Diplomarbeit

Ein metaheuristisches Optimierungsverfahren für Ausbildungspläne im Ärzt*innenausbildungsmanagement

ausgeführt zum Zwecke der Erlangung des akademischen Grades eines

Diplomingenieurs

unter der Leitung von

Priv.-Doz. Dr.-Ing., M.Sc., B.Sc. Fazel Ansari

(E330 Institut für Managementwissenschaften,

Forschungsgruppe: Smart & Knowledge-Based Maintenance)

Dr. techn. Thomas Sobottka

(Fraunhofer Austria Research GmbH)

Dipl.-Ing. Alexander Gaal

(Fraunhofer Austria Research GmbH)

eingereicht an der Technischen Universität Wien

Fakultät für Maschinenwesen und Betriebswissenschaft

von

Wolfgang Dummer

[redacted]

Wien, am 9. November 2022 .....

(Wolfgang Dummer)

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Ich habe zur Kenntnis genommen, dass ich zur Drucklegung meiner Arbeit unter der Bezeichnung

Diplomarbeit

nur mit Bewilligung der Prüfungskommission berechtigt bin.

Ich erkläre weiters Eides statt, dass ich meine Diplomarbeit nach den anerkannten Grundsätzen für wissenschaftliche Abhandlungen selbstständig ausgeführt habe und alle verwendeten Hilfsmittel, insbesondere die zugrunde gelegte Literatur, genannt habe. Weiters erkläre ich, dass ich dieses Diplomarbeitsthema bisher weder im In- noch Ausland (einer Beurteilerin/einem Beurteiler zur Begutachtung) in irgendeiner Form als Prüfungsarbeit vorgelegt habe und dass diese Arbeit mit der vom Begutachter beurteilten Arbeit übereinstimmt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Inhaltsverzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Danksagung

Ich danke meinen Betreuern Priv.-Doz. Fazel Ansari, Dr. techn. Thomas Sobottka und Dipl.-Ing. Alexander Gaal für ihr konstruktives Feedback sowie deren Betreuung, Begutachtung und Korrektur meiner Arbeit.
Ebenso danke ich Lisa Holzgruber und der rotable technologies GmbH für die gute Zusammenarbeit und den fachlichen Input während der gesamten Dauer meiner Arbeit.
Ich danke meinen Kommilitonen, insbesondere Noah Ladner und Lorenz Pfanner, die maßgeblich zum Erfolg in meinem Studium beigetragen haben.
Dank gilt außerdem meiner Freundin Raphaela, die mich in jeder Phase meines Studiums begleitet und unterstützt hat.
Ganz besonders danke ich meinen Eltern Birgit und Johannes, sowie meinem Bruder Alexander Dummer, die mich über mein gesamtes Studium hinweg in jeder erdenklichen Art und Weise unterstützt haben – diese Arbeit widme ich ihnen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Kurzfassung

Die Einsatzplanung in der postgradualen Ausbildung von Turnusärzt*innen stellt sich als komplexes Planungsproblem dar. Dieses Problem wird in Österreich aktuell mangels Alternativen vorwiegend händisch und unter Zuhilfenahme einfacher, ungeeigneter Hilfsmittel behandelt, was mit hohem Personal- und Zeitaufwand einhergeht. In dieser Arbeit wurde ein metaheuristisches Verfahren, basierend auf einem genetischen Algorithmus, zur automatisierten Erstellung und Optimierung von Einsatzplänen für die Ausbildung von Turnusärzt*innen entwickelt. Dazu wurden relevante Ziele und Randbedingungen unter Berücksichtigung von Stakeholder-Interessen und den gesetzlichen Rahmenbedingungen identifiziert und formuliert. Das in dieser Arbeit erzeugte Artefakt umfasst, neben einem Algorithmus zur Lösung vorliegender Probleminstanzen, Werkzeuge zur Bewertung existierender Lösungen hinsichtlich ihrer Güte und Korrektheit sowie zur Visualisierung besagter Lösungen. Das gesamte Artefakt kann in der operativen Personalplanung in österreichischen Krankenhäusern eingesetzt werden, um den künftigen Personaleinsatz in der ärztlichen Ausbildung zu planen und zu optimieren. Das entwickelte Verfahren ist dabei in der Lage, zulässige und vergleichsweise gute Lösungen für vorliegende, reale Planungsprobleme zu finden. Damit kann es als Assistenzsystem für die automatisierte Einsatzplanung verstanden werden, das sowohl gesellschaftlichen als auch wirtschaftliche Nutzen stiftet. Es trägt zur Entlastung und Einsparung erforderlicher Personalressourcen in der Planung sowie in der Ausbildung bei und kann so der aktuellen Personalknappheit im Gesundheitsbereich entgegenwirken. Das Verfahren liefert eine Basis für künftige Entwicklungen in der automatisierten Einsatzplanung von Turnusärzt*innen, und kann für den Einsatz in anderen Ländern adaptiert und hinsichtlich seiner Laufzeiteigenschaften optimiert werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Abstract

The scheduling of postgraduate medical residents is a complex planning problem. In Austria, this problem is currently handled mainly manually and with the help of simple, unsuitable aids due to a lack of alternatives, which is associated with high personnel and time efforts. In this work, a metaheuristic method, based on a genetic algorithm, was developed for the automated generation and optimization of training schedules for medical residents. For this purpose, relevant objectives and constraints were identified and formulated, taking into account stakeholder interests as well as the underlying legal framework. The artifact generated in this work includes, alongside an algorithm for solving existing problem instances, tools for evaluating existing solutions in terms of their quality and correctness, as well as for visualizing said solutions. The entire artifact can be used in operative personnel planning in Austrian hospitals to plan and optimize future personnel assignments in medical training. The developed method is able to find feasible and comparatively good solutions for existing, real planning problems. Thus, it can be considered an assistance system for automated personnel scheduling, which provides both social and economic benefits. It contributes to the relief and reduction of necessary personnel resources in planning as well as in training and thus can counteract the current shortage of personnel in the health care sector. The method provides a foundation for future developments for automated scheduling of medical residents, and can be adapted for application in other countries and optimized with regard to its runtime properties.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Kapitel 1

Einleitung

In Österreich, so wie in vielen anderen (europäischen) Staaten auch, ist eine praktische Ausbildung nach erfolgtem Studienabschluss notwendig, um die Approbation als Ärzt*in zu erwerben. Wenngleich dieser Ansatz der postgradualen Ausbildung international häufig vorkommt, unterscheiden sich die Rahmenbedingungen dieser Ausbildungen auf nationaler Ebene teils stark [1]. Diese Ausbildung erfolgt in Österreich in Krankenhäusern und bringt aufgrund ihrer inhärenten Struktur erhebliche Komplexität mit sich, was ihre Planung angeht. Aktuell wird die erforderliche Planung der ärztlichen Ausbildung manuell und unter Zuhilfenahme einfachster, meist ungeeigneter Hilfsmittel vollzogen. Dabei ist das Planungspersonal mit einer hohen Problemkomplexität und damit verbundenen Schwierigkeiten, was die Beurteilung der Güte eines erstellten Plans angeht, konfrontiert. Demzufolge ist die Erstellung eines „guten“ Ausbildungsplans schwierig und dessen Güte stark vom planenden Personal und dessen Fähigkeiten und Erfahrungen abhängig. Im Zuge der vorliegenden Arbeit soll eine automatisch optimierende Planungsmethode für die Erstellung von Ausbildungsplänen entwickelt werden.

1.1 Begriffsdefinitionen

Im Sinne einer einheitlichen Verwendung und einem eindeutigen Verständnis der verwendeten Ausdrücke werden nachfolgend die wichtigsten Begrifflichkeiten definiert

und abgegrenzt.

Personalplanung (engl. personnel planning) ... umfasst alle notwendigen Tätigkeiten, um den zukünftigen personellen Ressourcenbedarf in Qualität, Quantität und Zeit zu decken. Die Teilaufgaben der Personalplanung umfassen Bedarfsermittlung, Planung der Personalbeschaffung und -entwicklung, die Personaleinsatzplanung sowie die Planung der Personalfreisetzung. [2]

Personaleinsatzplanung (engl. staff/personnel scheduling) ... ist eine Teilaufgabe der Personalplanung und befasst sich mit der zeitlichen, qualitativen und quantitativen Zuteilung von Personen zu Aufgaben bzw. Arbeitsplätzen. Qualitativ bezieht sich hierbei auf die Abstimmung von Anforderungs- und Qualifikationsprofil [3].

Dementsprechend kann die Personaleinsatzplanung als zeitliches Ressourcenzuteilungsproblem verstanden werden.

Algorithmus ... ist eine wohldefinierte Handlungsvorschrift, die eine Menge an Eingangsgrößen in eine Menge an Ausgangsgrößen überführt. Somit kann ein Algorithmus als Werkzeug zur Lösung eines konkreten Problems verstanden werden [4].

Methode ... ist im Kontext der Algorithmik ein nicht terminierender Algorithmus (rechnergestützte Methode), also ein Algorithmus, der niemals abbricht [5].
Im allgemeinen Sprachgebrauch ist eine Methode eine Strategie bzw. Vorgehensweise zur Behandlung eines Problems. Im Kontext der vorliegenden Arbeit ist der Begriff „Methode“ gemäß letzterer Definition zu verstehen.

Turnusärzt*innen (engl. Residents) ... sind Ärzt*innen, in Ausbildung zum*zur Ärzt*in für Allgemeinmedizin oder zum*zur Fachärzt*in [6])

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

1.2 Allgemeine Einleitung in das Themenfeld

In Österreich müssen alle angehenden Ärzt*innen, nach vollendetem Studium und ungeachtet ihrer Fachrichtung, eine Ausbildung nach den geltenden gesetzlichen Bestimmungen in einer akkreditierten Ausbildungsstätte (in der Regel (i.d.R) einem Krankenhaus) durchlaufen. Die für die vorliegende Arbeit relevanten gesetzlichen Rahmenbedingungen hinsichtlich der ärztlichen Ausbildung werden in Abschnitt 2.2 genauer erörtert. An dieser Stelle sei lediglich erwähnt, dass diese Rahmenbedingungen in der Ärztinnen-/Ärzte- Ausbildungsordnung 2015 (ÄAO 2015) [6] und im Bundesgesetz über die Ausübung des ärztlichen Berufes und die Standesvertretung der Ärzte (ÄrzteG 1998) [7] geregelt sind. Diese Rechtsnormen schreiben für jede mögliche Ausbildung die qualitativen und quantitativen Erfordernisse, für deren Abschluss, also zu erwerbende Kompetenzen, Dauern, Übergangsbestimmungen, Einschränkungen etc. vor.
Im Groben können zwei Wege in der ärztlichen Ausbildung beschritten werden, eine allgemeinmedizinische Ausbildung oder eine fachärztliche Ausbildung. Diese beiden Ausbildungen teilen sich eine einheitliche neunmonatige Basisausbildung, die zu Beginn zu absolvieren ist. Nach dieser Basisausbildung müssen einzelne Ausbildungsbestandteile entsprechend der gewählten Spezialisierungsrichtung belegt und absolviert werden. Der schematische Aufbau dieser Ausbildungen ist in Abbildung 1.1 ersichtlich.

Turnusärzt*innen gehören, wie alle anderen Ärzt*innen in einer Krankenanstalt zum Personal dieser Einrichtung. Im Unterschied zum Großteil des restlichen Personals haben diese jedoch keine feste Zugehörigkeit zu einer Organisationseinheit (z.B. einer Station/Abteilung) in einem Krankenhaus, sondern müssen im Rahmen ihrer Ausbildung verschiedensten Tätigkeiten in mehreren Abteilungen nachgehen und in den jeweiligen Fachbereichen ausgebildet werden, um die zuvor erwähnten Kompetenzen zu erwerben.
In Österreich waren im Jahr 2020 insgesamt 8.017 Turnusärzt*innen in Ausbildung – Tendenz steigend [10]. Im Juni 2022 belief sich die Anzahl der, in ärztlicher Ausbildung befindlichen, Personen auf 8.872 Personen für ganz Österreich [11]. Diese Zahl mag erscheint in absoluter Betrachtung klein, setzt man sie allerdings in Relation zum gesam-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

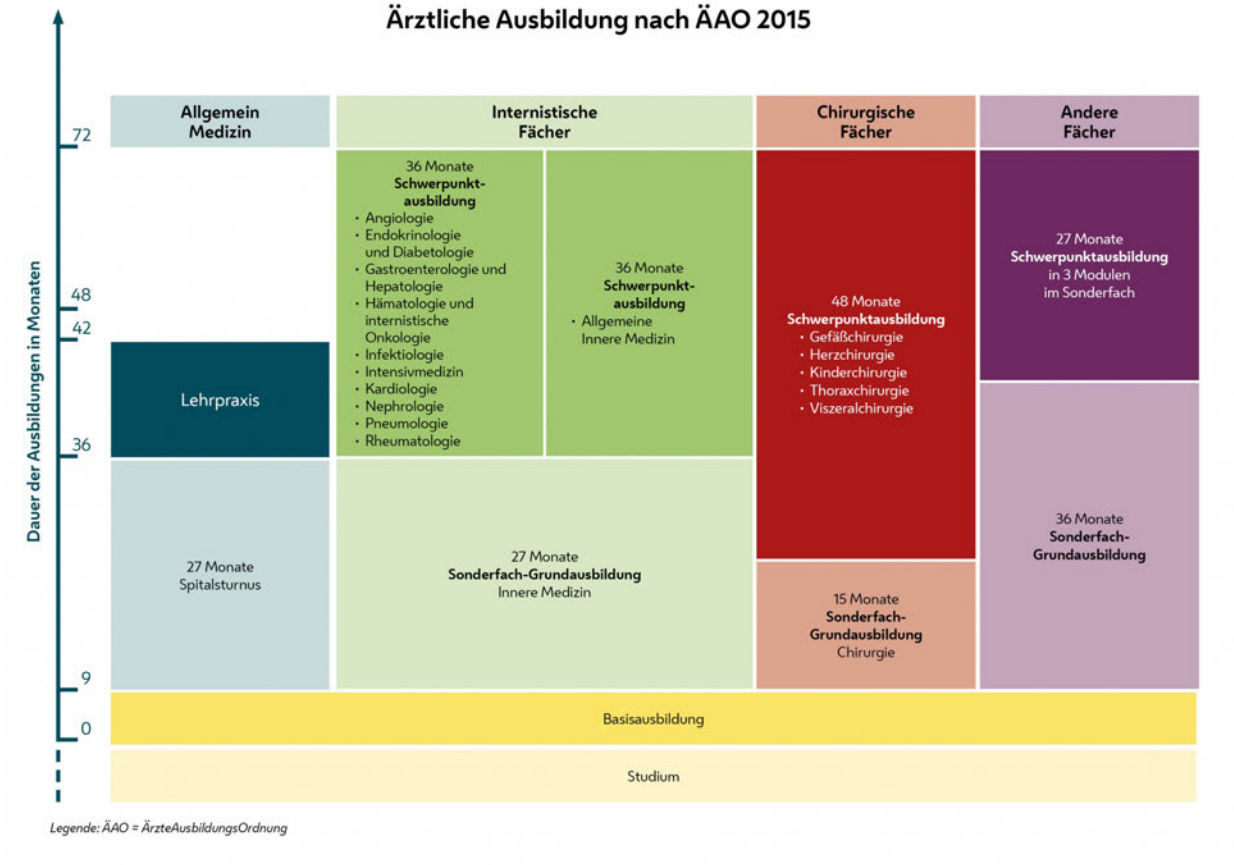


Abbildung 1.1: Postgraduale Ausbildung von Ärzt*innen in Österreich gemäß ÄAO 2015 [8]

ten Bestand an ärztlichem Personal in Österreich, so lässt sich feststellen, dass Turnusärzt*innen etwa ein Sechstel der gesamten Ärzt*innenschaft in Österreich ausmachen. (siehe Abbildung 1.2).

Im nicht niedergelassenen Bereich ist dieser Anteil noch höher, da der Großteil der Turnusärzt*innen darauf entfällt und der niedergelassene Bereich beinahe ausschließlich von bereits vollständig ausgebildeten Ärzt*innen abgedeckt wird. Somit spielen Turnusärzt*innen in der Patient*innenversorgung im Spitalswesen eine große Rolle und der Einsatz dieser wichtigen Humanressource muss mit Bedacht erfolgen. Einen essenziellen Teil zur Sicherung von Versorgungsqualität und -sicherheit kann eine zielgerichtete Einsatzplanung dieser Turnusärzt*innen beitragen.

Häufig können Turnusärzt*innen nicht ihre gesamte Ausbildung in ein und demselben Krankenhaus absolvieren, da einzelne, insbesondere kleinere Krankenanstalten nicht

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

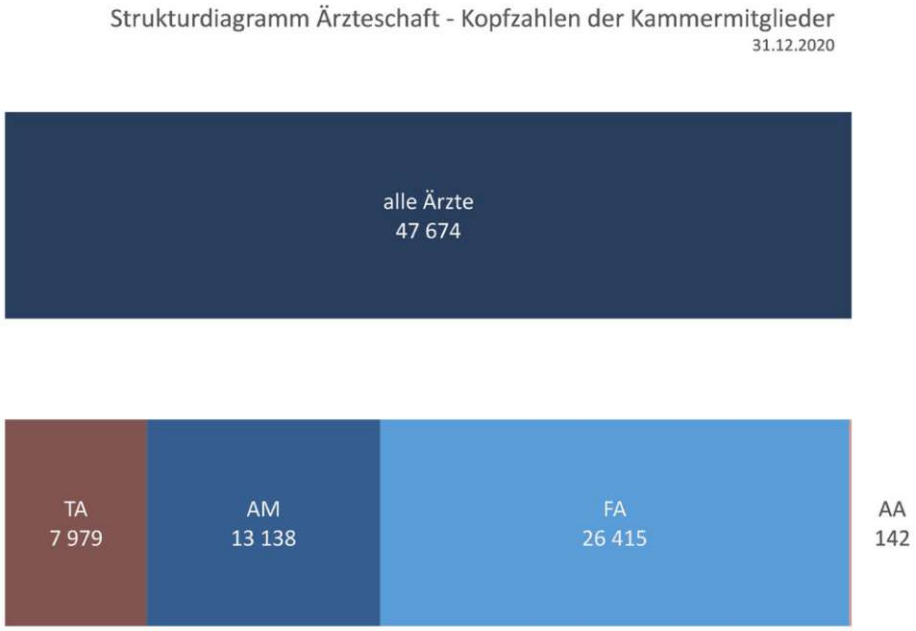


Abbildung 1.2: Verteilung der Ärzt*innenschaft in Österreich – Österreichische Ärztekammer (ÖÄK) [9]

über die gesamte, notwendige Bandbreite an medizinischem Versorgungsangebot und die damit assoziierten Kompetenzen verfügen. Somit können diese Häuser keine Ausbildungen in einer oder mehreren erforderlichen Disziplinen anbieten, weshalb Ortswechsel für eine vollständige Ausbildung notwendig sind. Diese Tatsache erfordert die Kooperation mehrerer Ausbildungsstätten und eine damit einhergehende umfassende, zentrale Planung der Ausbildungen aller betreffenden Personen in den jeweiligen Ausbildungsstätten. Denn nur so kann eine, global betrachtet, näherungsweise gleichbleibende und optimale Besetzung von Ausbildungsstellen bei gleichzeitigem Fortkommen der Einzelpersonen in ihrer Ausbildung ermöglicht werden. Die Verantwortung für die Zuteilung von Turnusärzt*innen zu Organisationseinheiten und damit zu Dienstposten, im Sinne ihrer Ausbildung, liegt somit in der Regel bei der Leitung der jeweiligen Ausbildungsstätte oder gar bei der Leitung einer übergeordneten Organisationsstruktur, wie z.B. einem Krankenanstaltenverbund oder Planungsverband. Das primäre Ziel dieser Planungsinstanz ist die optimale Einsatzplanung von Turnusärzt*innen unter Berücksichtigung mehrerer Gesichtspunkte, die üblicherweise in einem Spannungsverhältnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

zueinander stehen. Eine genauere Erläuterung dieser Kriterien erfolgt in Abschnitt 2.3.

In den vergangenen Jahren wurden die oftmals langen Wartezeiten für Ausbildungsplätze nach dem Studium oder während der Ausbildung vonseiten der österreichischen Ärztekammer ÖÄK bemängelt, die mitverantwortlich für die Abwanderung von Jungmedizinern ins Ausland und damit für den drohenden Ärzt*innenmangel in Österreich verantwortlich sein sollen [12]. Die gleichzeitig hohe Zahl unbesetzter Ausbildungsplätze sei laut ÖÄK-Vizepräsident Mayer auf einen Mangel an Dienstposten in den Ausbildungsstätten zurückzuführen, die notwendig sind, um Turnusärzt*innen anstellen und ausbilden zu können [13].
Diese Aussagen unterstreichen die Notwendigkeit eines nachvollziehbaren, übergreifenden und insbesondere ressourceneffizienten Personaleinsatzes, mit dessen Hilfe die Ausbildungsqualität verbessert und gleichzeitig die Versorgung in Österreichs Krankenhäusern gewährleistet werden kann.
Die postgraduale Ausbildung zum/zur Ärzt*in kann für jede einzelne auszubildende Person eine Vielzahl an verschiedenen Ausprägungen (Reihenfolge von Fächern, Schwerpunkte, Dienstorte etc.) annehmen. Somit ergibt sich mit der Gesamtheit der auszubildenden Turnus*ärztinnen, und deren Ausbildungsplänen eine noch viel größere Anzahl an möglichen Kombinationen aus Zuteilungen von Personen zu Ausbildungsstellen. Mit zunehmender Systemgröße (Personen, Ausbildungsstellen, Ausbildungsstätten) wächst auch die Komplexität des vorliegenden Planungsproblems, wobei diese Komplexität ein notwendiges Übel ist, das in Kauf genommen werden muss, um die zuvor benannten Ziele (Versorgung, Ausbildungsfortschritt etc.) zu erreichen.

Diese Einsatzplanung erfolgt in Österreich aktuell, laut Expertinnenangaben[1], meist manuell und unter Zuhilfenahme rudimentärer Hilfsmittel wie z.B. Tabellenkalkulationsprogrammen.

[1]Lisa Holzgruber (rotable technologies GmbH) hat die ärztliche Rotationsplanung in einem Wiener Krankenanstaltenverbund (Vinzenzgruppe) für über fünf Jahre in leitender Funktion verantwortet und ist Geschäftsführerin eines Unternehmens, das eine automatisierte Rotationsplanung in Österreich und in weiterer Folge im europäischen Raum, ermöglichen will.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

1.3 Problemstellung

Das nicht standardisierte und arbeitsintensive Vorgehen in der Planung unter Einsatz ungeeigneter Hilfsmittel, kombiniert mit der enormen Komplexität des Planungsproblems bringt eine Vielzahl verschiedener Probleme mit sich, die nachfolgend erörtert werden. Die beschriebenen Probleme wurden in Zusammenarbeit mit einer Expertin im Rahmen eines leitfadengestützten Interviews identifiziert.[2]

P 1: Zur Reduktion der Komplexität eines Planungsproblems werden häufig die geplanten Zeiträume verkürzt, was einerseits nicht konform mit der ÄAO 2015 § 21 ist und andererseits in weiterer Folge zu Stehzeiten (Beschäftigungszeiten, in denen kein Ausbildungsfortschritt erreicht wird) und einer damit einhergehenden Verlängerung der Durchlaufzeit des Personals durch die Ausbildung führt. Eine weitere gängige Methode zur Vereinfachung des Problems ist das „Aufteilen" der zu verplanenden Personen in kleinere Gruppen, die nacheinander, anstatt gleichzeitig verplant werden. Die so etappenweise erzeugten Pläne mögen zwar lokal näherungsweise optimal sein, bergen jedoch Potenzial für suboptimale Pläne in zukünftigen Planungsläufen und in einer globalen Betrachtung. Eine vollständige Betrachtung des Planungsproblems erfordert überproportional viele Personalressourcen, wobei laut Expertinnenangaben für 50 zu verplanende Ärzt*innen in etwa eine Vollzeitkraft in der Einsatzplanung notwendig ist.

P 2: Die erzeugten Ausbildungspläne sind, insbesondere durch Dritte, nur schwer nachvollziehbar, vor allem, was deren Entstehung und die dabei getroffenen Entscheidungen angeht. Da die maßgeblichen Rahmenbedingungen, die mit dem Planungsproblem einhergehen, nur schwer überprüfbar sind, wenn ein Ausbildungsplan in nicht strukturierter Form vorliegt, können somit weder Korrektheit noch Vollständigkeit eines erzeugten Plans (einfach) überprüft werden. Die damit verbundene Unsicherheit wirkt sich somit negativ auf die Planungssicherheit für die betroffenen Ärzt*innen aus, die im Falle einer Fehlplanung kurzfristig neu zugewiesen werden müssen.

[2] siehe Fußnote 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

P 3: Da sich üblicherweise jede planende Person ihre eigenen Hilfsmittel und Faustregeln zur Entscheidungsfindung zurechtlegt, um Problemkomplexität bewältigen zu können, sind die von einer Person erzeugten Pläne stark von deren Vorgehen abhängig. Diese Werkzeuge und Vorgehensweisen können sich gleichermaßen negativ auf die Fairness eines Plans aus Sicht der Ärzt*innen und die Einhaltung betrieblicher Ziele auswirken.

P 4: Für die Berechnung und Beurteilung eines Ausbildungsplans ist eine mathematische Formulierung des damit verbundenen Optimierungsproblems notwendig. Zwar wurden in der Wissenschaft für ähnliche oder artverwandte Probleme bereits mathematische Formulierungen eingeführt, für das vorliegende Problem in Österreich ist das jedoch nicht der Fall.

P 5: Zum Treffen von Planungsentscheidungen ist eine Priorisierung von Zielen notwendig, um verschiedene Entscheidungsmöglichkeiten gegeneinander abzuwägen. Bei der aktuell üblichen, manuellen Planung wird eine Prioritätensetzung vom Planungspersonal selbst vorgenommen oder implizit in Form von Entscheidungsregeln herangezogen und häufig zugunsten der Vereinfachung des Problems vernachlässigt. Damit ist die gleichzeitige Verfolgung und Anpassung mehrerer übergeordneter betrieblicher Ziele im Rahmen der Planung kaum möglich.

Die angeführten Probleme (**P 1** - **P 5**) werden nachfolgend zusammengefasst.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Probleme (Problems – **P**):

P 1 Hoher Personal- und Ressourcenaufwand bei der Erstellung von Ausbildungsplänen von Turnus*ärztinnen.

P 2 Schwer nachvollziehbare und reproduzierbare Planungsergebnisse, sowie keine Möglichkeit zur Verifikation des jeweiligen Planungsergebnisses.

P 3 Starke Abhängigkeit des Planungsergebnisses von der planenden Person (die persönliche Planungslogik und Priorisierung ist nicht notwendigerweise im Einklang mit einer „globalen" Prioritätensetzung im Krankenanstaltenverbund).

P 4 Fehlende mathematische Formulierung des Planungsproblems, sowie das Fehlen von Planungsmethoden für den Einsatz in Österreich.

P 5 Fehlende Charakterisierung, Standardisierung sowie Priorisierung der zu berücksichtigenden Kriterien bei der Verplanung von Ärzt*innen.

Somit bringt das Planungsproblem hohe Kosten bei unzureichender Planungsqualität und mangelnder Effizienz durch fehlende Standardisierung des zugrundeliegenden Prozesses mit sich, wobei von diesen Problemen sowohl Krankenhäuser und Ärzt*innen als auch Patient*innen unmittelbar betroffen sind. Zur Behandlung dieser Problematik sind am freien Markt aktuell keine geeigneten (Software-)Lösungen verfügbar.

Im Zuge dieser Arbeit soll untersucht werden, ob die Planungsergebnisse einer Expertin[3] unter Zuhilfenahme mathematischer Methoden, die bei verwandten Problemstellungen Anwendung finden, hinsichtlich ihrer Planungsqualität verbessert oder übertroffen werden können. Die Planungsqualität beschreibt hierbei sowohl die Erfüllung betrieblicher Ziele als auch die Berücksichtigung der Bedürfnisse und Präferenzen der unmittelbar betroffenen Personen, den Turnusärzt*innen.

[3] siehe Fußnote 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Um ein allgemeines Optimierungsproblem lösen zu können, bedarf es einer Metrik für die Optimalität, also die Güte, einer Lösung. Dementsprechend muss zuallererst eine geeignete Funktion gefunden werden, die alle, für das vorliegende Problem relevanten Kriterien in mathematisch wohldefinierter Gestalt und in einem angemessenen Verhältnis zueinander berücksichtigt und abbildet. Basierend auf einer solchen Funktion kann in den darauffolgenden Optimierungsschritten, unabhängig vom tatsächlich verwendeten Verfahren, eine Aussage über die Güte einer Lösung getroffen werden.

Die Ziele der Arbeit sind dementsprechend:

Ziele (Objectives – **O**):

O 1 Die Entwicklung einer Planungsmethode basierend auf einem geeigneten mathematischen Optimierungsverfahren (Metaheuristik) für die Optimierung des RSP.

O 2 Die Identifikation relevanter Zielgrößen für die Einsatzplanung von Turnusärzt*innen (sowohl aus Unternehmenssicht als auch aus Ärzt*innensicht).

O 3 Die mathematische Formulierung des RSP aus Sicht eines Krankenanstaltenverbundes in Österreich inklusive der Formulierung der zu optimierenden Zielfunktion.

Aus den dargelegten Zielen der Arbeit ergibt sich die folgende übergeordnete Forschungsfrage und die zur Beantwortung notwendigen untergeordneten Fragen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Fragen (Questions – **Q**):

Q 1 Wie können Ausbildungspläne für das vorliegende Zuteilungsproblem im Ärzt*innenausbildungsmanagement (ÄAM) mithilfe einer mathematischen Methode (exakt oder Näherungsverfahren) erzeugt und optimiert werden, sodass die Qualität von manuell erstellten Ausbildungsplänen angenähert oder übertroffen werden kann?

- **Q 1.1** Welche Zielgrößen muss die, dem Optimierungsproblem zugrundeliegende, Zielfunktion berücksichtigen, und welche Gestalt muss diese Zielfunktion aufweisen, um eine umfassende Bewertung eines Lösungskandidaten zu ermöglichen?
- **Q 1.2** Welche mathematischen Verfahren sind als Basis zur Entwicklung einer Optimierungsmethode für das vorliegende Problem geeignet?
- **Q 1.3** Wie kann diese Methode gestaltet werden, um das vorliegende Problem praxisrelevant zu lösen? (Laufzeit, Berücksichtigung der zuvor identifizierten Zielgrößen)
- **Q 1.4** Wie stark weichen die Ergebnisse des entwickelten Verfahrens in ihrer Güte von manuell, durch Planungspersonal, erzeugte Pläne ab und können diese Pläne verbessert werden?

Der Zusammenhang zwischen den identifizierten Problemstellungen, den daraus abgeleiteten Zielen und den somit zu beantwortenden Forschungsfragen ist in Abbildung 1.3 veranschaulicht.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

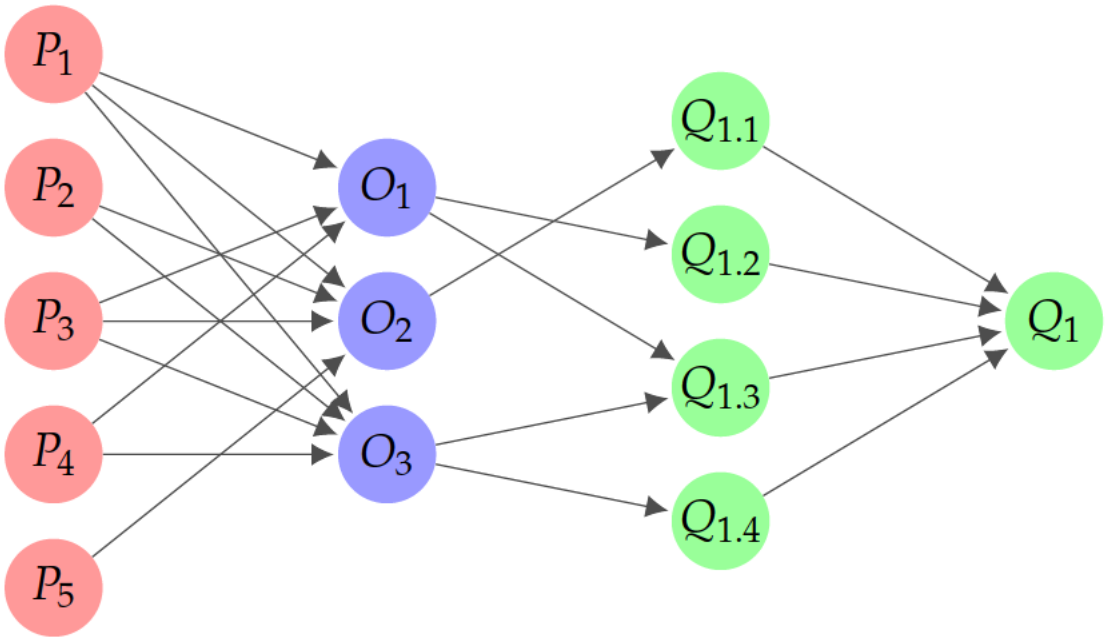


Abbildung 1.3: Zusammenhang: Probleme, Ziele und Forschungsfragen

1.4 Lösungsansatz und erwartete Ergebnisse

Im Zuge dieser Arbeit sollen die zuvor dargelegten Probleme behandelt und die identifizierten Ziele erreicht werden, um in letzter Konsequenz die daran geknüpften Forschungsfragen beantworten zu können. Die übergeordnete Forschungsfrage **Q 1**, baut auf den vorgelagerten Unterfragen **Q 1.1** - **Q 1.4** auf und kann somit genau dann beantwortet werden, wenn die vier vorgelagerten Unterfragen beantwortet sind.

Im ersten Schritt dieser Arbeit werden in Zusammenarbeit mit einer Expertin aus dem ÄAM relevante Zielgrößen, die maßgeblich für das vorliegende Planungsproblem sind, identifiziert. Die Erkenntnisse daraus werden in Abschnitt 2.3 beschrieben. Damit wird **O 1** behandelt und in weiterer Folge der Grundstein zur Beantwortung von **Q 1.1** gelegt.

Anschließend werden in Kapitel 3 die Grundlagen der infrage kommenden Optimierungsverfahren erläutert und einige Charakteristika des vorliegenden Optimierungsproblems diskutiert. In Kapitel 4 werden Lösungsansätze aus der Wissenschaft für das vorliegende und vergleichbare Probleme angeführt und diskutiert. Die Erkenntnisse aus diesen Kapiteln bilden die Entscheidungsgrundlage für die Auswahl eines geeigneten Optimierungsverfahrens.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die mathematische Formulierung des Optimierungsproblems und die Einführung einer einheitlichen Notation (**O 3**) erfolgt zu Beginn von Kapitel 5, und ist die Basis für die Gestaltung der tatsächlich implementierten Methode. Somit kann die Auswahl des zu entwickelnden Optimierungsverfahrens **O 1** basierend auf zuvor gewonnenen Informationen getroffen und in weiterer Folge **Q 1.2** beantwortet werden.

Aufbauend auf der mathematischen Formulierung des Problems (**O 3**) erfolgt im weiteren Verlauf von Kapitel 5 die Implementierung des gewählten Verfahrens (**O 1**). Am Ende diese Implementierungsschrittes kann **Q 1.3** beantwortet werden.

Abschließend werden in Kapitel 6 Lösungen des entwickelten Verfahrens mit einem manuell erzeugten Plan verglichen und deren Unterschiede aufgezeigt. Mit diesen Erkenntnissen kann **Q 1.4** beantwortet werden. Ein regelbasiertes Planungsverfahren befindet sich zum Zeitpunkt der Fertigstellung dieser Arbeit in Entwicklung. Dessen Funktionsweise ist in Anhang A.2 genauer erläutert. Dieses Verfahren ist zum Zeitpunkt der Fertigstellung dieser Arbeit jedoch noch nicht einsatzbereit und kann somit nicht zum Vergleich mit der, in dieser Arbeit entwickelten, Methode herangezogen werden.

Die Beantwortung der übergeordneten Forschungsfrage **Q 1** kann schließlich basierend auf den Antworten für **Q 1.1** - **Q 1.4** in Kapitel 7 erfolgen. In diesem Zuge werden Einschränkungen der entwickelten Methode, sowie Potenziale für künftige Verbesserungen aufgezeigt.

Diese Arbeit liefert nach ihrer Vollendung folgende Artefakte:

A 1 Ein System aus zu berücksichtigenden Zielgrößen für die Einsatzplanung im ÄAM.

A 2 Eine Zielfunktion bestehend aus einer geeigneten Kombination der einzelnen Zielgrößen.

A 3 Eine mathematische Formulierung des vorliegenden Optimierungsproblems gemäß geltender Bestimmungen in Österreich.

A 4 Ein Planungs- und Optimierungsverfahren (Software) für Planungsprobleme der Einsatzplanung für Turnusärzt*innen entsprechend der ÄAO 2015.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A 5 Eine Software zur Bewertung und Visualisierung bestehender Lösungen.

Nachfolgend wird die Gesamtheit dieser einzelnen Komponenten (**A 1**-**A 5**) als „das Artefakt“ für das vorliegende Problem bezeichnet.

1.5 Forschungsdesign

Die verwendete Forschungsmethode in dieser Arbeit orientiert sich an der von Peffers et al. [14] beschriebenen Design Science Research Methodology (DSRM), der gestaltungsorientierten Forschungsmethode.

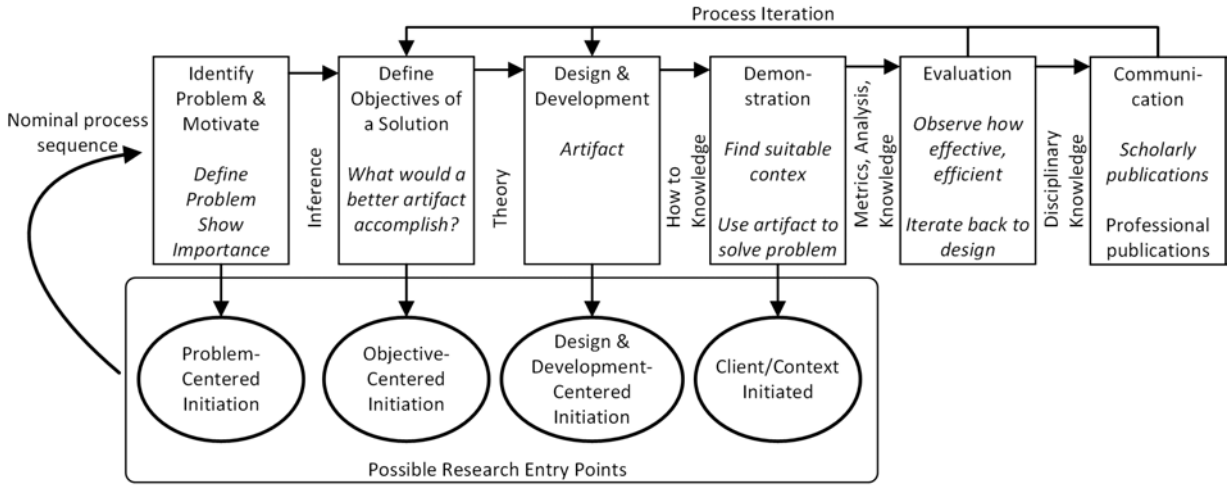


Abbildung 1.4: Das DSRM Prozessmodell (Grafik repliziert aus [14])

Die sechs Schritte der DSRM

1. Problemidentifikation und Motivation,
2. Zieldefinition,
3. Gestaltung und Entwicklung,
4. Demonstration,
5. Bewertung und
6. Kommunikation

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

beschreiben den gesamten Ablauf eines anwendungsorientierten Forschungsprozesses und sind in Abbildung 1.4 dargestellt.

Dieses Kapitel umfasst die Behandlung der Schritte 1 und 2 angeführten Ziele und Forschungsfrage verkörpern dabei die Zieldefinition.
In der Umsetzungsphase (Schritt 3) der vorliegenden Arbeit werden die zu optimierenden Zielgrößen basierend auf der ÄAO 2015 und einer Expertinnenbefragung identifiziert. Aufbauend auf diesen Ergebnissen werden geeignete Maßzahlen zur Quantifizierung der Zielgrößen festgelegt. Diese Maßzahlen werden anschließend in geeigneten Verhältnissen und Abhängigkeiten zu einer Zielfunktion formuliert. Idealerweise ermöglicht die so gestaltete Zielfunktion eine einfache Parametrisierung und somit eine einfache Gewichtung der einzelnen zu optimierenden Zielgrößen. Nach Entwicklung der Zielfunktion wird das zugrundeliegende mathematische Problem als (gemischt) ganzzahliges Optimierungsproblem formuliert. Aus den zuvor erläuterten Schritten und dem wohldefinierten mathematischen Problem, ergeben sich Anforderungen an das Lösungsverfahren. Diese Anforderungen dienen als Entscheidungsgrundlage für die Auswahl des eingesetzten Optimierungsverfahrens. Die tatsächliche Implementierung des problemspezifischen Optimierungsverfahrens ist der letzte und umfangreichste Schritt der Implementierungsphase und umfasst Tätigkeiten, wie die Formulierung geeigneter Lösungsrepräsentationen und die Entwicklung des Optimierungsalgorithmus, sowie die Umsetzung dieses Algorithmus in Form eines ausführbaren Programms. Dieser Ablauf der Implementierungsphase ist in Abbildung 1.5 dargestellt.

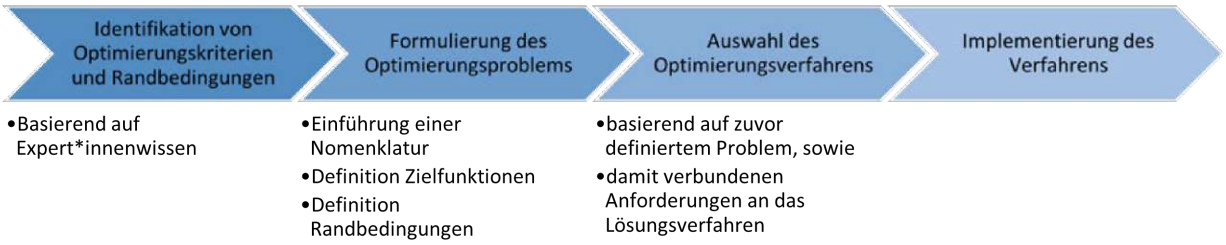


Abbildung 1.5: Implementierungsphase der Arbeit

Die Demonstration (Schritt 4) erfolgt, wie in [14] beschrieben, in Form der Anwendung des Artefakts auf eine konkrete Probleminstanz, also auf ein konkretes Planungspro-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

blem.

Bewertet (Schritt 5) wird die Güte des Ergebnisses einerseits mithilfe der in Schritt 3 entwickelten Zielfunktion und andererseits durch Fachpersonal aus der Rotationsplanung. Die Kommunikation (Schritt 6) erfolgt mithilfe dieser Arbeit und möglicher nachgelagerter Publikationstätigkeit.

1.6 Aufbau und Struktur der Arbeit

Eine Übersicht über den gesamten Aufbau ist in Abbildung 1.6 dargestellt. Dabei stellen die durchgezogenen Verbindungslinien den chronologischen Aufbau der Arbeit und die strichlierten Linien logische Einflüsse einzelner Abschnitte aufeinander dar. Im folgenden Kapitel (2) wird das vorliegende Planungsproblem im Kontext der österreichischen Ärzt*innenausbildung und seine Rahmenbedingungen, sowie Herausforderungen und Anforderungen identifiziert und definiert.

In Kapitel 3 werden alle notwendigen theoretischen Grundlagen zur Behandlung des zuvor charakterisierten Problems erläutert und zu berücksichtigende Aspekte aufgezeigt, die für die Auswahl eines geeigneten Verfahrens ausschlaggebend sind.

Der aktuelle Stand von Wissenschaft und Technik in Bezug auf das zu behandelnde Problem und ähnliche Probleme wird in Kapitel 4 dargelegt. Außerdem wird das, in dieser Arbeit verwendete, Optimierungsverfahren in diesem Kapitel beschrieben.

Das entwickelte Verfahren wird in Kapitel 5 beschrieben. Die damit erzielten Resultate werden in Kapitel 6 ausgewertet und mit menschlichen Planungsergebnissen verglichen.

In Kapitel 7 werden die Resultate im Kontext der zuvor beschriebenen Problemstellung und Forschungsfragen, sowie des Forschungsdesigns betrachtet, bewertet und diskutiert. Außerdem werden dort die Einschränkungen der Ergebnisse aufgezeigt. Am Ende dieses Kapitels werden Potenziale für zukünftige Weiterentwicklungen des Verfahrens erörtert.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

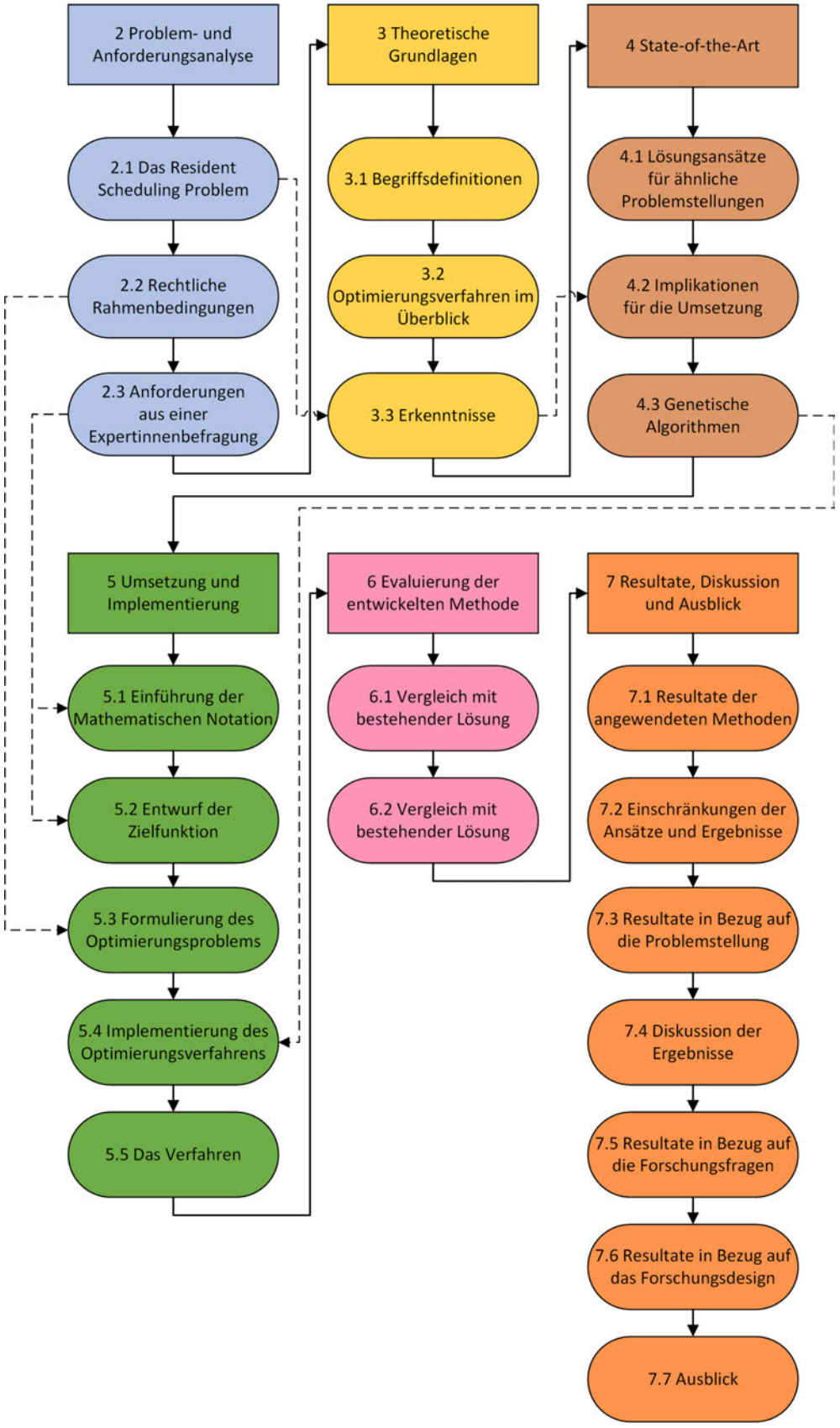


Abbildung 1.6: Übersicht – Struktur der Arbeit

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Kapitel 2

Problem- und Anforderungsanalyse

Das dieser Arbeit zugrundeliegende Planungsproblem im ÄAM wird in der einschlägigen Literatur als Resident Scheduling Problem (RSP) bezeichnet und wird in Abschnitt 2.1 erläutert.

Die postgraduale Ausbildung von Ärzt*innen ist in Österreich in der ÄAO 2015 geregelt [6]. Diese Verordnung legt gemeinsam mit dem ÄrzteG 1998 [7] alle gesetzlichen Rahmenbedingung hinsichtlich qualitativer sowie quantitativer Aspekte der Ausbildung von Ärzt*innen fest. In Abschnitt 2.2 werden die, für diese Arbeit, wesentlichen Aussagen dieser Rechtsnormen dargelegt.

In Abschnitt 2.3 werden alle im Rahmen einer Expert*innenbefragung identifizierten Herausforderungen und Ziele, die mit der Behandlung des vorliegenden RSP verbunden sind, besprochen.

