Understanding the Drivers (which settings matter)

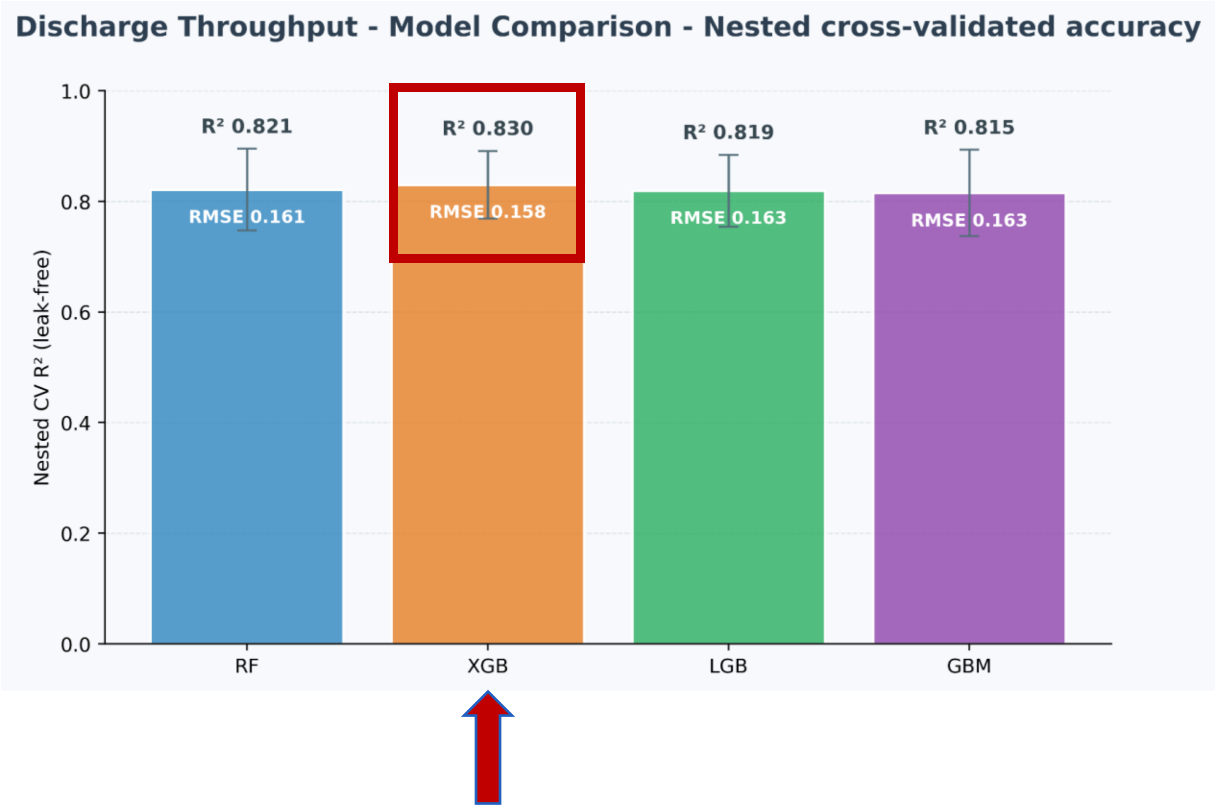
Half Kernel Yield (%)

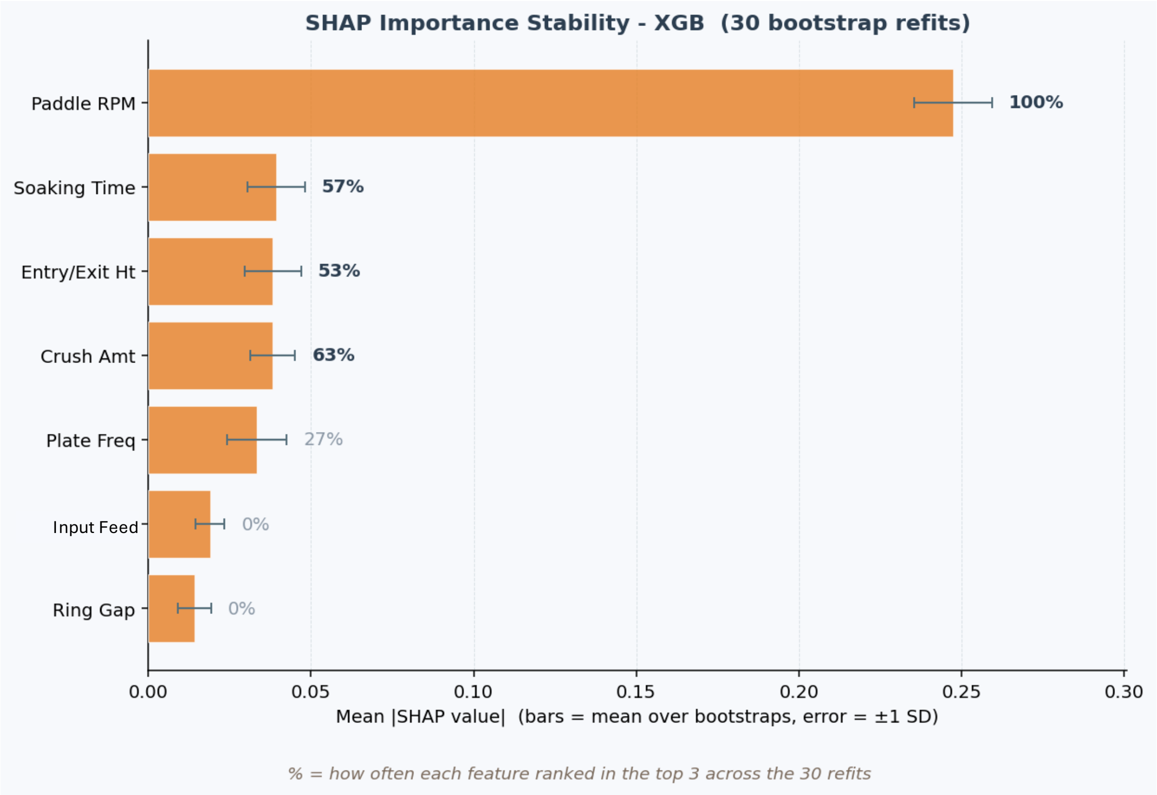
Discharge Throughput (%)

Sticktight (%)

