

# Constraint-Handling mit Supportvektor-Dekodern in der verteilten Optimierung

Jörg Bremer<sup>1</sup>

**Abstract:** Optimierungsansätze in dynamisch organisierten, kooperativen Systemen stehen oft vor dem Problem, nicht auf ein statisches Optimierungsmodell zurückgreifen zu können. Im Smart Grid muss bei der Koordination dezentraler Energieerzeugungsanlagen jeder Anlage ein Fahrplan aus dem eigenen, spezifischen Lösungsraum zugewiesen werden, um einen gewünschten Gesamtlastgang zu erzielen. Diese Arbeit entwickelt ein Verfahren, mit dem abstrakte Darstellungen von Lösungsräumen einschließlich beschränkender Nebenbedingungen ohne spezifisches Wissen über die individuelle Modellierung automatisiert zu einem gemeinsamen Optimierungsmodell integriert werden können. Durch einen Supportvektor-Dekoder werden zulässige Lösungen systematisch generiert, sodass eingesetzte Optimierungsverfahren weder über eigene Constraintbehandlung noch über analgenspezifische Modellierungskennnisse verfügen müssen.

## 1 Einleitung

Eine schrittweise Transformation von den heutigen, zentralen Markt- und Netzstrukturen in der Energieversorgung hin zu einem dezentralen Aufbau ist nicht nur Ziel aktueller politischer Entscheidungen [Eu09, De10] sondern auch dringende Notwendigkeit für eine erfolgreiche Integration regenerativer Stromerzeugung [Ra12, Bu08, Cr10].

Im Gegensatz zu dem relativ überschaubaren Planungsproblem für den Einsatz zentraler Großkraftwerke führt die stetige Integration dezentraler, individuell operierter Kleinerzeuger zu neuen Herausforderungen in der Einsatzplanung. Eine zunehmende Dezentralisierung der Netzsteuerung, bedingt durch die stetig wachsende Zahl an Erzeugern, führt zur Notwendigkeit dezentraler Algorithmen, um der wachsenden Komplexität entgegenzuwirken. Markterfordernisse lassen zudem die Entwicklung hin zu einem sich regelmäßig anpassenden System erwarten, bei dem sich die verteilten Anlagen selbsttätig je nach Marktlage produktorientiert zusammenfinden. Diese Dynamik macht aber auch die Zusammenarbeit von verschiedenen sich unbekanntenen Anlagen erforderlich. Ohne zentrale Steuerungseinheit (mit zentraler, statischer Modellierung) müssen im Rahmen von Optimierungsansätzen die individuellen Suchräume mit den verschiedenen alternativen Betriebsmöglichkeiten der beteiligten Anlagen ad hoc und automatisiert zu einem Modell des aktuell zu lösenden Optimierungsproblems integriert werden können.

Die Entwicklung einer Planung für nicht a priori feststehende Anlagenverbünde erfordert eine Modellabstraktion von den konkreten Energieanlagen welche sich einerseits zur Laufzeit zu einem Optimierungsmodell integrieren lässt, ohne spezifische Informationen über technische Nebenbedingungen inkorporieren zu müssen. Andererseits wird ein System zur

---

<sup>1</sup> Carl von Ossietzky Universität Oldenburg, joerg.bremer@uni-oldenburg.de

systematischen Erzeugung zulässiger Lösungskandidaten benötigt, da die modellspezifische Implementierung eines Planungsalgorithmus nicht möglich ist, wenn zur Laufzeit die Anlagen noch nicht bekannt sind. Dennoch muss sichergestellt sein, dass jede Anlage den zugewiesenen Fahrplan technisch umsetzen kann. Hierfür war bisher kein Ansatz bekannt.

In [Br15] wurde zu diesem Zweck ein Ansatz basierend auf einer Supportvektor-Beschreibung von Fahrplanräumen entwickelt. Ziel der Arbeit war die Entwicklung eines hybriden Ansatzes für die Integration modellunabhängiger Beschreibungen beliebiger elektrischer Anlagen mit verschiedenen Optimierungsverfahren. Hierzu wurde ein Dekoder entwickelt, welcher auf einem Supportvektor-Modell zur Beschreibung möglichen Anlagenhaltens aufsetzt. Die Hybridisierung mit verschiedenen Optimierungsansätzen ergibt sich durch eine Reformulierung der Problemstellung, welche durch den entwickelten Dekoder das Optimierungsproblem dadurch vereinfacht, dass alle technischen Modelle der Anlagen einschließlich der individuellen Constraints abstrakt gekapselt werden. Der Lösungsalgorithmus selbst braucht daher keine Nebenbedingungen mehr zu beachten. Hierdurch wird eine constraintfreie Formulierung möglich, wodurch verschiedene Standardansätze zur Optimierung ohne spezielle Anpassung an das Problem und ohne ansatzspezifische Constraintmodellierung eingesetzt werden können.

Im Folgenden wird zunächst das allgemeine Konzept hergeleitet bevor ein Modell zur Abstraktion der Fahrplanräume vorgestellt wird. Um für prinzipiell beliebige Planungsalgorithmen hieraus systematisch zulässige Lösungen generieren zu können wird anschließend aus dem Modell ein Dekoder zur Lösungserzeugung hergeleitet. Die praktische Anwendbarkeit wird abschließend an einem Beispiel basierend auf einer Simulated Annealing Optimierung demonstriert.

