Adaptive Design of Experiments using Bayesian Optimisation

Caitriona Ryan
triona.ryan@mu.ie

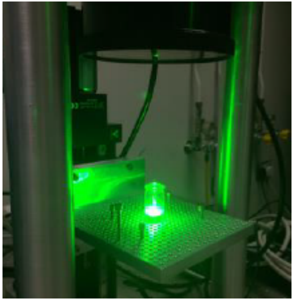
Maynooth University
National University of Ireland Maynooth

Outline

- I-Form Motivating Example
- Fixed vs. Adaptive Design of Experiments
- Bayesian Optimisation
- How can we use Bayesian Optimisation to help us adaptively design?
- Our first real-time I-Form example.
- Will it work for you? (break into groups to discuss)
- Browser app demo

I-Form Motivating Example

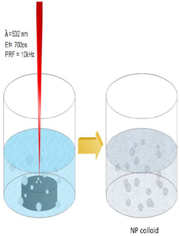
- Nano particle fabrication via laser ablation synthesis in solution (LASiS)
- e.g. Inks that can be used to 3D print electronics. . . smart phone of the future?!
- Industry partner example: Switch to copper instead of current silver solar cell interconnects as cheaper, more environmentally sustainable.



- INPUTS/SETTINGS: laser power, liquid flow speed, laser focal position, time, repetition rate, scan speed,. . .
- OUTPUTS/TARGETS: Specific nano particle mean size, size distribution and optimal particle concentration

Pulsed laser ablation in liquids (PLAL)

Laser synthesis in liquid



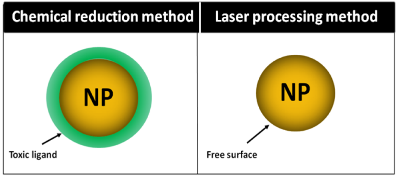
Simple & low cost: No purification

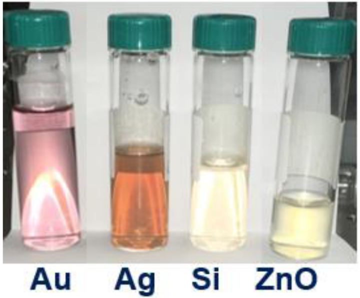
Clean : No-toxicity

Versatile :
Materials (Si, Au, Ag,...)
Biocompatible Ligand (Dextran, DNA,...)
In situ-functionalization

NPs: Unique surface chemistry

Value add of material – 40 €/g – 1000 €/g

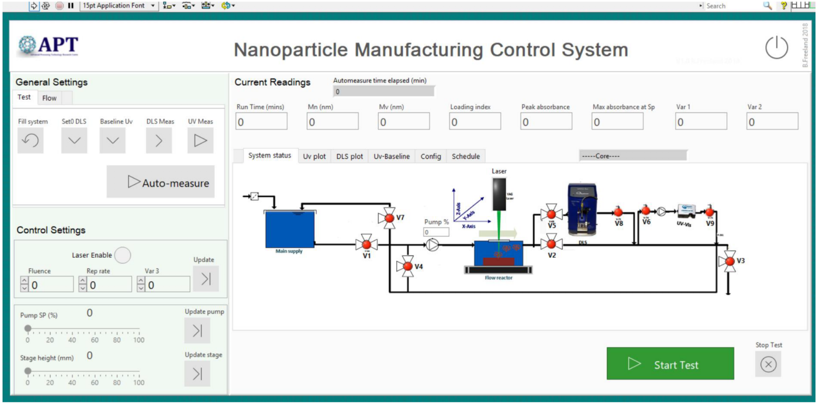




LaSiS System

APT
Advanced Processing Technology Research Centre

Full process control and test scheduling



Full process characterisation

Online process control

- Laser Fluence & rep-rate, scan-speed.
- Solvent flow-rate, pressure.
- Continuous/recycling flow.

Online process monitoring

- Particle size & quality – DLS
- Concentration – Uv-Vis

Fixed vs Adaptive Design of Experiments

- Fixed: the experimental design is fully determined beforehand e.g. factorial, central composite, latin hypercube, Taguchi
- Box Behnken (BB) is perhaps the design of choice for pragmatic engineers?! Just 17 experiments to carry out (12 + 5 central reps).

