UNIVERSITÄT OSNABRÜCK

Institut für Informatik

Seminar Complex Scheduling

Spielplanerstellung für die Österreichische und Deutsche Fussball-Bundesliga

Dorit Borrmann

Wintersemester 2007/08

Basierend auf dem Artikel "Scheduling the professional soccer leagues of Austria and Germany" von Thomas Bartsch, Andreas Drexl und Stefan Kröger

Zusammenfassung

Diese Arbeit bezieht sich auf meinen Vortrag vom 08. Januar 2008 im Seminar "Complex Scheduling" und basiert auf dem Artikel "Scheduling the professional soccer leagues of Austria and Germany" von Thomas Bartsch, Andreas Drexl und Stefan Kröger.

Es wird die Spielplanerstellung für Sportligen, im Speziellen für die Deutsche und Österreichische Fussball-Bundesliga, behandelt. Dabei werden zuerst die Anforderungen an einen Spielplan erläutert. Diese Anforderungen werden als partiell erneuerbare Ressourcen modelliert. Anschließend werden Algorithmen vorgestellt um in kurzer Zeit automatisiert passende Spielpläne zu erzeugen.

Inhaltsverzeichnis

Kapitel 1

Einleitung

Professionellem Sport kommt in der heutigen Zeit eine große Bedeutung zu. In Europa ist Fussball die dominierende Sportart. Das Interesse der Menschen wird nicht zuletzt durch die starke Präsenz in den Medien unterstützt. Aus diesem Grund werden Sportligen nicht nur von sportlichen sondern viel stärker von wirtschaftlichen Interessen beeinflusst. Diese beziehen sich unter anderem auf Eintrittsgelder, Werbeeinnahmen sowie Rundfunk- und Fernsehgelder. Um diese zu optimieren stellen die einzelnen Verantwortlichen unterschiedliche Anforderungen an die Spielpläne der Sportligen. Während die übertragenden Medienanstalten eine gleichmäßige Verteilung der attraktiven Spiele auf alle Spieltage wünschen, legen Zuschauer Wert auf wechselnde Heim- und Auswärtsspiele ihrer Mannschaften. Die Mannschaften selbst fordern beispielsweise aus Fairnessgründen wechselnde Gegnerstärken oder attraktive Heimspiele zu besonderen Ereignissen, Vereinsjubiläen oder Volksfesten wie dem Oktoberfest in München.

Ein Spielplan ist eine Verteilung von Spielpaarungen der Mannschaften auf die Spieltage eines Turniers. Je nach Wettbewerb existieren unterschiedliche Schemen. Eine Menge von Mannschaften spielt gegeneinander. Bei Round Robin Turnieren bestreitet jede Mannschaft eine feste Anzahl von Spielen gegen jede andere Mannschaft. Spielt die Mannschaft im eigenen Stadion spricht man von einem Heimspiel, ansonsten von Auswärtsspielen.

Die Fussball-Bundesligen in Deutschland und Österreichisch sind als Round Robin Turniere organisiert. Die Spielplanerstellung für diese Ligen wird im Folgenden behandelt. Dabei werden die Anforderungen als Nebenbedingungen basierend auf Ressourcen modelliert. Diese Anforderungen an die Spielpläne werden in Kapitel 2 vorgestellt und in Kapitel 3 mathematisch modelliert. Das darauffolgende Kapitel stellt Algorithmen zur automatisierten Generierung von Spielplänen dar. Eine experimentelle Evaluation dieser Verfahren inklusive ihrer konkreten Anwendung für die beiden Ligen folgt in Kapitel 5. Zum Schluss kommt eine abschließende Zusammenfassung und Diskussion der Ergebnisse.

Kapitel 2

Grundlagen

Die Erstellung eines Spielplans erfordert die Berücksichtigung einer großen Anzahl von Bedingungen. Einige dieser Bedingungen sind hart, das heißt sie müssen eingehalten werden, während andere Bedingungen nur wünschenswert sind und Aussagen über die Güte eines Spielplans erlauben. Im nächsten Abschnitt wird die Terminologie für die anschließende Modellierung definiert. Es folgt eine Auflistung der konkreten Anforderungen an die beiden zu erstellenden Spielpläne und die Belegung der Symbole und Zuordnung der Anforderungen zu den beiden Ligen.

2.1 Terminologie und Symbole

Zur mathematischen Modellierung der Sportwettbewerbe müssen die Charakteristiken der Sportligen symbolisch definiert werden. Eine *Liga* $\mathscr{L} = \{i|1, \ldots, N\}$ besteht aus einer geraden Anzahl $N \in \mathbb{N}$ von *Mannschaften*. Ungerade Mannschaftsanzahlen werden durch hinzufügen eines Dummy-Teams vermieden. Je zwei Mannschaften $i, j \in \mathscr{L}, i \neq j$ bestreiten ein *Spiel* $w = (nr, (i, j))$, wobei $nr \in \mathbb{N}$ das Spiel eindeutig definiert, sofern eine Spielpaarung (i, j) mehrfach vorkommt. Das Spiel findet im Stadion von Mannschaft i statt, d.h. i hat ein Heimspiel während j auswärts spielen muss. W Spiele werden zusammengefasst zu einem *Wettbewerb* $\mathscr{W} = \{w|w = 1, \ldots, W\}$.

Ist ein Spiel für einen Tag t eingeplant, so wird es als *datiertes Spiel* $\bar{w} = (w, t)$ bezeichnet. Die Spiele sind in S *Spieltage* oder *Runden* s unterteilt. Jede Runde besteht aus einer Menge $\mathscr{D}_s$ von Tagen an denen die Spiele stattfinden. Da jede Mannschaft pro Runde genau einmal, spielt gibt es $N/2$ Spiele. In einem Double Round Robin Turnier spielen alle Mannschaften zweimal gegeneinander. Eine *Saison* $\mathscr{S}$ besteht somit aus $S = W/(N/2)$ Runden. In vielen Ligen besteht die Saison aus zwei Halbserien, der Hinrunde und der Rückrunde, die komplementär zueinander sind. Das heißt, die Spieltage unterscheiden sich einzig durch geändertes Heimrecht voneinander. Eine Runde $s \in \mathscr{S}$ ist *disjunkt*, wenn sie keinen Tag gemeinsam hat mit einer anderen Runde $s' \in \mathscr{S}$, also wenn $\mathscr{D}_s \bigcap \mathscr{D}_{s'} = \emptyset$. Ein *Spielplan* $\mathscr{P} = \bar{w}|\bar{w} = 1, \ldots, W$ vereinigt die W datierten Spiele einer Saison. Die Teilmenge $\mathscr{P}_s = \{\bar{w}|\bar{w} = 1, \ldots, (N/2)\} \subseteq \mathscr{P}$ umfasst die Spiele eines Spieltags s. Die Saison $\mathscr{S}$ heißt geordnet, wenn keine zwei Runden $s, s' \in \mathscr{S}, s \neq s'$ existieren, bei denen ein Tag von s' vor dem ersten Tag der Runde s und ein Tag nach dem letzten Tag der Runde s liegt.

Eine Zeichenfolge bestehend aus `H` (für Heimspiele) und `A` für Auswärtsspiele mit der Länge der Anzahl der Spiele einer Halbserie wird als *Pattern* bezeichnet. Ein *Home-Away-Pattern* ist eine $N \times S/2$ Matrix, welche die Pattern der Länge $S/2$ der N Mannschaften für die Hinrunde einer Saison verwaltet. Ein *Opponent Pattern* g_i ist eine Liste der Gegner j von Mannschaft i in jeder Runde einer Halbserie. Der *Opponent Plan* ist eine $N \times S/2$ Matrix aus den Opponent Pattern der N Mannschaften.

Als *Break* bezeichnet man ein Heim- oder Auswärtsspiel einer Mannschaft, die in der vorhergehenden Runde ebenfalls ein Heim- bzw. Auswärtspiel hatte. Die minimale Anzahl von Breaks beträgt $N - 2$, d.h. zwei Mannschaften haben kein Break und alle anderen jeweils ein Break. Ein Beweis hierfür findet sich in [4].

2.2 Anforderungen an die Spielpläne

Neben dem generellen Problem einen Spielplan zu erstellen bei dem alle Mannschaften gegeneinadner spielen, gilt es eine großen Anzahl an Anforderungen zu berücksichtigen. In diesem Kapitel werden Organisatorische- und Attraktivitätsanforderungen aufgelistet und solche, die Fairness der Spielplanerstellung betreffen. Die Anforderungen sind in Zusammenarbeit mit den Fussballverbänden aus Deutschland (`DFB`) und Österreich (`ÖFB`) entstanden.

2.2.1 Organisatorische Anforderungen

Organisatorische Anforderungen betreffen die Verwendung der Stadion und die Sicherstellung eines geordneten Ablaufs der Fussballspiele, unter anderem was Polizei und Verkehr betrifft.

O1. Stadionverfügbarkeit: Einige Stadien sind nur eingeschränkt für Fussballspiele einer Mannschaft verfügbar. Dies kann durch Renovierungsarbeiten begründet sein, aber auch durch anderweitige Nutzung, für Konzerte, andere Sportarten und nicht zuletzt auch eventuelle Fussballspiele anderer Mannschaften, die das gleiche Stadion nutzen.

O2. Regionen: Fussballspiele erfordern organisatorischen Aufwand unterschiedlicher Art. Anreisende Zuschauer verursachen ein erhöhtes Verkehrsaufkommen. Außerdem muss die Sicherheit durch ein ausreichendes Polizeiaufgebot gewährleistet werden. Dieses hat Einfluss auf die ganze Region, in der das Spiel stattfindet. In einigen Regionen, in Deutschland zum Beispiel im Ruhrgebiet, gibt es viele Mannschaften, die in der Bundesliga spielen. Aus verkehrs- und sicherheitstechnischen Gründen ist es sinnvoll, dass nur eine bestimmte Anzahl von Spielen gleichzeitig in dieser Region stattfindet.

O3. Sicherheitsanforderungen: Es gibt Spiele, für die spezielle Sicherheitsvorkehrungen nötig sind. Ist bei zwei Mannschaften bekannt, dass es bei ihren Begegnungen häufig zu Auschreitungen kommt, kann gefordert werden, dass ihre Spiele gegeneinander zu bestimmten Zeitpunkten, beispielsweise nachmittags, stattfinden.