Aktuell ist ein regelbasiertes Planungsverfahren, das basierend auf Expertinnenwissen plant, in Entwicklung. Dieses Verfahren wird in Anhang A.2 kurz beschrieben. Da es jedoch noch keine verwendbaren Lösungen liefert, kann es in dieser Arbeit nicht für einen Vergleich herangezogen werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

2.1 Das Resident Scheduling Problem

Wie eingangs erwähnt, handelt es sich beim RSP um ein Planungsproblem aus dem ÄAM. Konkret geht es dabei um die optimale Zuteilung von Ärzt*innen in Ausbildung zu den jeweiligen Ausbildungsplätzen, die diese benötigen, um ihre Ausbildung abzuschließen. Das damit verbundene Optimierungsproblem gehört zur Klasse der $\mathcal{NP}$-schweren (nichtdeterministisch-polynomiell) Probleme [1], [15], [16]. Die, mit dieser Problemklasse einhergehenden, Herausforderungen werden in Abschnitt 3.1.4 beschrieben. Was in diesem Kontext optimal bedeutet, wird in Abschnitt 2.3 diskutiert.
Die zugrundeliegende Logik ist bei der Verplanung in der allgemeinärztlichen und der fachärztlichen Ausbildung dieselbe, allerdings wird in der ÄAO 2015 keine einheitliche Benennung der kleinsten abzuschließenden Ausbildungsbestandteile verwendet. In dieser Arbeit wird auf die Unterscheidung in der Nomenklatur, im Sinne der Einfachheit, bewusst verzichtet. Die Begriffe „Fachgebiet“ (allgemeinärztliche Ausbildung) und „Modul“ (fachärztliche Sonderfach-Schwerpunktausbildung) beschreiben diese kleinsten Bestandteile der jeweiligen Ausbildung in der ÄAO 2015 und sind in dieser Arbeit als synonym zu verstehen. Ein Fachgebiet bzw. Modul ist in dieser Arbeit damit als ein einzelner Ausbildungsgegenstand, der im Zuge einer beliebigen ärztlichen Ausbildung absolviert werden muss, zu verstehen. Wenn im weiteren Verlauf von Personen gesprochen wird, sind i.A. Turnusärzt*innen gemeint, sofern nicht explizit anders angegeben.

Um die allgemeinärztliche oder die fachärztliche Ausbildung in Österreich abschließen zu können, muss nach vollendetem Medizinstudium eine praktische Ausbildung entsprechend der gewählten Fachgebiete absolviert werden.
Basierend auf der ÄAO 2015 müssen Turnus*ärztinnen im Rahmen ihrer Tätigkeit einen vordefinierten Umfang an Fächern in vorgegebenem Ausmaß und teilweise vorgegebener Reihenfolge absolvieren, wobei die Ausbildungen in mehrere Abschnitte unterteilt sind (siehe Abbildung 1.1). Die allgemeinärztliche Ausbildung gliedert sich beispielsweise in drei Abschnitte (siehe Abbildungen 1.1 und 2.1), die Basisausbildung, den Spitalsturnus und die Ausbildung in einer Lehrpraxis. Die Ausbildungsabschnitte müssen dabei in der jeweils vorgegebenen Reihenfolge abgeschlossen werden. Um einen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Ausbildungsabschnitt absolvieren zu können, müssen Turnusärzt*innen verschiedenste, darin enthaltene Fächer belegen und die dafür vorgeschriebenen Erfahrungen, Fertigkeiten und Kenntnisse erwerben. Um diesen Erwerb von Kompetenzen zu bewerkstelligen, müssen Turnusärzt*innen i.d.R mehrere Stationen, also Ausbildungsstellen und Abteilungen, in einem oder mehreren Ausbildungskrankenhäusern (Ausbildungsstätten) durchlaufen.

Ausschlaggebend für den Abschluss eines Ausbildungsmoduls gemäß ÄAO 2015 ist dementsprechend sowohl die Tätigkeit in einer definierten Position für eine vorgeschriebene Mindestdauer, als auch der Erwerb der erwähnten Erfahrungen, Fertigkeiten und Kenntnisse (siehe [6]). Daraus folgt, dass bei der Erstellung von Ausbildungsplänen sowohl eine Zeit-, als auch eine Kompetenz-Komponente berücksichtigt werden muss, um eine ganzheitliche Planung zu bewerkstelligen.

Die Zuteilung von Turnus*ärztinnen zu ihren Ausbildungsstellen erfolgt im Rahmen der sogenannten „Rotationsplanung“, der Planung einer zeitlichen und örtlichen Rotation von Personen durch verschiedene Posten und damit meist durch Abteilungen bzw. sogar Krankenhäuser.

Die grundlegende Annahme, dass die erforderlichen Kompetenzen eines Ausbildungsmoduls üblicherweise in der vorgegebenen Zeit erworben werden können, ermöglicht die Abstraktion des Planungsproblems von einer zeit- und kompetenzorientierten Betrachtung hin zu einer rein zeitbasierten Problembetrachtung. Bei dieser rein zeitbasierten Behandlung sind die jeweiligen Qualifikationen und Kompetenzen der zu verplanenden Individuen zwar trotzdem in gleichem Maße relevant, die Abstraktion lässt allerdings die Vernachlässigung der Unsicherheit des Kompetenzerwerbs im geplanten Zeitraum zu. Einfach gesagt wird die Äquivalenz von Ausbildungszeit und Kompetenzerwerb als Modellannahme getroffen, um die Problem- und damit die Planungskomplexität zu reduzieren.

Basierend auf den vorgegebenen (Mindest-) Dauern für die einzelnen Fächer eines Abschnitts kann die Verplanung von Personen auf Ausbildungsfächer und damit auf einzelne Organisationseinheiten und in weiterer Folge auf Dienstposten sowie Ausbildungsstellen erfolgen.

Wie bereits erläutert, müssen alle Fächer eines Ausbildungsabschnittes absolviert wer-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

den, um diesen Abschnitt erfolgreich abzuschließen und in weiterer Folge Fächer des darauffolgenden Abschnittes belegen zu können. Innerhalb eines Abschnittes gibt es keine rechtlichen Einschränkungen, was die Belegung der einzelnen Fächer anbelangt. Ferner können Fächer auch unterbrochen und stückweise belegt werden. Beispielsweise könnte der Spitalsturnus (siehe Abbildung 2.1), mit dem Fach „Kinder- und Jugendheilkunde“ begonnen werden, nach zwei Monaten ein anderes Fach belegt und das verbleibende dritte Monat später erledigt werden. Maßgeblich ist lediglich, dass alle geforderten Fächer im vorgegebenen Ausmaß ungeachtet ihrer Reihenfolge belegt und abgeschlossen werden.

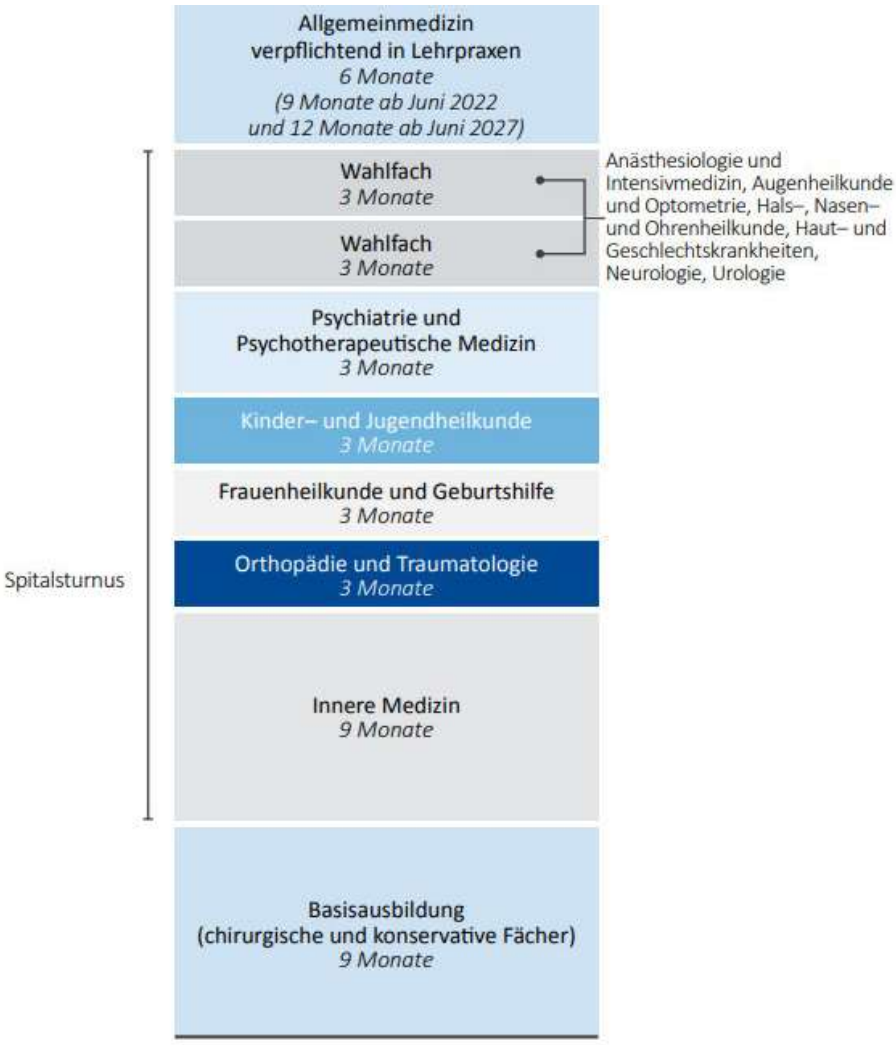


Abbildung 2.1: Fächer und Dauern in der Ausbildung für Allgemeinmedizin [10]

Gesondert hervorzuheben sind hierbei Wahlfächer. Ein Ausbildungsabschnitt, wie der

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Spitalsturnus in allgemeinärztlichen Ausbildung (siehe Abbildung 2.1), kann über ein oder mehrere Wahlfächer verfügen, die belegt werden müssen. Deren Inhalte können dabei allerdings frei gewählt werden. Im Fall des Spitalsturnus müssen zwei Fächer aus insgesamt sechs wählbaren Fächern mit einer Dauer von jeweils drei Monaten absolviert werden. Hierbei ist zu beachten, dass Personen i.d.R Präferenzen hinsichtlich der zu absolvierenden Wahlfächer haben und diese im Idealfall in der Planung berücksichtigt werden sollten.
Die Implikationen, die sich damit für das Planungsproblem ergeben, werden am Ende dieses Kapitels diskutiert und fließen in die Problemformulierung in Kapitel 5 und damit in die Gestaltung des Optimierungsverfahrens ein.

2.2 Rechtliche Rahmenbedingungen

Ausbildungsstätten und die Anzahl und der dort verfügbaren Ausbildungsstellen werden durch die ÖÄK gemäß § 9 und § 10 des ÄrzteG 1998 [7] anerkannt. Die so anerkannten Ausbildungsstellen befähigen eine Ausbildungsstätte zur Ausbildung von Ärzt*innen in den, den Ausbildungsstellen zugeordneten Fachgebieten. Eine Ausbildungsstelle kann dementsprechend verwendet werden, um einzelne Bestandteile der ärztlichen Ausbildung zu absolvieren. Üblicherweise sind diese Ausbildungsstellen und die dort genehmigten Fachgebiete einzelnen Organisationseinheiten einer Krankenanstalt (Abteilungen bzw. Stationen) zugeordnet. Damit ist sowohl die Qualität, als auch die Quantität der verfügbaren Ausbildungsstellen in einer Ausbildungsstätte festgesetzt. Die Anerkennung einer Ausbildungsstätte erfolgt für eine Dauer von sieben Jahren und muss nach Verstreichen dieser sieben Jahre gemäß dem Rezertifizierungsverfahren nach § 13a ÄrzteG 1998 [7] erneuert werden. Daraus folgt das „Ablaufen“ von Ausbildungsstellen und die damit verbundene zeitlichen Limitierung der Ressourcenverfügbarkeit dieser Ausbildungsstellen, die wiederum in der Planung berücksichtigt werden muss.
Die Kernausbildungszeit bei Vollzeitanstellung beträgt 35 Wochenstunden. Diese 35 Wochenstunden müssen von Ärzt*innen in Ausbildung absolviert werden, um das jewei-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

lige Ausbildungsziel des Ausbildungsabschnittes erreichen zu können, die tatsächliche Arbeitszeit von Turnusärzt*innen ist üblicherweise höher. Bei Teilzeitbeschäftigung verhält sich die absolvierte Ausbildungszeit proportional zum Beschäftigungsausmaß, weshalb sich die notwendige Ausbildungsdauer entsprechend verlängert. Ist eine Person beispielsweise lediglich für 50 % des eigentlichen Beschäftigungsausmaßes angestellt, so verdoppelt sich damit die Dauer ihrer Ausbildung, da für den Abschluss eines Moduls eine vordefinierte Ausbildungsdauer notwendig ist, die sich an der Vollzeitbeschäftigung orientiert. Der bereits angesprochene Kompetenzerwerb (Erfahrungen, Fertigkeiten und Kenntnisse) ist in der ÄAO 2015 geregelt. Aus rein rechtlicher Betrachtung müsste dem ÄrzteG 1998 und der ÄAO 2015 zufolge lediglich eine monatliche Zuordnung von Personen zu Ausbildungsstellen erfolgen, um das vorliegende Planungsproblem zu behandeln und zu lösen. Im folgenden Abschnitt wird sich zeigen, dass dem nicht so ist und aus Sicht der Ausbildungsstätte eine weitere Komponente die Komplexität des Zuteilungsproblems maßgeblich beeinflusst.

2.3 Anforderungen aus einer Expertinnenbefragung

Die in Abschnitt 1.3 angeführten Probleme und Herausforderungen, die mit der Einsatzplanung von Turnusärzt*innen verbunden sind, wurden im Zuge einer leitfadengestützten Befragung einer Expertin [4] identifiziert. Die dazu verwendeten Fragen sind in Anhang A.1 angeführt.

Neben den, in Abschnitt 1.3 angeführten, Problemen und Herausforderungen in Verbindung mit der Rotationsplanung, wurde im Rahmen der Befragung besagter Expertin die relevanten, zu berücksichtigenden Aspekte bei der Erstellung eines Ausbildungsplans identifiziert. Die nachfolgenden Behauptungen und Aussagen beruhen auf ihren Erfahrungen.

Die Planung der Ausbildung von Turnusärzt*innen erfolgt, wie im vorigen Abschnitt beschrieben, indem diese Ärzt*innen einer Ausbildungsstelle zugeordnet werden. Hierbei ist gesondert hervorzuheben, dass eine Ausbildungsstelle in ihrem ursprünglichen

[4] siehe Fußzeile 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Sinn eine Ausbildungsstätte lediglich dazu befähigt, Ärzt*innen auszubilden. Diese Stelle hat jedoch nichts mit dem tatsächlichen Anstellungsverhältnis oder der Kapazitätsverteilung innerhalb einer Ausbildungsstätte zu tun. Die „dienstrechtliche“ und kapazitäre Zuteilung von Turnusärzt*innen erfolgt über ihre **Dienstposten**, die sie **gleichzeitig** mit einer Ausbildungsstelle innehaben müssen. Einfach gesagt ermöglicht ein Dienstposten, dass Ärzt*innen in einer bestimmten Abteilung arbeiten, die Ausbildungsstelle ist lediglich eine Bescheinigung dafür, dass eine ihr zugeteilte Person im jeweiligen Fach ausgebildet werden kann. Somit ist zwar keine Ausbildungsstelle erforderlich, um der Tätigkeit in einer Abteilung nachzugehen, um Ausbildungsfortschritt zu erzielen, ist sie jedoch zwingend notwendig. Das Problem, das sich dahinter verbirgt, liegt in der Beziehung zwischen Dienstposten und Ausbildungsstellen. Wünschenswert wäre eine 1:1 Beziehung, da man diese so als eine Kapazitätseinheit in der Planung betrachten könnte. In der Realität besteht jedoch eine **m:n** Beziehung zwischen ihnen. Das bedeutet, dass m Dienstposten mit n Ausbildungsstellen kombiniert werden können, wobei i.d.R $m \neq n$ gilt.

Im Zuge der Befragung wurden die Interessen der wesentlichen Stakeholder*innen in der Ausbildungsplanung erfasst. Aus Sicht der **Turnusärzt*innen** ergeben sich im Sinne ihrer Ausbildung die Forderungen nach

- einer möglichst unterbrechungsfreien Ausbildung,
- einem zügigen Abschluss der Ausbildung,
- der Berücksichtigung ihrer Präferenzen bezüglich der Wahlfächer und
- Vielfalt in der Ausbildung bei gleichzeitig wenig unnötigen Ortswechseln.

Aus Sicht der **Krankenhausverwaltung** sind

- kurze Durchlaufzeiten von Ärzt*innen durch deren Ausbildung,
- eine gleichmäßige Kapazitätsverfügbarkeit in allen zu besetzenden Abteilungen,
- eine qualitativ hochwertige Patient*innenversorgung,
- langfristige Planbarkeit,

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

- möglichst wenig Ausbildungszeiten in externen Häusern,
- sowie möglichst wenig Aufwand und geringe Fehlerquoten in der Planung

erstrebenswert.

Aus Sicht der **Patient*innen** steht die Versorgungsqualität an oberster Stelle.

Die Ziele der Ärzt*innen, Patient*innen sowie der Verwaltung überschneiden sich in einem gewissen Ausmaß und stehen teilweise auch im Widerspruch zueinander.

Erschwerend hinzu kommen die Interessen einer vierten beteiligten Personengruppe, den **Einsatzplaner*innen**. Da sich das vorliegende Planungsproblem als komplex darstellt, greift das Planungspersonal häufig auf Entscheidungsregeln, die auf persönlichen Erfahrungen beruhen, zurück. Solche Entscheidungs- oder Prioritätsregeln können etwa die Reihung von Personen in der Verplanung oder auch die Reihung von Ausbildungsstellen umfassen. Obwohl sich diese Vereinfachungen bei der Erstellung von Rotationsplänen als hilfreich erweisen, können die angewandten Priorisierungen und Annahmen im Widerspruch zu den zuvor angeführten Zielen stehen und somit die Güte der erzeugten Pläne maßgeblich negativ beeinflussen.

Das zu entwickelnde Planungswerkzeug soll neben einer automatischen Planung in der Lage sein, existierende Pläne auf deren Zulässigkeit und Güte zu überprüfen und diese anschließend zu optimieren. So kann einerseits dem Wunsch nach geringeren personellen und finanziellen Aufwänden in der Planung Genüge getan und gleichzeitig dem aktuellen Fachkräftemangel, sowie der Abwanderung von Wissen, aufgrund von Personalfluktuation in der planenden Belegschaft, entgegengewirkt werden.

Aus den angeführten Zielen der einzelnen Interessengruppen wurden in Zusammenarbeit mit der Expertin folgende, in Tabelle 5.8 angeführte, Indikatoren identifiziert. Diese Indikatoren sollen in weiterer Folge verwendet werden, um die Güte eines Rotationsplans messbar zu machen und werden nachfolgend erläutert.

Metriken #1 & #2 - Stehmonate

Die Monate, in denen eine Person arbeitet, aber keinen Ausbildungsfortschritt erzielt (in weiterer Folge als Stehmonate bezeichnet), sind weder aus Sicht der Turnusärzt*innen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

#	Name der Metrik	Beschreibung	Optimal, wenn ...
1	Stehmonate	Monate ohne Ausbildung	min
2	Konsekutive Stehmonate	Monate ohne Ausbildung direkt hintereinander	min
3	Ortswechsel	Anzahl der Ortswechsel, Abteilungsebene und Krankenhausebene	min
4	Einmonatige Zuweisungen	Einmonatige Zuweisungen auf Abteilungsebene	min
5	Monate extern	Ausbildungsmonate in externer Ausbildungsstätte	min
6	Präferenzen	Berücksichtigung präferierter Fächer	max

Tabelle 2.2: Übersicht der relevanten Zielgrößen

noch aus Sicht der Krankenhausleitung wünschenswert, da die Ausbildung damit unnötig verlängert wird. In weiterer Konsequenz führt dies dazu, dass die zukünftige Ausbildungskapazität des Krankenhauses verkleinert wird und Turnusärzt*innen länger vom Arbeitsmarkt für ausgebildete Ärzt*innen ferngehalten werden. Gleichzeitig belegen Turnusärzt*innen in einem Stehmonat einen Dienstposten, auf dem eine andere Person möglicherweise ausgebildet werden könnte und verursachen so Ressourcenverschwendung aus Sicht der Ausbildungsstätte. Diese Aussage beruht auf der Annahme, dass im betrachteten Monat, in dem ein Stehmonat geplant ist, an sich ausreichend Ausbildungsplätze vorhanden wären, das Potenzial jedoch aufgrund einer nicht optimalen Planung ungenutzt bleibt.

Die unmittelbare Aneinanderreihung mehrerer Stehmonate für eine einzelne Person ist als noch schlechter zu beurteilen, als die gleiche Anzahl an Stehmonaten auf mehrere Personen einzeln verteilt. Dies trifft insbesondere aus einem Fairness-Aspekt zu, da sich die Ausbildungen von einzelnen Personen zugunsten anderer immens verlängern könnten, wenn die Stehmonate so verkettet werden, obwohl die Gesamtzahl der Stehmonate

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

gleich bleibt. Davon abgesehen wird, sofern die Person keinen Ortswechsel vollzieht, immer dieselbe Ressource (Dienstposten) und damit immer die Möglichkeit zur Ausbildung in dieser Abteilung für andere Personen blockiert.
Die schlichte Anzahl an Stehmonaten ist davon abgesehen ein indirektes Maß für die Gesamtdauer der Ausbildungen, da jeder Monat, der kein Stehmonat ist, im Umkehrschluss ein Monat mit Ausbildung ist und somit zum Ausbildungsfortschritt beiträgt. Somit ist die Gesamtausbildungsdauer genau dann minimal, wenn die Anzahl der Stehmonate minimal ist.

Metrik #3 - Ortswechsel

Mit Ortswechseln von Turnusärzt*innen geht zwangsläufig eine Übergabe ihrer Patient*innen an andere Ärzt*innen einher. Eine Studie aus dem Jahr 2015 zeigt, dass die Mortalitätsrate mit dem Personalwechsel von Turnusärzt*innen in Krankenhäusern korreliert [17]. Die Thematik der durchgängigen Patient*innenversorgung wird in der wissenschaftlichen Literatur als „Continuity of Care“ (Versorgungskontinuität) bezeichnet. Solche Ortswechsel verursachen Veränderungen in den jeweiligen Teams auf den betroffenen Stationen, was deren Leistungsfähigkeit maßgeblich beeinträchtigt, da Turnusärzt*innen sich in der neuen Umgebung eingewöhnen müssen und das bestehende Personal sich auf die neuen Gegebenheiten einstellen muss [1]. Ein stetiger Personalwechsel kann somit sowohl die Versorgungsleistung als auch die Versorgungsqualität empfindlich stören. Häufige Ortswechsel stellen für Turnusärzt*innen neben der genannten Eingewöhnungsphase auch eine Herausforderung hinsichtlich der Anreise zu verschiedenen Dienstorten dar. Der organisatorische Aufwand für Verwaltung und Personal, der mit Ortswechseln einhergeht, ist ebenfalls nicht außer Acht zu lassen. Somit sind Ortswechsel aus den genannten Gründen sowohl innerhalb eines Krankenhauses, als auch krankenhausübergreifend möglichst gering zu halten. Dabei sind die Auswirkungen eines Wechsels zwischen zwei Krankenhäusern als gravierender anzusehen als der Wechsel zwischen zwei Abteilungen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Metrik #4 - Einmonatige Zuweisungen

Diese Metrik überschneidet sich mit Metrik #3, da hierfür Monate herangezogen werden, in denen Turnusärzt*innen in eine Abteilung kommen und sie am Ende desselben Monats wieder verlassen. Der Grund für die separate Berücksichtigung dieser Metrik liegt in den besonders negativen Auswirkungen für alle beteiligten Stakeholder, wenn Turnusärzt*innen für so kurze Zeit in einer Abteilung tätig sind. Die Patient*innenversorgung, das Team vor Ort und die Ärzt*innen selbst leiden darunter und der damit verbundene Verwaltungsaufwand ist überproportional groß.

Metrik #5 - Monate Extern

Aus Sicht eines Krankenanstaltenverbundes sind Ausbildungszeiten, die in einer externen Ausbildungsstätte absolviert werden, sowohl mit organisatorischem Aufwand, als auch mit Kosten verbunden. Deswegen gilt es die Monate, die dort belegt werden, ebenfalls zu minimieren.

Metrik #6 - Präferenzen

Die Präferenzen von Turnusärzt*innen hinsichtlich der Wahlfächer, die in ihren jeweiligen Ausbildungen verfügbar sind, stellen ein weiteres Optimierungskriterium dar. Wenn Personen bestimmte Fächer präferieren, sollte die Belegung dieser, sofern umsetzbar, ermöglicht werden, um ihnen den gewünschten Ausbildungsschwerpunkt zu ermöglichen. Die Zahl der berücksichtigten Präferenzen in einem Ausbildungsplan sollte demnach maximiert werden.

Hauptaugenmerk wird laut Angaben der Expertin aktuell auf die Minimierung von Krankenhauswechseln, Stehmonaten und einmonatigen Zuweisungen gelegt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Kapitel 3

Theoretische Grundlagen

3.1 Begriffsdefinitionen

Um die Diskussion relevanter Konzepte in den darauffolgenden Kapiteln zu ermöglichen, werden in den nachfolgenden Abschnitten einige dieser Konzepte und Begriffe erläutert.

3.1.1 Kriterien und Zielfunktionen

Am Anfang jedes Optimierungsvorhabens stehen eine oder mehrere, Zielsetzungen. Diese Zielsetzungen müssen im Sinne der Vergleichbarkeit als messbare **Kriterien** formuliert werden. Diese Kriterien werden in Form von Zielfunktionen, also mathematisch auswertbaren Funktionen definiert, um die Güte einer Lösung hinsichtlich einer bestimmten Zielsetzung bewerten zu können [18].

3.1.2 Mehrzieloptimierungsprobleme und deren Optimalität

Die Optimierung einer skalaren Zielfunktion $f(x)$ beliebiger Gestalt stellt sich häufig als relativ trivial dar und kann in diesem Fall mithilfe einfachster mathematischer Methoden erfolgen. Funktionen, die in Optimierungsproblemen auftreten, verfügen häufig

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

nicht nur über eines, sondern mehrere Optima. Dabei existiert jedoch nur ein Wert (der allerdings an mehreren Stellen auftreten kann) der ein globales Optimum darstellt. Ein **globales** Minimum an der Stelle x_{opt} zeichnet sich durch die Eigenschaft aus, dass es im betrachteten Bereich keine Stelle x_j gibt, für die $f(x_j) < f(x_{opt})$ gilt. Ein **lokales** Minimum hingegen ist lediglich in seiner unmittelbaren Umgebung, jedoch nicht global optimal. Dieser Sachverhalt ist in Abbildung 3.1 verdeutlicht. Zur Optimierung eines

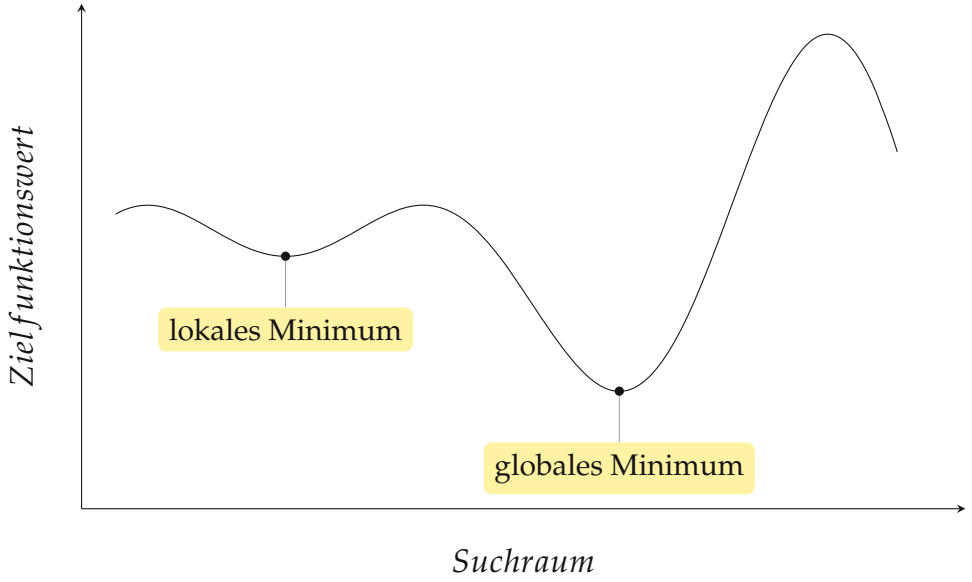


Abbildung 3.1: Optima einer einfachen Zielfunktion

Mehrzieloptimierungsproblem (*engl*: Multi-Objective Optimization Problem, MOP) ist man ebenfalls am Finden eines globalen Optimums der vorliegenden Zielfunktion interessiert. Bei einem MOP beinhaltet die zugrundeliegende Zielfunktion zumindest zwei oder mehr Zielfunktionen, die möglicherweise miteinander konkurrieren. Die Beschreibung eines MOP erfolgt mithilfe mehrerer Zielfunktionen $(f_1(x), f_2(x), ...f_n(x))$ und den zugehörigen Randbedingungen. Das Ziel ist es, für jede einzelne Zielfunktion ein Optimum unter Einhaltung der Optimierungsnebenbedingungen zu erreichen. Bei einem Minimierungsproblem soll somit für jede dieser Zielfunktionen ein Minimum erzielt werden. Häufig stehen die, durch die einzelnen Zielfunktionen $f_i(x)$ ausgedrückten, Teilziele in Konkurrenz, die Verbesserung eines Teilziels impliziert also die Verschlechterung eines anderen.

Üblicherweise sind MOPs nicht nur deren Zielfunktionen, sondern zusätzlichen Rand-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

bedingungen unterworfen, die den zulässigen Lösungsraum einschränken.
Ein allgemeines MOP lässt sich gemäß [18] wie in Gleichung 3.1 definieren.

$$
\begin{gathered}
f(\mathbf{x}) = \begin{bmatrix} f_1(\mathbf{x}), & f_2(\mathbf{x}), & \dots & f_n(\mathbf{x}) \end{bmatrix}^{\mathbf{T}} \\
\text{mit dem Vektor der Entscheidungsvariablen} \\
\mathbf{x} = \begin{bmatrix} x_1, & x_2, & \dots, & x_k \end{bmatrix}^{\mathbf{T}} \\
\text{und den Randbedingungen} \\
g_i(\mathbf{x}) \leq 0 \quad i = 1, \dots, m \\
\text{und} \\
h_j(\mathbf{x}) = 0 \quad j = 1, \dots, p
\end{gathered}
\tag{3.1}
$$

Das Problem wird durch Ungleichheitsrandbedingungen $g_i(\mathbf{x})$ und Gleichheitsrandbedingungen $h_j(\mathbf{x})$, die eingehalten werden müssen, um einen zulässigen Lösungsvektor $\mathbf{x}$ zu finden, eingeschränkt. Ein idealer Vektor $\mathbf{x}_{ideal}$ ist ein Vektor, der alle Zielfunktionen $f_i(\mathbf{x})$ optimieren (minimieren oder maximieren) würde. In der Praxis existiert ein solcher Vektor jedoch nicht oder erfüllt eine oder mehrere Randbedingungen nicht und ist somit nicht zulässig.
Bis jetzt wurde lediglich von der Optimierung der einzelnen Zielfunktionen gesprochen. In konkreten Anwendungsfällen ist man allerdings an der gleichzeitigen Optimierung aller Zielfunktionen interessiert [18]. Der Begriff „Optimum“ im Kontext eines MOP ist nicht mehr so trivial wie in Abbildung 3.1 dargestellt und muss neu definiert werden. Da die einzelnen Zielfunktionen f_i i.d.R miteinander konkurrieren, gilt es in der Mehrzieloptimierung gute Kompromisse zu finden, anstatt einzelne Zielfunktionen zu optimieren. Um zu entscheiden, ob eine Lösung für ein MOP optimal ist, wurde der Begriff der Pareto-Optimalität bzw. des Pareto-Optimums von Vilfredo Pareto eingeführt [18].

Ein Vektor **x*** ist Pareto-optimal, wenn kein anderer zulässiger Vektor existiert, der eine Zielfunktion verbessert, ohne dabei mindestens eine andere zu verschlechtern. Eine Pareto-optimale Lösung wird auch als nicht-dominiert bezeichnet. Im Umkehrschluss ist ein Vektor $\mathbf{x}_a$ genau dann von einem anderen Vektor $\mathbf{x}_b$ dominiert, wenn $\mathbf{x}_b$ hinsichtlich keiner Zielgröße schlechter als $\mathbf{x}_a$, jedoch hinsichtlich mindestens einer Zielgröße

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

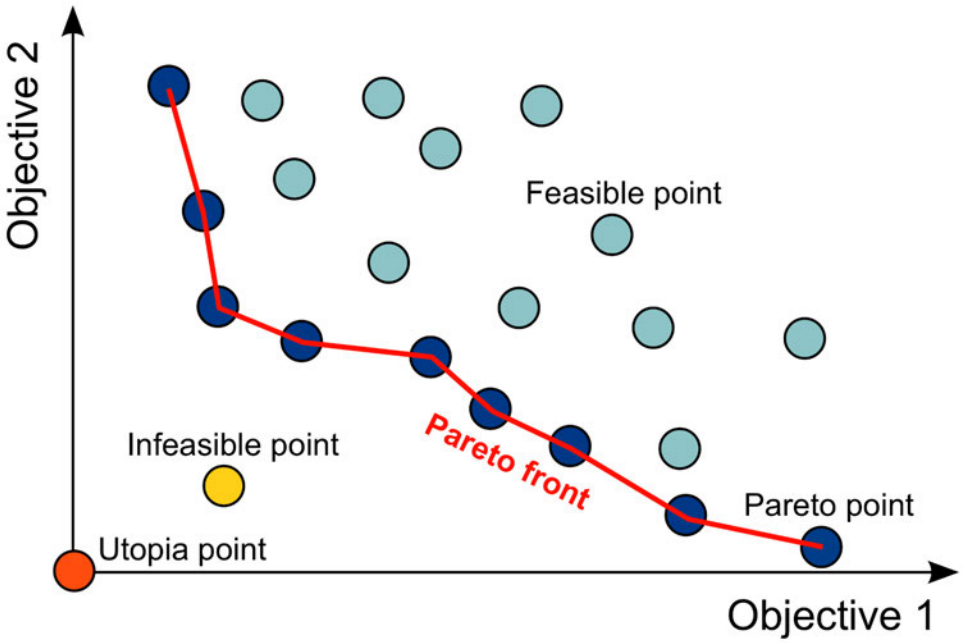


Abbildung 3.2: Pareto-Optimum und die Pareto-Front [19]

besser als $\mathbf{x}_a$ ist [18], [20]. Man schreibt auch $\mathbf{x}_b \succ \mathbf{x}_a$.

Die Menge aller Pareto-Optima wird auch als Pareto-Front bezeichnet [18]. Diese Front ist in Abbildung 3.2 für zwei Zielgrößen veranschaulicht. Alle Lösungen an der Pareto-Front (dunkelblau) zeichnen sich dadurch aus, dass die benachbarten Lösungen hinsichtlich zumindest einer der beiden Ziele schlechter sind. Für Lösungen, die nicht an der Pareto-Front liegen (hellblau), existiert zumindest eine Lösung, die diese Lösungen dominiert, also besser hinsichtlich mindestens eines Optimierungsziels und nicht schlechter hinsichtlich des anderen Ziels ist.

Somit existiert für ein gegebenes MOP nicht **eine** optimale Lösung, sondern vielmehr **eine Menge** an nicht-dominierten Lösungen, die als Pareto-optimal oder Pareto-effizient für das vorliegende Problem bezeichnet werden können. Welche dieser effizienten Lösungen schlussendlich als globales Optimum für das Problem betrachtet wird, hängt von den Präferenzen der entscheidenden Instanz ab. Zur Verdeutlichung ist in Anhang A.3 ein Beispiel angeführt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

3.1.3 Ganzzahlige- und Gemischt-Ganzzahlige Optimierungsprobleme

Zur Behandlung diskreter Probleme, wie Problemen in der Produktionsplanung, Routenplanungsproblemen wie dem Handlungsreisendenproblem (*engl*: Traveling Salesman Problem, TSP) oder dem Rucksackproblem (*engl*: Knapsack Problem, KP) müssen im Gegensatz zu klassischen Optimierungsproblemen mit stetigen Entscheidungsvariablen zusätzlich Forderungen hinsichtlich der Ganzzahligkeit dieser Variablen berücksichtigt werden. In klassischen Ressourcenzuteilungsproblemen, wie der Produktionsplanung, werden in der Regel Ressourcen finiter Größe verplant, da etwa die Produktion von 5,7 Teilen auf 3,25 Maschinen wenig Sinn ergibt. Diese finiten Größeneinheiten werden üblicherweise durch ganzzahlige Variablen ausgedrückt. Die Optimierung solcher Probleme wird als ganzzahlige Optimierung (*engl*: Integer Programming, IP) bezeichnet, wenn alle Entscheidungsvariablen ganzzahlig sind oder gemischt-ganzzahlige Optimierung (*engl*: Mixed Integer Programming, MIP) genannt, wenn zumindest ein Teil dieser Variablen ganzzahlig ist. Häufig werden IP-Optimierungsprobleme treffend als kombinatorische Optimierungsprobleme bezeichnet [21]. Viele dieser Optimierungsprobleme und insbesondere kombinatorische Optimierungsprobleme wie das TSP (siehe Abschnitt 3.1.5) können mithilfe von Graphen und Bäumen beschrieben und veranschaulicht werden. In Anhang A.4 werden die Grundlagen zu Graphen und Bäumen im Sinne der Vollständigkeit erläutert.

3.1.4 Komplexitätstheorie und Komplexitätsklassen

Verschiedene algorithmische Problemstellungen sind üblicherweise unterschiedlich leicht oder schwer zu lösen. Beispielsweise ist das Sortieren einer unsortierten Liste bestehend aus ganzen Zahlen trivial und in geringer Laufzeit ($\mathcal{O}(n \log(n))$) möglich. Anders sieht das jedoch bei komplizierteren Problemen, wie dem TSP (siehe 3.1.5) oder dem KP aus [22]. Für diese und viele andere Optimierungsprobleme ist es keineswegs trivial, in geringer Laufzeit eine optimale Lösung zu finden. Um die Schwere solcher algorithmischer Probleme einteilen und miteinander vergleichen zu können, gibt es sogenannte

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Komplexitätsklassen. Die Schwierigkeit ein Entscheidungsproblem zu lösen, ist generell von seiner Zeit- und Platzkomplexität, also seiner Ressourcenintensität, abhängig. Die **Zeitkomplexität** entspricht der Anzahl der notwendigen Iterationsschritte zur Lösung eines Problems, die **Platzkomplexität** hingegen beschreibt den für die Lösung erforderlichen Platz in einem Speicher (i.d.R Platz im Speicher eines Rechners) [21]. Beide dieser Komplexitätsarten sind üblicherweise von der Größe des zu lösenden Problems abhängig. Im Fall des Sortierens einer Liste entspräche die Größe n des Problems der Länge dieser Liste. Die Komplexität eines Problems in Platz und Zeit wird üblicherweise in der Landau- oder O-Notation angegeben, womit obere und untere Schranken für den Komplexitätsbereich eines Problems definiert werden [21]. Von besonderem Interesse ist in der vorliegenden Arbeit die obere Schranke eines Problems in Abhängigkeit seiner Größe ($\mathcal{O}(n)$), die einen Anhaltspunkt für die maximal mögliche Berechnungskomplexität bzw. -dauer liefert. In der theoretischen Informatik werden algorithmische Probleme ähnlicher Berechnungskomplexität in sogenannte Komplexitätsklassen gruppiert.

Zwei wichtige Komplexitätsklassen sind polynomiell ($\mathcal{P}$) und nichtdeterministisch-polynomiell ($\mathcal{NP}$). Probleme der Komplexitätsklasse $\mathcal{P}$ sind im Allgemeinen „schnell“, also in polynomieller Zeit ($\mathcal{O}(n^k)$) lösbar, da für diese Probleme Algorithmen mit polynomieller Laufzeit bekannt sind [21], [23].

Für Probleme der Klasse $\mathcal{NP}$ sind i.A. keine solchen Algorithmen bekannt [23]. Diese Klasse umfasst Probleme, für die zwar die Güte eines vorliegenden Lösungskandidaten „schnell“ mithilfe einer Zielfunktion überprüft werden kann, das Finden einer Lösung allerdings nicht zwingend in polynomieller Zeit erfolgen muss [21]. $\mathcal{NP}$-vollständige Probleme liegen in $\mathcal{NP}$ und sind zusätzlich $\mathcal{NP}$-schwer. Somit sind sie die schwersten Probleme in $\mathcal{NP}$ (siehe Abbildung 3.3). Ein Problem ist $\mathcal{NP}$-schwer, wenn ein Algorithmus zur Lösung dieses Problems polynomiell auf einen Algorithmus reduzierbar ist, der jedes Problem in $\mathcal{NP}$ lösen kann [21]. $\mathcal{NP}$-schwere Probleme sind folglich mindestens so schwer wie das schwerste Problem in $\mathcal{NP}$ [23]. Das nach wie vor ungelöste $\mathcal{P}$-$\mathcal{NP}$ (P vs. NP) Problem wird an dieser Stelle nicht weiter diskutiert, da die aktuell gängige Annahme $\mathcal{P} \neq \mathcal{NP}$ für die vorliegende Arbeit ausreichend ist (siehe linke Seite in Abbildung 3.3).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

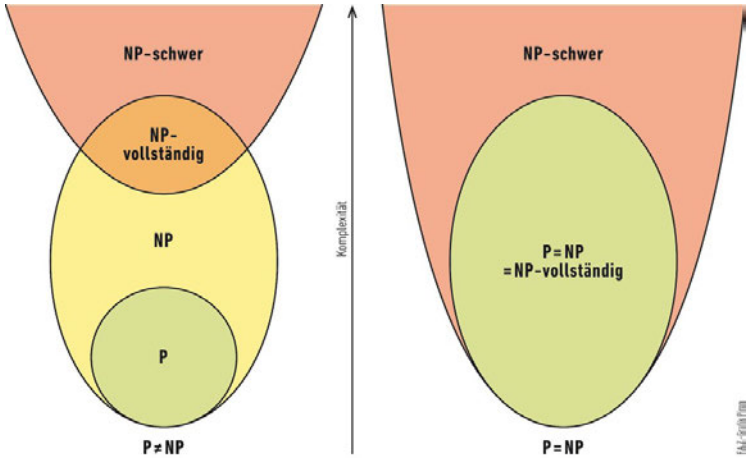


Abbildung 3.3: Komplexitätsklassen im Überblick [24]

3.1.5 Das Handlungsreisendenproblem

Das Handlungsreisendenproblem (TSP) soll zur Veranschaulichung eines $\mathcal{NP}$-schweren Optimierungsproblems dienen und wird deshalb an dieser Stelle kurz erläutert. Es ist eines der gängigsten Probleme, die sowohl für Veranschaulichung, als auch Benchmarking verschiedenster Such- bzw. Optimierungsverfahren verwendet wird und ist $\mathcal{NP}$-**schwer** [21], [22], [25]–[28]. Es stellt sich wie folgt dar: Ein*e Handlungsreisende*r will eine vorgegebene Menge an Orten hintereinander auf einer Rundreise besuchen und dabei seine/ihre Reisedistanz minimieren. Die Distanz zwischen den Orten I und J wird hierbei mit $w_{i,j}$ bezeichnet. Im vorliegenden Fall wird das symmetrische TSP dargestellt. Symmetrisch heißt, dass die Distanz zwischen Stadt I und Stadt J unabhängig von der Richtung gleich groß ist ($w_{i,j} = w_{j,i}$). Eine vollständige Lösung für dieses Problem ist eine Kombination von Reisezielen, in der jeder Ort/jede Stadt genau einmal vorkommt und die erste Stadt gleich der letzten Stadt ist. In 3.4 ist ein einfaches Beispiel für das TSP mit den Städten {A,B,C,D,E} und den jeweiligen Distanzen $w_{i,j}$ dargestellt. Eine valide Lösung wäre für den vorliegenden Fall etwa die Sequenz **D-A-C-B-E-D**.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

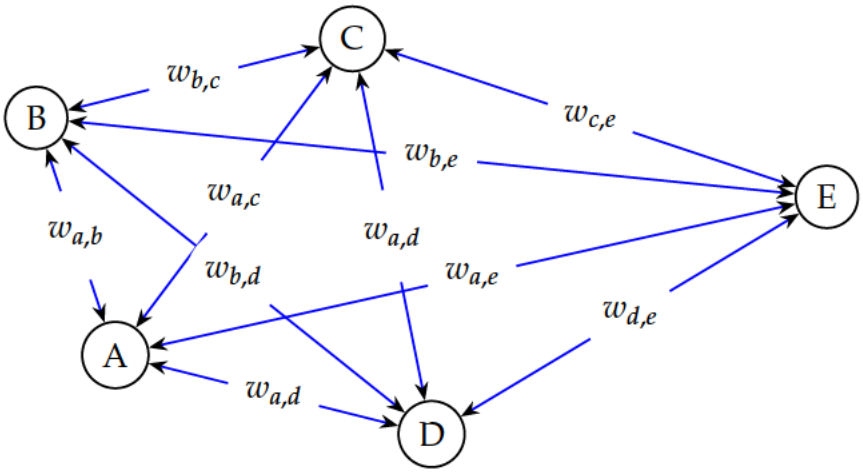


Abbildung 3.4: Veranschaulichung des Handlungsreisendenproblems als Graph

Das TSP ist ein klassisches kombinatorisches Problem, dessen Komplexität mit der Anzahl der Städte n mit $\frac{(n-1)!}{2}$ anwächst [21], [22]. Zur Verdeutlichung der Problematik ist die Anzahl der möglichen Kombinationen abhängig von der Anzahl der zu besuchenden Orte n in Tabelle 3.2, sowie in Abbildung 3.5 dargestellt, wobei die logarithmische Skalierung der vertikalen Achse des Diagramms zu berücksichtigen ist.

n	3	4	5	6	7	8	9	10	11	12
$\mathcal{O}(n)$	1	3	12	69	360	2.520	20.160	181.440	1.814.400	19.958.400

Tabelle 3.2: Größe des Suchraums in Abhängigkeit von der Anzahl der Städte im TSP

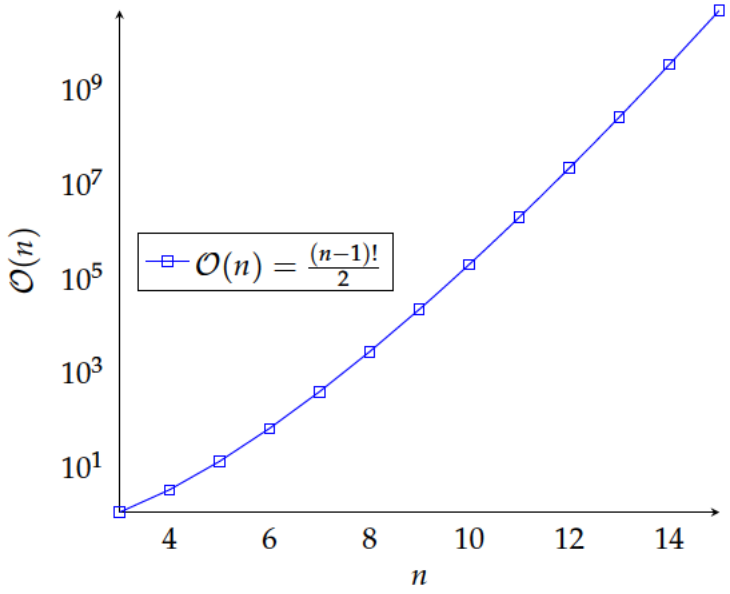


Abbildung 3.5: Komplexität des TSP in Abhängigkeit von n

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

3.1.6 No Free Lunch Theorem

Wolpert und Macready haben ihrer Publikation aus dem Jahr 1993 zum No Free Lunch Theorem (*deutsch*: nichts-ist-umsonst-Theorem, NFL) gezeigt, dass, obwohl der Wunsch nach einem universell überlegenen Optimierungsalgorithmus auf der Hand liegt, es einen solchen nicht gibt:

> One might expect that there are pairs of search algorithms A and B such that A performs better than B on average, even if sometimes B outperforms A. ... One of the main results of this paper is that such expectations are incorrect [29, p. 67].

Das NFL besagt, dass kein Algorithmus existiert, der im Mittel über alle möglichen Probleme besser ist als ein anderer Algorithmus [29], [30]. Grundlegend trifft das Theorem für ein spezifisches Problem keine Aussage, sondern besagt lediglich, dass es keinen universell überlegenen Optimierungsalgorithmus gibt, der besser ist als alle anderen. Bezogen auf eine konkrete Problemstellung gibt es jedoch sehr wohl mehr und weniger geeignete Verfahren, um zum Ziel zu kommen. Auf den mathematischen Beweis und vertiefende Erläuterungen des NFL wird an dieser Stelle verzichtet und auf die ursprüngliche Veröffentlichung [29] oder die Ausführungen dazu in [22], [25], [30] verwiesen.