Main drivers of Discharge Throughput

- Mean |SHAP| = average magnitude of a feature's effect on model prediction
- % = frequency the feature ranked in the top 3 across 30 bootstrap refits





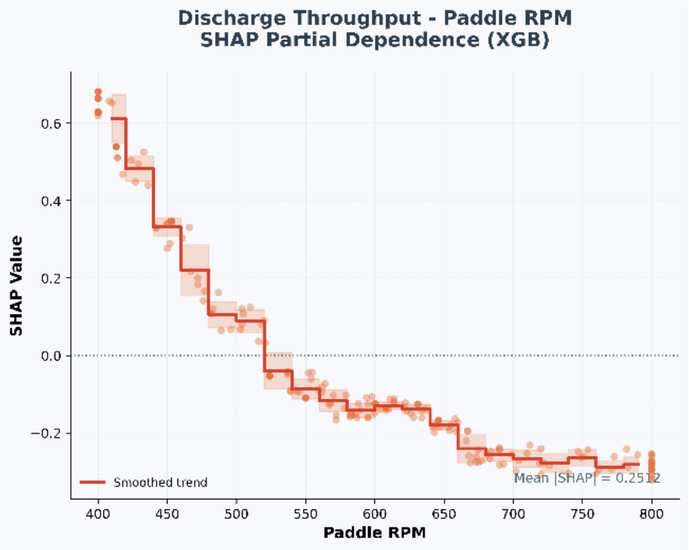
What is the direction of each factor's effect on Discharge Throughput?

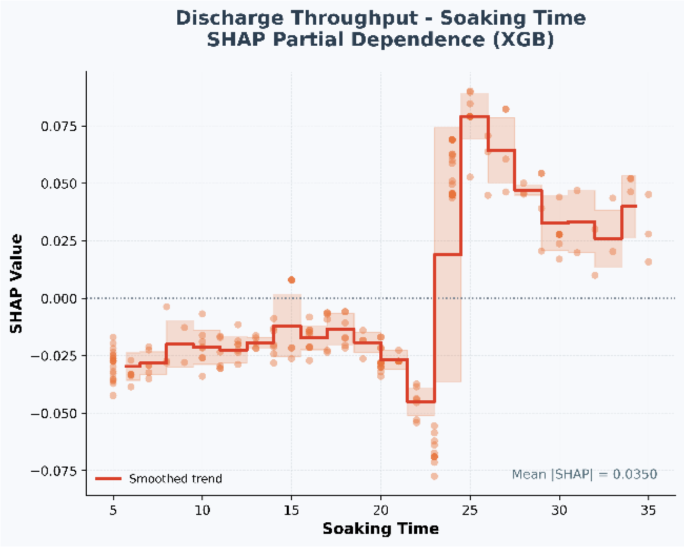
How much dependent variables pushes the GBM prediction of Discharge Throughput up or down from its average prediction.

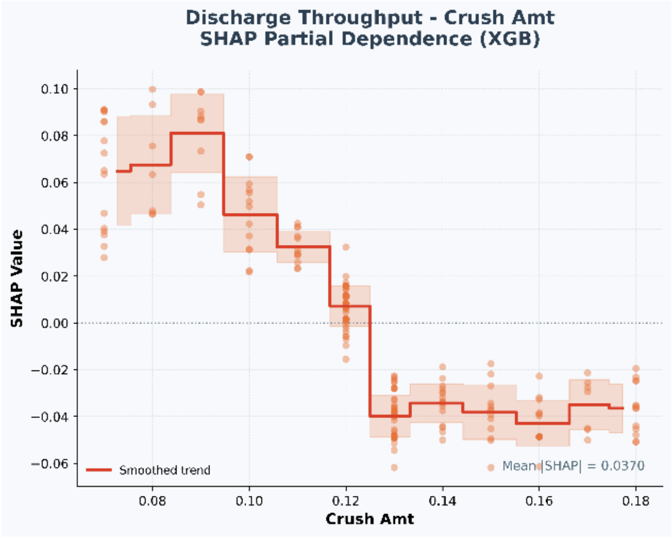
↑ Paddle Shaft RPM -> ↓ Discharge Throughput

↑ Soaking Time -> ↑ Discharge Throughput

↑ Crush Amount -> ↓ Discharge Throughput







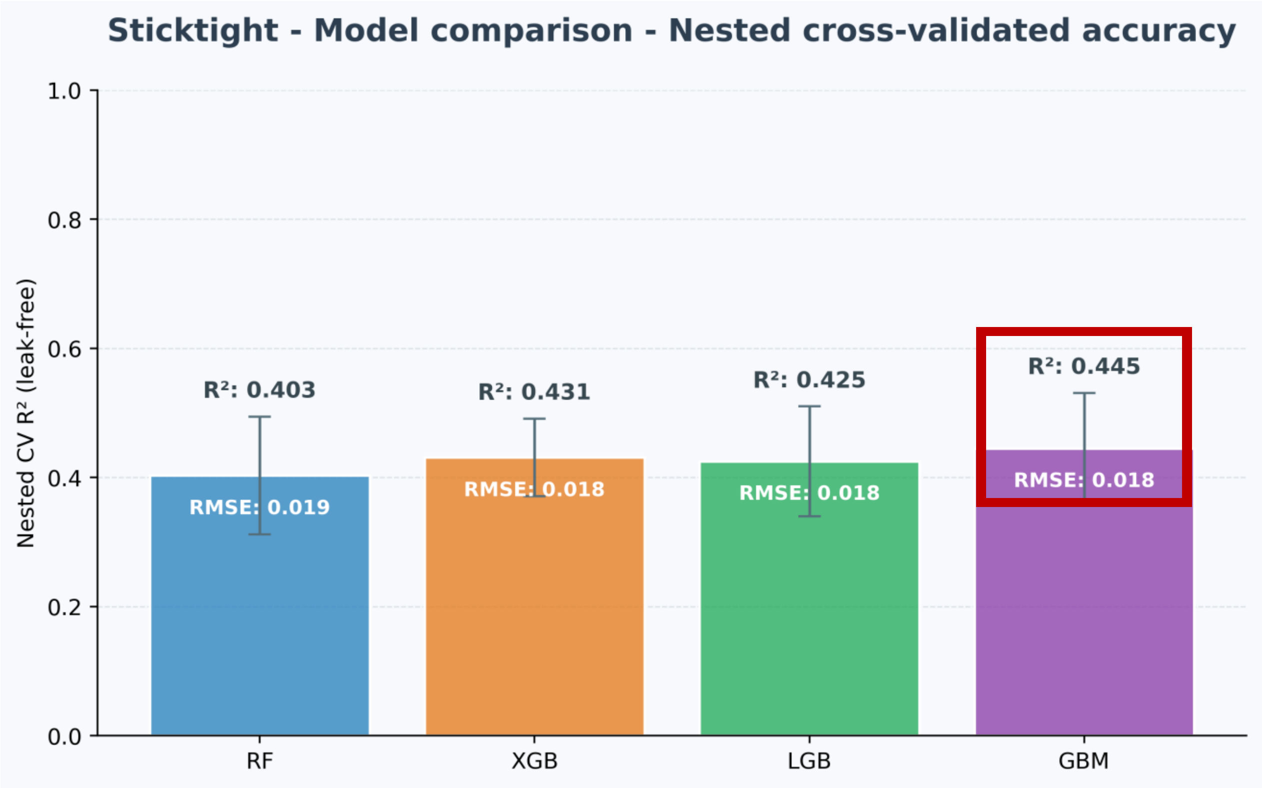
Understanding the Drivers (which settings matter)