O4. Wechselbeziehungen zwischen Ligen: Sowohl in Deutschland als auch in Österreich gibt es neben der Bundesliga noch eine weitere professionelle Fussball-Liga. Die Spiele dieser Liga müssen ebenso auf die Spiele der obersten Liga abgestimmt werden. Dies betrifft besonders die Stadionverfügbarkeit und die Anzahl der Spiele in einer Region.

2.2.2 Attraktivitäts-Anforderungen

Attraktivitätsanforderungen sind für Vereine und Medien von Bedeutung. Die Saison soll so gestaltet werden, dass den Spielen reges Interesse durch Zusacheuer zukommt.

A1. Breaks: Hat eine Mannschaft i an zwei aufeinanderfolgenen Spieltagen s und $s+1$ zwei Auswärtsspiele oder zwei Heimspiele, so hat i an Spieltag $s+1$ ein Break. Sowohl für die Spieler als auch für die Zuschauer ist es angenehmer abwechselnd Heim- und Auswärtsspiele zu haben. Die Anzahl der Breaks soll also minimiert werden. Es ist bekannt, dass die minimale Anzahl von Breaks $N-2$ beträgt, d.h. zwei Mannschaften haben kein Break und alle anderen jeweils ein Break.

A2. Minimaler Abstand zwischen Hin- und Rückspiel: Jede Mannschaft spielt eine gerade Anzahl von Spielen gegen jede andere Mannschaft, in jeder Runde einmal. Der Abstand zwischen zwei Spielen soll nicht zu gering sein.

A3. Heimspielpräferenzen: Bei Heimspielen können Mannschaften besonders auf die Unterstützung ihrer Fans bauen. Deswegen sollen Aufsteiger am ersten Spieltag zu Hause spielen dürfen. Bei den anderen Mannschaften wird versucht das Heimrecht entgegengesetzt zur vorhergegangenen Saison zu verteilen. Des Weiteren wünschen die Mannschaften bei Vereinsfeiern oder ähnlichen Ereignissen Heimspiele, sowie an Tagen, an denen sie mit besonders großem Zuschauerandrang rechnen können. Ein Beispiel dafür ist das Oktoberfest in München oder der Bremer Freimarkt.

A4. Attraktive Spiele: Besonderes Zuschauerinteresse wird bei Begegnungen traditionell starker Mannschaften, sogenannten Derbies, d.h. Spielen von Mannschaften aus benachbarten Städten, erwartet. Die Medien wünschen sich eine möglichst gleichmäßige Verteilung dieser Spiele über die gesamte Saison.

A5. Feste Spiele: Die Spannung wird besonders aufrecht erhalten, wenn der Ausgang des Wettbewerbs möglichst lange offen bleibt. So wird festgelegt, dass der Erst- und Zweitplatzierte der Vorsaison erst in einer der letzten drei Runden gegeneinander spielen, da von diesen Mannschaften erwartet wird, dass sie auch in der aktuellen Saison am Ausgang der Meisterschaft beteiligt sind.

A6. Spiele pro Tag: Bei der Vergabe der Fernsehrechte wird die Übertragung der Spiele für die einzelnen Tage geregelt. Damit die Fernsehsender sich darauf einstellen können, werden die Anzahlen der Spiele die an einem Tag stattfinden festgelegt.

2.2.3 Fairness Anforderungen

Da sportlicher Erfolg im professionellen Sport auch wirtschaftliche Vorteile bedeutet ist Fairness unabdinglich. Deswegen sollen allen Mannschaften die gleichen fairen Rahmenbedingungen garantiert werden.

F1. Spielfreie Tage: Zur Erholung sollen zwischen zwei Spielen einer Mannschaft müssen immer mindestens zwei spielfreie Tage liegen. Dabei müssen auch die Spiele anderer Wettbewerbe, z.B. nationaler und europïscher Pokalwettbewerbe berücksichtigt werden, an denen Mannschaften der Ligen teilnehmen.

F2. Wechselnde Gegnerstärke: Mannschaften sollen gleichmäßig verteilt starke und weniger starke Gegner haben. Die Gegnerstärke wird durch den Endstand aus der Vorsaison bestimmt.

F3. Verbotene Breaks: Um Wettbewerbsverzerrung zu vermeiden sind in der zweiten und letzten Runde keine Breaks erlaubt.

2.3 Anforderungen der beiden Ligen

Die beiden Fussballligen aus Österreich und Deutschland haben unterschiedliche Charakteristiken. Die Wettbewerbe verfügen über unterschiedliche Schemata. Diese Schemata führen dazu, dass auch die Anforderungen sich unterschieden. Tabelle 2.1 liefert eine Gegenüberstellung der beiden Ligen. In den folgenden Abschnitten werden zuerst die Deutsche Fussball-Bundesliga und anschließend die Österreichische Bundesliga im Detail charakterisiert.

2.3.1 Deutsche Fussball-Bundesliga

Die "1. Fussball-Bundesliga" ist die höchste Fussballliga in Deutschland. Der `DFB` (Deutscher Fussball Bund) ist verantwortlich für die Organisation und damit auch die Spielplanerstellung dieser Liga, die als Double Round Robin Wettbewerb durchgeführt wird. Die 18 Mannschaften spielen jeweils zweimal gegeneinander, einmal im eigenen und einmal im Stadion des Gegners. Die Anordnung der Spiele geschieht in 34 Spieltagen, wobei die 17 Spieltage der Hinrunde komplementär zu denen der Rückrunde sind. Die Runden sind disjunkt und geordnet. An jedem Spieltag spielt jede Mannschaft einmal, so dass die Gesamtanzahl der Spiele $(18 \cdot 34)/2 = 306$ beträgt.

Die Spieltage finden entweder komplett am Wochenende oder in der Woche statt. Bei Spieltagen in der Woche wird am Dienstag und am Mittowch gespielt. Am Wochenende sind die Spiele auf Freitag, Samstag und Sonntag verteilt. Die Verteilung der Spiele auf die Wochentage am Beispiel der Saison 2007/2008 ist in Tabelle 2.2 aufgelistet.

Die Anforderungen, die vom `DFB` gestellt werden, sind in Tabelle 2.1 dargestellt. Harte Anforderungen die erfüllt werden müssen sind mit einem Punkt • gekennzeichnet. Die mit einem

Tabelle 2.1: Auflistungen der Anforderungen der Österreichischen und Deutschen Fussball-Bundesliga

	Charakteristik	DFB	ÖFB
	Ligen	2	2
	Teams	18	10
	Spiele	306	180
	disjunkt u. geordnet	•	•
O1	Stadionverfügbarkeit	•	•
O2	Regionen	•	•
O3	Sicherheit	•	
O4	Ligenkonflikte	○	○
A1	Breaks	•	•
A2	min. Zeit zw. Hin- und Rückspiel	•	•
A3	Heimpräferenzen	○	•
A4	Attraktivität	•	
A5	Feste Spiele	•	•
A6	Min/max Anzahl Spiele	•	
F1	Spielfreie Tage	•	
F2	Gegnerstärke	○	○
F3	Verbotene Breaks	•	•

Tabelle 2.2: Verteilung der Spiele auf die Tage einer Runde

Wochentag	Dienstag	Mittwoch	Freitag	Samstag	Sonntag
Min	4	4	1	6	2
Max	5	5	1	6	2

Kreis ○ markierten Anforderungen sind nur wünschenswert und bieten später Möglichkeit zur Optimierung des Spielplans.

Von den organisatorischen Anforderungen sind die Konflikte mit anderen Ligen zweitrangig. Die erste Bundesliga hat Vorrang vor der zweiten Bundesliga und wird deswegen zuerst geplant. Bei der Spielplanerstellung der zweiten Bundesliga ergeben sich dann Einschränkungen aus den Stadienbelegungen der bereits eingeplanten Liga.

Aufgrund der großen wirtschaftlichen Interessen an der Fussball-Bundesliga in Deutschland sollen viele Attraktivitätsanforderungen erfüllt werden. Um den Wettbewerb möglichst spannend zu halten und interessante Fernsehübertragungen konstant über die ganze Saison und bestenfalls auch für jeden Tag des Spieltags zu ermöglichen, sollen die Spiele gleichmässig verteilt werden. Dies betrifft die Verteilung auf die Spieltage, die Terminfestsetzung innerhalb eines Spieltags und auch die Vermeidung von Breaks. Heimpräferenzen einzelner Mannschaften sind für die

Attraktivität der gesamten Liga irrelevant und werden folglich nicht als zwingend erachtet.

Fairness soll verbessert werden durch verbotene Breaks in der zweiten und letzten Runde. Ein guter Start in die Saison ist eine große Hilfe zur Motivation und das Saisonende ist häufig die entscheidende Phase. Deswegen soll besonders an diesen Spieltagen keine Mannschaft durch zwei Heimspiele bevorteilt oder durch zwei Auswärtsspiele benachteiligt werden. Zur ausreichenden Erholung soll jede Mannschaft zwischen zwei Spiele mindestens zwei spielfreie Tage haben. Hierbei gilt es auch die Spieltage des Deutschen Pokalwettbewerbs (DFB-Pokal) und der europäischen Pokale (UEFA-Cup und Champions League) zu berücksichtigen. Wechselnder Gegnerstärke kommt eine geringere Bedeutung zu. Es kann zwar eine Abschätzung der Stärke aufgrund der Vorsaison getroffen werden, die tatsächliche Stärke einer Mannschaft lässt sich aber nicht im Voraus bestimmen und kann sich auch im Verlauf der Saison ändern.

2.3.2 Österreichische Fussball-Bundesliga

Die höchste Österreichische Liga im Fussball ist die Bundesliga, die vom ÖFB organisiert wird. Die Bundesliga besteht aus 10 Mannschaften. Die Saison ist in vier Teile unterteilt, in denen jede Mannschaft einmal gegen jede andere spielt. Somit gibt es 36 Spieltage, die disjunkt und geordnet sind und an denen jede Mannschaft jeweils einmal spielt. Alle Spiele eines Spieltags finden am selben Tag statt. Insgesamt gibt es also $(10 \cdot 36)/2 = 180$ Spiele.