## 2 Problemstellung und konzeptioneller Ansatz

Sei ein Fahrplan definiert als Vektor  $\mathbf{x} \in \mathbb{R}^d$ , wobei das  $i$ -te Element von  $\mathbf{x}$  die mittlere Leistung über das Zeitintervall  $t_i$  kennzeichnet. Nun ist in der Praxis nicht jeder Fahrplan für jede Anlage umsetzbar. Dem stehen sowohl technisch begründete harte (max. Leistungsgradienten, Speicherfüllstände, etc.) als auch weiche Nebenbedingungen (etwa wirtschaftlich begründet, einzuhaltende Komfortgrößen, o. ä. ) entgegen. Exemplarisch für den Fall des prädiktiven Scheduling – bei dem durch vorausschauende Planung von Energieerzeugung (ggf. unter Einbezug steuerbaren Verbrauchs) Fahrpläne für eine Gruppe von Anlagen erstellt werden, sodass in Summe möglichst genau ein gewünschter Gesamtlastgang erreicht wird – ließe sich eine mögliche Problemstellung wie folgt formalisieren:

$$\delta \left( \sum_{i=1}^n \mathbf{x}_i, \zeta \right) \rightarrow \min, \text{ wobei } \mathbf{x}_i \in \mathcal{F}_i \ 1 < i \leq n. \quad (1)$$

Zu minimieren ist also die Distanz  $\delta$  (z. B. Euklidischer Abstand) zwischen der Summe aller Fahrpläne  $\mathbf{x}_i$  aller  $n$  Anlagen und der Zielvorgabe  $\zeta \in \mathbb{R}^d$  für  $d$  zukünftiger Zeitintervalle, wobei jeder Fahrplan aus dem Unterraum  $\mathcal{F}_i$  der für Anlage  $i$  zulässigen Fahrpläne stammen muss.  $\mathcal{F}_i$  ist üblicherweise eine anlagenspezifische, nichtlineare Region welche auch vom jeweiligen Betriebszustand zu Beginn des Zeitraums und Prognosen (z. B.

Wetter) abhängig ist und daher nicht statisch modelliert werden kann. Aus Gründen der Gleichbehandlung unterschiedlicher Anlagen bietet es sich an, alle Fahrpläne zu skalieren, sodass im Folgenden o.B.d.A. nur noch Fahrpläne  $\mathbf{x} \in \mathcal{F}_{[0,1]} = [0, 1]^d$  betrachtet werden. Mittels eines geeigneten Samplingverfahrens kann eine Menge zulässiger Fahrpläne als Trainingsmenge leicht aus einem gegebenen Simulationsmodell einer Anlage generiert werden [Br15]. Diese Menge an Beispielfahrplänen, welche nach obiger Definition als Vektoren aus  $\mathcal{F}_{[0,1]}$  aufgefasst werden, soll nunmehr als Schablone für die geometrische Region  $\mathcal{F}_{[0,1]}$  im Fahrplanraum aufgefasst werden, welche genau nur die zulässigen Fahrpläne enthält, die ohne Verletzung von Nebenbedingungen umgesetzt werden können.

Die eingeschränkten Suchräume jeder Anlage können mit einem Supportvektor-Modell kodiert werden, das von jedweder individueller, mathematischer Beschreibung der Anlagenfähigkeiten und -constraints abstrahiert. Solche Stützvektormethoden (engl. support vector, SV) für die Datenmodellierung nutzen zur Beschreibung eines Datenraums eine Trainingsmenge mit Beispieldaten aus diesem Raum und wählen hieraus eine Teilmenge (die Supportvektoren) mit denen eine die Daten einschließende Hüllkurve (oder trennende Hyperebene) berechnet werden kann [TD04, Be01]. Constraints sind in einem solchen Modell nicht mehr mathematisch beschrieben, sondern sind lediglich noch implizit durch ihre Auswirkungen vertreten. Beschrieben ist ein Unterraum des Vektorraums aller Fahrpläne, welcher durch die Constraints begrenzt ist, jedoch beschrieben durch eine Menge von Beispielelementen aus diesem Unterraum und nicht länger durch die Constraints selbst. Im Ergebnis ergibt sich hieraus ein Blackbox-Modell welches ersatzweise für das Anlagenmodell entscheiden kann, ob ein beliebiger gegebener Fahrplan zulässig ist oder nicht. Eine systematische Erzeugung zulässiger Fahrpläne ergibt sich hieraus noch nicht. Ziel der Arbeit war aber die Entwicklung eines hybriden Ansatzes welcher die Integration modellunabhängiger Beschreibungen beliebiger elektrischer Anlagen mit verschiedenen Optimierungsverfahren ermöglicht, um eine systematische und zielgerichtete Navigation im Lösungsraum zu ermöglichen ohne dass dieser vollständig beschrieben vorliegen muss.

Hierzu wurde basierend auf dem SV-Modell ein Dekoder entwickelt. Ein Dekoder stellt eine Beziehung zwischen einer gültigen Lösung und einer Dekoderlösung her und stellt somit eine Konstruktionsvorschrift für gültige Lösungen dar. Die Hybridisierung mit verschiedenen Optimierungsansätzen ergibt sich durch die Reformulierung der Problemstellung. Bei dieser Umformulierung wird durch den entwickelten Dekoder das Optimierungsproblem dadurch vereinfacht, dass alle technischen Modelle der Anlagen einschließlich der individuellen Constraints in Dekodern gekapselt werden. Der Lösungsalgorithmus selbst braucht daher keine Nebenbedingungen mehr zu beachten. Hierdurch wird eine constraintfreie Formulierung möglich, wodurch verschiedene Standardansätze zur Optimierung ohne ansatzspezifische Constraintmodellierung eingesetzt werden können.

