

Simulation in Produktion und Logistik 2023
Bergmann, Feldkamp, Souren und Straßburger (Hrsg.)
Universitätsverlag Ilmenau, Ilmenau 2023
DOI (Tagungsband): 10.22032/dbt.57476

Optimierung einer stochastischen MRP-Simulation unter Anwendung der Bayes'schen Optimierung

Optimization of a Stochastic MRP Simulation Using Bayesian Optimization

Philipp Zmijewski, Nicolas Meseth, Hochschule Osnabrück, Osnabrück (Germany),
philipp.zmijewski@hs-osnabrueck.de

Abstract: This paper explores whether the Bayesian optimization algorithms GPEI, TurBO and SAASBO are effective for stochastic material requirements planning simulations. It includes a comparison of other methods, with a focus on the convergence speed, a key factor in simulation-based optimization. The study uses a simple material requirement planning simulation model that is progressively expanded in complexity by adding products and levels to the bill of materials. This results in a high-dimensional optimization problem, which poses a significant challenge for simulation-based optimization. The Bayesian optimization methods are compared at each level of complexity to determine if they produce satisfactory results. Additionally, the convergence speed is analyzed in relation to method and complexity. A genetic algorithm, CMA-ES, and Sobol serve as benchmarks for the Bayesian optimization methods.

1 Problemstellung und Motivation

Die Materialbedarfsplanung ist eine gängige Methode in der Produktions- und Bestandsplanung. Die Methode beruht auf gegebenen Bedarfen für Endprodukte. Durch die Stücklistenauflösung werden zu bestellende oder zu produzierende Sekundärbedarfe für jede Periode ermittelt. In der Praxis kann es vorkommen, dass diese Bedarfe durch etwa verspätete Lieferungen oder mangelnde Qualität bei der Anlieferung nicht vollständig gedeckt werden. Stochastische Simulationen können diese Unsicherheit adressieren und bei der Planung berücksichtigen. Innerhalb der Simulation werden dazu Parameter wie Sicherheitsbestand oder Bestellvorlaufzeit für jedes Produkt festgelegt, um etwaige Ausfälle oder Verzögerungen zu kompensieren. Die Ermittlung der optimalen Parametrisierungen für jedes Produkt ist eine anspruchsvolle Aufgabe und für menschliche Planer nicht leistbar. Das gilt umso mehr, wenn viele Produkte geplant werden müssen, da mit jedem zusätzlichen Produkt die Anzahl der festzulegenden Parameter steigt. Um den Planer zu

unterstützen, können Simulationen in Kombination mit Optimierungsalgorithmen eingesetzt werden. Es wird dabei von der simulationsbasierten Optimierung gesprochen. Gegenwärtig werden in diesem Kontext vorwiegend evolutionäre oder genetische Algorithmen verwendet (Amaran et al., 2016; Damand et al., 2022; Nguyen et al., 2014; Werth et al., 2023). Diese eignen sich gut zur Lösung unterschiedlicher Problemstellungen, weisen aber, verglichen mit Verfahren der Bayes'schen Optimierung (BO), eine geringere Konvergenzgeschwindigkeit auf (Bull, 2011). Die Anzahl der benötigten Evaluationen (hier: Simulationsläufe) zur Erreichung eines befriedigenden Ergebnisses ist bei diesen Verfahren deshalb verhältnismäßig hoch (Bull, 2011). Insbesondere bei rechenintensiven Simulationen mit langer Laufzeit kann das problematisch sein.

Es gibt einige Publikationen, die sich mit der Optimierung von MRP-Simulationen befassen. Im Folgenden werden Auszüge genannt, die auf die Optimierung mit sequentiellen Algorithmen abzielen. Die Autoren Werth et al. (2023) und Damand et al. (2022) behandeln den Mehrzielfall mit jeweils unterschiedlichen Zielgrößen und verwenden dabei den NSGA-II Algorithmus, einen evolutionären Algorithmus für hochdimensionale Mehrzielprobleme. Beide erzielen damit zufriedenstellende Ergebnisse. Damand et al. (2022) fokussiert dabei speziell das Demand-Driven Material Requirements Planning, eine Weiterentwicklung des klassischen MRP. Die zu optimierenden Parameter variieren je nach Publikation, oft werden jedoch der Sicherheitsbestand, die Bestellvorlaufzeit sowie die Losgröße optimiert (Damand et al., 2022; Gansterer et al., 2014; Werth et al., 2023). Die vorliegende Studie optimiert die Parameter Sicherheitsbestand und Bestellvorlaufzeit pro Material mit dem Ziel der Kostenminimierung. Zur Anwendung von BO-Algorithmen auf diese Art von MRP-Simulation ist den Autoren keine Publikation bekannt.