Die Viertel der Saison sind zu zwei Halbserien zusammengeschlossen. Die erste Halbserie, die Spieltage 1 bis 18, können unabhängig von der zweiten Halbserie, den Spieltage 19 bis 36, geplant werden. Jedoch sollen keine zwei Mannschaften i und j, die an Spieltag 17 oder 18 gegeneinander spielen, an Spieltag 19 oder 20 wieder aufeinandertreffen. Die Anordnung der Hin- und Rückspiele in den Halbserien geschieht nach dem Englischen System. Das bedeutet, dass die Runden 9 und 10 komplementär zueinander sind und ebenfalls die Runden 1 bis 8 komplementär zu den Runden 11 bis 18. Ebenso sind auch die Spieltage der zweiten Hälfte komplementär zueinander.

Bei den organisatorischen Anforderungen sind Stadionverfügbarkeit und Regionszugehörigkeit zu beachten. Sicherheitsaspekte können außer Acht gelassen werden, da alle Spiele am gleichen Tag ausgetragen werden und somit die Verteilung auf die Wochentage nicht an eine bestimmte Tageszeit gebunden ist. Konflikte mit der zweithöchsten Fussballliga werden wie in Deutschland dadurch gelöst dass die Bundesliga zuerst eingeplant wird und dadurch Restriktionen für die andere Fussballliga entstehen.

Bei den Attraktivitätsanforderungen fallen offensichtlich Beschränkungen bezüglich der Anzahl der Spiele pro Wochentag weg. Attraktive Spiele müssen nicht berücksichtigt werden. Stattdessen sind aber Heimspielpräferenzen auch als harte Bedingung angelegt.

Die Fairnessanforderungen bezüglich wechselnder Gegnerstärke und verbotener Breaks entsprechen den deutschen Anforderungen. Spielfreie Tage können außer Acht gelassen werden, da dieses bereits bei der Terminfestlegung des Tages an dem eine Runde stattfindet berücksichtigt werden kann.

Kapitel 3

Modellierung

In diesem Kapitel werden die Spielpläne für den `DFB` und den `ÖFB` modelliert. Dazu wird zuerst ein grundlegendes Modell eingeführt, das in den darauffolgenden Abschnitten auf die Bedarfe der beiden Fussballverbände angepasst wird.

3.1 Grundlegendes Modell

Das grundlegende Modell definiert eine Generierung von Spielplänen unabhängig von organisatorischen, Attraktivitäts- und Fairness Anforderungen.

Der Wettbewerb $\mathscr{W} = w = (nr, (i, j))|w = 1, \ldots, W, i, j \in \mathscr{L}, nr \in \mathbb{N}$ beschreibt die W Spiele der N Mannschaften aus Liga $\mathscr{L}$. Die Spiele sind in Runden $\mathscr{S} = \{s|s = 1, \ldots, S\}$ mit $S = W/(N/2)$ zusammengefasst. Die Tage jeder Runde $s \in \mathscr{S}$ sind durch $\mathscr{D}_s = \{t|$ Runde s kann an Tag t stattfinden$\}$ definiert.

Die binäre Variable $x_{wt} = 1$ gibt für jedes Spiel w an, ob es am Tag t stattfindet:

$$x_{wt} \in \{0, 1\} \qquad \forall w \in \mathscr{W}, \forall s \in \mathscr{S} \forall t \in \mathscr{D}_s. \tag{3.1}$$

Gleichung (3.2) garantiert, dass jedes Spiel genau einmal stattfindet:

$$\sum_{s \in \mathscr{S}} \sum_{t \in \mathscr{D}_s} x_{wt} = 1, \qquad \forall w \in \mathscr{W}. \tag{3.2}$$

Jede Mannschaft soll pro Runde genau einmal spielen. Dies wird mit folgender Gleichung sichergestellt:

$$\sum_{\substack{w \in \mathscr{W} \\ w=(nr,(i,j)) \\ i'=i \vee i'=j}} \sum_{t \in \mathscr{D}_s} x_{wt} = 1, \qquad \forall i' \in \mathscr{L}, \forall s \in \mathscr{S}. \tag{3.3}$$

Keine Mannschaft darf an einem Tag mehr als ein Spiel bestreiten:

$$\sum_{\substack{w \in \mathscr{W} \\ w=(nr,(i,j)) \\ i'=i \vee i'=j}} x_{wt} \leq 1, \qquad \forall i' \in \mathscr{L}, \forall s \in \mathscr{S}, \forall t \in \mathscr{D}_s. \tag{3.4}$$

Die Gleichungen (3.2) bis (3.4) definieren das grundlegende Modell eines Spielplans. Für ein Double Round Robin Turnier mit disjunkten und geordneten Runden kann der Spielindex entfallen, so dass Modell (3.5) entsteht.

$$\begin{aligned}
& x_{ijt} \in \{0,1\} && \forall i,j \in \mathscr{L}, \forall s \in \mathscr{S} \forall t \in \mathscr{D}_s, \\
& \sum_{s \in \mathscr{S}} \sum_{t \in \mathscr{D}_s} x_{ijt} = 1, && \forall i,j \in \mathscr{L}, i \neq j, \\
& \sum_{s \in \mathscr{S}} \sum_{t \in \mathscr{D}_s} x_{jit} = 1, && \forall i,j \in \mathscr{L}, i \neq j, \\
& \sum_{\substack{j \in \mathscr{L} \\ j \neq i}} \sum_{t \in \mathscr{D}_s} (x_{ijt} + x_{jit}) = 1, && \forall i' \in \mathscr{L}, \forall s \in \mathscr{S}, \forall t \in \mathscr{D}_s.
\end{aligned} \tag{3.5}$$

3.1.1 Deutsche Fussball-Bundesliga

Das Modell für die Spielplanerstellung einer konkreten Sportliga geht weit über die Möglichkeiten des im vorherigen Abschnitt definierten grundlegenden Modells hinaus. In Kapitel 2.3.1 wurden die Anforderungen die der *DFB* an Spielpläne für die Bundesliga stellt eingeführt. Für ein mathematisches Modell des Spielplans, das zur automatisierten Erstellung eines Spielplans und Überprüfung der Anforderungen verwendet werden kann, ist es notwendig auch diese Anforderungen mathematisch zu formalisieren. Hierbei muss unterschieden werden zwischen den strikten Anforderungen, die als Bedingungen definiert werden und denen die als eine Zielfunktion optimiert werden können aber nicht eingehalten werden müssen.

1. Liga $\mathscr{L} = \{i|1,\ldots,N\}$ besteht aus $N = 18$ Mannschaften.

2. Im Wettbewerb $\mathscr{W}$ spielt jede Mannschaft gegen jede ander Mannschaft ein Heim- und ein Auswärtsspiel, insgesamt sind das $W = N \cdot (N-1) = 306$ Spiele.

3. Die Tage $\mathscr{T} = \{t|t = 1,\ldots,T\}$ an denen `DFB` und Europa Pokalspiele stattfinden sind in Voraus bekannt.

4. Die Mengen $\mathscr{G}_m \subset \mathscr{L}, m \in \mathscr{M} = \{1,2,3\}$ unterteilen die Liga entsprechend ihrer Spielstärke. $\mathscr{G}_1$ enthält die $\lfloor N/3 \rfloor$ stärksten Mannschaften der Vorsaison, also die Bestplatzierten. In $\mathscr{G}_3$ sind die $\lfloor N/3 \rfloor$ schlechtesten Mannschaften der Vorsaison und in $\mathscr{G}_2$ die übrigen Mannschaften.

Tabelle 3.1: Ressourcenprofil für minimale und maximale Anzahl von Spielen pro Tag

$K_{r_1 s}$	Freitag	Samstag	Sonntag	Dienstag	Mittwoch	k_{ijr_1}
Min	-1	-6	2	-4	-4	-1
Max	1	6	2	5	5	1

5. Auf dem Zeitraum den die Saison umfasst werden U nicht disjunkte Perioden und auf diesen partiell erneuerbare Ressourcen definiert. Die Indizes der U Mengen von Tagen sind in $\mathscr{S}^G = \{s|s = 1, \ldots, U\}$ enthalten. $\mathscr{D}_s$ beinhaltet die Tage der Periode s. Es gibt mehrere Teilmenden von $\mathscr{S}^G$, die Perioden von unterschiedlicher Bedeutung zusammenfassen. Die ersten S Teilmengen definieren die Saison $\mathscr{S} = \{s|s = 1, \ldots S\}$ bestehend aus $S = W/(N/2) = 34$ Runden. Die Indizes der Hinrunde sind in $\mathscr{S}^H = \{s|s = 1, \ldots S/2\} \subset \mathscr{S}$ enthalten. $\mathscr{S}^V \subset \mathscr{S}$ definiert die Spieltage an denen Breaks verboten sind. Die übrigen Perioden $\mathscr{S}^G \backslash \mathscr{S}$ beinhalten alle weiteren relevanten Perioden, z.B. die Tage an denen andere nationale oder internationale Spiele stattfinden, an denen Mannschaften aus $\mathscr{L}$ beteiligt sind. $\mathscr{S}^G$ enthält auch einelementige Teilmengen für alle Tage $t \in D_s | s \in \mathscr{S}$ also die Tage an denen Bundesligaspiele stattfinden. Diese Teilmengen haben die Indizes $S + 1, \ldots, S + T$.

6. Ein Großteil der Anforderungen wird durch R partiell erneuerbare Ressourcen $\mathscr{R} = \{r|r = 1, \ldots, R\}$ realisiert. Für jede Ressource r ist die Kapazität K_{rs} gegeben, die an den Tagen $\mathscr{D}_s$ einer jeden Periode s zur Verfügung steht. k_{ijr} bestimmt den Bedarf einer Spielpaarung (i, j) an einer Ressource r. Im Folgenden werden zur besseren Unterscheidung die Ressourcen mit einem Index x versehen $r_x \in \mathscr{R} | x \in 1, \ldots, 6$, wobei der Index x den Typ der Ressource bestimmt. In den Tabellen A.1 bis A.6 werden die verwendeten partiell erneuerbaren Ressourcen genauer definiert.