### 3 Surrogatmodell und Dekoder

Ausgehend von einer gegebenen Menge an zulässigen Fahrplänen wird ein Modell durch Erlernen einer Entscheidungsgrenze um die Fahrplanmenge herum umgesetzt. Durch Verwenden einer so genannten Support Vector Domain Description (SVDD) ist es nicht er-

forderlich, dass der Unterraum der zulässigen Fahrpläne zusammenhängend, konvex oder linear begrenzt ist. Zunächst soll kurz die Erstellung dieses Modells erläutert werden. Die nachfolgenden Darstellungen orientieren sich u. a. an [TD04, Be01].

Sei  $\{\mathbf{x}_i\}_n \subseteq \mathcal{X} \subset \mathbb{R}^d$  eine Menge von  $n$  gegebenen Datenpunkten aus dem Fahrplanraum mit der Dimension  $d$ . Sei ferner  $\Phi: \mathcal{X} \rightarrow \mathcal{H}$ ,  $\mathbf{x} \mapsto \Phi(\mathbf{x})$  eine nicht-lineare Abbildung (Featuremap) von Datenpunkten  $\mathbf{x}_i$  aus  $\mathbb{R}^d$  in einen hoch- (möglicherweise unendlich-)dimensionalen Featureraum  $\mathcal{H}$ . Zu jedem positiv semidefiniten Kern  $k$  gehört ein assoziierter Reproducing Kernel Hilbert Space RKHS  $\mathcal{H}$ . Dies ist der durch die Featuremap erzeugte Featureraum. Eine ausführliche Diskussion findet sich beispielsweise in [STC04]. Nun beschreibt  $\|\Phi(\mathbf{x}_i) - \mathbf{a}\|^2 \leq R^2 \forall i$  eine Kugel in  $\mathcal{H}$  welche die Bilder aller Datenpunkte  $\{\Phi(\mathbf{x}_i)\}_n$  enthält. Hierbei bezeichne  $\|\cdot\|$  den Euklidischen Abstand und  $\mathbf{a}$  den Mittelpunkt dieser Hypersphäre.

Abhängig von der genauen Ausgestaltung der Featuremap  $\Phi$  sind unterschiedlich große Sphären denkbar, welche  $\{\Phi(\mathbf{x}_i)\}_n$  jeweils einschließen. Um die kleinste Sphäre zu finden lässt sich nach Einführung von Schlupfvariablen  $\beta$  und diversen Umformungen [TD04, Br15] ein Optimierungsproblem definieren, welches im Kern eine Belegung für  $\beta$  für nachfolgende Gleichung (2) liefert. Hierbei kann auf das Theorem von Mercer (Kern-Trick) zurückgegriffen werden, um die Berechnung von Skalarprodukten in  $\mathcal{H}$  durch geeignete Kernfunktionen (hier: Gauß-Kern  $k(\mathbf{x}_i, \mathbf{x}_j) = e^{-\frac{1}{2\sigma^2}\|\mathbf{x}_i - \mathbf{x}_j\|^2}$ ) direkt in  $\mathbb{R}^d$  durchzuführen [Sc99]:  $\Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j) = k(\mathbf{x}_i, \mathbf{x}_j)$ .

$$R^2(\mathbf{z}) = 1 - 2 \sum_i \beta_i k(\mathbf{x}_i, \mathbf{z}) + \sum_{i,j} \beta_i \beta_j k(\mathbf{x}_i, \mathbf{x}_j). \quad (2)$$

Gleichung (2) ermöglicht die Berechnung des Abstandes eines Bildes  $\Phi(\mathbf{x})$  vom Zentrum  $\mathbf{a}$  der Hypersphäre  $\mathcal{S}$  ohne  $\Phi$  oder den zugehörigen RKHS zu kennen. Weiteres Ergebnis des Trainings sind diejenigen Datenpunkte, welche exakt auf die Oberfläche der Sphäre abgebildet werden. Sie definieren mittels (2) den Radius der Sphäre und erlaubt somit eine Entscheidung bezüglich der Zugehörigkeit neuer Datenpunkte zur kodierten Datenmenge. Ein beliebiger Vektor gehört zur Positivklasse genau dann, wenn er unter  $\Phi$  in die Kugel oder auf die Oberfläche abgebildet wird; wenn also sein Bild einen Abstand von  $\mathbf{a}$  kleiner oder gleich des Radius der Kugel hat:  $\mathcal{X} = \{\mathbf{x} \mid R(\mathbf{x}) \leq R_{\mathcal{S}}\}$ . Das auf SVDD basierende Suchraummodell besteht somit aus den folgenden Komponenten [BRS11]:

1. Eine Menge  $SV = \{\mathbf{s}_i \in \mathcal{X} \in [0, 1]^d \mid \beta_i \neq 0\}$  von Supportvektoren. Jeder Supportvektor  $\mathbf{s} \in SV$  ist ein skaliertes Fahrplan aus der Trainingsmenge der Fahrpläne, die durch das Sampling aus dem Einheitensimulationsmodell erzeugt wurden.
2. Ein Gewichtsvektor  $\mathbf{w} = (\beta_1, \dots, \beta_n) \quad \forall \beta_i \neq 0$ , der nur noch die Gewichtungen für die Supportvektoren enthält. Die Gewichte aller anderen Fahrpläne aus der Trainingsmenge sind 0, wesentlichen sowohl die zugehörigen Fahrpläne als auch deren Gewichte zur Berechnung von (2) nicht benötigt werden.
3. Anlagenparameter, die zur Skalierung der Fahrpläne benötigt werden. Dies sind in der Regel die minimale und maximale Wirkleistung der Anlage.