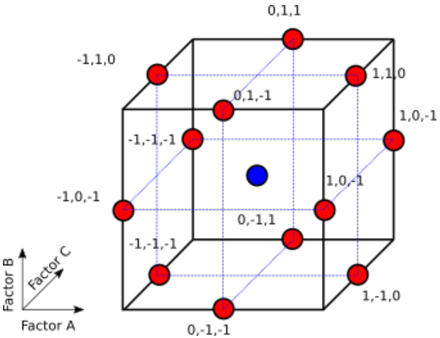
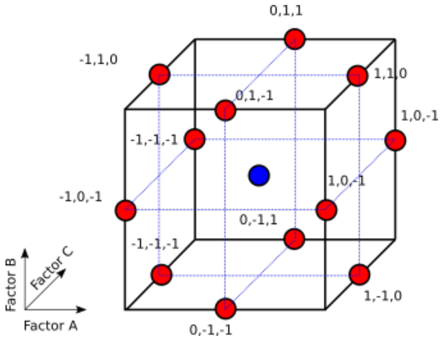
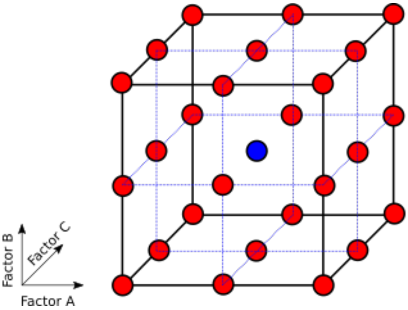


Figure 1: Box Behnken Design for 3 factors. Low = -1, Medium = 0, High = 1

Conventional Factorial Design vs BB





- Typically 3 runs of each setting of FF are carried out. (81 experiments in total)
- Theoretical results exist to show that BB with 5 central replicates is enough to characterise the response surface.
- Notice the corners are left out for BB! Can be useful if prohibitively expensive or not possible for your machine setup.

What is Adaptive Design of Experiments (ADoE)?

- A sequential process that incorporates each experimental result in turn to help decide on the next best design (input / machine settings at which to run the next experiment)
- The chosen designs will be from areas that have not been previously explored and areas where optimal target behaviour is observed.
- In contrast, fixed designs cannot adapt to features that appear during the experiment and do not learn from the data being gathered.
- Examples of adaptive design: Response Surface Methodology (RSM) i.e. Optimise the 2nd order polynomial surface using path of steepest ascent.
  - Non-parametric RSM uses GPs, neural networks and other more general models.

I-Form Industry Context

- Typically takes 20-100 builds to optimise a parameter recipe (Fully factorial is typically the fixed design of choice with perhaps some response surface (hill climbing) methodology).
- This is expensive! I-Form goal is to reduce this by $1/3$.
- I hope to convince you that adaptive design can contribute to this goal!

Advantages of Adaptive Design

- The design space can be explored cheaply and rapidly
- Prior / expert knowledge can be incorporated in a clever way
- Can balance multiple design objectives e.g. cost, target optimisation, modelling of the process.

What is Bayesian Optimisation?

- Bayesian optimisation is a machine learning tool used for global optimisation of an expensive black box system

STEPS:

1. Fit a Gaussian Process (GP) model over the design space given the prior experiment results
2. Optimise an acquisition function based on the parameters of the GP

The idea is to sample more densely in “interesting” areas

Why Gaussian Processes?

- Any statistical model that can produce predictions with quantified uncertainties should be OK.
- I like Gaussian processes because they are very flexible!
- Where a normal distribution will give you a scalar response f(x), a Gaussian process will give you a mean and variance of a normal distribution over the possible values of f at x.
- Wiki definition: A stochastic process (a collection of random variables indexed by time or space), such that every finite collection of those random variables has a multivariate normal distribution, i.e. every finite linear combination of them is normally distributed.