   Tabelle A.1 zeigt die Charakteristiken der partiell erneuerbaren Ressourcen für minimale und maximale Anzahlen von Spielen pro Tag. Hierzu wird eine Dummy-Ressource r_1 verwendet. Für die Mengen $s = S + 1, \ldots, S + T$, also jeden Tag an dem ein Bundesligaspiel stattfinden kann, werden die Kapazitäten $K_{r_1 s} = Max$ gesetzt. Ist keine maximale Spielanzahl gegeben, so beträgt $K_{r_1 s} = \infty$. Der Bedarf einer Spielpaarung (i, j) an der Ressource beträgt immer 1. Minimale Spielanzahlen werden mit $K_{r_1 s} = -Min$ oder $K_{r_1 s} = -\infty$ und dem Bedarf $k_{ijr_1} = -1$ garantiert. Tabelle 3.1 zeigt exemplarisch die Ressourcen für die minimale und maximale Anzahl von Spielen pro Tag, wie in Tabelle 2.2 für die Deutsche Bundesliga angegeben. Für minimale und maximale Anzahl werden zwei unterschiedliche Ressourcen benötigt. Die Kapazitäten für die einzelnen Tage stehen im linken Teil der Tabelle. Die ganz rechte Spalte gibt den Bedarf eines Spiels an diesen Ressourcen an.

   Stadien sind in der Regel mit der Kapazität $K_{r_2 s} = 1$ verfügbar (vgl. TabelleA.2). Ist ein Stadion der Mannschaft i in einer Runde nicht verfügbar, so gilt $K_{r_2 s} = 0$ für die entsprechende Ressource r_2. Eine Spielpaarung (i, j) benötigt eine Einheit der Ressource r_2 für Stadion i. Heimspiele können garantiert werden, indem das Stadion in der komplementären

Runde unverfügbar gemacht wird.

Tabelle A.3 zeigt die Ressourcen r_3 für die Verwaltung von Regionen. Die Zuordnung von einer maximalen Anzahl von Spielen zu einer Region geschieht identisch zur maximalen Anzahl von Spielen pro Tag. Jedoch muss hierbei berücksichtigt werden, dass Max mindestens $\lceil 0.5 \cdot M \rceil$ beträgt bei M Mannschaften in der Region, da jede Mannschaft durchschnittlich alle zwei Runden ein Heimspiel haben muss. Unter Umständen muss auch die Anzahl der Tage einer Runde in Betracht gezogen werden.

Sicherheitsaspekte dienen dazu Spiele an bestimmten Tagen zu verbieten bzw. an anderen Tagen zu erzwingen. Dies kann dazu verwendet werden Spiele in der Woche oder am Wochenende zu erzwingen. Dadurch kann auch erreicht werden, dass Spiele aus Sicherheitsgründen oder auch wegen mangelndem Flutlicht am Nachmittag stattfinden. Die Modellierung sieht vor unterschiedliche Sicherheitstypen r_4 zu definieren (vgl. Tabelle A.4) deren Kapazität an entsprechenden Tagen $K_{r_4 s} = 1$, an den restlichen Tagen aber 0 beträgt. Verlangt ein Spiel den Sicherheitstyp r_4 so wird eine Einheit der Ressource benötigt.

Ähnlich zu den Sicherheitsaspekten wird mit attraktiven Spielen verfahren (vgl. Tabelle A.5). Für jeden Typ von attraktivem Spiel existiert eine Ressource r_5 deren Kapazität dazu verwendet werden kann gleich attraktive Spiele gleichmässig auf die Saison zu verteilen oder auch an einem bestimmten Spieltag ein attraktives Spiel zu erzwingen. Die 15 Spiele der Mannschaften aus $\mathscr{G}_1$ pro Halbserie werden gleichmässig verteilt wenn eine entsprechende Ressource die Kapazität $K_{r_5 s} = 1$ hat.

Spielfreie Tage werden mit Hilfe eigens dafür definierter Perioden $s = S + T + 2, \ldots, U$ gewährleistet. Für jede Mannschaft wird eine Ressource r_6 eingeführt. Die Perioden umfassen Tage t, t' aus aufeinanderfolgenden Runden, zwischen denen weniger als zwei Tage liegen. Die Kapazität eines jeden Teams für die gesamte Periode beträgt $K_{r_6 s} = 1$. Ist t oder t' ein Tag, der für Spiele aus anderen Wettbewerben reserviert ist, so wird die Kapazität aller Mannschaften die an diesem Wettbewerb teilnehmen auf 0 gesetzt.

7. Heimspielwünsche der einzelnen Mannschaften werden angegeben durch eine binäre Variable. Ist $h_{is} = 1$ gesetzt, möchte Mannschaft i am Spieltag s zu Hause spielen. Möchte eine Mannschaft ein Auswärtsspiel in Runde s, kann dies erreicht werden, indem in der komplementären Runde s' $h_{is'} = 1$ gesetzt wird.

8. Heimpräferenzen (`A3`) und wechselnde Gegnerstärken (`F2`) sind keine harten Bedingungen. Sie werden in die Zielfunktion integriert, die minimert wird um besonders gute Spielpläne zu erzeugen. Mit anderen Worten, ein Verstoß gegen eine dieser Bedingungen hat nicht zur Folge, dass der Spielplan ungültig wird, aber unter den gültigen Spielplänen wird der bevorzugt, der die wenigsten Verstöße aufweist. Die beiden Ziele gehen unterschiedlich gewichtet in die Zielfunktion ein. $\hat{\omega_1}$ gewichtet Attribut $k = 1$, die missachteten Heimspielpräferenzen, während $\hat{\omega_2}$ gleichbleibende Gegnerstärke bestraft.

   Durch Verwendung von

   β_k : Priorität für Attribut $k, \beta_1 + \beta_2 = 1, \beta_1, \beta_2 \geq 0$,

   $B_k^{min,N}$: kleinster Wert des Attributs k bei N Mannschaften

$B_k^{max,N}$: kleinster Wert des Attributs k bei N Mannschaften

ergeben sich die Gewichte

$$\hat{w}_k = \begin{cases} \frac{\beta_k}{B_k^{max,N} - B_k^{min,N}}, & \text{wenn } B_k^{min,N}, \qquad k = 1, 2, \\ 0, & \text{sonst.} \end{cases}$$

Die Werte für $B_k^{max,N}$ und $B_k^{min,N}$ wurden experimentell bestimmt. Die Anzahl der Mannschaften N beeinflusst die Anzahl des Auftretens von wechselnder Gegnerstärke. Um Vergleichbarkeit der Ergebnisse zu erreichen wird die Zielfunktion auf das Intervall $[0, 1]$ normalisiert durch Verwendung der Konstanten $\Delta = \delta_1 + \delta_2$ mit

$$\hat{w}_k = \begin{cases} \frac{\beta_k \cdot B_k^{min,N}}{B_k^{max,N} - B_k^{min,N}}, & \text{wenn } B_k^{min,N}, \qquad k = 1, 2. \\ 0, & \text{sonst.} \end{cases}$$

Im Folgenden werden $\beta_1 = \beta_2 = 0.5$ verwendet.

Mit den getroffenen Definitionen lassen sich nun formal die Zielfunktion und die Bedingungen unter denen sie minimiert werden soll definieren. Damit erreicht man ein komplettes Modell der Anforderungen des `DFB`.

Die Zielfunktion

$$\min \left(\hat{w}_1 \cdot \left(\sum_{i \in \mathscr{L}} \sum_{j \in \mathscr{L}} \sum_{s \in \mathscr{S}} \sum_{t \in \mathscr{D}_s} h_{is} x_{jit} \right) \right.$$
$$\left. + \hat{w}_2 \cdot \left(\sum_{i \in \mathscr{L}} \sum_{m \in \mathscr{M}} \sum_{j \in \mathscr{G}_m} \sum_{j' \in \mathscr{G}_m} \sum_{\substack{s \in \mathscr{S}^H \\ s \neq S/2}} \left(\sum_{t \in \mathscr{D}_s} (x_{ijt} + x_{jit}) \sum_{t \in \mathscr{D}_{s+1}} \left(x_{ij't} + x_{j'it} \right) \right) \right) - \Delta \right)$$

umfasst die weichen Anforderungen an den Spielplan der Fussball-Bundesliga. Der erste Term zählt die unerfüllten Heimpräferenzen. Möchte eine Mannschaft i in Runde s ein Heimspiel haben ($h_{is} = 1$), ist aber für ein Auswärtsspiel eingeplant (x_{jit}, mit $t \in \mathscr{D}_s$), so wird gegen die Heimpräferenz verstoßen und die Zielfunktion verschlechtert sich. Der zweite Term zählt Verstöße gegen die Forderung nach wechselnder Gegnerstärke. Spielt eine Mannschaft in zwei aufeinanderfolgenden Runden s und $s + 1$ gegen zwei Mannschaften aus der gleichen Menge $\mathscr{G}_m$ vergrößert dies den Wert der Zielfunktion. Die beiden Anzahlen an Anforderungsverstößen werden mit $\hat{\omega}_1, \hat{\omega}_2$ gewichtet summiert. Ziel ist es nun einen Spielplan zu generieren, bei dem die Zielfunktion minimiert wird und zusätzlich die folgenden Nebenbedingungen eingehalten werden.

Wie bereits im grundlegenden Modell wird wieder eine binäre Variable verwendet, die aussagt ob ein Spiel (i,j) in Runde s stattfindet:

$$x_{ijt} \in \{0,1\}. \qquad \forall i,j \in \mathscr{L}, \forall s \in \mathscr{S}^G, \forall t \in \mathscr{D}_s.$$

Wird für eine bestimmte Spielpaarung ein fester Termin gewünscht, so muss diese Variable auf 1 gesetzt werden. Jede Mannschaft soll gegen jede andere Mannschaft genau einmal in der Hinrunde spielen. Das heißt entweder $x_{ijt} = 1$ oder $x_{jit} = 1$ muss genau einmal für alle i und j an einem beliebigen Tag t gelten. Dies wird formalisiert durch

$$\sum_{s \in \mathscr{S}^H} \sum_{t \in \mathscr{D}_s} (x_{ijt} + x_{jit}) = 1, \qquad \forall i,j \in \mathscr{L}, i \neq j.$$

Garantiert man nun noch die Komplementarität von Hin- und Rückrunde, dass heißt, dass zwei Mannschaften am gleichen Spieltag der Hin- und Rückrunde mit wechselndem Heimrecht gegeneinander spielen durch

$$\sum_{t \in \mathscr{D}_s} x_{ijt} - \sum_{t \in \mathscr{D}_{s+S/2}} x_{jit} = 0, \qquad \forall i,j \in \mathscr{L}, \forall s \in \mathscr{S}^H,$$

so spielen alle Mannschaften i und j genau zweimal gegeneinander. Eine Mannschaft darf an den Tagen $t \in \mathscr{D}_s$ eines jeden Spieltags s nur genau einmal spielen, egal ob im eigenen Stadion oder dem des Gegners:

$$\sum_{\substack{j \in \mathscr{L} \\ j \neq i}} \sum_{t \in \mathscr{D}_s} (x_{ijt} + x_{jit}) = 1, \qquad \forall i \in \mathscr{L}, \forall s \in \mathscr{S}.$$

