

E V O A P ↘

produce more, consume less

Testing results: Kudos Demo Model

Date: 24/09/2025

TECHNICAL TESTING RESULTS

During the final testing stages of EvoAPS it was important to validate the results from EvoAPS in different data models that could replicate the varied scheduling environments that typically see.

During the final testing stages of EvoAPS, it was important to validate the results from different data models that replicated the different scheduling environments that are typically seen.

These included:

- Single operation process
- Finite resource calendar patterns
- Multiple operation processes (Operation 10, 20, 30 etc.)
- Secondary constraints
- Sequence dependant changeovers
- Internal and external material dependencies
- Operational dependencies through a presented BoM (bill of material)
- Resources that could process multiple operations at the same time (such as cooking or curing ovens and heat machines)

During the testing, rather than test each element independently, models were constructed that could reflect combinations of these points so that real examples could be tested.

The same data models were constructed with varying numbers of operations so that the speed of achieving a result could also be tested.

Different strategies were tested in all models to see how different strategies affected the results and could be used to create a schedule.

Model tests were compared against Siemens Opcenter APS to give a benchmark. This was in part so we were able to easily validate the results, but mainly because Opcenter APS has a long history of creating very good scheduling results, making it an ideal benchmarking tool. We also we have over 30 years of implementation experience with Opcenter APS and are able to utilise advanced scheduling logic to build what is referred to as a 'trained model'. A 'trained model' is where an implementor configures Opcenter APS scheduling during an implementation to achieve a specific desired result. This could be built up using multiple scheduling rules, but the logic would always be heuristic based-and so no other result would be considered.

Models were tested against both due date order scheduling and against advanced scheduling logic.

MODEL 1 – GENERAL DEMONSTRATION MODEL

Demo model used for showing potential customers examples of how scheduling issues can be overcome.

Operation Count	Multiple Operation	Secondary Constraints	Sequence dependant changeovers	BoM dependencies	Material dependencies	Ovens
71	No	Yes	Yes	Yes	Yes	Yes

The model is designed to replicate scheduling at a fictitious food factory. Orders are represented by a single operation, but some have a BoM attached so that operation dependencies on both production and supply can be created. Demand and stock are also available so that supply and dispatch information can be visualised also. Constraints are included to show how labour can be applied to some parts of the process.

Sequence dependant changeovers and Ovens are also represented.

RESULTS COMMENTS

The improvements seen against the baseline result show a difference in end time of over 11 hours, but it is worth noting that this included an off-shift period of 8 hours. So this could be reflected as closer to 3 hours 38 minutes improvement. The setup time was also dramatically reduced by 3 hours 23 minutes to only 1 hour 35 minutes.

There was also an improvement on the finishing time of the schedule against the trained model. This was less (as we would expect) at 37 minutes, but a big improvement was still found against the setup and this was reduced by 2 hours 33 minutes.

In both cases the setup time does not have an impact on the end time of the overall schedule. But EvoAPS revisits the schedule and looks to continually improve the result until either a number of generations has been achieved, a specific time of day has passed or the user has paused or ended the process.

The conclusion is that we are able to use the strategies to achieve results that are typically only achievable by key individuals assisting with training the model based on their accumulative domain knowledge.

Results from base line test – schedule by due date

	Baseline test	EvoAPS		
Schedule Creation method	Due Date	Demo Strategy	Difference	Improvement
Schedule Start	02-11-2020 05:30	02-11-2020 05:30	-	-
Schedule End	03/11/2020 08:56	02/11/2020 21:19	11 h 38 mins	Yes
Make Span	1 day 3 h 27 mins	15 h 49 mins	11 h 38 mins	Yes
Total Operations	71	71	-	-
Late Operations	2	0	-2	Yes
Late Orders	2	0	-2	Yes
Total Setup Time	4 h 58 mins	1 h 35 mins	3 h 23 mins	Yes

Results from the Advance scheduling logic

	Opcenter APS	EvoAPS		
Schedule Creation method	Advanced logic	Demo Strategy	Difference	Improvement
Schedule Start	02-11-2020 05:30	02-11-2020 05:30	-	-
Schedule End	02/11/2020 21:56	02/11/2020 21:19	37 mins	Yes
Make Span	16 h 26 min	15 h 49 mins	37 mins	Yes
Total Operations	71	71	-	-
Late Operations	0	0	-	-
Late Orders	0	0	-	-
Total Setup Time	4 h 08 mins	1 h 35 mins	2 h 33 mins	Yes

As we can see improvements were made against both the baseline test and the advance scheduling logic.

TESTING PROCESS

The process of testing in EvoAPS is simply done by building a strategy and running a set of data against this. In this case a few different approaches were used.