Gaussian Process Example - 1 Design Factor

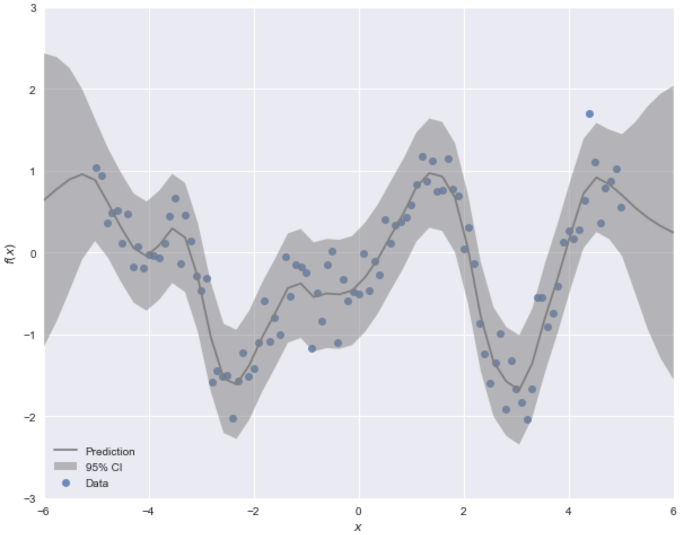


Figure 2: Gaussian process model - 1 design factor

Gaussian Process Example - 2 Design Factors

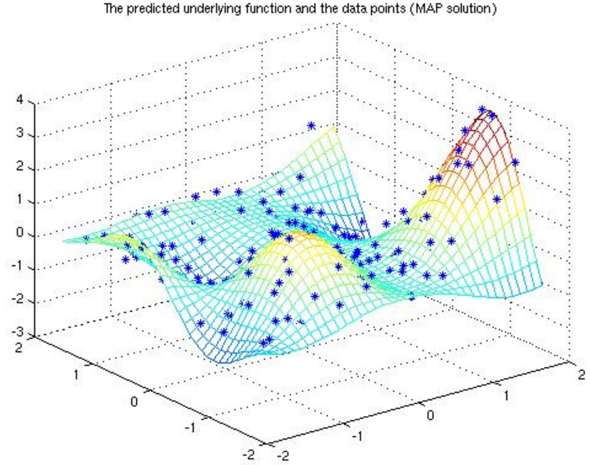


Figure 3: Gaussian process model - 2 design factors

For More Info on Gaussian Processes

- Andrew Parnell will talk more about GP models in the training course "Code surrogates and emulation with Gaussian Processes" on Fri Feb 19.
- (Thomas Huijskens - Bayesian optimisation with scikit-learn) is a really nice video for an intuitive GP explanation watch from 17 - 21 minutes.
- (The Kriging Model : Data Science Concepts) is a more mathsy blackboard explanation. (Note that kriging is the old name for GPs from geostatistics).

What is an acquisition function and how do I choose?

- There are many!
- Aim to balance exploration and exploitation based on the idea of expected improvement or efficient global optimization (EGO) (Jones et. al 98)

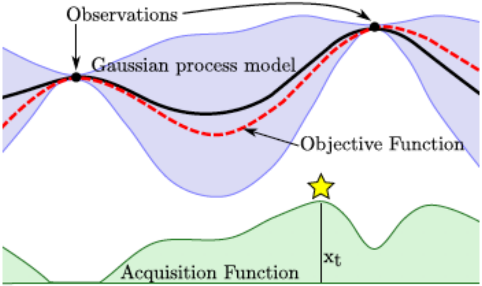


Figure 4: EGO

Figure from Greenhill et. al. 2020

What is an acquisition function and how do I choose?

- The idea is to compute an acquisition function based on the current model parameters (e.g. mean and variance of the GP model fit to the experiments carried out so far).
- Compute the gradient of the acquisition function.
- Use it to optimise and suggest best design for the next experiment (derivative-based searches in combination with evolutionary search algorithms to find the mode).

Acquisition Functions

- For noisy experimental data literature recommends

1. Approximate Knowledge Gradient (AKG)
2. Expected Quantile Improvement (EQI)

- We also suggest to use

3. Best predictions using the fitted model (make a grid of parameter values, calculate the predicted response at each grid-point, choose the biggest)
4. Random search (pick a random uniform number between your min and max for each input parameter / machine setting)

There are more! Do your research and / or invent some but be warned that you need to focus on noisy experimental data rather than computer experiments.