4. Die Abstandsfunktion  $R^2(\mathbf{x}) = 1 - 2\sum_i \mathbf{w}_i k(\mathbf{s}_i, \mathbf{x}) + \sum_{i,j} \mathbf{w}_i \mathbf{w}_j k(\mathbf{s}_i, \mathbf{s}_j)$ , mit  $1 \leq i, j \leq |SV|$ , welche die Klassifikatorausgabe des Modells berechnet. Der Suchraum, der durch dieses Modell kodiert wird ist somit definiert als  $\mathcal{X} \approx \{\mathbf{x} | R(\mathbf{x}) \leq R_{\mathcal{S}}\}$ .

Der Gewichtungsvektor  $\mathbf{w}$  stellt eine reduzierte Version der Lagrangemultiplikatoren dar, welcher nur noch die relevanten Werte größer 0 enthält. Somit lässt sich die Darstellung des Bildes eines Fahrplans  $\Phi(\mathbf{x}) \in \mathcal{H}$ , welches durch die Expansion  $\Psi_x = \sum_{i=1}^n \beta_i \phi(\mathbf{x}_i)$  repräsentiert wird vereinfachen zu  $\Psi_x = \sum_{i=1}^{\ell} \mathbf{w}_i \phi(\mathbf{s}_i)$ . Die Zahl der Supportvektoren  $\ell$  ist deutlich kleiner als die Anzahl  $n$  der Fahrpläne in der Trainingsmenge. Für die Umsetzung des Dekoders findet die folgende Abbildung Verwendung:

$$\gamma: [0, 1]^d \rightarrow \mathcal{F}_{[0,1]} \subseteq [0, 1]^d, \mathbf{x} \mapsto \gamma(\mathbf{x}), \quad (3)$$

welche den Einheitshyperwürfel  $[0, 1]^d$  auf den Unterraum der zulässigen Fahrpläne  $\mathcal{F}_{[0,1]}$  abbildet. Konzeptionell werden für die Konstruktion bekannte geometrische Zusammenhänge der Sphäre ausgenutzt. Drei Schritte sind zu bewerkstelligen (vgl. Abbildung 1). Ein gegebener (ungültiger) Fahrplan  $\mathbf{x}$  müsste abgebildet werden mit  $\Phi(\mathbf{x})$ . Da  $\Phi$  nicht bekannt ist, muss eine alternative Abbildung  $\hat{\Phi}_{\ell}$  als Näherung verwendet werden [Sc99]. Das Bild von  $\mathbf{x}$  muss so verändert werden, dass es das Bild eines gültigen Fahrplans darstellt. Hierzu muss es mit einer geeigneten Abbildung  $\Gamma_a$  in Richtung des Mittelpunktes  $a$  der Sphäre verschoben werden [Pa07]. Schließlich muss von dem verschobenen Bild das Urbild gefunden werden, welches den nunmehr gültigen Fahrplan repräsentiert. Dies ist nicht direkt, sondern nur mittels einer Näherung  $\Phi_{\ell}^{-1}$  möglich (vgl. z. B. [KT04]).

Anstatt also eine direkte Abbildung nach  $\mathcal{F}_{[0,1]}$  zu suchen, wird ein Weg über den durch den Kern  $k$  induzierten RKHS eingeschlagen. Ausgangspunkt ist das SV-Modell. An dieser Stelle sei davon ausgegangen, dass dieser Schritt wie zuvor erläutert bereits durchgeführt wurde. Dies sind die einzigen Informationen, die für die Konstruktion des Dekoders benötigt werden. Im Folgenden werden die einzelnen Schritte näher erläutert.

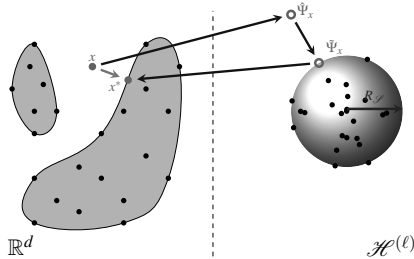


Abb. 1: Prinzipische Skizze des Supportvektor-Dekoderansatzes.

**Schritt 1: Abbildung in den Supportvektor RKHS  $\mathcal{H}^{(\ell)}$  mittels empirischer Kernabbildung** Das SVDD-Training findet auf einer endlichen Trainingsmenge statt. SV-Algorithmen arbeiten daher in einem Unterraum des Feature-raums, der durch die Bilder der Trainingsvektoren aufgespannt wird. Dieser Unterraum kann durch eine empirische

