

Szenariobaumapproximation für stochastische Optimierungsprobleme in der Energiewirtschaft

Dr. **H. Heitsch**, Institut für Mathematik, Humboldt-Universität zu Berlin

Kurzfassung

Viele Entscheidungsprobleme in der Energiewirtschaft führen in ihrer Modellierung zu mehrstufigen stochastischen Optimierungsproblemen. Typisch für solche Probleme ist eine Entscheidungsfindung in Unkenntnis von Realisierungen zufallsbehafteter Daten. Eine mögliche Beschreibung des Zufalls besteht in der Abbildung aller unsicheren Daten in Form eines (multivariaten) stochastischen Datenprozesses, der als Parameter in das Optimierungsmodell Eingang findet. Dabei müssen Annahmen sowohl über die Verteilung als auch hinsichtlich der Entscheidungsstruktur getroffen werden. Untersuchungen in aktuellen Veröffentlichungen belegen die Bedeutung der Informationsstruktur in Verbindung mit der Verteilungsinformation. Darüber hinaus sind stochastische Optimierungsprobleme in der Regel nur dann numerisch behandelbar, wenn die zugrunde liegende Stochastik mit einer endlichen Anzahl von Szenarien in Form eines Szenariobaums beschrieben wird. Daher spielt eine adäquate Szenariobaumapproximation bei der Lösung dieser Probleme eine zentrale Rolle. Vorgestellt werden aktuelle Methoden zur optimalen Konstruktion und Reduktion von Szenariobäumen. Mit Anwendungsbeispielen aus der Energiewirtschaft werden Praxisrelevanz sowie Einsatzvielfalt der Methoden zur Szenariobaumapproximation belegt.

1 Einleitung

Optimierungsprobleme in der Energiewirtschaft enthalten häufig unsichere Parameter, die in der Zielfunktion oder in den Gleichungen bzw. Ungleichungen, die die Restriktionen beschreiben, auftreten. Solche unsicheren Parameter sind beispielsweise zukünftige Strompreise am Spot- und Terminmarkt, Preise von Stromderivaten an den Strombörsen, Primärenergiepreise (für thermische Kraftwerke), die elektrische Last, Zuflüsse zu Reservoirs hydraulischer Kraftwerke, oder das Dargebot an erneuerbaren Energien für entsprechende Erzeugungsanlagen. Alle diese unsicheren Parameter erlauben eine Interpretation als Zufallsgrößen oder stochastische Prozesse. In den meisten Fällen existieren historische Daten bzw. ausgereifte stochastische Modelle. Als Beispiele dafür seien die Modelle für Strompreise [2, 3, 13, 17, 19, 21], die elektrische Last [5, 19] oder die Windgeschwindigkeit [6] genannt.

Bezeichnet $\xi = \{\xi_t\}_{t=1}^T$ den stochastischen Prozess mit d -dimensionalen Zufallsvariablen ξ_t für jeden Zeitpunkt t , der in ein energiewirtschaftliches Modell über einem diskreten Zeithorizont mit T Zeitpunkten eingeht, so kann ein solches Modell häufig in der Form

$$\min \left\{ \mathbb{E} \left(\sum_{t=1}^T f_t(x_t, \xi_t) \right) : \begin{array}{l} x_t \in X_t, \sum_{\tau=0}^{t-1} A_{t\tau} x_{t-\tau} = h_t(\xi_t), \\ B_t x_t = d_t(\xi_t), t = 1, \dots, T \end{array} \right\} \quad (1)$$

geschrieben werden. Dabei sind die Entscheidungen $x = (x_1, \dots, x_T)$ stochastische Prozesse, die den in (1) enthaltenen Restriktionen genügen müssen. Einerseits sollen sie zu jedem Zeitpunkt t zu einer gewissen Menge X_t gehören, die durch endlich viele lineare Gleichungen und Ungleichungen beschrieben wird. Überdies liegen zeitübergreifende Restriktionen und Bilanzgleichungen für den Entscheidungsprozess x an jedem Zeitpunkt vor. Erstere beschreiben z.B. die zeitliche Entwicklung der Zustände eines Energiesystems. Die Matrizen $A_{t\tau}$ und B_t besitzen dabei geeignete Dimension und $h_t(\xi_t)$ bzw. $d_t(\xi_t)$ sind evtl. vom stochastischen Prozess ξ abhängige Vektoren, die z.B. Zuflüsse, die elektrische Last und Windenergie beschreiben. Die Funktionen $f_t(\cdot, \xi_t)$ beschreiben Kostenfunktionen, die von unsicheren Preisen abhängen können, und z.B. (stückweise) lineare oder quadratische Funktionen von Komponenten von x_t sind. Ziel ist hier die Minimierung der erwarteten Kosten (das Funktional \mathbb{E} liefert den Erwartungswert).

Der Zeitpunkt $t = 1$ repräsentiert die Gegenwart oder unmittelbar bevorstehende Zukunft, d.h. der Wert von ξ_1 ist bekannt oder kann hinreichend gut vorhergesagt werden. Daher wird im Modell (1) angenommen, dass die Entscheidung x_1 der ersten Zeitstufe deterministisch ist. Dies wird durch die Restriktion $x_1 = \mathbb{E}(x_1)$ beschrieben. Dann läßt sich das stochastische Optimierungsmodell (1) in das Optimierungsproblem

$$\min \left\{ f_1(x_1, \xi_1) + \mathbb{E} \left(\Phi(x_1, \bar{\xi}) \right) : \begin{array}{l} x_1 = \mathbb{E}(x_1), x_1 \in X_1, \\ A_{10} x_1 = h_1(\xi_1), B_1 x_1 = d_1(\xi_1) \end{array} \right\} \quad (2)$$

bzgl. der Erststufenentscheidung x_1 umformulieren, wobei $\bar{\xi} := (\xi_2, \dots, \xi_T)$ der unsicheren Zukunft entspricht und die Funktion Φ definiert ist durch

$$\Phi(x_1, \bar{\xi}) := \inf \left\{ \mathbb{E} \left(\sum_{t=2}^T f_t(x_t, \xi_t) \right) : \begin{array}{l} x_t \in X_t, \sum_{\tau=0}^{t-1} A_{t\tau} x_{t-\tau} = h_t(\xi_t), \\ B_t x_t = d_t(\xi_t), t = 2, \dots, T \end{array} \right\}.$$

Im Problem (2) ist dann die kostenminimale Erststufenentscheidung gesucht, die auch die erwarteten zukünftigen Kosten bei unsicherer Zukunft minimiert. Hierbei handelt es sich um ein *zweistufiges stochastisches Optimierungsproblem*.

Während die Erststufenentscheidung x_1 deterministisch ist, hängt die Zweitstufenentscheidung $\bar{x} := (x_2, \dots, x_T)$ und auch jede ihrer Komponenten vom gesamten stochastischen Prozess $\bar{\xi}$ ab.