Approximate Knowledge Gradient

- AKG is the conditional expectations of the following improvement functions under the kriging model
- $I_n(x) = min_n(m_n(x^{n+1}) - min(m_{n+1}(x^{n+1}))),$
- where m_{n+1} denotes the kriging mean of step $n+1$.
- i.e. highest improvement between the minimum of the current and new kriging means.
- So it chooses a design expecting that the new observation will lower the kriging mean in the area. (i.e. this is for minimisation, to optimise a target simply use -ve values of the response!)
- See this Picheny paper (Picheny et. al., 2014 CSDA) for the full maths for noisy experiments

Approximate Knowledge Gradient

Figure from Picheny et. al. 2014

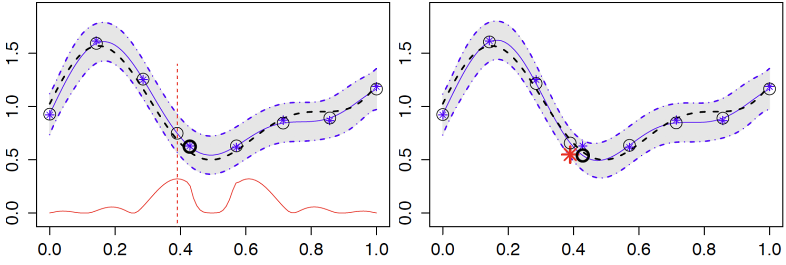


Figure 5: Initial (left) and updated kriging models, with observation chosen by AKG (right). The AKG criterion (multiplied by ten for readability) is represented in red (left). The point chosen is the one that is likely to provide the highest improvement between the minimum of the current and new kriging means (bold circles). Here, the new observation is not chosen close to the current minimum but lowers the kriging mean in this region.

Expected Quantile Improvement

- Similar to AKG but instead of aiming to lower then mean of the GP, it chooses a design expecting that the new observation will lower the quantiles.
- The kriging percentile $q_n(x) = m_n(x) + \Phi^-1(\beta)s_n(x)$ as a measure of reference, an EQ improvement between steps n and $n+1$ is defined in the noisy case as:
- $I_n(x) = (min_n(q_n(x^i)) - Q_{n+1}(x)))$
- where $Q_{n+1}(.)$ is the quantile function (random, seen from step n) of the kriging model updated with a new measurement at $x_{n+1} = x$.

Expected Quantile Improvement

Figure from Picheny et. al. 2014

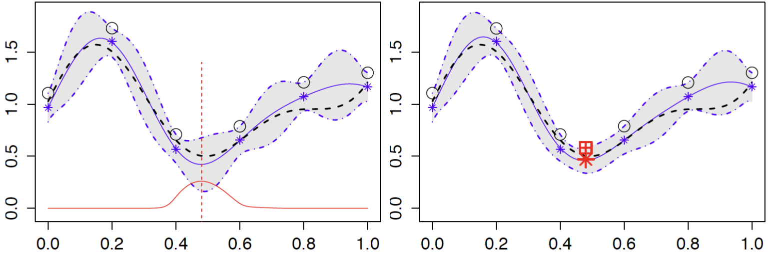


Figure 4: Initial kriging model (left) and kriging model updated with an observation chosen by EQI (right). The EQI criterion is represented in red (left). Here, a *quantile improvement* is achieved since the quantile of the updated kriging model at the new point (square) has a lower value that the lowest quantile of the initial kriging model.

BREAK 10 minutes

Back to our LaSiS Adaptive Design Experiment

- Our target output/response is production / efficiency / ablation rate. Higher is better. 20 – 50% expected. (We are also interested in particle concentration and even a distribution of particle sizes. . . this requires more research, very interesting problem!)
- Model Inputs i.e. Design Variables to optimise (We picked 3 for now)

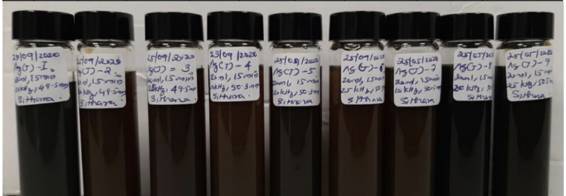
Parameter	Minimum	Maximum	Unit
Time	10	30	minutes
Scan Speed	1.8	2.6	mm/s
Repetition Rate (laser)	10	30	kHz
Stage Height	Fixed at	50.3	mm
Laser power (fluence)	Fixed at	1.46	W
Flow Rate (solvent)	Fixed at	100	mL/min
Pressure (solvent)	Fixed at	n/a	bar
DI Water	Fixed at	20	ml

- Note: All settings were continuous, accurately set to 1 decimal place and to the nearest second for time.