3.2 Optimierungsverfahren im Überblick

In mehr oder weniger allen Bereichen des heutigen Alltags begegnet der Mensch Optimierungsproblemen. Beginnend mit der recht einfachen Frage nach der optimalen Weckzeit, um rechtzeitig zur Arbeit zu kommen und gleichzeitig seine Ruhezeit zu maximieren. Auch mit komplexeren Aufgaben, wie die der optimalen Routenplanung, um, unter Berücksichtigung von Reisedauer, Ressourcenverbrauch, Transportmittel etc. von Punkt A nach Punkt B zu kommen, wird der Mensch im Alltag konfrontiert. Für Probleme dieser Art hat der Mensch sich einfache Hilfsmittel und Entscheidungsstrategien antrai-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

niert (siehe Abschnitt 3.2.2) oder Methoden entwickelt.
In der Technik ist man mit Optimierungsproblemen in der Entwicklung, wie der optimalen Auslegung eines Fahrzeugs, das gleichzeitig energiesparend, leistungsfähig und sicher sein soll, konfrontiert.
Die Vielzahl an Paketen vom Versandhandelshaus des Vertrauens sollen möglichst schnell und dabei gleichzeitig ressourceneffizient zugestellt werden, was eine optimale logistische Tourenplanung erfordert.
Die genannten Probleme haben alle eines gemein: die Herausforderung eine optimale Lösung für ein Problem mit mehren, häufig konkurrierenden, Zielen zu finden. Um solche Probleme in angemessener Art und Weise zu lösen, gibt es verschiedenste Methoden, die sich primär in ihrer **Genauigkeit** und **Laufzeit** unterscheiden.
Optimierungsprobleme und dafür geeignete Verfahren lassen sich insbesondere auch nach der Art des Problems (stetig vs. unstetig/diskret) und der Definition der zu optimierenden Zielfunktion unterscheiden. Die vorliegende Arbeit beschäftigt sich ausschließlich mit der Lösung diskreter (gemischt-) ganzzahliger Optimierungsprobleme (IP, MIP), also kombinatorischen Optimierungsproblemen, wie sie in Abschnitt 3.1.3 beschrieben wurden. Deshalb werden nachfolgend ausschließlich geeignete Verfahren zur Behandlung solcher Probleme diskutiert. MIPs unterscheiden sich grundlegend in ihrer Gestalt von „klassischen“ Optimierungsproblemen mit wohldefinierten 2-fach stetig differenzierbaren Funktionen, die durch einfaches Differenzieren und Lösen eines Gleichungssystems gelöst werden können. Ein Einordnungsversuch der gängigsten Verfahren im Kontext dieser Arbeit ist in Abbildung 3.6 dargestellt. Hierbei sei erwähnt, dass diese Übersicht keinen Anspruch an Vollständigkeit stellt, sondern lediglich einen Überblick über einige der gängigsten Verfahren bieten soll.

3.2.1 Exakte Verfahren

Die erste übergeordnete Gruppe an Optimierungsverfahren bilden die exakten Optimierungsverfahren. Exakte Verfahren garantieren eine optimale (exakte) Lösung für eine gegebene Problemstellung. Dieser Anspruch an eine exakte Lösung bringt im Falle der (multikriteriellen) Optimierung von größeren oder komplexeren Problemen hohe Lauf-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

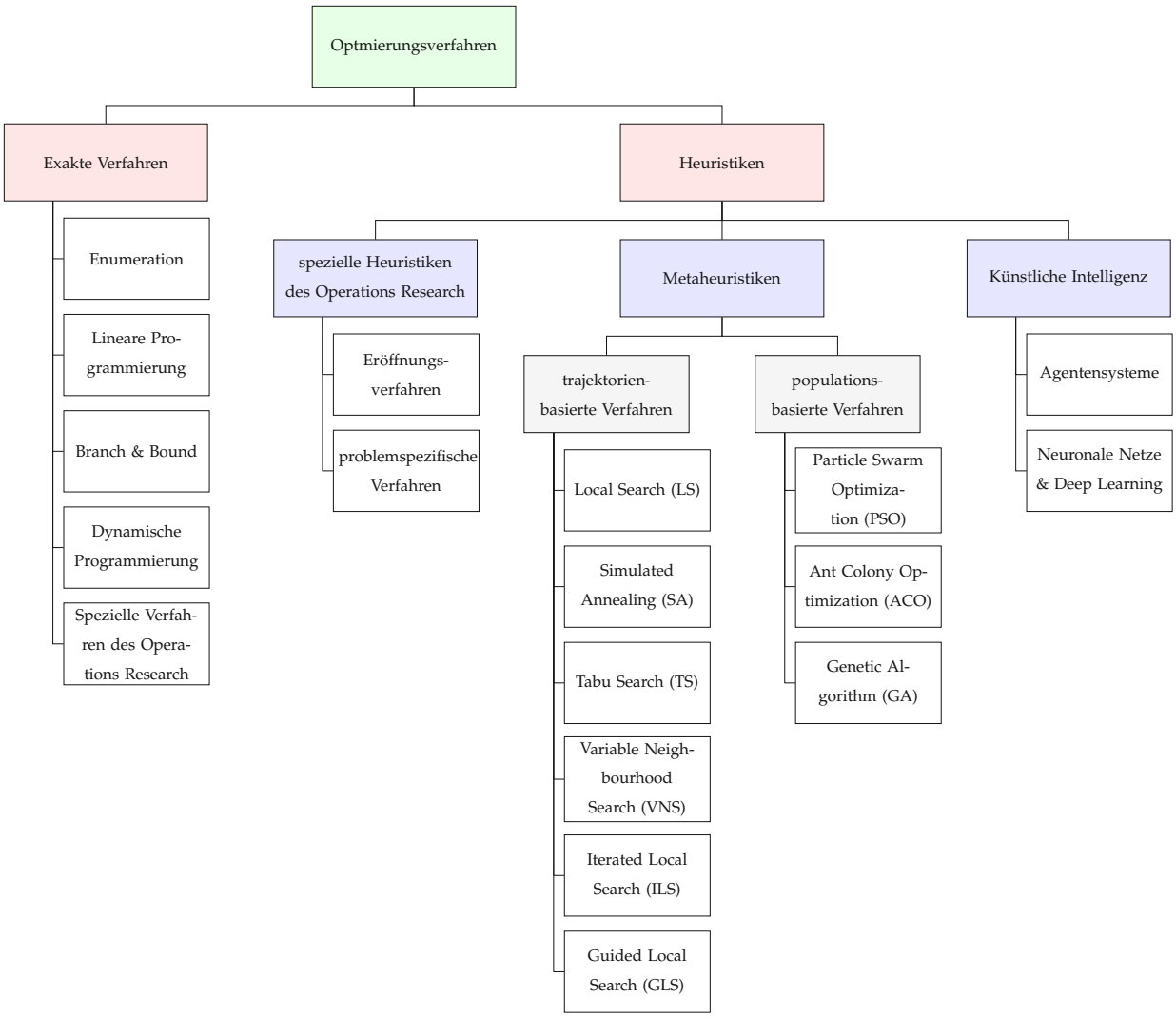


Abbildung 3.6: Übersicht über gängige Optimierungsverfahren für Probleme der Reihenfolgeplanung in Anlehnung an [21], [31]–[37]

zeiten mit sich, wobei das Finden einer, näherungsweise, optimalen Lösung bei vorzeitigem Abbruch der Verfahren nicht garantiert werden kann. Demzufolge eignen sich exakte Verfahren insbesondere für die Lösung von Problemen überschaubarer Komplexität bzw. Größe.

Im Allgemeinen lassen sich die Such- bzw. Lösungsräume von kombinatorischen Optimierungsproblemen hervorragend mithilfe von Bäumen abbilden, wobei die Tiefe des Baumes mit jeder zusätzlichen Entscheidungsvariable in der Kombination um 1 zunimmt. Das einfache Beispiel in Abbildung 3.7 zeigt den aufgespannten Suchraum eines kombinatorischen Optimierungsproblems mit drei binären Entscheidungsvariablen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

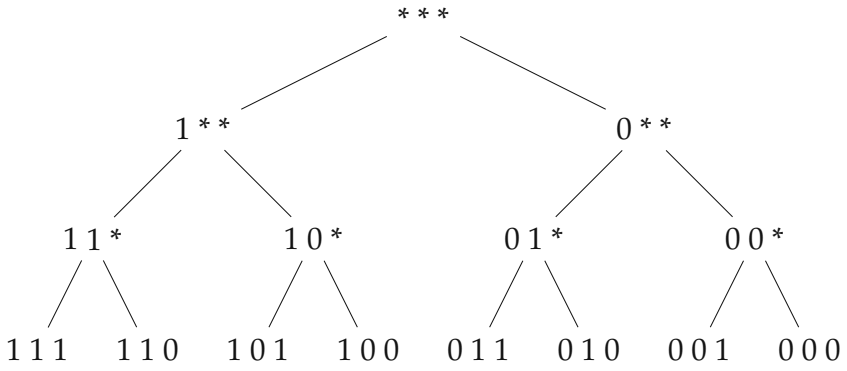


Abbildung 3.7: Suchraum eines kombinatorischen Optimierungsproblems mit drei binären Entscheidungsvariablen

Enumerative Verfahren zählen den gesamten möglichen Lösungsraum bzw. Suchraum auf (enumerieren), durchsuchen diesen und wählen aus den vorliegenden Lösungskandidaten den Besten aus.

Unterschieden werden kann bei der Enumeration in informierte und uninformierte Suchverfahren. **Uninformierte Suchverfahren** umfassen z.B. die Breiten oder Tiefensuche (breadth- oder depth-first search) oder die uniforme Kostensuche (uniform-cost search) [21]. Die Laufzeit dieser Verfahren beläuft sich auf $\mathcal{O}(b^d)$ mit dem Verzewigungsfaktor b (Anzahl der Verzweigungen je Knoten) und der Tiefe des Baumes d [21]. Im vorliegenden Fall eines binären Baumes ist der Verzweigungsfaktor $b = 2$. Die Laufzeit ist demzufolge $\mathcal{O}(2^d)$, was der Anzahl der möglichen Lösungskombinationen gleich kommt. Das ist auch intuitiv begründbar, da in diesen Verfahren jede mögliche Lösung getestet wird.

Informierte Suchverfahren wie der A*-Algorithmus (A-Star) oder die Bestensuche (best-first search) nutzen eine problemspezifische Bewertungsfunktion, um eine zielgerichtete Suche zu ermöglichen [21]. Diese Bewertungsfunktion schätzt die Kosten eines Pfades zwischen dem aktuellen Knoten im Graphen und dem Zielknoten, also dem Optimum, ab. Basierend auf der so ermittelten Bewertung wird über den weiteren Suchverlauf entschieden und zielgerichtet durch den Lösungsraum navigiert [21]. Im besten Fall, also wenn die Bewertungsfunktion genau ist, haben diese Verfahren eine Laufzeit von $\mathcal{O}(bd)$, im schlimmsten Fall kann die Laufzeit der eines uninformierten Suchverfahrens gleich kommen und somit $\mathcal{O}(b^d)$ betragen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Lineare Programmierung und im vorliegenden Fall ganzzahlige lineare Programmierung kann verwendet werden, um lineare Optimierungsprobleme zu lösen. Optimierungsprobleme sind linear, wenn deren Zielfunktion linear von den Entscheidungsvariablen des Problems abhängig ist und alle Entscheidungsvariablen untereinander nur linear voneinander abhängen bzw. unabhängig voneinander sind [21]. Ganzzahlige lineare Optimierungsprobleme können in klassische lineare Optimierungsprobleme übergeführt werden, indem ihre Ganzzahligkeitsbedingungen vernachlässigt und sie somit als klassische kontinuierliche Probleme betrachtet werden. Diese Vernachlässigung von Randbedingungen wird als Relaxation bezeichnet [21]. Das relaxierte ganzzahlige Optimierungsproblem kann mit klassischen Verfahren der linearen Optimierung gelöst werden. Das verbreitetste lineare Optimierungsverfahren ist das Simplex Verfahren [38]. Die so erhaltene Lösung ist i.A. nicht ganzzahlig und muss anschließend wieder angepasst werden, um die Randbedingungen des Ausgangsproblems zu erfüllen. Diese Anpassung kann beispielsweise durch Runden oder durch die Anwendung lokaler Suchverfahren erfolgen [21]. Die so erhaltene ganzzahlige Lösung ist allerdings nicht notwendigerweise (näherungsweise) optimal, kann allerdings vielfach als Anhaltswert für weitere Optimierungsschritte verwendet werden. Für Minimierungsprobleme kann die optimale Lösung des relaxierten Problems beispielsweise als untere Schranke herangezogen werden [38].

Branch – and Bound Verfahren zerlegen das zu lösende Problem und damit auch den zugehörigen Suchraum in kleinere Subprobleme durch die Einführung von zusätzlichen Randbedingungen und Entscheidungsvariablen [21], [38]. Diese Verfahren bestehen aus zwei Schritten, dem „Branching“ (verzweigen) und dem anschließenden „Bounding“ (beschränken). Beim „Branching“ wird der betrachtete Suchraum (Baumstruktur) durch Einführen von zusätzlichen Randbedingungen aufgeteilt. Beim „Bounding“ werden anschließend irrelevante Zweige des Baumes abgeschnitten, wodurch der Suchraum systematisch verkleinert wird. Irrelevant bedeutet in diesem Fall, dass die in einem Zweig liegenden Lösungen entweder nicht zulässig sind oder hinsichtlich ihrer Zielfunktionswerte schlechter sind als eine bereits gefundene Lösung. Am Beispiel des TSP würde das Branching dem Fixieren einer Route zwischen zwei Orten entsprechen und das Boun-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

ding dem Abschneiden von Ästen im Baum, die zwangsläufig eine längere Gesamtdistanz ergeben würden, als die Distanz einer bereits gefundenen Lösung. Als Startwert kann etwa die Lösung des relaxierten Ausgangsproblems verwendet werden. Nachdem Branch – and Bound Verfahren im schlimmsten Fall den gesamten Lösungsraum nach einer optimalen Lösung durchsuchen, und die Tiefe n der Baumstruktur des Lösungsraums von der Anzahl der Entscheidungsvariablen abhängt, ist die Laufzeit dieser Verfahren mit $\mathcal{O}(2^n)$ beschränkt [21].

Dynamische Programmierung beruht auf dem Optimalitätsprinzip von Bellman [38] und der Idee, ein Problem in mehrere, sequenziell lösbare Teilprobleme aufzuteilen und diese vom letzten Schritt beginnend rückwärts zur gesamten Lösung zu aggregieren. Jeder Schritt in dieser Abfolge entspricht einer Entscheidung, die im Ausführungsverlauf des Verfahrens getroffen wird. Im Falle des TSP würde man ausgehend vom letzten Ort in der Rundreise in jedem Schritt des Optimierungsverfahrens einen weiteren Ort als Vorgänger festlegen. Die Auswahl des Vorgängers erfolgt immer so, dass die gesamte Reisedistanz bis zum aktuell betrachteten Punkt minimiert wird. Bei diesen Berechnungen wird von der rekursiven Definition der lokalen Zielfunktionen Gebrauch gemacht, indem mehrfach verwendete Lösungen zwischengespeichert werden [21]. Dynamische Programmierung ist jedoch nur auf eine sehr eingeschränkte Gruppe an Problemen anwendbar und die Laufzeit ist problemabhängig, aber i.d.R niedriger als bei klassischer Enumeration [21].

3.2.2 Heuristiken

Heuristiken sind problemspezifische Lösungsstrategien zur Bestimmung von nicht notwendigerweise optimalen Lösungen und werden i.d.R eingesetzt, wenn für das vorliegende Problem keine effizienten Verfahren bekannt sind [39]. Laut [40] soll eine Heuristik als

> ...eine nichtwillkürliche und häufig iterative Methode verstanden werden, die darauf abzielt, für eine gegebene Problemstellung in begrenzter Zeit eine

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

oder mehrere möglichst gute Lösungen zu finden, ohne daß garantiert werden kann, eine global optimale Lösung zu finden.

Diese Näherungsverfahren können in Metaheuristiken, spezielle Verfahren aus dem Operations Research und künstliche Intelligenzen unterteilt werden.

3.2.2.1 Spezielle Verfahren des Operations Research

Im Operations Research (*deutsch*: Unternehmensforschung, OR) kommen häufig problemspezifische Verfahren zum Einsatz, die auf Expertenwissen basieren (siehe auch Anhang A.2) und auf eine spezielle Problemklasse angewandt werden können.

Eröffnungsverfahren dienen zur Erzeugung einer zulässigen Startlösung für das vorliegende Problem. Solche Verfahren werden auch als Konstruktionsverfahren bezeichnet, da sie eine Lösung konstruieren [41].
Unter anderem können hierfür Greedy-Heuristiken, uninformierte - oder auch vorausschauende Verfahren zum Einsatz kommen [42]. Der Begriff „Greedy“ im Kontext der Algorithmik wird in Anhang A.2 erläutert. Greedy-Heuristiken liegen i.d.R statische Prioritätsregeln zugrunde, auf deren Grundlage Entscheidungen im Sinne des Optimierungsproblems getroffen werden. Vorausschauende Verfahren hingegen versuchen mithilfe dynamischer Prioritätsregeln, die nach jeder Entscheidung angepasst werden, bessere Ergebnisse als mit statischen Regelwerken zu erzielen [42].
Im OR existiert eine Vielzahl an problemspezifischen Verfahren, die an dieser Stelle jedoch nicht weiter erläutert werden, da sie für die vorliegende Arbeit keine Relevanz haben.

3.2.2.2 Metaheuristiken

Als Metaheuristiken werden Heuristiken bezeichnet, die unabhängig von einem konkreten Problem sind [43]. Diese Verfahren können als übergeordnete Lösungsstrategien betrachtet werden, die auf konkrete Probleme angewandt werden können. Alternativ

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

können sie auch als Rahmenwerke verstanden werden, die spezielle Heuristiken vereinen, um so einen Suchraum effizient zu durchsuchen und haben ihren Ursprung in den 1970er-Jahren [36]. Der Begriff Metaheuristik wurde dabei erstmals in [44] verwendet. Zuvor wurde diese Gruppe an Verfahren oftmals als *moderne Heuristiken* bezeichnet [36]. Häufig werden auch Algorithmen mit inhärenter Randomisierung und lokaler Suche als metaheuristische Algorithmen bezeichnet [34].
Zusammenfassend können Metaheuristiken als übergeordnete Verfahren verstanden werden, die problemspezifische Verfahren steuern, um so den Lösungsraum möglichst effizient und effektiv zu durchsuchen.
Ein großer Teil der metaheuristischen Verfahren ist von Abläufen, die in der Natur vorkommen, inspiriert. Unterscheiden kann man metaheuristische Verfahren in trajektorienbasierte und populationsbasierte Verfahren [34], [35], [37]. Der große Vorteil von Metaheuristiken, im Vergleich zu exakten Methoden, liegt in der verhältnismäßig geringen Laufzeit, in der eine näherungsweise optimale Lösung gefunden werden kann.

Metaheuristische Optimierungsverfahren sind im Idealfall sowohl in der Lage, den gesamten Suchraum zu erkunden (*Exploration*), also ihn nach erfolgversprechenden Regionen zu durchkämmen und gleichzeitig die Information über diese Regionen effektiv auszunutzen, um neue Optima zu finden (*Exploitation*) [22], [36], [40].

Trajektorienbasierte Metaheuristiken vollziehen eine Suche im Suchraum, indem sie eine einzelne Lösung iterativ verbessern und somit einen Pfad (Trajektorie) durch den Suchraum erzeugen [34], [35], [45].

Local Search (*deutsch*: Lokale Suche, LS) umfasst eine Vielzahl verschiedener heuristischer Suchverfahren, die ausgehend von einer Lösung in deren Nachbarschaft nach besseren Lösungen suchen [46]. Wie der Name bereits erahnen lässt, sind diese Verfahren lediglich geeignet, um lokale Optima zu finden [36]. Um diese Schwachstellen von ausschließlich lokal operierenden Suchverfahren, die ihre Stärken in der Exploitation haben, um die Fähigkeit zur Exploration zu erweitern, wurden die nachfolgenden Verfahren entwickelt [36]. Damit können lokale Optima überwunden werden und die

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Wahrscheinlichkeit ein globales Optimum zu finden nimmt zu.

Simualted Annealing (*deutsch*: Simulierte Abkühlung, SA) orientiert sich am Abkühlungs- und Kristallisationsverhalten von Metallen und Gläsern und wurde erstmals in [47] veröffentlicht. Im Ablauf des Suchprozesses wird in jedem Schritt mit der aktuellen Lösung l die nächste Lösung l' zufällig in der Nachbarschaft von l ausgewählt. Ist l' besser als l, so wird sie jedenfalls als neue Lösung akzeptiert. Ist l' schlechter als die aktuelle Lösung, so wird sie mit einer Wahrscheinlichkeit p trotzdem als neue Lösung akzeptiert. Dabei ist diese Wahrscheinlichkeit eine Funktion der aktuellen Temperatur T ($p = f(T)$) und nimmt mit fallender Temperatur, also im Abkühlungsverlauf, ab. Die Temperatur wird in jedem Iterationsschritt des Verfahrens angepasst und beeinflusst so die Auswahlwahrscheinlichkeit im darauffolgenden Schritt. Mit diesem Verhalten wird erreicht, dass im frühen Stadium häufiger schlechtere Lösungen akzeptiert werden und somit Nachbarschaften verlassen (Exploration) und lokale Optima überwunden werden. Am Ende des Suchprozesses wird dadurch im unmittelbaren Umfeld der Lösungen optimiert (Exploitation) [34], [36].

Tabu Search (*deutsch*: Tabu Suche, TS) wurde erstmals in [44] beschrieben, wobei „Tabu“ in diesem Kontext bedeutet, dass die Vergangenheit in Form von sogenannten Tabu-Listen bei der Suche nach neuen Lösungen berücksichtigt wird. Diese Listen enthalten Lösungsmerkmale, die kürzlich in Lösungen verwendet wurden, um so die mehrfache Betrachtung von gleichen oder ähnlichen Lösungen zu vermeiden und aus lokalen Optima auszubrechen. In jedem Iterationsschritt des Verfahrens werden Lösungen in der Nachbarschaft der aktuellen Lösung erzeugt, die den Kriterien der Tabu-Listen entsprechen. Diese Lösungen werden von der gesamten Nachbarschaft ausgeschlossen, wodurch eine reduzierte Menge an zulässigen Lösungen verbleibt. Aus dieser Menge wird die beste Lösung ausgewählt. Um zu vermeiden, dass gute Lösungen ausgeschlossen werden, gibt es sogenannte Aspirationskriterien, die die Verletzung von Tabus ermöglichen. Nach jedem Iterationsschritt werden die Tabu-Listen aktualisiert, indem neue Kriterien hinzugefügt und alte entfernt werden [36], [44].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Zu den trajektorienbasierten Verfahren zählen darüber hinaus Verfahren wie die Variable Neighborhood Search (*deutsch*: Variable Nachbarschaftssuche, VNS), Iterated Local Search (*deutsch*: Iterierte Lokale Suche, ILS) und die Guided Local Search (*deutsch*: Geführte Lokale Suche, GLS) [46], [48], [49].

Populationsbasierte Metaheuristiken verwenden im Gegensatz zu trajektorienbasierten Verfahren in jedem Lösungsschritt eine Vielzahl von einzelnen Lösungen (Population). Die Lösungen einer Population werden in jedem Iterationsschritt durch entsprechende Operatoren verändert [34], [35].

Particle Swarm Optimization (*deutsch*: Partikelschwarmoptimierung, PSO) wurde in [50] vorgestellt und orientiert sich am Verhalten von Vogel- oder Fischschwärmen auf Nahrungssuche. In einem Schwarm an Lösungen (Population) hat jedes Individuum eine Geschwindigkeit v, eine Position x und seine bisher beste Lösung p, wobei mit der geteilten Information über die insgesamt beste Lösung $\hat{p}$ die Schwarmintelligenz modelliert wird, die das Suchverhalten des gesamten Schwarms steuert. In jedem Iterationsschritt werden diese drei Größen jedes Individuums, basierend auf seiner Vergangenheit und auf der bisher besten Lösung im Schwarm $\hat{p}$, angepasst [36]. So bewegen sich die einzelnen Partikel durch den Lösungsraum und orientieren sich dabei sowohl zufällig als auch am aktuellen globalen Optimum $\hat{p}$ und dem eigenen Optimum p. Wenn durch diese Bewegung ein neues globales Optimum gefunden wird, so wird $\hat{p}$ aktualisiert und damit das weitere Bewegungsverhalten des gesamten Schwarms beeinflusst [34].

Ant Colony Optimization (*deutsch*: Ameisenalgorithmus, ACO) ist ein weiteres, von der Natur inspiriertes Verfahren, und spiegelt das Verhalten von Ameisen auf der Suche nach dem kürzesten Weg zwischen Nahrungsquellen und ihrem Nest unter Einsatz von Pheromonspuren wider. Verfahren, die sich an diesem Phänomen orientieren, wurden zum ersten Mal in [51] unter dem Begriff ACO zusammengefasst, der erste dieser Algorithmen ist in [52] beschrieben und wurde als Ant System (*deutsch*: Ameisensystem, AS) bezeichnet. Das System besteht aus einer Population von künstlichen Ameisen, die

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

auf der Suche nach einer optimalen Lösung eine sogenannte Pheromonspur hinterlassen. Bei der Erzeugung einer Lösung für ein kombinatorisches Optimierungsproblem, traversiert eine künstliche Ameise einen Graphen, der die Gesamtheit aller möglichen Lösungen im Lösungsraum abbildet. Beim Traversieren dieses Graphen trifft die Ameise an jeder Stelle, die eine Komponente der Lösung widerspiegelt, eine Entscheidung über ihren weiteren Weg. Diese Entscheidung passiert mithilfe einer stochastischen Funktion, die von den einzelnen Pheromonkonzentrationen der nächsten möglichen Schritte abhängt. Am Ende hinterlässt die Ameise eine bestimmte Menge an Pheromonen, die von der Güte der so erzielten Lösung abhängt. Nachfolgende Ameisen werden basierend auf den aktualisierten Pheromonkonzentrationen zunehmend in vielversprechende Gebiete des Lösungsraums geleitet [53]. Einfach illustriert werden kann dieses Verfahren anhand des TSP, da sich das Vorgehen der künstlichen Ameisen direkt auf die Suche nach dem kürzesten Weg eines Handlungsreisenden anwenden lässt.

Genetic Algorithm (*deutsch*: Genetischer Algorithmus, GA) ist ein Verfahren, das sich an der natürlichen Evolution orientiert und der Gruppe der evolutionären Algorithmen angehört [22]. GAs wurden von John Holland und seinen Studierenden in den 1960er und 1970er-Jahren entwickelt und zum ersten Mal in der Erstausgabe von [54] beschrieben. Die Grundidee eines GAs beruht auf dem Prinzip „Survival of the Fittest“, demzufolge sich die stärksten Individuen einer Population durchsetzen und fortpflanzen. Der Algorithmus basiert auf den Operationen Rekombination (auch Kreuzung) (*engl:* crossover), Mutation (*engl:* mutation) und Selektion (*engl:* selection) und den zugehörigen Operatoren, die auf die Individuen angewandt werden. Ein Individuum ist im einfachsten Fall eine binärkodierte Zeichenkette, kann in der Theorie allerdings unterschiedlichste Gestalten annehmen und stellt die Repräsentation einer Lösung für das vorliegende Problem dar. Entsprechend der Formulierung der Individuen, müssen die Selektions-, Mutations- und Rekombinationsoperatoren so formuliert werden, dass sie auf die jeweiligen Individuen anwendbar sind. Ausgehend von einer Population P bestehend aus μ Individuen und deren Fitness F (Güte hinsichtlich der Zielfunktion) wird eine Menge an λ Individuen aus P selektiert (Elternselektion) und unter Anwendung des Rekombinationsoperators in Kinder übergeführt, indem Merkmale der Elternindi-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

viduen „vermischt“ werden. Anschließend an die Rekombination erfolgt eine Mutation der so gewonnenen Kindindividuen. Für die mutierten Kindindividuen wird deren Fitness bestimmt und genutzt, um aus der vorigen Population und den neuen Kindindividuen eine neue Population mithilfe der Umweltselektion zu erzeugen [22]. Die Selektion erfolgt in der Regel zumindest in einer der beiden Selektionsstufen fitnessproportional, also entsprechend der Güte der Individuen. So „überleben“ über mehrere Generationen hinweg jeweils die besten Individuen und eine näherungsweise optimale Lösung kann gefunden werden.

3.2.2.3 Künstliche Intelligenz

Methoden der künstlichen Intelligenz bieten eine attraktive Alternative zu den zuvor erläuterten Verfahren. Systeme, wie künstliche neuronale Netze, werden in aktuellen Veröffentlichungen eingesetzt, um akademische Probleme zu behandeln. Methoden, die dem überwachten Lernen (engl. supervised Learning) zuzuordnen sind, erfordern Daten, die diese Art des Lernens ermöglichen. Besagte Daten müssen aus (historischen) (Experten-) Entscheidungen generiert werden. Das zugrundeliegende Modell versucht anschließend diese Entscheidungen zu imitieren [55]. Die Limitierung solcher datengetriebener Verfahren liegt im Mangel an verfügbaren Daten zum Training eines Modells. Davon abgesehen kann ein Modell, das mithilfe von (historischen) Daten trainiert wird und in weiterer Folge nicht aus „Erfahrung“ lernt, lediglich so gute Entscheidungen treffen, wie sie in den Daten abgebildet sind. Erfahrungsbasierte Systeme hingegen wählen einen anderen Ansatz und erscheinen vielversprechend, wenn es darum geht, bestehende Verfahren zukünftig zu übertreffen [55].

Agentenbasierte Systeme wie z.B. Reinforcement Learning (*deutsch*: Bestärkendes Lernen, RL)-Agenten erscheinen vielversprechend für die Behandlung kombinatorischer Optimierungsprobleme [56], befinden sich allerdings noch in einem relativ frühen Stadium, was ihre Entwicklung anbelangt [55]. RL-Systeme basieren auf der Idee, des Lernens eines Agenten durch Interaktion mit dessen Umfeld. Konkret ist ein solcher Agent mit einem Entscheidungsproblem und der Aufgabe dieses zu lösen konfrontiert. Ent-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

sprechend einer getroffenen Entscheidung verändert sich der Zustand (State) der Umgebung des Agenten und der Agent selbst erhält Feedback aus dieser Umgebung in Form einer Belohnung. Das Ziel des Agenten ist es, seine Belohnung zu maximieren, wobei die Belohnung mit der Güte der getroffenen Entscheidung zusammenhängt. Basierend auf der Belohnung soll der Agent lernen, optimale Entscheidungen in einer gegebenen Situation (Umgebung) zu treffen [55], [57]. Für graphenbasierte Optimierungsprobleme (siehe. TSP) existieren RL-basierte Ansätze, die in der Lage sind, für vorliegende Probleme Greedy-Heuristiken, also regelbasierte Entscheidungsmodelle, zu lernen wie z.B. in [58] beschrieben.

3.3 Erkenntnisse

Aus den zuvor angeführten Erläuterungen zur Problemkomplexität und den Laufzeiteigenschaften von exakten Optimierungsverfahren kann geschlossen werden, dass exakte Optimierungsverfahren nur bedingt für $\mathcal{NP}$-schwere Optimierungsprobleme wie dem RSP geeignet sind und das nur, sofern die Problemgröße sich stark in Grenzen hält [21]. Für allgemeine Probleme der Klasse $\mathcal{NP}$-schwer kann deren Laufzeit als zu hoch und die Anwendung dieser Verfahren somit als nicht zielführend angesehen werden. Daraus ergibt sich die Notwendigkeit der Verwendung eines Näherungsverfahrens, das insbesondere für größere Probleminstanzen in vertretbarer Laufzeit annähernd optimale Lösungen findet. Datengetriebene Verfahren, wie überwachte Lernmethoden erscheinen für das vorliegende Problem nicht verwendbar, da keine nutzbaren Daten vorliegen. RL-Systeme scheinen zwar vielversprechend, sind allerdings zum aktuellen Zeitpunkt noch zu wenig weit entwickelt, um eine fundierte Aussage über deren Anwendbarkeit treffen zu können, geschweige denn, um einen RL-Agenten für das vorliegende Problem implementieren zu können.

Somit verbleibt die Familie der Heuristiken zur Anwendung in dieser Arbeit. Regelbasiert-heuristische Verfahren werden häufig und erfolgreich als Eröffnungsverfahren für metaheuristische Optimierer eingesetzt, reichen jedoch üblicherweise nicht aus, um die vollständige Problemkomplexität abbilden und somit den Suchraum ausnutzen zu können.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Zur Wahl einer geeigneten Methode ist in [59] ein Entscheidungsmodell angeführt. Für das vorliegende Problem decken sich die gezogenen Schlussfolgerungen mit den Aussagen dieser Veröffentlichung und der Wahl eines inkrementellen, (meta-)heuristischen Verfahrens.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Kapitel 4

State-of-the-Art

Das, diesem Ressourcenzuteilungsproblem zugrundeliegende, mathematische Problem ist ein kombinatorisches Optimierungsproblem und lässt sich als (gemischt) ganzzahliges Optimierungsproblem (IP bzw. MIP) beschreiben. Das beschriebene Planungsproblem wird in der einschlägigen Fachliteratur als RSP bezeichnet [1], [16], [60], wobei es in der Wissenschaft bisher vergleichsweise spärlich behandelt wurde. Erschwerend zur dünn gesäten Literatur kommt hinzu, dass Ergebnisse aus internationaler Literatur nur bedingt auf die Problemstellung in Österreich übertragbar sind, da sich die Bestimmungen für die postgraduale Ausbildung von ärztlichem Personal von Land zu Land häufig stark unterscheiden und damit andere Herausforderungen und Problemstellungen einhergehen. Außerdem variieren die betrachteten Zeiträume und zeitlichen Granularitäten in den verfügbaren Publikationen zum RSP stark. Eine einschlägige Literaturübersicht zum RSP aus dem Jahr 2019 ist in [1] zu finden und eine umfassende Literaturanalyse zur generellen Einsatzplanung aus 2017 wurde in [61] durchgeführt.

Die wenigsten Veröffentlichungen behandeln das vorliegende mehrperiodige, langfristige Planungsproblem, wie es in [16] beschrieben wird, sondern Probleme mit einem Zeithorizont von einem oder wenigen Monaten bis zu einem Jahr [62] oder gar auf Schichtebene [60] mit einem Zeithorizont von einem Monat, wie es beim Nurse Scheduling Problem (NSP) üblich ist [15], [63]. Ein kurzer Planungshorizont bei gleichzeitig grobem Detaillierungsgrad vereinfacht die Planung ungemein, da die Problemgröße und in wei-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

terer Folge die Problemkomplexität erheblich reduziert wird. Da in der ÄAO 2015 § 21 [6] allerdings das Vorhandensein eines Ausbildungsplans für die gesamte Ausbildung von Turnusärzt*innen, bereits zu Ausbildungsbeginn, gefordert wird, kommt diese Vereinfachung in der vorliegenden Arbeit nicht infrage.
Die, in dieser Arbeit behandelte, Variante des RSP ist von der Klasse $\mathcal{NP}$-schwer (siehe Abschnitt 3.1.4) [1], [15], [16]. Die Behandlung eines solchen $\mathcal{NP}$-schweren Optimierungsproblems ist mit zunehmender Problemkomplexität (Anzahl der zu verplanenden Ressourcen und der damit einhergehenden schnell anwachsenden Größe des möglichen Lösungsraums) nur schwer mit exakten Methoden, wie Branch-and-Bound Algorithmen, in einer vertretbaren Laufzeit möglich [1].

4.1 Existierende Lösungsansätze für die Ärzt*innenausbildungsplanung

Die aktuellste Publikation zum RSP, in vergleichbarer Gestalt zur Problemstellung in Österreich, stammt aus dem Jahr 2019 [15]. In dieser Veröffentlichung wird das Problem mithilfe eines GA behandelt, da ein exaktes Branch and Bound Verfahren nicht in der Lage ist, innerhalb von 24 Stunden für ein Problem mit 36 Ärzt*innen brauchbare Lösungen zu finden. Das betrachtete Problem ist zwar ähnlich, der Autor wählt jedoch einen kompetenzorientierten Ansatz zur Verplanung von Turnusärzt*innen, beschränkt das Planungsproblem auf einen verkürzten Teil der Ausbildung (ein Jahr) und behandelt lediglich ein Krankenhaus und dessen Abteilungen. Die Verplanung von Personen erfolgt dabei im Wochentakt und beschränkt sich ausschließlich auf das Fachgebiet der Anästhesiologie. Im Gegensatz dazu soll in der vorliegenden Arbeit ein verallgemeinertes, zeitbasiertes Verfahren für eine beliebige Anzahl an Krankenhäusern und den zugehörigen Abteilungen, sowie alle zulässigen Ausbildungen gefunden werden. Das Hauptaugenmerk soll dabei auf der Erfüllung betrieblicher Ziele und der Berücksichtigung persönlicher Präferenzen liegen.
In [15] und [1] wird das RSP in ähnlicher Form behandelt, wobei die Erstellung von Ein-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

satzplänen in [15] nur ein Nebenerzeugnis der eigentlichen Arbeit darstellt. In beiden Arbeiten wird ein langfristiges Einsatzplanungsproblem in der Ärzt*innenausbildung in einem deutschen Krankenhaus beschrieben, wobei in diesen Arbeiten zwischen einem aufgabenbezogenem (task-related) und einem zeitbezogenen (time-related) Ansatz der Ausbildungsmodelle und der damit verbundenen Planung unterschieden wird. In Deutschland, wie auch in Österreich und einigen anderen europäischen Ländern, ist der Kompetenzerwerb (Eingriffe) ein zentrales Kriterium für den Abschluss einer Ausbildung [15]. Die Implikationen für die diese Arbeit wurden bereits aus einer praxisbezogenen Betrachtung in Kapitel 1 diskutiert. An dieser Stelle sei lediglich angemerkt, dass in der vorliegenden Arbeit, trotz der aufgabenbezogenen Natur der ÄAO 2015, ein zeitbezogener Ansatz in der Planung gewählt wird, wie es in Österreich aktuell üblich ist. In [15] steht die Kapazitätsbestimmung von Abteilungen als strategisches Entscheidungskriterium für die Ausbildungsplanung im Vordergrund. Ausbildungspläne werden mithilfe von Column Generation (*deutsch*: Spaltengenerierung, CG) erzeugt. CG ist ein Verfahren, das ein lineares Programm löst, indem ein kleiner Teil des vorliegenden Problems gelöst wird und anschließend „spaltenweise“ neue Variablen eingeführt werden und das zu lösende Problem somit sukzessive vergrößert wird. So wird das Problem iterativ aufgelöst, bis alle Variablen des ursprünglichen Problems festgelegt sind [36].
Ein weiterer Ansatz zur Verplanung von Ärzt*innen in Ausbildung mithilfe eines GA wird in [60] beschrieben. Dieses Verfahren erzeugt Dienstpläne auf Schichtebene für einen Monat und unterscheidet sich in der Problemstellung grundlegend von der vorliegenden Arbeit.
Für ähnliche Probleme existieren somit funktionierende Verfahren, die metaheuristische Ansätze verwenden.

In [63] wird ein ähnliches Problem in einer Abteilung eines belgischen Krankenhauses mithilfe eines Branch-and-Price Ansatzes behandelt. Das Verfahren ist jedoch hinsichtlich der Problemgröße stark limitiert und ab 12 Personen und 10 Aufgaben nicht mehr in der Lage in einer vertretbaren Zeit optimale Lösungen zu finden.
Das Verfahren in [62] ist eine Mischung aus einem exakten Löser für IP und heuristischen Ansätzen, sowie interaktiven Bestandteilen, die dem Planungspersonal Eingriffe

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

erlauben. Die genaue Funktionsweise des Verfahrens wird nicht näher erläutert, allerdings erfordert das Verfahren kontinuierliche menschliche Eingriffe und ist nicht in der Lage vollständig automatisch zu planen.

In [64] wird ein Goal-Programming Ansatz zur Erstellung wöchentlicher Rotationen unter Berücksichtigung von Continuity of Care und Fairness innerhalb eines einzigen Krankenhauses in den USA eingesetzt. Dieses Verfahren beschränkt sich ebenfalls auf den Einsatz innerhalb eines Krankenhauses und kann mitunter mit hohen Laufzeiten (40 Stunden bei 57 Personen) verbunden sein, wobei die Laufzeit stark von den Einschränkungen des Problems abhängig ist. Goal-Programming Ansätze erfordern die Zerlegung eines MOPs in mehrere Einzieloptimierungsprobleme (*engl*: Single-Objective Optimization Problems, SOPs) und die Priorisierung der zugehörigen Optimierungsziele. Umgelegt auf das vorliegende Problem würde das bedeuten, Lösungen für jedes einzelne SOP für jedes Kriterium in Tabelle 2.2 zu berechnen. Im beschriebenen Problem in [64] werden vier Optimierungsziele erwähnt, für die 24 verschiedene Optimierungsprobleme (alle möglichen Reihenfolgen der vier genannten Ziele) gelöst werden müssen, um anhand eines hierarchischen Entscheidungsverfahrens (Analytical Hierarchy Process - AHP) die „beste“ Lösung auszuwählen. Bei sechs Kriterien, wie in Tabelle 2.2 aufgelistet, wären anstatt der 24 Lösungen in der Veröffentlichung bereits 720 Lösungen zu bestimmen, um alle Reihungen der Optimierungskriterien abzubilden, was einer Verdreißigfachung der zu lösenden Optimierungsprobleme für ein vergleichbar großes Problem entsprechen würde. Damit wäre eine immense Steigerung der Laufzeit verbunden, da die Laufzeit des Lösers mit zunehmender Anzahl an eingeführten Randbedingungen (für jedes zusätzliche Kriterium) überproportional zunimmt [64]. Goal-Programming Ansätze stehen außerdem in der Kritik, nicht Pareto-effizient zu sein, sofern die Zielwerte der einzelnen Optimierungskriterien zu „pessimistisch“ angesetzt werden [65]. Da in der Publikation weder die Pareto-Effizienz der erreichten Lösungen noch die Definition der entsprechenden Optimierungszielwerte diskutiert werden, wird davon ausgegangen, dass das implementierte Verfahren die Forderung nach Pareto-effizienten Lösungen nicht erfüllt. Davon abgesehen unterscheidet sich die zugrundeliegende amerikanische Ausbildung weitestgehend von der österreichischen, weshalb das Verfahren nicht auf den Einsatz in Österreich übertragbar ist.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

4.2 Lösungsansätze für ähnliche Problemstellungen

Nurse Scheduling Problem

Das NSP stellt ein artverwandtes Problem dar, das in der Wissenschaft jedoch wesentlich intensiver behandelt wurde. Deshalb werden die dafür gängigen Ansätze an dieser Stelle kurz diskutiert. Das NSP beschreibt die Herausforderung der Zuteilung von Krankenpflegepersonal, also die Erstellung eines optimalen Dienstplans unter Berücksichtigung verschiedenster Restriktionen und Zielgrößen, wie Arbeitszeiten, Präferenzen des Personals, Qualifikationsanforderungen, Dienstverträgen etc. [26], [66], [67] und ähnelt somit dem RSP in seiner grundlegenden Problemstellung stark. In zahlreichen Veröffentlichungen zum NSP werden GAs [26], [66]–[73] und andere metaheuristische Verfahren [74] erfolgreich eingesetzt. Die gewählten Ansätze zeigen die Verwendbarkeit von metaheuristischen Verfahren und insbesondere genetischen Algorithmen für artverwandte Problemstellungen. Daraus kann zwar deren prinzipielle Einsetzbarkeit im vorliegenden Fall abgeleitet werden, die konkret beschriebenen Methoden sind jedoch nicht direkt auf das vorliegende Optimierungsproblem übertragbar, da sich die Rahmenbedingungen für die Einsatzplanung von Pflegepersonal und der von ärztlichem Personal in Ausbildung wesentlich voneinander unterscheiden.

4.3 Implikationen für die Umsetzung

Das Problem der Rotationsplanung von Turnusärzt*innen unterscheidet sich grundlegend vom NSP und anderen Einsatzplanungsproblemen durch den betrachteten Planungshorizont, die Anzahl der zu verplanenden Ressourcen sowie die vorliegenden Optimierungsziele und Randbedingungen. Dienstpläne werden in der Regel maximal für einen Zeithorizont von einem oder wenigen Monaten erstellt, die Rotationsplanung hingegen muss zum Teil für 72 Monate (Mindestdauer der Ausbildung in einigen Fachgebieten) durchgeführt werden, um den Anforderungen der ÄAO 2015 gerecht zu werden. Außerdem erfolgt die Verplanung von Ärzt*innen in Ausbildung, üblicherweise

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

zentral innerhalb eines Krankenanstaltenverbundes, also einer Gruppe an Krankenhäusern und nicht auf Krankenhaus- oder gar Stations-Ebene, wie es beim NSP und auch vielen Ansätzen zum RSP der Fall ist.

Um dem Anspruch dieser Arbeit gerecht zu werden, und ein System zu entwickeln, das praxisrelevante Planungsergebnisse für größere Zeiträume (gemäß ÄAO 2015) liefert, ohne dabei zu stark vereinfachende Annahmen zu treffen, werden, basierend auf den Erkenntnissen aus Kapitel 3 und diesem Kapitel, metaheuristische Verfahren und im Speziellen GAs als Basis der Planungsmethode gewählt, da sie für zahlreiche artverwandte Probleme erfolgreich eingesetzt werden. Genetische Algorithmen (und generell populationsbasierte Verfahren) sind robuster, was deren Konvergenzverhalten angeht, da sie üblicherweise besser darin sind, lokale Optima in multimodalen Lösungsräumen zu überwinden (siehe Abschnitt 3.2.2.2).

Im Gegensatz zu den üblicherweise großen Populationen, die bei klassischen GAs Anwendung finden, konnten in der Vergangenheit auch Verfahren mit kleinen Populationen ($|P| < 10$), bei denen die Bewertung von Lösungen sehr rechenintensiv ist, erfolgreich implementiert werden. Gemessen an der Anzahl der Bewertungsvorgänge (die maßgeblich für die Laufzeit sind), konnte gezeigt werden, dass kleine Populationen verbunden mit mehr Generationen (= gleiche Anzahl an Bewertungsvorgängen) zu besseren Ergebnissen führen können [75]. Populationsgrößen in diesem Bereich sind in der Regel bei Evolutionsstrategien (ES) üblich.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

4.4 Genetische Algorithmen

Wie in Abschnitt 3.2.2.2 beschrieben, haben GAs ihre Inspiration in der natürlichen Evolution und nehmen ihren Ursprung in einer Publikation von Holland et al. aus dem Jahr 1975 [54]. Die Genetik, die diesem natürlichen Evolutionsmechanismus zugrunde liegt, wird in der vorliegenden Arbeit nicht weiter erläutert, jedoch sei erwähnt, dass GAs eine stark vereinfachte Vorstellung der Evolution mithilfe der Evolutionsfaktoren **Variation** und **Selektion** simulieren. Die Variation erfolgt hierbei, wie in der Natur, mithilfe von **Rekombination** und **Mutation** [22]. GAs sind eine Sonderform von Evolutionary Algorithms (*deutsch*: Evolutionäre Algorithmen, EAs). Im verallgemeinerten Fall nimmt ein GA eine Gestalt wie jene in Abbildung 4.1 an. Dabei können die Schritte Initialisierung (blau), Bewertung (grau), Selektion (rot), und die Variations-Operatoren Rekombination und Mutation (orange) unterschieden werden. Der äquivalente Ablauf eines verallgemeinerten GA ist in Algorithmus 1 dargestellt. Hierbei gilt zu beachten, dass je nach Ausprägung eines speziellen Algorithmus, einer der beiden Selektionsoperatoren wegfallen kann. Das Abbruchkriterium ist in diesem Algorithmus über die Anzahl der Generationen (Iterationsschritte) definiert. Prinzipiell kann jedoch beispielsweise auch ein Konvergenzkriterium oder eine Schranke (Optimierungsziel) als Abbruchkriterium verwendet werden.