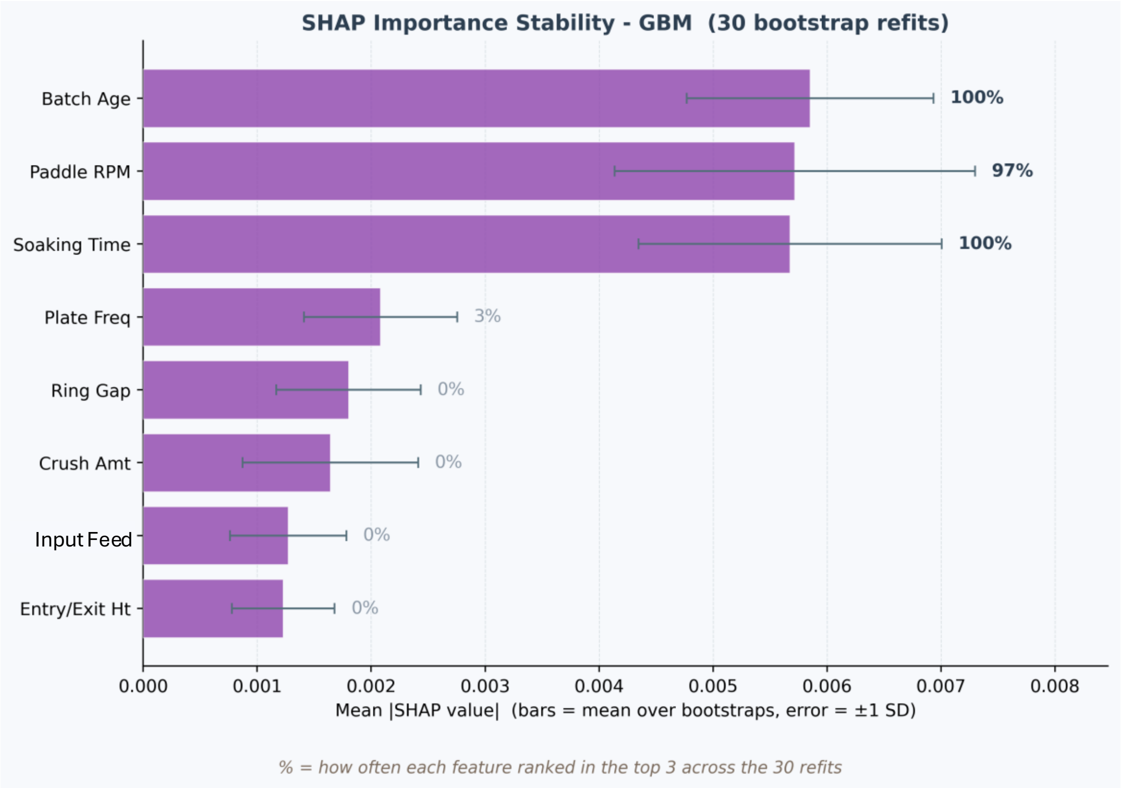
Half Kernel Yield (%)

Discharge Throughput (%)

Sticktight (%)

