Combining game theory and Bayesian optimization to solve many-objective problems

Victor Picheny (INRA), Mickael Binois (Argonne national lab.), Abderrahmane Habbal (Université Côte d’Azur)

Mascot-Num, Nantes

March 23rd, 2018
Bayesian optimization context

“Black-box” model, multiple outputs

\[ \mathbf{x} \in \mathbb{X} \subset \mathbb{R}^d \rightarrow \text{Black-box} \rightarrow y_1 \rightarrow \vdots \rightarrow y_p \]

Working hypotheses: expensive to compute, complex \( y_i \)'s

- non-convex
- no derivatives available
- possibly observed in noise
- \( 2 \leq p \leq 20 \)

\( \mathbb{X} \): typically a box of dimension \( 2 \leq d \leq 50 \)
Multi-objective optimization

\[
\begin{aligned}
\min_{x \in X} & \quad y_1(x) \\
\vdots & \\
\min_{x \in X} & \quad y_p(x)
\end{aligned}
\]

Ensemble of non-dominated solutions

Pareto front (objective space) and Pareto set (design space)

Classical BO algorithm objective

Obtain a good discrete approximation of the Pareto set = non-dominated solutions along the entire Pareto front
Challenging situation: many objectives / restricted budget

Accurate Pareto set may not be attainable
- Many objectives $\Rightarrow$ very large Pareto set
- Not enough budget to cover all the solutions

Accurate Pareto set may not be desirable
Too many solutions to choose from

Our proposition: use game theory to seek compromise solutions

Game theory literature
Extensive for small discrete problems or convex objectives, scarce for expensive black-boxes
$\Rightarrow$ Explore BO alternatives for solving game equilibrium problems
Outline

1. Introduction
2. Games and equilibria
3. Solving games with Bayesian Optimization
4. Application to model calibration
5. Conclusion
Game Theory and equilibrium problems

*Game theory is the study of mathematical models of conflict and cooperation between intelligent rational decision-makers*

- Initially from economics
- Now common in artificial intelligence, engineering or machine learning

**Game**

- Multiple *decision makers* (*players*) with *antagonistic goals*
- Acceptable *compromise* = game equilibrium
Nash equilibrium problems (NEP)

NEP: canonical antagonist game
Each player tries to solve his optimization problem:

\[(P_i) \quad \min_{x_i \in X_i} y_i(x), \quad 1 \leq i \leq p\]

with \(x = [x_1, \ldots, x_p] \in X\). We assume here some territory splitting:

\[X = X_1 \times \ldots \times X_p\]

Definition: writing \(x = (x_i, x_{-i})\) (no actual permutation), \(x^*\) is a NE if:

\[\forall i, 1 \leq i \leq p, \quad x^*_i = \arg \min_{x_i \in X_i} y_i(x_i, x^*_{-i})\]

\[\Rightarrow \text{When everyone plays a NE, then no player has incentive to move from it}\]
An engineering example: multidisciplinary optimization

- Each team makes a decision with antagonistic goals
- Natural territory splitting: each team acts on its own set of variables

J.A. Désidéri (2012), Cooperation and competition in multidisciplinary optimization, *Comp. Optim. and Applications*
A PDE example: data completion

Steady-state heat equation

\[
\begin{cases}
\nabla \cdot (k \nabla u) = 0 & \text{in } \Omega \\
u = f & \text{on } \Gamma_c \quad \text{(Dirichlet)} \\
k \nabla u \cdot \nu = \psi & \text{on } \Gamma_c \quad \text{(Neumann)}
\end{cases}
\]

Boundary partially unobservable

- \( \partial \Omega = \Gamma_c + \Gamma_i \)
- Data to recover: \( u|_{\Gamma_i} \) and \( k \nabla u \cdot \nu|_{\Gamma_i} \)

Nash game: Dirichlet vs. Neumann

Games and equilibria

The Kalai-Smorodinsky solution

Disadvantages of Nash equilibria

- Assumes some \textit{territory splitting}
- In general \textbf{not efficient} (in the Pareto sense)

The Kalai-Smorodinsky (KS) idea

- Players start negotiation from a \textit{disagreement point}
- Progress towards an \textit{efficient} solution while ensuring \textit{equity of marginal gains}

KS solution

Intersection of the disagreement point - \textit{ideal point} (= utopia or shadow) straight line with the Pareto front

Kalai, Smorodinsky (1975). Other solutions to Nash’s bargaining problem, \textit{Econometrica}
Illustration: 2 objectives, 2 variables

Here: disagreement point = Nadir
Some properties

For 2 players case, KS is proved to be the unique bargaining solution which fulfills the following

Axioms
- Pareto optimality
- Symmetry
- Invariance w.r.t. affine transformations
- Restricted Monotonicity

For N players
- Efficiency, Symmetry, Affine invariance
- Effective selection device even when combined with refinement concepts based on stability with respect to perturbations [1].