2 Methodik

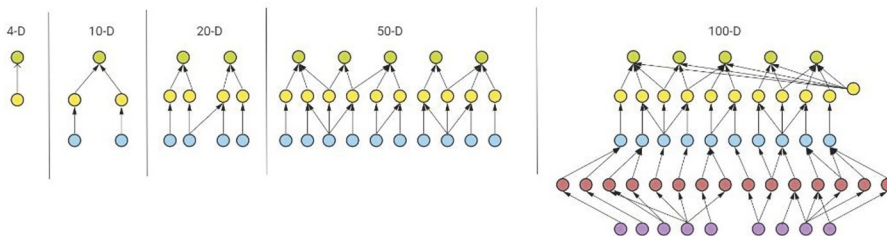


Abbildung 1: Gozinto-Graph der geplanten Stücklisten der Experimente. Aufsteigend sortiert nach der Anzahl der Dimensionen (D). Pro Knoten gibt es zwei Parameter

Um den ersten Teil der Forschungsfrage nach der Eignung der Bayes'schen Optimierung für die Anwendung auf stochastische Materialbedarfsplanungssimulationen zu beantworten, wird die Komplexität der Simulationsmodelle schrittweise erhöht und geprüft, wie die Algorithmen abschneiden. Dies geschieht zum einen durch die Hinzunahme von Produkten und zum anderen durch die Vertiefung der Stücklisten um zusätzliche Ebenen (Abbildung 1). Die Ergebnisse der Algorithmen werden auf jeder Komplexitätsstufe relativ

zueinander verglichen, da ein absoluter Vergleich hinsichtlich der Kosten aufgrund des unbekanntem Optimums unmöglich ist.

Für den zweiten Teil der Forschungsfrage hinsichtlich einer möglichen verbesserten Konvergenzgeschwindigkeit sollen ausgewählte Algorithmen aus der Familie der Bayes'schen Optimierungsverfahren mit dem CMA-ES-Algorithmus (Hansen, 2016) als einem gegenwärtig etablierten Verfahren verglichen werden. Zusätzlich wird die Sobol-Sequenz (Sobol', 1967) als Random-Search-Verfahren in den Vergleich einbezogen. Insgesamt werden für die Experimente in diesem Vorhaben folgende Verfahren betrachtet: GPEI (Frazier, 2018), SAASBO (Eriksson and Jankowiak, 2021), TuRBO (Eriksson et al., 2019), CMA-ES und Sobol'.

2.1 Bayes'sche Optimierung

Die Bayes'sche Optimierung (BO) ermöglicht eine stichprobeneffiziente Optimierung stochastischer und kostspieliger (hier: zeitaufwändiger und rechenintensiver) Black-Box-Funktionen. Dabei geht BO sequentiell vor, um neue vielversprechende Parametrisierungen basierend auf dem aktuellen Informationsstand vorzuschlagen. Die Parametrisierungen werden durch das Simulationsmodell evaluiert und samt Ergebnis dem Informationsstand hinzugefügt. Um das Verfahren zu beschleunigen, können pro Iteration mehrere Parametrisierungen vorgeschlagen werden. Die Optimierung ist beendet, wenn das zuvor definierte Evaluationsbudget erschöpft ist oder zuvor ein befriedigendes Ergebnis gefunden wurde. Es handelt sich bei BO um eine stochastische, heuristische Optimierungsmethode (Balandat et al., 2020; Frazier, 2018).