STRATEGIES

Category	Sub-Strategy	Progress (approx.)
REDUCE WIP RISK	WIP Risk	95%
	Schedule Length	85%
	Over Due Date	75%
DEMO STRATEGY (DEFAULT)	Schedule Length	90%
	Total Setup Time	10%
	Independent Changeovers	80%
KUDOS DEMO STARTER	Reduce Setup Time	10%
	Reduce Schedule Length	90%
	Reduce Late Operations	10%
KUDOS DEMO STARTER	Task Over Due Date	90%
	Task Make Early	10%
	Make Operations Early	10%
KUDOS DEMO STARTER	Reduce Oven Door Openings	10%

But eventually settling on using two strategies, Demo Strategy and Kudos Demo Starter. These would then be selected and the evolutionary process run and the results that gave would be analysed strategy adjust and re-run. These could be run concurrently so testing time was reduced.

COMPARISONS

Strategy	Strength	Service Level	Schedule End	Total Setup	WIP Risk	Current Strongest
Demo Strategy	High	100%	02/11/2020, 21:49:00	1h, 30 m	High	9102
Demo Strategy	High	100%	02/11/2020, 21:19:00	1h, 35 m	High	9535
Demo Strategy	High	96%	03/11/2020, 09:20:00	1h, 37 m	High	8463
Kudos Demo starter	High	90%	03/11/2020, 11:33:00	1h, 34 m	High	87107