Prior Data

- This is usually done even for a fixed design to gain expert knowledge of suitable high, medium, low values of each design parameter
- Time = 15 for all, SS = 2 for all, 3 each of Stage height and Repetition rate

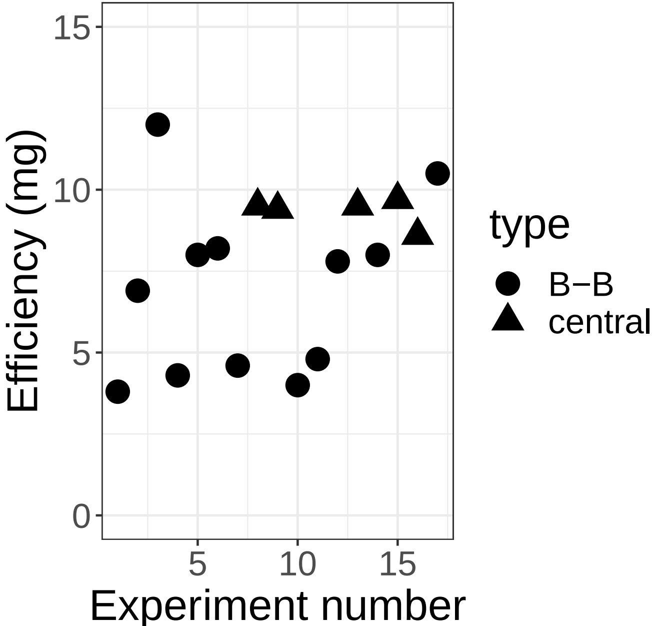
No	Time (min)	DI Water (ml)	Flow rate (ml/min)	Laser power (%)	Scan Speed (mm/s)	Stage height (mm)	Repetition Rate (kHz)	Efficiency (mg)	Size Distribution (nm)
1	15	20	100	100	2	49.5	10	3.3	1.52
2	15	20	100	100	2	49.5	20	3.5	21.32
3	15	20	100	100	2	49.5	25	5.0	15.58
4	15	20	100	100	2	50.3	10	3.7	19.54
5	15	20	100	100	2	50.3	20	4.5	16.34
6	15	20	100	100	2	50.3	25	3.9	14.17
7	15	20	100	100	2	50.5	10	4.1	21.31
8	15	20	100	100	2	50.5	20	6.5	19.69
9	15	20	100	100	2	50.5	25	5.3	17.86



Live LaSiS Experiment

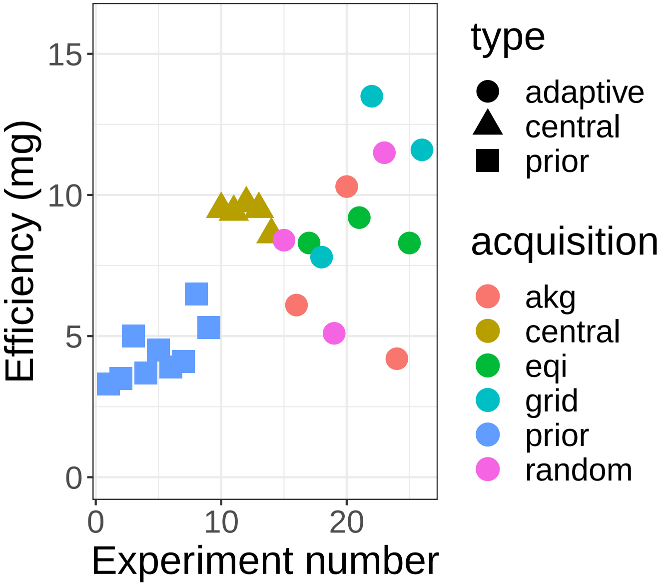
- Experiments were run one at a time and took around 10 minutes to setup after each one.
- Box Behnken fixed setup 17 experiments
- Goal is to compare BB fixed design with the BO adaptive design!
- Fit GP given prior data and 5 central reps, optimise acquisition and feed the design suggestions back to the engineer one at a time.
- Via email with the engineer. I learned a lot from this! Now that I am confident in the approach, we have built an app to do this so for 3 inputs 1 output, you can run this very easily independently of me!
- We chose to cycle through these in turn in this order: Random, AKG, EQI, Grid search.