BO gilt bei einigen Black-Box-Problemstellungen als Stand der Technik, wie etwa dem Hyperparametertuning von Algorithmen des Maschinellen Lernens (u.a. Shahriari et al., 2016). Black-Box-Probleme sind auch in anderen Disziplinen, wie dem Werkstoffdesign (Zhang et al., 2020) und der Robotik (Berkenkamp et al., 2020) bekannt und können dort ebenfalls mit BO adressiert werden. Simulationen können ebenso als eine Black-Box betrachtet werden und weisen zusätzliche Eigenschaften auf, die Fragen für die Optimierung aufwerfen. So verfügen Simulationsmodelle oft über gemischte Parametertypen (kontinuierlich und diskret), Mehrzielproblematiken oder eine hohe Dimensionalität. Diese Charakteristika können durch diverse Algorithmen aus der Familie der Bayes'schen Optimierung adressiert werden (Daulton et al., 2021; Eriksson et al., 2019; Zhang et al., 2020). Die vorliegende Studie untersucht die Anwendung ausgewählter BO-Algorithmen auf stochastische, hochdimensionale MRP-Simulationen mit ganzzahligen Parametern.

2.2 Parametrisierung der Algorithmen und Experimentdesign

Das Verhalten und damit das Optimierungsergebnis jedes der genannten Verfahren kann über die Festlegung verschiedener Hyperparameter beeinflusst werden. Die systematische Veränderung der Hyperparameter ist Teil des Experimentierplans und der Einfluss der Hyperparameter auf den besten gefundenen Wert sowie die Konvergenzgeschwindigkeit wird auf jeder Komplexitätsstufe untersucht. Aufgrund der kombinatorischen Zunahme an benötigten Experimenten wurde nur ein Hyperparameter, das Evaluationsbudget, angepasst. Ein zweiter BO-Hyperparameter, die Anzahl an zufälligen Initialisierungspunkten, ist mit 20 % des Budgets festgelegt. Für das Evaluationsbudget wurden 6 Abstufungen gewählt: 40, 100, 200, 300, 500 und 1000 Evaluationen. Die Budgets wurden festgesetzt, um Limitationen, wie sie in

der Praxis gegeben sein können, abzubilden. Der Initialisierungsanteil wurde fixiert, um diesen über alle Budgets proportional zu halten. Alternativ hätte er in Abhängigkeit der Dimensionalität definiert werden können. Für die Hyperparameter der Algorithmen wurden Standardwerte aus der oben genannten Literatur verwendet. Jedes Experiment stellt folglich eine Kombination aus dem Modell (Komplexitätsstufen), dem Optimierungsverfahren, und dem Evaluationsbudget dar. Aufgrund der Stochastik sowohl im Simulationsmodell als auch in den Optimierungsverfahren (u.a. zufällige Parametrisierungen für die Initialisierung) müssen die Experimente mehrfach durchgeführt werden. Es wurden pro Experiment mindestens 15 Replikationen durchgeführt. Aufgrund spezifischer Verhaltensweisen der Algorithmen kann das Evaluationsbudget punktuell geringfügig überschritten werden. Es ist anzumerken, dass die Algorithmen das verfügbare Evaluationsbudget nicht kennen und sich unabhängig davon verhalten. So ist es nicht der Fall, dass ein Algorithmus etwa mehr exploriert, wenn ein höheres Budget zur Verfügung steht. Die Bayes'schen Algorithmen arbeiten sequenziell. Pro Sequenz (im Verlauf Trial genannt) wurden $n=1$ Parametrisierungen generiert und evaluiert (im Verlauf Arm genannt). Bei CMA-ES gilt $n=2$, da $n=1$ technisch nicht möglich ist. Es ist bekannt, dass CMA-ES mit zunehmendem n einen höheren Explorationsgrad erreicht. Da vorab keine sinnvolle Annahme aufgestellt werden konnte, wurde der Wert, wie bei den anderen Algorithmen, auf das Minimum gesetzt. Sobol generiert alle Parametrisierungen auf einmal, erst dann werden diese evaluiert.

2.3 MRP-Simulation

Die MRP-Simulation besteht aus zwei Komponenten: dem Solver und der Simulation. Der Solver ist eine Python-Implementierung des klassischen MRP-Algorithmus, einschließlich der Stücklistenauflösung und der Terminierung (Stadtler et al., 2015). Die Eingabewerte für den Solver sind die Stückliste, Bestellmengen pro Periode, Lieferzeiten und Parameter für jedes Produkt. Die Ausgabe des Solvers sind die terminierten Bedarfe für alle Materialien, die als Eingabegrößen für die Simulation dienen. Bis hierhin ist der Prozess deterministisch.

