POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Comparative analysis of trust region filter methods with different mathematical models in surrogate-based optimization

Laurea Magistrale in Chemical Engineering - Ingegneria Chimica

Author: Hikmet Batuhan Oztemel

Advisor: Prof. Flavio Manenti

Co-advisor: Ing. Luis Felipe Sánchez

Academic year: 2023-2024

1. Introduction

In chemical engineering, large-scale complex optimization problems can be formulated and solved effectively with equation-oriented (EO) approaches, where the system equations are solved spontaneously. However, some complex process parts cannot always be formulated for process optimization using first principle models consisting of explicit analytical equations. Surrogate-based optimization (SBO) provides an efficient optimization strategy for those complex parts where mathematical models approximate computationally expensive truth models. The surrogate-based strategies were elevated with trust region filter (TRF) methods for constrained nonlinear optimization problems.
The TRF method defines a "trust region" with a radius and optimizes it within this region, where the surrogate model is considered accurate. In TRF, the feasibility of a surrogate model is assessed by examining its deviations from the truth or rigorous model. The filter mechanism was integrated into the trust region strategy to balance the tradeoff between objective improvement and the feasibility of a surrogate model. These trust regions are controlled with the radius, which is dynamically updated based on filter acceptance or rejection.
This study presents a comparative framework for TRF algorithms and different mathematical models as a surrogate model. The surrogate-based process optimization is performed on a methanol synthesis plant case study. A comparative framework is presented by discussing the nature of each model and the TRF method.

2. Surrogate Modeling

In the advanced computation world, evaluating expensive process information may be infeasible. A surrogate model is iteratively constructed to approximate an expensive function to search for optimal solutions. Surrogate model training is performed using the experimental data collected from physical experiments or numerical simulations. The data-driven surrogate models are then combined with an optimization method to explore a solution that is close to optimality—the key steps are summarized in Figure 1.
Several sampling methods, such as Latin Hypercube Sampling or Sobol Sequences, can generate initial samples for building the surrogate model as experimental design points.
This study evaluates various mathematical models to determine their suitability for trust re-

gion methods. The models compared include linear regression (`LR`), quadratic polynomial regressions (`2PR` and `2PRRIDGE` if Ridge regularization is applied), cubic polynomial regression (`3PR`), decision trees (`DT`), support vector machines (`SVM`), gradient boosting (`GB`), random forests (`RF`), Kriging, k-nearest neighbors (`KNN`), and artificial neural networks (`ANN`).

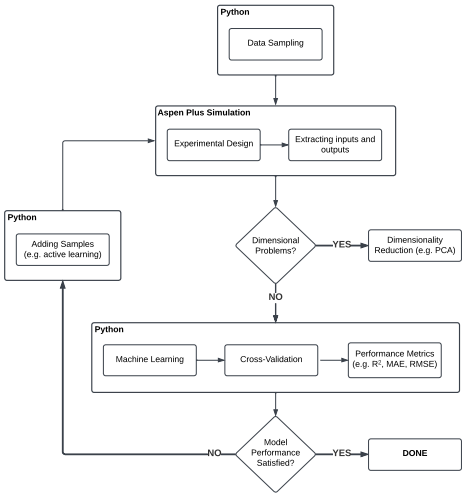


Figure 1: General diagram for the procedure of surrogate model training.

3. Trust Region Filters

Trust region methods emerged from classic Newton-based optimization methods. They evaluate the acceptability of a step toward the next iteration point. Figure 2 demonstrates a 2D plot of two decision variables x_1 and x_1 of the optimization problem.

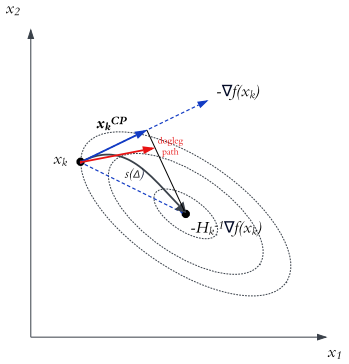


Figure 2: Trust region subproblem with gradient and Hessian-based steps.

