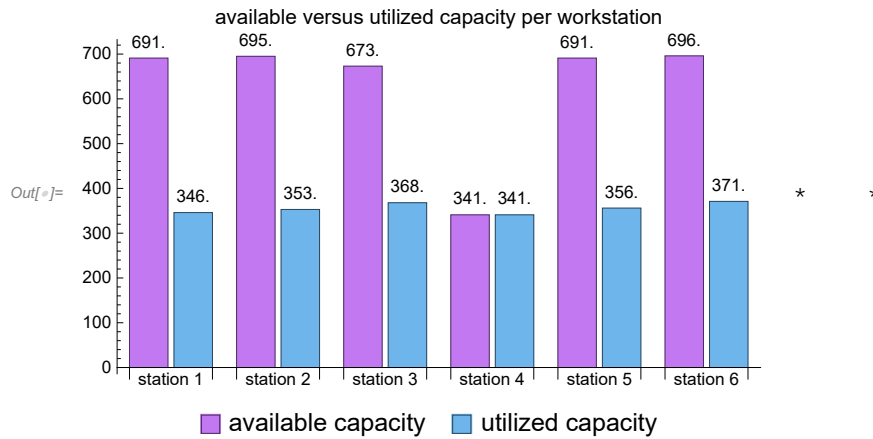
Apart from *station 4*, the utilized capacities of all other stations are significantly reduced. More important is that the work-in-process could be reduced from 422 pieces to 68 units.

5.5 Comparison of Throughput Accounting figures

	Balanced Line	Constrained Line	TOC–DBR
Throughput	132 800.	74 200.0	74 200.0
Inventory	48 600.0	104 400.	33 600.0
Operating Expenses	30 000.0	30 000.0	30 000.0
Net Profit	102 800.	44 200.0	44 200.0
ROI	2.115%	0.9847%	1.315%
Customer Service	100%	100%	100%
Productivity	4.42667	2.47333	2.47333
Inventory Turnover	4.64336	0.879147	5.45588

Okay, in this step we did not increase the Throughput, but we could significantly reduce WIP and

therefore we get a much lower Inventory.

The **Rope** launch does typically belong to step 3 of the *Five Focusing Steps*: *3. Subordinate Everything Else to the Constraint*

For didactical reasons, I decided to launch the Rope first and we will make use of it in the following scenarios.

---

6. TLS

TLS is an abbreviation for **T**heory of Constraints, **L**ean, and **S**ix Sigma. These three improvement methodologies are known in operations management. Each methodology has its own advantages. By using the term TLS it is meant to take the best from each method and combine them.

In this section, we apply **TLS** at our constraint to full-fill step number 2 of the *Five Focusing Steps*: *2. Exploit the Constraint*
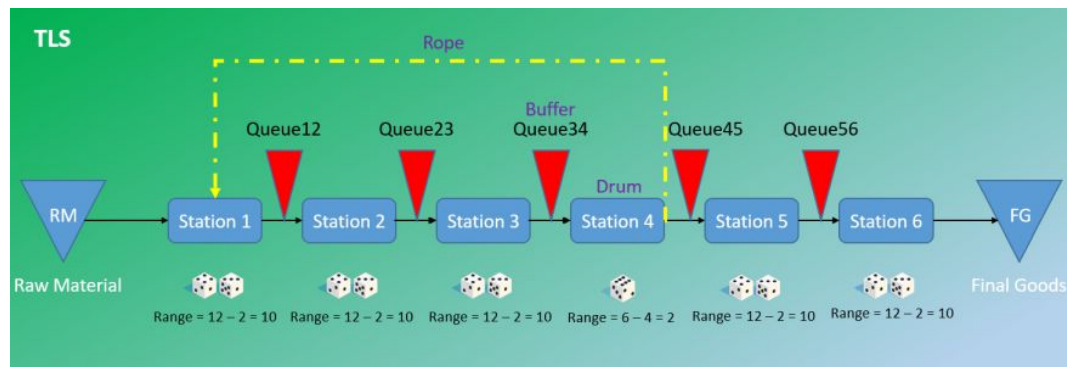
Let us assume that we could stabilize the process of *station 4* with the help of Lean and Six Sigma. The result is a lower variance. We can (only) roll now a 4, 5, or 6. This action should allow us to increase the Throughput.