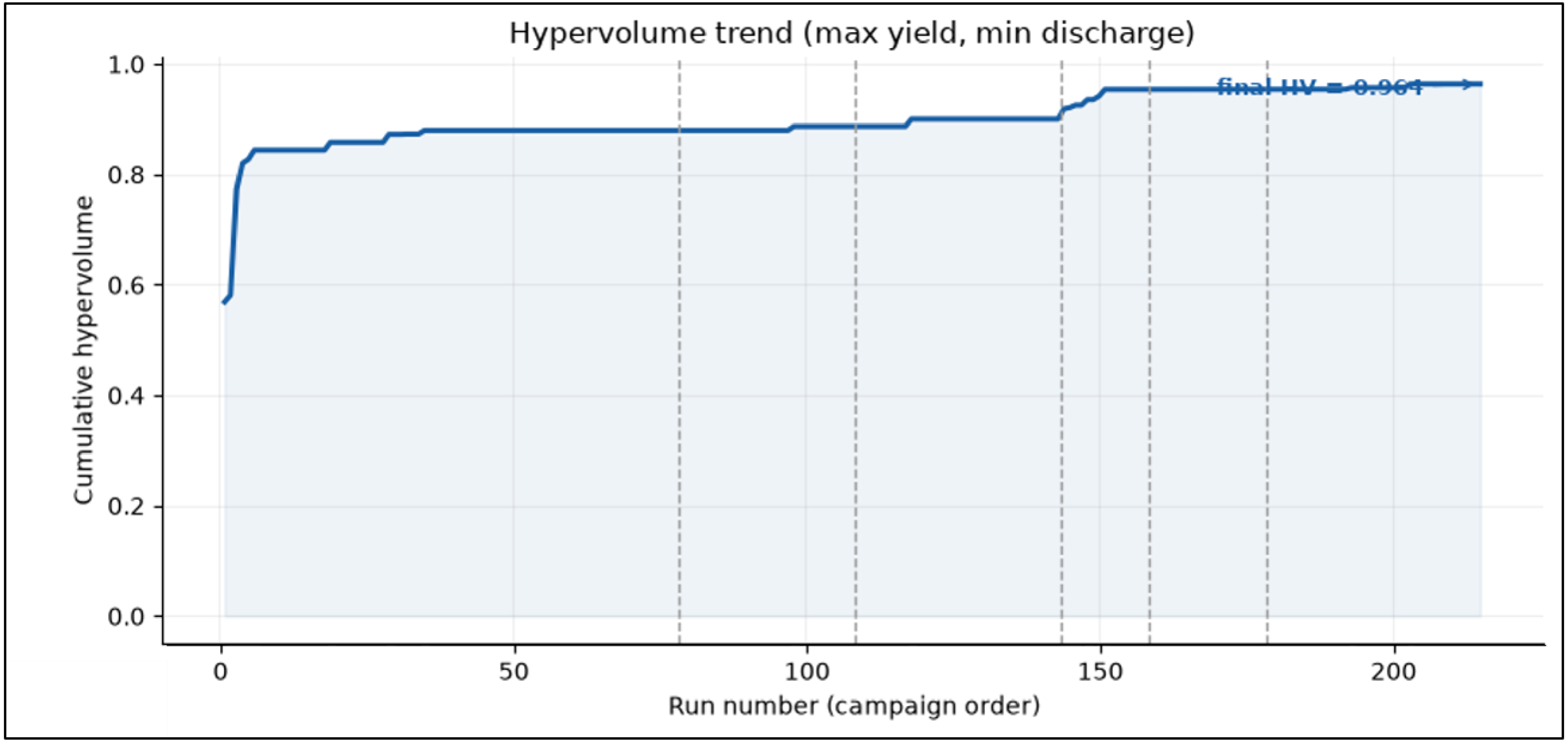
Main Drivers of Sticktight





Multi-Objective Bayesian Optimization (MOBO)