Die Simulation berechnet u.a. die Lagerkosten für die terminierten Bedarfe sowie die Lieferbereitschaft (Service Level) der zugrundeliegenden Bestellungen. Sie wird durch verspätete Anlieferungen und Mindermengen stochastisch beeinflusst. Beiden stochastischen Einflüssen liegen diskrete Wahrscheinlichkeitsverteilungen zugrunde. Da die verwendeten Algorithmen nur eine Zielgröße optimieren können, wurden Strafzahlungen für nicht vorrätige Mengen implementiert. Diese werden pro Stück und Periode erhoben und ermöglichen somit eine Monetarisierung der Lieferbereitschaft. Der Ausgabewert der Simulation sind die Kosten, die sich aus den Lagerkosten und Strafzahlungen zusammensetzen, sowie das berechnete Service-Level. Letzteres wird zur Plausibilisierung herangezogen und geht nicht in die Optimierung ein.

3 Ergebnisse

Tabelle 1: Beste Kostenwerte, Median und Anzahl der Experimentwiederholungen pro BOM für alle Algorithmen und Budgets

BOM	Kennzahl	Sobol	GPEI	TurBO	CMA-ES	SAASBO
4	Min.	17.032	12.984	14.392	13.381	13.960
	Median	25.164	21.367	18.085	21.299	16.903
	n	190	167	166	171	40
10	Min.	37.933	38.203	34.621	35.829	36.013
	Median	47.197	45.836	43.616	47.418	41.250
	n	255	294	234	150	10
20	Min.	44.554	38.906	36.627	36.440	34.799
	Median	53.178	52.376	47.468	49.154	40.393
	n	150	110	110	130	17
50	Min.	190.086	193.818	166.340	184.991	-
	Median	239.926	233.201	239.658	277.536	-
	n	115	115	130	120	0
100	Min	378.105	300.594	238.459	284.882	-
	Median	1.257.125	610.017	572.873	751.689	-
	n	131	114	115	90	0

Tabelle 1 zeigt das absolute gefundene Kostenminimum über alle Experimentwiederholungen, den Median der besten gefundenen Kosten jeder Experimentwiederholung, sowie die Anzahl der Experimentwiederholungen für jede BOM und alle untersuchten Algorithmen. Bei den drei kleinsten Probleminstanzen (4, 10 und 20 Dimensionen) finden SAASBO und TurBO im Median die niedrigsten Kosten. Bei 50 Dimensionen findet TurBO zwar den niedrigsten Wert, ist im Median aber nur marginal besser als Sobol. GPEI findet hier die besten Werte. CMA-ES ist leicht abgeschlagen. Bei 100 Dimensionen dominiert TurBO beim besten Wert und im Median. CMA-ES findet zwar den zweitbesten Wert, ist im Median aber höher als TurBO und GPEI. Die Ergebnisse von Sobol sind signifikant schlechter.

Die unterschiedliche Anzahl an Replikationen ist darauf zurückzuführen, dass anfangs für jedes Experiment 15 Replikationen als Mindestanzahl durchgeführt wurden, mit Ausnahme von SAASBO. Auf die Gründe wird in der Diskussion eingegangen. Anschließend wurden bedarfsweise weitere Replikationen veranlasst. Hierbei wurde keiner Systematik gefolgt.

3.1 Verteilung der Kosten über alle Replikationen (BOM 20)

Abbildung 2 zeigt die aggregierten Ergebnisse aller Algorithmen für die 20-Dimensionen Probleminstanz (BOM 20). Diese wurde ausgewählt, da sich die Aussagen mit denen der anderen Instanzen weitestgehend decken und so Redundanzen vermieden werden. 20 Dimensionen gelten zudem als weiche Obergrenze des GPEI-Algorithmus (Frazier, 2018). Die entsprechenden Analysen für die anderen Komplexitätsstufen können auf der Webseite des Beitrags mit ergänzendem Material eingesehen werden.