Will man überdies modellieren, dass die Entscheidung x_t zum Zeitpunkt t nur von der bis dahin vorhandenen Information (ξ_2, \dots, ξ_t) abhängt, dass also der Entscheidungsprozess rekursiv ist und zu jedem neuen Zeitpunkt auf der Basis neuer Information erfolgt, so kann dies über neue Restriktionen der Form

$$x_t = \mathbb{E}(x_t | \xi_1, \xi_2, \dots, \xi_t) \quad (t = 1, \dots, T) \quad (3)$$

in (1) formuliert werden. Dabei stellt $\mathbb{E}(\cdot | \xi_1, \xi_2, \dots, \xi_t)$ die bedingte Erwartung bzgl. des Zufallsvektors $(\xi_1, \xi_2, \dots, \xi_t)$ dar und die Restriktion bedeutet, dass für jedes $t = 2, \dots, T$, die Entscheidung x_t eine Funktion von $(\xi_1, \xi_2, \dots, \xi_t)$ darstellt. Die neuen Nebenbedingungen (3) heißen auch *Nichtantizipativitätsrestriktionen* und Problem (1) mit (3) *mehrstufiges stochastisches Optimierungsproblem*. Die Restriktion für $t = 1$ in (3) entspricht der bisherigen Bedingung $x_1 = \mathbb{E}(x_1)$ in (2).

2 Approximation durch Szenarien

Um zwei- oder mehrstufige stochastische Optimierungsprobleme in Anwendungen näherungsweise numerisch zu lösen, muss der stochastische Prozess ξ durch einen Prozess mit endlich vielen Szenarien (oder *Szenarios*) formuliert werden. Das bedeutet, dass die zugrundeliegende Wahrscheinlichkeitsverteilung des stochastischen Prozesses ξ durch eine diskrete Verteilung mit endlich vielen Atomen ersetzt wird. Es seien nun ξ^i diese Szenarien mit den zugehörigen Wahrscheinlichkeiten p_i , $i = 1, \dots, N$. Alle Szenarien besitzen die Form

$$\xi^i = (\xi_1^i, \xi_2^i, \dots, \xi_T^i) \quad \text{mit} \quad \xi_1^i = \xi_1 \quad (i = 1, \dots, N).$$

Das stochastische Optimierungsproblem (2) nimmt dann die Form

$$\min \left\{ f_1(x_1, \xi_1) + \sum_{i=1}^N p_i \Phi(x_1, \bar{\xi}^i) : x_1 = \mathbb{E}(x_1), A_{10}x_1 = h_1(\xi_1) \right\} \quad (4)$$

an. Bei der Lösung von (4) stellt sich das folgende Problem. Einerseits ermöglicht eine große Zahl von Szenarien eine gute Darstellung der stochastischen Parameter. Andererseits wächst mit der Zahl der Szenarien der Aufwand zur Lösung des Optimierungsproblems. Daher ist es mitunter notwendig, sich bei der Lösung von (4) auf eine deutlich reduzierte Auswahl aus den N Ausgangsszenarien zu beschränken.

Betrachtet man ein mehrstufiges stochastisches Optimierungsproblem, so modelliert dieses einen Prozess sukzessiver Beobachtungen und Entscheidungen. Die im Verlauf der Zeit sukzessive erfolgende Beobachtung der stochastischen Parameter kann durch die Anordnung der Szenarien in einer Baumstruktur modelliert werden. Bezeichnet k_t die Anzahl der möglichen Werte von ξ bzw. x zum Zeitpunkt t , so wächst diese Anzahl mit t , wobei $k_1 = 1$ der Wurzel

und $k_T = N$ den Blättern des Baums entsprechen. Die Restriktion (3) nimmt dann die Gestalt (szenarienverkoppelnder) Gleichungsrestriktionen

$$x_t^i = \mathbb{E}(x_t | \xi_1, \xi_2, \dots, \xi_t)^i = \left(\sum_{j \in C_t^i} p_j \right)^{-1} \sum_{j \in C_t^i} p_j x_t^j \quad (i = 1, \dots, k_t, t = 1, \dots, T) \quad (5)$$

an, wobei die Mengen C_t^i , $i = 1, \dots, k_t$, eine Zerlegung der Menge $\{1, \dots, N\}$ darstellen und (wegen $k_1 = 1$ bzw. $k_T = N$) $C_1^1 = \{1, \dots, N\}$ bzw. $C_T^i = \{i\}$, $i = 1, \dots, k_T = N$ gilt. Sämtliche Mengen C_t^i werden mit wachsendem t kleiner. Unterteilt sich die Menge C_t^i ($i = 1, \dots, k_t$) beim Übergang von t zu $t + 1$ in die von i abhängigen Mengen C_{t+1}^j , $j = j_i, \dots, j_{i+1} - 1$ (mit $j_1 = 1$ und $j_{k_{t+1}} - 1 = k_{t+1}$), so bedeutet dies, dass sich Szenario i in t in $j_{i+1} - j_i$ Szenarien *verzweigt* (siehe auch Bild 4). Allerdings weisen aus historischen oder simulierten Daten gewonnene Szenarien eine solche Baumstruktur im Allgemeinen nicht auf. Im folgenden Abschnitt wird beschrieben, wie durch sukzessive Anwendung der Szenarioreduktion eine Baumstruktur hergestellt werden kann.

3 Konstruktion von Szenariobäumen

Im mehrstufigen Fall modellieren die Szenarien einen Prozess sukzessiver Beobachtungen und Entscheidungen und besitzen deshalb Baumstruktur. Diese beschreibt den fortschreitenden Informationszuwachs bzw. die zunehmenden Entscheidungsvarianten innerhalb des Optimierungszeitraums. Die Grundlage für die Konstruktion von Szenariobäumen für mehrstufige stochastische Optimierungsprobleme bilden Techniken der Szenarioreduktion.

3.1 Szenarioreduktion

In diesem Abschnitt werden die in [4, 9, 12, 11] entwickelten Ansätze und Algorithmen zur Szenarioreduktion für lineare zweistufige stochastische Optimierungsprobleme beschrieben. Bei der numerische Lösung stochastischer Optimierungsprobleme wird in der Regel die ursprüngliche Wahrscheinlichkeitsverteilung P durch eine Verteilung Q mit endlich vielen Atomen bzw. Szenarien ersetzt. Dabei ist es naheliegend, die Wahrscheinlichkeitsverteilung Q derart zu wählen, dass die Ergebnisse der Optimierung (d.h. Optimalwerte und Lösungen) sich bei dieser Ersetzung möglichst wenig verändern, oder anders ausgedrückt, die Szenarioreduktion basiert auf der Verwendung von solchen Metriken für Wahrscheinlichkeitsverteilungen bzgl. derer sich die stochastischen Optimierungsprobleme *stabil* verhalten (vgl. [14, 15, 18]).