Going further: robust KS using copula

Motivation: unlike Nash, KS is sensitive to non-linear rescaling

Why copulas?
- Nice link to Pareto optimality (see next slide)
- Insensitive to non-linear transformations on the marginals
Games and equilibria

Copulas and Pareto optimality

Pareto optimality and multivariate statistics

- Take: \( Y = (y_1(X), \ldots, y_p(X)) \), \( X \sim U(X) \)
- Pareto front = (part of the) zero level-line of the multivariate CDF \( F_Y \)
- Multivariate CDF = marginal densities + copulas

\[
F_Y(y) = C(F_1(y_1), \ldots, F_p(y_p))
\]

with \( C : \mathbb{R}^p \rightarrow \mathbb{R} \) copula, \( F_i = \mathbb{P}(Y_i \leq y_i) \) marginal densities

M. Binois, D. Rulliere, D., O. Roustant,
On the estimation of Pareto fronts from the point of view of copula theory

V. Picheny, M. Binois, A. Habbal
Bayesian optimization and game theory
March 23rd, 2018 14 / 37
Our proposition: KS in the copula space (CKS)

Assuming a distribution for $X$ (e.g. $X \sim U(\mathbb{X})$)

We look for the KS of $F_1(y_1(X)), \ldots, F_p(y_p(X))$

Bonus of copulas: since $F_i \left[ \min y_i(X) \right] = 0$, $F_i \left[ \max y_i(X) \right] = 1$

- Utopia point is $(0 \ldots 0)$
- Worst solution for all objectives is $(1 \ldots 1) \Rightarrow \text{new disagreement point}$
KS in the copula space (CKS)

On discrete spaces: rank space

CKS = Pareto optimal solution with closests ranks among objectives

Invariant by any (strictly) monotonic transformation of marginals

... but depends on X’s distribution
Outline

1. Introduction
2. Games and equilibria
3. Solving games with Bayesian Optimization
4. Application to model calibration
5. Conclusion
Bayesian optimization: two ingredients

- Quick-to-evaluate surrogate of the objective: **Gaussian process regression**
- Sequential sampling via maximizing **acquisition functions**

```
Initial set of observations

Objectives evaluation

Surrogate building

Acq. func. maximization
```

(nested loop)

(nested loop)
Gaussian process regression

Prior $\pi : y$ realization of a GP $Y$

Entirely defined by its mean $\mathbb{E}_\pi(Y(x))$ and covariance $\text{cov}_\pi(Y(x), Y(x'))$

Regression model $(\hat{y}) = \text{law of the GP conditioned on observations}$

(Parametric form assumed for mean and covariance, estimation by maximum likelihood)
Sequential sampling (EGO algorithm)

The acquisition function balances between *exploration* and *exploitation*
BO and equilibria

Regression: one GP model for each obj $y_i$ (correlation is neglected)

- GP is used to “denoise” the objectives (if needed)
- Multivariate regressor: $\mathbf{Y}(.) \sim \mathcal{GP} (\mu(.), \Sigma(.,.))$ with $\Sigma$ diagonal.

Sequential sampling boils down to finding a “good” acquisition function

$\Rightarrow$ Based on $\mathbf{Y}(.)$, which $\mathbf{x}_{\text{new}}$ should I visit next?

Canonical acquisitions: *Expected Improvement, Upper Confidence Bound*

Not usable here due to the complex learning task *(measure of progress?)*

- Nash: first-order stationarity
- Kalai-Smorodinsky: part of the Pareto front $+$ Nadir $+$ Shadow
- Copula-Kalai-Smorodinsky: marginals $+$ copula
What we will use: stepwise uncertainty reduction

What do we know about the equilibrium?

Computing equilibria on discrete sets is easy
The search for Nash, KS, or CKS can be done by (smart) exhaustive search on a grid.

What we can do
- Discretize $\mathbf{X}$ (grid, LHS...)
- Draw GP samples $\mathbf{Y}_1, \ldots, \mathbf{Y}_M$ of $\mathbf{Y}(\mathbf{X}_{\text{disc}})$
- Get (quickly) equilibrium of each $\mathbf{Y}_i$
What do we know about the equilibrium?

GP sample ⇒ cloud of potential solutions (+)

⇒ We want to choose $x_{new}$ so that the cloud shrinks!
Formalizing: Stepwise uncertainty reduction (SUR)

Random equilibrium

- \( y(x^*) = \Psi(y) \) (Nash, KS or CKS) equilibrium for cost function \( y \)
- \( Y \) random (GP) \( \Rightarrow \) \( \Psi(Y) \) random (\( \rightarrow \) the +’s)

Uncertainty measure of equilibrium \( \equiv \) “volume” of \( \Psi(Y) \):

\[
\Gamma(Y) = \det [\text{cov}(\Psi(Y))]
\]

\( \Rightarrow \Gamma_k \) if \( Y \) conditioned on observations \((x_1, y_1), \ldots, (x_k, y_k)\)

Greedy (stepwise) uncertainty reduction

- Optimal choice: \( x_{k+1} = \arg \min \Gamma_{k+1} \)
- Out of reach without \( y(x_{k+1}) \)!