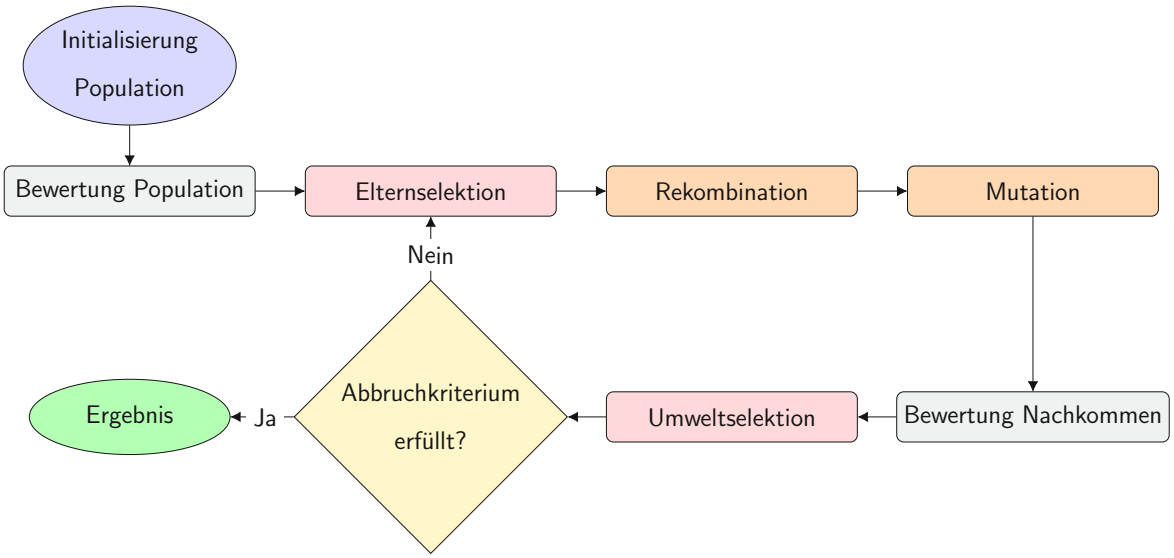


Abbildung 4.1: Schematischer Ablauf eines GA in Anlehnung an [22], [76]

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Algorithmus 1: Genetischer Algorithmus, angelehnt an [76]

Params: $t_{max} \in \mathbb{N}$

$p_m \leftarrow \in [0,1]$

$p_c \leftarrow \in [0,1]$

$\mu \leftarrow \in \mathbb{N}$

$\lambda \leftarrow \in \mathbb{N}$

`/* Mutationswahrscheinlichkeit` p_m`, Rekombinationswahrscheinlichkeit` p_c `,`
`Populationsgröße` μ`, Anzahl Nachkommen` λ `*/`

Output: Bestes Individuum I

`/* Start Algorithmus */`

1 $t \leftarrow 0$ `/* Initialisiere Zähler für Generationen */`

2 $P(t=0) \leftarrow$ Initialisiere Startpopulation

3 $P.F \leftarrow$ Berechne Fitness für P

4 **while** $t \leq t_{max}$ **do**

5 $E \leftarrow$ Selektiere Eltern aus $P(t)$ `/* Elternselektion */`

6 $N \leftarrow$ erzeuge λ Nachkommen aus E mit p_c `/* Rekombination */`

7 $N' \leftarrow$ Mutiere Individuen in N mit p_m `/* Mutation */`

8 $N'.F \leftarrow$ Berechne Fitness für N' `/* Bewertung */`

9 $P(t+1) \leftarrow$ Selektiere μ Individuen aus $P(t)$ und N' `/* Umweltselektion */`

10 $t \leftarrow t+1$

11 **end**

12 **return** Bestes Individuum in P(t)

Verschiedenste Arten von GAs bzw. EAs verfolgen unterschiedliche Selektionsstrategien bzw. erzeugen auf unterschiedliche Arten Selektionsdruck. In speziellen GAs gilt so beispielsweise $\mu = \lambda$ und die Nachkommen ersetzen in der Umweltselektion die Elternpopulation vollkommen. An dieser Stelle wird jedoch nicht weiter auf die Eigenheiten spezieller Ausprägungen von GAs eingegangen, sondern auf die folgenden Abschnitte und Kapitel 5 verwiesen. Die einzelnen Bestandteile eines GA werden in 4.4.2 im Detail erläutert.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

4.4.1 Grundbegriffe und Definitionen

Zur Beschreibung und Diskussion eines GA bedarf es der Definition einiger Grundbegriffe. Einige dieser Definitionen sind lediglich im Kontext evolutionärer Algorithmen gültig sind und weichen vom allgemeinen Sprachgebrauch in anderen wissenschaftlichen Disziplinen ab.
Die Repräsentation einer Lösung im Lösungsraum eines vorgegebenen Optimierungsproblems kann vielfältige Strukturen annehmen, die für das jeweilige Problem geeignet sind. Die ursprüngliche und elementarste Form zur Beschreibung eines Optimierungsproblems für einen GA ist die Kodierung einer Lösung des Problems als binäre Zeichenkette.

Phänotyp ist die reale, natürliche Struktur der problemspezifischen Beschreibung einer Lösung [22], [30].

Genotyp ist die Bezeichnung für die Kodierung des Problems in einer nicht notwendigerweise problemspezifischen Struktur, die für die Manipulation durch einen GA verwendet wird.

Genotyp und Phänotyp sind durch eine Dekodierungsfunktion miteinander verknüpft [22], wobei eine Kodierung des Problems nicht immer notwendig und sinnvoll ist [22], [25]. Welche Vorteile sich aus einer solchen Kodierung ergeben können, wird im Nachgang erläutert.

Individuum wird eine vollständige Repräsentation eines Lösungskandidaten für ein vorliegendes Optimierungsproblem genannt. Es verfügt über die eigentliche Kodierung der Lösung (Genotyp), den Gütewert (Fitness) dieser Lösung und alle notwendigen Zusatzinformationen [22].

Chromosom ist die genotypische Kodierung eines Individuums.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Gene sind die elementaren Bausteine, aus denen sich ein Chromosom zusammensetzt, also die einzelnen Abschnitte einer vollständigen Lösung [77].

Allele sind die zulässigen Ausprägungen, die ein Gen annehmen kann [77].

Population ist eine Menge an Individuen in einer Generation.

Fitness wird die Güte eines Lösungskandidaten hinsichtlich einer vorgegebenen Zielfunktion genannt. Die Fitness wird üblicherweise mithilfe des Phänotyps einer Lösung bestimmt.

Generationen sind die Iterationsschritte, die ein GA durchläuft.

Diversität beschreibt die Vielfalt in einer Population. Eine Population ist divers, wenn sie einen möglichst großen Bereich des Suchraums mit ihren einzelnen Individuen abdeckt. Generell ist eine hohe Diversität vorwiegend in frühen Stadien eines GA erstrebenswert, um eine vorzeitige Konvergenz in einem lokalen Optimum zu verhindern. Mathematisch kann die Diversität als Abstandsmaß betrachtet werden, wobei unter anderem Metriken wie der Hamming-Abstand oder die Shannon-Entropie zum Einsatz kommen können [22]. Geeignete Metriken zur Beurteilung der Diversität einer Population können sich von Problem zu Problem allerdings stark unterscheiden.

Zum einfacheren Verständnis der erläuterten Begriffe dient Abbildung 4.2.

Diese Begriffe sollen anhand des TSP (Grundlagen dazu siehe Abschnitt 3.1.5) im nachfolgenden Beispiel verdeutlicht werden. Im Fall des TSP könnte die Lösungsrepräsentation im phänotypischen Raum in Form einer Sequenz von verschiedenen Städten ohne weitere Kodierung direkt als Genotyp verwendet werden. Alternativ ist allerdings auch eine Standardbinärkodierung des Problems denkbar, die, ungeachtet ihrer Sinnhaftigkeit, wie folgt aussehen könnte.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Das TSP wird für die acht Städte $=\{0,1,2,3,4,5,6,7\}$ betrachtet. Eine vollständige Lösung kann binär kodiert wie folgt dargestellt werden:

Genotyp: 011 - 101 - 111 - 000 - 001 - 100 - 110 - 010

Die zugehörige Lösung im Suchraum sieht folgendermaßen aus:

Phänotyp: 3 - 5 - 7 - 0 - 1 - 4 - 6 - 2

Die obige Darstellung des Genotyps wird auch als **Chromosom** bezeichnet. Eine Einheit des Chromosoms, bestehend aus drei Bits, ist ein **Gen**.

Das erste Gen ist mit dem Allel, 011 besetzt. Die **Fitness** des Lösungskandidaten erhält man, indem am dargestellten Phänotyp des Lösungskandidaten eine Zielfunktion ausgewertet wird. (Im vorliegenden Fall ist das die Berechnung der zurückgelegten Strecke)

Wenn man diese Fitness einem Genotyp und seinen Zusatzinformationen zuordnet, erhält man ein **Individuum**.

Eine **Population** besteht aus mehreren Individuen und könnte z.B. Individuen mit den folgende Phänotypen enthalten:

3 - 5 - 7 - 0 - 1 - 4 - 6 - 2

2 - 7 - 3 - 0 - 5 - 1 - 4 - 6

1 - 7 - 3 - 5 - 6 - 2 - 0 - 4

...

Kodierung

Die Kodierung des der einzelnen Gene eines Chromosoms ist in der Regel abhängig von der Struktur des zugrundeliegenden Problems. Prinzipiell ist eine Kodierung des Problems in mehr oder weniger jeder Form denkbar, die ein Computer abbilden kann. Die nachfolgende Diskussion verschiedener Möglichkeiten zur Kodierung orientieren sich an [22], [25], [77].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

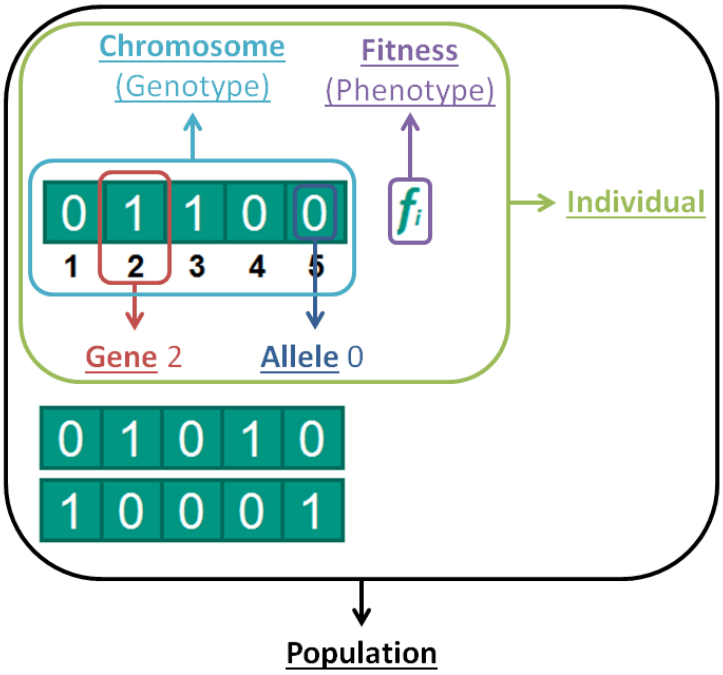


Abbildung 4.2: Veranschaulichung von Population, Chromosom und Genen [78]

Binäre Kodierung

Im einfachsten und ursprünglichen Fall erfolgt die Kodierung als binäre Repräsentation. Ein Chromosom kann dementsprechend nur aus den Werten 0 oder 1 bestehen und beispielsweise wie folgt aussehen.

Chromosom binär	011100010101001

Diese Darstellung ermöglicht die direkte Anwendung vieler gängiger Standardoperatoren (siehe Abschnitt 4.4.2). Die Größe des Chromosoms ist abhängig von der Komplexität des Problems und wächst insbesondere bei komplexeren Problemen schnell an. Davon abgesehen ist die binäre Darstellung eines Optimierungsproblems nur selten zweckmäßig [77]. Gängige Arten der binären Kodierung sind die standardbinäre- sowie die Gray-Kodierung [22].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Kodierung in anderen Basissystemen

Die Kodierung eines Chromosoms kann auch in anderen Zahlensystemen, wie z.B. dem oktalen (0-7) oder hexadezimalen System (0-9, A-F) erfolgen [77].

Chromosom oktal	10563751
Chromosom hexadezimal	1A7E3B

Permutationskodierung

Die Permutationskodierung bzw. die ganzzahlige Kodierung ist insbesondere für ganzzahlige Optimierungsprobleme, wie Probleme der Reihenfolgenbildung oder generell kombinatorische Optimierungsprobleme geeignet. Das Chromosom besteht, wie der Name bereits vermuten lässt, aus einer Sequenz von ganzen Zahlen.

Chromosom ganzzahlig	1 9 5 8 7 6

Diese Art der Kodierung ist beispielsweise auch für das TSP zweckmäßig. Allerdings gilt zu beachten, dass nach der Manipulation von Chromosomen mittels Standardoperatoren i.A. Anpassungen (Reparaturen) vorgenommen werden müssen, um die Randbedingungen des Problems einzuhalten.

Wertkodierung

Ist eine Repräsentation, die direkt das zugrundeliegende Problem beschreibt und insbesondere für komplexere Probleme geeignet, die mit den vorigen Darstellungsarten nicht sinnvoll abgebildet werden können. In diesem Fall entspricht die genotypische- der phänotypischen Darstellung eines Lösungskandidaten. Die einzelnen Bestandteile des Chromosoms können z.B. reelle Zahlen aber auch die Beschreibung komplexerer Objekte bzw. Anweisungen, oder alphanumerische Zeichenketten beinhalten [77].

Chromosom reelle Zahlen	2,756 5,678 89,883 7,687
Chromosom alphanumerisch	ABHJD HHSKD XHSFD ZXBLM

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Diese Art der Kodierung ist gut geeignet, um eine problemspezifische Beschreibung einer Lösung zu erreichen, allerdings geht damit i.A. auch die Notwendigkeit problemspezifischer Operatoren einher.

4.4.2 Bestandteile und Operatoren eines Genetischen Algorithmus

In diesem Abschnitt werden die einzelnen, in Abbildung 4.1 dargestellten Schritte und die damit verbundenen Operatoren eines GA in chronologischer Reihenfolge beschrieben. Um einen GA vollständig beschreiben zu können, bedarf es vorab einer Beschreibung seiner Charakteristika mithilfe der dafür notwendigen Parameter.

Parameter

Die wesentlichen Stellgrößen zur Beeinflussung des globalen Verhaltens eines GA angelehnt an Eiben et al. [79] sind in Tabelle 4.2 abgebildet, wobei insbesondere κ lediglich notwendig ist, falls die häufig eingesetzte Turnierselektion (siehe Abschnitt 4.4.2.1) zum Einsatz kommt.

μ	Populationsgröße	Anzahl der Individuen je Population
λ	Nachkommensgröße	Anzahl der erzeugten Nachkommen (Kindindividuen) in jeder Generation
p_m	Mutationsrate	Wahrscheinlichkeit für die Mutation eines Gens in einem Chromosom
p_c	Rekombinationsrate	Wahrscheinlichkeit für die Rekombination Anzahl der aus Rekombination erzeugten Nachkommen
κ	Turniergröße	Größe des Turniers bei Turnierselektion (siehe 4.4.2.1)
σ	Mutations-Schrittweite	Anzahl der Veränderungen bei einer Mutation (nur in speziellen Mutationsoperatoren)

Tabelle 4.2: Parameter eines GA

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Gängige Verfahren zur Parameteranpassung, -steuerung und -optimierung sind in [25] und in [79] beschrieben. Hierbei wird unterschieden zwischen dem Parameter-Tuning, also der Bestimmung optimaler, konstanter Parameter vor dem tatsächlichen Ablauf des GA und der Parameter-Steuerung, bei der die Parameter des GA während seines Ablaufs fortlaufend angepasst werden [79]. Das Parameter-Tuning kann als eigenständiges Optimierungsproblem betrachtet und dementsprechend mit gängigen Optimierungsverfahren (siehe 3.2) behandelt werden. Dazu kommen beispielsweise auch Meta-GAs zum Einsatz, die die Parameterkombination des eigentlichen GA als Optimierungsproblem behandeln [25], [79]. Zu den Strategien zur Parametersteuerung (Anpassung der Parameter während des Ablaufs des GA) zählen u. a. die „zeitabhängige“deterministische Anpassung der Mutationsrate σ (abhängig von der Anzahl der Generationen), sowie die adaptive Parameteranpassung nach Rechenberg [80] und selbst-anpassende Verfahren [25].

Initialisierung

Die Initialisierung eines GA erfolgt, in der Regel durch die Erzeugung von Individuen für eine Startpopulation mithilfe von Zufallsverfahren, regelbasierten Abläufen oder anderen Optimierungsverfahren [22]. Bei der Initialisierung gilt zu beachten, dass die so erzeugte Startpopulation mit der Größe μ über ein Maximum an Diversität verfügen sollte, um den Suchraum der nachfolgenden Optimierung nicht zu beschränken und eine vorzeitige Konvergenz der Population zu vermeiden [22], [25], [77]. Die Startpopulation hat einen maßgeblichen Einfluss auf die Güte der schlussendlich bestimmten Lösung und auf die Konvergenzgeschwindigkeit des GA.

Bewertung

Die Bewertung der Fitness eines Individuums erfolgt am dekodierten Chromosom, dem Phänotyp, mithilfe der vorgegebenen Zielfunktion. Mathematisch betrachtet ist die Fitnessfunktion F eine Abbildung vom Suchraum in den reellen Zahlenraum ($F : \Omega \rightarrow \mathbb{R}$) und mit der Kenntnis des Optimierungsproblems (Minimierungsproblem vs. Maximie-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

rungsproblem) ergibt sich eine Ordnungsrelation zwischen einzelnen Fitnesswerten und damit auch zwischen einzelnen Individuen ($I_k \succ I_j$) [22]. Im einfachsten Fall ist die Fitness ein skalarer, numerischer Wert, der einen einfachen Vergleich von mehreren Individuen ermöglicht. Im Fall des TSP liefert die Bewertungsfunktion die zurückzulegende Strecke für eine definierte Route.

4.4.2.1 Selektion

Im Grunde ist das Ziel der Selektion die Auswahl von einem oder mehreren Individuen aus einer Population. In der Regel, sollen dabei die besten (fittesten) oder unterschiedlichsten (diversesten) Individuen gewählt werden, um eine Weiterentwicklung der Population zu ermöglichen. Hierbei stehen jedoch die Forderungen nach einer möglichst großen Fitness einer Population und der gleichzeitig zu wahrenden Diversität häufig im Widerspruch [22]. In jeder Generation des in Algorithmus 1 dargestellten GA werden zwei Selektionsoperationen angewandt, die Elternselektion und die Umweltselektion.
Das Ziel der **Elternselektion** ist die Auswahl von Eltern(-paaren) aus einer Population, um aus ihnen Nachkommen zu erzeugen. Im Gegensatz dazu soll bei der **Umweltselektion** aus der ursprünglichen Population $P(t)$ und deren Nachkommen N eine neue Population $P(t+1)$ für die darauffolgende Generation erzeugt werden, wobei auch hier die Diversität gewahrt und die besten Individuen gewählt werden sollen [22], [25]. Je nach Ausprägung des jeweiligen Algorithmus kommen einer oder beide Selektionsoperatoren zum Einsatz.
Ausgehend von einer Population P, bestehend aus Individuen I und deren zugeordnete Fitnesswerte $I.F$, sollen bei der **Elternselektion** Eltern(-paare) für die Fortpflanzung ausgewählt werden. Zu beachten gilt hier, dass bei den verwendeten Selektionsverfahren jedes Individuum die Chance haben sollte, gewählt zu werden, da ansonsten eine große Population nicht gerechtfertigt wäre, wenn ohnehin nur ein bestimmter Teil der Population gewählt werden kann [22]. Um dies zu gewährleisten, können entweder alle Individuen ungeachtet ihrer Fitness gleich behandelt und in mindestens ein Kindindividuum eingehen, oder fitnessproportional stochastisch aus der Population gewählt werden. Fitnessproportional heißt in diesem Kontext, dass ein proportionaler Zusam-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

menhang zwischen Auswahlwahrscheinlichkeit und Fitness eines Individuums besteht. Ein gängiger Vertreter von probabilistischen, fitnessproportionalen Verfahren sind die fitnessproportionale Selektion (Roulette Wheel Selection) oder das stochastisch universelle Sampling [22], [77]. Als probabilistisches Verfahren, das keine fitnessproportionalen Auswahlwahrscheinlichkeiten verwendet, wäre eine reine Zufallsauswahl denkbar. Im Kontrast zu den probabilistischen Verfahren steht die rangbasierte Selektion, bei der die Individuen nach ihrer Fitness gereiht werden. Nachfolgend werden einige der genannten Selektionsverfahren beschrieben.

Turnierselektion

Bei der Turnierselektion mit Turniergröße q wird aus der bestehenden Population eine Menge an q Individuen ausgewählt, die gegeneinander antreten. Das fitteste Individuum gewinnt das Turnier und wird in die nächste Population eingefügt. Dieser Vorgang wird so lange wiederholt, bis die gewünschte Anzahl an Nachkommen ausgewählt wurde [76].

Fitnessproportionale Selektion

Die Roulette Wheel Selection (*deutsch*: Fitnessproportionale Selektion, RWS) hat ihren Namen, da man sich die Summe aller einzelnen Fitnesswerte der Individuen als Fläche eines Rouletterades vorstellen kann, wobei dessen Teilflächen in ihrer Größe proportional den jeweiligen Fitnesswerten der Individuen sind. In Abbildung 4.3 ist das Verfahren schematisch dargestellt. Die einzelnen Fitnesswerte der Individuen in einer Population werden aneinandergereiht. Anschließend eine Zufallszahl im Intervall $[0, \sum_i I_i.F)$ gewählt. Das Individuum, in dessen Intervall die gewählte Zufallszahl liegt, wird selektiert. Dieser Vorgang wird so lange wiederholt, bis die gewünschte Größe λ der neuen Population erreicht ist.

Unter diesem Verfahren kann die Vielfalt der neuen Population, insbesondere, wenn die Fitnesswerte stark variieren, leiden, da es leicht möglich ist, dass viele Individuen gar nicht und einige wenige mehrmals gewählt werden. Andererseits kann auch genau das

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

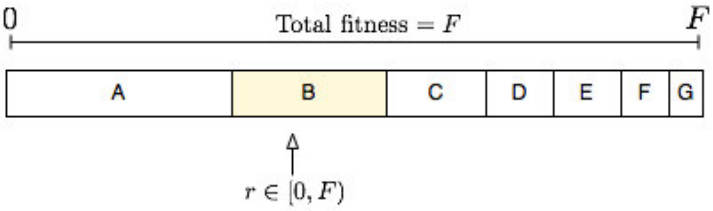


Abbildung 4.3: Fitnessproportionale Selektion [81]

Gegenteil eintreten, indem das fitteste Individuum der Population gar nicht gewählt wird, obwohl erwartet wird, dass es mehrfach gewählt werden würde.

Stochastisch Universelles Sampling

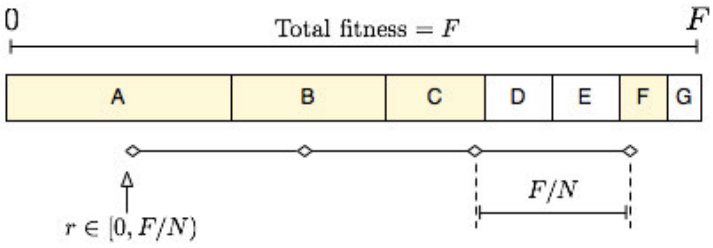


Abbildung 4.4: Stochastisch Universelles Sampling [82]

Das Stochastisch Universelles Sampling (*engl*: Stochastic Universal Sampling, SUS) löst die Probleme der zuvor beschriebenen RWS. Im Gegensatz zum vorigen, mehrfachen Wählen einer Zufallszahl, wird in diesem Verfahren nur ein einziges Mal eine Zufallszahl gewählt. Bei einer Anzahl an zu selektierenden Individuen λ und der Summe der gesamten Fitness in der Population $F = \sum_i I_i.F$ wird diese Zahl aus dem Wertebereich $[0, \frac{F}{\lambda}]$ gewählt. Anschließend wird der Wert $\frac{F}{\lambda}$ mit Häufigkeit $(\lambda - 1)$-mal zu dieser Zufallszahl hinzugezählt, sodass man λ äquidistante Zeiger über den gesamten Wertebereich erhält (siehe Abbildung 4.4). Das entspräche λ äquidistant verteilten Kugeln auf einem Rouletterad. Die Auswahl der Individuen erfolgt anschließend für jeden Zeiger, gleich wie bei der RWS. Die SUS mit $\lambda = 1$ ist äquivalent zur RWS. [76] Sowohl beim SUS, als auch bei der RWS gilt zu beachten, dass die fitnessproportionalen Bereiche dem Optimierungsproblem entsprechend gestaltet werden müssen - bei einem Minimierungsproblem müssen die Inversen der Zielfunktionswerte verwendet werden, um

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

passende Bereiche zu erhalten.

Elitismus

Da sich durch die Operationen Selektion und Rekombination die neu erzeugten Individuen bezüglich ihrer Fitness nicht zwangsläufig verbessern, wird der sogenannte Elitismus eingeführt, um zu vermeiden, dass Informationen über bereits gefundene, gute Lösungen wieder verloren gehen. Der Elitismus wird sichergestellt, indem entweder die Selektionsoperatoren garantiert elitär sind oder indem eines oder mehrerer Elite-Individuen eingeführt werden. Diese Elite-Individuen sind die besten der Elterngeneration und werden garantiert in die nachfolgende Generation übernommen, sofern die erzeugten Kindindividuen diese in ihrer Güte nicht übertreffen [76]. Grafisch bedeutet das, dass der zeitliche Verlauf der Fitness des besten Individuums streng monoton fallend ist – die beste Fitness in der Population kann lediglich stagnieren, jedoch nicht schlechter werden [83]. Damit ist die Konvergenz des GA gesichert, dies sagt jedoch nichts über die Güte oder die Geschwindigkeit der so gefundenen, asymptotischen Lösung aus. Ein GA wird elitär genannt, wenn er diese Eigenschaft besitzt.

4.4.2.2 Rekombination

Die Rekombination (*engl:* Crossover) lässt sich am einfachsten mit der direkten Übersetzung aus dem Englischen erklären – der Kreuzung. Ziel dieser Kreuzung ist die Weitergabe von Merkmalen zweier oder mehrerer Elternteile an ein Kindindividuum. Abhängig von der Diversität in der Population kann die Rekombination sowohl zur Exploration als auch zur Exploitation beitragen. Bei großer Diversität hilft der Rekombinationsoperator dabei unerforschte Gebiete zu finden, bei kleinerer Diversität trägt er zur Feinabstimmung in einem kleinen Gebiet bei [22]. Wie sich erahnen lässt, hat dieser Operator seinen Ursprung in der natürlichen Fortpflanzung und bildet diesen Prozess vereinfacht nach. Im Sinne der Abstandsmetrik, die für die vorliegenden Individuen definiert ist und ein Maß für die Diversität einer Population darstellt (siehe Absatz 4.4.1), sollte das erzeugte Kindindividuum zwischen seinen Eltern liegen, da es Merkmale bei-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

der Elternteile erbt [21]. Nachfolgend werden einige Operatoren zur Verdeutlichung der Methodik angeführt.

Ein- und Mehr-Punkt-Crossover

Die einfachste Form eines Rekombinationsoperators ist der **Ein-Punkt-Crossover** (siehe Abbildung 4.5). Dieser Operator „schneidet“ beide Eltern an derselben, zufällig gewählten, Stelle auseinander und setzt die so erhaltenen Teile zu einem neuen Individuum zusammen. Die Verallgemeinerung dieses Operators ist der **n-Punkt-Crossover**, der die Elternindividuen an n Punkten zerteilt und alternierend zusammensetzt [22], [30].

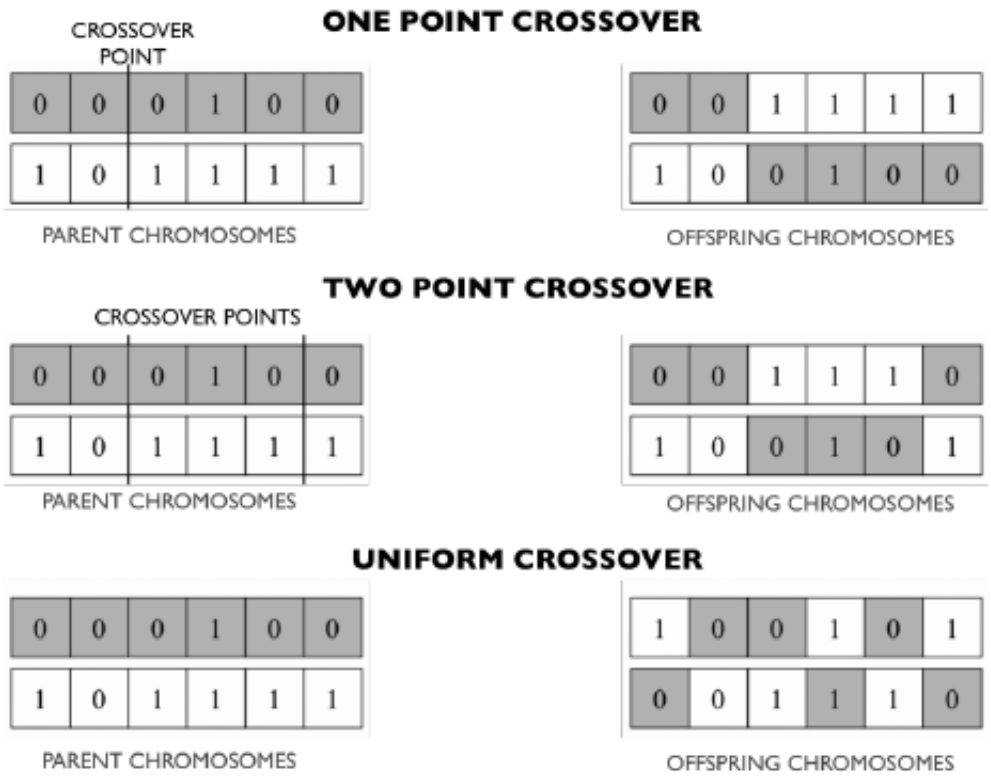


Abbildung 4.5: Rekombinationsverfahren [84]

Diese Operatoren sind zwar simpel, können jedoch auch nur auf bestimmte Genotypen angewandt werden. Im Beispiel des TSP, sowie bei vielen anderen kombinatorischen Optimierungsproblemen, darf ein Operator die grundlegende Zusammensetzung des Genpools nicht verändern. Jede Stadt muss in einer zulässigen Lösung für das TSP exakt einmal vorkommen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Rekombiniert man jedoch die Sequenzen **A – B – C – D** und **D – C – B – A** an einer beliebigen Stelle, kommt es zwangsläufig zu einer Verletzung dieser Forderung.

Uniformer Crossover

Dieser Operator wählt für jede Stelle des Kindindividuums zufällig den Eintrag eines Elternteils und setzt diesen Eintrag an derselben Stelle im Kindindividuum ein [30]. Dieser Operator bringt dieselben Probleme wie der zuvor erwähnte n-Punkt-Crossover mit sich.

Partially Mapped Crossover

Für Permutationsprobleme wie dem TSP ist ein Operator notwendig, der die Natur der Permutation berücksichtigt und keine Verletzungen verursacht. Der Partially Mapped Crossover (*deutsch*: Abbildungsrekombination, PMX) bewerkstelligt das, indem nur zulässige Reihenfolgen und Anordnungen von Genen in den Kindindividuen verwendet werden. Dazu wird ein Abschnitt (zwischen den roten Linien) des ersten Elternteils direkt in ein Kindindividuum kopiert, das die Einträge des zweiten Elternteils enthält. Somit ergibt sich eine fehlerbehaftete Struktur, die im abschließenden Schritt repariert wird [22], [30]. Dazu werden die Entsprechungen der beiden Eltern im kopierten Bereich bestimmt und im erzeugten Kind ausgetauscht. Der Ablauf des PMX ist in Abbildung 4.6 veranschaulicht. Dabei entspricht der Eintrag 6 in Parent1 dem Eintrag 5 in Parent2. Child1 ist eine Kopie von Parent1, in die der grüne Abschnitt aus Parent2 eingefügt wird. Die grünen Bereiche der Eltern überschneiden sich nur teilweise. Da in Child1 der Eintrag 4 nun doppelt vorkommt, muss er einmal mit einem fehlenden Eintrag 7 ersetzt werden. Dasselbe gilt für den Eintrag 5, dieser wird mit 6 ersetzt. Damit ist die Reparatur vollzogen und das neu erzeugte Individuum zulässig.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

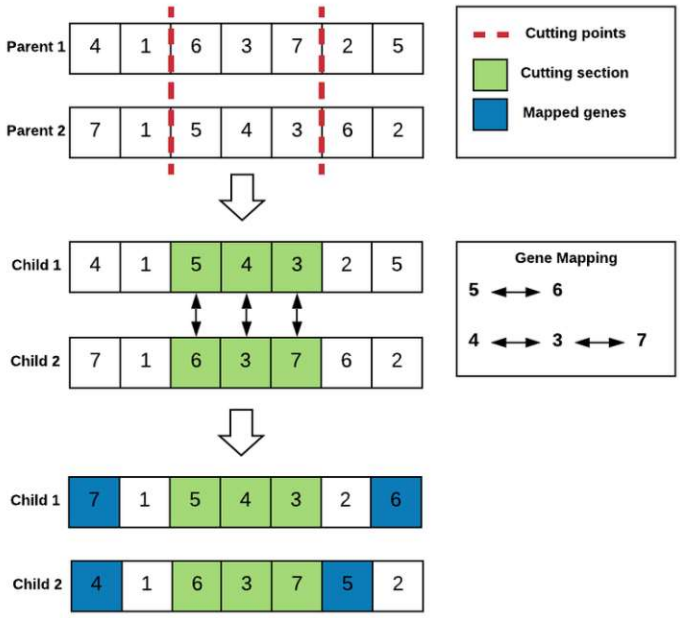


Abbildung 4.6: Abbildungsrekombination (PMX) [85]

Order Crossover

Ziel dieses Operators ist die Erhaltung der relativen Reihenfolge von Einträgen beim Crossover, was bei Problemen der Reihenfolgebildung von Vorteil sein kann. Hierbei muss zwischen zwei Varianten des Operators unterschieden werden, dem Operator von Davis [86] und dem Operator von Syswerda [87]. Der erstere hat mehr Ähnlichkeit mit dem zuvor beschriebenen PMX und wird hier beschrieben und mit Order Crossover (*deutsch*: Ordnungsrekombination, OX) bezeichnet, der letztere ist in seiner Funktion mit den gleichen Parametern äquivalent zum nachfolgend beschriebenen Position Crossover (*deutsch*: Positionsrekombination, PX) [76].

Der OX ist ebenfalls für Permutationen gedacht, und es wird gleich wie beim PMX ein ganzer Abschnitt aus dem ersten Elternteil in ein Kind kopiert. Hier ist das Kind jedoch zu Beginn leer. Danach wird begonnen, das Kind mit den verbleibenden Einträgen aus dem zweiten Elternindividuum aufzufüllen. Dabei wird nicht am Beginn der Individuen begonnen, sondern am Ende. Andernfalls wäre der Operator nicht „rein“ [88] (*engl:* pure). Ein *reiner* Rekombinationsoperator erzeugt bei der Kreuzung von zwei identischen Eltern ein ebenso identisches Kind. In Abbildung 4.7 ist das Vorgehen schematisch dar-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

gestellt.

```
Parent 1: A B . C D E F . G H I
Crossover-section: _ _ C D E F _ _ _

Parent 2: h d . a e i c . f b g
Available elements in order: b g h a i

Offspring: a i C D E F b g h.
```

Abbildung 4.7: Ordnungsrekombination nach [86], [76]

Position Crossover

Der PX wurde von Syswerda [87] entwickelt und erhält die relative Ordnung beider Eltern im erzeugten Kind. Es werden aus dem ersten Elternteil zufällig n Einträge ausgewählt und an dieselbe Stelle eines leeren Kindindividuums kopiert. Die restlichen freien Plätze werden entsprechend der Sortierung des zweiten Elternteils aufgefüllt. In Abbildung 4.8 werden die schwarzen, unterstrichenen Einträge aus dem Elternindividuum in das Kindindividuum kopiert und anschließend mit den roten Einträgen aus dem zweiten Elternindividuum aufgefüllt.

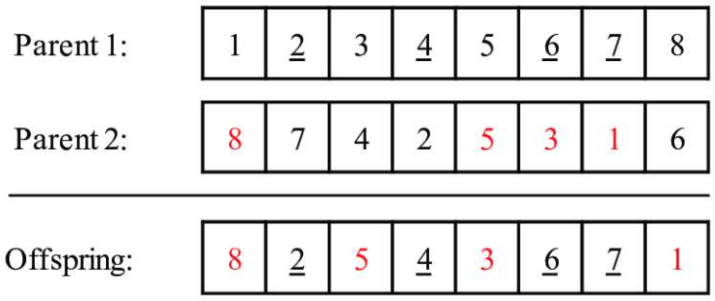


Abbildung 4.8: Positionsrekombination [89]

4.4.2.3 Mutation

Der Mutationsoperator ist der zweite genetische Operator, der Veränderungen an Individuen vornimmt. Er erzeugt ein neues Individuum, indem ein Elternindividuum gemäß

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

einer bestimmten Handlungsvorschrift verändert wird. Ebenso wie der Rekombinationsoperator beinhaltet auch der Mutationsoperator eine probabilistische Komponente. Auch die Mutation kann, abhängig von ihrer Schrittweite σ, sowohl zur Exploration als auch zur Exploitation beitragen, wobei große Schrittweiten die Exploration und kleinere Schrittweiten die Exploitation fördern [22]. Der eingesetzte Mutationsoperator hängt noch wesentlich stärker von der genotypischen Repräsentation der Individuen ab als der Rekombinationsoperator. Im Fall einer binären Zeichenkette ist die Mutation denkbar simpel, da die logische Negation eines einzelnen Bits ausreicht, um eine Veränderung des Individuums herbeizuführen und so das Genmaterial des Kindes zu verändern.
Bei allen anderen Problemen und Repräsentationen und insbesondere bei Permutationen müssen ausgefeiltere Operatoren verwendet werden, die die Korrektheit der erzeugten Nachkommen garantieren. Für reellwertige Repräsentationen sind Verfahren wie die Gauss-Mutation gängig. Bei ganzzahligen, alphanumerischen und komplexeren Darstellungen des Genotyps sind meist problemspezifische Operatoren notwendig.
Bei Permutationen greift das Schema Theorem nicht richtig, deshalb ist der Mutationsoperator bei diesen Problemen häufig wichtiger als der Rekombinationsoperator [22].

Bei der **Bit-Binären-Mutation** wird jedes Bit einer Zeichenkette mit einer vorgegebenen Mutationswahrscheinlichkeit p_m negiert (Bit-Flip – siehe Abbildung 4.9). Untersuchungen haben gezeigt, dass

1. eine zu Beginn große Mutationsrate, die exponentiell abnimmt, gute Ergebnisse liefert und eine
2. untere Schranke für die Mutationsrate mit $p_m = 1/l$, wobei l der Länge der Zeichenkette entspricht,

bekannt ist [76].

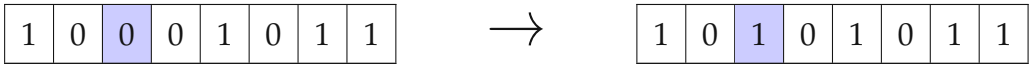


Abbildung 4.9: Binäre Mutation

Die nachfolgenden Mutationsoperatoren sind insbesondere für Permutationsprobleme

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

vorgesehen. Diese Operatoren garantieren die Erhaltung der Zulässigkeit der Permutation. Für alle Beispiele wird die Zeichenkette **A-B-C-D-E-F-G-H** zur Veranschaulichung verwendet, die im Falle des TSP die Menge aller zu besuchenden Städte beinhalten könnte.

Bei der **Vertauschenden Mutation** (*engl:* Swap Mutation) werden, wie der Name schon vermuten lässt, zwei zufällig gewählte Einträge des Individuums miteinander vertauscht (siehe Abbildung 4.10) [22], [76].

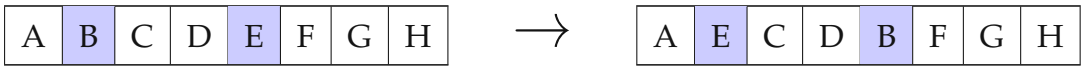


Abbildung 4.10: Vertauschende Mutation

Die **Verschiebende Mutation** (*engl:* Insert Mutation bzw. Position Based Mutation [76], [87])verschiebt einen zufällig gewählten Eintrag des Individuums an eine zufällig gewählte Stelle (siehe Abbildung 4.11) [22].

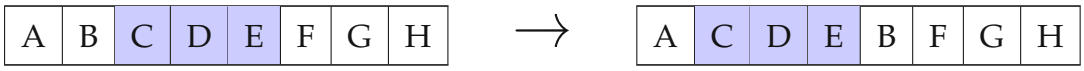


Abbildung 4.11: Verschiebende Mutation

Die **Mischende Mutation** (Scramble Mutation) verändert die Reihenfolge der Einträge eines zufällig gewählten Abschnitts, indem dieser zufällig umsortiert wird [22], [76], [86], [87].

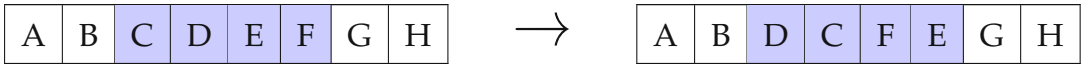


Abbildung 4.12: Mischende Mutation

Die **Invertierende Mutation** (*engl:* Inversion Mutation), sortiert ein zufällig gewähltes Teilstück des Individuums so um, dass die Reihenfolge der Einträge in diesem Abschnitt genau umgedreht (invertiert) ist (siehe Abbildung 4.13) [22].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

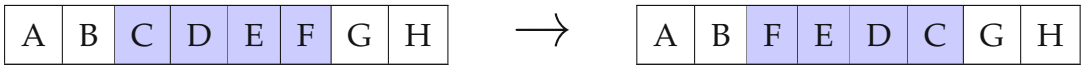


Abbildung 4.13: Invertierende Mutation

Die Stärke der verursachten Veränderung, die durch die Operatoren für Permutationen (Vertauschen, Verschieben, Invertieren, Mischen) hervorgerufen wird, ist einerseits von der Größe der gewählten Bereiche und andererseits vom zugrundeliegenden Problem abhängig.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Kapitel 5

Umsetzung und Implementierung

Das vorliegende Ressourcenzuteilungsproblem besteht im Grunde in der Zuweisung von Personen zu Positionen (Dienstposten) und Ausbildungsstellen für jede betrachtete Zeitperiode. Im folgenden Abschnitt wird dieses Problem zuerst verallgemeinert betrachtet und anschließend eine Vereinfachung eingeführt, die die Formulierung und Behandlung des Problems erheblich simplifiziert. Diese Vereinfachung und deren Auswirkungen werden an dieser Stelle kurz erläutert.

Modellannahme und Vereinfachung:
Aus dienstrechtlicher Sicht könnte theoretisch jede Person in Ausbildung für ein Stundenausmaß von weniger als 40 beschäftigt, also teilzeitbeschäftigt sein. Einschränkungen hinsichtlich des genauen Stundenausmaßes gibt es hierbei nur wenige, die Ausbildungsdauer einer Person in Teilzeit verlängert sich allerdings entsprechend dem Beschäftigungsausmaß [6]. Da Personen auch Dienstposten und Ausbildungsstellen teilweise besetzen und sie sich diese mit anderen Personen teilen könnten, ergäbe sich in der Theorie ein beträchtliches Potenzial zur optimalen Auslastung von Dienstposten und Ausbildungsstellen, indem die vorliegenden Teilkapazitäten der jeweiligen Ressourcen voll ausgelastet würden.
Um dieses Potenzial zur Kapazitätsoptimierung tatsächlich nutzen zu können, wäre eine wesentlich komplexere Problembetrachtung und -beschreibung notwendig, da

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

einzelne Dienstposten, Ausbildungsstellen und Personen nicht mehr als ganze Einheiten betrachtet werden können, sondern in finite Teile zerlegt und zugeteilt werden müssten. Außerdem müssten zwei oder mehr Personen dieselben Stellen zur selben Zeit belegen, um tatsächlich einen Nutzen daraus ziehen zu können. Das wiederum würde erfordern, dass diese zwei Personen zusammen nicht über mehr als eine Vollzeitanstellung verfügen, dieselbe Ausbildung machen und in etwa denselben Ausbildungsfortschritt aufweisen.
Da in der Realität jedoch nur ein sehr geringer Teil (< 5% laut Expertinnenschätzung[5]) der Turnusärzt*innen über eine Teilzeitanstellung verfügt, wird dieses Potenzial als gering im Vergleich zum damit verbundenen Implementierungs- und Optimierungsaufwand (Laufzeit) eingeschätzt. Diese Vereinfachung hat keinerlei Auswirkungen auf die Zulässigkeit der erzeugten Lösungen, da lediglich Optimierungspotenzial ungenutzt bleibt, das als vernachlässigbar eingeschätzt wird und keine Randbedingungen dadurch verletzt werden.
Deshalb werden nachfolgend alle zu verplanenden Personen wie Vollzeitkräfte, die ganze Dienstposten und ganze Ausbildungsstellen benötigen, behandelt und das Problem so erheblich vereinfacht.

Weitere Annahmen:

Des Weiteren wird angenommen, dass genehmigte Ausbildungsstellen eine „unendliche Lebensdauer“ aufweisen, also, dass deren Rezertifizierung (siehe Abschnitt 2.2) immer erfolgreich ist und sich damit die Quantität und Qualität der verfügbaren Ausbildungsstellen im Zeitverlauf nicht ändert. Dieselbe Annahme wird auch für die verwendeten Dienstposten getroffen. Da eine Streichung von Ausbildungsstellen bzw. jedoch nicht üblich ist und damit eine fortlaufende Verfügbarkeit der Stellen einhergeht, stellt diese Annahme in der Realität keine Einschränkung in der Anwendbarkeit des so entwickelten Verfahrens dar.

Außerdem werden fiktive Ausbildungsstellen eingeführt, die in der Basisausbildung

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

verwendet werden, um die folgende Problembeschreibung allgemeingültig zu machen (an sich wären in der Basisausbildung keine gesonderten Ausbildungsstellen zu den Dienstposten erforderlich).

Gemäß der getroffenen Vereinfachungen und Annahmen und der Problembeschreibung aus den Kapiteln 1 und 2 kann das vorliegende mathematische Problem definiert werden.

Zusätzlich werden für jede Abteilung und jedes Ausbildungsfach, das in dieser Abteilung absolviert werden kann, vonseiten der ÖÄK Obergrenzen für die maximal zulässige Ausbildungsdauer festgelegt. In dieser Arbeit wird davon ausgegangen, dass die zulässige Ausbildungsdauer der benötigten Ausbildungsdauer für ein Fach entspricht. Diese Einschränkung muss zum Zeitpunkt der Erstellung dieser Arbeit getroffen, werden, da die notwendige Datengrundlage zur Berücksichtigung dieser Randbedingung nicht verfügbar ist. Sie tut jedoch der Anwendbarkeit des entwickelten Verfahrens keinen Abbruch und eine nachträgliche Anpassung wird als einfach umsetzbar angesehen.

5.1 Einführung der Mathematischen Notation

Für die oben genannten Größen existieren die jeweils zugehörigen Mengen an zu verplanenden Zeiteinheiten, Positionen, Ausbildungsstellen und Personen (Residents). Die nachfolgend angeführten Zahlen in den einzelnen Mengen sind hierbei als eindeutige Identifikatoren der jeweiligen Entitäten zu verstehen, die Zahlen repräsentieren also wesentlich komplexere Informationsbestandteile des vorliegenden Problems.
Somit existieren:

$$P = \{0\} \cup \{1, ..., p\} \in \mathbb{Z} \text{ Dienstposten (Positionen),}$$
$$Q = \{0\} \cup \{1, ..., q\} \in \mathbb{Z} \text{ Ausbildungsstellen,}$$
$$R = \{1, ..., r\} \in \mathbb{Z} \text{ Residents und}$$
$$T = \{1, ..., t\} \in \mathbb{Z} \text{ Zeitperioden.}$$

Besonders hervorzuheben ist hierbei die Menge $\{0\}$ in der Gesamtmenge an Ausbil-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

dungsstellen sowie in der Menge der Dienstposten. Diese „0“ ist notwendig, um die Möglichkeit einer Zuweisung eines Dienstpostens ohne gleichzeitige Zuweisung einer Ausbildungsstelle zu ermöglichen. Das ist sowohl notwendig, um Stehmonate abbilden zu können, als auch für die Basisausbildung, da in diesem Ausbildungsabschnitt keine Ausbildungsstelle benötigt wird, um den notwendigen Fortschritt zu erzielen.
Wenn eine Person allerdings einen Dienstposten „0“ zugewiesen hat, so verfügt die Person über kein Anstellungsverhältnis in dieser Periode, ist beurlaubt oder in Krankenstand.
Die letzte Zeitperiode $t_{max} = max(T)$ (die „Länge“ des Problems) hängt von der notwendigen Mindestdauer ab, die benötigt wird, sodass alle zu verplanenden Personen ihre Ausbildung abschließen können.
Jede Position in P, sowie jede Ausbildungsstelle in Q gehört einer Abteilung d eines Krankenhauses h an.

$$D = \{1, ..., d\} \in \mathbb{Z} \text{ Abteilungen und}$$
$$H = \{1, ..., h\} \in \mathbb{Z} \text{ Krankenhäuser.}$$

Jede Abteilung d umfasst i Dienstposten p und k Ausbildungsstellen q, jedes Krankenhaus umfasst m Abteilungen d. Somit hat jeder Dienstposten bzw. jede Ausbildungsstelle eine eindeutige Zuordnung zu einer Abteilung und damit zu einem Krankenhaus. Dieser Zusammenhang wird durch die Notation p_h bzw. q_h für das zugehörige Krankenhaus und mit p_d bzw. q_d für die zugehörige Abteilung signalisiert. Somit gilt $(p_d, q_d) \in D$ und $(p_h, q_h) \in H$.
Außerdem gilt zu berücksichtigen, dass von einer Person in einem Monat üblicherweise nur Dienstposten mit Ausbildungsstellen innerhalb derselben Abteilung verwendet werden dürfen. Eine Ausnahme hiervon stellen sogenannte „abteilungsunabhängige Dienstposten“ dar. Diese Dienstposten werden i.d.R geschaffen, um Kapazitätsengpässe innerhalb eines Krankenhauses abzufedern und ein gewisses Maß an Flexibilität in der Personalzuordnung zu Abteilungen zu ermöglichen. Somit können diese, zumindest theoretisch, in der gesamten Ausbildungsstätte (Krankenhaus) verwendet werden

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

und stellen eine gesondert zu betrachtende Untermenge von P dar. Die Menge der verfügbaren Dienstposten in einer Abteilung d wird mit P_d und die entsprechende Menge der verfügbaren Ausbildungsstellen mit Q_d bezeichnet, wobei abteilungsunabhängige Dienstposten des entsprechenden Krankenhauses jeder Menge P_d hinzuzurechnen sind. Diese Dienstposten werden im Sinne der Unterscheidbarkeit einer fiktiven Abteilung $d = 0$ im zugehörigen Krankenhaus h zugeordnet.
Im Idealfall, wenn alle Dienstposten mit allen Ausbildungsstellen kombinierbar wären, wäre die Menge der so verfügbaren Kombinationen je Abteilung an Dienstposten und Ausbildungsstellen das kartesische Produkt $PQ_{d,ideal}$ dieser zwei Mengen (siehe Gleichung 5.1).

$$\begin{gathered} PQ_{d,ideal} = P_d \times Q_d = \{(p,q) \mid p \in P_d,\ q \in Q_d\} \\ \text{mit} \\ P_d \subseteq P \quad \text{und} \quad Q_d \subseteq Q \\ P_d := \{p \in P \mid (p_d = d) \vee (p_d < 1 \wedge p_h = d_h)\} \\ \text{und} \\ Q_d := \{q \in Q \mid (q_d = d) \vee (q < 1)\} \end{gathered} \tag{5.1}$$

Erschwerend hinzu kommt die Einschränkung hinsichtlich der zulässigen Kombinationen aus Dienstposten und Ausbildungsstellen innerhalb einer Abteilung. Ausbildungsstellen können üblicherweise nur für bestimmte Fächer in bestimmten Ausbildungsabschnitten verwendet werden und Dienstposten sind auf bestimmte Ausbildungen und deren Abschnitte beschränkt. Sofern sich diese erlaubten Fächer einer Ausbildungsstelle q mit den erlaubten Ausbildungen und Ausbildungsabschnitten eines Dienstpostens p überschneiden, so nennt man p kombinierbar mit q (siehe Gleichung 5.2). In der Realität beläuft sich die Menge der verfügbaren Kombinationen somit auf eine Untermenge von $PQ_{d,ideal}$, weshalb gilt $PQ_d \subseteq PQ_{d,ideal}$.

$$PQ_d = \{x \in PQ_{d,ideal} \mid x_p \text{ kombinierbar mit } x_q\} \tag{5.2}$$

Daraus ergibt sich für jede Person (abhängig von ihrem bisherigen Ausbildungsverlauf und -fortschritt und ihrer gewählten Ausbildung) eine bestimmte Teilmenge aus PQ_d für jede Abteilung d, die verwendet werden kann. Verallgemeinert für das gesamte Pla-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

nungsproblem kann die Menge aller zulässigen Kombinationen aus Dienstposten und Ausbildungsstellen als Vereinigung aller PQ_d für jede Abteilung geschrieben werden.

$$PQ = \bigcup_{d \in D} PQ_d \tag{5.3}$$

Mit den zuvor beschriebenen Mengen kann für jede Zeitperiode eine eindeutige Ressourcenzuordnung zwischen Person, Ausbildungsstelle und Dienstposten erfolgen. Diese Zuteilung erfolgt mit der Entscheidungsvariable x_{pqrt}.

$$x_{pqrt} = \begin{cases} 1, & \text{Zuordnung} \\ 0, & \text{keine Zuordnung} \end{cases} \tag{5.4}$$

Eine Zuordnung bedeutet, dass Resident r im Monat t Position p und Ausbildungsstelle q zugeordnet ist. Der gesamte Entscheidungsraum besteht demzufolge aus vier Dimensionen und bildet für jedes x_{pqrt} einen binären bzw. booleschen Wert ab. Der Raum ist definiert durch $P \times Q \times R \times T$. Dieses kartesische Produkt erzeugt eine Menge von 4-Tupeln aller möglichen „Koordinaten“ im Entscheidungsraum. Die Mächtigkeit des Raumes, also die Anzahl der einzelnen binären Entscheidungsvariablen ergibt sich somit zu $|P \times Q \times R \times T| = |P| \cdot |Q| \cdot |R| \cdot |T|$, wobei $|X|$ der Mächtigkeit der Menge X also ihrer „Größe“ entspricht. Eine binäre Zeichenkette, die diesen Entscheidungsraum abbilden kann, hätte also die zuvor genannte Länge.