Besonders geeignet sind Metriken, die sich als Monge-Kantorovich Transportproblem formulieren lassen. Solche Transport-Funktionale haben allgemein die Form

$$\mu_c(P, Q) := \inf \left\{ \int_{\Xi \times \Xi} c(\xi, \tilde{\xi}) \eta(d\xi, d\tilde{\xi}) : \eta \in M(P, Q) \right\}, \quad (6)$$

wobei $M(P, Q)$ die Menge aller Wahrscheinlichkeitsverteilungen auf $\Xi \times \Xi$ ist, deren Marginalverteilungen auf der Trägermenge Ξ gerade P und Q sind, und c eine symmetrische und nichtnegative Kostenfunktion ist (vgl. [14, 16]). Die Trägermenge Ξ beschreibt hier die Menge aller möglichen Szenarien und stellt im Allgemeinen eine abgeschlossene Teilmenge im \mathbb{R}^s dar. Damit μ_c endlich ist, muss c bzgl. P und Q integrierbar sein. Für die spezielle Wahl der Kostenfunktion c

$$c(\xi, \tilde{\xi}) := |\xi - \tilde{\xi}|^r \quad (\forall \xi, \tilde{\xi} \in \Xi)$$

mit $r \in \{1, 2\}$ entsteht dabei die L_r -minimale Metrik ℓ_r , definiert durch

$$\ell_r(P, Q)^r = \inf \left\{ \int_{\Xi \times \Xi} |\xi - \tilde{\xi}|^r \eta(d\xi, d\tilde{\xi}) : \eta \in M(P, Q) \right\} \quad (7)$$

auf der Menge $\mathcal{P}_r(\Xi)$ aller Wahrscheinlichkeitsverteilungen mit endlichem r -ten Moment (vgl. [16, Chapter 2.6]).

Aufgabe der Szenarioreduktion ist die Verringerung der Anzahl von Szenarien bei möglichst geringem Informationsverlust. Dazu sei P nun eine diskrete Wahrscheinlichkeitsverteilung mit Szenarien ξ^i und Wahrscheinlichkeiten p_i für $i \in \{1, \dots, N\}$. Weiter sei Q_J eine Wahrscheinlichkeitsverteilung, deren Träger $\text{supp}(Q_J)$ nur die folgende Teilmenge der Szenarien von P umfasst:

$$\text{supp}(Q_J) = \{\xi^i : i \in \{1, \dots, N\} \setminus J\} \quad \text{und} \quad J \subset \{1, \dots, N\}.$$

Die Wahrscheinlichkeiten von Q_J werden mit q_i ($i \notin J$) bezeichnet. Motiviert durch aktuelle Stabilitätsresultate besteht das Ziel nun darin, die Wahrscheinlichkeitsverteilung Q_J so zu bestimmen, dass der Abstand $\mu_c(P, Q_J)$ möglichst klein wird. Dies bedeutet, dass bei fixiertem $n \in \mathbb{N}$ mit $n < N$, die Indexmenge J und die Wahrscheinlichkeiten q_i , $i = 1, \dots, n$, als Lösung des Minimumproblems

$$\min \left\{ \mu_c(P, Q_J) : J \subset \{1, \dots, N\}, \#J = N - n, q_i \geq 0, i \notin J, \sum_{i \notin J} q_i = 1 \right\} \quad (8)$$

bestimmt werden (*optimale Szenarioreduktion*). Das Minimumproblem (8) läßt sich durch

$$\min_J \left\{ \min_q \left\{ \mu_c(P, Q_J) : q_i \geq 0, i \notin J, \sum_{i \notin J} q_i = 1 \right\} : J \subset \{1, \dots, N\}, \#J = N - n \right\}$$

in ein inneres Optimierungsproblem zur Bestimmung von q_i , $i \notin J$, und ein äußeres Minimierungsproblem für die Indexmenge J aufspalten. Bei der inneren Minimierung ergibt sich das von J abhängige Infimum D_J durch

$$D_J := \min \left\{ \mu_c(P, Q_J) : q_i \geq 0, i \notin J, \sum_{i \notin J} q_i = 1 \right\}. \quad (9)$$

Das innere Optimierungsproblem (9) lässt sich explizit lösen (vgl. [4, Theorem 2]). Zum Beispiel für die Metrik ℓ_r ergibt sich demnach

$$D_J = \min \left\{ \ell_r(P, Q_J) : q_i \geq 0, i \notin J, \sum_{i \notin J} q_i = 1 \right\} = \left(\sum_{j \in J} p_j \min_{i \notin J} |\xi^i - \xi^j|^r \right)^{\frac{1}{r}}. \quad (10)$$

Darüber hinaus wird das Minimum in (10) durch die Wahrscheinlichkeitsverteilung Q_J^* mit

$$q_i^* = p_i + \sum_{\substack{j \in J \\ i(j)=i}} p_j \quad \text{und} \quad i(j) \in \arg \min_{i \notin J} |\xi^i - \xi^j|^r, \quad i \notin J \quad (11)$$

realisiert. Die Formel (11) liefert eine Vorschrift zur *optimalen Neuverteilung*. Diese besteht darin, die Wahrscheinlichkeit eines gestrichenen Szenarios zu der Wahrscheinlichkeit eines (bzgl. des Abstands $|\cdot|^r$) nächstgelegenen Szenarios zu addieren.

Mit Hilfe des Funktionals D_J kann das Problem (8) der optimalen Szenarioreduktion nun in der Form

$$\min \{ D_J : J \subset \{1, \dots, N\}, \#J = N - n \} \quad (12)$$

geschrieben werden. Das Funktional D_J wird dabei für die Metrik ℓ_r mit Hilfe der Formel (10) berechnet. Die Optimierungsaufgabe besteht in der Bestimmung einer Indexmenge J mit vorgeschriebener Kardinalität $1 \leq N - n = \#J < N$ und minimalen Kosten D_J . Solche kombinatorischen Optimierungsprobleme sind im Allgemeinen \mathcal{NP} -schwer. Das Problem (12) stellt insbesondere ein sog. n -median-Problem dar (vgl. [20, Chapter 25]).

Aufgrund ihrer hohen Komplexität können kombinatorische Probleme dieser Art nur selten exakt gelöst werden (vgl. [1]) und man greift in der Regel auf Näherungsalgorithmen zurück. Speziell für das Problem der optimalen Szenarioreduktion wurden in [4] und [8] zwei heuristische Algorithmen zur näherungsweise Lösung von (12) vorgeschlagen, der Vorwärts- und Rückwärtsalgorithmus.

Heuristische Algorithmen

Die Basis für den Vorwärtsalgorithmus der Szenarioreduktion bildet die einfache Gestalt des Problems im Spezialfall $n = 1$. Es läßt sich kurz in der Form

$$\min_{u \in \{1, \dots, N\}} \sum_{\substack{j=1 \\ j \neq u}}^N p_j |\xi^j - \xi^u|^r$$

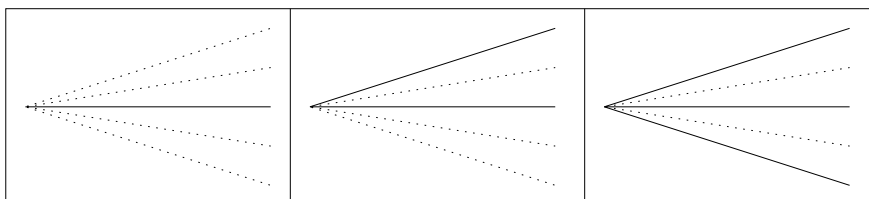
darstellen. Wird das Minimum in u^* angenommen, erhält man mit $J = \{1, \dots, N\} \setminus \{u^*\}$ eine Lösung von (12) im Fall $n = 1$. Damit ist das erste Element des Trägers von Q_J gefunden. Unter Verwendung der additiven Gestalt von D_J in (10) bestimmt man bei fixiertem ersten Element analog das zweite Element des Trägers von Q_J und setzt diesen Prozess bis $\#J = N - n$ fort. Dies führt zu folgender Heuristik zur approximativen Lösung von (12) für beliebiges n .