Acquisition function: expected reduction

\[
J(x_{k+1}) = \mathbb{E}_{F_{k+1}} (\Gamma [Y|F_{k+1} = Y(x_{k+1})])
\]

with \( Y(x_{k+1}) \sim \mathcal{GP}(\mu(x_{k+1}), \Sigma(x_{k+1}, x_{k+1})) \)
Computing the acquisition function

Acquisition function

\[ J(x_{\text{new}}) = \mathbb{E}_{F_{\text{new}}} \left( \Gamma \left[ Y | F_{\text{new}} = Y(x_{\text{new}}) \right] \right) \]

with \( Y(x_{\text{new}}) \) Gaussian

Two-layer Monte Carlo

- \( Y_1^{(\text{new})}, \ldots, Y_K^{(\text{new})} \) observation drawings at \( x_{\text{new}} \)
- We condition the paths by these observations:
  \( Y_m | Y_k^{(\text{new})} \)
- \( \hat{J}(x_{\text{new}}) \approx \)

\[ \frac{1}{K} \sum_{k=1}^{K} \det \left[ \text{cov} \left( Y_1 | Y_k^{(\text{new})}, \ldots, Y_M | Y_k^{(\text{new})} \right) \right] \]
Illustration: iteration 1, 3 and 7 (KS search)

- With only 13 points: KS identified, very small residual uncertainty
- Trade-off between learning the center and the extremities of the Pareto front
In practice: Many numerical tricks are necessary...

- Efficient data structure and discrete games solver
- Use of parallel computation
- Fast algorithm for drawing and updating simulated GP paths
  

- Small discretized space by choosing only useful points
  (⇒ shorter paths, smaller games)

- Optimization of the acquisition function only over the discretized space
Outline

1. Introduction
2. Games and equilibria
3. Solving games with Bayesian Optimization
4. Application to model calibration
5. Conclusion
**li-BIM**: an agent-based model for designing buildings

Simulated behavior of occupants in a building

- Numerical modeling of the building: thermal, air quality, lighting, etc.
- Evolved occupational cognitive model (Belief-Desire-Intention)

---

F. Taillandier, A. Micolier, P. Taillandier (2017). Li-BIM (Version 1.0.0), *CoMSES Computational Model Library*
The li-BIM calibration problem

Many parameters to tune
- Behavioral: sensitivity to hot, cold, air quality, hunger, tidiness...
- Appliances characteristics: electrical power, efficiency...

Model should match real data (records or surveys)
- Electrical consumption
- Average temperature, air quality
- Time spent on various activities

A challenging optimization problem
- 11 parameters, 10 objectives
- Expensive, stochastic model (30min / simulation)

Outputs of different nature → how to define an acceptable compromise?
Experimental setup

**BO configuration**
- 100-point LHS + 100 infills
- Space discretization: 1000 points (renewed at each iteration)
- Choosing the next point $\approx 2$ min
- Objectives: square errors w.r.t. targets (averaged over 8 repetitions)
- KS + CKS (with empirical copula)

**Preliminary result**
Out of the 300 points, only 5 were dominated
![Results - objective space (% error)](image)

- Apparent trade-off between exploration and exploitation
- CKS more exploratory and sensitive to local mass on marginals
Results - objective space (2D)
Outline

1. Introduction
2. Games and equilibria
3. Solving games with Bayesian Optimization
4. Application to model calibration
5. Conclusion
Summary and future steps

Ingredients
- Game theory concepts to define compromise solutions
- GPs + Stepwise Uncertainty Reduction to find them
⇒ Parsimonious algorithm to tackle automatically many black-box objectives

SUR is very accommodating, so why not also...
- Generalized Nash problems (i.e. with constraints)
- KS with smart disagreement points: e.g. Nash-Kalai-Smorodinsky
- Batch-sequential strategies
- Smart use of repetitions (for stochastic simulators)
Want to give it a try? R package GPGame
https://CRAN.R-project.org/package=GPGame
Implements Nash, KS, CKS + Nash-KS (experimental)

Want to know more?

V. Picheny, M. Binois, A. Habbal
A Bayesian optimization approach to find Nash equilibria (2017+)
preprint: https://arxiv.org/abs/1611.02440

M. Binois, V. Picheny, A. Habbal
The Kalai-Smorodinsky solution for many-objective Bayesian optimization (2017+)