The steepest descent direction $-\nabla f(x_k)$ represents the fastest local objective function decrease. The first-order derivatives may lead to inefficient convergence rates in ill-conditioned problems. Newton's direction $H_k \nabla f(x_k)$ uses the second-order Hessian information that is more accurate but computationally intensive. The trust region step $s_k(\Delta)$ is represented by the black curved arrow, which starts along the steepest descent direction, $-\nabla f(x_k)$, but bends to the Newton direction $H_k \nabla f(x_k)$. The transition depends on the trust region radius, Δ, since it directly limits or expands the step taken, $s_k(\Delta)$. The dogleg path indicates the combination of the steepest descent and Newton direction that is often used to express the transition between these two strategies.

The filtering concept is introduced as an alternative to traditional penalty-based functions, where two objectives compete for optimization search criteria: Objective function minimization and constraint violation reduction. The filter decides whether to accept or reject the step based on criteria of improved objective function or reduced infeasibility parameter.

3.1. Optimization Procedure

The reduced TRF (`RTRF`) framework was applied for a hybrid glass box/black box optimization problem, where some system components are explicitly defined by the first-order equations, and others are computationally expensive black-box functions without gradient information. The hybrid model optimization problem is reformulated as the trust region sub-problem (TRSP_k), as given in Equation (1). TRSP_k is solved at each iteration.

$$\begin{aligned} &\min_x f(x) \\ &\text{s.t.} \\ &h(x) = 0, \quad g(x) \leq 0 \\ &y = d(w) \\ &\|x - x_k\| \leq \Delta_k \end{aligned} \tag{1}$$

The rigorous model ($d(w)$) represents the actual model y, and it is approximated by the κ-fully linear surrogate model ($r(w)$). The κ-fully linear property ensures the surrogate model can produce reliable approximation and derivative information within a trust region. The infeasibility metric is denoted as θ and used to measure the level of surrogate model deviations from the rigorous model as $\theta = \|y - d(w)\|$.

The compatibility check, β, given as Equation (2), ensures that the sub-problem TRSP$_k$ is always feasible at $x = x_k$ with satisfied constraints. Its objective function, $\|y - r_k(w)\|$, searches for a feasible point x within the trust region compatibility constraint. At each iteration of compatibility, the new iterate updates as $x^*_{n,k} = x_k + n_k$ at which $n_k : \{n_{w,k}, n_{y,k}, n_{z,k}\}$. If the condition holds, then TRSP$_k$ passes the compatibility test.

$$\begin{gathered}\beta = \min \|y - r_k(w)\| \\ \text{s.t.} \\ h(x) = 0, \quad g(x) \leq 0 \\ \|x - x_k\| \leq \kappa_\Delta \Delta_k \min\left[1, \kappa_\mu \Delta_k^\mu\right]\end{gathered} \tag{2}$$

The criticality condition, $\chi(x)$, is computed using Equation (3) to assess the closeness of the current step to the optimality. It assesses whether if the proposed step size s_k is insufficiently small to proceed.

$$\begin{gathered}\chi(x) = \left|\min_v \nabla f(x)^T v\right| \\ \text{s.t.} \\ \nabla h(x)^T v = 0 \\ g(x) + \nabla g(x)^T v \leq 0 \\ v_y - \nabla r_k(w)^T v_w = 0 \\ \|v\| \leq 1 \quad \text{where } v^T : \{v_w^T, v_y^T, v_z^T\}\end{gathered} \tag{3}$$