BB Results

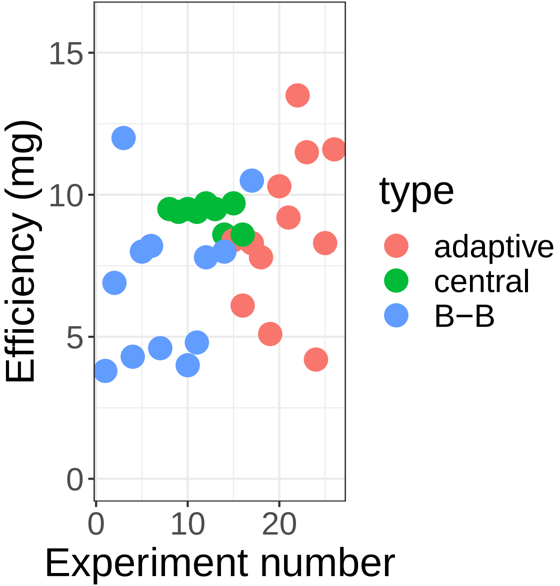


15
10
5
0
Efficiency (mg)
5
10
15
Experiment number
type
B−B
central

BO ADoE Results

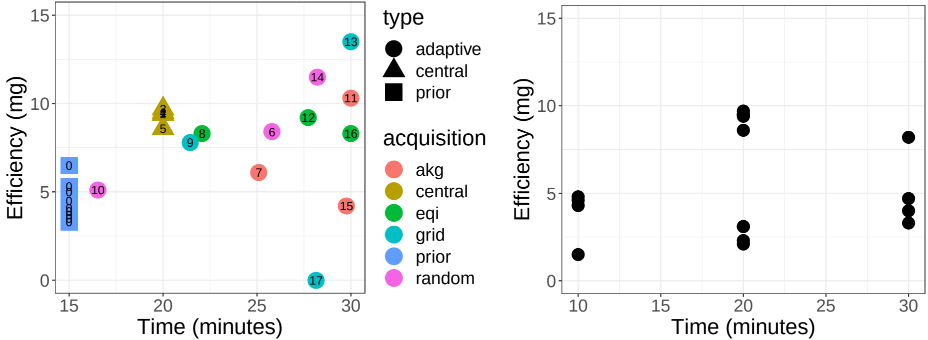


BB vs. BO ADoE Results



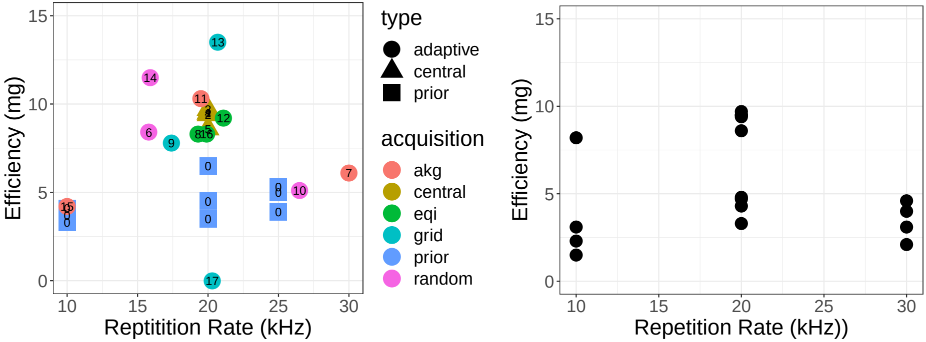
A closer look at time vs. efficiency

Optimum time using GP model on the final dataset is 28.9 minutes



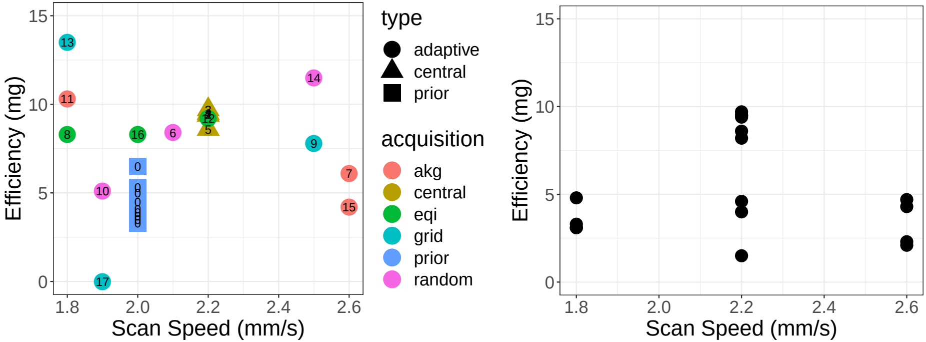
A closer look at repetition rate vs. efficiency