Kernabbildung in einen Euklidischen Raum eingebettet werden, sodass paarweise Abstände und Winkel zwischen den Vektoren erhalten bleiben [XSA05]. Analog kann der Unterraum betrachtet werden, der durch die Bilder der Supportvektoren aufgespannt wird. Abbildung

$$\hat{\Phi}_\ell: \mathbb{R}^d \rightarrow \mathcal{H}^{(\ell)}, \mathbf{x} \mapsto \mathbf{K}^{-\frac{1}{2}}(k(\mathbf{s}_1, \mathbf{x}), \dots, k(\mathbf{s}_\ell, \mathbf{x})), \quad (4)$$

bildet beliebige Punkte  $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$  so nach  $\mathcal{H}^{(\ell)}$  ab, dass gilt [Sc99]:  $k(\mathbf{x}, \mathbf{y}) = \hat{\Phi}_\ell(\mathbf{x}) \cdot \hat{\Phi}_\ell(\mathbf{y})$ .  $\mathbf{K}$  ist hierbei die Kern Gram-Matrix  $\mathbf{K}_{ij} = k(\mathbf{s}_i, \mathbf{s}_j)$ . Mit  $\hat{\Phi}_\ell$  ist es einfach möglich, beliebige Punkte (insbesondere aus  $[0, 1]^d \subseteq \mathbb{R}^d$ ) in einen  $\ell$ -dimensionalen Raum  $\mathcal{H}^{(\ell)}$  abzubilden, der eine  $\ell$ -dimensionale Projektion der Hypersphäre enthält. Auch hier gilt: die geometrische Struktur bleibt erhalten [XSA05], d.h. Punkte aus  $\mathcal{F}_{[0,1]}$  werden in die projizierte Hypersphäre abgebildet, Punkte von außerhalb liegen auch in  $\mathcal{H}^{(\ell)}$  außerhalb der Sphäre. Das in diesem Schritt erzeugte Bild  $\hat{\Psi}_x$  muss im nächsten Schritt modifiziert werden, um zu einem Bild eines gültigen Fahrplans zu werden.

**Schritt 2: Rejustierung im RKHS** Das Bild der mittels SVDD kodierten Region wird im RKHS  $\mathcal{H}$  als Sphäre  $\mathcal{S}$  mit Mittelpunkt  $\mathbf{a}$  und Radius  $R_{\mathcal{S}}$  dargestellt. Punkte außerhalb sind keine Bilder aus  $\mathcal{X}$ . Zulässige Fahrpläne werden unter  $\Phi$  in die Sphäre abgebildet (oder auf die Oberfläche). Durch die Verwendung eines Gaußkerns bildet  $\Phi$  genaugenommen in eine maximal  $n$ -dimensionale Mannigfaltigkeit (mit  $n$ : Anzahl Fahrpläne) ab, welche eingebettet ist in  $\mathcal{H}$ . Diese Mannigfaltigkeit wird aufgespannt von den Bildern der Stichprobe. Dies gilt analog auch für die niedriger dimensional, von den  $\ell$  Supportvektoren aufgespannten Unterraum und der niedriger ( $\ell$ -) dimensionalen Projektion der Sphäre darin, wenn  $\hat{\Phi}_\ell$  (Gleichung 4) für die Abbildung verwendet wird. Ziel dieses Schritts ist es, Punkte von außerhalb  $\mathcal{F}_{[0,1]}$  auf möglichst direktem Weg in den zulässigen Bereich zu ziehen und so gültig zu machen. Zu diesem Zweck kann beispielsweise die folgende Gleichung genutzt werden:

$$\tilde{\Psi}_x = \mathbf{a} + (\hat{\Psi}_x - \mathbf{a}) \cdot R_{\mathcal{S}} \cdot R_x^{-1} \quad (5)$$

wie sie in ähnlicher Form auch beim Entrauschen (dort allerdings durch geodätische Projektion mit Hilfe einer Bestimmung der trennenden Hyperebene) Verwendung findet (vgl. beispielsweise [Pa07]). Mit (5) kann das Bild eines ungültigen Fahrplans auf die Oberfläche von  $\mathcal{S}$  gezogen werden. Verbesserte Verfahren, welche auch das innere der Sphäre ausnutzen finden sich in [Br15]. Im Anschluss an diese Rejustierung liegt das modifizierte Bild  $\tilde{\Psi}_x$  des Ausgangspunktes  $\mathbf{x}$  in Form eines veränderten Gewichtungsvektors  $\tilde{\mathbf{w}}^{\Gamma_a}$  vor und repräsentiert bereits das Bild eines Punktes aus  $\mathcal{F}_{[0,1]}$ .

**Schritt 3: Urbildbestimmung** Als letzter Schritt muss noch das Urbild des Bildes  $\tilde{\Psi}_x$  aus Schritt 2 bestimmt werden, um einen Punkt aus  $\mathcal{F}_{[0,1]}$  zu erhalten. Ein Problem hierbei ist, dass nicht jeder Punkt aus der linearen Hülle  $\mathcal{L}(\Phi)$  das Bild eines unter  $\Phi$  abgebildeten Punktes ist [Sc99]. Stattdessen wird derjenige Punkt  $\mathbf{y}^* \in [0, 1]^d$  bestimmt, für den der Abstand zum tatsächlichen Urbild  $\|\mathbf{y}^* - \Phi^{-1}(\mathbf{y})\|$  minimal ist. Dieser Umstand kann auch formuliert werden als  $\mathbf{y}^*$  ist approximiertes Urbild von  $\mathbf{y}$ , falls  $\|\Psi_y - \Phi(\mathbf{y}^*)\|$  klein

ist [Sc99], wobei die Größenordnung von *klein* vom konkreten Problem abhängt [Sc99]. Es bietet sich an, ein Näherungsverfahren zu verwenden [Mi99], wie es bzw. für die Verwendung von Gaußkernen vorgeschlagen wird.