Let us say $\chi_k : \chi(x + n_k)$. If the criticality condition $\chi_k < \xi \Delta_k$ holds, then the TRSP$_k$ is close to optimality relative to the Δ_k, where $\xi > 0$ is a fixed parameter. In this case, trust region radius shrinks by ω where $\Delta_{k+1} = \omega \Delta_k$ at $k + 1^{\text{th}}$ iteration and initialize TRSP$_k$ at $x^*_{n,k} = x_k + n_k$. If TRSP$_k$ finds a feasible solution, the proposed step is defined as $s_k = x^*_{s,k} - x_k$. After solving TRSP$_k$, the filtering mechanism decides whether to accept or reject the step based on the improvement of objectives (Equation 4). If the current step is successful, we assign $x_{k+1} = x_k + s_k$; otherwise, $x_{k+1} = x_k$. The filter is defined as $\mathcal{F}_k = \{(\theta_j, f_j) : j < k, j \in Z\}$.

$$\begin{gathered}\theta(x_k + s_k) \leq (1 - \gamma_\theta)\theta_k \\ \text{or} \quad f(x_k + s_k) \leq f(x_k) - \gamma_f \theta_k\end{gathered} \tag{4}$$

If the filter condition holds, the step s_k is acceptable to the filter set. However, the type of accepted step is determined before recording it to the filter. The switching condition is given in Equation (5) for the step type decision.

$$f(x_k) - f(x_k + s_k) \geq \kappa_\theta \theta(x_k)^{\gamma_s} \tag{5}$$

Where κ_θ and γ_s are tuning parameters, if Equation (5) holds, then the step is "f-type" step. Otherwise, the step is "θ-type", and the current solution is added to the filter set. Dynamic trust region radius update is based on the step quality ratio ρ_k (Equation (6)).

$$\rho_k = 1 - \frac{\theta(x_k + s_k)}{\theta(x_k)} \tag{6}$$

Trust region radius is updated as in Equation (7) using parameters $0 < \eta_1 \leq \eta_2$ and $0 < \gamma_c < 1 < \gamma_e$.

$$\Delta_{k+1} = \begin{cases} \gamma_c \Delta_k & \rho_k < \eta_1 \\ \Delta_k & \eta_1 \leq \rho_k \leq \eta_2 \\ \gamma_e \Delta_k & \rho_k \geq \eta_2 \end{cases} \tag{7}$$

If TRSP$_k$ is not compatible, that is, the compatibility check in Equation 12 is not convergence, a restoration procedure is applied to find a new point x_{k+1} with Δ_{k+1} and $r_{k+1}(w)$ to restore the algorithm, given that the new solution is compatible and acceptable by the filter. The reduced TRF converges as Δ_k approaches zero.

3.2. Modified Algorithm

The TRF framework is modified (`MTRF`) using additional strategies. Instead of trust region radius, locally accurate surrogate models are constructed within a closed ball $B(x_k, \sigma_k)$, where σ_k indicates the sampling region radius. The modified algorithm integrates σ_k for handling constraint violations, while the trust region radius Δ_k is responsible for objective function minimization. In the modified TRF, the criticality check lowers the sampling region. Therefore, the criticality check is changed from $\chi_k < \xi \Delta_k$ to $\chi_k < \xi \sigma_k$. If the new criticality condition holds, the sampling region is updated as in Equation (8), at which the criticality condition no longer holds [2].

$$\sigma_k = \max\left(\min\left(\sigma_{k-1}, \frac{\chi_k}{\xi}\right), \Delta_{\min}\right) \tag{8}$$

The filter mechanism is modified as demonstrated in Equation (9) for all $(f_j, \theta_j) \in \mathcal{F}_k$. An

additional requirement is combined with the filtering mechanism to prioritize reducing infeasibilities through iterations, that is $\theta(x_k) \leq \theta_{\min}$.

$$\begin{aligned} &\theta(x_k + s_k) \leq (1 - \gamma_\theta)\theta_j \\ \text{or} \quad &f(x_k + s_k) \leq f_j - \gamma_f \theta_j \end{aligned} \tag{9}$$