Algorithm 3.1 (*Vorwärtsalgorithmus Szenarioreduktion*)

Step 0: $J^{[0]} := \{1, \dots, N\}.$

Step k: $u_k \in \arg \min_{u \in J^{[k-1]}} \sum_{j \in J^{[k-1]} \setminus \{u\}} p_j \min_{i \in J^{[k-1]} \setminus \{u\}} |\xi^j - \xi^i|^r,$
 $J^{[k]} := J^{[k-1]} \setminus \{u_k\}.$

Step n+1: *Optimale Neuverteilung mit $J := J^{[n]}$.*



Das Prinzip des Algorithmus besteht darin, das Szenario mit dem Index u_k im Schritt k so zu wählen, dass der Wert des Funktional $D_{J^{[k]}}$ minimiert wird, d.h.

$$D_{J^{[k-1]} \setminus \{u_k\}} = \min_{u \in J^{[k-1]}} D_{J^{[k-1]} \setminus \{u\}}.$$

Die Menge der verbleibenden Szenarien und damit der Träger der Wahrscheinlichkeitsverteilung Q_J ist am Ende durch die Indexmenge $\{u_1, \dots, u_n\}$ bestimmt.

Die Idee des Rückwärtsalgorithmus der Szenarioreduktion basiert auf dem zweiten Spezialfall des Problems (12). Dieser entsteht für $n = N - 1$ und besitzt die Form

$$\min_{l \in \{1, \dots, N\}} p_l \min_{i \neq l} |\xi^l - \xi^i|^r.$$

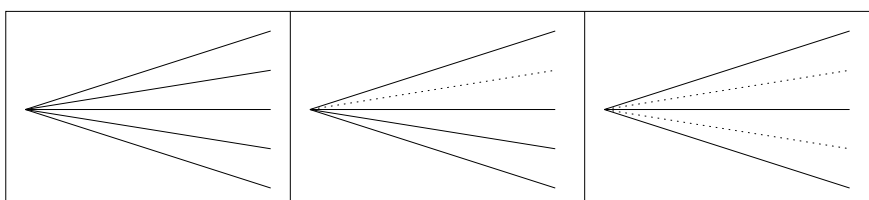
Wird das Minimum in l^* angenommen, ergibt sich $J = \{l^*\}$ als eine Lösung von (12) im Fall $n = N - 1$. Auch dieser Spezialfall läßt sich schrittweise zu einem heuristischen Algorithmus zur approximativen Lösung von (12) für beliebige n fortsetzen.

Algorithm 3.2 (*Rückwärtsalgorithmus Szenarioreduktion*)

Step 0: $J^{[0]} := \emptyset.$

Step k: $l_k \in \arg \min_{l \notin J^{[k-1]}} \sum_{j \in J^{[k-1]} \cup \{l\}} p_j \min_{i \in J^{[k-1]} \cup \{l\}} |\xi^j - \xi^i|^r,$
 $J^{[k]} := J^{[k-1]} \cup \{l_k\}.$

Step N-n+1: *Optimale Neuverteilung mit $J := J^{[N-n]}$.*



Der Algorithmus erlaubt eine ähnliche Interpretation wie zuvor. Der Szenarioindex l_k im Schritt k wird erneut so gewählt, dass der Wert des Funktionals $D_{J^{[k]}}$ minimiert wird, d.h.

$$D_{J^{[k-1]}\cup\{l_k\}} = \min_{l \notin J^{[k-1]}} D_{J^{[k-1]}\cup\{l\}}.$$

Die Menge der verbleibenden Szenarien nach Ablauf des Algorithmus und damit der Träger des Maßes Q_J ist hier durch die Indexmenge $\{1, \dots, N\} \setminus \{l_1, \dots, l_{N-n}\}$ bestimmt.

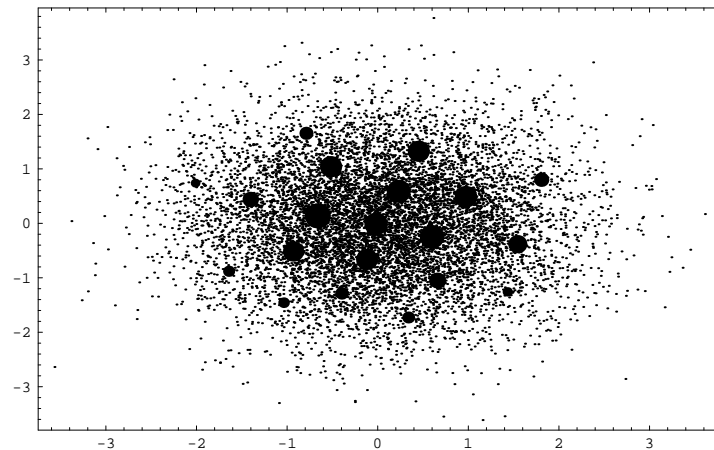


Bild 1: Ergebnis der Szenarioreduktion am Beispiel der zweidimensionalen Standardnormalverteilung. Ausgehend von 10 000 simulierten Szenarien, wurde mit dem Vorwärtsalgorithmus eine Szenarioreduktion auf 20 Szenarien bzgl. ℓ_2 durchgeführt. Dabei sind die verbleibenden 20 repräsentativen Szenarien durch unterschiedlich große Punkte proportional zu ihren neuen Wahrscheinlichkeiten dargestellt.

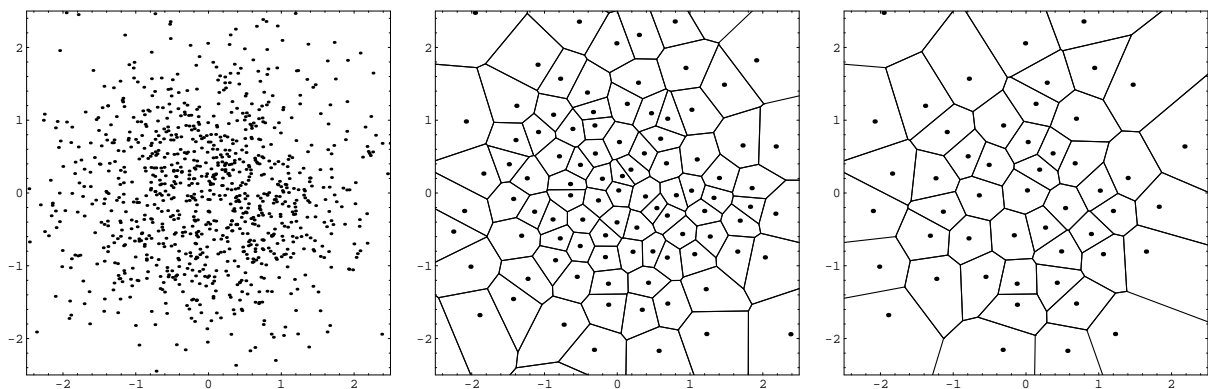


Bild 2: Veranschaulichung der Szenarioreduktion bzgl. ℓ_2 am Beispiel der 2-dimensionalen Standardnormalverteilung ausgehend von 1 000 generierten Szenarien (links) bei einer Reduktion auf 100 Szenarien (Mitte) und 50 Szenarien (rechts). Dargestellt sind die durch die jeweilige Szenarioreduktion vorgenommenen Zerlegungen der Ebene in sog. Voronoi-Regionen.