Damit gilt $x_{pqrt} \in \mathbb{B}$ mit $\mathbb{B} = \{0, 1\}$. Der so aufgespannte Lösungsraum verfügt, den vorliegenden Randbedingungen (siehe Abschnitt 5.3) geschuldet, über einen verhältnismäßig kleinen nutzbaren Bereich, da der Großteil des Raumes unzulässige Bereiche abbildet. Eine simple Abschätzung der Größe des so aufgespannten Entscheidungsraums ist in Abbildung 5.1 dargestellt, wobei dieser Abschätzung die Annahme $|P| = |Q| = |R|$ zugrunde liegt. Das bedeutet, dass in jedem Monat für jede Person genau eine Ausbildungsstelle und ein Dienstposten verfügbar wäre. Damit ergibt sich eine erste, stark vereinfachte Abschätzung zu $|P \times Q \times R \times T| = |T| \cdot |R|^3$. Mit dieser Abschätzung soll lediglich verdeutlicht werden, wie schnell die Anzahl der möglichen Zustände und damit die Komplexität des Problems mit der Anzahl der zu verplanenden Personen und den dafür notwendigen Ressourcen anwachsen kann. Hierbei gilt anzumerken, dass der Großteil der so möglichen Kombinationen an Dienstposten und Ausbildungsstellen unzulässig sind. Zulässig wären alle Kombinationen nur dann, wenn das Planungsproblem

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

eine Abteilung in einem Krankenhaus umfassen würde und P und Q vollständig kombinierbar wären.

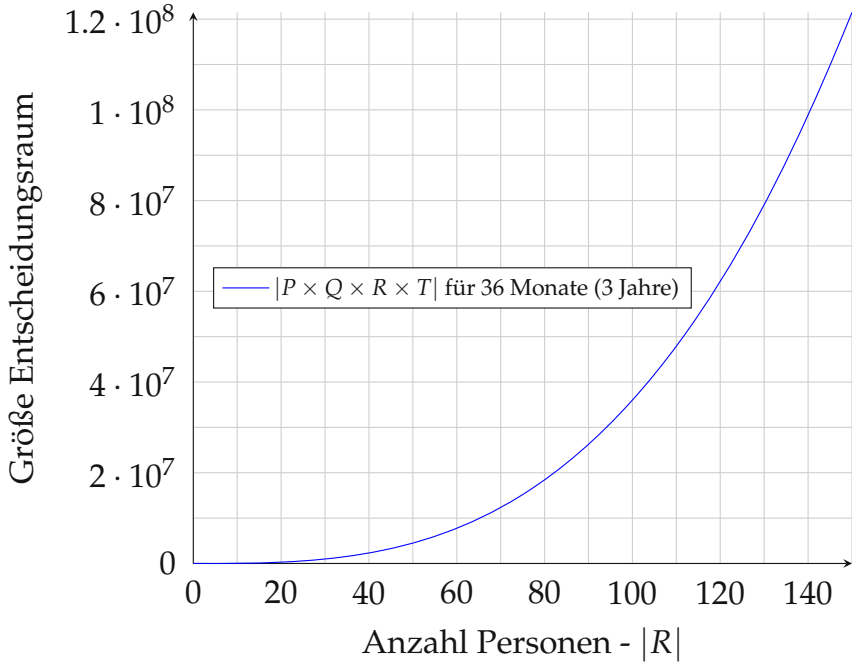


Abbildung 5.1: Schätzung der Anzahl der (binären) Entscheidungsvariablen in Abhängigkeit von der Anzahl der Personen (Annahme: Anzahl Personen = Anzahl Ausbildungsstellen = Anzahl Dienstposten)

Um den, für die Ressourcenzuordnung notwendigen, Raum dichter besetzt und somit effizienter zu definieren wird nachfolgend eine alternative Notation verwendet. Für diese vereinfachte Betrachtung wird die in der Einleitung dieses Kapitels (Kapitel 5) erläuterte Modellannahme herangezogen. Basierend auf dieser Vereinfachung wird in der vorliegenden Arbeit davon ausgegangen, dass sich alle zu verplanenden Personen in Vollzeitbeschäftigung befinden und alle Dienstposten, sowie Ausbildungsstellen Vollzeitstellen entsprechen. Somit kann jede Person nach wie vor in genau einer Abteilung arbeiten. Da sich zusätzlich alle Personen in Vollzeitbeschäftigung befinden, ist das Teilen von mehreren Dienstposten und Ausbildungsstellen weder notwendig noch sinnvoll. Damit kann jede Zuordnung einer Person zu einer Ausbildungsstelle und einem Dienstposten in einem Monat mit genau einem 2-Tupel ausgedrückt werden:

$$x_{r,t} = (p, q) \tag{5.5}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Ein solcher 2-Tupel entspricht in seiner Gestalt den 2-Tupeln in der Menge PQ (siehe Gleichung 5.3) und muss, um zulässig zu sein, in dieser Menge enthalten sein.

Einschub: Verallgemeinerte Betrachtung Wäre die Annahme von ausschließlich Vollzeitkräften und Vollzeitstellen nicht zulässig, so wäre die folgende Notation denkbar:

$$x_{r,t} = (\{p_1, ..., p_n\}, \{p_1, ..., q_m\}) \tag{5.6}$$

Hier stellt der erste Eintrag des Tupels alle zugeordneten Dienstposten in Form einer Menge und der zweite Eintrag die Menge aller zugeordneten Ausbildungsstellen dar (die Indizes m und n sind nicht notwendigerweise gleich groß). Alle bisher angeführten Formulierungen vernachlässigen die Tatsache, dass bei einer Zuteilung einer Person zu mehreren Stellen zusätzlich noch eine Verteilung der entsprechenden Kapazitäten erfolgen müsste. Mit der bisherigen Abbildung kann eine Person maximal zu gleichen Teilen auf eine Position zugeteilt werden. In der Realität könnte eine Person allerdings auch zu ungleichen Teilen auf mehrere Dienstposten bzw. Ausbildungsstellen verteilt werden. Um diesen Fall abbilden zu können, kann Notation 5.4 wie folgt verallgemeinert werden.

$$x_{pqrt} = \begin{cases} y, & \text{Kontinuierliche Zuordnung von } r \text{ zu } p \text{ und } q \text{ in } t \\ 0, & \text{Keine Zuordnung} \end{cases} \tag{5.7}$$

$$\text{mit } \{y \in \mathbb{R} : 0 < y \leq 1\}$$

Nachfolgend wird nur mehr das vereinfachte Problem mit ausschließlich einfachen Ressourcenzuteilungen, **ganzen Stellen** und **Vollzeitkräften** betrachtet und die dafür zulässige Notation aus den **Gleichungen** 5.4 und 5.5 herangezogen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Ausbildung und Ausbildungsfortschritt

Jede Person aus R verfügt über eine Menge an zu absolvierenden Ausbildungen C, diese Menge wird nachfolgend auch als Curriculum bezeichnet. Dieses persönliche Curriculum jeder Person kann, wie bereits in den Kapiteln 1 und 3 erläutert, aus mehreren Ausbildungen und untergeordneten Ausbildungsabschnitten bestehen, wobei jeder Ausbildungsabschnitt mindestens ein erforderliches Modul umfasst. Die gesamte ärztliche Ausbildung einer Person besteht zumindest aus zwei Ausbildungen, der Basisausbildung und der darauffolgenden Spezialisierung. Die Menge aller Ausbildungen wird mit $ED = \{1, ..., ed\} \in \mathbb{Z}$ bezeichnet. Jede Ausbildung wiederum setzt sich aus einem oder mehreren Ausbildungsabschnitten zusammen, wobei die Menge aller Ausbildungsabschnitte mit der $ES = \{1, ..., es\} \in \mathbb{Z}$ beschrieben wird. Hierbei besteht eine Ordnungsbeziehung zwischen den Ausbildungsabschnitten, sodass die Reihenfolge von Ausbildungsabschnitten in einer Ausbildung einer aufsteigend sortierten Liste der Abschnitte entspricht (Abschnitt 5 kommt zeitlich betrachtet nach Abschnitt 2). Das gilt analog für die einzelnen Ausbildungen.
Jeder Ausbildungsstelle q ist ein Ausbildungsabschnitt und mindestens ein Ausbildungsfach zugeordnet. Der Ausbildungsabschnitt einer Ausbildungsstelle wird mit q_{es} bezeichnet. Jedem Dienstposten p ist eine Menge an zulässigen Ausbildungen zugeordnet, diese Menge wird mit p_{ed} beschrieben und gibt vor, in welchen Ausbildungen eine Person diesen Posten besetzen darf. Da ein Ausbildungsabschnitt eindeutig einer Ausbildung zugeordnet werden kann, ergibt sich eine Einschränkung hinsichtlich der kombinierbaren Dienstposten und Ausbildungsstellen (siehe Gleichung 5.2).
Da die Reihenfolge der Module bzw. Fächer innerhalb eines Ausbildungsabschnittes zwar unwesentlich ist, die Reihenfolge der Ausbildungsabschnitte jedoch eingehalten werden muss (Abschnitt 2 darf nicht begonnen werden, wenn noch nicht alle Fächer aus Abschnitt 1 abgeschlossen sind), kann das Curriculum C einer Person R als geordnete Liste von Listen mit Länge ≥ 1 bestehend aus nicht leeren Mengen (Abschnitten) modelliert werden. Ein Abschnitt besteht wiederum aus einem oder mehreren Fächern, die teils Pflicht- und teils Wahlfächer sein können. Ein Modul verfügt ebenfalls über eine vorgeschriebene Dauer, die zu absolvieren ist, um das Modul abschließen zu können.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

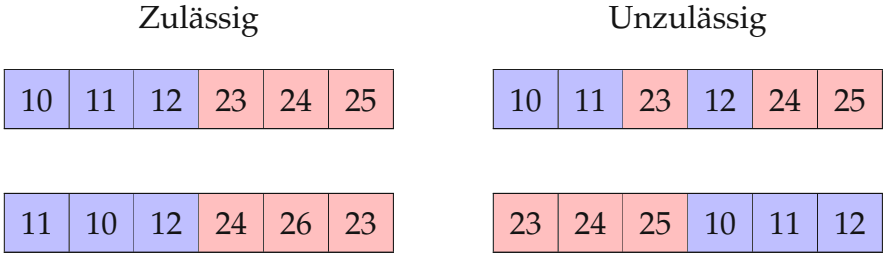
Die Gesamtdauer der Ausbildung einer Person ergibt sich somit aus der Summe aller erforderlichen Ausbildungszeiten je Modul und wird mit $C.DUR$ bezeichnet. Jede Person verfügt außerdem über eine Menge an präferierten Modulen, die sie im Rahmen der Wahlfächer belegen will, wobei die Anzahl der Präferenzen die Anzahl der Wahlfächer in der Ausbildung einer Person nicht übersteigen darf. Zur Verdeutlichung soll folgendes Beispiel und dessen Visualisierung dienen.

Beispiel Modulreihenfolge und Abschnitte:

Eine Person R verfügt über ein Curriculum $R.C$, das aus zwei Ausbildungen besteht. Die erste Ausbildung $ed = 1$ setzt sich aus zwei Ausbildungsabschnitten zusammen. Der erste Abschnitt $es = 1$ besteht aus den Modulen 10 und 11 und 12 und der zweite $es = 2$ aus den Modulen 23, 24, 25 und 26, wobei die Module 25 und 26 Wahlfächer sind, von denen lediglich eines absolviert werden muss. Das Curriculum stellt sich wie folgt dar:

$$\Big[\big[\{10, 11, 12\}, \{23, 24, \{25||26\}\}\big], [...]\Big]$$

Die Person R präferiert das Modul 25, hat also die „Präferenzmenge“ $R.MP = 25$. Nachfolgend sind zulässige und nicht zulässige Reihenfolgen der Module und Abschnitte veranschaulicht. In der ersten Variante der zulässigen Reihenfolgen wird die Präferenz der Person berücksichtigt, in der zweiten nicht.



Der Ausbildungsfortschritt einer Person zum Zeitpunkt t und der aktuelle Ausbildungsabschnitt, in dem sich eine Person zum Zeitpunkt t befindet, hängt von den vergangenen Zuweisungen ab. Damit ergibt sich ein dynamischer Zusammenhang von Zeit und Ausbildungsfortschritt einer Person. Der Ausbildungsabschnitt, in dem sich eine Person

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

zum Zeitpunkt t befindet, wird mit $r_{es,t}$ bezeichnet, wobei dieser Zustand abhängig von den vorigen Zuweisungen im Zeitverlauf ist. Analog wird die aktuelle Ausbildung, in der sich eine Person befindet, mit $r_{ed,t}$ bezeichnet.
Der Zeitpunkt, zu dem eine Person ihre Ausbildung abschließt, wird mit $t_{r,fin}$ bezeichnet und ist abhängig von den geplanten Zuweisungen einer Person.
Ein Ausbildungsabschnitt ist Bestandteil einer Ausbildung, diese Relation wird vereinfacht mit $es \in ed$ beschrieben. Da ein eindeutiger Zusammenhang zwischen Ausbildungsabschnitt und Ausbildung besteht, kann in weiterer Folge die Zugehörigkeit einer Ausbildungsstelle zu einer Ausbildung bestimmt werden. Dieser Zusammenhang wird für eine Ausbildungsstelle mit q_{ed} geschrieben. Zur Übersicht ist in Anhang A.5 eine Visualisierung des Zusammenhangs in Form einer Datenstruktur abgebildet.

5.2 Entwurf der Zielfunktion

Zur Bestimmung der Güte einer erzeugten Lösung müssen die, in Abschnitt 2.3 identifizierten, Zielgrößen miteinander in Einklang gebracht und gewichtet werden. Eine Abbildung des Zuweisungsproblems kann man sich als Matrix vorstellen, deren Zeilen die Einträge einer Person abbilden und deren Spalten die betrachteten Zeitperioden darstellen. Diese Matrix kann als Kalender verstanden werden, in dessen Felder die Zuweisungen aus Dienstposten und Ausbildungsstellen eingetragen sind (siehe Tabelle 5.6). Die

Periode / Person	T_1	T_2	...	T_n
R_1	(p_i, q_j)	(p_i, q_j)		(p_i, q_j)
R_2	(p_i, q_j)	(p_i, q_j)		(p_i, q_j)
...				
R_m	(p_i, q_j)	(p_i, q_j)		(p_i, q_j)

Tabelle 5.6: Zeitliche Zuweisungen von Personen zu Dienstposten und Ausbildungsstellen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Größe S eines behandelten Problems kann so in guter Näherung mit der Anzahl der zu verplanenden Personen und der Anzahl der zu verplanenden Monate abgeschätzt werden. S kann also als die maximal mögliche Anzahl der Zuweisungen im vorliegenden Problem verstanden werden.

$$S = |R \times T| = |R| \cdot |T| \tag{5.8}$$

Diese signifikante Größe eines Problems wird in weiterer Folge zur Normalisierung der einzelnen Zielfunktionswerte verwendet, um Probleminstanzen, die sich in ihrer Größe S maßgeblich unterscheiden, vergleichen zu können und so eine einheitliche Formulierung der Zielfunktion zu ermöglichen.
Für die Normalisierung der berücksichtigten Präferenzen ist eine weitere signifikante Größe notwendig. Dazu wird die maximale Anzahl der Modulpräferenzen, die berücksichtigt werden können, SMP herangezogen. Diese ist die Summe aller Präferenzen, die für jede einzelne Person zum Planungszeitpunkt vorliegen und noch eingehalten werden können.

$$SMP = \sum_{r \in R} |r.MP| \tag{5.9}$$

Die relevanten Metriken aus Tabelle 2.2 werden in Tabelle 5.8 aufgeführt und mit ihren zugehörigen Zielfunktionen versehen. Das vorliegende MOP wird als skalares Problem formuliert, indem alle zu optimierenden Zielfunktionen als gewichtete Summe in eine übergeordnete Zielfunktion übergeführt werden. Diese Methode nennt sich **Weighted Sum Method** (*deutsch:* gewichtete Summenmethode) und ermöglicht die Behandlung von MOPs als SOPs [90].

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

#	Name der Metrik	Zielfunktion	Gewicht	Gleichung
1	Stehmonate	$f_1(\vec{x})$	w_1	5.11
2	Konsekutive Stehmonate	$f_2(\vec{x})$	w_2	5.12
3	Ortswechsel - Krankenhaus	$f_3(\vec{x})$	w_3	5.13
4	Ortswechsel - Abteilung	$f_4(\vec{x})$	w_4	5.14
5	Einmonatige Zuweisungen	$f_5(\vec{x})$	w_5	5.15
6	Monate Extern	$f_6(\vec{x})$	w_6	5.16
7	Präferenzen	$f_7(\vec{x})$	w_7	5.17

Tabelle 5.8: Zuordnung von Metriken zu einzelnen Zielfunktionen und Gewichten

Die globale, zu optimierende Zielfunktion $F(\vec{x})$ ergibt sich somit als Linearkombination der einzelnen Zielfunktionen aus Tabelle 5.8.

$$G(\vec{x}) = \sum_{i}^{7} w_i \times f_i(\vec{x}) \tag{5.10}$$

In den nachfolgenden Funktionen wird vorausgesetzt, dass ein erzeugter Plan, der mit $\vec{x}$ repräsentiert wird, keine überschüssigen Zuweisungen enthält und nur über zulässige Einträge verfügt.
Die verwendeten Formelzeichen zur Übersicht:

- $\vec{z}_{rp}$ ... ein Vektor mit den Dienstposten einer Person r je Periode
- $\vec{z}_{rq}$ ... ein Vektor mit den Ausbildungsstellen einer Person r je Periode
- $\vec{z}_{rd}$ ... ein Vektor mit den Abteilungen, in denen die Person r in den jeweiligen Zeitperioden befindet
- $\vec{z}_{rh}$ ... ein Vektor mit den Krankenhäusern, in denen sich Person r in den jeweiligen Zeitperioden befindet
- H_{ext} ... die Menge der Krankenhäuser, die als Kooperationspartner geführt werden
- $\vec{y}_{rms}$ ... Ein Vektor, der die Reihenfolge abbildet, in der die Module im Curriculum einer Person r begonnen werden

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

- $\vec{y}_{rmf}$ ... Ein Vektor, der die Reihenfolge abbildet, in der die Module im Curriculum einer Person r abgeschlossen werden
- y_{rmf} ... Die Menge der absolvierten Module aus $\vec{y}_{rmf}$
- $\vec{e}$ ... Der Einheitsvektor
- $\vec{c}_r$ ... Ein Vektor für Person r, der die Längen der zusammenhängenden Stehmonate beinhaltet

Die Vektoren $\vec{z}_{rp}$ und $\vec{z}_{rq}$ bilden gemeinsam die Zeilen der Matrix in Tabelle 5.6 ab und sind damit die Einträge der Stellen in $\vec{x}$, die ungleich 0 sind. Alle beschriebenen Vektoren $\vec{z}_{ri}$ haben die Länge $|T|$ und können direkt aus $\vec{x}$ abgeleitet werden. Die Notation $\vec{z}_{rit}$ stellt den skalaren Eintrag des Vektors $\vec{z}_{ri}$ an der Stelle t dar.
Die Länge der Vektoren $\vec{y}_{rmi}$ hängt vom Curriculum der jeweiligen Person r ab und muss sich mit der Anzahl der noch zu absolvierenden Module in diesem Curriculum decken. Der Zusammenhang zwischen dem Vektor $\vec{c}_r$ und den Stehmonaten einer Person wird im folgenden Beispiel verdeutlicht. Der Vektor $\vec{d}_r = \begin{bmatrix}1, & 0, & 0, & 1, & 1, & 0, & 1, & 1, & 1, & 0\end{bmatrix}^{\mathbf{T}}$ bildet die Stehmonate einer Person ab, wobei der Wert 1 für einen Stehmonat steht. Der zugehörige Vektor $\vec{c}_r$ hat die Form $\begin{bmatrix}1, & 2, & 3\end{bmatrix}^{\mathbf{T}}$. Im Vektor $\vec{c}_r$ werden somit nur die Größen der längsten, zusammenhängenden Teilstücke des Vektors $\vec{d}_r$ zusammengefasst, wobei $\vec{d}_r$ wiederum die binäre Auswertung von $(\vec{z}_{rpt} > 0 \wedge \vec{z}_{rqt} = 0)$ auf $\vec{x}_r$ darstellt (vgl. Gleichung 5.11). Diese Teilstücke in $\vec{c}_r$ werden in Gleichung 5.12 zu einem Vektor $\vec{c}$ zusammengefasst, der alle zusammenhängenden Teilstücke für alle Personen beinhaltet.

$$f_1(\vec{x}) = \frac{1}{S}\sum_r\sum_t \begin{cases} 1, & \vec{z}_{rpt} > 0 \wedge \vec{z}_{rqt} = 0 \\ 0 \end{cases} \tag{5.11}$$

$$f_2(\vec{x}) = \frac{\vec{c}^{\,2} \cdot \vec{e}}{\vec{c} \cdot \vec{e}} - 1 \quad \text{mit} \quad \vec{c} = (\vec{c}_r \mid \forall r \in R) \tag{5.12}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$$f_3(\vec{x}) = \frac{1}{S} \sum_r \sum_t^{|T|-1} \begin{cases} 1, & \vec{z}_{rht} \neq \vec{z}_{rht+1} \\ 0 \end{cases} \tag{5.13}$$

$$f_4(\vec{x}) = \frac{1}{S} \sum_r \sum_t^{|T|-1} \begin{cases} 1, & \vec{z}_{rdt} \neq \vec{z}_{rdt+1} \\ 0 \end{cases} \tag{5.14}$$

$$f_5(\vec{x}) = \frac{1}{S} \sum_r \sum_t \begin{cases} 1, & \vec{z}_{rdt-1} \neq \vec{z}_{rdt} \ \wedge \ \vec{z}_{rdt} \neq \vec{z}_{rdt+1} \\ 0 \end{cases} \tag{5.15}$$

$$\text{mit} \quad \vec{z}_{rd0} = \vec{z}_{rd1} \quad \text{und} \quad \vec{z}_{rd|T|} = \vec{z}_{rd|T|+1}$$

$$f_6(\vec{x}) = \frac{1}{S} \sum_r \sum_t \begin{cases} 1, & \vec{z}_{rht} \in H_{ext} \\ 0 \end{cases} \tag{5.16}$$

$$f_7(\vec{x}) = 1 - \frac{1}{SMP} \sum_r |r.MP \cap y_{rmf}| \tag{5.17}$$

Alle Zielfunktionen haben ein Optimum, wenn sie minimal sind.

Der Wertebereich aller Einzelzielfunktionen liegt im Intervall $[0, 1]$, die einzige Ausnahme stellt die Zielfunktion f_2 in Gleichung 5.12 dar, ihr Wertebereich liegt in $[0, |R| \cdot |T|^2]$, wobei die obere Schranke bedeuten würde, dass jeder einzelne Monat ein Stehmonat ist.

Die Anzahl der Stehmonate in Gleichung 5.11 ergibt sich aus der Summe aller Zuweisungen, bei denen eine Person keinen Ausbildungsfortschritt macht.

Gleichung 5.12 gewichtet zusammenhängende Perioden mit Stehmonaten bei einer Person quadratisch.

Gleichungen 5.13 und 5.14 zählen die Krankenhauswechsel jeder Person ab und summieren diese auf.

In Gleichung 5.15 werden Monate gezählt, an deren Beginn eine Person neu in einer Abteilung ist und am Ende des Monats diese Abteilung wieder verlässt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Das Abzählen von Ausbildungsmonaten bei Kooperationspartnern erfolgt in Gleichung 5.16.

Gleichung 5.17 bildet Schnittmengen aus tatsächlich gewählten Modulen und präferierten Modulen und summiert die Größen dieser Schnittmengen für alle Personen auf.

5.3 Formulierung des Optimierungsproblems

Aus der Problemdefinition in diesem Kapitel und den Kapiteln 1 und 2 ergeben sich folgende harte Randbedingungen:

(Die Notation x_{prt} bzw. x_{qrt} repräsentiert hierbei jeweils einen skalaren Eintrag des Vektors $\vec{x}$, der um die Dimension q bzw. p reduziert wurde und somit lediglich die zeitliche Zuordnung von Dienstposten bzw. Ausbildungsstellen zu Personen abbildet.)

RB 1 Jede Person kann in jedem Monat auf maximal einen Dienstposten p verplant werden (Gleichung 5.18).

RB 2 Jede Person kann in jedem Monat auf maximal eine Ausbildungsstelle q verplant werden. (Gleichung 5.19)

RB 3 Jede Zuteilung erfolgt binär (Zuteilung oder keine Zuteilung, Ganzzahligkeit) (Gleichung 5.20).

RB 4 Ausbildungsstelle und Dienstposten einer Zuweisung müssen derselben Abteilung zugeordnet sein ($\Longrightarrow$ selbes Krankenhaus) (Gleichung 5.21).

RB 5 Jeder Dienstposten darf in jedem Monat maximal einer Person zugewiesen sein (Gleichung 5.22). (Ausnahme für „leeren Dienstposten“ $p = 0$)

RB 6 Die Reihenfolge der Ausbildungsabschnitte einer Person muss eingehalten werden (Gleichung 5.23).

RB 7 Jede Ausbildungsstelle darf in jedem Monat maximal einer Person zugewiesen sein (Gleichung 5.24), (Ausnahme für „leere Ausbildungsstelle“ $q = 0$).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

RB 8 Die Kombination aus Ausbildungsstelle und Dienstposten muss zulässig sein - der Ausbildungsabschnitt der Ausbildungsstelle muss in den erlaubten Ausbildungen des Dienstpostens enthalten sein (Gleichung 5.25).

RB 9 Der aktuelle Ausbildungsabschnitt einer Person muss auf der jeweiligen Ausbildungsstelle erlaubt sein siehe 5.1) (Gleichung 5.26).

RB 10 Die Ausbildung einer Person (Basisausbildung, Allgemeinmedizin, ...) muss auf dem jeweiligen Dienstposten erlaubt sein (Gleichung 5.27).

RB 11 Die Zuweisungen müssen zusammenhängend sein - jede Person benötigt in jedem Monat zumindest einen Dienstposten, bis ihre Ausbildung abgeschlossen ist (Gleichung 5.28).

$$1 \leq \sum_{p \in P} x_{prt} \quad \forall\, r \in R,\ t \in T \tag{5.18}$$

$$1 \leq \sum_{q \in Q} x_{qrt} \quad \forall\, r \in R,\ t \in T \tag{5.19}$$

$$x_{pqrt} \in \mathbb{B} \quad \forall\, p \in P,\ q \in Q,\ r \in R,\ t \in T \quad \text{mit} \quad \mathbb{B} = \{0,1\} \tag{5.20}$$

$$x_{pqrt} * (p_d - q_d) = 0 \quad \forall\, p \in P,\ q \in Q,\ r \in R,\ t \in T \tag{5.21}$$

$$1 \leq \sum_{r \in R} x_{prt} \quad \forall\, p \in P \setminus \{0\},\ t \in T \tag{5.22}$$

$$r_{es,t-1} \leq r_{es,t} \quad \forall\, r \in R,\ t \in T \setminus \{1\} \tag{5.23}$$

$$1 \leq \sum_{r \in R} x_{qrt} \quad \forall\, q \in Q \setminus \{0\},\ t \in T \tag{5.24}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

$$x_{pqrt} * (p_{ed} - q_{ed}) = 0 \quad \forall\, p \in P,\ q \in Q \setminus \{0\},\ r \in R,\ t \in T \tag{5.25}$$

$$r_{es,t} - q_{es} = 0 \quad \forall\, r \in R,\ t \in T \tag{5.26}$$

$$x_{prt} * (r_{ed,t} - p_{ed}) = 0 \quad \forall\, p \in P,\ r \in R,\ t \in T \tag{5.27}$$

$$1 = \sum_{p \in P} x_{prt} \quad \forall\, r \in R,\ t \in \{t \in T \mid t \leq t_{r,fin}\} \tag{5.28}$$

Da durch die reine Minimierung der Zielfunktion $G(\vec{x})$ (5.10) und die Einhaltung der angeführten Randbedingungen nicht garantiert ist, dass ein vollständiger Ausbildungsplan vorliegt bzw. erzeugt wird, der den Abschluss aller Personen in der vorgegebenen Zeitdauer sicherstellt, wird eine zusätzliche Funktion $P(\vec{x})$ eingeführt, die in weiterer Folge als Straffunktion (*engl:* Penalty Function) bezeichnet wird. Die Forderung nach dem Ausbildungsabschluss jeder Person im vorgegebenen Zeitraum wird als weiche Randbedingung betrachtet, da diese, wenngleich wünschenswert, nicht notwendig für einen zulässigen Ausbildungsplan ist. Aus diesem Grund fließt der Fertigstellungsgrad der einzelnen Curricula der zu verplanenden Personen in die Bewertung einer Lösung ein.

$$P(\vec{x}) = k \cdot \sum_{r \in R} \sum_{i=1}^{3} p_i(\vec{x}_r)$$

mit

$$p_1(\vec{x}_r) = \begin{cases} 1, & \text{wenn Ausbildung von Person } r \text{ nicht abgeschlossen} \\ 0 & \end{cases} \tag{5.29}$$

und

$$p_2(\vec{x}_r) = 1 - \frac{t_{r,comp}}{r.c.dur}$$

und

$$p_3(\vec{x}_r) = 1 - \frac{t_{r,comp,viol}}{r.c.dur}$$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Mit den Formelzeichen (deren Werte im Zuge der Bewertung eines Individuums errechnet werden müssen):

- $r.c.dur$ ... Restdauer der Ausbildung einer Person zu Planungsbeginn
- $t_{r,comp}$ ... Anzahl der absolvierten Monate einer Person im Ausbildungsplan ohne Berücksichtigung der Randbedingungen
- $t_{r,comp,viol}$ ... Anzahl der absolvierten Monate einer Person im Ausbildungsplan, bis zum ersten Mal im Zeitverlauf eine Randbedingung verletzt wird

Die Straffunktion $p_1(\vec{x}_r)$ erzeugt einen Sprung, wenn die Ausbildung einer Person nicht abgeschlossen ist, $p_2(\vec{x}_r)$ ist ein Maß für den Fertigstellungsgrad der Ausbildung einer Person, wobei die Reihenfolge der einzelnen Module nicht berücksichtigt wird. Mit $p_2(\vec{x}_r)$ wird somit gemessen, wie viele Monate in einer Ausbildung generell noch fehlen. Die Funktion $p_3(\vec{x}_r)$ berücksichtigt die vorliegenden Randbedingungen und ist ein Maß für den tatsächlichen Fertigstellungsgrad. Jeder einzelne Funktionswert liegt im Wertebereich $[0, 1]$, wobei ein Wert von 1 jeweils dem schlechtesten Fall entspricht. Die gesamte Funktion $P(\vec{x})$ ist für eine einzelne Person in Abbildung 5.2 dargestellt, wobei ein Fertigstellungsgrad von 0 % bedeuten würde, dass die Person über den gesamten Zeitverlauf keinen Ausbildungsfortschritt macht. Die dargestellte Gerade im Bereich $[0, 100]$ beruht auf der Annahme $p_2(\vec{x}_r) = p_3(\vec{x}_r)$, was bedeutet, dass keine verletzten Randbedingungen vorliegen. Die Straffunktion hat zwei Aufgaben, sie dient einerseits dazu, ein „Gefälle“ in der Fitnessfunktion einzuführen (siehe roter Bereich in Abbildung 5.2), das dazu führt, dass Lösungen als besser bewertet werden, wenn deren Ausbildungsfortschritt aller Personen größer ist und dient andererseits zur Bestrafung unzulässiger Ausbildungsreihenfolgen (**RB 6**, **RB 9**, **RB 10**). Diese Verwendung einer Straffunktion ist ein gängiger Ansatz zur Berücksichtigung von Randbedingungen bei der Optimierung mithilfe evolutionärer Algorithmen [91], [92]. Die vorliegende Straffunktion wird als sta-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

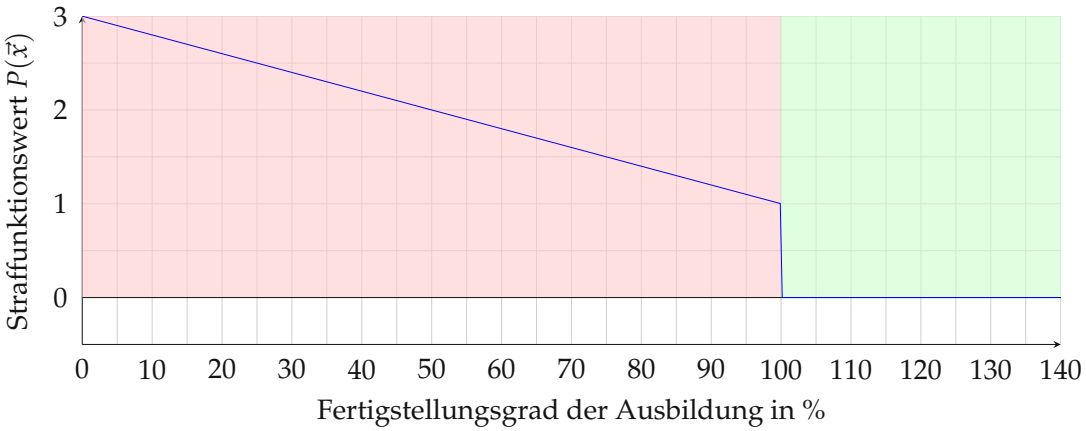


Abbildung 5.2: Straffunktionswert für die Ausbildung einer Person

tische Straffunktion bezeichnet, wobei p_1 die einfachste Form einer simplen Fallunterscheidung darstellt. Die Funktionen p_2 und p_3 führen einen Gradienten in Richtung der zulässigen Region ein, da sie sich proportional zur Schwere der Verletzung einer Randbedingung verhalten, also entsprechend der Entfernung einer Lösung vom zulässigen Bereich zu- bzw. abnehmen. Diese Art der Straffunktion deckt sich mit der Empfehlung, eine Straffunktion zu verwenden, die sich proportional zu den Kosten der „Reparatur" einer Lösung verhält, aus [93].
Die Fitness eines Individuums ergibt sich damit zu

$$F(\vec{x}) = G(\vec{x}) + P(\vec{x}) \tag{5.30}$$

und das Optimierungsproblem ergibt sich somit zu:

Minimiere $F(\vec{x})$ (aus 5.10)
unter Einhaltung der Nebenbedingungen (5.31)
5.18 – 5.27.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

5.4 Implementierung des Optimierungsverfahrens

Das gesamte Optimierungsverfahren wurde in der Programmiersprache `Python` [94] implementiert, wobei die Bibliotheken `matplotlib` [95] und `plotly` [96] für Visualisierungen, und `numpy` [97] sowie `pandas` [98] für Datenverarbeitung und Bewertungsfunktionen verwendet wurden. In den nachfolgenden Abschnitten werden einige, essenzielle Implementierungsdetails des entwickelten Verfahrens erläutert, wobei die Grundlagen aus Abschnitt 4.4 vorausgesetzt und nicht weiter ausgeführt werden. Auf die Verwendung von Softwarebibliotheken für EAs bzw. MOPs, wie z.B. [99] oder [100] wurde verzichtet, um das entworfene Verfahren möglichst gut an die vorliegenden Rahmenbedingungen anpassen zu können.

5.4.1 Das Individuum

Aufgrund des stark beschränkten Lösungsraums ist ein Großteil der theoretisch möglichen Kombinationen aus Ausbildungsstellen, Dienstposten, Personen und Zeitperioden, ungeachtet ihrer Sinnhaftigkeit, unzulässig. Diese Tatsache erfordert eine wohlüberlegte Definition der genotypischen und phänotypischen Lösungsrepräsentation. Erstrebenswert ist ein genotypischer Suchraum, der keinerlei Restriktionen unterliegt und somit nur zulässige Lösungen für das vorliegende Problem ermöglicht. Der Phänotyp einer Lösung muss die Bewertung der Lösung ermöglichen, diese also möglichst problemspezifisch und vollständig abbilden, sodass sowohl Randbedingungen als auch Zielfunktionen einfach ausgewertet werden können.
Der Wunsch nach einem genotypischen Suchraum ohne Einschränkungen konnte im Zuge dieser Arbeit nicht erfüllt werden, da die gleichzeitige Permutation mehrerer Ressourcen (Dienstposten kombiniert mit Ausbildungsstellen) eine erhebliche Problemkomplexität mit sich bringt und die mehrfache Ressourcenverwendung vermieden werden muss (siehe **RB 5** und **RB 7**). Da keine solche genotypische Darstellung gefunden werden konnte, wurde eine problemspezifische Wertkodierung gewählt, deren genotypische Abbildung ihrer phänotypischen Gestalt entspricht. Damit entfällt die Notwendigkeit von Kodierungs- und Dekodierungsfunktionen, sowie der damit verbundene

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Berechnungsaufwand. Außerdem ermöglicht die Anwendung der evolutionären Operatoren Mutation und Selektion auf diese Darstellung die Berücksichtigung der zuvor definierten Randbedingungen und die gezielte, problemspezifische Gestaltung besagter Operatoren. Mit der problemspezifischen Kodierung wird jedoch die Definition eigener Operatoren notwendig, die auf diese Darstellung anwendbar sind.

In Anlehnung an Tabelle 5.6 erfolgt die Kodierung eines Lösungskandidaten als zweidimensionaler Array (Matrix) (siehe Abbildung 5.3). Eine Zuweisung benötigt, um vollständig definiert zu sein, Informationen bezüglich Zeit, Person, Dienstposten und Ausbildungsstelle. Die Zeit entspricht dem Spaltenindex der Matrix, die Person dem Zeilenindex der Matrix. Die einzelnen Einträge der Matrix werden mit 3-Tupeln modelliert, wobei der erste Eintrag den Dienstposten p und der zweite Eintrag die Ausbildungsstelle q darstellt. Der dritte Eintrag des Tupels ist ein boolescher Wert, der Auskunft darüber gibt, ob dieser Eintrag verändert werden darf oder nicht bzw. umkehrt betrachtet „fixiert“ ist oder nicht. Diese zusätzliche Information ist notwendig, um Abwesenheitszeiten wie Urlaube, Krankenstände oder Karenzierung, aber auch bereits fest vorgegebene Zuweisungen oder in der Zukunft liegende Startzeitpunkte einer Ausbildung berücksichtigen zu können.

$$(p,\ q,\ fix)$$

Ein solcher Tupel stellt ein einzelnes **Gen** eines Individuums dar, die Aneinanderreihung der Gene und die anschließende „Faltung“ dieser Sequenz liefert die Matrix. Die Zeilenvektoren (1D-Arrays) des Individuums sind dabei die Zuweisungen einer Person r, wobei der jeweilige Zeilenindex r entspricht, wenn ein 1-basierter Index verwendet wird. Die Spaltenindizes entsprechen der jeweiligen Zeitperiode einer Zuweisung.

Auf diesem Array können einzelne vektorielle Operationen für die Überprüfung der Randbedingungen aus 5.3 und für die Berechnung der einzelnen Zielfunktionswerte definiert werden. Alle Zielfunktionen können somit zeilenweise (für jede Person) ausgewertet werden, die meisten Randbedingungen können spaltenweise überprüft werden. Diese Überprüfbarkeit der Randbedingungen je Spalte ermöglicht einerseits eine schnelle Berechnung der Zulässigkeit einer Lösung und ist andererseits zweckdienlich bei der Variation des Individuums (Mutation und Rekombination).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

(2, 9, true)	(2, 9, false)	...	(13, 6, false)
(1, 2, false)	(10, 8, false)	...	(5, 16, false)
...	...	...	
(5, 6, false)	(13, 6, false)	...	(25, 4, false)

(a) Schema

```
[ [ (2, 9, true),   (2, 9, false),   ..., (13, 6, false) ],
  [ (1, 2, false),  (10, 8, false),  ..., (5, 16, false) ],
  [ (...,...,...),  (...,...,...),   ..., (...,...,...)  ],
  [ (5, 6, false),  (13, 6, false),  ..., (25, 4, false) ], ]
```

(b) Datenstruktur

Abbildung 5.3: Beispielhafte Kodierung des Individuums schematisch und als 2D-Array

Die Wahl dieser Struktur der Lösungsrepräsentation hat folgende Konsequenzen hinsichtlich der Randbedingungen:

- **RB 1**, **RB 2** und **RB 3** werden zwangsläufig eingehalten.
- **RB 4** kann überprüft werden, indem zwei Arrays mit den Abteilungen der Dienstposten und Ausbildungsstellen erzeugt und elementweise auf Gleichheit überprüft werden.
- **RB 5** und **RB 7** können sichergestellt werden, indem spaltenweise die Häufigkeit der Einträge (Dienstposten, Ausbildungsstellen) gezählt wird (max. zulässig = 1).
- **RB 6** kann zeilenweise überprüft wird, indem für jede Zeile ein Vektor mit den entsprechenden Ausbildungsabschnitten erzeugt wird (der so erzeugte Vektor muss aufsteigend sortiert sein).
- **RB 8** kann analog zu **RB 4** überprüft werden, indem elementweise die zugehörige Ausbildung von Dienstposten und Ausbildungsstelle verglichen werden,

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

- **RB 9** und **RB 10** müssen zeilenweise und sequenziell überprüft werden, indem der Ausbildungsfortschritt einer Person fortgerechnet wird.

- **RB 11** kann nach Bestimmung des Fertigstellungszeitpunktes der Ausbildung einer Person $t_{r,fin}$ überprüft und in der Regel stets eingehalten werden, sofern je zu verplanender Person pro Monat zumindest ein Dienstposten verfügbar ist.

Da die meisten Randbedingungen relativ leicht überprüft werden können, können sie bei der Veränderung der Individuen ebenfalls einfach berücksichtigt werden. Die verbleibenden Randbedingungen werden bei der Auswertung eines Individuums mithilfe der Straffunktion berücksichtigt. Die zeilenweise Unabhängigkeit der Zuweisungen ermöglicht eine vollständig parallele Berechnung der Zielfunktionswerte für jede einzelne Person. Für die Modellierung des Individuums wurde das `NumPy` Package [97] und der darin verfügbare n-dimensionale Array `ndarray` (siehe Abbildung 5.4 und Listing 5.1) verwendet.

```python
GeneDtype = np.dtype(
    [("position", np.int32), ("training_position", np.int32), ("fixed",
    ↪ np.bool8)]
)

class Assignment(NamedTuple):
    """
    Assignment Container for use in Individual
    """
    position: int
    training_position: int
    fixed: bool


class Individual(np.ndarray):
    def __new__(cls, n_persons: int, n_months: int, *args, **kwargs):
        inst = super().__new__(cls, shape=(n_persons, n_months),
        ↪ dtype=GeneDtype, *args, **kwargs)
        inst.fill((0, 0, 0))
        inst.evaluated = False
        inst.fitness: Fitness | None = None
        return inst

    def __array_finalize__(self, obj):
        self.evaluated: bool = getattr(obj, "evaluated", False)
        self.fitness: Fitness | None = getattr(obj, "fitness", None)
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
25
26      def __hash__(self) -> int:
27          x = xxhash.xxh64()
28          x.update(self.tobytes())
29          return x.intdigest()
30
31      def __eq__(self, other: "Individual") -> bool:
32          return np.array_equal(self, other)
33
34      def __ne__(self, other: "Individual") -> bool:
35          return not np.array_equal(self, other)
36
37      @property
38      def n_months(self) -> int:
39          return self.shape[1]
40
41      @property
42      def n_persons(self) -> int:
43          return self.shape[0]
44
45      @property
46      def genes(self) -> list[Assignment]:
47          return list(chain(*[list(i) for i in self]))
48
49      def hamming_distance(
50          self, other: "Individual", only_training_positions: bool = True
51      ) -> int:
52          if self.shape == other.shape:
53              return np.count_nonzero(
54                  np.not_equal(self.training_positions,
                    ↪  other.training_positions))
55          else:
56              raise ValueError("dimensions of individuals must be equal")
```

Code 5.1: Implementierung des Individuums

Auf einige Implementierungsdetails wurde in den Darstellungen im Sinne der Verständlichkeit verzichtet.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

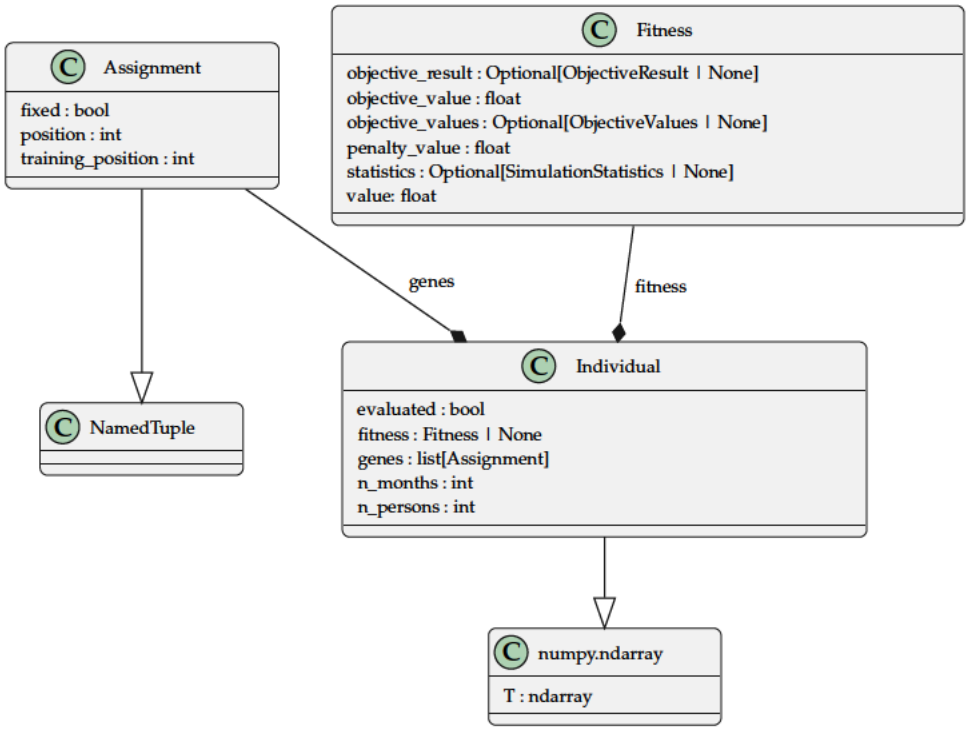


Abbildung 5.4: UML-Diagramm des Individuums

Die Operatoren **Rekombination** und **Mutation** wurden so definiert, dass bei ihrer Anwendung auf Individuen zwar Verschlechterungen hinsichtlich der Lösungsgüte, jedoch keine Verletzung der harten Randbedingungen möglich ist (siehe 5.4.2). Wenn bei Anwendung eines Operators auf ein Individuum I ein Individuum I' erzeugt wird und die Fitness der so erzeugten Lösung schlechter als die Fitness der ursprünglichen Lösung ist, so gilt $I.F \succ I'.F$, bzw. $I'.F \prec I.F$ das Individuum I ist also „fitter“ als I'.