Die Anzahl der Breaks soll minimiert werden. Das bedeutet bei einer gerade Anzahl von Mannschaften, dass die Anzahl von Breaks maximal $N-2$ beträgt. Diese Forderung wird modelliert durch

$$\sum_{i \in \mathscr{L}} \sum_{j \in \mathscr{L}} \sum_{j' \in \mathscr{L}} \sum_{\substack{s \in \mathscr{S}^H \\ s \neq S/2}} \left(\left(\sum_{t \in \mathscr{D}_s} x_{ijt} \cdot \sum_{t \in \mathscr{D}_{s+1}} x_{ij't} \right) + \left(\sum_{t \in \mathscr{D}_s} x_{jit} \cdot \sum_{t \in \mathscr{D}_{s+1}} x_{j'it} \right) \right) \leq N - 2.$$

Ein Break kommt vor, wenn für zwei aufeinanderfolgende Spieltage s und $s+1$ Mannschaft i gegen zwei Mannschaften j und j' spielt und dabei beide Spiele zu Hause oder auswärts bestreitet. An bestimmten Spieltagen sollen Breaks komplett verboten werden:

$$\left(\sum_{j\in\mathscr{L}}\sum_{t\in\mathscr{D}_{s-1}} x_{ijt}\cdot\sum_{j\in\mathscr{L}}\sum_{t\in\mathscr{D}_s} x_{ijt}\right)+\left(\sum_{j\in\mathscr{L}}\sum_{t\in\mathscr{D}_{s-1}} x_{jit}\cdot\sum_{j\in\mathscr{L}}\sum_{t\in\mathscr{D}_s} x_{jit}\right)=0, \qquad \forall i\in\mathscr{L},\forall s\in\mathscr{S}^V.$$

Die Einhaltung der Bedingungen die durch partiell erneuerbare Ressourcen modelliert sind geschieht durch die Überprüfung, ob die Bedarfe der eingeplanten Spiele für keine der Perioden die Kapazität der Ressourcen überschreiten. Die Summe der Bedarfe muss kleiner oder gleich der jeweiligen Kapazität sein:

$$\left(\sum_{i\in\mathscr{L}}\sum_{j\in\mathscr{L}} k_{ijr}\right)\cdot\left(\sum_{t\in\mathscr{D}_s} x_{ijt}\right)\leq K_{rs}, \qquad \forall r\in\mathscr{R},\forall s\in\mathscr{S}^G.$$

3.1.2 Österreichische Fussball-Bundesliga

Aufgrund der unterschiedlichen Anforderungen des `DFB` und des `ÖFB` muss das im vorherigen Abschnitt entwickelte Modell für die Erstellung eines Spielplans für die Österreichische Fussball-Bundesliga angepasst werden.

Anstatt die komplementären Spiele aus der Hin- und Rückrunde zu überprüfen, müssen die Indizes der Runden s und $\bar{s}$ mit den bekannten Komplementaritäten explizit angegeben werden. Dies geschieht mit folgender Bedingung:

$$\sum_{t\in\mathscr{D}_s} x_{ijt}-\sum_{t\in\mathscr{D}_{\bar{s}}} x_{jit}=0, \qquad \forall i,j\in\mathscr{L},\forall s\in\mathscr{S}^H.$$

Die Saison ist in vier Teile unterteilt, wobei die beiden Hälften beinahe unabhängig voneinander eingeplant werden. Es muss jedoch überprüft werden, dass keine zwei Mannschaften innerhalb der Runden 17 bis 20 zweimal gegeneinander spielen. Dies wird gewährleistet durch die Ungleichung:

$$\sum_{s=17}^{20}\sum_{t\in\mathscr{D}_s}(x_{ijt}+x_{jit})\leq 1, \qquad \forall i,j\in\mathscr{L}.$$

Bei der Berechnung der Zielfunktion muss die Bestimmung der Verstöße gegen wechselnde Gegnerstärke derart verändert werden, dass die gesamte Saison betrachtet wird. Anstatt nur über alle $s\in\mathscr{S}^H$ zu summieren müssen alle $s=\{1,\dots,S-1\}$ berücksichtigt werden.

Kapitel 4

Lösungsalgorithmen

Dieses Kapitel stellt Algorithmen zur Erstellung von Spielplänen mit den definierten Bedingungen vor. In den Abschnitten 4.1 bis 4.5 werden die von Bartsch et al. verwendeten Verfahren vervorgestellt. Hierbei wird die Spielplanerstellung in mehrere Teilprobleme unterteilt, die unabhängig voneinander gelöst werden können. Im Abschnitt 4.6 wird ein kurz ein Verfahren mittels Integer Programmierung vorgestellt, das eine ganzheitliche Lösung des Problems liefert.

4.1 Grundlegender Algorithmus

Die Spielplanerstellung für eine große Sportliga mit vielen Nebenbedingungen ist sehr komplex und erstreckt sich über einen großen Suchraumi, in dem nur wenige Lösungen zulässig sind. Eine Möglichkeit dieses Prolem anzugehen liegt in der Unterteilung in voneinander unabhängige Teilprobleme:

- Überprüfung auf Zulässigkeit mit minimaler Breakanzahl bzgl.Stadionverfügbarkeit,

1. Erstellung eines Schlüsselplans,

2. Zuordnung der Mannschaften zu Schlüsseln,

3. Verteilung der Spiele auf die Tage einer Runde.

Nachdem für die Stadionverfügbarkeiten überprüft wurde, ob sie minimale Breakanzahl in komplementärer Hin- und Rückrunde erlauben, wird im ersten Teil des tatsächlichen Algorithmus ein Schlüsselplan erstellt. Ein Schlüsselplan für eine Liga $\mathscr{L}$ mit N Mannschaften erstellt einen Spielplan bestehend aus Opponent Plan und Home-Away-Pattern für Schlüssel anstatt Mannschaften. Die Bedingungen des grundlegenden Modells werden eingehalten sowie die minimale Anzahl an Breaks, verbotene Breaks in bestimmten Runden und die komplementäre Hin- und Rückrunde. Die Generierung eines Schlüsselplans kann auf graphentheoretische Grundlagen zurückgeführt werden (vgl. [5]). Betrachtet man den Spielplan als einen vollständigen Graphen

K_N, dann lassen sich mit der kanonischen 1-Faktorisierung des Graphen Spielpläne mit balancierter minimaler Breakanzahl erzeugen (vgl. [4]). Ein Spielplan entspricht der Partitionierung der Kanten des Graphen mittels Kantenfärbung in $N/2$ nicht-adjazente Mengen $(F_1, \ldots, F_(N-1))$. Jede Farbe, bzw. Menge enspricht den Spielen eines Spieltags, die Ausrichtung der Kanten bestimmt Heim- und Auswärtsspiele. Die Kantenmengen sind gegeben durch

$$F_i = \{\{N, i\}\} \cup \{\{(i+k) \mod (N-1), (i-k) \mod (N-1)\} | k = 1, \ldots, N/2 - 1\}.$$

Die Orientierung der Kanten ist

(i, N) bei geradem i,	(N, i) bei ungeradem i,
$(i+k, i-k)$ bei geradem k,	$(i-k, i+k)$ bei ungeradem k.

Für eine graphische Beschreibung der 1-Faktorisierung und ein Beispiel eines so generierten Spielplans mit $N = 6$ Mannschaften sei auf [5] verwiesen. Mehrere Schlüsselpläne lassen sich mit dem Algorithmus `ENUMOPP` (vgl. Appendix A.1) generieren.

Phase 2 des Algorithmus ordnet die Mannschaften unter Berücksichtung der gegebenen Bedingungen den Schlüsseln zu. In Abschnitt 4.2 wird eine Semi-Greedy Heuristik hierfür vorgestellt, in Abschnitt 4.3 und 4.4 folgen dann beschränkte und exakte Branch-and-Bound Algorithmen.

Im letzten Schritt werden die Spiele den einzelnen Tagen einer Runde zugeordnet. Dabei werden zuerst an jedem Spieltag die sicherheitsrelevanten Spiele und jene bei denen Einschränkungen aufgrund von Mindestabständen wegen Pokalspielen bestehen eingeplant. Anschließend folgen Spiele mit Regions-Einschränkungen und dann alle restlichen Spiele. Abschließend werden Ressourcenbeschränkungen überprüft. Verstöße werden durch Tauschoperationen behoben, sofern möglich. Andernfalls terminiert die Funktion ohne Lösung.

4.2 Semi-Greedy Algorithmus

In Phase 2 des Spielplangenerierungsalgorithmus werden die Mannschaften nacheinander den Schlüsseln des Schlüsselplans unter Berücksichtigung der Ressourcenbeschränkungen bzgl. der Runden zugeordnet. Dazu wird die Funktion `KEYPATTERN` aus Appendix A.1 verwendet. Ausgehend von allen Mannschaften als initiale Liste $\mathscr{L}^{free}$ der Mannschaften i, die noch keinem Schlüssel zugeordnet wurden, werden der Reihe nach allen Mannschaften aus $\mathscr{L}^{free}$ Home-Away-Pattern aus den Menge $\mathscr{H}_i^{free}$ zugeordnet. $\mathscr{H}_i^{free}$ beinhaltet die Pattern des Spielplans, die noch keiner Mannschaft zugeordnet wurden und die zu i zugeordnet gegen keine Stadionunverfügbarkeiten verstoßen.