For f-type steps, the trust region update is modified to $\Delta_{k+1} = \max(\gamma_e \|s_k\|, \Delta_k)$, where $\gamma_e \geq 1$ and $\|s_k\|$ is the norm of the proposed step. The trust region update modification is based on ρ_k is given by Equation (10), where ζ_k represents $\max(\gamma_e \|s_k\|, \Delta_k)$

$$\Delta_{k+1} = \begin{cases} \gamma_c \|s_k\| & \rho_k < \eta_1 \\ \Delta_k & \eta_1 \leq \rho_k < \eta_2 \\ \zeta_k, & \rho_k \geq \eta_2 \end{cases} \tag{10}$$

The step quality ratio ρ_k is modified as Equation (11), where ϵ_θ is an infeasibility tolerance. The modified TRF algorithm converges as $\sigma_k \to 0$.

$$\rho_k = \frac{\theta(x_k) - \theta(x_k + s_k) + \epsilon_\theta}{\max\left(\|y_k - r_k(w_k)\|, \epsilon_\theta\right)} \tag{11}$$

3.3. Demand-Based Algorithm

The TRF framework was refined for demand-based optimization, where the production level is dynamically adjusted based on market demand fluctuations. The demand-based TRF (`DMTRF`) avoided incorrect demand estimations in the chlorobenzene process. Their application also showed the scalability of TRF methods for diverse applications, particularly in large-scale industrial processes under uncertain market conditions. The authors used a low-fidelity CSTR-based model to replace high-fidelity models.
The step quality ratio ρ_k was upgraded by introducing an additional condition. Current infeasibility θ_k is compared with the minimum infeasibility value θ_{min}. The switching condition in Equation (5) is also employed in this algorithm. Unlike previously defined algorithms, the demand-based algorithm does not include a compatibility check or a criticality condition. The demand-based TRF algorithm was defined as it converges as $s_k \to 0$.

4. Case Study

Methanol (CH_3OH) is an essential component of the global economy and is one of the world's most widely operated chemical compounds. Among its production processes, the ICI Synetix process is the major technology for methanol production worldwide, accounting for the largest share of global methanol output with 61% [1].
We conducted a case study for the TRF application with an ICI Synetix plant in Aspen Plus process simulation. The process flow diagram is given in Figure 3. The methanol converter is an adiabatic, four-staged, fixed-bed quench reactor. Exothermic methanol synthesis reactions rapidly increase the reactor temperature. Therefore, cold syngas are injected into each fixed catalyst bed to maintain the reactor temperature. The reactor model accounted for more than 65% of the overall simulation time. Thus, the reactor model is approximated with data-driven surrogate models using the reactor's inlet and outlet stream informationl.

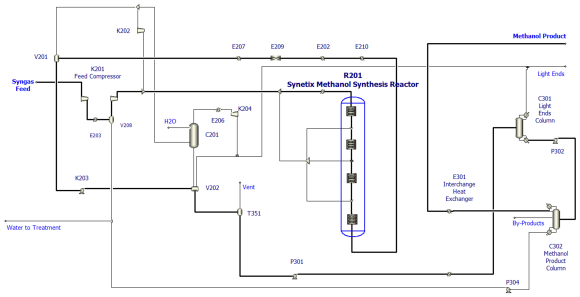


Figure 3: Process flow diagram of the ICI Synetix methanol plant.

The objective of the case study is to maximize the plant's profit by accounting for several aspects. For instance, the produced methanol and light gases are the profit components of the objective function. On the other hand, the objective function also has cost components due to syngas raw materials, undesired by-products, operational expenses, and carbon tax. The operational limits of the reactor are respected for the experimental design and surrogate model training. Otherwise, the catalyst sintering reduces the activity of the $Cu/Zn/Al_2O_3$ catalyst, providing less efficient operations.