Der paarweise Unterschied des Genmaterials von einzelnen Individuen ist ausschlaggebend für die **Diversität** einer Population. Als geeignetes Maß für die „Distanz“ zwischen zwei Individuen wird die Anzahl aller voneinander verschieden Ausbildungsstellen verwendet. Diese Anzahl kann bestimmt werden, indem zwei Matrizen mit den Einträgen der Ausbildungsstellen elementweise subtrahiert und anschließend die von 0 verschiedenen Einträge abgezählt werden. Diese Metrik spiegelt näherungsweise den Unterschied zwischen den Ausbildungsverläufen in zwei Individuen wider und kann als Anpassung der Hamming-Distanz auf dieses nicht-binäre Problem interpretiert werden. In Abbildung ist die Berechnung der Metrik schematisch veranschaulicht, wobei die Matrizen die Ausbildungsstellen der Zuweisungen enthalten.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

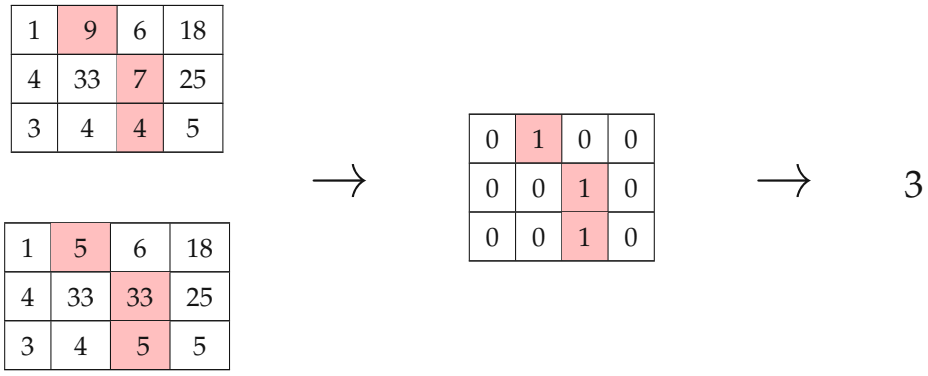


Abbildung 5.5: Distanzmetrik für das Individuum

5.4.2 Die Evolutionären Operatoren

Der Ablauf des entwickelten Verfahrens nimmt die Gestalt aus Darstellung 4.1 bzw. des Algorithmus 1 an. Daher werden in diesem Abschnitt lediglich die einzelnen Bestandteile des Algorithmus genauer beschrieben, nicht jedoch der GA selbst.
Die notwendigen Informationen zur Planung werden mithilfe einer Datenstruktur, wie sie in Anhang A.5 stark vereinfacht dargestellt ist, bereitgestellt. Aus diesen Informationen werden sowohl Ressourcenverfügbarkeiten und Kapazitätsbedarfe (Bedarf an Dienstposten und Ausbildungsstellen) abgeleitet.
Um eine effizientere Suche zu ermöglichen, indem für jede Person nur tatsächlich zulässige Paarungen aus Ausbildungsstellen und Dienstposten verplant werden, werden vor der Initialisierung des Optimierungsverfahrens die dafür notwendigen Mengen an zulässigen Kombinationen aus Dienstposten und Ausbildungsstellen für jede Person bestimmt und gespeichert.

Für jede Person existiert somit eine Menge zulässiger Kombinationen aus Ausbildungsstellen und Dienstposten, die diese belegen kann. Zusätzlich werden aus den vorliegenden Daten alle notwendigen Informationen für die Bewertung eines Individuums bestimmt. Alle erforderlichen Informationen für die Ausführung des Optimierungsverfahrens werden dem Algorithmus gebündelt bei dessen Initialisierung übergeben.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

5.4.2.1 Selektion

Für die Selektion kann ein einfacher Standardalgorithmus wie z.B. die RWS, das SUS oder die Turnierselektion verwendet werden, da eine einfache Ordnungsrelation zwischen einzelnen Individuen besteht. Nichtsdestotrotz ermöglicht die Fitness eines Individuums auch komplexere Selektionsverfahren, die Funktionswerte einzelner Ziele benötigen. Ein solches Verfahren kommt in der Umweltselektion zum Einsatz. Dieses Verfahren ist eines der gängigsten Selektionsverfahren zur Behandlung von MOPs und wird NSGA-II (**N**on-Dominated-**S**orting-**G**enetic-**A**lgorithm-**II**) genannt [101]. Der Name lässt zwar auf einen komplett andersartigen GA schließen, der tatsächliche Unterschied zu klassischen GAs liegt jedoch im Selektionsmechanismus. Dieser Mechanismus ist von Natur aus elitär und basiert auf dem Begriff der Paretodominanz. Er ist in zwei aufeinanderfolgende Schritte aufgeteilt, das nicht-dominierte Sortieren (non-dominated sorting) und die anschließende Sortierung der letzten relevanten Front anhand der Crowding Distance der enthaltenen Individuen (siehe Abbildung 5.6).

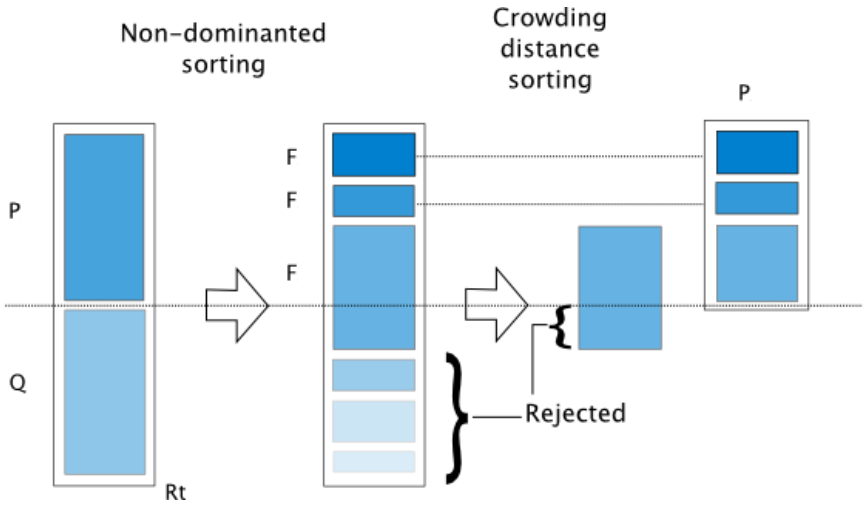


Abbildung 5.6: NSGA-II Selektion [102]

Beim nicht-dominierten Sortieren werden Pareto-Fronten aus der bestehenden Population erzeugt, in denen die darin enthaltenen Individuen von keinem der anderen Individuum dominiert werden. So entstehen in der Regel mehrere Fronten F (siehe Abbildung 5.6). Die letzte Front, von der noch Individuen in der nachfolgenden Population Platz finden würden, wird gemäß der Crowding-Distance (entspricht dem Manhattan

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Abstand im Zielfunktionsraum [102]) sortiert. Mit der so sortierten letzten Front, wird die Population der Nachkommen aufgefüllt. Für eine detaillierte Beschreibung des Verfahrens wird auf [101] verwiesen. Die adaptierte Implementierung für das entwickelte Verfahren ist in Anhang A.6 als Quellcode angeführt.

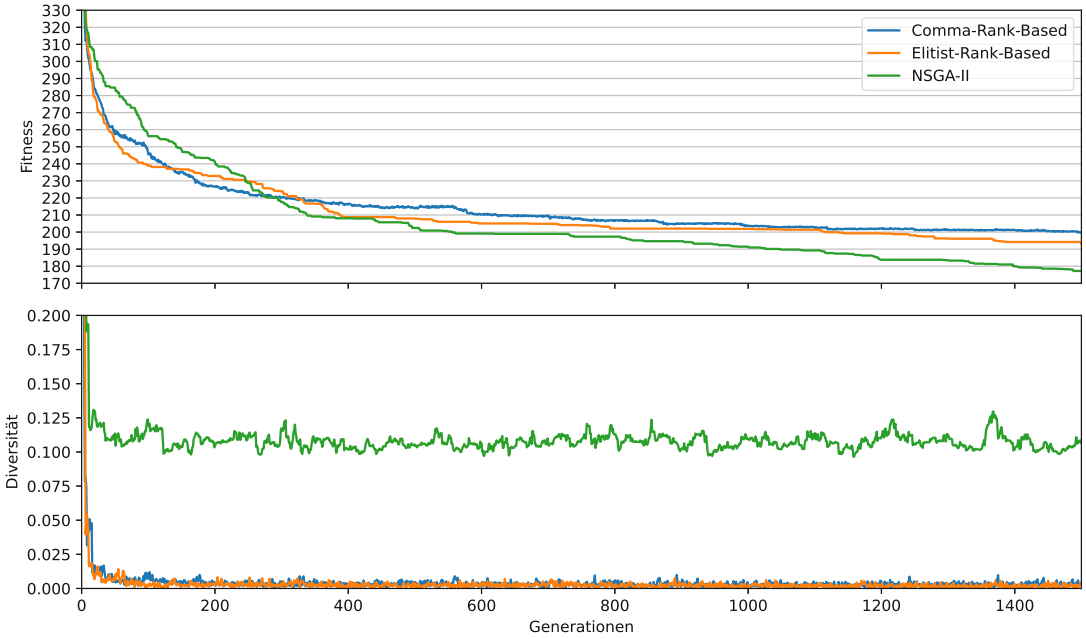


Abbildung 5.7: Vergleich Komma-Selektion, Plus-Selektion mit Überlappungsgrad 1 und NSGA-II

Diese Art der Selektion ermöglicht im Gegensatz zu rangbasierten oder rein fitnessproportionalen Verfahren die Erhaltung einer größeren Vielfalt in der Population bei gleichzeitiger Berücksichtigung aller Zielgrößen eines MOPs. So werden vielfältigere, Paretooptimale Lösungen erzeugt und erhalten [101]. Der Operator trägt zur Erhaltung der Diversität der Population bei und verhindert so eine vorzeitige Konvergenz, wie es bei Algorithmen mit rangbasierter Selektion beobachtet werden kann (lokale Optima werden überwunden und unerforschte Bereiche erkundet). Im Rahmen der Entwicklung des Verfahrens wurden sowohl Konfigurationen mit klassischer Selektion, ausschließlich aus den Nachkommen (Komma-Selektion), elitärer, rangbasierter Selektion mit Überlappungsgrad n (Selektion aus den Nachkommen und den n besten Eltern) sowie mit dem elitären NSGA-II erprobt. In Abbildung 5.7 sind diese drei Varianten gegenübergestellt. Allgemein kann beobachtet werden, dass die Variante mit nicht-dominierter

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Sortierung zwar zu Beginn langsamer als ihre Mitstreiter, dafür jedoch gleichmäßiger konvergiert. Am Ende der Optimierungsläufe kann beobachtet werden, dass die rangbasierten Verfahren zu stagnieren beginnen und kaum mehr Verbesserungen erzielen können, wohingegen der NSGA-II hier seine Stärken ausspielen kann.

In der vorliegenden Arbeit wird die Turnierselektion für die **Elternselektion** verwendet, da dieses Verfahren in [101] ergänzend zum NSGA-II empfohlen wird und den nötigen Selektionsdruck aufbaut, um das verstärkte „Überleben" von zulässigen Individuen zu fördern. Das SUS wurde ebenfalls für die Elternselektion getestet, weist jedoch zu geringen Selektionsdruck auf. Somit führt es zwar zu einer diverseren Population, allerdings werden damit wesentlich mehr schlechte Individuen in die Nachkommen übernommen, die unnötigen Aufwand in der Bewertung mit sich bringen, ohne dabei Mehrwert in Hinblick auf die Lösungsgüte zu stiften.

5.4.2.2 Rekombination

Da insbesondere in höherdimensionalen MOPs die Wirksamkeit von Rekombinationsoperatoren zu hinterfragen ist [103], wurde bei der Implementierung des Verfahrens auf die Entwicklung eines ausgefeilteren Rekombinationsverfahrens verzichtet und das Hauptaugenmerk auf die Entwicklung des Mutationsoperators gelegt. Trotzdem wird ein funktionierender Rekombinationsoperator für das vorliegende Problem beschrieben und eingesetzt.

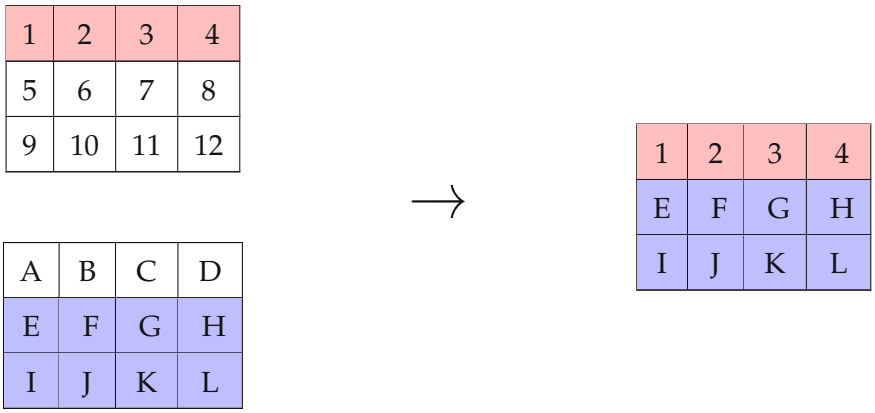


Abbildung 5.8: Theoretisch ideale Rekombination

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Der in Abbildung 5.8 dargestellte Operator wäre denkbar ideal, da die Ausbildungsverläufe einzelner Personen erhalten bleiben würden und so eine Mischung des Genmaterials der Eltern bei gleichzeitiger Erhaltung der gefundenen Schemata möglich wäre. Die Anwendung eines solchen Operators würde in der Realität jedoch eine extreme Verletzung der Randbedingungen mit sich bringen, da eine Mehrfachverwendung von Dienstposten und Ausbildungsstellen nicht ausgeschlossen werden kann. Somit würde dieser Operator eine anschließende Korrektur (Löschen von Einträgen) erfordern, um eine zulässige Lösung zu erhalten. Diese Korrektur führt allerdings mit sehr hoher Wahrscheinlichkeit zur Verschlechterung der Lösung, da benötigte Ausbildungsbestandteile für einzelne Personen dadurch wegfallen. Damit scheidet ein solcher Operator aus. Die Alternative zu dieser Art von Operatoren bildet eine Verallgemeinerung des klassischen n-Punkt-Crossover für ein zweidimensionales Individuum. Der in Abbildung 5.9 dargestellte und in Algorithmus 2 formulierte Operator zerteilt beide Elternindividuen „vertikal“ und setzt sie stückweise wieder zusammen. Damit werden keine Randbedingungen verletzt, eine Verschlechterung der Ausbildungsreihenfolgen einzelner Personen ist jedoch sehr wahrscheinlich, sofern die Individuen nicht identisch sind. Der Operator liefert also zulässige, jedoch nicht zwangsläufig sinnvolle bzw. zielführende Ergebnisse.

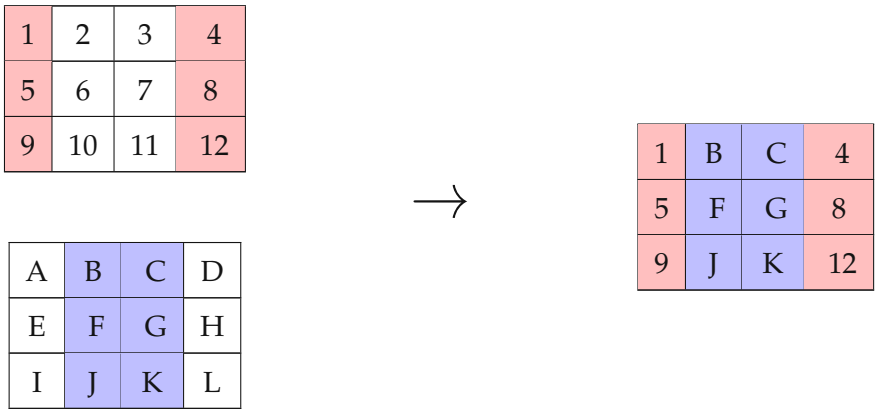


Abbildung 5.9: Verwendeter Rekombinationsoperator - 2-Punkt-Crossover

Zur Bestimmung einer geeigneten Rekombinationswahrscheinlichkeit wurden zwei Experimente für gleichmäßig verteilte Parameterwerte für p_c im Bereich [0,1] durchgeführt. Im ersten Versuch wurden für alle Werte die entsprechenden Konvergenzgeschwindigkeiten bestimmt, die Ergebnisse sind in Abbildung 5.10 ersichtlich. Aus dieser Be-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithmus 2: 2-Punkt-Crossover

Input : I_1, $I_2 \leftarrow$ Individuen

Output : I'_1, I'_2 gekreuzte Individuen

$j \leftarrow$ wähle Zufallszahl in $[0, I_1.n\_monate]$

$k \leftarrow$ wähle Zufallszahl in $[0, I_1.n\_monate]$

if $j > k$ **then**

    k, j = j, k

end

```
/* Verkette einzelne Spalten der Individuen                    */
/* mit I[Zeilenindex, Spaltenindex]                            */
```

$I'_1 \leftarrow I_1[:, :j] + I_2[:, j:k] + I_1[:, k:]$

$I'_2 \leftarrow I_2[:, :j] + I_1[:, j:k] + I_2[:, k:]$

return I'_1, I'_2

trachtung könnte man schließen, dass eine hohe Rekombinationswahrscheinlichkeit zu schnellerer Optimierung führt.

In Abbildung 5.11 ist jedoch genau das Gegenteil der Fall, sie zeigt den Optimierungsverlauf ausgehend von ein und derselben Population, die in einem vorigen Durchlauf innerhalb von etwa 600 Generationen erzeugt wurde.

Die gezeigten Verläufe sind lediglich Momentaufnahmen und damit einer verhältnismäßig großen Streuung unterworfen. Man kann feststellen, dass der Rekombinationsoperator zu Beginn einer Optimierung einen tendenziell größeren Nutzen aufweist, der mit zunehmender Konvergenz abnimmt. Das lässt sich intuitiv begründen, da zu Beginn der Optimierung die rekombinierten Individuen nicht gravierend schlechter sind als rein mutierte Individuen und der Rekombinationsoperator zur Exploration beiträgt. Im weiteren Verlauf nimmt jedoch die durchschnittliche Güte in der Population zu und die Diversität ab, was bedeutet, dass Individuen, die durch Rekombination erzeugt werden, mit hoher Wahrscheinlichkeit wesentlich schlechter sind als nicht rekombinierte Individuen. Deshalb weist die Rekombination wenig Relevanz bei der Verbesserung der Lösung auf, da der vorliegende Operator nicht zielgerichtet zur Exploitation beitragen kann. Diese Erklärung ist nicht allgemein, sondern ausschließlich im Hinblick auf den

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

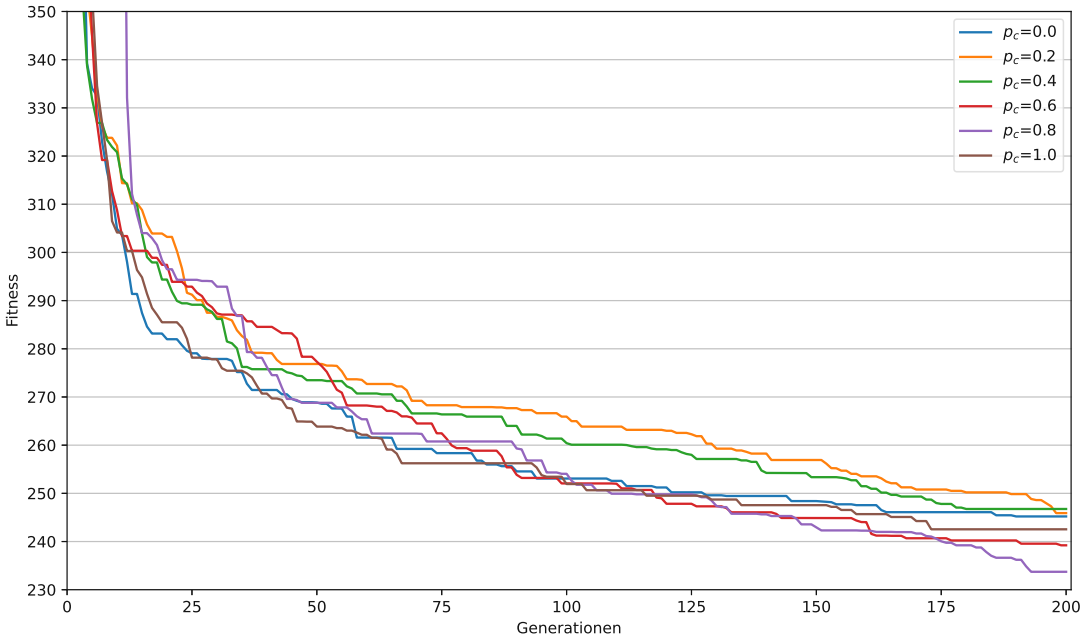


Abbildung 5.10: Auswirkungen der Rekombinationswahrscheinlichkeit am Anfang der Optimierung

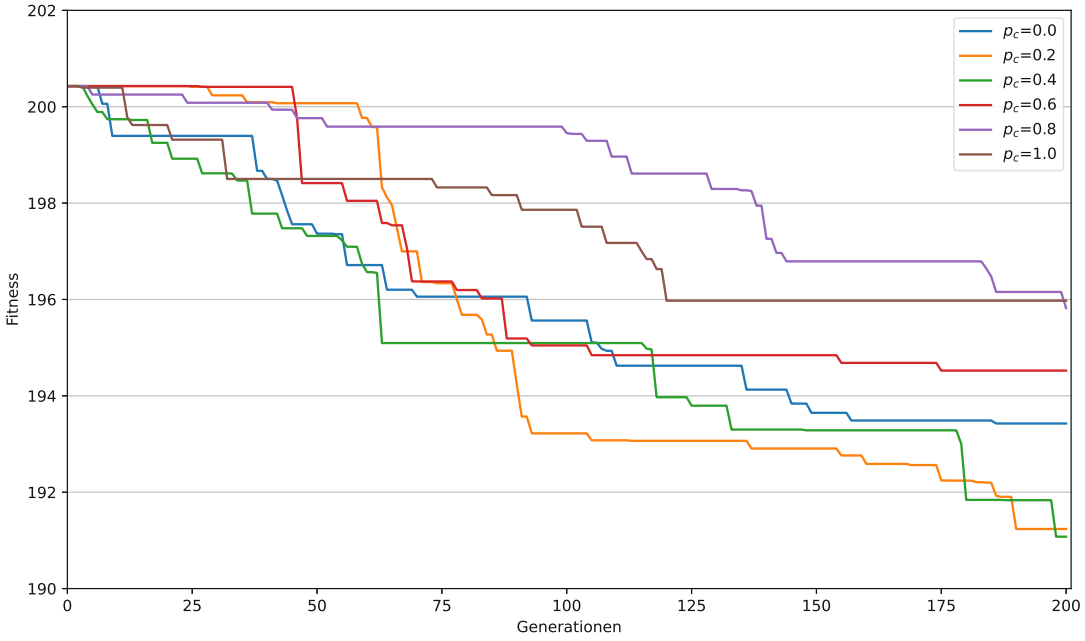


Abbildung 5.11: Auswirkungen der Rekombinationswahrscheinlichkeit nach ca. 600 Generationen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

beschriebenen Operator gültig. Da der Rekombinationsoperator im Optimierungsverlauf zunehmend an Relevanz verliert, werden kleine Rekombinationswahrscheinlichkeiten ($c_p \leq 0.2$) als sinnvoll angesehen. In dieser Arbeit wird der Rekombinationsoperator mit einer Wahrscheinlichkeit $p_c = 0.2$ eingesetzt, da sich bei längeren Optimierungsläufen gezeigt hat, dass sich der in Abbildungen 5.10 und 5.11 erkennbare Trend fortsetzt und der Rekombinationsoperator mit zunehmendem Optimierungsfortschritt an Relevanz verliert. Für die Weiterentwicklung des Verfahrens wäre allerdings eine dynamische Anpassung von c_p denkbar, sodass mit verhältnismäßig größeren Rekombinationswahrscheinlichkeiten begonnen wird, die im weiteren Optimierungsverlauf abnehmen.

5.4.2.3 Mutation

Der Mutationsoperator ist das Herzstück des entwickelten Verfahrens und ist sowohl für die Exploration als auch die Exploitation des Suchraums zuständig. Davon abgesehen übernimmt er auch die Aufgabe der Reparatur von „beschädigten“ Individuen, wobei der Begriff Reparatur hierbei nicht als eine Reparatur des Individuums bis hin zur vollständigen Zulässigkeit verstanden werden darf, sondern vielmehr einer stückweisen Annäherung an den zulässigen Bereich durch verbessernde Maßnahmen entspricht. Der Einsatz von Reparaturalgorithmen in GAs für kombinatorische Optimierungsprobleme ist gängige Praxis [104].

Nach der Erzeugung einer Startpopulation aus zufällig generierten Individuen (siehe Algorithmus 6) sind die darin enthaltenen Individuen zwar zulässig im Sinne der harten Randbedingungen, erfüllen im Allgemeinen jedoch die Bedingungen, die die Straffunktion (Gleichung 5.29) abbildet, nicht. Im Zuge der Entwicklung des Mutationsoperators hat sich gezeigt, dass eine gesonderte Behandlung von Individuen mit Straffunktionswert > 0 zweckmäßig ist, um diese in einem ersten Schritt in den zulässigen Bereich des Zielfunktionsraums ($P(\vec{x}) = 0$) zu bringen und anschließend eine Optimierung der Zielfunktionswerte zu vollziehen. Demnach sieht der entwickelte Mutationsoperator eine Fallunterscheidung zwischen „zulässigen“ und „unzulässigen“ Individuen vor. Der gesamte Operator ist in Listing 5.2 als Quellcode abgebildet. Die dort verwendeten

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Funktionen sind in Anhang A.7 vollständig angeführt. In Algorithmus 3 wird deshalb die Funktionsweise lediglich oberflächlich, mit Fokus auf die grundlegenden Funktionsprinzipien des Operators erklärt. Unabhängig von der Güte eines Individuums, werden zu Beginn der Mutation alle Zuweisungen ohne Ausbildungsstellen gelöscht, sodass insbesondere jene Posten frei werden, die eine Person nach Ausbildungsabschluss besetzt und nicht mehr benötigt.

Für unzulässige Individuen werden die Zeilen der Matrix, in denen ein Straffunktionswert > 0 auftritt, mithilfe zielgerichteter Operatoren, die fehlenden Ausbildungsbestandteile der betreffenden Person eingeführt. Dies geschieht, indem zuerst zeilenweise alle Zuweisungen, die keinen Ausbildungsfortschritt erzielen, gelöscht werden, und die übrigen Zuweisungen gruppiert und sortiert werden, sodass sie möglichst am Beginn des Ausbildungsplans stehen und die Zuweisungen nach Krankenhäusern und Abteilungen gruppiert sind. Die so entstehenden Kollisionen mit Zuweisungen anderer Personen werden gelöst, indem die Zuweisungen der betroffenen Personen gelöscht werden. Abschließend werden alle Lücken in der betreffenden Zeile zufällig mit noch freien Kombinationen aus Dienstposten und Ausbildungsstellen gefüllt. Diese Prozedur wird für jede „unzulässige“ Zeile durchlaufen und führt zu einer verhältnismäßig schnellen Erzeugung zulässiger Individuen.

Befindet sich ein Individuum bereits im zulässigen Bereich, so durchläuft es einen alternativen Pfad. Auf diesem Pfad wird ein Individuum, sofern die Schwelle zur Mutationswahrscheinlichkeit überschritten wird ($0 \leq$ Zufallszahl $\leq p_m$), zufällig einem der drei vordefinierten Operatoren (siehe Anhang A.7) unterworfen. Von diesen Operatoren ist der erste für das Ersetzen von einmonatigen Zuweisungen verantwortlich. Das führt bei Erfolg zu einer direkten Verbesserung der Zielgrößen „Einmonatige Zuweisungen“ ($f_5(\vec{x})$) und „Ortswechsel“ ($f_3(\vec{x})$ und $f_4(\vec{x})$). Die restlichen Zielgrößen können dadurch allerdings durchaus negativ beeinflusst werden.

Der zweite Operator versucht den vollständigen Durchlauf durch ein Modul am Stück für eine Person an ein und demselben Ort zu vollziehen. Dazu werden alle verwendeten Zuweisungen für ein Modul, für eine Person gelöscht und die benötigte Menge an Zuweisungen an einer anderen Stelle (Monat) wieder eingefügt. Um dem entstandenen Schaden und der damit verbundenen Verletzung von Randbedingungen entgegenzu-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

wirken, werden nach vollzogener Operation die betroffenen Einträge anderer Personen (doppeltes Auftreten von Zuweisungen in einem Monat) gelöscht und nach Möglichkeit wiederhergestellt. Im Idealfall führt dieser Operator zur Reduktion der einmonatigen Zuweisungen und der Ortswechsel bei gleichzeitiger Minimierung von Stehmonaten.
Der dritte Operator versucht gezielt einmonatige Zuweisungen, die mit einem Krankenhauswechsel verbunden sind, zu reduzieren, indem an diesen Stellen benachbarte Zuweisungen kopiert werden. Die Mutation von Individuen im zulässigen Bereich ist somit ein Wechselspiel der definierten Operatoren. Die Auswahl dieser Operatoren war keineswegs willkürlich, sondern wurde im Rahmen einer Vielzahl an Versuchen getroffen. Die Kombination der Operatoren ermöglicht eine gleichzeitige, zielgerichtete Optimierung aller Zielgrößen, obwohl nicht für alle Zielgrößen ein gesonderter Operator vorgesehen ist. Klarerweise wurden zu Beginn der Entwicklung, einige stark zufällig agierende Operatoren, angelehnt an die in Abschnitt 4.4 beschriebenen Mutationsoperatoren getestet. Da diese Versuche jedoch zu keiner bzw. keiner zufriedenstellenden Verbesserung der Startlösungen führten, mussten gesonderte, problemspezifische und informiert agierende Operatoren entwickelt werden.

```
class Algorithm:
    def mutation(self:"Algorithm", ind: Individual):

            ind = deepcopy(ind)
            penalty = ind.fitness.penalty_value > 0
            # lösche alle Zuweisungen, die keinen Ausbildungsfortschritt
             ↪ bringen
            ind = delete_assignments_without_training(ind)

            if penalty:
                p_idx = []
                idx_map = self.calculation_data.person_calculation_data.invert
                 ↪ ed_index
                # bestimme individuen deren Ausbildung nicht fertig ist
                for res in ind.simulation_result:
                    if not res.education_finished_without_violation:
                        p_idx.append(idx_map[res.person_id])

                if len(p_idx) > 0:

                    random.shuffle(p_idx)
                    # sortiere Zuweisungen um
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
                    ind = sort_positions_for_person(ind, p_idx,
                    ↪ self.calculation_data)

                    # befülle greedy und zufällig mit freien Kombinationen
                    # aus Dienstposten und Ausbildungsstellen
                    ind = fill_all_possible_indices_with_unused_combinations(
                        ind, p_idx, self.calculation_data
                    )

            if random.random() <= self.mutation_probability:
                    if penalty == False:
                    operators = [
                         # suche einmonatige Zuweisungen und ersetze
                         # sie durch eine andere freie Kombination
                        insert_unused_combinations_instead_of_sma,

                        # wähle module einer Peron und sortiere die
                        ↪ Zuweisungen so um,
                        # dass das Modul am Stück und in einer Abteilung
                        ↪ abgeschlossen wird
                        complete_module_in_one_piece_for_person,

                        #tausche eine einmonatige Zuweisung einer Person
                        # mit der Zuweisung einer anderen Person
                        swap_sma_assignment,
                    ]
                    # wähle zufällig zwischen den Operatoren mit bias
                    weights = [0.2, 0.4, 0.4]
                    operator = random.choices(operators, weights=weights,
                    ↪ k=1)[0]
                    ind = operator(ind, self.calculation_data)

                    else:
                        # wenn das Individuum nicht zulässig ist, befülle
                        ↪ greedy mit neuen Kombinationen
                        ind = insert_unused_combinations(
                            ind, self.calculation_data, fraction=0.15
                        )

            # repariere und befülle mit allen noch verfügbaren Dienstposten
            ↪ und Ausbildungsstellen
            ind = infer_unassigned_positions_v3(ind, self.calculation_data)
            ind = infer_unassigned_training_positions(ind,
            ↪ self.calculation_data)

            return ind
```

Code 5.2: Mutation eines Individuums

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithmus 3: Mutationsoperator

Input : $I \leftarrow$ Individuum

Input : $M \leftarrow$ Berechnungsinformationen des GA

Input : $p_m \leftarrow$ Mutationswahrscheinlichkeit

Output : $I \leftarrow$ mutiertes Individuum

```
1  I ← lösche alle Zuweisungen in I, die keine Ausbildungsstellen haben
2  if I.fitness.penalty_value > 0 then
3      p_idx ← bestimme Indizes mit unzulässigen Ausbildungen
4      p_idx ← sortiere p_idx zufällig um
5      forall idx in p_idx do
6          I[idx] ← lösche alle Zuweisungen in Zeile, die keinen Fortschritt bringen und
             sortiere/gruppiere übrige Zuweisungen nach Krankenhaus und Abteilung
7      end
8      forall idx in p_idx do
9          I[idx] ← befülle alle unbesetzten Plätze in Zeile mit verfügbaren Kombinationen
10     end
11 end
12 u ← Zufallszahl in [0, 1] ∈ R
13 if u ≤ p_m then
14     if I.fitness.penalty_value == 0 then
15         ops ← Liste der verfügbaren Operatoren
16         OP ← wähle einen der verfügbaren Operatoren aus ops zufällig mit Gewichtung
17         I ← wende OP auf Individuum an
18     else
19         I ← fülle einen vorgeegebenen Teil der Zeilen mit freien Kombinationen auf
20     end
21 end
   /* Reparatur des erzeugten Individuums                                    */
22 I ← fülle mit Dienstposten (Algorithmus 7)
23 I ← fülle mit Ausbildungsstellen (Algorithmus 8)
24 return I
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

5.4.3 Bewertung

Die Bewertung einer neuen Population erfolgt gemäß Algorithmus 4. Dabei gliedert sich das Vorgehen in drei wesentliche Teile,

1. die Bestimmung des Ausbildungsfortschritts für jede Person, anhand der verplanten Zuweisungen in einem Individuum,

2. das anschließende Bestimmen und Löschen von überflüssigen Zuweisungen im Individuum und

3. die abschließende Berechnung und Zuweisung der Straf- und Zielfunktionswerte.

Die Bereinigung des Individuums ändert nichts am Ausbildungsfortschritt der einzelnen Personen und damit auch nichts an der Güte eines Individuums, bewirkt jedoch, dass unnötigerweise verplante Ressourcen nicht verwendet, also wieder frei werden. Algorithmus 4 ist vollständig parallelisierbar, da Zugriffe und Veränderungen am Individuum nur zeilenweise geschehen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithmus 4: Bewertung und Bereinigung eines Individuums

Input : $I \leftarrow$ Individuum

Input : $M \leftarrow$ Berechnungsinformationen des GA

Output : $I \leftarrow$ bewertetes Individuum

1 **for** $j \leftarrow 0$ **to** *length(I)* **do**

2 $r \leftarrow I\,[j]$ `/* Zeilenvektor aus` I `(Menge aller Zuweisungen) */`

3 $C \leftarrow$ leite Curriculum für Person aus M mit j ab

4 $A \leftarrow$ leite Zuweisungen mit Info über belegte Module aus r mit C ab

5 $y \leftarrow$ Bestimme Ausbildungsfortschritt aus A gemäß Curriculum C

6 $A' \leftarrow$ Bestimme Zuweisungen, die Ausbildungsfortschritt erzielen mit C aus A

7 $B \leftarrow A \setminus A'$ berechne Zuweisungen ohne Ausbildungsfortschritt

8 $D \leftarrow$ bestimme Menge an Zuweisungen nach Ausbildungsabschluss aus A

9 $B' \leftarrow B \setminus D$ entferne Zuweisungen nach Fertigstellungszeitpunkt aus B

10 $B'' \leftarrow$ lösche Ausbildungsstellen aus jeder Zuweisung in B'

11 $r \leftarrow A' \cup B''$

12 **forall** *Zielfunktionen* f_i **do**

13 $f_{i,j} \leftarrow$ berechne Einzelzielfunktionswert aus r

14 $I.f_i \leftarrow I.f_i + f_{i,j}$

15 **end**

16 $p_j \leftarrow$ Berechne Straffunktionswert für Zeile r mit y

17 $I.p \leftarrow I.p + p_j$

18 $I\,[j] \leftarrow r$ ... schreibe bereinigte Zuweisungen in Individuum

19 $I.F \leftarrow I.p + \sum I.f_i$

20 **end**

21 **return** P

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

5.5 Das Verfahren

Das entwickelte Verfahren ist in Algorithmus 5 definiert. Die Problembeschreibung, die dem Algorithmus in 5 übergeben wird, ist in Anhang A.5 dargestellt und enthält alle notwendigen Informationen zur vollständigen Definition der Ausbildungen aller Personen und der Ressourcen- sowie Kapazitätsverfügbarkeit in den betroffenen Ausbildungsstätten, als auch notwendige Informationen für die Randbedingungen, die dem beschriebenen Optimierungsproblem zugrunde liegen.

Algorithmus 5: Optimierungsverfahren basierend auf Algorithmus 1

Parameter: $t_{max} \in \mathbb{N}$

$p_m \leftarrow \in [0,1]$

$p_c \leftarrow \in [0,1]$

$\mu \leftarrow \in \mathbb{N}$

$\lambda \leftarrow \in \mathbb{N}$

$n\_monate \leftarrow$ Anzahl der Monate für die geplant werden soll

Input : $D \leftarrow$ Problemdefinition als Datenstruktur siehe Anhang A.5

Output : Bestes Individuum I

1 $M \leftarrow$ bestimme notwendige Informationen für Berechnung aus D
2 $t \leftarrow 0$
3 $P(t=0) \leftarrow$ Initialisiere Startpopulation gemäß Algorithmus 6
4 $P(t=0) \leftarrow$ Bewerte I in P gemäß Algorithmus 4 mit M
5 **while** $t < t_{max}$ **do**
6 $E \leftarrow$ Selektiere Eltern Turnierselektion aus $P(t)$
7 $N \leftarrow$ erzeuge λ Nachkommen durch Rekombination aus E mit p_c nach Algorithmus 2
8 $N' \leftarrow$ Mutiere Individuen in N mit p_m und M gemäß Algoritmus 3
9 $N'' \leftarrow$ Bewerte alle I in N' gemäß Algorithmus 4 mit M
10 $P(t+1) \leftarrow$ Selektiere μ Individuen aus $P(t)$ und N' gemäß Funktion (NSGA-II) in Anhang A.6
11 $t \leftarrow t+1$
12 **end**
13 **return** Bestes Individuum in $P(t)$

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die Berechnungsinformationen M, die in mehreren Routinen des Verfahrens verwendet werden, enthalten unter anderem personenspezifische Daten, wie Curricula und die jeweils zulässigen Kombinationen aus Dienstposten und Ausbildungsstellen. Wenn eine Zuweisung einer Person zu einem bestimmten Zeitpunkt verwendet wird, ist deren Struktur, wie in Abschnitt 5.4.1 beschrieben. Die Indizes einer Zuweisung beginnen mit 0, wobei der Index 0 dem gewählten Dienstposten, der Index 1 der Ausbildungsstelle und der Index 2 der Information über die Veränderbarkeit der Zuweisung entspricht.
Die Algorithmen 7 und 8 finden sowohl bei der Erzeugung von zufälligen Startlösungen, als auch als Reparaturalgorithmen nach der Rekombination und Mutation von Individuen Anwendung. Ihr grundlegendes Vorgehen ist mehr oder weniger identisch, sie versuchen jeweils an noch freien Stellen Dienstposten, respektive Ausbildungsstellen einzufügen, sofern welche verfügbar sind. Sie sind „greedy“, da sie jeweils danach trachten, für die aktuell betrachtete Position im Ausbildungsplan ein Optimum zu finden. Die zufällige Sortierung der Personenreihenfolge sichert den nicht deterministischen Charakter der Routinen, so „schnappt“ nicht immer eine Person einer anderen Person eine, von beiden benötigte, Ressource weg.

Algorithmus 6: Initialisierung der Startpopulation

Parameter: *n_monate* ← Anzahl der Monate für die geplant werden soll

Parameter: μ ← Populationsgröße

Input : M ← Berechnungsinformationen des GA

1 $P \leftarrow \{\ \}$ `/* initialisiere leere Population */`
2 $I \leftarrow$ erzeuge leeres Individuum mit M und n_{monate}
3 **while** $|P| < \mu$ **do**
4 $\quad I' \leftarrow$ kopiere I
5 $\quad I'' \leftarrow$ befülle I' mit Dienstposten gemäß Algorithmus 7
6 $\quad I''' \leftarrow$ befülle I'' mit Ausbildungsstellen gemäß Algorithmus 8
7 $\quad P \leftarrow P \cup I'''$
8 **end**
9 **return** P

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithmus 7: Greedy-Algorithmus - Befüllen eines Individuums mit Dienstposten

Input : $I \leftarrow$ Individuum

Input : M $\leftarrow$ Berechnungsinformationen des GA

Output : $I' \leftarrow$ befülltes Individuum

```
1  person_indices = [1...length(I)]                    /* Menge aller Zeilenindizes */
2  person_indices ← sortiere person_indices zufällig
3  month_indices = [1...length(I[0])]                  /* Menge aller Spaltenindizes */
4  forall p in person_indices do
5      forall m in 0..length(I[p]) do
6          asg ← I[p, m]                               /* Zuweisung an der Stelle [p,m] */
           /* Wenn kein Dienstposten vorhanden und Zuweisung veränderbar          */
7          if asg[0] ≤ 0 && asg[2] == false then
8              A ← Menge an Dienstposten die für Person p erlaubt sind aus M
9              B ← I[:, m][0]        /* Menge an verwendeten Dienstposten in Monat m */
10             C ← A \ B                               /* Mögliche freie Dienstposten */
11             if length(C) > 0 then
12                 D ← sortiere C, nach zugehörigen Abteilungen und Krankenhäusern,
                    beginnend mit den entsprechenden benachbarten Zuweisungen
13                 I[p, m] ← (D[0], 0, false)    /* Weise Dienstposten an Stelle [p,m] zu
                    */
14             end
15         end
16     end
17 end
18 return I
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Algorithmus 8: Greedy-Algorithmus - Befüllen eines Individuums mit Ausbildungsstellen