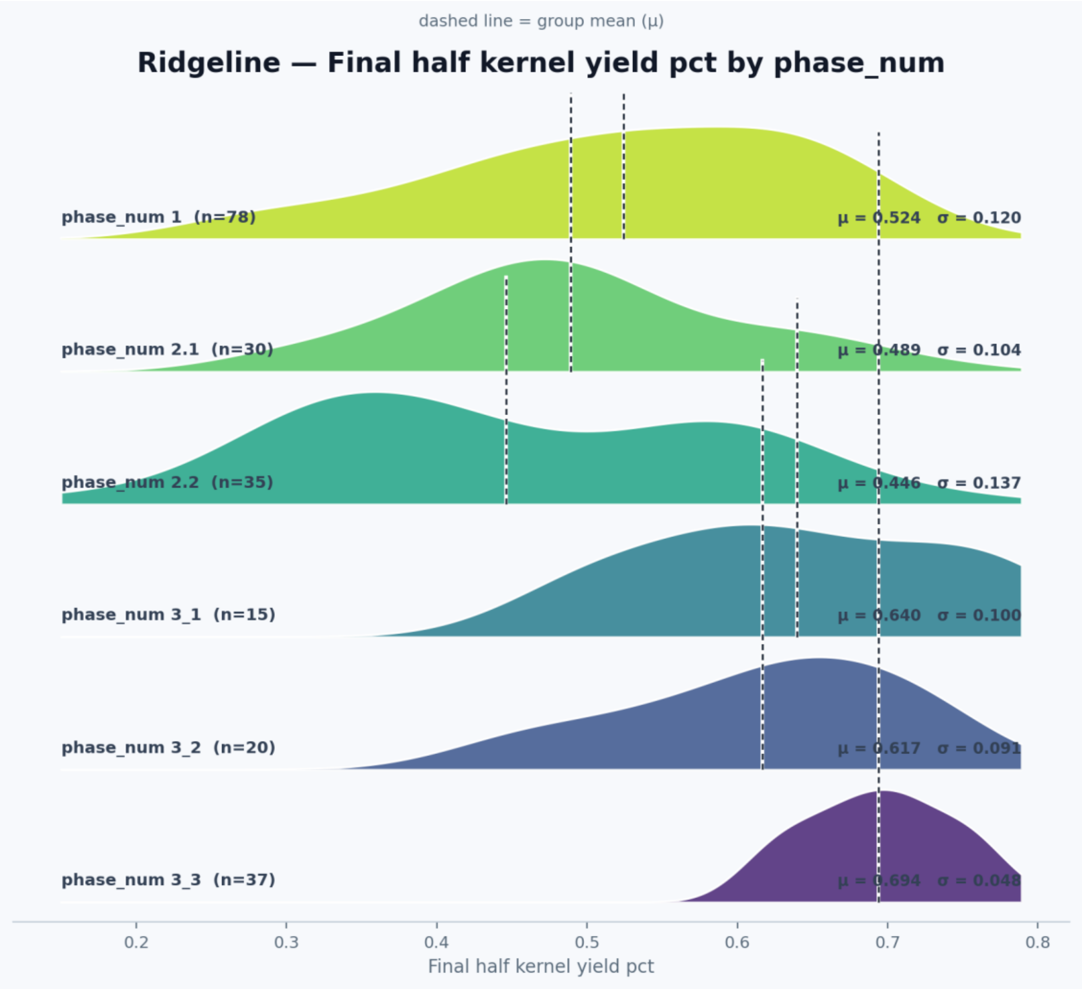
Hypervolume Improvement in MOBO



Distribution of Final Half Kernel Yield (%) by Phase Number

Mean: 52% ⟶ 70% ↑

SD: 12 % ⟶ 4%↓

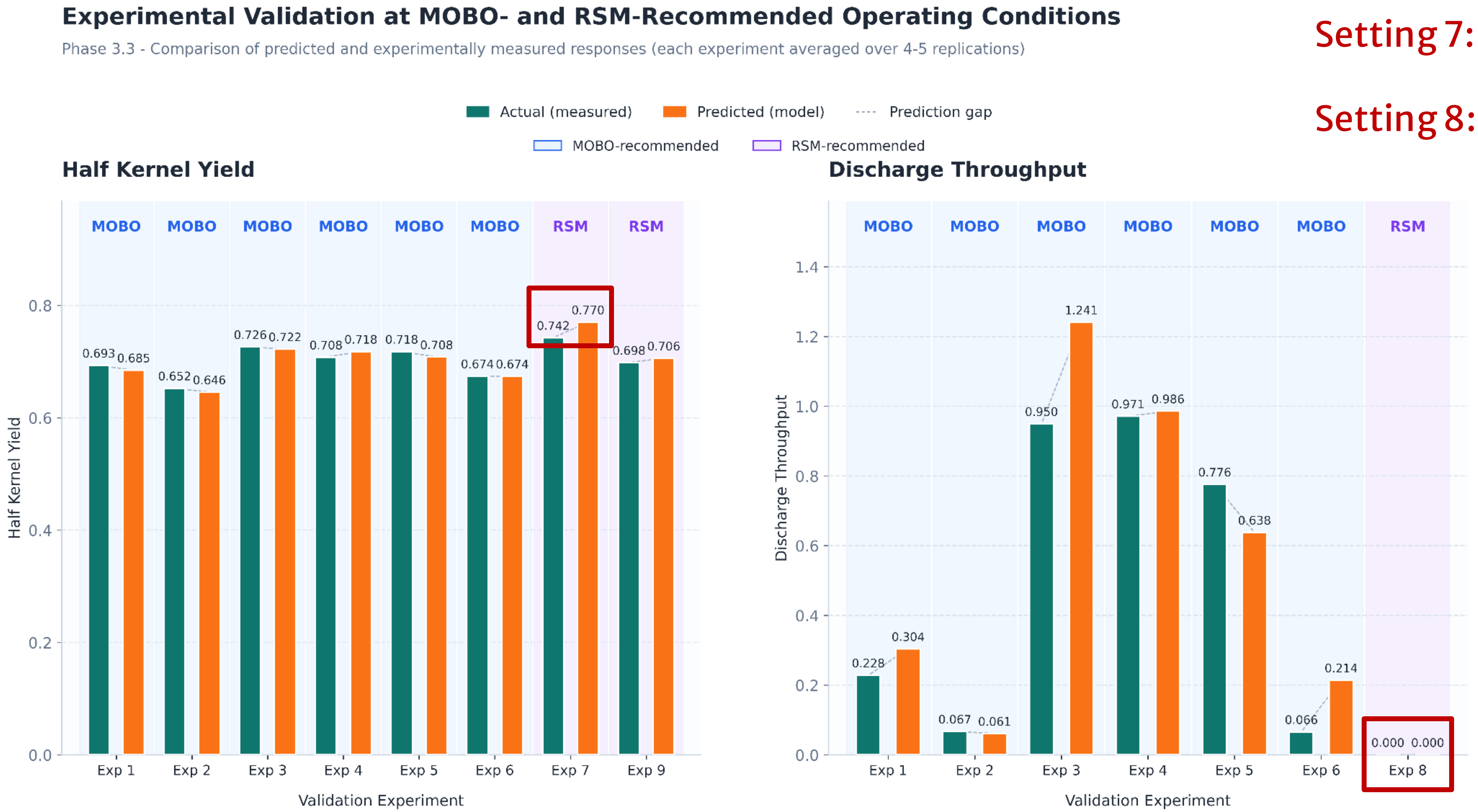


Validation on Phase 3_3 and Optimization Result

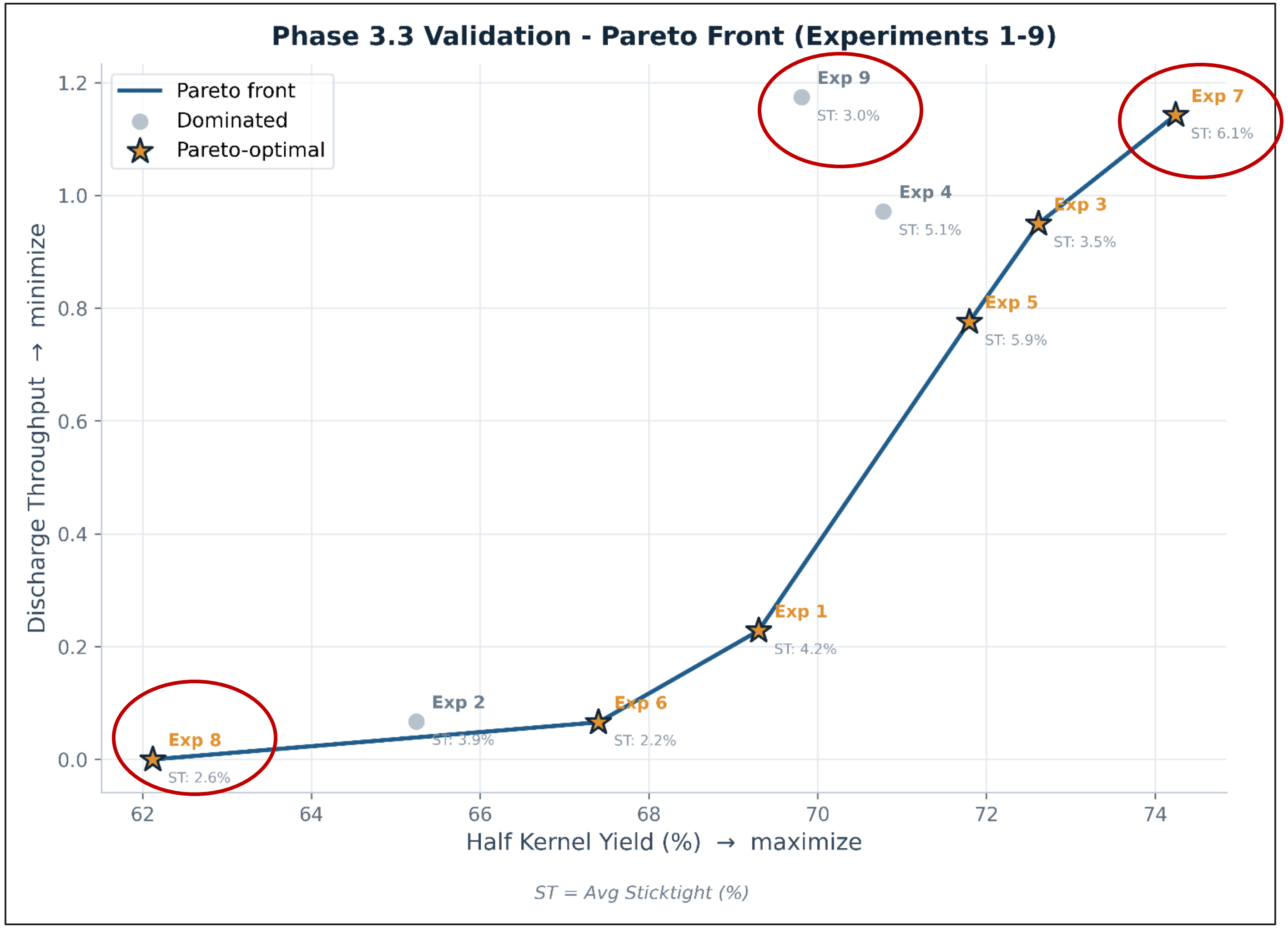
Validation Result - Phase 3_3

Setting 7: 75% Half Kernel Yield

Setting 8: 0% Discharge Throughput



Pareto Front



We can raise Half Kernel Yield by accepting more Discharge Throughput

- Clear frontier (6 optimal): Exp 8, 6, 1, 5, 3, and 7 are Pareto-optimal

- **Exp 8** = <u>minimum</u> Discharge Throughput (~62%)
- **Exp 7** = <u>maximum</u> Half Kernel Yield (~74%).