6.1 Calculation part

6.2 Rolling dice statistics

	station 1	station 2	station 3	station 4	station 5	station 6
Min	2.0	2.0	2.0	4.0	2.0	2.0
Mean	6.91	6.95	6.73	4.93	6.91	6.96
Median	7.0	7.0	7.0	5.0	7.0	7.0
StandardDeviation	2.531	2.45104	2.43234	0.807227	2.78196	2.59728
Variance	6.40596	6.00758	5.91626	0.651616	7.73929	6.74586
Max	12.0	11.0	12.0	6.0	12.0	12.0

The Rolling Dice Statistics shows a higher mean, median, and much lower variance for *station 4*.
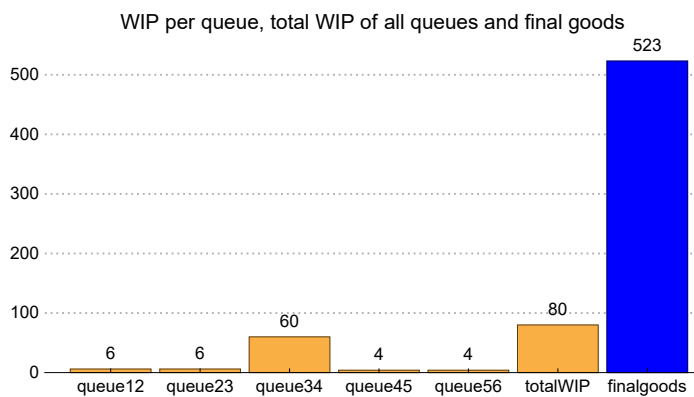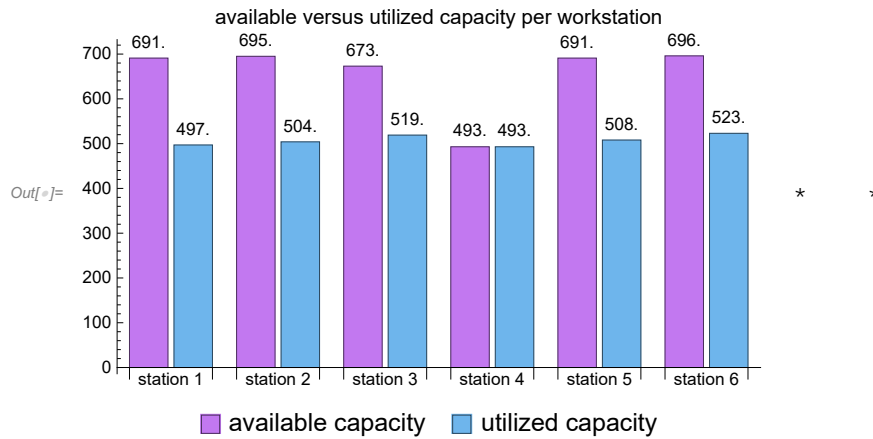
6.3 Queue statistics