Input : $I \leftarrow$ Individuum

Input : M $\leftarrow$ Berechnungsinformationen des GA

Output : $I \leftarrow$ befülltes Individuum

```
1  person_indices = [1...length(I)]                    /* Menge aller Zeilenindizes */
2  person_indices ← sortiere person_indices zufällig
3  month_indices = [1...length(I[0])]                  /* Menge aller Spaltenindizes */
4  forall p in person_indices do
5      forall m in 0..length(I[p]) do
6          asg ← I[p, m]                               /* Zuweisung an der Stelle [p,m] */
             /* Wenn Dienstposten vorhanden, keine Ausbildungsstelle vorhanden und
            Zuweisung veränderbar */
7          if asg[0] > 0 && asg[1] ≤ 0 && asg[2] == false then
8              A ← Menge an Ausbildungsstellen die für Person p und Dienstposten asg[0]
                 erlaubt sind aus M
9              B ← I[:, m][1]        /* Menge an Ausbildungsstellen belegt in Monat m */
10             C ← A \ B                          /* Mögliche freie Ausbildungsstellen */
11             if length(C) > 0 then
12                 D ← sortiere C zufällig
13                 I[p, m][1] ← d[0]        /* Weise Ausbildungsstelle an Stelle [p,m] bei
                    gleichbleibendem Dienstposten zu */
14             end
15         end
16     end
17 end
18 return I
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Algorithmus 7 weist eine verhältnismäßig hohe Laufzeit mit $\mathcal{O}(m \cdot n \cdot d \log(d))$ auf, wobei m der Anzahl der Monate, n der Anzahl der Personen und d der Anzahl der freien Dienstposten entspricht. Eine einfache Verbesserung wäre die monatsweise parallele, Ausführung der Routine, was zu einer Reduktion der Laufzeit auf $\mathcal{O}(n \cdot d \log(d))$ führen könnte. Vernachlässigt wurden in dieser Betrachtung die vielen Operationen mit konstanter Laufzeit, wobei diese in der Realität einen beträchtlichen Anteil der Laufzeit einnehmen, insbesondere, da m, n und d in ihrer Größe i.d.R stark beschränkt sind (in etwa $m < 100, n < 200, d < 50$). Die Laufzeit von 8 verhält sich ähnlich.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Kapitel 6

Evaluierung der entwickelten Methode

Zur Entwicklung und Bewertung des Verfahrens liegt ein Datensatz vor, der 68 Ärzt*innen umfasst, die die sich zum Planungszeitpunkt bereits inmitten ihrer allgemeinärztlichen Ausbildung befinden, oder ihre Ausbildung in näherer Zukunft beginnen. Dieser Datensatz stellt eine Problembeschreibung mit realistischer Problemgröße für einen Krankenanstaltenverbund in Österreich dar, der sich an den tatsächlichen Gegebenheiten in einem solchen Krankenanstaltenverbund orientiert. Der Datensatz wurde von einer Expertin[6] aufbereitet. Da in diesem Datensatz keine Krankenhäuser als Kooperationspartner vermerkt sind und kein anderer Datensatz vorliegt, wird in den folgenden Ausführungen auf die Berücksichtigung der entsprechenden Metrik verzichtet, wobei die Berücksichtigung in der Optimierung analog zu allen anderen Metriken geschehen würde. Abgesehen von den fehlenden Krankenanstalten mit Kooperationsvertrag bildet der vorliegende Datensatz die volle Komplexität eines realen Planungsproblems ab und unterliegt keinerlei Einschränkungen.

Für das implementierte Verfahren (Algorithmus 5) wurden, mit dem vorliegenden Datensatz, Versuchsreihen durchgeführt, um dessen Leistungsfähigkeit beurteilen zu können. Alle gezeigten Ergebnisse wurden auf einem Intel NUC mit 16 GB Arbeitsspeicher und Intel Core i7-1165G7 Prozessor mit vier Kernen und einer maximalen Frequenz von 4,70 GHz berechnet, wobei das entwickelte Verfahren aktuell lediglich in der Lage ist

[6] siehe Fußnote 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

einen Prozessorkern zu nutzen (siehe Abschnitt 7.7).

Das Parameterset

Zur Abstimmung der Zielfunktion wurden Parameter identifiziert, sodass die erzeugten Ausbildungspläne die, in Abschnitt 2.3 identifizierten, Ziele erfüllen. Die Bestimmung erfolgte hierbei anhand der eingangs definierten Beschreibung der Optimierungsziele sowie unter Berücksichtigung der üblicherweise auftretenden Größenordnungen der einzelnen Metriken. Das Parameterset wurde so gewählt, dass eine möglichst ausgeglichene Gewichtung zwischen den einzelnen Zielfunktionswerten und den damit verbundenen Inkrementen bzw. Dekrementen bei deren Verbesserung/Verschlechterung für den vorliegenden Datensatz besteht. Diese ausgewogene Gewichtung wurde im Zuge zahlreicher Optimierungsversuche experimentell angepasst, sodass eine möglichst ausgewogene Optimierung der einzelnen Zielgrößen stattfindet. Dieses Parameterset ist in Tabelle 6.2 angeführt. Die Gewichtung der Monate in einem externen Krankenhaus wurde in dieser Parameterkonfiguration vernachlässigt, da diese Zielfunktion für den vorliegenden Datensatz zwangsläufig einen Wert von 0 liefert. Alle nachfolgend abgebildeten Planungsergebnisse werden mithilfe dieses Parametersets bewertet und verglichen. Dabei gilt zu berücksichtigen, dass diese Konfiguration verwendet wird, um die finalen Lösungen der Optimierungsläufe zu erzeugen, zu bewerten und zu vergleichen. Im weiteren Verlauf dieses Kapitels wird gezeigt, dass eine anfängliche Aufweichung eines spezifischen Gewichts zu besseren Lösungen führen kann. Die beschriebene Parameterkonfiguration stellt lediglich eine von vielen möglichen Kombinationen dar, mit denen der Algorithmus für die vorliegenden Testdaten zufriedenstellende Lösungen liefert. Für Probleme anderer Gestalt oder andere Anforderungsprofile müssen die Parameter klarerweise angepasst werden. Zur Berücksichtigung konkreter Präferenzen, hinsichtlich der Gewichtung verschiedener Zielgrößen aus einer Management-Perspektive, kann das Parameterset entsprechend angepasst werden. Der Parameter k kann dabei verwendet werden, um zu steuern, wie stark zwischen unzulässigen/unvollständigen und zulässigen/vollständigen Lösungen in der Bewertung unterschieden wird. Im vorliegenden Fall wird ein verhältnismäßig starkes „Gefälle“ in Richtung des zulässigen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Lösungsgebiets eingeführt, da ein Wert von $k = 100$ bedeutet, dass sich der Wert der Straffunktion für eine einzelne Person, deren Ausbildung nicht abgeschlossen ist, zwischen 100 und 300 bewegt.

#	Name der Metrik	Zielfunktion	Gewicht	Wert	Gleichung
1	Stehmonate	$f_1(\vec{x})$	w_1	**200**	5.11
2	Konsekutive Stehmonate	$f_2(\vec{x})$	w_2	**10**	5.12
3	Ortswechsel - Krankenhaus	$f_3(\vec{x})$	w_3	**500**	5.13
4	Ortswechsel - Abteilung	$f_4(\vec{x})$	w_4	**50**	5.14
5	Einmonatige Zuweisungen	$f_5(\vec{x})$	w_5	**500**	5.15
6	Monate Extern	$f_6(\vec{x})$	w_6	**0**	5.16
7	Präferenzen	$f_7(\vec{x})$	w_7	**50**	5.17
8	Straffunktion	$P(\vec{x})$	k	**100**	5.29

Tabelle 6.2: Wahl einer Parameterkonfiguration

Die gewählten Hyperparameter für das Verfahren und die vorliegende Problemstellung sind in Tabelle 6.4 aufgelistet. Dabei ist lediglich die Größe des Planungshorizonts

μ	Populationsgröße	10
λ	Nachkommensgröße	μ
p_m	Mutationsrate	1
p_c	Rekombinationsrate	0,2
n_{monate}	Planungshorizont	45

Tabelle 6.4: Gewählte Parameter für das Verfahren

maßgeblich vom konkreten Planungsproblem abhängig, die restlichen Hyperparameter wurden auf das entwickelte Verfahren abgestimmt. Die Populationsgröße μ wurde klein gewählt, um die Laufzeit für eine Generation möglichst gering zu halten und dabei trotzdem die Vorteile des verwendeten Selektionsoperators nutzen zu können. Kleinere

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

μ (< 10) würden zur Einschränkung der Diversität in der Population führen.[7] Davon abgesehen konnte im Zuge der Entwicklung festgestellt werden, dass kleinere Populationen anfänglich zwar zu einer schnelleren Verbesserung der Zielfunktion (gemessen an der Anzahl der Funktionsauswertungen), jedoch ebenfalls zu einer frühzeitigen Konvergenz in einem suboptimalen Bereich führen und die Optimierung damit „stecken bleibt". Kleine Populationen mit weniger Individuen als der Anzahl an Optimierungszielen, schränken davon abgesehen auch die Möglichkeit der Ausbildung einer Pareto-Front ein. Größere Populationen hingegen, bewirken eine Verlangsamung des Verfahrens ohne nennenswerte Verbesserungen hinsichtlich des Konvergenzverhaltens.[8] Die Größe des Planungshorizontes n_{monate} wurde experimentell bestimmt, wobei ein großzügiger Aufschlag auf die Anzahl der notwendigen Ausbildungsmonate gewählt wurde. Der Grund für diesen Aufschlag liegt in der Notwendigkeit eines gewissen „zeitlichen Spielraums", den das Verfahren benötigt, um vollständige Ausbildungen zu konstruieren. Versuche haben gezeigt, dass ein Überschuss von etwa 10 bis 15 Monaten das Verfahren zu Beginn befähigt, schnell zulässige und vollständige Lösungen zu finden.

Die Rekombinationsrate wurde in Abschnitt 4.4.2.2 experimentell bestimmt und die Mutationsrate wird zu 1 gesetzt, sodass jedes Individuum zwangsläufig einer Veränderung unterworfen wird.

6.1 Ergebnisse des Verfahrens

Das Verfahren findet für das vorliegende Problem in der Regel innerhalb der ersten fünf bis zehn Generationen mit den Parametern aus Tabelle 6.4 die erste zulässige und voll-

[7] Bei der Berechnung der Crowding Distance einer Front wird den größten Ausreißern bezüglich jeder Teilzielfunktion eine Distanz von ∞ zugeordnet. Ist die Population klein im Verhältnis zur Anzahl der Zielfunktionen, so ist die Wahrscheinlichkeit groß, dass jedem Individuum in der Front eine Distanz von ∞ zugeordnet wird und der anschließende Vergleich der Distanzen nicht den gewünschten Effekt der Erhaltung der Diversität liefert.

[8] Im Falle einer Verbesserung der Laufzeiteigenschaften des Verfahrens könnte sich eine Vergrößerung der verwendeten Population als vielversprechend herausstellen, aktuell wird jedoch der damit verbundene Mehraufwand im Hinblick auf die Laufzeit als nicht vertretbar angesehen.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

ständige Lösung. In Abbildung 6.2 sind jeweils die besten Lösungen der ersten acht Generationen einer beispielhaften Ausführung des Verfahrens mit den angegebenen Parametern dargestellt. Zur Erläuterung der Grafik ist das erste Bild dieser Darstellung

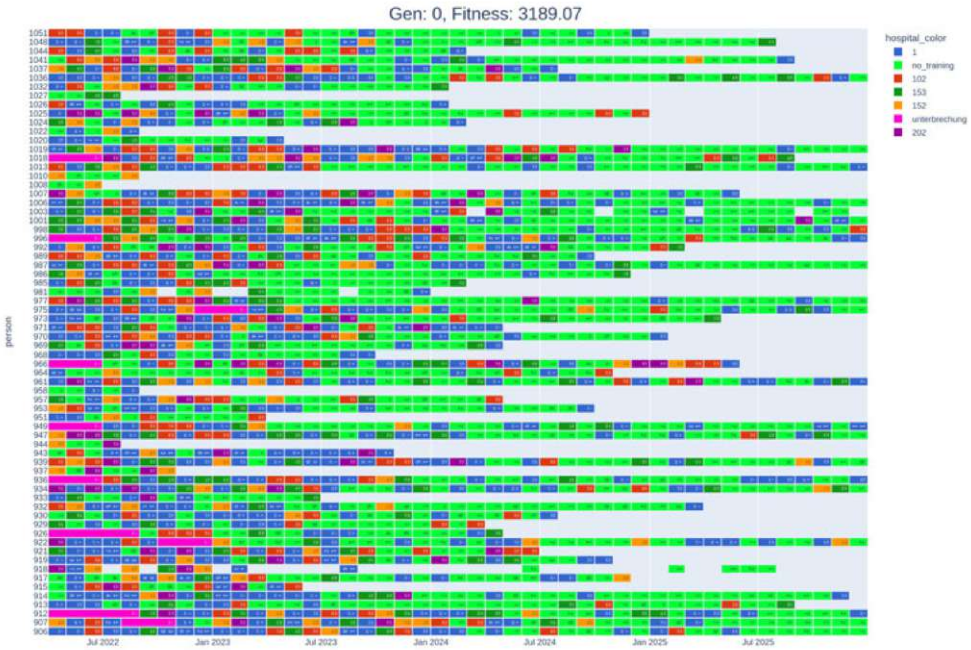


Abbildung 6.1: Beste Lösung nach zufälliger Initialisierung der Population - Gen 0

vergrößert in Abbildung 6.1 ersichtlich. Dabei stehen die Farben der eingezeichneten Boxen für das jeweilige Krankenhaus, in dem eine Person im entsprechenden Monat beschäftigt ist. Eine Ausnahme davon bilden die hellgrünen und pinken Kästchen, diese symbolisieren Monate ohne Ausbildung respektive Abwesenheitszeiten. Die Grafiken sind interaktiv gestaltet, sodass im Anwendungsfall alle relevanten Informationen zu einer Zuweisung auf Knopfdruck aufgerufen werden können.

Die Grafiken in Abbildung 6.2 stellen ausgewertete und bereinigte Individuen dar. In den ersten sechs Abbildungen (Gen 0 - Gen 5) liegen noch unvollständige Ausbildungen bei einzelnen Personen vor, weshalb der Straffunktionswert größer 0 ist. Die letzte Zeile der Darstellung beinhaltet zwei vollständig zulässige Lösungen, die jedoch noch weit entfernt von einem Optimum sind.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Abbildung 6.2: Anfangsphase des Verfahrens, die besten Lösungen der ersten 8 Generationen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Abbildung 6.3: Optimierungslauf über 10000 Generationen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Abbildung 6.4: Beste Lösung nach 40000 Generationen mit aufgeweichter Zielfunktion f_2

Abbildung 6.5: Beste Lösung nach 50000 Generationen, mit eingeführter f_2 nach 40000 Generationen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Eine Lösung, die innerhalb von 50000 Generationen gefunden werden konnte, ist in Abbildung 6.4 dargestellt. Im Vergleich zu den anfänglichen Lösungen in Abbildung 6.3 ist ersichtlich, dass der Algorithmus zusammenhängende „Blöcke“ aus Zuweisungen erzeugt, indem diese nach Krankenhäusern und Abteilungen gruppiert und zugewiesene Module variiert werden.

Die zugehörige Entwicklung der besten Fitnesswerte in der Population im „Zeitverlauf“ ist in Abbildung 6.7 dargestellt. Die Entwicklung der einzelnen Zielfunktionswerte ist in Abbildung 6.6 dargestellt. Die abgebildeten Werte sind, abgesehen von den konsekutiven Stehmonaten auf einen Wertebereich von [0,1] normalisiert, wie in Kapitel 5 beschrieben.

Bei der Bestimmung geeigneter Parameter für die Zielfunktion (siehe Tabelle 6.2) konnte festgestellt werden, dass eine anfängliche Aufweichung der Metrik für aufeinanderfolgende Stehmonate ($f_2(\vec{x})$ in Gleichung 5.12) im Optimierungsverlauf zu einer wesentlich schnelleren Konvergenz und besseren Zielfunktionswerten bei lediglich geringfügig höheren Funktionswerten dieser einen Zielfunktion erzielt werden können. Eine nachgelagerte Berücksichtigung von $f_2(\vec{x})$, indem das ursprünglich vorgesehene Gewicht w_2 verwendet wird, führt zu einer vergleichsweise schnellen Minimierung von f_2 ohne, dass die anderen Zielfunktionswerte dabei maßgeblich verschlechtert werden.

Gemäß dieser Vorgehensweise wurde die präsentierte Lösung erzeugt. Dazu wurde das Gewicht w_2 für die Metrik für konsekutive Stehmonate ($f_2(\vec{x})$ in Gleichung 5.12) zu 0 gesetzt und die Metrik bei der Optimierung im ersten Schritt ignoriert. Nach 40000 Generationen wurde w_2 wieder gemäß Tabelle 6.2 gewählt und der Optimierungslauf fortgesetzt. Die Rückkehr zu einem niedrigen Niveau der gesamten Zielfunktion und von f_2 geschieht nach dieser Einführung vergleichsweise schnell.

In Abbildung 6.7 ist der Zielfunktionswert ohne Berücksichtigung (blaue Linie) und mit Berücksichtigung konsekutiver Stehmonate (rote Linie) dargestellt.

Somit empfiehlt sich eine Parameteranpassung der Zielfunktion im zeitlichen Verlauf der Optimierung, also ein anfängliches Tolerieren von aufeinanderfolgenden Stehmonaten und eine zeitabhängige Zunahme der Gewichtung dieser Metrik. Eine frühzeitige

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Einschränkung dieser Metrik erschwert die Exploration des Suchraums.

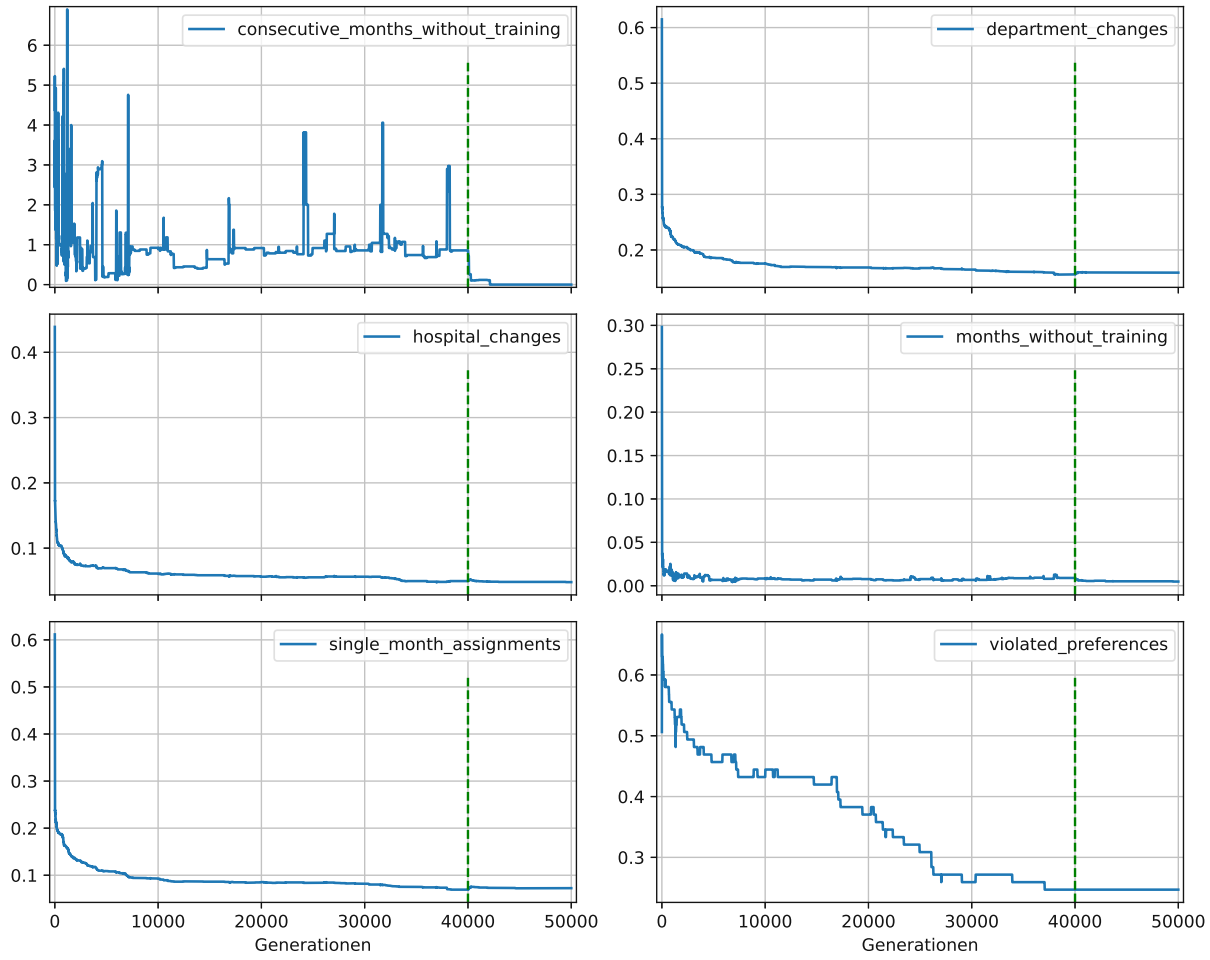


Abbildung 6.6: Einzelne normalisierte Zielfunktionswerte im „Zeitverlauf“

Aktuell dauert die vollständige Erzeugung einer neuen Generation (eine Iteration des Algorithmus) auf dem zuvor beschriebenen Rechner im Mittel etwa **5,2 Sekunden**. Demzufolge entspricht eine Ausführung mit **10000 Generationen** einer Dauer von etwa 52.000 Sekunden, also ca. **14,5 Stunden**.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

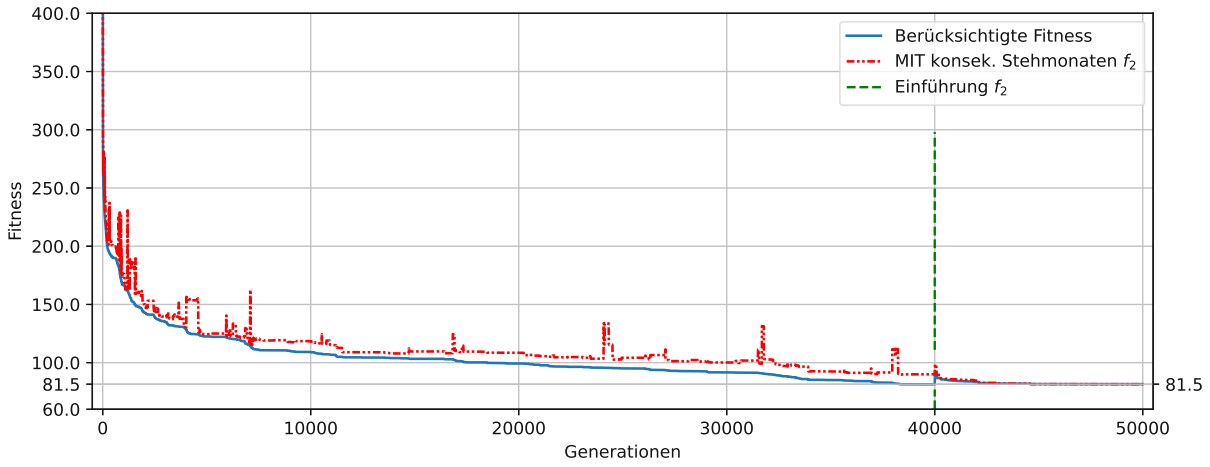


Abbildung 6.7: Zielfunktionswert im „Zeitverlauf“

6.2 Vergleich mit bestehender Lösung

Zum Vergleich des entwickelten Verfahrens mit einer bestehenden Lösung, wurde ein, von einer Expertin[9] erstellter, Ausbildungsplan herangezogen. Zu beachten gilt, dass bei der Erstellung dieses Plans die Präferenzen der jeweiligen Personen in keiner Weise berücksichtigt wurden, da im vorliegenden Datensatz keine Präferenzen vermerkt waren (Vereinfachung des Planungsproblems). Um das Planungsproblem realistischer zu gestalten, wurde, basierend auf den Planungsergebnissen der Expertin, der ursprüngliche Datensatz so angepasst, dass die tatsächlich verplanten Wahlfächer den vermerkten Präferenzen entsprechen. Für die automatische Planung mithilfe des entwickelten Verfahrens wurde der adaptierte, mit Präferenzen versehene Datensatz verwendet und das Planungsproblem somit realistischer, aber auch komplizierter gestaltet. Zur Bewertung wurde ebenfalls der adaptierte Datensatz verwendet, um die Vergleichbarkeit der Pläne sicherzustellen.

Daher verfügt die vorliegende manuelle Lösung über keinerlei verletzte Präferenzen (100 % berücksichtigte Präferenzen), was im Realfall meist nur möglich ist, indem überproportional viele zusätzliche Ortswechsel oder Stehmonate in Kauf genommen werden.

[9] siehe Fußnote 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Der manuell erzeugte Plan wurde bewusst als eine (vermutlich) so gut wie optimale Lösung konzipiert, sodass dieser Plan als ein schwer zu übertreffendes „Optimum“ und damit als Benchmark betrachtet werden kann.

Abbildung 6.8: Manuell erzeugter und optimierter Plan

Das entwickelte Verfahren übertrifft das manuell erzeugte Planungsergebnis hinsichtlich der Stehmonate und konsekutiven Stehmonate. Im gezeigten Ergebnis des Optimierungsverfahrens werden 75 % aller Präferenzen erfüllt, im Vergleich zu den nicht repräsentativen 100 % der manuellen Lösung. Bezüglich der Ortswechsel weist das entwickelte Verfahren noch das größte Verbesserungspotenzial auf. Die Anzahl Krankenhauswechsel bewegt sich zwar in einem vergleichbar niedrigen Bereich, wie den entsprechenden Grafiken 6.8 und 6.4 bzw. 6.5 entnommen werden kann. In Sachen Abteilungswechsel und insbesondere einmonatige Zuweisungen (f_4 und f_5) haben die automatisch erzeugten Pläne noch einiges an Aufholbedarf. In Abbildung 6.9 ist die Annäherung der einzelnen Zielfunktionswerte des Optimierers mit der manuell erzeugten Lösung gegen-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

übergestellt, wobei die horizontalen, roten Linien den Gütewert der manuellen Lösung darstellen. In Tabelle 6.6 sind die einzelnen Werte, die in die Zielfunktionen eingehen, dargestellt und für die manuelle Lösung, die Startlösung des Verfahrens und die beste Lösung des Verfahrens gegenübergestellt. Die Startlösung ist hierbei das beste, zufällig generierte Individuum aus Algorithmus 6. Diese Lösung erfüllt bereits alle Randbedingungen, stellt jedoch im Allgemeinen noch keinen vollständigen Ausbildungsplan dar. In der letzten Zeile der Tabelle ist die Fitness der Lösungen abgebildet. Die aufgelisteten Werte stellen, abgesehen von der Kennzahl für konsekutive Stehmonate, die tatsächlich am Plan abzählbaren Werte dar.

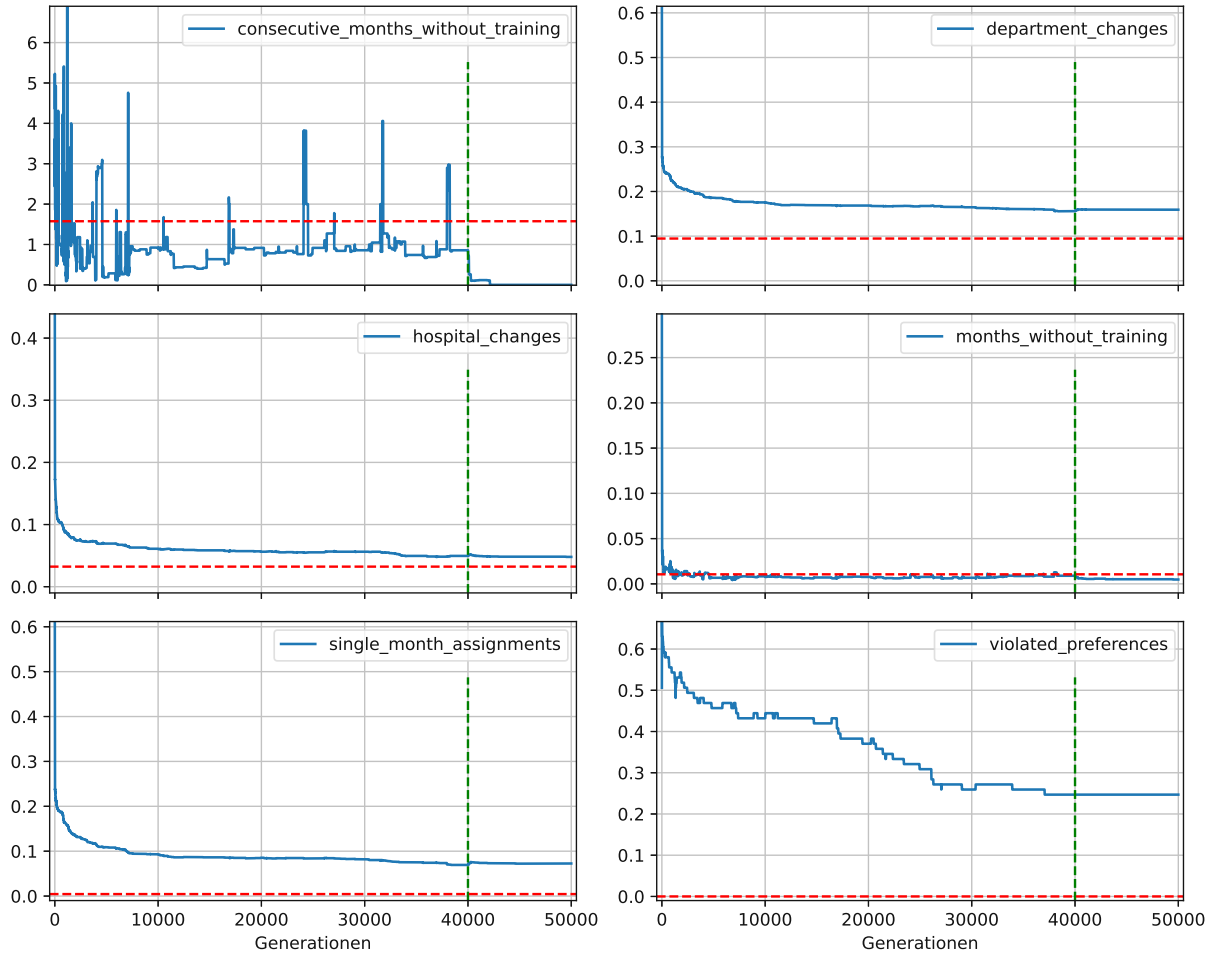


Abbildung 6.9: Normalisierte Zielfunktionswerte: Verfahren vs. manuelle Lösung

Der, mithilfe des entwickelten Verfahrens, erzeugte Plan aus Abbildung 6.4 wurde anschließend von derselben Expertin subjektiv bewertet. Die Expertin beurteilt die Opti-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Metrik	Manuelle Lösung	Entwickeltes Verfahren	Beste Startlösung
Stehmonate	33	15	933
Konsekutive Stehmonate	1,5757	0,0	4,3815
Ortswechsel - Krankenhaus	101	150	1373
Ortswechsel - Abteilung	296	498	1923
Einmonatige Zuweisungen	14	227	1913
Monate Extern	0	0	0
Präferenzen verletzt	0	20	41
Fitness gesamt	40,98	81,53	3012,79

Tabelle 6.6: Ergebnisse Vergleich manuell vs. automatisch in absoluten Zahlen

mierung der Stehmonate als „sehr gut“. In der Anzahl der einmonatigen Zuweisungen sieht sie noch das größte Verbesserungspotenzial, die Anzahl der Krankenhauswechsel beurteilt sie als „gut“ mit weiterem Potenzial zur Verbesserung. Der Erfüllungsgrad der Präferenzen von 75 % kann laut ihren Angaben ebenfalls als „gut“ bezeichnet werden.

Aktuell ist das entwickelte Verfahren noch nicht in der Lage, die Planungsqualität von erfahrenem Planungspersonal zu erreichen oder gar zu übertreffen. Allerdings liefert es zulässige, praxistaugliche Ergebnisse, die von erfahrenem Personal als Ausgangspunkt für weitere Verbesserungen herangezogen werden können.
Abschließend muss erwähnt werden, dass der Vergleich zwischen manuell erzeugter und automatisch generierter Lösung bewusst mit einem Startvorteil zugunsten der manuellen Lösung gestaltet wurde, um eine möglichst gute Lösung zum Vergleich zu erhalten, die eine Zielfunktion vollständig optimiert. Da nicht sicher ist, ob menschliches Planungspersonal die manuell erzeugte Lösung auch ohne die zuvor erläuterte Vereinfachung gefunden hätte, kann davon ausgegangen werden, dass der Unterschied zwischen manuell erzeugten Lösungen und maschinell gefundenen Lösungen in der Realität kleiner ist als im dargestellten Vergleich.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Kapitel 7

Resultate, Diskussion und Ausblick

7.1 Resultate der angewendeten Methoden

Die Ergebnisse des implementierten Verfahrens wurden im vorigen Kapitel ausführlich beschrieben und diskutiert, weshalb an dieser Stelle auf weitere Ausführungen verzichtet wird. Es konnte ein Verfahren implementiert werden, das in der Lage ist, Ausbildungspläne zu erzeugen und dabei die identifizierten Ziele zu berücksichtigen, sowie die damit verbundenen Zielfunktionen zu minimieren. Es konnte gezeigt werden, dass ein GA für die Mehrzieloptimierung mit speziellem Selektionsverfahren (NSGA-II) bessere Ergebnisse liefert als ein klassischer elitärer oder nicht elitärer Algorithmus mit rangbasierter Selektion. Überdies wurde gezeigt, dass sich eine zielgerichtete Bestrafung unzulässiger Lösungen in Kombination mit geeigneten Reparaturalgorithmen für das vorliegende Problem ebenfalls als sinnvoll erweist. Es wurde festgestellt, dass niedrige Rekombinationswahrscheinlichkeiten p_c im vorliegenden Fall zu verhältnismäßig schneller und gleichmäßiger Konvergenz führen. Die Wichtigkeit der Mutation, als treibender Operator für den evolutionären Fortschritt bei MOPs, konnte analog zu den Aussagen in [103] dargelegt werden.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

7.2 Einschränkungen der Ansätze und Ergebnisse

Die, zu Beginn von Kapitel 5, beschriebenen Vereinfachungen stellen Einschränkungen des Verfahrens dar. Somit ist das Verfahren nicht in der Lage Dienstposten und Ausbildungsstellen zu zerteilen und auf mehrere Personen aufzuteilen und kann das damit verbundene, als sehr gering eingeschätzte, Optimierungspotenzial nicht ausschöpfen. Das tut jedoch der prinzipiellen Funktion des Verfahrens keinen Abbruch, da der Großteil der zu verplanenden Personen vollzeitbeschäftigt ist und somit Dienstposten und Ausbildungsstellen ohnehin meist von einer Person vollständig besetzt werden.
Die theoretisch zeitabhängige Verfügbarkeit von Ausbildungsstellen und Dienstposten wurde vernachlässigt. Demzufolge kann das Verfahren zeitbezogene Einschränkungen von Ressourcen in der Planung nicht berücksichtigen. Da jedoch Ausbildungsstellen üblicherweise fortlaufend rezertifiziert werden, stellt diese getroffene Annahme keine wesentliche Einschränkung der Anwendbarkeit des Verfahrens in der Realität dar. Die Implementierung von Routinen zur zusätzlichen Berücksichtigung der zeitlichen Verfügbarkeit von Ressourcen wäre, sofern erwünscht, mit geringem Implementierungsaufwand verbunden. Im Rahmen dieser Arbeit wurde trotzdem darauf verzichtet, da die vorliegende Informationsgrundlage nicht ausreicht, um die Gegebenheiten abzubilden.
Darüber hinaus werden etwaige zeitliche Obergrenzen, was die Verweildauer in einer Abteilung angeht, nicht berücksichtigt. Hierfür lag abermals keine ausreichende Informationsgrundlage vor, um eine umfassende Berücksichtigung dieses Kriteriums vorzunehmen. Die nachträgliche Implementierung ist jedoch mit geringem Aufwand verbunden und kann vorgenommen werden, sobald die notwendigen Informationen dafür vorliegen.
Trotz aller erwähnten Einschränkungen ist das Verfahren im realen Planungsbetrieb anwendbar und liefert zulässige und praxistaugliche Ergebnisse.
Die Verallgemeinerung des Verfahrens auf andere (insbesondere ausländische) Ausbildungsordnungen ist nicht uneingeschränkt möglich. Zur Anpassung auf andere, aber vergleichbare Ausbildungen, wie z.B. in Deutschland, ist insbesondere eine Anpassung der Bewertungsfunktion, die Auskunft über die Zulässigkeit einer Lösung gibt, erfor-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

derlich. Der damit verbundene Aufwand kann in Relation zum gesamten Implementierungsaufwand des Verfahrens als vergleichsweise gering angesehen werden. Das grundlegende Verfahren kann dabei bestehen bleiben.

Aktuell ist das entwickelte Verfahren ausschließlich für die in Österreich gängige ÄAO 2015 uneingeschränkt einsetzbar.

Die Laufzeiteffizienz stellt derzeit die größte Schwäche des Verfahrens dar (siehe 6), wobei in den nächsten Abschnitten Handlungsempfehlungen zur Effizienzsteigerung angeführt sind, um dieser Problematik entgegenzuwirken. Davon abgesehen findet das Verfahren verhältnismäßig schnell brauchbare Lösungen zu Beginn des Optimierungsverlaufs (siehe Abbildung 6.7).

7.3 Resultate in Bezug auf die Problemstellung

Mithilfe des entwickelten Verfahrens kann der personelle Aufwand (**P 1**) in der Einsatzplanung reduziert werden, da die automatische Erstellung von Ausbildungsplänen keinerlei menschliche Interaktion benötigt. Im Rahmen dieser Arbeit wurde sowohl eine Methodik zur Bewertung und Validierung, als auch zur interaktiven Visualisierung von Ausbildungsplänen entwickelt. Damit können verschiedene Pläne sowohl optisch als auch mathematisch miteinander verglichen werden (siehe **P 2**).
Durch das entwickelte Zielgrößensystem und die Möglichkeit der Bewertung von erzeugten Ausbildungsplänen, ist die Standardisierung des planerischen Vorgehens möglich und die Erfüllung einzelner Teilziele, unabhängig von menschlichen Präferenzen und Tendenzen, messbar. Die einfache Anpassbarkeit der Gewichtung von Teilzielen im Zielsystem ermöglicht die Verfolgung verschiedenster Zielsetzungen bei der Planung (siehe **P 3**).
Im Zuge dieser Arbeit wurde das zugrundeliegende mathematische Optimierungsproblem charakterisiert, das als Grundlage für weitere Entwicklungen in diesem Umfeld dienen kann und soll. Die Implementierung des problemspezifischen Verfahrens unter Berücksichtigung der zuvor identifizierten Zielgrößen und der entsprechenden Mög-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

lichkeit zur Prioritätensetzung wurde erfolgreich durchgeführt (siehe **P 4** und **P 5**).

7.4 Diskussion der Ergebnisse

Das entwickelte Verfahren weist aktuell noch Schwächen bei der Minimierung der einmonatigen Zuweisungen (Gleichung 5.15) auf, da eine Verbesserung der zugehörigen Metrik häufig eine überproportionale Verschlechterung der anderen Zielfunktionswerte mit sich bringt. Die Verbesserung der existierenden Operatoren im Sinne dieser Zielgrößen wäre somit angemessen. Zur besseren Minimierung der Abteilungswechsel wäre ein zusätzlicher Mutationsoperator zweckmäßig. Die Berücksichtigung dieses Ziels ist jedoch vermutlich mithilfe eines heuristischen Verfahrens wesentlich einfacher und kann so mithilfe eines Eröffnungsverfahrens behandelt werden.
Generell sind die erzeugten Ergebnisse, abgesehen von den Zielfunktionen (5.12 und 5.11), schlechter als manuell erzeugte Pläne und weisen somit noch Optimierungspotenzial auf. Eine starke Berücksichtigung von Zielfunktion 5.12 führt zu einer Verlangsamung des Optimierungsfortschritts und einer „Zerklüftung“ des Suchraums, da der Spielraum des Algorithmus für Verbesserung der Lösung durch Einführen von Stehmonaten stark eingeschränkt wird (Stehmonate, insbesondere konsekutive, werden so kaum zugelassen). Wie in Kapitel 6 gezeigt wurde, kann durch eine spätere Einführung dieser Zielfunktion ein besseres Gesamtergebnis bei ebenfalls guten Ergebnissen hinsichtlich f_2 gefunden werden.
Das Verfahren weist aktuell noch großes Potenzial hinsichtlich seiner Laufzeiteigenschaften auf. Dieses Potenzial ist einerseits der Wahl der Programmiersprache und andererseits der verbesserungswürdigen Laufzeiteigenschaften der einzelnen Algorithmen geschuldet. Da das Hauptaugenmerk bei der Umsetzung dieser Arbeit allerdings vorrangig auf der Implementierung eines funktionsfähigen Prototyps lag, war die Wahl der Programmiersprache zur Entwicklung rückblickend jedenfalls die richtige. Python ermöglicht verhältnismäßig kurze Implementierungszeiten, was insbesondere zum schnellen Testen von neuen Ansätzen von großer Wichtigkeit ist. Überdies verfügt die Sprache über ein umfangreiches Ökosystem an nützlichen Bibliotheken zur Vereinfachung der

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Implementierungsarbeit.

Die Laufzeiteffizienz kann in erster Linie durch Überarbeitung der einzelnen Algorithmen verbessert und in einem weiteren Schritt durch Einsatz einer effizienteren Programmiersprache gesteigert werden. Genauere Hinweise dazu werden in Abschnitt 7.7 gegeben.

7.5 Resultate in Bezug auf die Forschungsfragen

Q 1.1 Welche Zielgrößen muss die, dem Optimierungsproblem zugrundeliegende, Zielfunktion berücksichtigen, und welche Gestalt muss diese Zielfunktion aufweisen, um eine umfassende Bewertung eines Lösungskandidaten zu ermöglichen?
Die identifizierten Zielgrößen und die zugehörigen Zielfunktionen wurden in Abschnitt 5.2 definiert und sind in Tabelle 5.8 aufgelistet. Zur vollständigen Bewertung einer Lösung wird zusätzlich die, in Gleichung 5.29 definierte, Straffunktion herangezogen, um die Einhaltung der Anforderungen an die Zulässigkeit und Vollständigkeit der Lösung zu bestimmen. Die Zielfunktion ist als gewichtete Summenfunktion (Linearkombination der Teilfunktionen) definiert (siehe 5.10), um eine einfache Gewichtung der berücksichtigten Teilziele zu ermöglichen. Um die Zielfunktion annähernd unabhängig von der Problemstellung zu gestalten, werden die enthaltenen Teilzielfunktionen auf die jeweilige Problemgröße skaliert.

Q 1.2 Welche mathematischen Verfahren sind als Basis zur Entwicklung einer Optimierungsmethode für das vorliegende Problem geeignet?
Da das vorliegende Planungsproblem der Komplexitätsklasse $\mathcal{NP}$-schwer angehört und Publikationen zu ähnlichen und artverwandten Problemstellungen zeigen, dass exakte Lösungsverfahren nur für stark eingeschränkte Problemgrößen in vertretbarer Zeit Lösungen liefern, wurde für die Umsetzung in dieser Arbeit ein metaheuristisches Verfahren gewählt. Konkret wurde ein GA, der in zahlreichen Mehrzieloptimierungsproblemen Anwendung findet, erfolgreich eingesetzt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Q 1.3 Wie kann diese Methode gestaltet werden, um das vorliegende Problem praxisrelevant zu lösen? (Laufzeit, Berücksichtigung der zuvor identifizierten Zielgrößen)

Bei der Gestaltung der Methode wurde insbesondere auf eine schnelle Erzeugung zulässiger Lösungen und eine gleichmäßige Optimierung aller Zielgrößen geachtet. Um eine möglichst zielgerichtete und damit schnellere Konvergenz zu erreichen, wurden, eigens für die vorliegende Problemstellung geeignete, Operatoren entwickelt. Zusätzlich wurde bei der Implementierung des Verfahrens die Reparatur von unzulässigen Lösungen vorgesehen, um die Suche zielgerichteter zu gestalten. Die Bestrafung unzulässiger Lösungen treibt die Suche im zulässigen Bereich des Lösungsraums voran.

Q 1.4 Wie stark weichen die Ergebnisse des entwickelten Verfahrens in ihrer Güte von manuell, durch Planungspersonal, erzeugte Pläne ab und können diese Pläne verbessert werden?

Zur Bewertung und Beurteilung wurde eine Lösung des entwickelten Verfahrens mit einer Referenzlösung, die von einer Expertin[10] mit mehrjähriger Erfahrung erstellt wurde, verglichen. Aktuell ist das entwickelte Verfahren zwar nicht in der Lage, das Können einer Einsatzplanerin zu replizieren oder zu übertreffen, liefert jedoch vergleichsweise gute und zulässige Lösungen, die entweder direkt genutzt werden, oder als Ausgangspunkte für weitere manuelle Anpassungen durch Fachpersonal dienen können.

Q 1 Wie können Ausbildungspläne für das vorliegende Zuteilungsproblem im ÄAM mithilfe einer mathematischen Methode (exakt oder Näherungsverfahren) erzeugt und optimiert werden, sodass die Qualität von manuell erstellten Ausbildungsplänen angenähert oder übertroffen werden kann?

Die Beantwortung der übergeordneten Forschungsfrage lässt sich abschließend mithilfe der Antworten auf die Fragen **Q 1.1** - **Q 1.4** beantworten. Zusammenfassend kann festgehalten werden, dass die Erstellung und Optimierung mithilfe eines problemspezifischen metaheuristischen Verfahrens, also einem Näherungsverfahren, insbesondere

[10] siehe Fußnote 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

für größere Probleme, zweckmäßig ist.

7.6 Resultate in Bezug auf das Forschungsdesign

In der angewandten Forschungsmethode (DSRM) [14] liegt das Hauptaugenmerk auf dem erzeugten Artefakt und dessen Relevanz für die Problemstellung. Das in den Kapiteln 1 und 2 identifizierte Problem wurde nach einem problemzentrierten Ansatz behandelt. Der nachgelagerte iterative Prozess (siehe Abbildung 1.4) wurde im Zuge dieser Arbeit vollständig durchlaufen. Die, aus der Problemstellung abgeleiteten, Ziele und Fragen (**Zieldefinition**) wurden eingangs, in der Phase der Zieldefinition erfasst. Im Rahmen der **Gestaltung und Entwicklung** wurden, wie in Kapitel 5 beschrieben, die Erkenntnisse aus den vorigen Kapiteln genutzt, um ein geeignetes Artefakt zur Behandlung der vorliegenden Problemstellung zu entwickeln. Das so erzeugte Artefakt wurde anschließend auf eine konkrete Probleminstanz zur **Demonstration** erfolgreich angewendet (siehe Kapitel 6). Die **Bewertung** der Ergebnisse erfolgte ebenfalls in Kapitel 6 mithilfe des entwickelten Bewertungssystems und durch eine Expertin[11]. Die **Kommunikation** der Ergebnisse erfolgt mithilfe dieser Arbeit. Die gesetzten Ziele konnten erreicht und das gewünschte Artefakt zur Behandlung der Problemstellung konnte erzeugt werden.

7.7 Ausblick

Bei der Wahl der nächsten Schritte zur Behandlung des vorliegenden Planungsproblems kann unterschieden werden zwischen Verbesserungen des bestehenden Verfahrens und alternativen Implementierungen mithilfe anderer Verfahren.

Zur Verbesserung des bestehenden Verfahrens können die entwickelten Algorithmen hinsichtlich ihrer Laufzeiteffizienz optimiert werden, wobei die Reparaturalgorithmen und die Bewertungsfunktionen bei Laufzeitanalysen mithilfe eines Profilers das größ-

[11] siehe Fußnote 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

te Optimierungspotenzial zeigten. Insbesondere die Bewertungsfunktion verfügt über einen großen Teil an Routinen, die zwar für die Entwicklung hilfreich sind, im Produktivbetrieb jedoch negative Auswirkungen auf die Laufzeit des Verfahrens haben und keinen Mehrwert bei der Optimierung liefern.

Sowohl die Hyperparameter des Verfahrens, als auch die Parameter der Zielfunktion können in weiteren Schritten methodisch angepasst und optimiert werden. Dafür ist für beide Parametersets eine adaptive Anpassung sinnvoll. Die Parameter der Zielfunktion können so gesteuert werden, dass zu Beginn des Verfahrens das Hauptaugenmerk auf die Einhaltung von Präferenzen und die Minimierung von Ortswechseln gelegt wird. Die Rekombinationswahrscheinlichkeit kann ebenfalls zeitabhängig abnehmend gestaltet werden.

Die Implementierung der Weiterentwicklung des NSGA-II (NSGA-III [103]) für das vorliegende Problem liegt außerdem auf der Hand. Ebenso wäre ein alternativer Ansatz mit einem weiteren gängigen EA, dem MOEA/D [105] denkbar.

Eine etwaige komplexere, nichtlineare Modellierung der Zielfunktion stellt Verbesserungspotenzial dar, um das stückweise Aufweichen von Randbedingungen und Zielfunktionen zu ermöglichen.
Da die aktuelle Software vollständig in der interpretierten Programmiersprache Python implementiert ist und die Sprache im Vergleich zu effizienteren, kompilierten Sprachen einen wesentlichen Nachteil hinsichtlich ihrer Leistungsfähigkeit aufweist, ist die Neuimplementierung von leistungskritischen Komponenten in einer effizienteren Programmiersprache anzudenken. Obwohl der bestehende Programmcode weitestgehend parallelisiert wurde (Multithreading), limitiert der GIL (global interpreter lock) [106], der in der eingesetzten CPython Implementierung verwendet wird, die Nutzung von Rechnerressourcen in mehrläufigen Programmen. Aus diesem Grund kann aktuell lediglich ein Rechnerkern genutzt werden und Parallelisierungen bringen nicht den erhofften Effekt mit sich. GAs eignen sich jedoch an sich gut, um die genetischen Operatoren vollständig parallel für die gesamte Population auszuführen. Diese Probleme kann eine Sprache, wie Rust [107], die explizit für parallelisierte, hocheffiziente Anwendungen entwickelt wurde, lösen. Eine erwartbare Effizienzsteigerung hinsichtlich der Lauf-

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

zeit, aufgrund einer Neuimplementierung in dieser Sprache, von etwa einer Größenordnung (Faktor 10) erscheint somit plausibel. Insbesondere für größere Probleminstanzen würde das eine bessere Skalierbarkeit der Anwendung sicherstellen.

Zur Erzeugung besserer Startlösungen für das entwickelte Verfahren würde sich ein heuristisches Eröffnungsverfahren eignen. Mit dessen Hilfe ist eine verhältnismäßig große Reduktion der Laufzeit realistisch, da das Finden von zufriedenstellenden Startlösungen von der Heuristik übernommen werden kann und die zeitintensive Optimierung von einem wesentlich besseren Niveau ausgehend gestartet werden kann.

Eine weitere vielversprechende Option zur Weiterentwicklung liegt in der Entwicklung eines hybriden Verfahrens, das mithilfe eines GA eine Greedy-Heuristik steuert, indem deren Eingaben durch den GA optimiert werden. Ein vergleichbarer Ansatz für das NSP ist in [26] beschrieben und erfolgreich umgesetzt. Mit einem solchen Verfahren können sowohl die Stärken einer problemspezifischen Heuristik, als auch einer Metaheuristik genutzt und das Erzeugen „sinnloser“ Lösungen vermieden werden.
Alternativ wäre eine zusätzliche lokale Suchmethode für das entwickelte Verfahren denkbar, das, basierend auf einem bestehenden Ausbildungsplan, versucht, einzelne Zuweisungen zu vertauschen und kleine Abschnitte im Ausbildungsplan zu optimieren. Mit einem solchen Suchverfahren könnte insbesondere die Häufigkeit der Ortswechsel weiter minimiert werden.

Ungeachtet der zuvor diskutierten Einschränkungen und Verbesserungspotenziale der entwickelten Methode können mithilfe dieses Verfahrens Ausbildungspläne für den realen Einsatz in der österreichischen Ärzt*innenausbildung erstellt werden, die alle relevanten Rahmenbedingungen erfüllen und die identifizierten Zielgrößen optimieren.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Anhang A

Anhang

A.1 Leitfadengestütztes Interview einer Expertin

Das leitfadengestützte Interview, das mit Lisa Holzgruber (siehe Fußnote 1) geführt wurde, wurde anhand der nachfolgend aufgelisteten Fragen durchgeführt. Die Ziele des Interviews waren die Erfassung des aktuell üblichen Vorgehens in der Rotationsplanung, sowie die Identifikation relevanter Ziele, Randbedingungen und Herausforderungen bei der Einsatzplanung im ÄAM. Der verwendete Leitfaden für das leitfadengestützte Expert*inneninterview [108] ist vergleichsweise einfach gehalten und enthält lediglich notwendige Leitfragen, um das betreffende Optimierungsproblem charakterisieren zu können und gleichzeitig das Gespräch so offen wie möglich zu halten. Die Gespräche wurden in mehreren Etappen, verteilt über mehrere Monate durchgeführt, um eine Art Kontrollschleife in den Entwicklungsprozess der Methode einzubetten und neu aufgekommene Fragen, die sich im Zuge der Entwicklung ergaben zu beantworten. Auf die Transkription der Gespräche wurde verzichtet, jedoch wird in der gesamten Arbeit auf die Erkenntnisse aus diesen Interviews verwiesen und eingegangen.

Gespräch 1 (am 11.02.2022)

F1 Welche Hilfsmittel werden aktuell bei der Rotationsplanung eingesetzt?

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

F2 Wie gehen Planer*innen bei der Einsatzplanung von Turnusärzt*innen vor?

F3 Was sind die größten Herausforderungen bei der Rotationsplanung und wie wird mit ihnen umgegangen?

F4 Welche Probleme sollte ein Planungswerkzeug lösen können?

F5 Wo sind die rechtlichen Rahmenbedingungen für die ärztliche Ausbildung geregelt?

F6 Wie geht das Planungspersonal aktuell mit der großen Problemkomplexität um, welche Vereinfachungen bzw. Annahmen werden getroffen?

Gespräch 2 (am 25.03.2022)

F1 Nach welchen Gesichtspunkten wird ein erzeugter Plan aktuell beurteilt?

F2 Bieten die eingesetzten Hilfsmittel Möglichkeiten zur Beurteilung von Ausbildungsplänen?

F3 Welche Ziele verfolgt die Krankenhausleitung aus strategischer Sicht in der Rotationsplanung von Turnus*ärztinnen?

F4 Wie können die Bedürfnisse von Turnusärzt*innen in der Rotationsplanung berücksichtigt werden und welche Wünsche haben diese üblicherweise?

F5 Welche Einschränkungen gibt es bei der Planung (Randbedingungen, die keinesfalls verletzt werden dürfen)?

F6 Welche Ziele sollten, abgesehen von den aktuell bereits berücksichtigten Gesichtspunkten, idealerweise noch beachtet werden?

Gespräch 3 (am 20.05.2022)

F1 Welche der identifizierten Optimierungskriterien sind die wichtigsten und sollten von einem Planungsalgorithmus vorrangig behandelt werden?

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

F2 Wie werden Personen in Teilzeitbeschäftigung in der Verplanung behandelt und wie viele Personen in ärztlicher Ausbildung verfügen über eine Teilzeitbeschäftigung?

F3 Ist die Ressourcenverfügbarkeit (Ausbildungsstellen und Dienstposten) im Zeitverlauf gleichbleibend?

F4 Wieviele Ärzt*innen müssen im Regelfall gleichzeitig verplant werden?

A.2 Regelbasierte Planungsheuristik

Aktuell ist ein regelbasiertes Planungswerkzeug, das die mehrjährige Berufserfahrung einer Einsatzplanerin[12] aus einem österreichischen Krankenanstaltenverbund in Form eines strikt regelbasierten Planungsalgorithmus verkörpert, in Entwicklung. Dieses Planungswerkzeug soll zuverlässige und zulässige Ausbildungspläne liefern, wobei diese besagten Pläne, aller Voraussicht nach, Optimierungspotenzial aufweisen werden. Der Grund dafür liegt in der inhärenten Planungslogik des Werkzeugs, die in ihrer Komplexität beschränkt und somit nicht in der Lage ist, den gesamten potenziellen Lösungsraum nach geeigneten, (annähernd) optimalen Lösungskandidaten zu durchsuchen. Die Beschränktheit des zum Einsatz kommenden Algorithmus liegt in seiner grundlegenden Beschaffenheit. Die Strategie, die der Algorithmus verfolgt, wird als Greedy-Strategie bezeichnet, der Algorithmus fällt also unter die Klasse der Greedy-Algorithmen (gierige Algorithmen). Ein Greedy-Algorithmus für Optimierungsprobleme zeichnet sich dadurch aus, dass er viele sequenzielle Entscheidungsprobleme löst, indem er bei jedem auftretenden Subproblem nach den aktuell verfügbaren Informationen die Entscheidung zugunsten eines lokalen Optimums trifft. [4]

Im vorliegenden Fall handelt es sich, exakt formuliert, um eine Greedy-Heuristik, da die Entscheidung für die erwähnten lokalen Optima auf Annahmen und Regeln basiert. Diese Annahmen und Regeln haben ihren Ursprung in der mehrjährigen Erfahrung der, in Abschnitt 2.3 befragten, Expertin. Der Ablauf des Planungsverfahrens ist stark ver-

[12] siehe Fußnote 1

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

einfacht in Algorithmus 9 dargestellt. Jede Person verfügt über ein ihr zugeordnetes Curriculum, einer geordneten Liste an zu absolvierenden Fächern und deren Dauern.

Algorithmus 9: Regelbasierte Planungsheuristik