- **Exp 9** = the short-soak win. With the lowest Soaking Time (15 h), it still reached ~70% Half Kernel Yield, however, had high Discharge Throughput (~1.17). Thus, it's dominated high Half Kernel Yield achievable with less Soaking Time, though not on the Pareto Front trend.

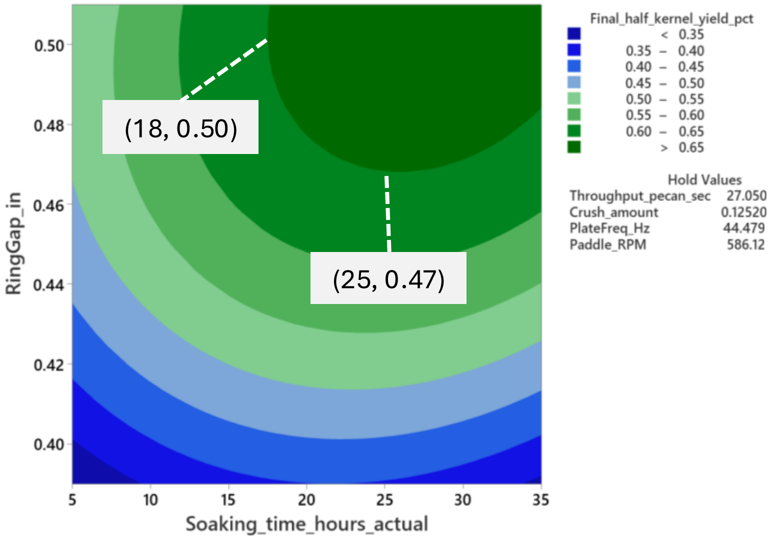
- 2.2% ≤ Sticktight ≤ 6.1%

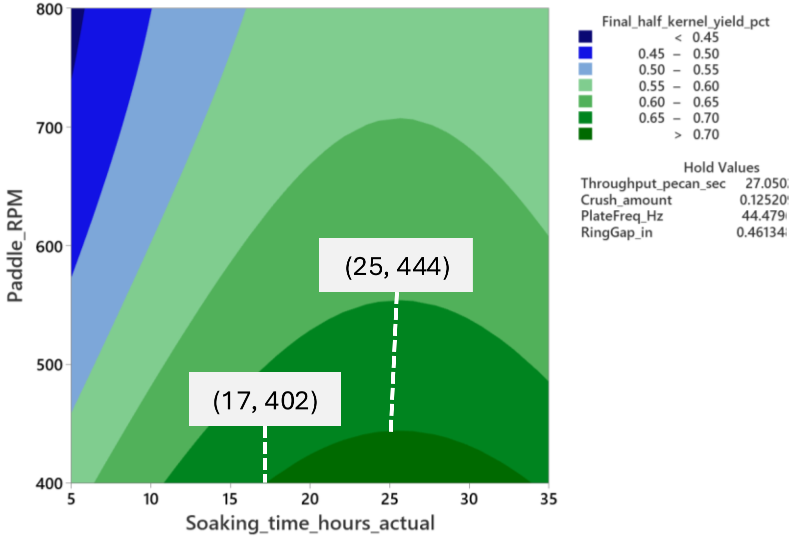
Two Final Prediction Models

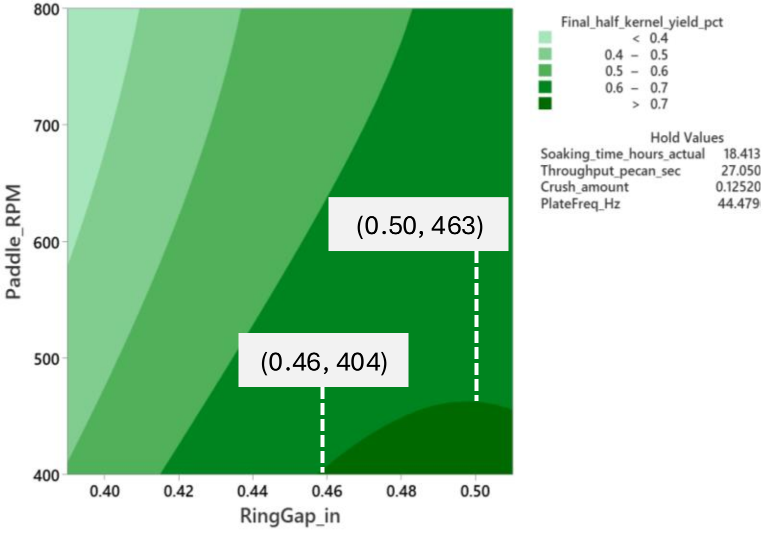
- Response Surface Methodology (RSM)
- Gaussian Process Model (GP)

Response Surface Contours Plot - Half Kernel Yield

The predictive R^2 improved across successive phases, reaching 84% for Half Kernel Yield.