In[ ]:= **TLSQueueGraph**

Out[ ]//TableForm=
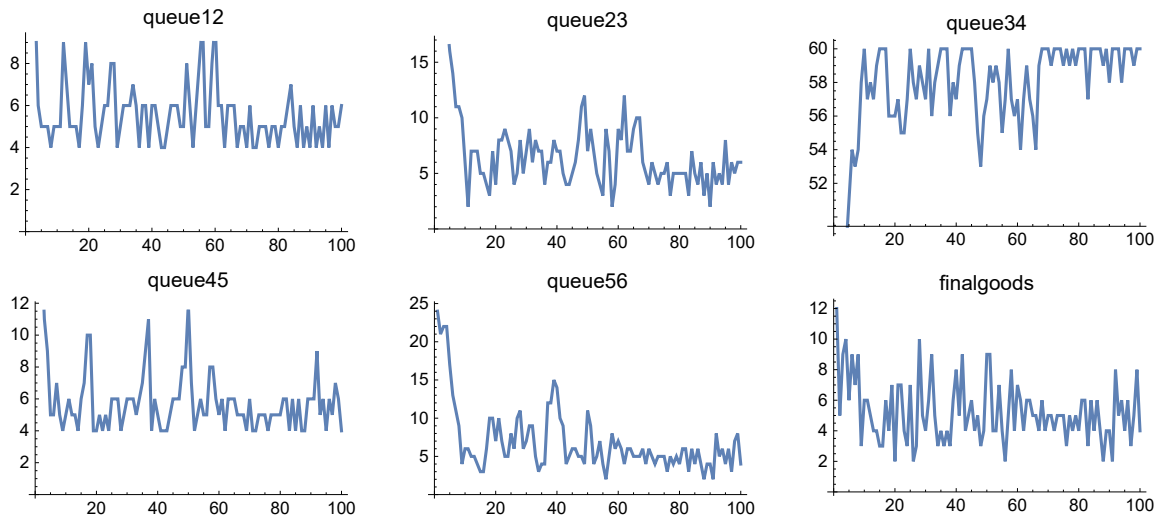
	queue 12	queue 23	queue 34	queue 45	queue 56	final goods
Min	4.0	2.0	37.0	4.0	2.0	2.0
Mean	5.76	6.95	57.52	5.96	6.88	5.23
Median	5.0	6.0	59.0	5.0	6.0	5.0
StandardDeviation	1.91285	3.87005	3.8257	2.42429	4.25282	2.02437
Variance	3.65899	14.9773	14.636	5.87717	18.0865	4.09808
Max	15.0	23.0	60.0	20.0	24.0	12.0

6.4 Comparison of a) available and utilized capacities and b) last run WIP plus Final goods

In[ ]:= `Row[{plot1, plot2}, "*        *"]`

Out[ ]=

available versus utilized capacity per workstation

* *

WIP per queue, total WIP of all queues and final goods

6.5 Comparison of Throughput Accounting figures

|	Balanced Line	Constrained Line	TOC–DBR	TLS
Throughput	132 800.	74 200.0	74 200.0	104 600.
Inventory	48 600.0	104 400.	33 600.0	36 000.0
Operating Expenses	30 000.0	30 000.0	30 000.0	30 000.0
Net Profit	102 800.	44 200.0	44 200.0	74 600.0
ROI	2.115%	0.9847%	1.315%	2.072%
Customer Service	100%	100%	100%	100%
Productivity	4.42667	2.47333	2.47333	3.48667
Inventory Turnover	4.64336	0.879147	5.45588	6.5375

7. Full TOC

The term Full TOC is invented by the author to describe the next step of the *Five Focusing Steps*: **4. Elevate the Constraint**. After all these improvement steps the moment has come to invest some

money to enlarge the capacity of *station 4*, but not earlier.

With a further focus on Lean and Six Sigma improvement opportunities and the investment in one new dice, we can further improve the Throughput of our system.