$$\mathbf{x}_{n+1}^* = \frac{\sum_{i=1}^{\ell} (\tilde{\mathbf{w}}_i^{\Gamma_a} e^{-\|\mathbf{s}_i - \mathbf{x}_n^*\|^2 / 2\sigma^2} \mathbf{s}_i)}{\sum_{i=1}^{\ell} (\tilde{\mathbf{w}}_i^{\Gamma_a} e^{-\|\mathbf{s}_i - \mathbf{x}_n^*\|^2 / 2\sigma^2})}. \quad (6)$$

Mit Ausführung dieses dritten Schrittes ist das Ziel erreicht: Ein ursprünglich ungültiger Punkt  $\mathbf{x}$  wurde abgebildet auf einen nunmehr gültigen Punkt  $\mathbf{x}^* = \gamma(\mathbf{x})$ , wobei sich  $\gamma$  als  $\gamma = \Phi_{\ell}^{-1} \circ \Gamma_a \circ \hat{\Phi}_{\ell}$  zusammensetzt. Damit ist der Dekoder fertig konzipiert.

In vielen Optimierungsszenarien sind Lösungskandidaten mit unterschiedlichen Kennzahlen zur Bewertung hinsichtlich verschiedener Zielfunktionen verbunden. In klassischen Mehrziel-Optimierungsszenarien ist normalerweise eine Annotation von Lösungskandidaten mit Kennzahlen zur Bewertung hinsichtlich verschiedener Kriterien nicht notwendig. Dort liegen die Bewertungskriterien direkt als mathematische Formulierung in Form eines Satzes von Zielfunktionen vor. Ein solches Vorgehen ist in dem hier vorgeschlagenen Ansatz bei dem die Optimierung auf einem Satz von dezentral erstellten Suchraummodellen mit Dekoder arbeitet nicht immer sinnvoll möglich. Eine Erweiterung des Fahrplanvektors mit weiteren Elementen, welche die jeweilige Bewertung des Fahrplans hinsichtlich verschiedener Kriterien darstellen, ist jedoch einfach möglich. Auf diese Weise entsteht aus jedem Fahrplan ein Featurevektor, der sowohl eine Menge von Leistungswerten als auch Bewertungskennzahlen enthält. Der Vorteil bei diesem Ansatz ist, dass alle Verfahren für die Erstellung des Suchraummodells und des Dekoders unverändert verwendet werden können und das verwendete Optimierungsverfahren keine Bewertung von Lösungskandidaten integrieren muss, da diese Bewertungen bereits vom Dekoder mitgeliefert werden können. Eine ausführliche Diskussion dieser Erweiterung findet sich z. B. in [Br15].

## 4 Ergebnisse

Schließlich soll noch die Anwendung des Ansatzes an einem konkreten Beispiel betrachtet werden. Hierfür soll in einem Verbund von 100 BHKW der Betrieb jedes BHKWs im Rahmen eines prädiktiven Scheduling so angepasst werden, dass in Summe möglichst genau ein vorgegebener Gesamtlastgang nachgefahren wird. Als Optimierungsverfahren kommt Simulated Annealing (SA) [KE95] zum Einsatz; um sicherzustellen, dass alle gefundenen Fahrpläne auch von den jeweiligen Anlagen umgesetzt werden können, wird für jede Anlage ein Dekoder trainiert. Die Trainingsmengen wurden mit Hilfe eines Simulationsmodells für die BHKW (einschließlich angeschlossener Wärmespeicher und simuliertem Wärmebedarf eines Einfamilienhauses) mit folgenden Nebenbedingungen für den Betrieb erstellt: neben der jederzeitigen Deckung des Wärmestroms zum Ausgleich des Wärmeverlusts, sind Temperaturgrenzen des Wärmespeichers, die Grenzen der Leistungsvariation des Motors und Minimallaufzeiten zu beachten. Das Optimierungsproblem aus Gleichung (1) lässt sich durch den Einsatz von Dekodern zunächst wie folgt vereinfachen.

$$\delta \left( \sum_{i=1}^d r_i \circ \gamma_i(\mathbf{x}'_i), \zeta \right) \rightarrow \min, \quad (7)$$