Die Auswahl der Mannschaft, die als nächste eingeplant werden soll geschieht nach der Prioritätsregel ω. ω favorisiert Mannschaften mit wenigen verfügbaren Pattern. Dazu fungiert `KEYPATTERN` als *Multi-Pass* Prozedur, die von Kolish und Drexl in [7] für RCPSP-Instancen untersucht wurde. In Z einzelnen Durchläufen werden maximal Z eindeutige zulässige Lösungen erzeugt. Dabei wird die Prioritätsregel ω als Zufallsvariable verwendet, durch die mit der Zufallsverteilung $\Omega : i \in \mathscr{L}^{free} \rightarrow [0, 1]$ jeder nicht zugeordneten Mannschaft $i \in \mathscr{L}^{free}$ eine

Wahrscheinlichkeit zugeordnet wird, mit der i ausgewählt wird. So bewirken mehrere Durchläufe das unterschiedliche Mannschaften ausgewählt werden. Beim Regret-Based-Bias-Sampling ist abhängig vom Bias α der Zufallsverteilung die Zufälligkeit der Auswahl größer als bei kleinerem α während bei großem α die Auswahl deterministisch erfolgt und somit in jedem Durchlauf die gleichen Mannschaften ausgewählt werden. Regret-Based-Bias-Sampling basiert auf der Idee die Auswahl abhängig vom Ergebnis zu treffen, das durch die Auswahl erreicht wird. ω_i sorgt also dafür, dass i ausgewählt wird, wenn es besonders schlecht ist i nicht auszuwählen, also wenn i aufgrund der geringen Anzahl von verfügbaren Pattern eventuell nicht mehr eingeplant werden kann.

$$\omega'_i := \max_{j \in \mathscr{L}^{free}} \left| H_j^{free} \right| - \left| H_i^{free} \right|, \quad (\forall i \in \mathscr{L}^{free})$$

ordnet den Mannschaften mit den wenigstens freien Pattern die höchste Priorität zu. Einführung der Konstanten $\varepsilon > 0$ garantiert, dass die Auswahlwahrscheinlichkeit jeder Mannschaft größer als Null ist (in folgenden Experimenten ist $\varepsilon = 1$) und der Bias α bestimmt die Varianz der Verteilung:

$$\omega''_i = (\omega'_i + \varepsilon)^\alpha, \quad (\forall i \in \mathscr{L}^{free}).$$

Als Zufallsverteilung der Prioritätswerte ergibt sich dann

$$\Omega_i = \frac{\omega''_i}{\sum_{j \in \mathscr{L}^{free}} \omega''_j}, \quad (\forall i \in \mathscr{L}^{free}).$$

Aus der Menge der verfügbaren Pattern für Mannschaft i wird das ausgewählt, das für die wenigsten Mannschaften zulässig ist. Die Auswahl geschieht aus laufzeittechnischen Gründen deterministisch mit der Prioritätsregel

$$\psi: \text{ Wähle } p \in \mathscr{H}_i^{free}, \text{ so dass } \left| \mathscr{L}_p^{free} \right| = \min_{p' \in \mathscr{H}^{free}} \left| \mathscr{L}_{p'}^{free} \right|.$$

4.3 Beschränkter Branch-and-Bound Algorithmus

Eine weitere Variante zur Lösung eines RCPSP sind Branch-and-Bound Verfahren. Hierbei wird ein Baum über den Lösungsraum aufgespannt. Das Prinzip besteht darin, die Suche nach einer Lösung in Teilprobleme zu unterteilen und von diesen durch Verwendung von Schranken möglichst viele zu eliminieren [3].

Bartsch et al. [1] verwenden Tiefensuche in einem Baum aus Mannschaften i, der nach Regel τ erstellt wird. τ entspricht ω insofern dass für jeden Knoten k Nachfolgeknoten erzeugt werden, die jene Mannschaften repräsentieren, welche die wenigsten Home-Away-Pattern zur Verfügung haben. Beginnend bei der Wurzel werden den Mannschaften Schlüsselpattern zugeordnet. Dabei muss eine Pattern p das i zugeordnet mit den Stadionverfügbarkeiten von i übereinstimmen. Ist i in einer Region mit einer geraden Anzahl von nicht zugeordneten Mannschaften, muss außerdem das zu p komplementäre Pattern p' einer Mannschaft j aus der gleichen Region zulässig zugeordnet werden können, ohne dass Stadioneinschränkungen verletzt werden. Die Pattern die beide Bedinungen einhalten werden in $\mathscr{H}_i^{free*}$ abgelegt. Auf diese Weise werden Äste die zu

Unzulässigkeit führen vermieden. Das zu verwendende Pattern aus $\mathscr{H}_i^{free*}$ wird mit Regel ψ ausgewählt.

Untere Schranken werden verwendet um weitere Äste des Baumes zu eliminieren, mit denen kein besseres Ergebnis erzielt werden kann. Konkret heißt das, mit einer unteren Schranke wird abgeschätzt, wie gut das Ergebnis des aktuellen Astes sein kann. Ist es schlechter als das beste bereits erzielte Ergebnis, so wird ein Ast nicht weiter verfolgt. Als untere Schranke werden hier die Bedingungen der Zielfunktion verwendet. Die Anzahl der unerfüllten Heimpräferenzen eines Astes wird abgeschätzt mit

$$LB_n^{\hat{H}} = \sum_{i \in \mathscr{L}^{free}} \hat{H}_i(p) x_{pi} + \sum_{i \in \mathscr{L}^{free}} \left(\min_{p \in \mathscr{H}_i^{free*}} \hat{H}_i(p) \right).$$

$\hat{H}_i(p)$ gibt die Anzahl der unerfüllten Heimpräferenzen von i an, wenn p i zugeordnet wird. Als weitere untere Schranke $LB_n^{\hat{H}}$ wird die Anzahl der Verstöße gegen wechselnde Gegnerstärke bestimmt. Diese beiden unteren Schranken werden mit den Gewichten der Zielfunktion zu einer weiteren Schranke kombiniert

$$LB_n^{\hat{H}\hat{W}} = \hat{\omega}_1 \cdot LB_n^{\hat{H}} + \hat{\omega}_2 \cdot \hat{H}_i(p).$$

Wird in Schritt 2 kein zulässiger Plan erzeugt, so ist damit Unzulässigkeit der Bedingungen bewiesen.

4.4 Exakter Branch-and-Bound Algorithmus

Der exakte Branch-and-Bound Algorithmus unterscheidet sich von der bereits vorgestellten beschränkten Version hauptsächlich in der Generierung der Schlüsselpläne. Im ersten Schritt des Algorithmus werden nun alle Home-Away-Pattern mit minimaler Anzahl dan Breaks generiert. Rekursiv werden dann für alle Home-Away-Pattern gültige Opponent Pläne erzeugt. Ergebnisse zur Anwendbarkeit dieses Verfahrens folgen im Kapitel 5.1.

4.5 Österreichische Fussball-Bundesliga

Für die Österreichische Fussball-Bundesliga kann der grundlegende Plan vereinfacht werden. Der letzte Schritt, die Zuordnung der Spiele zu Tagen einer Runde entfällt. Schlüsselpläne werden durch zufällige Generierung von Home-Away-Pattern erstellt. Näheres dazu findet sich in [1]. Die Zuordnung von Mannschaften zu Schlüsseln geschieht mit dem heuristischen Verfahren.

4.6 Minimierung mittels Integer Programmierung

In [2] wenden Briskorn und Drexl den Standardproblemlöser für Integer Lineare Programme *Cplex* auf das hier beschriebene Modell an. Auf einem 3.8 GHz Pentium 4 Computer mit 3 Gigabyte Arbeitsspeicher lässt sich bereits das grundlegende Modell ohne zusätzliche Bedingungen mit $N \geq 18$ Mannschaften nicht mehr optimal lösen. Untersuchungen an Modellen mit weiteren Einschränkungen liefern keine eindeutigen Ergebnisse. Im Allgemeinen gilt, dass zusätzliche Einschränkungen den Suchraum verkleinern außerdem aber die Anzahl der zulässigen Lösungen verringern. Dies führt zu extrem langen Laufzeiten und erheblichem Speicheraufwand besonders bei großen Ligagrößen. Aus diesem Grund ist das dort vorgeschlagene Verfahren in dieser Form nicht anwendbar für die Spielplanerstellung der Deutschen und Österreichischen Fussball-Bundesligen.

Kapitel 5

Experimente und Ergebnisse

In diesem Kapitel werden die Ergebnisse der Evaluation der im Kapitel 4 vorgestellten Algorithmen zur Generierung von Spielplänen für Sportligen vorgestellt. Die Experimente sind in zwei Bereiche unterteilt. Zuerst wird an künstlich generierten Testinstanzen unterschiedlicher Größe und Schwierigkeit die Effizienz der Algorithmen überprüft. Anschließend werden tatsächliche Spielpläne basierend auf den Anforderungen des `DFB` und des `ÖFB` erstellt um die praktische Anwendbarkeit für reale Szenarien zu überprüfen.

5.1 Experimentelle Evaluation

Dieser Abschnitt stellt die Ergebnisse der im vorherigen Kapitel vorgestellten Algorithmen mit eigens für Testzwecke generierten Instanzen vor. Basierend auf den Anforderungen des `DFB` werden zufällig Instanzen generiert. Die Lösungsalgorithmen werden im Bezug auf zwei Charakteristiken dieser Instanzen evaluiert, die Größe und die Lenkbarkeit.

Die Instanzen $\gamma \in s, m, l$ unterscheiden sich in ihrer Größe. *Kleine* Instanzen ($\gamma = s$) bestehen aus 4 bis 8, *mittlere* ($\gamma = m$) aus 10 bis 14 und *große* aus 16 bis 20 Mannschaften. Die Lenkbarkeit $\sigma \in \{e, t\}$ ist ein Maß für die Größe des Lösungsraums. Dieser Wert ist stark abhängig von der Anzahl der Einschränkungen. Bartsch et al. verwenden die Stadionverfügbarkeit als Begrenzung für den Suchraum. Bei *einfachen* Instanzen ($\sigma = e$) sind alle Stadien zu jedem Zeitpunkt verfügbar sind während bei *schweren* Instanzen ($\sigma = h$) jedes Stadion einmal nicht verfügbar ist, wobei komplementäre Home-Away-Pattern innerhalb einer Region berücksichtigt werden müssen. *Sehr schwere* Instanzen verfügen über zusätzliche Bedingungen, die aber in vielen Fällen zu unlösbaren Problemen führen und deswegen keine sinnvolle Evaluation erlauben.