Our invest goes into four-sided dice that allows us to roll either a 5, 6, 7, or 8.
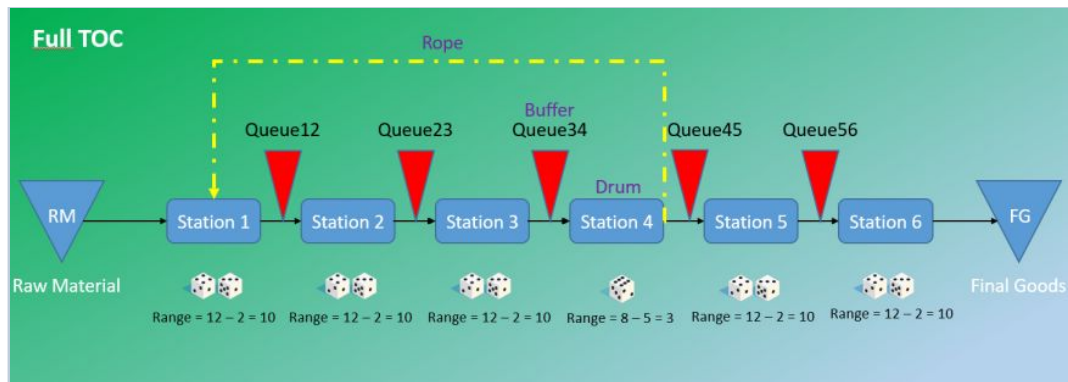
## 7.1 Calculation part

7.2 Rolling dice statistics

	station 1	station 2	station 3	station 4	station 5	station 6
Min	2.0	2.0	2.0	5.0	2.0	2.0
Mean	6.91	6.95	6.73	6.57	6.91	6.96
Median	7.0	7.0	7.0	7.0	7.0	7.0
StandardDeviation	2.531	2.45104	2.43234	1.09411	2.78196	2.59728
Variance	6.40596	6.00758	5.91626	1.19707	7.73929	6.74586
Max	12.0	11.0	12.0	8.0	12.0	12.0

The Rolling Dice Statistics shows that with the new dice the mean, and median values of *station 4* are almost the same compared to all other stations. Of course, the variance is still low.
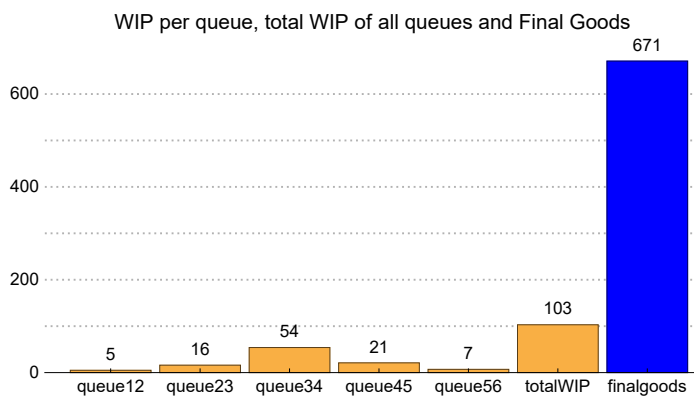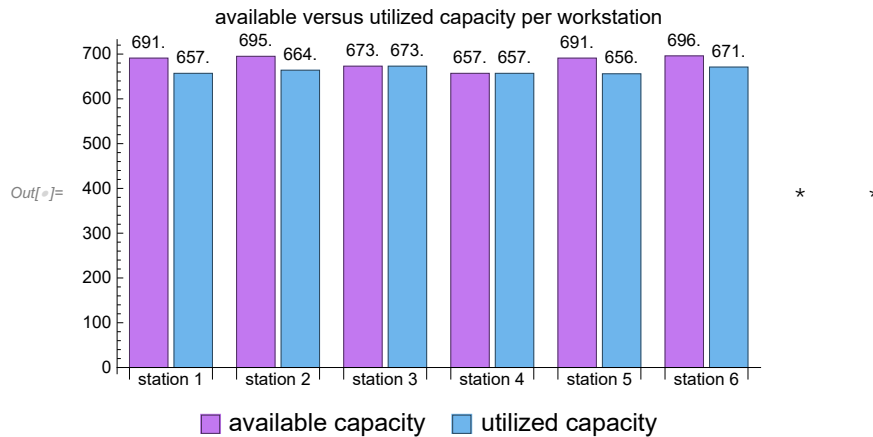
7.3 Queue statistics