5. Results and Discussion

The experimental design was conducted using Aspen Plus's built-in tool. The surrogate models were trained with 5-fold cross-validation for

TRF optimization. The ICI Synetix reactor model's decision variables are the operational conditions and the mass flow rates of reacting species entering and leaving the reactor. The Component Object Model (COM) was created in Python to facilitate data communication between the Aspen Plus process simulation and the Python environment. The TRF methodologies guided the simulation calls during optimization iterations.

The root-mean-squared-errors (RMSE) distribution across decision variables was illustrated with the box-and-whisker plot in Figure 4 for surrogate model candidates. The figure indicates that polynomial and Kriging models demonstrated smaller boxes and shorter whiskers, indicating lower variability across the surrogate decision variables. They performed the best results with the smallest RMSE median values. Although promising outcomes emerged via gradient boosting and multilayer perceptron models, they have higher computing costs than regression or polynomial models, making them inefficient for TRF comparisons.

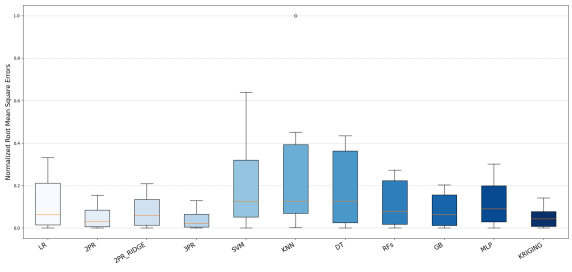


Figure 4: Normalized root-mean-square errors (RMSE) across each model.

Because the trust region filter approximation is a two-objective optimization problem, it is essential to account for the feasibility metrics for all TRF and surrogate model configurations. Figure 5 presents histograms and kernel density estimates (KDEs) of the normalized infeasibilities for different surrogate model configurations. Histograms represent the frequencies of normalized infeasibilities at each iteration, whereas the KDE plot approximates infeasibility densities, the totnumberers of optimization iterations. For instance, the peak in `RTRF_2PRRIDGE` configuration indicates a higher number of iterations than other configurations, resulting in increased computational expenses required to achieve convergence.

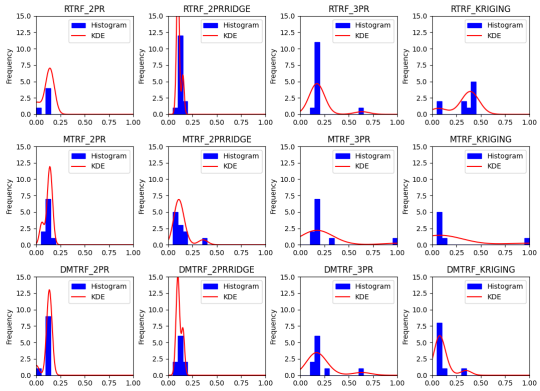


Figure 5: Distribution histograms and KDEs of infeasibility metrics across configurations.

The KDE plot for `RTRF_2PR` cumulates near zero values because the quadratic model in the reduced framework achieved sufficient surrogate model feasibility across iterations. On the other hand, the Kriging-based model performed relatively higher infeasibility with the reduced TRF algorithm (`RTRF_KRIGING`). However, The Kriging-based surrogate model yielded significantly better model feasibilities in the modified and demand-based algorithms.

5.1. Optimization Performances

The convergence profiles for f_k, θ_k, Δ_k, σ_k, and s_k trends were analyzed through iterations. The reduced algorithm with the quadratic model (`RTRF_2PR`) managed to minimize the objective function earlier in the iteration.