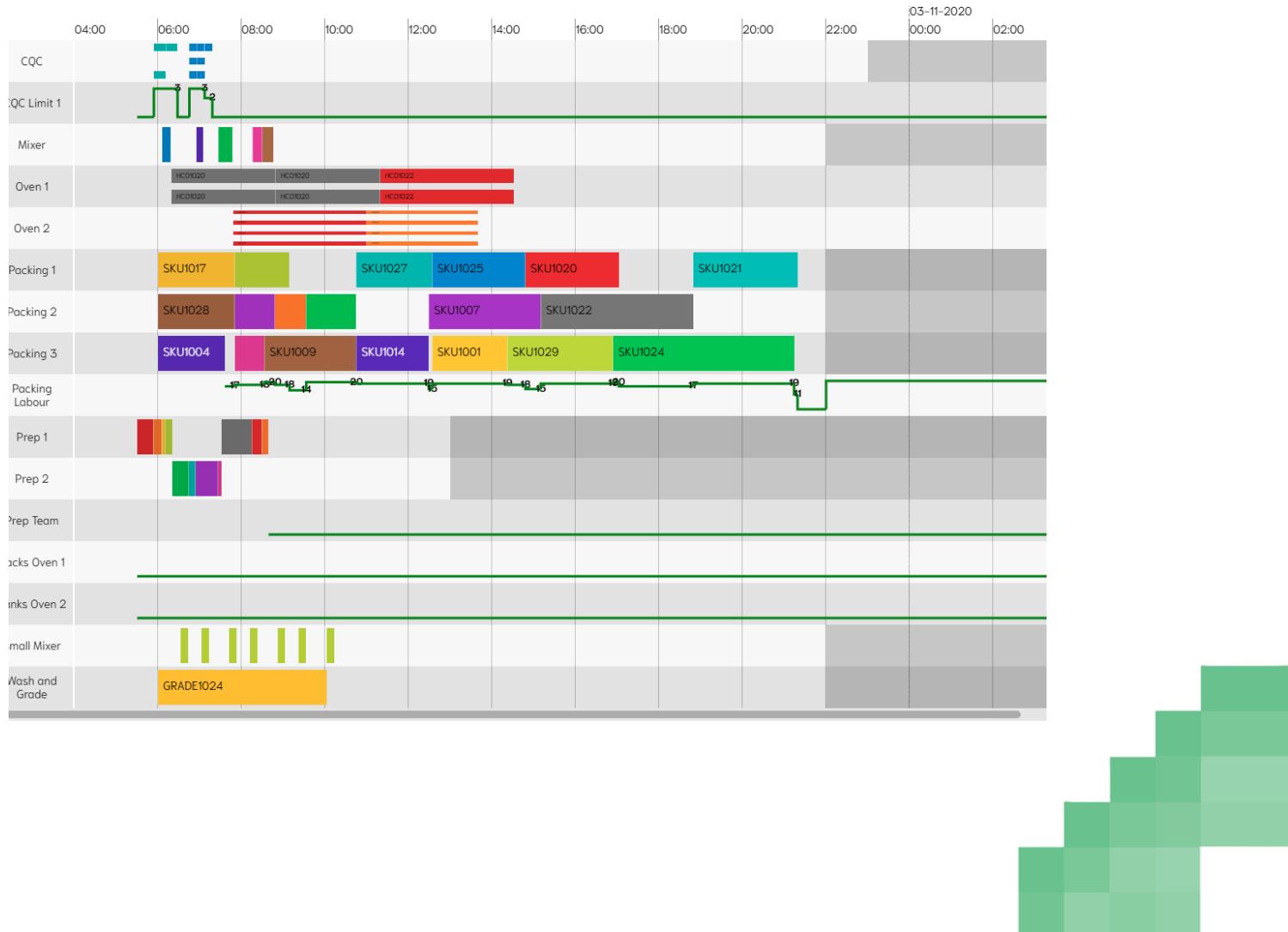
VIEWING RESULTS IN EVOAPS

The differences in 2 schedules can be viewed using the ‘Comparison’ feature. The comparison compares a schedule that is currently in place in the system providing the data left-hand column (in this case Opcenter APS), then the results from EvoAPS are in the column on the right.

The differences in the two schedules are show in brackets, with green highlighting an improvement and red showing a negative change.

Demo Strategy	High	100%	02/11/2020, 21:19:00	1h, 35 m	High	9535	Finished
Rerun	Insights	Comparison	Chart				
EXTERNAL SCHEDULE (169)							
Scheduled Tasks:	71			Scheduled Tasks:	71 (0↔)		
Avg order length:	1h, 28 m			Avg order length:	1h, 17 m (-11 m↓)		
Total Run Time:	3 d, 19 h, 7 m			Total Run Time:	3 d, 16 h, 52 m (-2 h, 15 m↓)		
Total Setup Time:	4 h, 58 m			Total Setup Time:	1h, 35 m (-3 h, 23 m↓)		
Orders before due date:	69			Orders before due date:	71 (+2↑)		
Orders after due date:	2			Orders after due date:	0 (-2↓)		
Total late time:	17 h			Total late time:	0 minutes (-17 h↓)		
Schedule End Time:	03-11-2020 08:56:33			Schedule End Time:	02-11-2020 21:19:00 (-11 h, 38 m↓)		
Schedule Length:	1d, 3 h, 27 m			Schedule Length:	15 h, 49 m (-11 h, 38 m↓)		
WIP Risk:	high			WIP Risk:	high (0↔)		
EVOLUTION SCHEDULE (272-9535)							

A simple ‘read only’ Gantt chart is also available for viewing the final schedule.

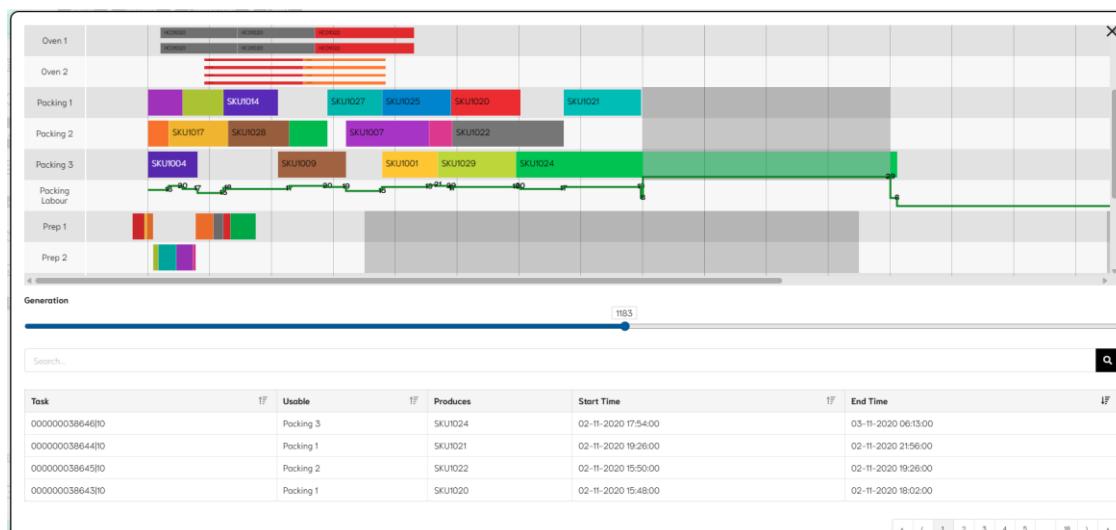


Generation history showing when and how improvements in the overall schedule are achieved can be viewed in EvoAPS. This is so that any improvements found by the Evolutionary Algorithm can be viewed and strategies and their impact on results be more easily understood.

As the number of schedules completed increases and the improvements made, these can be seen visually on the Generation History graph below. The graph shows each element of the strategy represented along the Generation timeline and when a change to that element occurs. Combined, these make up the overall fitness result of the schedule.



More detail on each generation is available in both a Gantt chart and tabular format by selecting the 'Show Generation History' button. The user can scroll between the generations and see the chart and the table update in line with each change.





Are you ready to re-think scheduling?

EVOLAPY