In[ ]:= **FullTOCQueueGraph**

Out[ ]//TableForm=
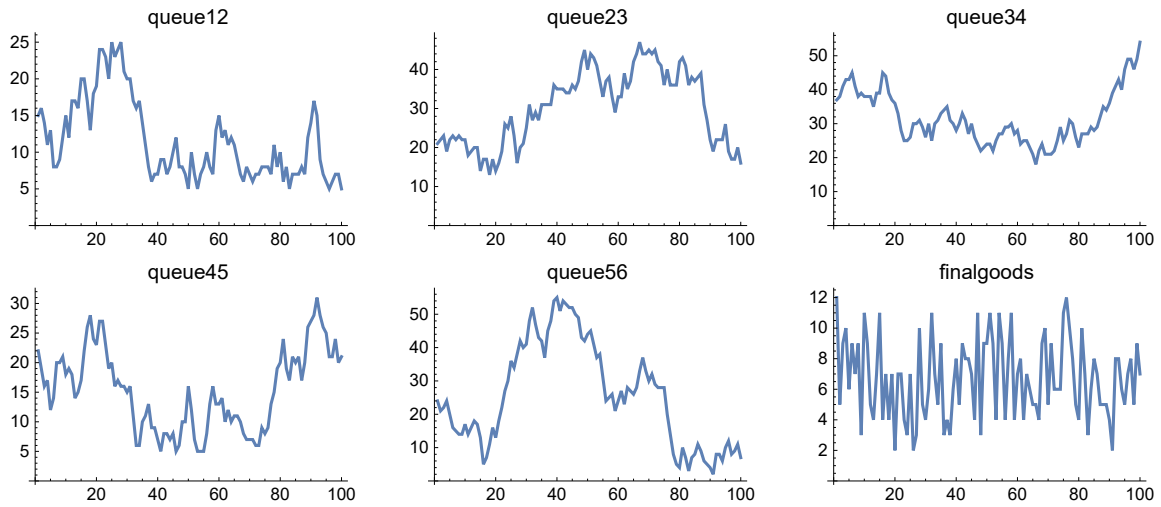
	queue 12	queue 23	queue 34	queue 45	queue 56	final goods
Min	5.0	13.0	18.0	5.0	2.0	2.0
Mean	11.56	30.17	31.64	15.28	25.57	6.71
Median	9.5	31.0	30.0	15.5	24.0	7.0
StandardDeviation	5.47782	9.48157	7.76228	6.83585	15.3703	2.57138
Variance	30.0065	89.9001	60.2529	46.7289	236.248	6.61202
Max	25.0	47.0	54.0	31.0	55.0	12.0

7.4 Comparison of a) available and utilized capacities and b) last run WIP plus Final goods

In[ ]:= `Row[{plot1, plot2}, "*        *"]`

Out[ ]=

available versus utilized capacity per workstation

WIP per queue, total WIP of all queues and Final Goods

The available and utilized capacities are now on almost the same level. The level of Final Goods is high and the Ending WIP low.

7.5 Comparison of Throughput Accounting figures

	Balanced Line	Constrained Line	TOC–DBR	TLS	Full TOC
Throughput	132 800.	74 200.0	74 200.0	104 600.	134 200.
Inventory	48 600.0	104 400.	33 600.0	36 000.0	50 600.0
Operating Expenses	30 000.0	30 000.0	30 000.0	30 000.0	30 000.0
Net Profit	102 800.	44 200.0	44 200.0	74 600.0	104 200.
ROI	2.115%	0.9847%	1.315%	2.072%	2.059%
Customer Service	100%	100%	100%	100%	100%
Productivity	4.42667	2.47333	2.47333	3.48667	4.47333
Inventory Turnover	4.64336	0.879147	5.45588	6.5375	6.51456