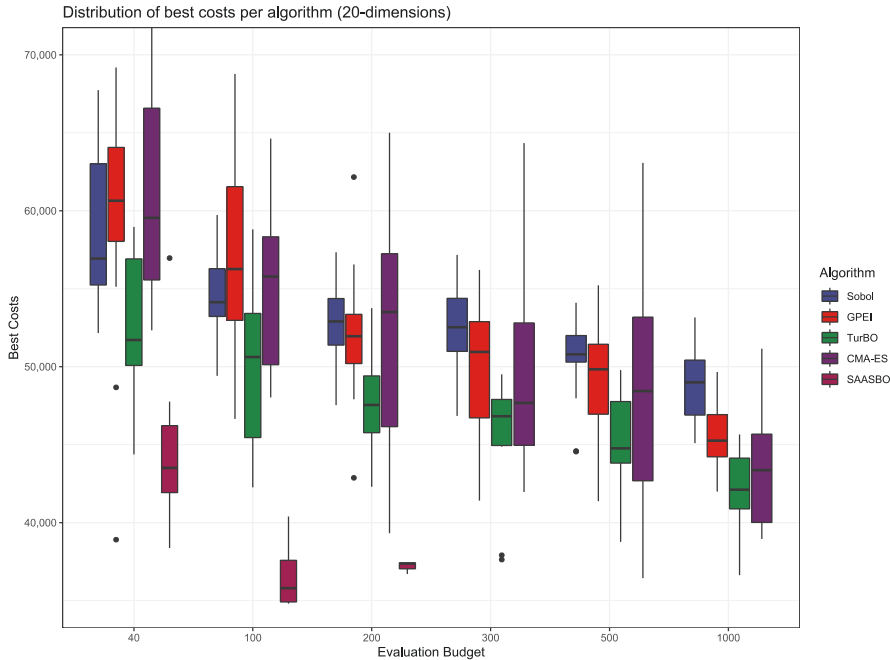


Abbildung 2: Box-Plots der kleinsten gefundenen Kosten über n Replikationen der BOM 20 pro Algorithmus und Budget mit den 25%- und 75%-Quantilen. Pro Algorithmus und Budget gilt $n=15$; Ausnahme SAASBO, dort gilt für Budget 40: $n=10$, 100: $n=2$, 200: $n=3$, Rest $n=0$

Auf der y-Achse sind die kleinsten gefundenen Kosten abgetragen. Auf der x-Achse die Budgetgrößen. SAASBO hat, wo vorhanden, im Median die Parametrisierungen für die niedrigsten Kosten identifiziert. Dahinter folgt TurBO. GPEI und CMA-ES sind bei kleineren Budgets hinsichtlich des Medians nicht oder nur geringfügig besser als die Sobol-Parametrisierungen. Bei höherem Budget erreichen sie bessere Ergebnisse. CMA-ES weist eine hohe Streuung innerhalb der Replikationen auf, wobei diese erwartungsgemäß mit höherem Budget sinkt. Dies kann unter Umständen durch die Stagnation in lokalen Minima erklärt werden, was aus dem zuvor erwähnten, geringen Explorationsfaktor resultieren kann.

3.2 Konvergenz über alle Budgets und Algorithmen

Abbildung 3 vergleicht die geglätteten Konvergenzkurven über alle Experimentwiederholungen für jeden Algorithmus. Die Konvergenzkurve zeigt den jeweils niedrigsten Kostenwert zum Stand des jeweiligen Arms. Die gestrichelte vertikale Linie kennzeichnet das Sobol-Initialisierungsbudget der BO-Algorithmen. Für SAASBO liegen nur Daten der Budgets 40, 100 und 200 vor. In diesen Fällen dominiert der SAASBO-Algorithmus deutlich. Ansonsten erzielt TurBO die besten Konvergenzen. Nennenswert ist, dass Sobol ähnliche Konvergenzen wie GPEI und CMA-ES aufweist.