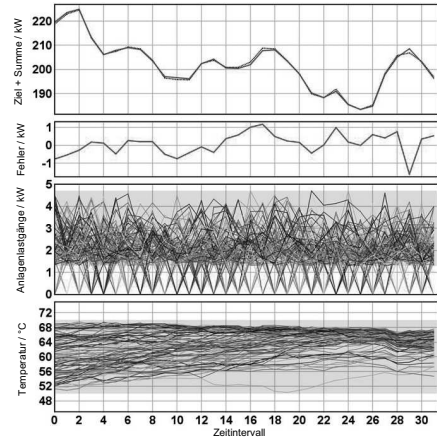
Hierbei stellt  $\mathbf{x}'_i$  als Genotyp die interne Repräsentation eines Fahrplans für BHKW  $i$  dar,  $\gamma_i(\mathbf{x}'_i)$  ist eine für Anlage  $i$  zulässige Dekoderlösung. Nebenbedingungen gibt es in dieser Formulierung keine mehr. Ein Simulated Annealing Ansatz (Abbildung 2(a)) kann also fast unverändert verwendet werden. Intern repräsentiert eine Matrix  $\mathbf{X}$  die Menge der Fahrpläne aus  $[0, 1]^d$  ohne Beachtung von Nebenbedingungen; skaliert auf  $[0, 1]$ . Daher enthält Gleichung (7) auch noch den Faktor  $\mathbf{r}$  zur elementweisen Skalierung. Eine parallel geführte zweite Repräsentation  $\mathbf{M}$  enthält alle zeilenweise mit den zugehörigen Dekodern abgebildeten Fahrpläne. Nach Mutation eines Fahrplans in  $\mathbf{X}$  muss auch nur dieser mit seinem Dekoder neu auf einen zulässigen Fahrplan abgebildet werden. Alle Lösungsbewertungen erfolgen auf  $\mathbf{M}$ .

```

 $\mathbf{X}_{ij} \leftarrow \mathbf{x}_i \sim U(0, 1)^d, 1 \leq i \leq n$ 
 $\mathbf{M}_{ij} \leftarrow \gamma_i(\mathbf{X}_i), 1 \leq i \leq n$ 
 $\vartheta \leftarrow \vartheta_{start}$ 
while  $\vartheta < \vartheta_{min}$  do
  choose random  $k; 1 \leq k \leq n$ 
   $\mathbf{x}^* \leftarrow \mathbf{X}_k$ 
  mutate( $\mathbf{x}^*$ )
   $\mathbf{M}^* \leftarrow \mathbf{M}; \mathbf{M}^*_k \leftarrow \gamma_k(\mathbf{x}^*)$ 
  if  $e^{-\frac{E(\mathbf{M}^*) - E(\mathbf{M})}{T}} > r \sim U(0, 1)$  then
     $\mathbf{M} \leftarrow \mathbf{M}^*; \mathbf{X}_k \leftarrow \mathbf{x}^*$ 
  end if
   $T \leftarrow \text{cooling}(T)$ 
end while

```

(a)



(b)

Abb. 2: Simulated Annealing mit Dekoder und Beispielergbnis.

Abbildung 2(b) zeigt exemplarisch eine Lösung mit dem Ziel- und Summenfahrplan und dem verbliebenen Restfehler in den beiden oberen Charts. Die beiden unteren Charts zeigen am Beispiel von Leistungsvariationsbereich und erlaubtem Speichertemperaturbereich (jeweils grau hinterlegt), dass alle Constraints eingehalten wurden, obwohl das Optimierungsverfahren selbst keinerlei integriertes Anlagenmodell beinhaltet oder Informationen über die Betriebsmöglichkeiten der BHKW integrieren musste.

## 5 Zusammenfassung

Ziel war es, eine abstrakte Beschreibung von Suchräumen bzw. von beschränkenden Constraints zu erarbeiten, die eine abstrakte Formulierung von Planungsproblemen ermöglicht, bei der nicht a priori die Menge der Anlagen (und damit auch nicht deren Modellierung) bekannt sein muss. Es ist nunmehr möglich, die zulässigen Handlungsalternativen verschiedener Akteure einer Gruppe auf abstrakter Ebene so zu erfassen, dass diese ad hoc und insbesondere automatisiert zu einem Optimierungsmodell für ein kombinatorisches Problem integriert werden können, bei dem verschiedene Handlungsoptionen der Einzel-



nen sich zu einer gewünschten Gesamthandlungsweise ergänzen. Zu diesem Zweck wurde eine Prozesskette erarbeitet, die eine Modellierung der Suchräume auf abstrakter, geometrischer Ebene vornimmt und ein System bereitstellt, welches eine systematische Bewegung in diesem Suchraum ermöglicht. Zentrales Element ist ein geometrisches Modell der individuellen Suchräume basierend auf einem 1-Klassen-Klassifikator. Realisierbare Fahrpläne können unabhängig vom Anlagentyp erfasst und angesprochen werden. Ein integrierendes Suchverfahren muss die zugrundeliegende Systematik (Modellierung sowie einschränkende Constraints) nicht kennen, da aufgrund des abstrakten Modells alle Anlagentypen gleich behandelt werden können. Die Integration kann automatisiert über einen algorithmisch ableitbaren Dekoder erfolgen, der eine einheitliche Navigation innerhalb der zulässigen Suchräume erlaubt.

Auch wenn im Rahmen des zur Evaluation verwendeten Anwendungsfalls der Wirkleistungsplanung im Smart Grid beinahe alle Teilaspekte anwendungsspezifisch umgesetzt wurden, ist das zugrundeliegende Konzept dennoch problemoffen und auf andere Domänen leicht anpassbar. Der Bedarf an intelligenten, selbsttätig arbeitenden Methoden zur Integration einer stetig wachsenden Zahl dezentraler, schwer prognostizierbarer Stromerzeuger in die Planung und Steuerung der jederzeitigen Abstimmung zwischen elektrischer Erzeugung und Verbrauch wird in Zukunft weiter zunehmen. Eine Heerschar verteilter, individuell konfigurierter, für sich selbst entscheidender Anlagen muss in der Lage sein, sich selbsttätig untereinander abzustimmen. Hierfür werden Algorithmen benötigt, die in der Lage sind, die alternativen Möglichkeiten dezentraler Energieanlagen systematisch zu betrachten und zu bewerten. Jedwede Planung im zukünftigen Smart Grid ist wertlos, wenn nicht bei der Planung bereits die Umsetzbarkeit der Einsatzpläne sichergestellt werden kann. Ein Beitrag zur Sicherstellung der praktischen Umsetzbarkeit der Ergebnisse von Planungsalgorithmen konnte in dieser Arbeit geleistet werden.