3.2 Szenariobaumkonstruktion für mehrstufige Modelle

Die vorgestellten Techniken zur Szenarioreduktion können als Grundlage für die Konstruktion von Szenariobäumen im Fall mehrstufiger stochastischer Optimierungsprobleme genutzt werden. Szenarien, die man aus historischen Daten oder durch Simulation aus angepassten Zeitreihen- oder Regressionsmodellen erhält, weisen im Allgemeinen noch keine Baumstruktur auf. Daher ist eine Methodik erforderlich, die aus Ausgangsszenarien einen geeigneten Szenariobaum ξ_{tr} generiert, der den stochastischen Prozess ξ approximiert.

Im Folgenden bezeichnen wir den stochastischen Prozess, der aus den Ausgangsszenarien $\xi^i = (\xi_1^i, \dots, \xi_T^i)$ mit den Wahrscheinlichkeiten p_i , $i = 1, \dots, N$, besteht, mit $\hat{\xi}$. Um eine deterministische erste Stufe zu realisieren, sei zusätzlich die Bedingung $\xi_1^1 = \dots = \xi_1^N =: \xi_1$ erfüllt (siehe Bild 3 zur Veranschaulichung).

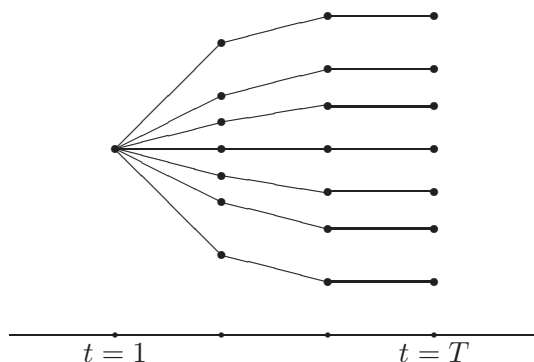


Bild 3: Prozess $\hat{\xi}$ mit Ausgangsszenarien für $T = 4$ und $N = 7$

Ein naheliegendes Konzept zur Erzeugung der Baumstruktur besteht in der rekursiven Anwendung der Szenarioreduktion unter Berücksichtigung variabler Zeitintervalle. Zieht man dabei nacheinander Zeitintervalle der Form $\{1, \dots, t\}$ für entweder wachsende oder fallende Zeitparameter t in Betracht, lassen sich erneut zwei Algorithmen ableiten, die je nach Ausrichtung als Vorwärts- und Rückwärtsalgorithmus in [10] ausführlich diskutiert werden. Beide Vorgehensweisen ermöglichen die schrittweise Rekonstruktion der implizit vorliegenden Informationsstruktur (vgl. [7]). Im Folgenden wird der Vorwärtsalgorithmus vorgestellt.

Vorwärtsalgorithmus zur Baumkonstruktion

Die Strategie des Vorwärtsalgorithmus basiert auf einer rekursiven Szenarioreduktion für wachsende Zeiträume $\{1, \dots, t\}$ für $t = 2, \dots, T$. Das Resultat sind aufsteigende Partitionen der Indexmenge $I := \{1, \dots, N\}$, die zum Zeitpunkt t die Gestalt

$$C_t := \{C_t^1, \dots, C_t^{k_t}\}, \quad k_t \in \mathbb{N}.$$

annehmen. Die Partitions Mengen erfüllen dabei für jedes t die Partitionseigenschaft

$$C_t^k \cap C_t^{k'} = \emptyset, \quad \text{falls } k \neq k', \quad \text{und} \quad \bigcup_{k=1}^{k_t} C_t^k = I.$$

Die Elemente der Partition C_t werden mitunter auch Szenario-Cluster genannt. Der nun folgende Algorithmus erlaubt die Konstruktion eines Szenariobaumprozesses ξ_{tr} , dessen Gestalt über sog. Toleranzparameter ε_t , $t = 2, \dots, T$, innerhalb des Algorithmus gesteuert werden kann.

Algorithm 3.3 (*Vorwärtsalgorithmus Baumkonstruktion*)

[Initialisierung]

Setze $C_1 = \{I\}$ und $t := 2$.

[Cluster-Berechnung]

Es sei $C_{t-1} = \{C_{t-1}^1, \dots, C_{t-1}^{k_{t-1}}\}$. Führe für jede Szenarioteilmenge $\{\xi_t^i\}_{i \in C_{t-1}^k}$ separat für jedes $k \in \{1, \dots, k_{t-1}\}$ eine Szenarioreduktion durch (d.h. nur bezüglich der t -ten Komponente). Definiere Indexmengen J_t^k und I_t^k der jeweils gestrichenen bzw. verbleibenden Szenarien sowie partielle Zuordnungen $i_t^k : J_t^k \rightarrow I_t^k$, so dass

$$i_t^k(j) \in \arg \min_{i \in I_t^k} |\xi_t^i - \xi_t^j|^r, \quad j \in J_t^k,$$

entsprechend der Szenarioreduktion (vgl. Abschnitt 3.1). Definiere eine Abbildung $\alpha_t : I \rightarrow I$, so dass

$$\alpha_t(j) = \begin{cases} i_t^k(j), & j \in J_t^k \text{ für ein } k = 1, \dots, k_{t-1}, \\ j, & \text{sonst.} \end{cases} \quad (13)$$

Die Partition zum Zeitpunkt t wird damit durch

$$C_t := \left\{ \alpha_t^{-1}(i) \mid i \in I_t^k, k = 1, \dots, k_{t-1} \right\}$$

definiert, was einer Verfeinerung der Partition C_{t-1} entspricht. Im Fall $t < T$ führe eine weitere Cluster-Berechnung mit $t := t + 1$ durch. Andernfalls gehe zur abschließenden Baumgenerierung.

[Baumgenerierung]

Entsprechend der Partition C_T und der Abbildungen (13) definiere den Szenariobaumprozess ξ_{tr} über die Szenarien

$$\xi_{\text{tr}}^k = \left(\xi_1^*, \xi_2^{\alpha_2(i)}, \dots, \xi_t^{\alpha_t(i)}, \dots, \xi_T^{\alpha_T(i)} \right) \quad \text{für ein } i \in C_T^k,$$

versehen mit den Wahrscheinlichkeiten $q_k := \sum_{i \in C_T^k} p_i$ für $k = 1, \dots, k_T$.