Optimal repetition rate using GP model on the final dataset is 20.3KHz



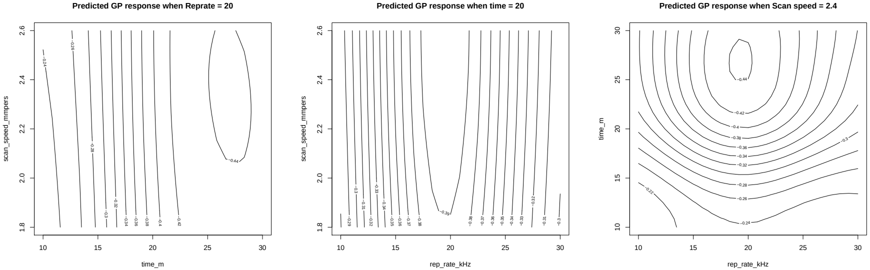
A closer look at scan speed vs. efficiency

- Optimal scan speed using GP model on the final dataset is 1.9mm/s



2D GP contour plots with 3rd factor fixed at central value

- Note the model minimises -efficiency (mg/ml). This is equivalent to maximising efficiency! e.g. contour of -0.44 represents $0.44 * 20 = 8.8$mg



Where to go from here with LaSiS

- Repeat some bad BB experiments as problem with the laser on the last day of the experiment. Hopefully the ADoE will still do better!
- Engineers are happy to stop here and write up a paper comparing the BB results and the BO ADoE results
- Budget / time allowing, I'd like to zoom in on the optimum design for both approaches and do some more experiments at both to make sure we weren't just lucky!
- We have performed 17 experiments for each. e.g. this Nature paper uses 22 training experiments and does 9 iterations of 4 at a time, that is 22+36=58 experiments $>>$17!
- We may try the fully factorial design (27 or up to 81 experiments). After how many experiments does the adaptive design clearly outperform the conventional design?
- Try for a few other materials.
- Research novelty here is in comparing BB fixed DoE to Adaptive DoE.
- Also Adaptive and/or Bayesian approaches quite rare in the AM context.

Summary: Advantages of Bayesian Optimisation adaptive desig

- The design space can be explored cheaply and rapidly.
- Industry context 20-100 builds to optimise a recipe. Can we reduce this?
- Can have up to 15-20 dimensions / inputs / machine settings and up to 3ish targets.
- OPEN Q: What is the trade off between number of design factors and experimental budget to get "good" results?
- How do we incorporate multiple targets / responses (multivariate response BO)?

- Caveat: there are no theoretical guarantees of its performance but we can see practically that it's working! And it's bound to work as its using the data as its being generated!

Where to go from here - more open questions

- What is the best way to reflect multiple experimental goals? e.g. cost of materials, cost of time, parameter inference, model choice (is GP model better or another competing model, we can choose designs that will inform this)
- How can we use simulation models (thanks Covid!) to improve our results?
- Can we develop / find an algorithm for a pragmatic adaptive design using a clever balance of both simulations AND real experiments.
- Develop Bayesian Adaptive Design approach (utility/acquisition is a function of the Bayesian posterior distribution, another days ?years training!). This would be a lot less pragmatic! Slow. Typically <5 inputs, 1 output. But would have theoretical guarantees of success rather than just experimental.

Software

- We used R: Dicekriging to fit the GP model; Diceoptim to optimize the AKG and EQI criteria (Picheny et. al., 2014 CSDA) (Kriging is the original name for Gaussian Process modelling comes from geostatistics)
- mlrMBO package (Bischl et. al., 2017) use EQI and AEI but not AKG; used in 2 manufacturing papers; "AI for Materials Science: Tuning Laser-Induced Graphene Production" and "Bio-like Composite Microstructure Designs for Enhanced Damage Tolerance via Machine Learning"
- Warning: Software written for deterministic computer experiments don't suit noisy engineering experiments.
- e.g. Python SciKitLearn use EI but EQI is better for noisy data. Matlab DACE only deterministic. GPfit in R is not suitable for noisy data (the nugget is set to be the smallest value that will avoid singularity, meaning that the nugget is never estimated.).
- Helpful paper: "Comparison of Gaussian process modeling software" (Erickson et. al., 2018)

Please stop recording now.