## Literaturverzeichnis

- [Be01] Ben-Hur, Asa; Siegelmann, Hava T.; Horn, David; Vapnik, Vladimir: Support Vector Clustering. *Journal of Machine Learning Research*, 2:125–137, 2001.
- [Br15] Bremer, Jörg: Constraint-Handling mit Supportvektor-Dekodern in der verteilten Optimierung. Dissertation, Carl von Ossietzky Universität Oldenburg, 2015.
- [BRS11] Bremer, Jörg; Rapp, Barbara; Sonnenschein, Michael: Encoding distributed Search Spaces for Virtual Power Plants. In: *IEEE Symposium Series on Computational Intelligence 2011 (SSCI 2011)*. Paris, France, 4 2011.
- [Bu08] Buchholz, Bernd; Bühner, Volker; Frey, Hellmuth; u.a.: VDE-Studie: Smart Distribution 2020 – virtuelle Kraftwerke in Verteilungsnetzen; technische, regulatorische und kommerzielle Rahmenbedingungen. Studie der Energietechnischen Gesellschaft im VDE (ETG), Verband der Elektrotechnik Elektronik Informationstechnik e.V. (VDE), 2008.
- [Cr10] Consentec GmbH; r2b energy consulting GmbH: Voraussetzungen einer optimalen Integration erneuerbarer Energien in das Stromversorgungssystem. Studie im Auftrag des Bundesministeriums für Wirtschaft und Technologie (BMWi), Endbericht, Consentec Consulting für Energiewirtschaft und -technik GmbH und r2b energy consulting GmbH, Aachen, Köln, 06 2010.

- [De10] Deutsche Bundesregierung (Bundeskabinett): National Renewable Energy Action Plan in accordance with Directive 2009/28/EC on the promotion of the use of energy from renewable sources. Nationaler Aktionsplan, 2010.
- [Eu09] European Parliament & Council: Directive 2009/28/EC of 23 April 2009 on the promotion of the use of energy from renewable sources and amending and subsequently repealing Directives 2001/77/EC and 2003/30/EC. Richtlinie, 2009.
- [KE95] Kennedy, J.; Eberhart, R.: Particle swarm optimization. In: Neural Networks, 1995. Proceedings., IEEE International Conference on. Jgg. 4. IEEE, S. 1942–1948 vol.4, November 1995.
- [KT04] Kwok, J. T. Y.; Tsang, I. W. H.: The pre-image problem in kernel methods. Neural Networks, IEEE Transactions on, 15(6):1517–1525, November 2004.
- [Mi99] Mika, Sebastian; Schölkopf, Bernhard; Smola, Alex; Müller, Klaus R.; Scholz, Matthias; Rätsch, Gunnar: Kernel PCA and de-noising in feature spaces. In: Proceedings of the 1998 conference on Advances in neural information processing systems II. MIT Press, Cambridge, MA, USA, S. 536–542, 1999.
- [Pa07] Park, Jooyoung; Kang, Daesung; Kim, Jongho; Kwok, James T.; Tsang, Ivor W.: SVDD-Based Pattern Denoising. Neural Computing, 19(7):1919–1938, 2007.
- [Ra12] Ramchurn, Sarvapali D.; Vytelingum, Perukrishnen; Rogers, Alex; Jennings, Nicholas R.: Putting the 'Smarts' into the Smart Grid: A Grand Challenge for Artificial Intelligence. Commun. ACM, 55(4):86–97, April 2012.
- [Sc99] Schölkopf, B.; Mika, S.; Burges, C.; Knirsch, P.; Müller, K.-R.; Rätsch, G.; Smola, A.: Input space vs. feature space in kernel-based methods. IEEE Transactions on Neural Networks, 10(5):1000–1017, 1999.
- [STC04] Shawe-Taylor, John; Cristianini, Nello: Kernel Methods for Pattern Analysis. Cambridge University Press, New York, NY, USA, 2004.
- [TD04] Tax, David M. J.; Duin, Robert P. W.: Support Vector Data Description. Mach. Learn., 54(1):45–66, 2004.
- [XSA05] Xiong, Huilin; Swamy, M.N.S.; Ahmad, M.O.: Optimizing the kernel in the empirical feature space. Neural Networks, IEEE Transactions on, 16(2):460–474, 03 2005.



**Jörg Bremer** wurde am 1. März 1968 in Oldenburg (Oldb) geboren. Nach dem Abitur auf dem zweiten Bildungsweg studierte er von 2000 bis 2006 Umweltinformatik an der Carl von Ossietzky Universität Oldenburg. Nach einer Tätigkeit am OFFIS Institut für Informatik kehrte er 2007 an die Universität zurück und arbeitete als wissenschaftlicher Mitarbeiter an den Lehrstühlen für Umweltinformatik und VLBA an diversen nationalen und internationalen Forschungsprojekten sowie in der Lehre in den Fachbereichen Umwelt- und Energieinformatik. Das Forschungsinteresse liegt hauptsächlich auf Methoden der Computational Intelligence und deren Anwendung in der Umwelt- und Energie-

domäne. Nach der Promotion erfolgte im Jahr 2015 ein Wechsel in die Abteilung für Energieinformatik der Universität Oldenburg wo er bis heute vorwiegend in der Lehre tätig ist.