In case you study the code, you will recognize that I have added a $10,000 investment for the new dice into the Inventory category.
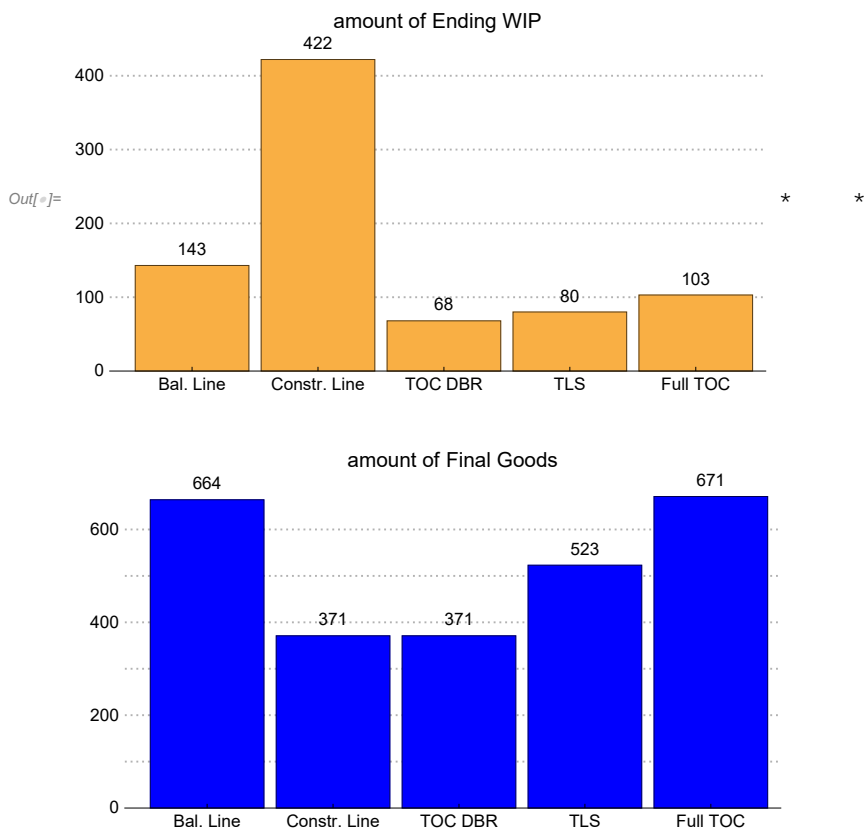
Nevertheless, our Throughput, Net Profit, Productivity, and Inventory Turnover is outstanding. Even compared to the Balanced Line that remains unconstrained. Great isn't it?

8. Comparison of Final Results

8.1 Calculation part

8.2 Comparison of Ending WIP and Final goods

In[ ]:= `Row[{plot1, plot2}, "*          *"]`

Out[ ]=

amount of Ending WIP

amount of Final Goods

8.3 Comparison of Throughput Accounting figures

	Balanced Line	Constrained Line	TOC–DBR	TLS	Full TOC
Throughput	132 800.	74 200.0	74 200.0	104 600.	134 200.
Inventory	48 600.0	104 400.	33 600.0	36 000.0	50 600.0
Operating Expenses	30 000.0	30 000.0	30 000.0	30 000.0	30 000.0
Net Profit	102 800.	44 200.0	44 200.0	74 600.0	104 200.
ROI	2.115%	0.9847%	1.315%	2.072%	2.059%
Customer Service	100%	100%	100%	100%	100%
Productivity	4.42667	2.47333	2.47333	3.48667	4.47333
Inventory Turnover	4.64336	0.879147	5.45588	6.5375	6.51456

Amazing results, right?

9. POOGI

POOGI is the abbreviation for **P**rocess **O**f **O**n**G**oing **I**mprovement. What does it mean?

So far we did not finish our *Five Focusing Steps* approach. Step **5. Prevent Inertia from Becoming the Constraint** has not been realized yet.
It often happens that managers are happy we the results achieved by Full TOC and they leave it as it is. In such a moment human Inertia is triggered and the attention moves to other areas of interest. It would be much better and re-focus on the work that has been done and to further improve the system or a new constraint may pop up somewhere else.
Step 5 is a warning sign. It means to go back to step 1 of the *Five Focusing Steps* and seek the next constraint. If you do this repeatedly you are automatically in POOGI.