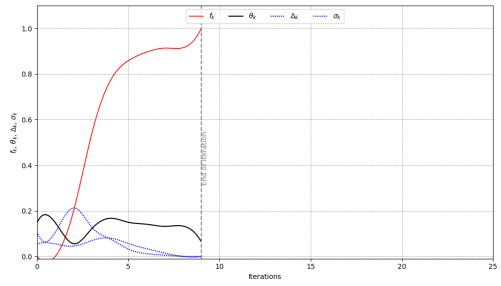
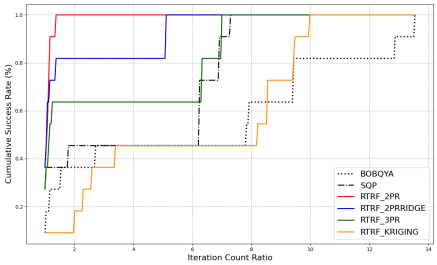


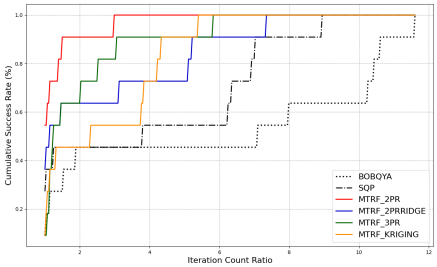
Figure 6: Convergence profiles of the modified TRF with a quadratic surrogate (`MTRF_2PR`).

However, `MTRF_2PR` gradually maximized the profit optimality and achieved the minimum across all optimizers (Figure 6). The modified TRF structure terminated the optimization efficiently by preventing unnecessary steps from

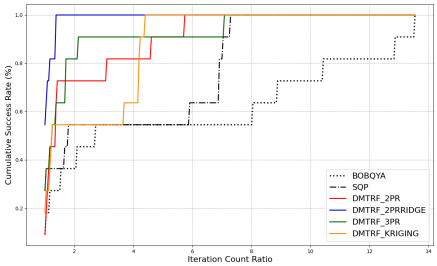
being taken. The modified TRF proved its advantage in handling constraint violations of surrogate models, yielding stabler model feasibility. Figure 7 illustrates the performance profiles for all TRF configurations, interpreting how efficient the algorithms were to achieve the minimum objectives.



(a) Reduced trust region filter configurations.



(b) Modified trust region filter configurations.



(c) Demand-based trust region filter configurations.

Figure 7: Comparison of optimality performance profiles.

For instance, `RTRF_KRIGING` was not a robust option for exploring better optimality. Although `2PRRIDGE` surrogates struggled to explore in latter iterations, `2PR` and `2PRRIDGE` models outperformed other solvers in most cases. Another key point is that the demand-based algorithms generally perform more efficiently than other configurations, making them an attractive solver for large-scale applications.

6. Conclusions

Polynomial models demonstrated significant performance compared to other surrogate models, indicating their adaptability for complex problems in surrogate-assisted TRF optimization. The quadratic models showed high success rates with steeper curves than others. However, Kriging models demonstrated the lowest performance compared to polynomial models. This resulted in 36.5% and 46.2% less quality solutions for normalized optimalities compared to the quadratic models. The quadratic Ridge models performed 6.2% less efficiently than the quadratic models. Across the TRF methods, the modified algorithm found better optimalities by around 5.4%. Regarding global comparisons, the TRF methods outperformed the SQP and BOBQYA algorithms. Different TRF algorithms had no significant differences between each other for infeasibility metrics. However, the quadratic-based models performed surrogate feasibility better than the cubic and Kriging models. The Kriging surrogate models outperformed the cubic surrogates by 3.2%.
Overall, the modified TRF algorithm exhibited stabler convergence relative to the reduced TRF. Using a sampling region radius with a trust region radius resulted in accelerated performance for the objectives of optimality and feasibility. The demand-based TRF architecture improved scalability, attaining strong surrogate feasibility and comparatively more rapid convergence rates across various surrogate models. Among the evaluated methodologies, demand-based TRF demonstrated the most reliable performance in terms of optimality and feasibility.

References

[1] Giulia Bozzano and Flavio Manenti. Efficient methanol synthesis: Perspectives, technologies and optimization strategies. *Prog. Energy Combust. Sci.*, 56:71–105, 2016.

[2] John P Eason and Lorenz T Biegler. Advanced trust region optimization strategies for glass box/black box models. *AIChE Journal*, 64(11):3934–3943, 2018.