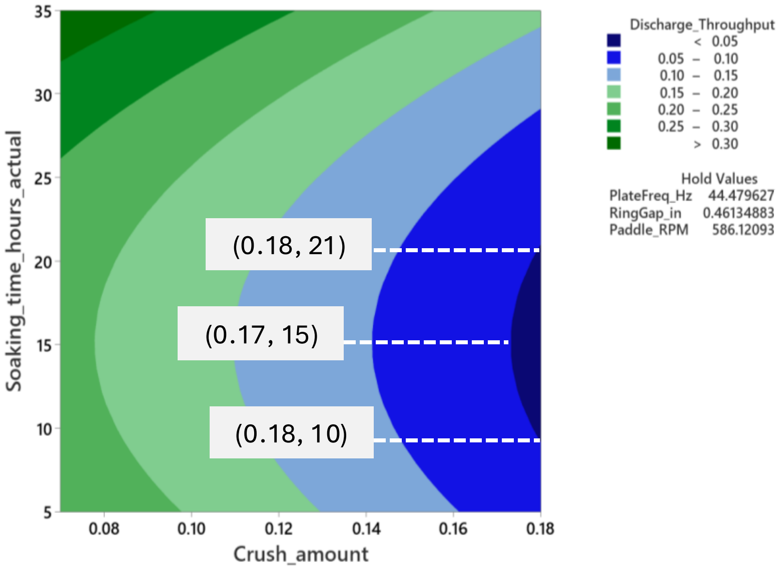


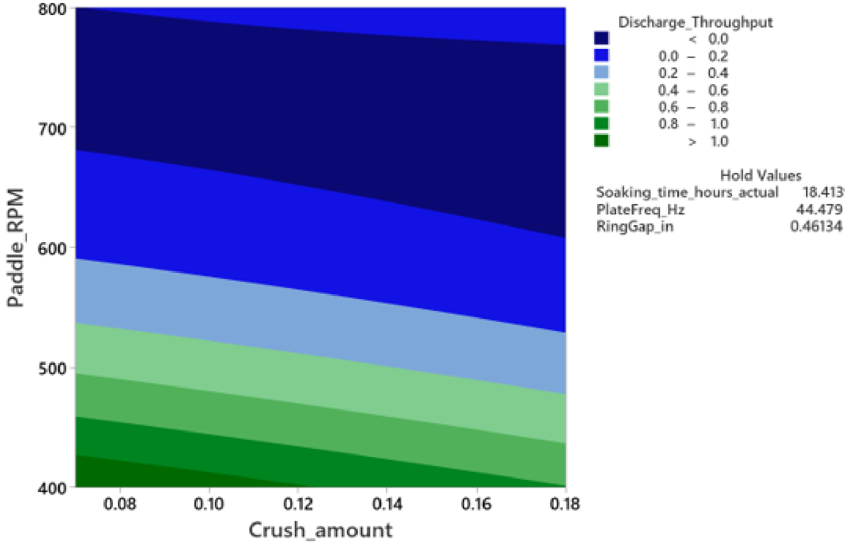


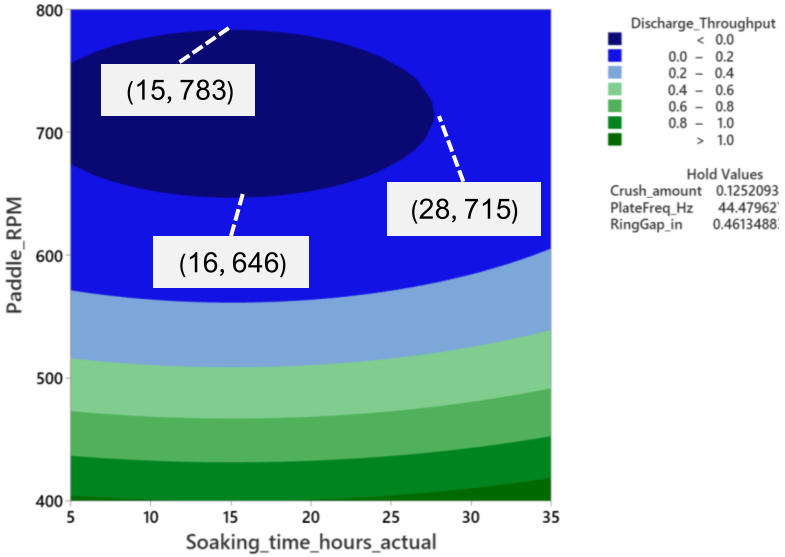


Response Surface Contours Plot - Discharge Throughput

The predictive R^2 improved across successive phases, reaching 85.7% for Discharge Throughput.

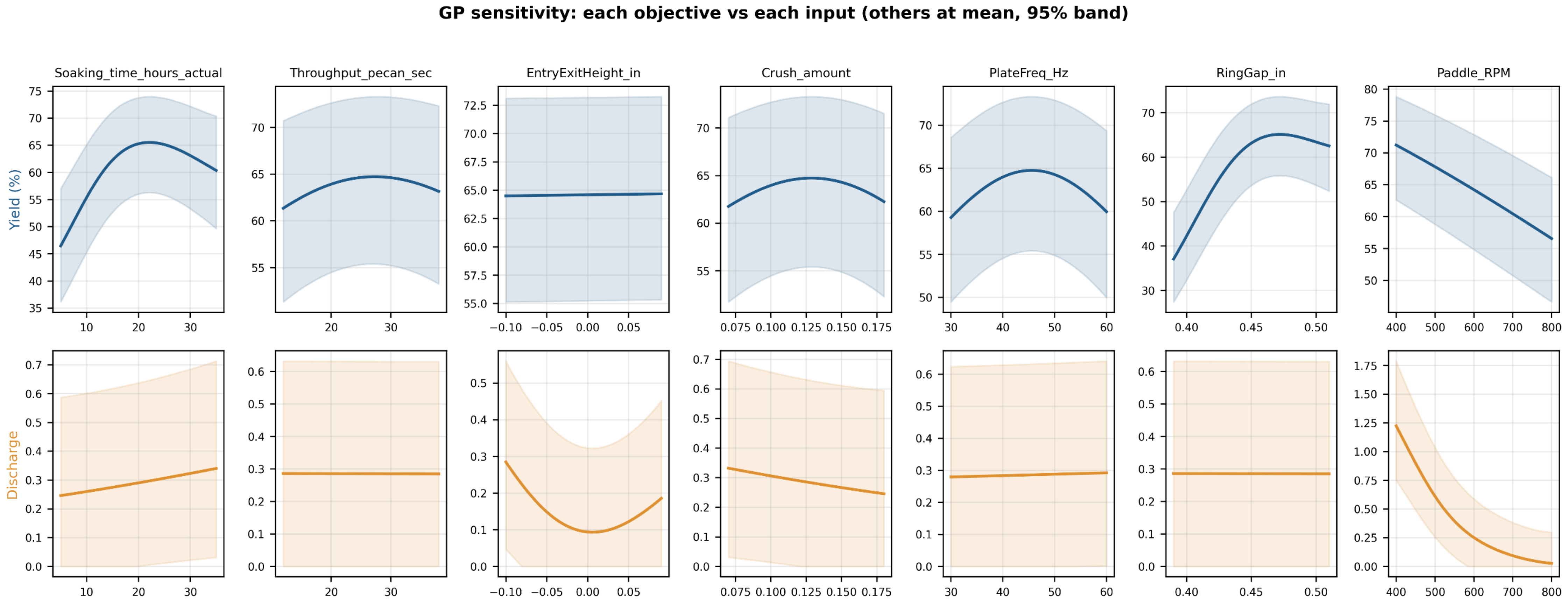






Gaussian Process Models' Sensitivity Plots

The predictive R^2 improved across successive phases, reaching around 85% for Half Kernel Yield and Discharge Throughput.