Die Experimente beinhalten die zufällige Generierung von 10 Instanzen jeder Größe und Lenkbarkeit. Die Anwendung der Algorithmen auf die Instanzen entspricht einem Zufallsexperiment. Ein Durchlauf des Semi-Greedy Algorithmus besteht aus 1000 Iterationen, d.h. 1000 Versuchen mit Hilfe von maximal 3000 Zuordnungen von Home-Away-Pattern zu Mannschaften

Efficiency rates and CPU times—semi-greedy algorithm

γ	α	Efficiency rate	Standard deviation	CPU time (in seconds)	Standard deviation
s	0	0.9827	0.0613	8.8	10.0
	2	0.9827	0.0613	10.1	12.9
	4	0.9827	0.0613	10.9	15.9
	8	0.9827	0.0613	11.2	16.3
m	0	0.8348	0.2268	899.9	1394.4
	2	0.8330	0.2223	1296.8	2120.9
	4	0.8286	0.2208	1752.0	3424.5
	8	0.8123	0.2211	2118.1	4518.2
l	0	0.9139	0.1461	9999.2	13,276.3
	2	0.8902	0.1420	11,937.4	15,545.9
	4	0.8970	0.1453	13,052.7	17,701.6
	8	0.8768	0.1513	13,341.4	18,186.0

einen zulässigen Spielplan zu erzeugen. Der beschränkte Branch-and-Bound Algorithmus wird nach maximal 1000 CPU Minuten abgebrochen.

Die Zufallsvariablen eines jeden Experiments sind der beste erreichte Zielfunktionswert und die benötigte CPU Zeit. Aus ihnen lässt sich die Effizienzrate des Algorithmus als Durchschnittswert der Quotienten aus bestem Zielfunktionswert und benötigter CPU Zeit der einzelnen Durchläufe berechnen. Ein weiterer Messwert für die Evaluation ist die durchschnittliche benötigte Laufzeit für ein Experiment. Tabellen 4 bis 7 stellen die Ergebnise der in C implementierten Algorithmen dar, die unter Unix auf einem IBM RS 6000 T41 getestet wurden.

Tabelle 4 demonstriert die Effekte des durch α gegebenen Bias für den Semi-Greedy Algorithmus. Für alle gegebenen Instanzgrößen verbessern sich Effizienz und Laufzeit je kleiner α ist. Das heißt je weniger deterministisch Prioritätsregel ω die Mannschaften auswählt desto besser sind die erzielten Ergebnisse, da mehr unterschiedliche Spielpläne erzeugt werden können. Im Folgenden wird daher $\alpha = 0$ verwendet.

Tabellen 5 bis 7 vergleichen die drei Algorithmen für die unterschiedlichen Instanzgrößen und Lenkbarkeitswerte. Der exakte Branch-and-Bound Algorithmus liefert für jede der kleinen Instanzen eine optimale Lösung. Bereits bei den mittleren Instanzen liefert er in der gegebenen Zeit für keine der Instanzen eine optimale Lösung. Deswegen beschränken sich Tabelle 6 und 7 auf einen Vergleich zwischen beschränktem Branch-and-Bound und Semi-Greedy Algorithmus. Für mittlere und große Instanzen benötigt der Branch-and-Bound Algorithmus deutlich mehr Laufzeit. Nur bei zweien der großen Instanzen erzielte er in der gegebenen Zeit ein suboptimales Ergebnis. Bei einfachen großen Instanzen liefert er trotz längerer Laufzeit schlechtere Ergebnisse als das heuristische Verfahren und auch bei schweren mittleren und großen Instanzen ist die Effizienzrate nur leicht über der des Semi-Greedy Verfahrens.

Jeder der Algorithmen hat seine Vorteile für unterschiedliche Anwendungsklassen. Da bei praktischen Anwendungen für die Österreichische und Deutsche Fussball-Bundesliga überwie-

Efficiency rates and CPU times—small instances

σ	Procedure	Efficiency rate	Standard deviation	CPU time (in seconds)	Standard deviation
e	Exact B&B	1.00	0.00	3888	7384
	Truncated B&B	0.70	0.28	10	20
	Greedy heuristic	0.68	0.30	12	10
t	Exact B&B	1.00	0.00	338	667
	Truncated B&B	0.67	0.25	< 1	< 1
	Greedy heuristic	0.67	0.25	12	10

Table 6
Efficiency rates and CPU times—medium instances

σ	Procedure	Efficiency rate	Standard deviation	CPU time (in seconds)	Standard deviation
e	Truncated B&B	1.00	0.00	19,834	23,909
	Greedy heuristic	0.51	0.19	1172	1285
t	Truncated B&B	1.00	0.00	3381	7768
	Greedy heuristic	0.92	0.12	1528	1787

gend schwere Instanzen mittlerer und großer Größer erwartet werden, wird hierfür das heuristische Semi-Greedy Verfahren angewandt. Dieses erzielt bei diesen Instanzen ähnlich gute Ergebnisse wie der beschränkte Branch-and-Bound Algorithmus, benötigt aber deutlich kürzere Laufzeit. Der Algorithmus wurde in ein interaktives Decision Support System (DSS) basierend auf einer Paradox-Datenbank integriert. Das DSS verfügt über eine graphische Oberfläche, die eine schnelle Anpassung der Parameter, beispielsweise des α Wertes, ermöglicht, wenn das DSS nicht sofort eine Lösung liefert, die alle Anforderungen berücksichtigt. Auf diese Weise kann schnell eine zufriedenstellende Lösung gefunden werden.

5.2 Anwendungsbeispiel `DFB`

Die Spielplänerstellung für den `DFB` konnte anhand von zwei Instanzen evaluiert werden. Für die Saisons 1996/97 und 1997/1998 standen Bartsch et al. die Anforderungen des `DFB` zur Verfügung. Für 1996/97 konnte das Ergebnis des DSS mit dem manuell erstellten Spielplan des `DFB` verglichen werden während für 1997/98 der automatisch generierte Spielplan tatsächlich zum Einsatz gekommen ist.

Die manuelle Spielplanerstellung ist ein aufwendiges Verfahren, das viel Zeit in Anspruch nimmt [6]. Dabei wird ein vorgegebener Schlüsselplan mit minimaler Breakanzahl verwendet, wobei jedes Jahr die Mannschaften manuell den Schlüsseln zugeordnet werden. Es werden hauptsächlich Stadionsperren berücksichtigt. Sobald alle Mannschaften zugeordnet sind und ein

Instances—DFB

Characteristics		Season				
		1996/97			1997/98	
		RE	SG	MA	RE	SG
O1	Stadium availability	28	28	27	20	20
O2	Regions	4	4	4	4	4
O3	Security aspects	•	•	∘	•	•
A1	Breaks	•	•	•	•	•
A2	Min no. days/matches	•	•	•	•	•
A3	Home preferences	3	2	2	3	3
A4	Attractive games	•	•	∘	•	•
A5	Fixed matches	5	3	0	3	3
A6	Min/max # of games	•	•	•	•	•
F1	Days without games	•	•	•	•	•
F2	Opponent strength	∘	∘		∘	∘
F3	Forbidden breaks	•	•	•	•	•
	Application			•		•

akzeptabler Spielplan entstanden ist, wird dieser angenommen. Ein ungefähres Maß für die benoötigte Zeit sind 15 Stunden, die aber je nach Anzahl der Anforderungen stark variieren können. Auch Versuche mit Excel-Tabellen führten zu keiner deutlichen Verbesserung. Der Semi-Greedy Algorithmus benötigte hingegen 54 CPU Sekunden um fünf zulässige Lösungen zu erzeugen.

Tabelle 8 stellt die Ergebnisse für die beiden untersuchten Saisons dar. Die Spalten `RE` dokumentieren die Anforderungen des `DFB`. Ein • bedeutet, dass die Anforderung erfüllt werden muss, eine Zahl gibt die Anzahl der entsprechenden Bedingungen an. Spalte `MA` stellt die Charakteristik des manuell erstellten Spielplans dar und die Spalten `SG` die mit dem Algorithmus generierten. Hier bedeutet ein • jeweils, dass die Anforderung erfüllt wurde, bei einem ∘ nur teilweise. Die Zahlen geben die Anzahlen der erfüllten Bedingungen an. Im Folgenden werden einige der Anforderungen genauer beschrieben.

Für die Saison 1996/97 ignorierte der `DFB` eine Stadionunverfügbarkeit, so dass eine Mannschaft ein Heimspiel in einem Stadion hatte, in dem zeitgleich Renovierungsarbeiten stattfanden. Des Weiteren wünschte der `SV Werder Bremen` fünf Heimspiele gegen attraktive Gegner zum Ende der Saison nach Fertigstellung der Stadionvergrößerung um durch Zuschauereinnahmen einen Gewinn von bis zu 1 Millionen Euro zu erzielen. Der `DFB` erstellte einen Spielplan ohne ein einziges Topspiel in dieser Zeit. Das DSS stellte fest, dass zumindest 3 dieser Spiele möglich gewesen wären.

Der manuell erstellte Spielplan berücksicht wechselnde Gegnerstärke nur zu 29 Prozent, während der Semi-Greedy Algorithmus diese zu 72 Prozent berücksichtigt. Dies führt zu einer Verbesserung der Zielfunktion um 35 Prozent. Mit dem beschränkten Branch-and-Bound Verfahren wurde sogar eine Verbesserung um 47 Prozent erzielt.

Table 9
Instances—ÖFB

Characteristics		Season							
		1997/98		1998/99		1999/00		2000/01	
		RE	SG	RE	SG	RE	SG	RE	SG
O1	Stadium availability	2	2	2	2	4	4	7	7
O2	Regions	3	3	2	2	3	2	4	2
A1	Breaks	•	•	•	•	•	•	•	•
A2	Min no. days/matches	•	•	•	•	•	•	•	•
A3	Home preferences	3	1	7	5	9	9	11	9
A5	Fixed matches . . .								
A5a	Within 1–3 rounds	4	3	6	3	13	4	3	3
A5b	Within 4–8 rounds	1	1	1	1	10	10	5	4
F2	Opponent strength	○	○	○	○	○	○	○	○
F3	Forbidden breaks	•	•	•	•	•	•	•	•
	Application		•		•		•		•

Für die Saison 1997/98 mussten einige Anforderungen der übertragenden Fernsehsender erfüllt werden. In der ersten Runde sollten zwei attraktive Spiele und am letzten Spieltag ein Topspiel stattfinden. Neben diesen Anforderungen wurden auch alle 20 Stadionunverfügbarkeiten berücksichtigt.

Trotz positiver Reaktionen auf die generierten Spielpläne kam das DSS beim `DFB` nicht weiter zum Einsatz.