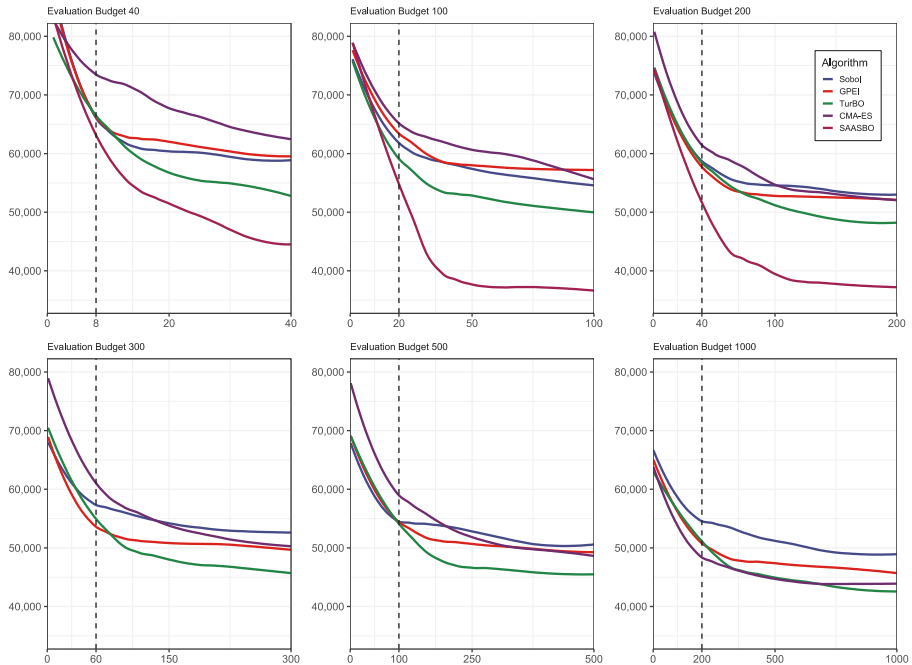


Abbildung 3: Geglättete Konvergenzen der Algorithmen für BOM 20 über alle Replikationen. Die gestrichelte Linie markiert die Initialisierung der BO-Algorithmen. Pro Algorithmus und Budget $n=15$; Ausnahme SAASBO, dort Budget 40: $n=10$, 100: $n=2$, 200: $n=3$, Rest $n=0$

4 Diskussion

4.1 Ergebnisse

In diesem Kapitel sollen die Ergebnisse aus Kapitel 4 eingeordnet werden. Es konnte gezeigt werden, dass TurBO und SAASBO die anderen Algorithmen hinsichtlich Ergebnis und Konvergenz dominieren. Dieser Effekt schwindet mit zunehmendem Budget.

Alle untersuchten Algorithmen finden bei zunehmendem Budget zuverlässig bessere Werte. Teilweise schneiden die Optimierungsalgorithmen GPEI und CMA-ES nicht wesentlich besser ab als Sobol. Dies ist unter anderem dadurch zu erklären, dass GPEI ab 20 Dimensionen Verluste in der Modellierungsgüte ausweist (Frazier, 2018), CMA-ES ein hohes Budget erfordert (Hansen, 2016) und Sobol bei diskreten, ganzzahligen Parameterräumen, verglichen mit kontinuierlichen, tendenziell bessere Ergebnisse erzielt. Die auffällig guten Ergebnisse von SAASBO gilt es vor dem Hintergrund des höheren Rechenaufwandes (Zeit und Ressourcen) zu betrachten. SAASBO benötigte exemplarisch bei der BOM 20 und dem 200. Arm auf hochmoderner Hardware über 10 Minuten für einen Trial. GPEI und TurBO lagen bei wenigen Sekunden und CMA-ES und Sobol im Millisekundenbereich. Diese Beobachtung deckt sich mit der Literatur (Eriksson and Jankowiak, 2021).

TurBO nähert sich mit größerem Budget den Werten von SAASBO an. Dies deutet

darauf hin, dass die Leistungsvorteile von SAASBO erst bei sehr aufwändigen Simulationen, mit langen Laufzeiten, zum Tragen kommen. Davor lassen sich vermutlich mit größerem Budget unter Anwendung von TurBO gleichwertige Ergebnisse erzielen.

CMA-ES kann bei geringerer Hardwareausstattung, einem angemessenen Explorationsfaktor und geringen Simulationslaufzeiten ebenfalls eine geeignete Option sein. Bei größeren Budgets weist es eine ähnliche Konvergenz zu TurBO auf und ist dabei ressourcenschonender. Die beträchtliche Varianz der Ergebnisse für CMA-ES kann nur mithilfe zusätzlicher Experimente analysiert werden. So wäre es interessant festzustellen, ob der Algorithmus selbst dafür verantwortlich ist, oder ob dies auf den erwähnten Explorationsfaktor zurückzuführen ist.