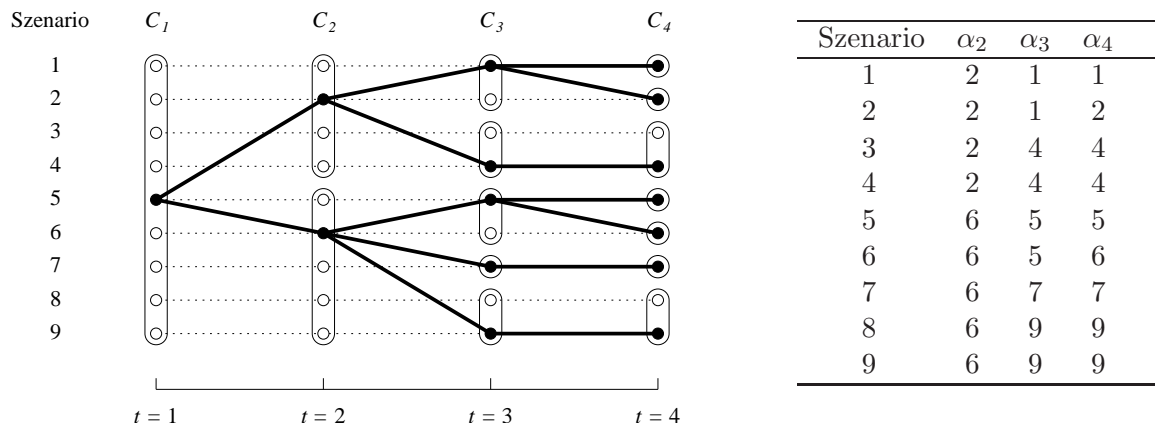


Bild 4: Schematische Darstellung der Clusterung durch den Vorwärtsalgorithmus 3.3 bei der Szenariobaumkonstruktion (links) und Veranschaulichung der zugehörigen Zuordnungsabbildungen α_t (rechts).

Für die Umsetzung der Cluster-Berechnung innerhalb des Algorithmus können beide Algorithmen der Szenarioreduktion aus Abschnitt 3.1 herangezogen werden. Der Fehler im Schritt t läßt sich durch

$$\text{err}_t := \sum_{k=1}^{k_{t-1}} \sum_{j \in J_t^k} p_j \min_{i \in I_t^k} |\xi_t^i - \xi_t^j|^r$$

bestimmen. Analog zu [7, Theorem 6.4] kann damit der gesamte Approximationsfehler durch

$$\ell_r(P_{\hat{\xi}}, P_{\xi_{\text{tr}}}) \leq \left(\sum_{t=2}^T \text{err}_t \right)^{\frac{1}{r}}$$

abgeschätzt werden, wobei $P_{\hat{\xi}}$ und $P_{\xi_{\text{tr}}}$ die Wahrscheinlichkeitsverteilungen von $\hat{\xi}$ bzw. ξ_{tr} bezeichnen. Die letzte Abschätzung erlaubt die Steuerung des Konstruktionsprozesses durch die Vorgabe von Toleranzen ε_t für err_t ($t = 2, \dots, T$).

4 Anwendung

Der Vorwärtsalgorithmus wurde im Rahmen verschiedener energiewirtschaftlicher Modelle angewendet. So wurden unter anderem Last-Spotpreis Szenariobäume sowie Wind-Spotpreis Bäume für unterschiedliche Zeiträume von bis zu einem Jahr und einer stündlichen Diskretisierung erzeugt.

Ein Beispiel, das hier exemplarisch diskutiert werden soll, basiert auf Szenarien für ein Zwei-jahresmodell zur Optimierung des Einsatzfahrplans eines Kraftwerkverbunds zur Energieerzeugung, die vom französischen Energiekonzern EDF (Electricité de France) bereitgestellt wurden. Die Daten stellen endlich viele Realisierungen eines bivariaten stochastischen Prozesses dar, dessen Komponenten sich einerseits aus elektrischer Last und andererseits aus

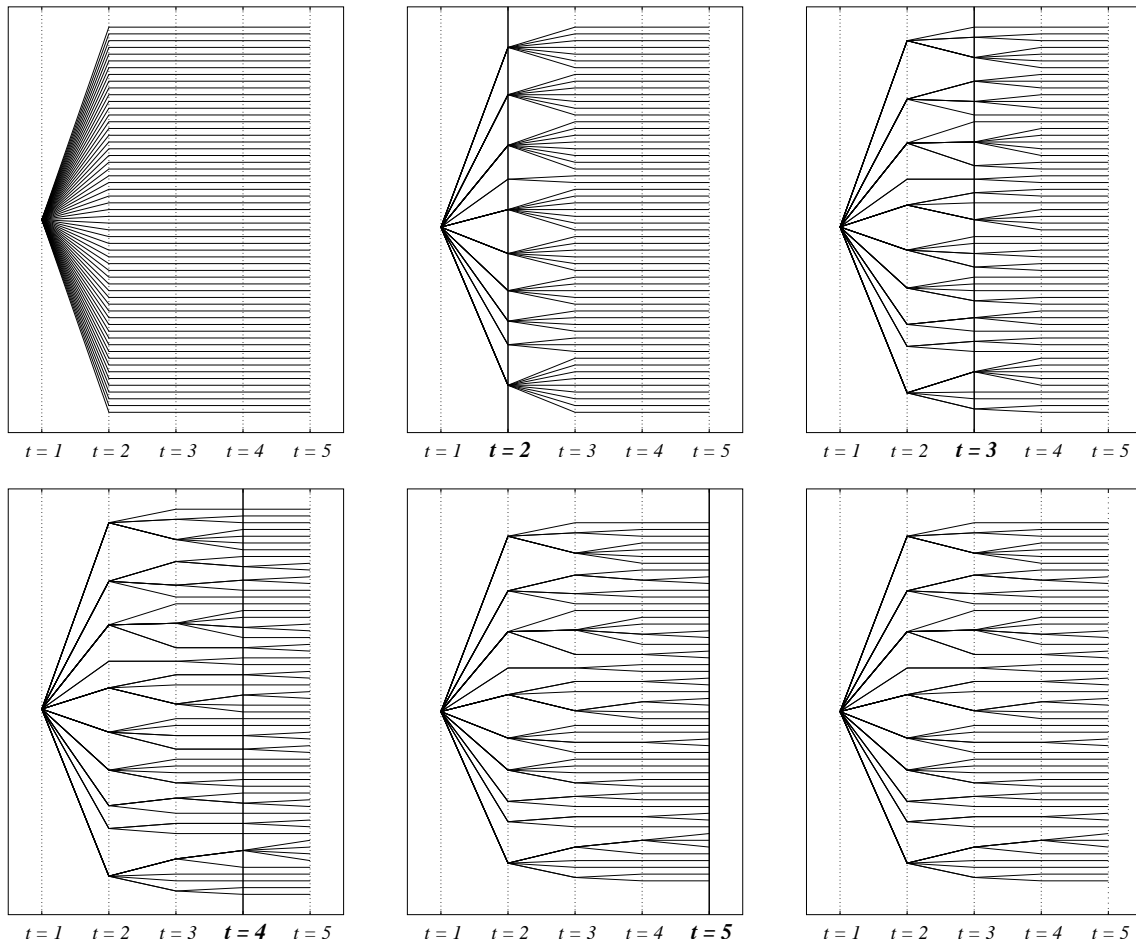


Bild 5: Illustration der schrittweisen Szenariobaumkonstruktion von Algorithmus 3.3 an einem Beispiel beginnend mit $N = 58$ (individuellen) Szenarien und $T = 5$.