BREAK 10 minutes

- The good news is that we have developed a browser app so you can try this without coding if you have a suitable experiment.
- After the break we will come up with some ideas of how to use this in I-Form projects. I will demonstrate the app. You can try it and we can break into groups to discuss.

Welcome back

Do you have a suitable experiment that you could try this? Past, present, future, or imagined!

Before I demonstrate the app please consider the following.

- Describe your experiment?
- Which 3 input parameters / machine settings are you most interested in?
- What is your target output?
- Do you want to minimise or maximise this target?
- What is the range of each of the parameters? min and max?
- What other parameters need to be fixed? Is it possible to keep these fixed across multiple experiments?
- How long does each experiment take?
- How long does it take to measure your response?
- Can you do one at a time or how many will be in each "batch"? (4 or 5 is ok. 1 is better if practical.)

App demo and cautions!

https://aminshn2.shinyapps.io/Experiment/

Need to include us 3 in authorship if using this app (Myself, Andrew, Amin app dev)

INITIALISATION

- Decide on your 3 input variables
- Decide on 1 target output
- What are your units?
- What are the min and max settings on your machine / experimental setup?
- Is this full range interesting? Can you go narrower based on your expert knowledge?
- Choose a min and max design setting to search between

PRIOR DATASET

- To get the modelling going, we need to have 4 rows of data.
- i.e. the prior data needs to be bigger than the dimension of the space and it needs to have a bit of a variation for it to work.
- What prior data do you have?
- Examples are - previous experiments / build, 5 central repetitions (this is what we've been doing as it gives the model a head start in estimating the noisiness of the response surface), 4 experiments set at random design inputs.
- Write the initial (first build) data into excel and export to .csv file.
- Only include the relevant columns (3 target columns 1 response column) no extra columns or it may break the app!

Upper and Lower Bounds

- The easiest and safest approach re bounds is to decide on upper and lower values for each factor and then copy and paste it into the app each time.
- Be careful of the comma placement. It will break the app if you get this wrong!
- e.g. Sithara's lower and upper limits are: 10 , 1.2 , 10 and 30 , 2.8 , 30
- e.g. Josiah's lower and upper limits are: 130 , 600 , 40 and 190 , 1000 , 70

time_m	scan_speed_mmpers	rep_rate_kHz	efficiency_mgperl
15	2	10	0.165
15	2	20	0.175
15	2	25	0.25
15	2	10	0.185
15	2	20	0.225
15	2	25	0.195
15	2	10	0.205
15	2	20	0.325
15	2	25	0.265
20	2.2	20	0.475
20	2.2	20	0.47
20	2.2	20	0.485
20	2.2	20	0.475

Other points to note

- How finely tuned are your input settings? E.g. laser power $= 131$ or 131.1 or 131.0001?
- I assumed to the nearest 0.1 for Sithara's project and the app.
- If you need to change this please either round up or contact Andrew Parnell.

Then try the app!

Notes tab from the app:

1. Having carried out your experiment, please update the dataset to include the factors and experimental result(s). The app can then be refreshed to optimise the settings for your next adaptive experiment.
2. This app provides 1 up to 4 settings per batch (# experiments per run) .
3. The smaller the batch the better for adaptive design!
4. I suggest cycling through the acquisitions in turn. First RANDOM, then AKG, then EQI, then GRID (following the Nature Comms paper).
5. Another option is to choose an acquisition function at random each time.
6. Should you need more than 4 in a batch, an easy option is to generate more random searches. (There are many other options such as using different models and or different acquisition functions but these require further work.)

What to do if the app breaks?

1. First run through each point in the notes tab of the app
2. Contact Andrew Parnell (MU, I-Form) / Mimi Zhang (TCD, I-Form)

Try the app

- Would anyone like to try the app individually or in groups?
- Feel free to PM each other to form groups and ask Angela to form a breakout meeting
- For the remainder of this session I am more than happy to talk to you separately or in groups about potential projects.