5.3 Anwendungsbeispiel ÖFB

Für die Österreichische Fussball-Bundesliga wurde das DSS sechsmal zur Generierung der Spielpläne verwendet. Die Erfüllung der Anforderungen ist in Tabelle 9 dargestellt. Für die Anforderung `A5` gibt es zwei verschiedene Ausprägungen. Soll ein Spiel in einer Periode von maximal 3 Runden stattfinden, so ist dies eine Anforderung der Klasse `A5a`, bei 4 bis 8 möglichen Runden eine Anforderung der Klassen `A5b`. Es folgen einige Erläuterungen zu den einzelnen gelösten Problemen, insbesondere auch im Bezug auf unlösbare Konflikte.

1997/98: Zur Einweihung des “Arnold Schwarzenegger-Stadions” in Graz sollte ein Spiel der beiden Grazer Mannschaften in der ersten Runde stattfinden. Das bislang benutze Schlüsselpattern des `ÖFB` enthielt keine Spielpaarung zweier Mannschaften mit komplementären Home-Away-Pattern am ersten Spieltag. Mit dem DSS konnte ein neues Schlüsselpattern generiert werden, das diese Anforderun erfüllte.

1998/99: Derbys zwischen Grazer und Wiener Mannschaften sollten im Herbst und Frühling angesetzt werden, da dort mehr Zuschauer zu erwarten sind als im Winter.

1999/00: Aufgrund zahlreicher Anforderungen von Spielen in festen Runden entstanden unlösbare Konflikte. Der ÖFB war somit gezwungen die Anforderungen mit Prioritäten zu belegen. Zum Beispiel waren bei den beiden Wiener Mannschaften aufgrund von Stadionunverfügbarkeiten keine komplementären Home-Away-Pattern möglich. Stattdessen wurden Pattern gewählt, bei denen die Stadionunverfügbarkeiten berücksichtigt wurden und die Pattern an möglichst vielen Spieltagen komplementär waren.

2000/01: Die beiden favorisierten Mannschaften sollten in der letzten Runde gegeneinander spielen. Bei zwei der vier Regionen wurden keine strikt komplementären Pattern der beiden Mannschaften gefordert. Stattdessen sollte die Anzahl der Spieltage an denen die beiden Mannschaften der Region gleichzeitig Heimspiele haben mininimiert werden.

2001/02: Eine große Anzahl von Heimspielwünschen und auf bestimmte Runden fixierter Spiele konnte erfolgreich berücksichtigt werden.

2002/03: Von den 17 blockierten Stadien konnten 16 berücksichtigt werden. Der eine Verstoß war bewiesenermaßen unumgänglich. Der ÖFB setzte das Spiel an einem zusätzlichen Spieltag an. Von den 10 festen Spielen konnten 8 zufriedenstellend eingeplant werden.

Kapitel 6

Zusammenfassung

Die Erstellung von Spielplänen für Sportligen ist eine große Herausforderung. Neben zahlreichen organisatorischen Hürden gilt es auch Bedingungen zur Fairness zu berücksichtigen und dabei auch Konflikte mit Spielen aus anderen Sportligen, und Pokalwettbewerben zu beachten.

Bartsch et al. behandeln in ihrem Artikel drei wichtige Aspekte der Spielplanerstellung. In Zusammenarbeit mit den Fussballverbänden aus Österreich und Deutschland erstellen sie eine Liste der typischen Anforderungen an einen Sportligaspielplan, wie sie aus Erfahrung der Verantwortlichen von `DFB` und `ÖFB` abgeleitet werden können. Darunter fallen auch Anforderungen, die nicht offensichtlich sind, sondern bei der manueller Spielplanerstellung nach langer Erfahrung intuitiv gemacht werden.

In der zugrundeliegenden Arbeit wird ein komplettes Modell für die Anforderungen als Integer Lineares Programm, d.h. als Zielfunktion mit Nebenbedingungen definiert. Dieses Modell ist eine formelle Beschreibung der Problemstellung unter Berücksichtigung aller Anforderungen.

Der dritte Beitrag von Bartsch et al. ist die Vorstellung von Algorithmen, mit denen die in kurzer Zeit zulässige Spielpläne für künstliche Instanzen und auch für tatsächliche Anwendungsfälle erzeugt werden können. Die so generierten Spielpläne wurden vom `DFB` einmal und vom `ÖFB` sechsmal akzeptiert. Vergleiche zwischen manuell und automatisiert erstellten Spielplänen basierend auf den gleichen Anforderungen zeigen Vorteile des algorithmischen Verfahrens auf.

Die verwendeten Algorithmen sind keine Standardverfahren zur Lösung von Integer Linearen Programmen. In anderen Arbeiten zeigen die Autoren Schwierigkeiten der Anwendung von Lösungsverfahren von Linearen Programmen auf ihr Modell. Unter dieser Voraussetzung erscheint es wenig sinnvoll die gesamten Anforderungen als Lineares Programm zu modellieren. Stattdessen sollte die Modellerierung mit den von den Algorithmen geforderten Eigenschaften übereinstimmen.

Anhang A

Anhang

A.1 Algorithmen

ENUMOPP

KEYPATTERN:
  berechne $\mathscr{H}_i^{free}$, $\forall\ i \in \mathscr{L}$
  success := FALSE;
  IF($\exists\ i \in \mathscr{L}^{free} \wedge \mathscr{H}_i^{free} \neq \emptyset$) THEN
    wähle Team $i\ \in\ \mathscr{L}^{free}$ nach Regel ω;
    WHILE(!success $\wedge \mathscr{H}_i^{free} \neq \emptyset$ DO
      wähle $p \in \mathscr{H}_i^{free}$ nach Regel ψ;
      update $\mathscr{H}_i^{free}$;
      assignment(i, p)
      IF(assignment(i, p) zulässig) THEN
        success := TRUE;
        update $\mathscr{H}^{free}$ and $\mathscr{L}^{free}$;
        IF($\mathscr{L}^{free} \neq \emptyset$) THEN KEYPATTERN;
          IF(KEYPATTERN nicht erfolgreich) THEN
            success := FALSE;
            entziehe assignment(i, p);
            update $\mathscr{H}^{free}$ and $\mathscr{L}^{free}$;

A.2 Partiell erneuerbare Ressourcen

Tabelle A.1

Spiele pro Tag	
Ressource	Dummy-Ressource $r \in \mathscr{R}$
Kapazität	$K_{rs} = \begin{cases} Max, & \text{Anzahl Spiele in } t \in \mathscr{D}_s \\ \infty, & \text{sonst} \end{cases}$
Bedarf	$k_{ijr} = 1$

Tabelle A.2

Stadionverfügbarkeit	
Ressource	Stadion $r \in \mathscr{R}$
Kapazität	$K_{rs} = \begin{cases} 1, & \text{Stadion verfügbar in } t \in \mathscr{D}_s \\ 0, & \text{sonst} \end{cases}$
Bedarf	$k_{ijr} = \begin{cases} 1, & (i,j) \text{ spielen in Stadion } r \\ 0, & \text{sonst} \end{cases}$

Tabelle A.3

Regionen	
Ressource	Region $r \in \mathscr{R}$
Kapazität	$K_{rs} = \begin{cases} Max, & \text{Spiele pro Region } r \text{ in } t \in \mathscr{D}_s \\ \infty, & \text{sonst} \end{cases}$
Bedarf	$k_{ijr} = \begin{cases} 1, & (i,j) \text{ spielen in Region } r \\ 0, & \text{sonst} \end{cases}$

Tabelle A.4

Sicherheit	
Ressource	Sicherheitstyp $r \in \mathscr{R}$
Kapazität	$K_{rs} = \begin{cases} 0, & \text{Spiel vom Typ } r \text{ in } t \in \mathscr{D}_s \text{ verboten} \\ \infty, & \text{sonst} \end{cases}$
Bedarf	$k_{ijr} = \begin{cases} 1, & \text{bei sicherheitsrelevanten Spielen vom Typ } r \\ 0, & \text{sonst} \end{cases}$

Tabelle A.5

Attraktivität

Ressource	Attraktivitätstyp $r \in \mathscr{R}$
Kapazität	$K_{rs} = \begin{cases} Max, & \text{Anzahl Spiele vom Typ } r \text{ in } t \in \mathscr{D}_s \\ \infty, & \text{sonst} \end{cases}$
Bedarf	$k_{ijr} = \begin{cases} 1, & \text{bei attraktiven Spielen vom Typ } r \\ 0, & \text{sonst} \end{cases}$

Tabelle A.6

Spielfreie Tage

Ressource	Team $r \in \mathscr{R}$
Kapazität	$K_{rs} = \begin{cases} Max, & \text{Anzahl der Spiele von } r \text{ in } t \in \mathscr{D}_s \\ \infty, & \text{sonst} \end{cases}$
Bedarf	$k_{ijr} = \begin{cases} 1, & \text{wenn Team } r \text{ in Spiel } (i,j) \text{ spielt} \\ 0, & \text{sonst} \end{cases}$

Literaturverzeichnis

[1] T. Bartsch, A. Drexl, and S. Kröger. Scheduling the professional soccer league of Austria and Germany. *Computers & Operation Research*, 33:1907 – 1937, 2006.

[2] D. Briskorn and A. Drexl. Integer Programming Modells for Round Robin Tournaments. *Manuskripte aus den Instituten für Betriebswirtschaftslehre der Universität Kiel*, Dezember 2006.

[3] P. Brucker and S. Knust. *Complex Scheduling*. Springer, 2006.

[4] D. de Werra. Scheduling in Sports. *Studies on Graphs and Discrete Programming*, pages 381–395, 1981.

[5] A. Drexl and S. Knust. Sports League Scheduling: Graph- And Resource-Based Model. *Omega*, 35:465–471, 2007.

[6] M. Dworschak. Fussballbundesliga: Die schwierige Tüftelei um den Spielplan könnte jetzt ein neues Computerprogramm übernehmen. *Die Zeit*, Mai 1997.

[7] R. Kolish and A. Drexl. Adaptive search for solving hard project scheduling problems. *Naval Research Logistics*, 43:23–40, 1996.

Da diese Arbeit hauptsächlich eine Zusammenfassung des Artikels "Scheduling the professional soccer leagues of Austria and Germany" von Thomas Bartsch, Andreas Drexl und Stefan Kröger [1] ist, sind die dort entnommenen Informationen, inklusive jener, die dort referenziert sind, nicht explizit gekennzeichnet. Ausschließlich zusätzliche Informationen aus anderen Arbeiten sind in dieser Arbeit explizit markiert. Für weitere Informationen verweise ich auf [1].