Zuflüssen zu hydraulischen Kraftwerken zusammensetzen. Jede der Realisierungen ist mit derselben Wahrscheinlichkeit versehen. Beide Komponenten des stochastischen Prozesses treten als rechte Seiten in den linearen Restriktionen des stochastischen Optimierungsmodells auf. Der Zeitraum von zwei Jahren ist in drei Zeitperioden pro Tag unterteilt, die für sich jeweils Stunden ähnlichen Bedarfs zusammenfassen. Die Tabellen 1 und 2 zeigen einen Überblick über die wichtigsten relevanten Eingangsparameter. Die nur wochenweise gegebenen Werte für die Zuflüsse wurden gleichmäßig auf die zugehörigen Zeitperioden verteilt.

In unterschiedlichen Testreihen wurden Szenariobäume unter Verwendung des Algorithmus 3.3 für den Fall $r = 1$ generiert. Um ein einfaches Maß für die Genauigkeit der generierten Szenariobäume in Bezug auf die Ausgangsverteilung zu bekommen, werden relative Toleranzen

$$\varepsilon_{rel} := \frac{\varepsilon}{\varepsilon_{max}} \quad \text{bzw.} \quad \varepsilon_{t,rel} := \frac{\varepsilon_t}{\varepsilon_{max}} \quad (t = 2, \dots, T) \quad (14)$$

betrachtet. Der Wert für ε_{max} bezeichnet den kleinstmöglichen Verteilungsabstand (hier bzgl. l_1), der durch die Ausgangsverteilung und der Verteilung bestehend aus nur einem der Szena-

Tabelle 1: Unterteilung des Optimierungszeitraums von zwei Jahren.

Zufallsdaten	Unterteilungen	Zeitperioden
Elektrische Last	3 pro Tag	2 184
Wasser-Zufluss	wöchentlich	104

Tabelle 2: Dimension der Ausgangsdaten.

	Anzahl
Szenarien	456
Zeitperioden	2 184
Knoten	995 449

rien versehen mit der Wahrscheinlichkeit 1 erzielt werden kann. Da die in den Einzelschritten des Algorithmus durchgeführte Szenarioreduktion nur auf die jeweilige Zeitperiode ausgerichtet ist und damit alle Zeitperioden prinzipiell gleich behandelt werden, ist auch eine nahezu gleichmäßige Aufteilung der gesamten ε -Toleranz auf die jeweiligen Zeitstufen sinnvoll. Numerische Erfahrungen zeigen jedoch, dass eine gewisse Variabilität in Form einer linearen Festlegung der individuellen Parameter ε_t durch

$$\varepsilon_t = \frac{\varepsilon}{T} \left[2 \left(\frac{1-2q}{T} t + q \right) \right], \quad t = 2, \dots, T, \quad (15)$$

vorteilhaft sein kann. Der Parameter q ist im Intervall $[0, 1]$ frei wählbar und sorgt je nach Wahl dafür, dass eine linear absteigende ($q < \frac{1}{2}$) bzw. aufsteigende Folge ($q > \frac{1}{2}$) von Werten ε_t generiert wird. Damit hat der Parameter q entscheidenden Einfluss auf das Verzweigungsverhalten der entstehenden Szenariobäume. In den hier durchgeführten Berechnungen wurde schließlich $q = 0.35$ gewählt.

Tabelle 3 zeigt die Ergebnisse der Baumkonstruktion bei unterschiedlichen ε -Toleranzen (erste Spalte). Die zweite bzw. dritte Spalte vergleicht die Dimension der Eingangsdaten mit denen der konstruierten Bäume hinsichtlich Anzahl der Szenarien und Knoten. Die vorletzte Spalte enthält Angaben über die tatsächliche Anzahl der beobachteten Verzweigungsstufen in den resultierenden Szenariobäumen. Des Weiteren geben die Tabellen Aufschluss über die Rechenzeiten auf einem Standard-PC. Alle aufgelisteten Rechenzeiten beinhalten rund 100 Sekunden für die Bestimmung der Szenarienabstände, die jeweils vor der eigentlichen Ausführung des Algorithmus berechnet wurden. Trotz der großen Datenmengen und des langen Zeithorizonts erzeugt der Konstruktionsalgorithmus die Szenariobäume sehr schnell und das weitgehend unabhängig von der vorgegebenen Toleranz. Die gewählte Toleranz wirkt sich jedoch erwartungsgemäß auf die Anzahl der Verzweigungen und auch auf die Anzahl der Knoten aus. In Bild 6 ist die Struktur zweier konstruierter Szenariobäume mit unterschiedlicher Größe dargestellt.

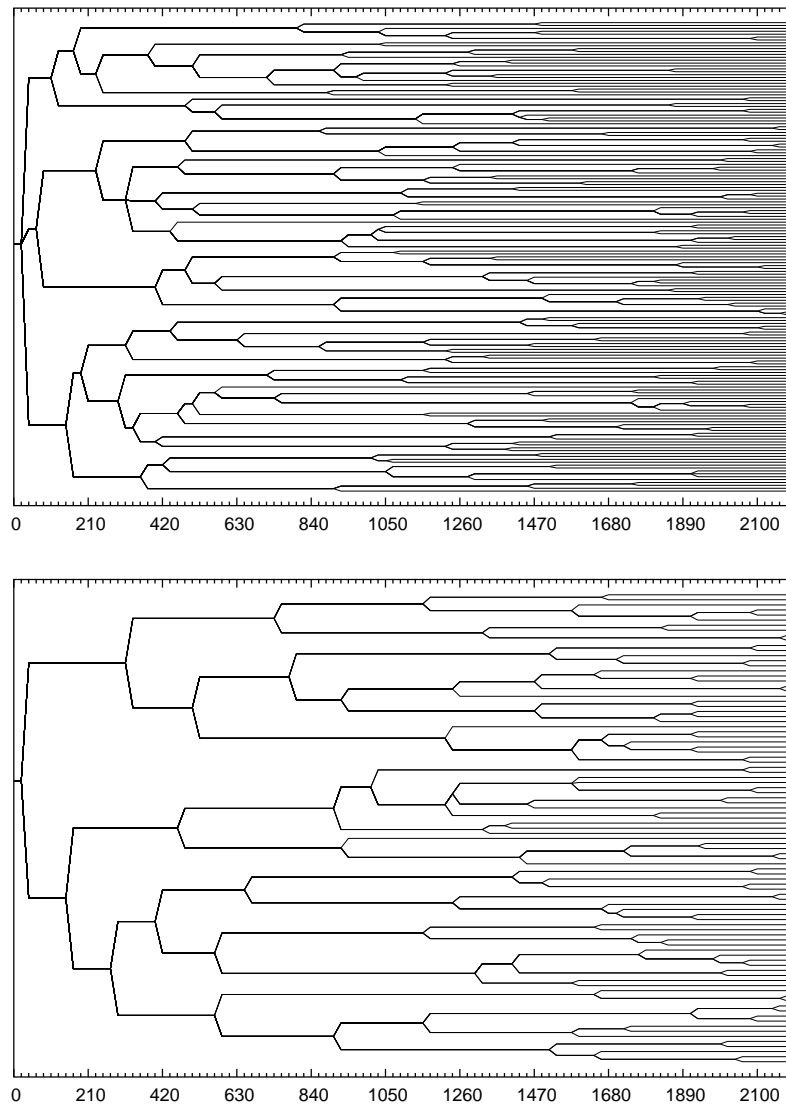


Bild 6: Mit dem Rückwärtsalgorithmus erzeugte Szenariobäume für $\varepsilon_{rel} = 0.2$ (oben) und $\varepsilon_{rel} = 0.5$ sowie wöchentlicher Verzweigung.