4.2 Methodik

Die MRP-Simulation wurde in einer stark vereinfachten Form modelliert. Sie dient im Rahmen dieser Untersuchung als angemessene Problemstellung für hierarchisch verknüpfte Eingabeparameter und weist mit den Stücklisten eine einfache Möglichkeit zur schrittweisen Erweiterung des Parameterraums auf. So konnte das Verhalten der Algorithmen bei unterschiedlicher Dimensionalität untersucht werden. Die Simulationszeiten sind mit wenigen Sekunden gering, verglichen mit in der Praxis üblichen Laufzeiten von bis zu mehreren Minuten oder Stunden. So konnten vielzählige Experimente in der begrenzten Zeit realisiert werden. Dennoch liegt in dieser Studie der Fokus auf der Konvergenz der Algorithmen, die vordergründig bei zeitaufwendigen Simulationen ein wichtiges Gütekriterium darstellt.

Die Auswahl der BO-Algorithmen repräsentiert nur einen kleinen Teil der existierenden Varianten der Bayes'schen Optimierungsverfahren. Die Selektion erfolgte anhand der in der jeweiligen Veröffentlichung genannten Eigenschaften des Algorithmus im Kontext der Problemstellung (hochdimensional, eine Zielgröße, stochastisch). GPEI wurde gewählt, da es als klassische Form der Bayes'schen Optimierung angesehen wird. BO-Algorithmen, die mit Dimensionseinbettung arbeiten, wie REMBO (Wang et al., 2016) oder ALEBO (Letham et al., 2020), wurden nicht ausgewählt. Diese Gruppe wies bereits in einer vorgelagerten Masterarbeit unzureichende Ergebnisse für die gleiche Problemstellung auf. Während des Experimentierzeitraums traten unerwartete Probleme mit dem SAASBO-Algorithmus auf. Zum einen kam es bei Experimenten mit großer BOM (≥ 50) und hohem Budget (>200) zu technischen Fehlern beim Fitting des Modells, die auf die GPU-Arbeitsspeichernutzung des in BoTorch implementierten Algorithmus zurückzuführen sind. Der Fehler konnte in Anbetracht der Zeit nicht behoben werden. Zum anderen wurden die Zeiten des Modellfittings unterschätzt, was die Anzahl der ausführbaren Replikationen reduzierte. Die Daten werden soweit möglich nachgereicht und sind über die weiter unten verlinkte Website einsehbar.

4.3 Weiterer Forschungsbedarf

Der vorliegende Beitrag kann als Auslotung der Möglichkeiten zur sequentiellen Optimierung von MRP-Simulationen betrachtet werden. Die Ergebnisse geben Grund zur Annahme, dass eine Anwendung von BO auf die Problemstellung und deren unterschiedliche Instanzen sinnvoll ist. Daraus ergeben sich weitere, breit gestreute Forschungsbedarfe. Basierend auf den generierten Ergebnissen und der praxisnäheren Mehrzielmodellierung der MRP-Simulation, wie sie auch von anderen Autoren

adressiert wird, ergibt sich für die Autoren eine hohe Priorität von Multi-Objective-SAASBO und Multi-Objective-TurBO. Zu ersterem gibt es einen Ansatz, welcher SAASBO mit der qNEHVI-Acquisition Function (Daulton et al., 2021) erweitert (ax.dev/tutorials/saasbo_nehvi). Dadurch wird es möglich, statt nur eines Punkts eine Pareto-Front der Zielgrößen zu generieren. Zu zweitem wurde ein Algorithmus, MORBO (Daulton et al., 2022), entwickelt. Dieser greift den Trust-Region Ansatz von TurBO auf und erweitert diesen um eine Multi-Objective-Acquisition-Function (HVI). Nachgelagert kann basierend auf diesen Ergebnissen ein Benchmark mit menschlichen Entscheidern erfolgen und theorisiert werden, wie die Entscheider die Methoden praktikabel nutzen können.

Verwendete Hilfsmittel und ergänzendes Material

Bei der Anfertigung dieses Beitrags wurde DeepL ([deepl.com](https://www.deepl.com)) für sprachliche Verbesserungen im englischen Abstract verwendet. Ferner wurde auf ChatGPT (chat.openai.com) zurückgegriffen, um einzelne Passagen des Beitrags sprachlich zu verbessern. Die Visualisierungen und Tabellen wurden mit R und dem Tidyverse unter Verwendung des R-Studio Desktop ([r-studio.com](https://www.rstudio.com)) erstellt. Als ergänzendes Material zu diesem Beitrag werden die in den Experimenten erhobenen Daten zusammen mit den Analyseergebnissen über eine GitHub-Website veröffentlicht. Im gleichen Repository befinden sich auch die R-Quelldateien. Der Quellcode für das entwickelte SimBO-Framework ist ebenfalls auf GitHub veröffentlicht:

- Analysen: <https://winf-hsos.github.io/paper-parameter-opt-mrp-bo>
- R-Quellcode: <https://github.com/winf-hsos/paper-parameter-opt-mrp-bo>
- SimBO-Framework: <https://github.com/pehzet/SimBO>

Literatur

- Amaran, S., Sahinidis, N.V., Sharda, B., Bury, S.J., 2016. Simulation optimization: A review of algorithms and applications. *Ann Oper Res* 240, 351–380.
- Balandat, M., Karrer, B., Jiang, D.R., Daulton, S., Letham, B., Wilson, A.G., Bakshy, E., 2020. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. ePrint arXiv:1910.06403.
- Berkenkamp, F., Krause, A., Schoellig, A.P., 2021. Bayesian Optimization with Safety Constraints: Safe and Automatic Parameter Tuning in Robotics. *Mach Learn* (2021)
- Bull, A.D., 2011. Convergence rates of efficient global optimization algorithms.
- Damand, D., Lahrachi, Y., Barth, M., 2022. A simulation-optimization approach to parameterize Demand-Driven Material Requirements Planning. *IFAC-PapersOnLine*, 10th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2022 55, 263–268.
- Daulton, S., Balandat, M., Bakshy, E., 2021. Parallel Bayesian Optimization of Multiple Noisy Objectives with Expected Hypervolume Improvement.
- Daulton, S., Eriksson, D., Balandat, M., Bakshy, E., 2022. Multi-Objective Bayesian Optimization over High-Dimensional Search Spaces.
- Eriksson, D., Jankowiak, M., 2021. High-Dimensional Bayesian Optimization with Sparse Axis-Aligned Subspaces.

- Eriksson, D., Pearce, M., Gardner, J., Turner, R.D., Poloczek, M., 2019. Scalable Global Optimization via Local Bayesian Optimization, in: *Advances in Neural Information Processing Systems*. Curran Associates, Inc.
- Frazier, P.I., 2018. A Tutorial on Bayesian Optimization. eprint arXiv:1807.02811
- Gansterer, M., Almeder, C., Hartl, R.F., 2014. Simulation-based optimization methods for setting production planning parameters. *International Journal of Production Economics* 151, 206–213.
- Hansen, N., 2016. The CMA Evolution Strategy: A Tutorial. ePrint arXiv.1604.00772
- Letham, B., Calandra, R., Rai, A., Bakshy, E., 2020. Re-Examining Linear Embeddings for High-Dimensional Bayesian Optimization.
- Nguyen, A.-T., Reiter, S., Rigo, P., 2014. A review on simulation-based optimization methods applied to building performance analysis. *Applied Energy* 113, 1043–1058.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., Chintala, S., 2019. PyTorch: An Imperative Style, High-Performance Deep Learning Library.
- Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., Freitas, N. de, 2016. Taking the Human Out of the Loop: A Review of Bayesian Optimization. *Proceedings of the IEEE* 104, 148–175.
- Sobol', I.M., 1967. On the distribution of points in a cube and the approximate evaluation of integrals. *USSR Computational Mathematics and Mathematical Physics* 7, 86–112.
- Stadtler, H., Kilger, C., Meyr, H. (Eds.), 2015. *Supply Chain Management and Advanced Planning: Concepts, Models, Software, and Case Studies*, Springer Texts in Business and Economics. Springer, Berlin, Heidelberg.
- Wang, Z., Hutter, F., Zoghi, M., Matheson, D., De Freitas, N., 2016. Bayesian Optimization in a Billion Dimensions via Random Embeddings. *jair* 55, 361–387.
- Werth, B., Karder, J., Beham, A., Altendorfer, K., 2023. Simulation-based Optimization of Material Requirements Planning Parameters. *Procedia Computer Science*, 4th International Conference on Industry 4.0 and Smart Manufacturing 217, 1117–1126.
- Zhang, Y., Apley, D.W., Chen, W., 2020. Bayesian Optimization for Materials Design with Mixed Quantitative and Qualitative Variables. *Scientific Reports* 10, 4924.