Recommendations

- **Disclaimer**: The experiments were done in a controlled lab environment. Results are based on our environment, pecan variety, and machine parameters ranges. The trends presented are valid, however, the absolute numbers in a facility will likely not be the same.

- **Note:** Manage the Half Kernel Yield–Discharge Throughput trade-off -> select balance on the Pareto front graph.

- **Main drivers:** Paddle Shaft, Ring Gap, and Mid Kernel Moisture.

Recommendations

Correlations findings:

- ↑ Half Kernel Yield:

  ↓ Paddle Shaft

  ↑ Mid Run Kernel Moisture

  ↑ Ring Gap

- ↓ Discharge Throughput:

  ↑ Paddle Shaft

  ↑ Mid Run Kernel Moisture

  ↑ Crush Amount

Recommendations

Variable	Half Kernel Yield – Priority	Discharge Throughput – Priority	Balanced Priority (Recommended)	Units
Inputs				
Mid Run Kernel Moisture	~8-9	~7 – 7.5	~7.6-8.2	%
Air Drying	15	15	15	minutes
Input Feed Rate	25 – 27	25 – 26	25 - 26	pecan/sec
Entry/Exit Height Diff	-0.06 – 0.09	0 – 0.02	0.01 – 0.02	in
Crush Amount	0.12 – 0.13	0.13 – 0.16	0.13 – 0.16	in
Plate Contact Frequency	27 – 52	45 -48	45 - 48	Hz
Ring Gap	0.47 – 0.50	0.45 – 0.48	0.48 – 0.50	in
Paddle Shaft	400 – 455	600 – 646	520 – 550	RPM
Drum	30	30	30	RPM
Outputs				
Half Kernel Yield	~72 – 75	~62 – 65	~67 – 69	%
Discharge Throughput	~78 – 114	~0 – 7	~7 – 23	%

Conditioning

Cracking

Shelling

High significant values

Future Work

- **Publications:** The design of experiment methodology, optimal machine parameters recommendations, prediction models, and improvements.

- **Experiments:**
    - Studying how the process can be optimized while reducing Soaking Time.
    - Linking the crack type and developed models for optimization of the process in the Multi-Stage.

Bruna Valiate LinkedIn

Ayoub Heydarzade LinkedIn

All slides available: https://pecanresearch.uga.edu/

Thank you!

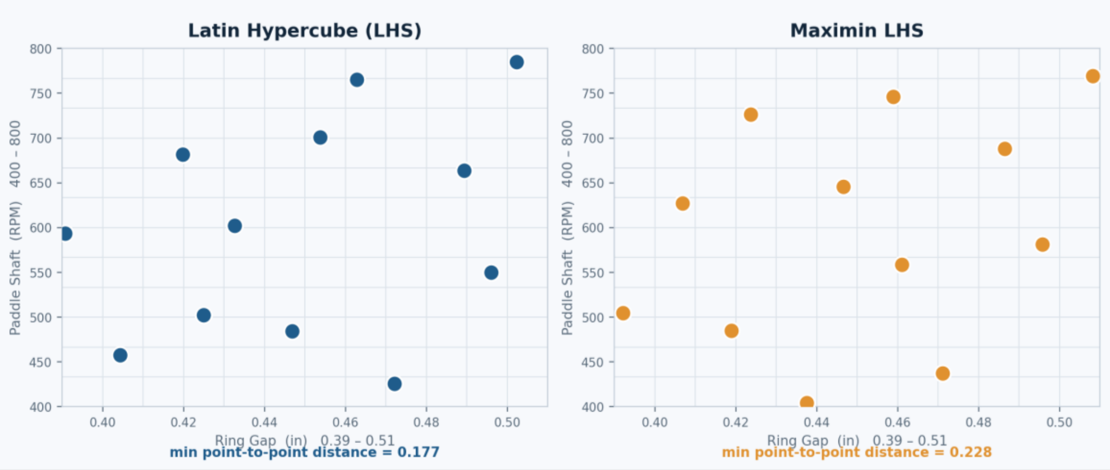
Questions & Comments

Sampling 7 Settings of the Process using Max-Min Latin Hypercube Sampling (LHS)

At the beginning, the goal was not to find the optimum immediately. The goal was to understand the design space efficiently.

Latin Hypercube (LHS): Splits each variable into equal levels and samples every level exactly once, so all 7 settings are evenly covered. But points can still clump by chance.

Max-Min LHS: A Latin hypercube that also maximizes the minimum distance between point; keeping the same even coverage while spreading runs as far apart as possible. Better fill, no gaps.



Note: Plots show only a 2D projection; the actual distance is computed in 7D.

Max-Min LHS helped to:

- Cover the full ranges of all 7 variables
- Avoid having many experimental runs too close to each other
- Provide information from different regions of the 7-dimensional space
- Give the initial surrogate/model enough diverse data to learn how the process behaves

How the Multi Objective Bayesian Optimization Engine Works

The Smarter Loop: it learns from every run and improves each round.

1 Learn form the data

Build two models, that predict performance, show how confident the predictions are and account for run-to-run variability

2 Search and score the possible settings

Rate every possible setting by how much it could improve our best results; one score, measured in a noise-aware way (qLogNEHVI score)

3 Choose next run

Pick the next experiments: some that already look excellent, and some that we're unsure to keep learning.

4 Track the score

Hypervolume improvement: One number for how good our set of best trade-offs is. It climbs fast at first, then levels off

5 Run & Repeat

Run them, add the results, and go again. Each round is smarter than the last.

↻ *repeat*

The score we track

"Hypervolume improvement"

When we stop

When that score stops climbing; the best trade-offs aren't improving anymore. The search has converged, so more runs aren't worth it.

The trade-off: we can't maximize yield and minimize discharge at the same time; MOBO finds the **menu of best compromises (the Pareto Front)** to choose from.