Abschließend sei angemerkt, dass innerhalb von Algorithmus 3.3 über die Wahl der Parameter ε_t eine sensible Einflussnahme auf Form und Größe der Szenariobäume gegeben ist. Die Ergebnisse der Anwendungsbeispiele spiegeln die Flexibilität der Algorithmen bei der Generierung unterschiedlicher Strukturen wider. Eine wichtige Bemerkung in diesem Zusammenhang betrifft auch die geeignete Wahl der gesamten Genauigkeit ε . Eine sehr hohe Genauigkeit, d.h. ein sehr kleiner Werte, mag in Bezug auf den Verteilungsabstand günstig sein, muss aber nicht zwangsläufig zu besseren Ergebnissen hinsichtlich der Approximation des Optimierungsproblems führen. Im Gegenteil, sind die Ausgangsszenarien fest vorgegeben und weisen sie formal keine Baumstruktur auf, würde eine zu hohe Genauigkeit, d.h. eine zu geringe ε -Toleranz, keinen Sinn bezüglich der Informationsstruktur des mehrstufigen Optimierungs-

Tabelle 3: Vorwärtskonstruktion mit wöchentlicher Verzweigung.

ε_{rel}	Szenarien		Knoten		Stufen	Zeit [s]
	initial	Baum	initial	Baum		
0.10	456	389	995 449	746 613	49	106.53
0.20	456	300	995 449	509 103	57	106.84
0.30	456	228	995 449	310 653	64	107.59
0.40	456	163	995 449	151 809	69	109.78
0.50	456	92	995 449	60 501	46	119.12

problems ergeben. Das Resultat wären Szenarien die wenig oder gar nicht verzweigen. Geht man von einem wirklich mehrstufigen Ausgangsproblem aus, d.h. von einem echten Zuwachs an Information über den Optimierungszeitraum, wäre dies eine schlechte Wahl. Das heißt, für ein endliche Menge von Ausgangsszenarien führt eine beliebige Verkleinerung der Toleranz bei gleichzeitig festgehaltener Szenarienzahl im Allgemeinen gar zur Verschlechterung der Approximationen. Ist also die Zahl der Szenarien von vornherein stark beschränkt, sollten die ε -Toleranzen innerhalb der Konstruktionsalgorithmen nicht zu klein gewählt werden, um dennoch brauchbare Resultate erzielen zu können (vgl. [7]).

Anwendern stehen die in dieser Arbeit vorgestellten Methoden zur Konstruktion und Reduktion von Szenariobäumen über das Software-Tool SCENRED zur Verfügung, welches frei mit der Modellierungssoftware GAMS (www.gams.com) erhältlich ist.

Danksagung

Diese Arbeit wurde im Rahmen des MATHEON-Projekts C7 „Mean-risk optimization of electricity production in liberalized markets“ und des Industrieprojekts „Scenario trees and risk“ mit der Firma EDF (Electricité de France) gefördert.

Literatur

- [1] P. Avella, A. Sassano, and I. Vasil'ev. Computational study of large-scale p-median problems. *Mathematical Programming*, 109:89–114, 2007.
- [2] M. Burger, B. Klar, A. Müller, and G. Schindlmayr. A spot market model for pricing derivatives in electricity markets. *Quantitative Finance*, 4:109–122, 2004.
- [3] L. Clewlow and C. Strickland. *Energy Derivatives: Pricing and Risk Management*. Lacima, London, 2000.
- [4] J. Dupačová, N. Gröwe-Kuska, and W. Römisch. Scenarios reduction in stochastic programming: An approach using probability metrics. *Mathematical Programming*, 95(A):493–511, 2003.
- [5] A. Eichhorn, W. Römisch, and I. Wegner. Mean-risk optimization of electricity portfolios using multiperiod polyhedral risk measures. In *IEEE St. Petersburg PowerTech Proceedings*, 2005.

- [6] B.T. Ewing, J.B. Kruse, and J.L. Schroeder. Time series analysis of wind speed with time-varying turbulence. Technical report, 2004. available at <http://www.ecu.edu/hazards/reports.htm>.
- [7] H. Heitsch. *Stabilität und Approximation stochastischer Optimierungsprobleme*. PhD thesis, Humboldt-Universität zu Berlin, 2007.
- [8] H. Heitsch and W. Römis. Scenario reduction algorithms in stochastic programming. *Computational Optimization and Applications*, 24:187–206, 2003.
- [9] H. Heitsch and W. Römis. A note on scenario reduction for two-stage stochastic programs. *Oper. Res. Lett.*, 35:731–738, 2007.
- [10] H. Heitsch and W. Römis. Scenario tree modeling for multistage stochastic programs. *Mathematical Programming*, to appear, 118:371–406, 2009.
- [11] R. Henrion, C. Küchler, and W. Römis. Discrepancy distances and scenario reduction in two-stage stochastic integer programming. *Journal of Industrial and Management Optimization*, 4:363–384, 2008.
- [12] R. Henrion, C. Küchler, and W. Römis. Scenario reduction in stochastic programming with respect to discrepancy distances. *Computational Optimization and Applications*, 43:67–93, 2009.
- [13] R. Hochreiter, G. Ch. Pflug, and D. Wozabal. Multi-stage stochastic electricity portfolio optimization in liberalized energy markets. In *System Modeling and Optimization*, pages 219–226. Springer, Boston, MA, USA, 2006.
- [14] S.T. Rachev. *Probability Metrics and the Stability of Stochastic Models*. Wiley, Chichester, 1991.
- [15] S.T. Rachev and W. Römis. Quantitative stability in stochastic programming: The method of probability metrics. *Mathematics of Operations Research*, 27:792–818, 2002.
- [16] S.T. Rachev and L. Rüschendorf. *Mass Transportation Problems*, volume I. Springer, Berlin, 1998.
- [17] H. K. Schmöller, T. Hartmann, I. Kruck, and H.-J. Haubrich. Modeling power price uncertainty for midterm generation planning. In A. Borghetti, C. A. Nucci, and M. Paolone, editors, *IEEE Bologna PowerTech Proceedings*, 2003.
- [18] R. Schultz. Rates of convergence in stochastic programs with complete integer recourse. *SIAM Journal of Optimization*, 6:1138–1152, 1996.
- [19] S. Sen, L. Yu, and T. Genc. A stochastic programming approach to power portfolio optimization. *Operations Research*, 54:55–72, 2006.
- [20] V. V. Vazirani. *Approximation Algorithms*. Springer, Berlin, 2001.
- [21] R. Weron, M. Bierbrauer, and S. Trück. Modeling electricity prices: jump diffusion and regime switching. *Physica A*, 336:39–48, 2004.