```
1 P ← [p_1, ... , p_n]                    /* geordnete Liste der zu verplanenden Personen */
2 S ← leerer Ausbildungsplan
3 for person in P do
4     for fach in person.curriculum do
5         platz, zeitraum ← suche freien Ausbildungsplatz für fach für seine Dauer
6         S ← plane platz für person und gefundenen zeitraum
7     end
8 end
9 return Befüllter Ausbildungsplan S
```

Der Algorithmus wählt „greedy“ (gierig) für jede Person und jedes Fach in deren Ausbildung den besten noch verfügbaren Ausbildungsplatz aus. Trotz der sehr abstrakten Beschreibung des Vorgehens lässt sich die Schwäche des Verfahrens unmittelbar festmachen. Da die gefundene Lösung in erster Linie von der Reihenfolge der zu verplanenden Personen und deren Curricula abhängt, hängt auch die Güte dieser Lösung davon ab. Dementsprechend kann das Verfahren zwar unter idealen Anfangsbedingungen eine näherungsweise optimale Lösung finden, diese idealen Anfangsbedingungen (Sortierung der Personen und Curricula) herzustellen, ist allerdings keinesfalls trivial. Das aktuelle Verfahren verwendet entweder zufällig oder manuell gewählte Reihenfolgen, ist allerdings nicht in der Lage selbst eine optimale Reihenfolge zu finden. Das Finden dieser optimalen Anfangsbedingungen stellt mit der zugrundeliegenden Greedy-Heuristik ein eigenes Optimierungsproblem dar, dessen Behandlung jedoch nicht Ziel der vorliegenden Arbeit.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

A.3 Beispiel Mehrzieloptimierung

Stellt man sich eine Autofahrt von Ort A zu Ort B vor, so könnten aus Sicht einer/eines Reisenden zwei wesentliche Zielgrößen der Kraftstoffverbrauch und die Reisedauer sein. Da mit höherer Geschwindigkeit üblicherweise der Kraftstoffverbrauch je Streckeneinheit eines Fahrzeugs steigt, ist eine minimale Reisedauer bei gleichzeitig minimalem Kraftstoffverbrauch schlicht unmöglich. Eine stark vereinfachte Modellierung dieses Problems wäre über die Durchschnittsgeschwindigkeit möglich. Somit könnte das Zielsystem unter der Annahme konstanter Geschwindigkeit und einem polynomiellen Zusammenhang zwischen Geschwindigkeit und Kraftstoffverbrauch wie folgt aussehen:

$$\begin{gathered} \bar{v}\text{...Durchschnittsgeschwindigkeit} \\ f_{Kraftstoff}(\bar{v}) = k_1 \bar{v}^a \\ \text{mit } a > 1 \\ f_{Zeit}(\bar{v}) = \frac{k_2}{\bar{v}} \end{gathered} \tag{A.1}$$

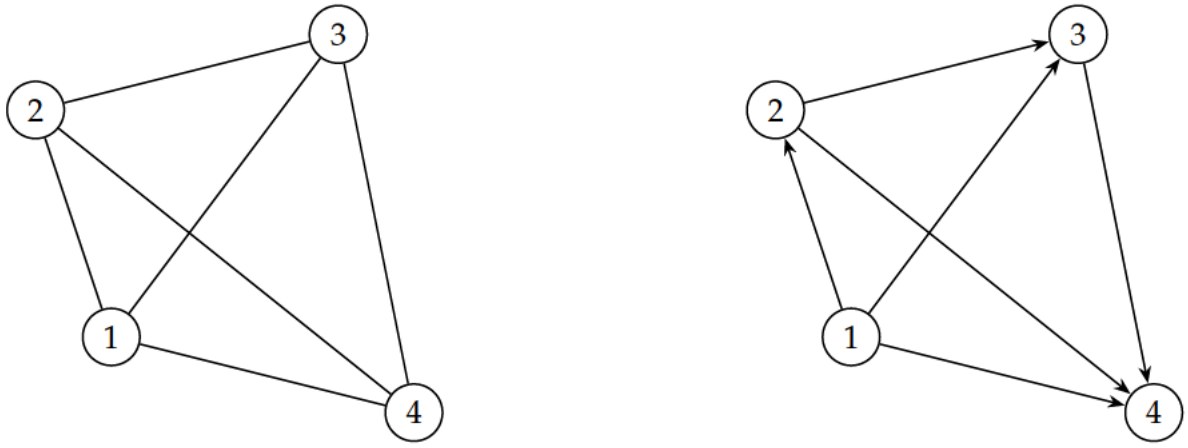
Die Minimierung der zwei Zielfunktionen steht im Widerspruch zueinander, da $f_{Kraftstoff}$ mit zunehmender Geschwindigkeit $\bar{v}$ zunimmt und f_{Zeit} abnimmt. Die eine Zielfunktion lässt sich also nicht verbessern, ohne dabei gleichzeitig die andere zu verschlechtern. Somit ergeben sich für verschiedene Werte $\bar{v}$ verschiedene Pareto-optimale Lösungen für das Optimierungsproblem. Wie schnell am Ende tatsächlich gefahren wird und welcher Kompromiss zwischen Kraftstoffverbrauch und Zeitverlust gewählt wird, obliegt der/dem Fahrer*in.

A.4 Graphen und Bäume

Graphen sind ein Konzept aus der theoretischen Informatik bzw. der Mathematik und sind geeignet, um Netzwerke bzw. komplexe Zusammenhänge zwischen verschiedenen Informationseinheiten abzubilden. Ein *ungerichteter Graph* G besteht in seiner einfachsten Form aus Ecken (Knoten) E, und Kanten K. [109] Die Kanten verbinden dabei die einzelnen Ecken des Graphen und können so genutzt werden, um einen Zusammenhang

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

zwischen zwei Knoten zu modellieren. Ein gerichteter Graph enthält zusätzlich noch Information über die Richtung einer Verbindung (Kante) zwischen zwei Ecken. [109] In Abbildung A.1 ist eine bildliche Darstellung eines gerichteten und eines ungerichteten Graphen ersichtlich. Eine Ecke eines Graphen wird als e_i bezeichnet. Ein Kantenzug in



(a) ungerichteter Graph (b) gerichteter Graph

Abbildung A.1: Graphen

einem Graphen wird mit <$e_1 \ldots e_z$> beschrieben und stellt die Verbindung zwischen einzelnen Ecken dar. Eine Ecke e ist in einem Graphen von der Ecke e' aus erreichbar, wenn es einen Kantenzug <e,...,e'> gibt. Ein Weg ist ein Kantenzug der keine Kante mehrfach enthält. Ein Weg ist geschlossen, wenn $e_1 = e_z$, also die erste Ecke des Weges die letzte Ecke ist (wobei auf diesem Weg keine Kante mehrfach verwendet werden darf). Als **Wald** wird ein ungerichteter Graph bezeichnet, der keinen geschlossenen Weg enthält (siehe Abbildung A.2). [109] Die einzelnen Bestandteile eines Waldes werden als Bäume bezeichnet. [109] Für Bäume gilt allgemein $m - n = 1$ mit Kantenanzahl m und der Knotenanzahl n. Ein ungerichteter Graph G ist ein Baum, wenn er eine der folgenden drei Bedingungen erfüllt:

- $m - n = 1$ und G ist zusammenhängend
- $m - n = 1$ und
- Zwischen jedem Knotenpaar gibt es genau einen Weg

In [109] wird ein Baum wie folgt definiert:

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

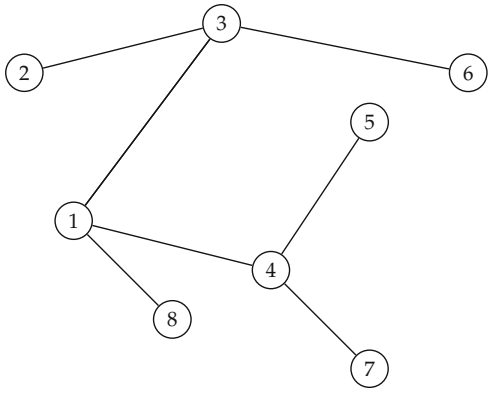


Abbildung A.2: Graph ohne geschlossenen Weg - Wald

Ein ungerichteter Graph G ist genau dann ein Baum, wenn eine der folgenden drei Bedingungen erfüllt ist:

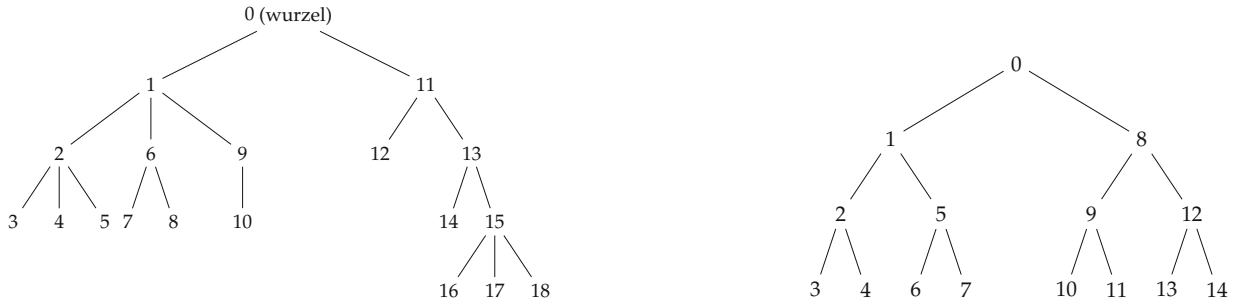
a) ist zusammenhängend und $m - n = 1$;

b) G enthält keinen geschlossenen Weg und $m - n = 1$

c) in G gibt es zwischen jedem Paar von Ecken genau einen Weg.

Ein Wurzelbaum ist eine Sonderform eines Baumes und hat einen Wurzelknoten, von dem der Baum aus verzweigt ist. Wurzelbäume werden i.d.R als gerichtete Graphen interpretiert und eignen sich hervorragend hierarchische und logische Zusammenhänge bzw. Abläufe darzustellen. Im Kontext dieser Arbeit sind Wurzelbäume von besonderem Interesse, da sie sich hervorragend eignen, um den Lösungsraum von kombinatorischen Optimierungsproblemen abzubilden. In Abbildung A.3 sind Wurzelbäume als verallgemeinerter Wurzelbaum und als Binärbaum dargestellt.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.



(a) allgemeiner Wurzelbaum

(b) Binärbaum

Abbildung A.3: Bäume

A.5 Datenstruktur zur Planung

Die Darstellungen A.4 und A.5 stellen dieselben Informationen in unterschiedlicher Art dar. Sie enthalten ein minimales Beispiel für die benötigten Informationen zur Planung von Personen auf Ausbildungsstellen (TrainingPositions) und Dienstposten (Positions).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

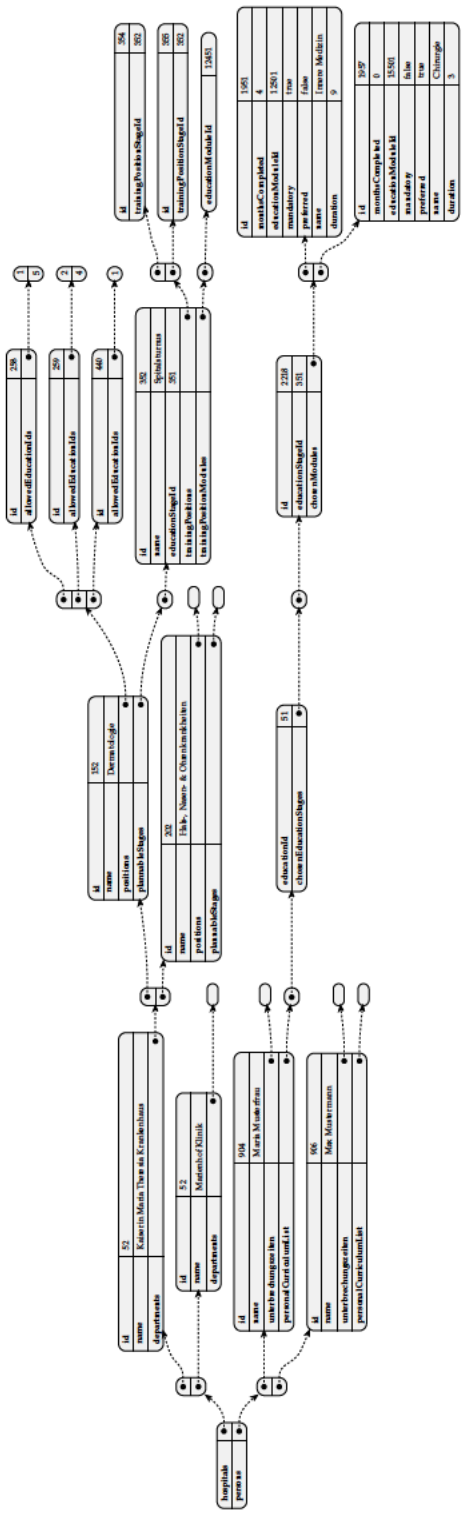


Abbildung A.4: Strukturdiagramm

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Data						
hospitals	id	52				
	name	Kaiserin Maria Theresia Krankenhaus				
	departments	id	152			
		name	Dermatologie			
		positions	id	258		
			allowedEducationIds	1		
				5		
			id	259		
			allowedEducationIds	2		
				4		
			id	440		
			allowedEducationIds	1		
		plannableStages	id	352		
			name	Spitalsturnus		
			educationStageId	351		
			trainingPositions	id	354	
				trainingPositionStageId	352	
				id	355	
				trainingPositionStageId	352	
			trainingPositionModules	educationModuleId	12451	
		id	202			
		name	Hals-, Nasen- & Ohrenkrankheiten			
		positions				
		plannableStages				
	id	52				
	name	Marienhof Klinik				
	departments					
persons	id	904				
	name	Maria Musterfrau				
	unterbrechungszeiten					
	personalCurriculumList	educationId	51			
		chosenEducationStages	id	2218		
			educationStageId	351		
			chosenModules	id	1951	
				monthsCompleted	4	
				educationModuleId	12501	
				mandatory	true	
				preferred	false	
				name	Innere Medizin	
				duration	9	
				id	1957	
				monthsCompleted	0	
				educationModuleId	15501	
				mandatory	false	
				preferred	true	
				name	Chirurgie	
				duration	3	
	id	906				
	name	Max Mustermann				
	unterbrechungszeiten					
	personalCurriculumList					

Abbildung A.5: Strukturdiagramm in tabellarischer Form

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

A.6 NSGA-Selektion

```python
def non_dominated_sorting_selection(
    population: Population, offspring_size: int, use_penalty: bool = True
) -> Population:
    """
    based on NSGA-II
    """
    K = 100  # inf according to publication but using big enough numerical
    ↪ value

    # make fronts
    domination_counter = [0] * len(population)
    dominated_solutions = {i: [] for i in range(len(population))}
    fronts = [
        [],
    ]
    for i, ind in enumerate(population):
        ind.fitness.objective_result
        for j, other_ind in enumerate(population):
            ind: Individual
            if dominates(ind, other_ind, use_penalty):
                dominated_solutions[i].append(j)
            elif dominates(other_ind, ind, use_penalty):
                domination_counter[i] += 1
        if domination_counter[i] == 0:
            fronts[0].append(i)

    front_num = 1
    current_front = fronts[front_num - 1]

    while len(current_front) > 0:
        next_front = []
        for i in current_front:
            for j in dominated_solutions[i]:
                domination_counter[j] -= 1
                if domination_counter[j] == 0:
                    next_front.append(j)

        fronts.append(next_front)
        front_num += 1
        current_front = fronts[front_num - 1]

    # nondominated sorting finished

    new_pop = []
    i = 0
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
    while len(new_pop) + len(fronts[i]) <= offspring_size:
        new_pop.extend(fronts[i])
        i += 1

    if len(new_pop) < offspring_size:
        # select inds by crowding distance
        indices = copy(fronts[i])
        distances = {idx: 0 for idx in indices}
        for objective in population[0].fitness.objective_values.objectives:
            indices.sort(
                key=lambda i:
                ↪ population[i].fitness.objective_result.single_values[
                objective
                ]
            )

            f_min =
            ↪ population[indices[0]].fitness.objective_result.single_values[
                objective
            ]
            f_max =
            ↪ population[indices[-1]].fitness.objective_result.single_values[
                objective
            ]
            span = f_max - f_min

            if span > 0.0:
                distances[indices[0]] += K  # np.inf
                distances[indices[-1]] += K  # np.inf
                for k, idx in enumerate(indices[1:-1]):
                    dist = (
                        population[
                            indices[k + 2]
                        ].fitness.objective_result.single_values[objective]
                        - population[indices[k]].fitness.objective_result.sing
                        ↪ le_values[
                            objective
                        ]
                    ) / span
                    distances[idx] += dist

        # sorting for penalty
        if use_penalty:
            indices.sort(
                key=lambda i: (
                    population[i].fitness.penalty_value,
                    population[i].fitness.value,
                )
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
            )
            f_min = population[indices[0]].fitness.penalty_value
            f_max = population[indices[-1]].fitness.penalty_value
            span = f_max - f_min

            if span > 0:
                distances[indices[0]] += K
                distances[indices[-1]] += K
                for k, idx in enumerate(indices[1:-1]):
                    dist = (
                        population[indices[k + 2]].fitness.penalty_value
                        - population[indices[k]].fitness.penalty_value
                    ) / span

                    if population[k + 1].fitness.penalty_value == 0.0:
                        dist += K  # np.inf
                    distances[idx] += dist

        sorted_last_front = sorted(indices, key=lambda x: distances[x],
        ↪ reverse=True)
    j = 0
    while len(new_pop) < offspring_size:
        new_pop.append(sorted_last_front[j])
        j += 1

    finished_pop = Population([population[i] for i in new_pop])
    if population.best not in finished_pop:
        finished_pop[
            -1
        ] = (
            population.best
        )  # ensuring elitism in some edge cases with small populations

    return finished_pop


def dominates(ind1: Individual, ind2: Individual, use_penalty: bool = True) ->
↪ bool:
    """
    determines if ind1 dominates ind2 in terms of pareto domination
    use penalty determines whether the penalty function is also treated like
↪ an objective
    if use_penalty == False elitism regarding overall fitness function is not
↪ guaranteed
    """
    all_objectives = ind1.fitness.objective_values.objectives
    dominates = False

```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
    for objective in all_objectives:
        obj1 = ind1.fitness.objective_result.single_values[objective]
        obj2 = ind2.fitness.objective_result.single_values[objective]

        if obj1 < obj2:
            dominates = True
        elif obj1 > obj2:
            return False

    if use_penalty:
        if ind1.fitness.penalty_value < ind2.fitness.penalty_value:
            dominates = True
        elif ind1.fitness.penalty_value > ind2.fitness.penalty_value:
            return False

    return dominates
```

Code A.1: Umweltselektion basierend auf NSGA-II [101]

A.7 Mutationsoperator – vollständiger Code

```
class Algorithm:
    def mutation(self:"Algorithm", ind: Individual):

            ind = deepcopy(ind)
            penalty = ind.fitness.penalty_value > 0
            # lösche alle Zuweisungen, die keinen Ausbildungsfortschritt
             ↪ bringen
            ind = delete_assignments_without_training(ind)

            if penalty:
                p_idx = []
                idx_map = self.calculation_data.person_calculation_data.invert
                 ↪ ed_index
                # bestimme individuen deren Ausbildung nicht fertig ist
                for res in ind.simulation_result:
                    if not res.education_finished_without_violation:
                        p_idx.append(idx_map[res.person_id])

                if len(p_idx) > 0:

                    random.shuffle(p_idx)
                    # sortiere Zuweisungen um
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
                ind = sort_positions_for_person(ind, p_idx,
                ↪ self.calculation_data)

                # befülle greedy und zufällig mit freien Kombinationen
                # aus Dienstposten und Ausbildungsstellen
                ind = fill_all_possible_indices_with_unused_combinations(
                    ind, p_idx, self.calculation_data
                )

        if random.random() <= self.mutation_probability:
                if penalty == False:
                operators = [
                     # suche einmonatige Zuweisungen und ersetze
                     # sie durch eine andere freie Kombination
                    insert_unused_combinations_instead_of_sma,

                    # wähle module einer Peron und sortiere die
                    ↪ Zuweisungen so um,
                    # dass das Modul am Stück und in einer Abteilung
                    ↪ abgeschlossen wird
                    complete_module_in_one_piece_for_person,

                    #tausche eine einmonatige Zuweisung einer Person
                    # mit der Zuweisung einer anderen Person
                    swap_sma_assignment,
                ]
                # wähle zufällig zwischen den Operatoren mit bias
                weights = [0.2, 0.4, 0.4]
                operator = random.choices(operators, weights=weights,
                ↪ k=1)[0]
                ind = operator(ind, self.calculation_data)

                else:
                    # wenn das Individuum nicht zulässig ist, befülle
                    ↪ greedy mit neuen Kombinationen
                    ind = insert_unused_combinations(
                        ind, self.calculation_data, fraction=0.15
                    )

        # repariere und befülle mit allen noch verfügbaren Dienstposten
        ↪ und Ausbildungsstellen
        ind = infer_unassigned_positions_v3(ind, self.calculation_data)
        ind = infer_unassigned_training_positions(ind,
        ↪ self.calculation_data)

        return ind
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

```python
def delete_assignments_without_training(ind: Individual):
    indices = ind.get_indices_without_training()
    for p_idx, m_idx in indices:
        ind[p_idx, m_idx] = Assignment(0, 0, False)

    return ind


def sort_positions_for_person(
    ind: Individual, p_idx: int | list[int], calculation_data: CalculationData
) -> Individual:
    indices = p_idx
    if isinstance(p_idx, int):
        indices = [p_idx]

    for p_idx in indices:
        positions = list(ind[p_idx, :]["position"])
        fixed = list(ind[p_idx, :]["fixed"])
        tps = list(ind[p_idx, :]["training_position"])

        # delete all unnecessary positions
        for idx, tup in enumerate(zip(positions[:], tps, fixed)):
            p, tp, fix = tup
            if tp <= 0 and not fix:
                positions[idx] = 0

        data = {}
        for idx, pos in enumerate(positions[:]):
            if pos > 0:
                pos_dep = calculation_data.position_department_mapping[pos]
                pos_hosp = calculation_data.position_hospital_mapping[pos]

                if not pos_hosp in data.keys():
                    data[pos_hosp] = {pos_dep: [pos]}

                else:
                    if data[pos_hosp].get(pos_dep) != None:
                        data[pos_hosp].get(pos_dep).append(pos)

                    else:
                        data[pos_hosp][pos_dep] = [pos]

        new_pos = []
        for h, di in data.items():
            x = []
            for d, l in di.items():
                x.append(l)
            random.shuffle(x)
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
            new_pos.append(x)
        random.shuffle(new_pos)

        final_pos = []

        for i in new_pos:
            for j in i:
                final_pos.extend(j)

        if ind.n_months - len(final_pos) > 0:
            pad = [0] * (ind.n_months - len(final_pos))
            final_pos.extend(pad)

        shift = 0
        idx = 0
        while idx + shift < ind.n_months:
            pos = final_pos[idx]
            while ind[p_idx, idx + shift]["fixed"]:
                shift += 1

            if idx + shift >= ind.n_months:
                break
            if not ind[p_idx, idx + shift]["fixed"]:

                j = (np.array(ind[:, idx + shift]["position"]) ==
                 ↪  pos).nonzero()[0]
                if len(j) > 0:
                    for k in j:
                        ind[k, idx + shift] = Assignment(0, 0, False)

                ind[p_idx, idx + shift] = Assignment(pos, 0, False)

            idx += 1

        return ind


def fill_all_possible_indices_with_unused_combinations(
    ind: Individual, person_indices: list[int], calc_data: CalculationData
):
    for p_idx in person_indices:
        p_data = calc_data.person_calculation_data.get_person_calc_data_by_ind
         ↪  ex(p_idx)
        indices = ind.get_indices_without_training_for_person(p_idx)
        m_indices = [x[0] for x in indices]

        for m_idx in m_indices:
            used_pos = set(ind.get_used_positions_in_month(m_idx))
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
            used_tpos = set(ind.get_used_training_positions_in_month(m_idx))
            unused_pos = list(set(p_data.allowed_positions) - used_pos)
            random.shuffle(unused_pos)

            combination = None
            for pos in unused_pos:
                combs = p_data.get_allowed_combinations_with_position(pos)
                if combs is not None:

                    random.shuffle(combs)
                    for c in combs:
                        if c.training_position_id not in used_tpos:
                            combination = c
                            break
                if combination is not None:
                    break

            if combination is not None:
                ind[p_idx, m_idx] = combination.to_assignment()

    return ind


def insert_unused_combinations_instead_of_sma(
    ind: Individual, calc_data: CalculationData
):
    NUM = 1
    person_indices = list(range(ind.n_persons))
    random.shuffle(person_indices)
    max_idx = min(len(person_indices), NUM)
    p_indices = person_indices[:max_idx]

    d = {}
    for p_idx in p_indices:
        d[p_idx] = get_sma_indices_for_person(ind, calc_data, p_idx)

    ind = fill_all_possible_indices_with_unused_combinations_v2(ind, d,
    ↪  calc_data)
    ind = fill_all_possible_indices_with_needed_combinations(
        ind,
        [
            p_idx,
        ],
        calc_data,
    )

    return ind
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```python

def get_sma_indices_for_person(
    ind: Individual, calc_data: CalculationData, p_idx: int
) -> list[int]:

    marker = []
    asg = Assignment(*ind[p_idx, 0])
    dep = calc_data.position_department_mapping.get(asg.position, 0)
    left_dep = dep + 1
    left = asg
    for m_idx in range(ind.n_months):
        if m_idx < ind.n_months - 1:
            right = Assignment(*ind[p_idx, m_idx + 1])
            right_dep =
             ↪  calc_data.position_department_mapping.get(asg.position, 0)
        else:
            right = asg
            right_dep = dep + 1

        if (
            (left_dep != dep and left.position >= 0)
            and (right_dep != dep and right.position >= 0)
            and not asg.fixed
        ):
            marker.append(True)
        else:
            marker.append(False)

        left = asg
        left_dep = dep
        asg = right
        dep = right_dep

    return [i[0] for i in np.transpose(np.nonzero(marker))]



def complete_module_in_one_piece_for_person(
    ind: Individual, calc_data: CalculationData
):
    p_idx = random.choice(range(ind.n_persons))
    modules = get_assigned_modules_for_person(ind, p_idx, calc_data)
    p_data =
     ↪  calc_data.person_calculation_data.get_person_calc_data_by_index(p_idx)
    # select random module that is not consecutive yet or has alternating
     ↪  training_positions
    used_mods = list({i for i in modules if i > 0})
    random.shuffle(used_mods)
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
    non_consec = False
    for mod in used_mods:
        indices = []
        for i, m in enumerate(modules):
            if m == mod:
                indices.append(i)
        # check if all indices are consec
        if len(indices) > 1:
            prev_idx = indices[0]
            for i in indices[1:]:
                if i - prev_idx != 1:
                    non_consec = True
                    sel_mod = mod
                    break
        if non_consec:
            break
    if non_consec:
        mod_indices = indices
        needed_months = modules.count(sel_mod)

        start_idx = min(mod_indices)

        combination =
        ↪ random.choice(p_data.get_allowed_combinations_for_module(sel_mod))
        asg = combination.to_assignment()
        hospital = calc_data.position_hospital_mapping.get(asg.position, -1)

        max_idx = 0
        prev_hosp = 0
        max_count = 0
        counter = 0
        for idx in range(ind.n_months - needed_months):
            hosp = calc_data.position_hospital_mapping.get(ind[p_idx,
            ↪ idx]["position"])
            if prev_hosp == hosp:
                counter += 1
                if counter > max_count:
                    max_idx = idx
            prev_hosp = hosp

        start_idx = max_idx + 1

        deleted_mods = []
        affected_persons = {}

        # clear up all indices in person row that are rescheduled
        for i in mod_indices:
            if i != start_idx:
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

```
                ind[p_idx, i] = Assignment(0, 0, False)

        if start_idx + needed_months <= ind.n_months:
            i = 0

            while i < needed_months:
                idx = start_idx + i
                deleted_mods.append(modules[idx])

                affected_pos_idx = ind.get_index_of_position_in_month(
                    position_id=asg.position, m_idx=idx
                )
                affected_t_pos_idx =
                ↪  ind.get_index_of_training_position_in_month(
                    training_position_id=asg.training_position, m_idx=idx
                )

                if affected_pos_idx is not None:
                    a_data = (
                        calc_data.person_calculation_data.get_person_calc_data
                        ↪  _by_index(
                            affected_pos_idx
                        )
                    )
                    a_asg = Assignment(*ind[affected_pos_idx, idx])
                    comb = a_data.get_combination(
                        a_asg.position, a_asg.training_position
                    )
                    if comb is not None:
                        mods = comb.education_modules_available
                        if mods:

                            if affected_pos_idx in affected_persons.keys():
                                affected_persons[affected_pos_idx].append(mods
                                ↪  [0])
                            else:
                                affected_persons[affected_pos_idx] = [mods[0]]

                    ind[affected_pos_idx, idx] = Assignment(0, 0, False)

                if (
                    affected_t_pos_idx is not None
                    and affected_t_pos_idx != affected_pos_idx
                ):
                    a_data = (
                        calc_data.person_calculation_data.get_person_calc_data
                        ↪  _by_index(
                            affected_t_pos_idx
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
                        )
                    )
                    a_asg = Assignment(*ind[affected_t_pos_idx, idx])
                    comb = a_data.get_combination(
                        a_asg.position, a_asg.training_position
                    )
                    if comb is not None:
                        mods = comb.education_modules_available
                        if mods:
                            if affected_t_pos_idx in affected_persons.keys():
                                affected_persons[affected_t_pos_idx].append(mo
                                ↪ ds[0])
                            else:
                                affected_persons[affected_t_pos_idx] =
                                ↪ [mods[0]]

                    new = Assignment(ind[affected_t_pos_idx, idx]["position"],
                    ↪ 0, False)
                    ind[affected_t_pos_idx, idx] = new

                ind[p_idx, idx] = asg
                i += 1

    ind = fill_all_possible_indices_with_needed_combinations(
        ind, list(affected_persons.keys()), calc_data
    )
    return ind


def swap_sma_assignment(ind: Individual, calc_data: CalculationData):
    """
    search single month assignment that also involves a hospital change
    find combination that is either in left or right hospital
    swap if no useful combination is available
    else use unused combination
    """

    p_idx = random.choice(range(ind.n_persons))
    p_data =
    ↪ calc_data.person_calculation_data.get_person_calc_data_by_index(p_idx)
    hospitals = []
    max_idx = 0
    for m_idx in range(ind.n_months):
        asg = Assignment(*ind[p_idx, m_idx])
        hospitals.append(calc_data.position_hospital_mapping.get(asg.position,
        ↪ 0))
        if asg.training_position > 0:
            max_idx = m_idx
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
381
382     hospitals = hospitals[: max_idx + 1]
383     single_indices = []
384     if hospitals:
385         prev_h = -1
386         for idx, h in enumerate(hospitals[:-1]):
387             next_h = hospitals[idx + 1]
388             if h != prev_h and h != next_h:
389                 single_indices.append(idx)
390             prev_h = h
391
392         if len(hospitals) > 1:
393             if hospitals[-1] != hospitals[-2]:
394                 single_indices.append(len(single_indices) - 1)
395
396     for m_idx in single_indices:
397         old_asg = Assignment(*ind[p_idx, m_idx])
398         old_hosp = hospitals[m_idx]
399         if not old_asg.fixed and old_asg.training_position > 0:
400
401             comb = p_data.get_combination(old_asg.position,
                ↪  old_asg.training_position)
402             if comb and comb.education_modules_available:
403                 mod = comb.education_modules_available[0]
404
405                 if m_idx > 0 and m_idx < ind.n_months - 1:
406                     right = hospitals[m_idx + 1]
407                     left = hospitals[m_idx - 1]
408                 elif m_idx == 0:
409                     right = hospitals[m_idx + 1]
410                     left = right
411                 else:
412                     left = hospitals[m_idx - 1]
413                     right = left
414
415                 hs = set((left, right))
416                 asg = None
417                 poss_combs = p_data.get_allowed_combinations_for_module(mod)
418                 allowed_combs = {
419                     c
420                     for c in poss_combs
421                     if calc_data.position_hospital_mapping.get(c.position_id)
                        ↪  in hs
422                 }
423                 used_pos = set(ind.get_used_positions_in_month(m_idx))
424                 used_tpos =
                    ↪  set(ind.get_used_training_positions_in_month(m_idx))
425                 for c in allowed_combs:
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
                    if (
                        c.position_id not in used_pos
                        and c.training_position_id not in used_tpos
                    ):
                        asg = c.to_assignment()
                        break

            if asg is None:
                pass
            else:
                ind[p_idx, m_idx] = asg
    return ind


def fill_all_possible_indices_with_needed_combinations(
    ind: Individual, person_indices: list[int], calc_data: CalculationData
):
    """
    tries to insert unused combinations that are useful for person
    preferring preferred modules

    """

    for p_idx in person_indices:

        p_data = calc_data.person_calculation_data.get_person_calc_data_by_ind
        ↪  ex(p_idx)

        modules = get_assigned_modules_for_person(ind, p_idx, calc_data)

        tracker = {
            edu_mod_id: mod.months_remaining
            for edu_mod_id, mod in p_data.unfinished_module_info.items()
        }

        preferred_mods = [
            mod_id
            for mod_id, mod in p_data.unfinished_module_info.items()
            if mod.preferred
        ]
        mandatory_mods = [
            mod_id
            for mod_id, mod in p_data.unfinished_module_info.items()
            if mod.mandatory
        ]
        random.shuffle(preferred_mods)
        random.shuffle(mandatory_mods)
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
        other = p_data.unfinished_modules - set(preferred_mods) -
        ↪ set(mandatory_mods)

        mod_prio = mandatory_mods + preferred_mods + list(other)

        for m_idx, mod in enumerate(modules):
            if mod > 0:
                tracker[mod] -= 1
                if tracker[mod] < 1:
                    mod_prio.remove(mod)

            elif not ind[p_idx, m_idx]["fixed"]:

                used_pos = set(ind.get_used_positions_in_month(m_idx))
                used_tpos =
                ↪ set(ind.get_used_training_positions_in_month(m_idx))

                for chosen_mod in mod_prio:
                    combinations = p_data.get_allowed_combinations_for_module(
                        chosen_mod
                    )
                    if combinations:
                        allowed_combinations = [
                            i
                            for i in combinations
                            if i.position_id not in used_pos
                            and i.training_position_id not in used_tpos
                        ]
                        if allowed_combinations:
                            combination = allowed_combinations[0]

                            ind[p_idx, m_idx] = combination.to_assignment()
                            break
    return ind

def fill_all_possible_indices_with_unused_combinations_v2(
    ind: Individual, indices: dict[int, list[int]], calc_data: CalculationData
):
    """
    insert combination preferably in same department or hospital as the
    ↪ previous one
    """

    person_indices = indices.keys()

    for p_idx in person_indices:
        p_data = calc_data.person_calculation_data.get_person_calc_data_by_ind
        ↪ ex(p_idx)
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
517
518         m_indices = indices[p_idx]
519
520         for m_idx in m_indices:
521             used_pos = set(ind.get_used_positions_in_month(m_idx))
522             used_tpos = set(ind.get_used_training_positions_in_month(m_idx))
523             unused_pos = list(set(p_data.allowed_positions) - used_pos)
524             if (
525                 m_idx > 0
526                 and ind[p_idx, m_idx - 1]["training_position"] > 0
527                 or m_idx >= ind.n_months - 1
528             ):
529                 neighboring_asg = Assignment(
530                     *ind[p_idx, m_idx - 1]
531                 )
532             else:
533                 neighboring_asg = Assignment(*ind[p_idx, m_idx + 1])
534
535             neigh_dep = calc_data.position_department_mapping.get(
536                 neighboring_asg.position, 0
537             )
538             neigh_hosp = calc_data.position_hospital_mapping.get(
539                 neighboring_asg.position, 0
540             )
541
542             all_combs = []
543             combination = None
544             for pos in unused_pos:
545                 combs = p_data.get_allowed_combinations_with_position(pos)
546                 if combs is not None:
547                     allowed_combs = [
548                         c for c in combs if c.training_position_id not in
                            ↪ used_tpos
549                     ]
550                     all_combs.extend(allowed_combs)
551
552             random.shuffle(all_combs)
553             if all_combs:
554                 combs = [
555                     (
556                         c,
557                         calc_data.training_position_department_mapping.get(
558                             c.training_position_id, 0
559                         ),
560                         calc_data.position_hospital_mapping.get(c.position_id),
561                         0,
562                     )
563                     for c in all_combs
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
                    if c.training_position_id not in used_tpos
                ]
                combs.sort(key=lambda x: (x[1] != neigh_dep, x[2] !=
                ↪  neigh_hosp))
                combination = combs[0][0]

                ind[p_idx, m_idx] = combination.to_assignment()

    return ind


def get_assigned_modules_for_person(
    ind: Individual, p_idx: int, calc_data: CalculationData
) -> list[int]:
    p_data =
    ↪  calc_data.person_calculation_data.get_person_calc_data_by_index(p_idx)
    modules = []
    module_tracker = {
        id: m.months_remaining for id, m in
        ↪  p_data.unfinished_module_info.items()
    }

    for i in ind[p_idx, :]:
        asg = Assignment(*i)
        combination = p_data.get_combination(asg.position,
        ↪  asg.training_position)
        m_found = False
        if combination is not None and combination.education_modules_available:

            for m in combination.education_modules_available:
                if m in module_tracker.keys() and module_tracker.get(m) > 0:
                    m_found = True
                    modules.append(m)
                    module_tracker[m] -= 1
                    break
        if not m_found:
            modules.append(0)

    return modules


def insert_unused_combinations(
    ind: Individual, calc_data: CalculationData, fraction: float
):

    NUM = max(1, round(ind.n_persons * fraction))
    indices = ind.get_indices_without_training()
    person_indices = list({i[0] for i in indices})
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
608     random.shuffle(person_indices)
609     max_idx = min(len(person_indices), NUM)
610     ind = fill_all_possible_indices_with_unused_combinations(
611         ind, person_indices[:max_idx], calc_data
612     )
613 
614     return ind
615 
616 
617 def infer_unassigned_positions_v3(ind:Individual,
    ↪  calculation_data:CalculationData) -> Individual:
618     """
619     changed v2 to also considering all other positions available
620     """
621     month_indices = list(range(ind.n_months))
622     person_indices = list(range(ind.n_persons))
623     all_pos = set(calculation_data.position_department_mapping.keys())
624     all_pos.discard(0)
625     random.shuffle(person_indices)
626 
627     for p_idx in person_indices:
628         p_data = calculation_data.person_calculation_data.get_person_calc_data
            ↪  _by_index(p_idx)
629         for m_idx in month_indices:
630             asg = Assignment(*ind[p_idx,m_idx])
631             if asg.position == 0 and not asg.fixed:
632                 used_pos = set(ind.get_used_positions_in_month(m_idx))
633                 allowed_positons = list(set(p_data.allowed_positions) -
                    ↪  used_pos)
634 
635                 if not allowed_positons:
636                     allowed_positons = list(all_pos - used_pos)
637                 if allowed_positons:
638 
639                     if m_idx > 0:
640                         prev_idx = m_idx-1
641                     elif m_idx < ind.n_months -1:
642                         prev_idx = random.choice((m_idx+1, m_idx-1))
643                     else:
644 
645                         prev_idx = m_idx+1
646 
647                     prev_asg = Assignment(*ind[p_idx,prev_idx])
648                     prev_hosp = calculation_data.position_hospital_mapping.get
                        ↪  (prev_asg.position,-1)
649                     prev_dep = calculation_data.position_department_mapping.ge
                        ↪  t(prev_asg.position,-1)
650 
```

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

```
651
652
653                     mapping = {p:(calculation_data.position_department_mapping
                        ↪  .get(p,0),calculation_data.position_hospital_mapping.g
                        ↪  et(p,0)) for p in
                        ↪  allowed_positons}
654
655                     allowed_positons.sort(key=lambda x:(mapping[x][0] !=
                        ↪  prev_dep, mapping[x][1] != prev_hosp))
656
657                     pos = allowed_positons[0]
658                     ind[p_idx,m_idx] = Assignment(pos,0,False)
659
660     return ind
661
662 def infer_unassigned_training_positions(ind:Individual,
    ↪  calculation_data:CalculationData):
663     month_indices = list(range(ind.n_months))
664     person_indices = list(range(ind.n_persons))
665
666     random.shuffle(person_indices)
667
668     for p_idx in person_indices:
669         p_data = calculation_data.person_calculation_data.get_person_calc_data
            ↪  _by_index(p_idx)
670         for m_idx in month_indices:
671             asg = ind[p_idx,m_idx]
672             if asg["training_position"] == 0 and asg["position"] != 0 and not
                ↪  asg["fixed"]:
673                 used_tpos =
                    ↪  set(ind.get_used_training_positions_in_month(m_idx))
674                 pos_id = asg["position"]
675                 alllowed_combs =
                    ↪  p_data.get_allowed_combinations_with_position(pos_id)
676                 if alllowed_combs is not None:
677                     allowed_t_positons = list(set([i.training_position_id for
                        ↪  i in alllowed_combs]) - used_tpos)
678                     if allowed_t_positons:
679                         tpos = random.choice(allowed_t_positons)
680                         ind[p_idx,m_idx] = Assignment(pos_id,tpos,False)
681
682     return ind
```

Code A.2: Der entwickelte Mutationsoperator inklusive aller Teilfunktionen

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Literaturverzeichnis

[1] Sebastian Kraul, „Annual scheduling for anesthesiology medicine residents in task-related programs with a focus on continuity of care," *Flexible Services and Manufacturing Journal*, Jg. 32, Nr. 1, S. 181–212, 2020, 3, ISSN: 19366590. DOI: 10.1007/S10696-019-09365-4/TABLES/8. Adresse: https://link.springer.com/article/10.1007/s10696-019-09365-4.

[2] Prof. Dr. Thomas Bartscher. „Definition: Personalplanung." (2018), Adresse: https://wirtschaftslexikon.gabler.de/definition/personalplanung-43721/version-267047 (besucht am 01. 07. 2022).

[3] ——, „Definition: Personaleinsatz." (2018), Adresse: https://wirtschaftslexikon.gabler.de/definition/personaleinsatz-46065/version-269351 (besucht am 01. 07. 2022).

[4] T. H. Cormen, C. E. Leiserson, R. L. Rivest und C. Stein, *Introduction to algorithms*, eng, Fourth edition. Cambridge, Massachusett: The MIT Press, 2022, 11276 S., Cormen, Thomas H. (VerfasserIn) Leiserson, Charles Eric (VerfasserIn) Rivest, Ronald Linn (VerfasserIn) Stein, Clifford (VerfasserIn), ISBN: 9780262367509.

[5] D. E. Knuth, *The art of computer programming*, eng, Third edition, forty-fifth printing [ff.], Ser. The classiv work. Boston: Addison-Wesley, 2021, 652 S., Knuth, Donald Ervin (VerfasserIn), ISBN: 9780201896831.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

[6] Bundesministerium für Gesundheit, *Verordnung der Bundesministerin für Gesundheit über die Ausbildung zur Ärztin für Allgemeinmedizin/zum Arzt für Allgemeinmedizin und zur Fachärztin/zum Facharzt*, Ärztinnen-/Ärzte-Ausbildungsordnung 2015 - ÄAO 2015, RIS, Hrsg., Version 22.04.2022, 2015. Adresse: https://www.ris.bka.gv.at/GeltendeFassung.wxe?Abfrage=Bundesnormen&Gesetzesnummer=20009186 (besucht am 22. 04. 2022).

[7] Nationalrat Österreich, *Bundesgesetz über die Ausübung des ärztlichen Berufes und die Standesvertretung der Ärzte (Ärztegesetz 1998 – ÄrzteG 1998)*, ÄrzteG 1998, Version 08-08-2022, 1998.

[8] Wiener Gesundheitsverbund. „Nach dem Studium," Wiener Gesundheitsverbund. (2022), Adresse: https://ausbildung.gesundheitsverbund.at/aerztliche-ausbildung/nach-dem-studium/#toggle-id-17 (besucht am 22. 04. 2022).

[9] Österreichische Ärztekammer. „Strukturdiagramm Ärzteschaft - Kopfzahlen der Kammermitglieder." (2020), (besucht am 11. 08. 2022).

[10] Rechnungshof Österreich, *Bericht des Rechnungshofes: Ärzteausbildung*, 2021. Adresse: https://www.rechnungshof.gv.at/rh/home/home/A-rzteausbildung.pdf (besucht am 09. 08. 2022).

[11] Bundesministerium für Soziales, Gesundheit, Pflege und Konsumentenschutz. „Ärztinnen- und Ärztemonitoring." (2022), Adresse: https://www.sozialministerium.at/Themen/Gesundheit/Medizin-und-Gesundheitsberufe/Aerztinnen--und-Aerzteausbildung/Aerztinnen--und-Aerztemonitoring.html (besucht am 11. 08. 2022).

[12] ÖÄK. „Mehr Medizin-Absolventen sind nicht die Lösung gegen Ärztemangel, Solange die Arbeitsbedingungen und die Arztausbildung in Österreich nicht verbessert werden, werden damit nur noch mehr Medizin-Absolventen Österreich nach ihrem Studium verlassen." (2019), Adresse: https://www.aerztekammer.at/home/-/asset_publisher/topnews/content/id/246719 (besucht am 11. 08. 2022).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[13] ——, „ÖÄK-Mayer: Bundesländer blockieren die Ärzteausbildung, Vizepräsident der Österreichischen Ärztekammer bemängelt fehlenden politischen Willen, für unbesetzte Ausbildungsplätze auch die nötigen Dienstposten zu schaffen.“ ÖÄK, Hrsg., ÖÄK. (2022), Adresse: https://www.aerztekammer.at/home/-/asset_publisher/topnews/content/pa-mayer-politik-blockiert-aerzteausbildung/261766 (besucht am 11. 08. 2022).

[14] K. Peffers, T. Tuunanen, M. A. Rothenberger und S. Chatterjee, „A Design Science Research Methodology for Information Systems Research,“ *Journal of Management Information Systems*, Jg. 24, Nr. 3, S. 45–77, 2007, ISSN: 0742-1222. DOI: 10.2753/MIS0742-1222240302.

[15] Sebastian Kraul, Andreas Fügener, Jens O. Brunner und Manfred Blobner, „A robust framework for task-related resident scheduling,“ *European Journal of Operational Research*, Jg. 276, Nr. 2, S. 656–675, 2019, 7, ISSN: 03772217. DOI: 10.1016/J.EJOR.2019.01.034.

[16] L. S. Franz und J. L. Miller, „Scheduling Medical Residents to Rotations: Solving the Large-Scale Multiperiod Staff Assignment Problem,“ *Operations Research*, Jg. 41, Nr. 2, S. 269–279, 1993, ISSN: 0030-364X. DOI: 10.1287/opre.41.2.269.

[17] J. L. Denson, M. McCarty, Y. Fang, A. Uppal und L. Evans, „Increased Mortality Rates During Resident Handoff Periods and the Effect of ACGME Duty Hour Regulations,“ eng, *The American Journal of Medicine*, Jg. 128, Nr. 9, S. 994–1000, 2015, Journal Article, ISSN: 0002-9343. DOI: 10.1016/j.amjmed.2015.03.023. eprint: 25863148. Adresse: https://www.sciencedirect.com/science/article/pii/S000293431500279X.

[18] C. A. Coello Coello, *Evolutionary Algorithms for Solving Multi-Objective Problems*, eng, 2. Aufl., unter Mitarb. von D. A. van Veldhuizen und D. E. Goldberg. Boston: Springer, 2007, 1599 S., Coello Coello, Carlos A (VerfasserIn) Van Veldhuizen, David A (MitwirkendeR) Goldberg, David E (MitwirkendeR), ISBN: 9781475751864. DOI: Coello.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[19] M. Klammer, J. N. Dybowski, D. Hoffmann und C. Schaab, „Pareto Optimization Identifies Diverse Set of Phosphorylation Signatures Predicting Response to Treatment with Dasatinib," eng, *PloS one*, Jg. 10, Nr. 6, e0128542, 2015, Journal Article Research Support, Non-U.S. Gov't Journal Article Research Support, Non-U.S. Gov't. DOI: 10.1371/journal.pone.0128542. eprint: 26083411.

[20] D. Simon, *Evolutionary optimization algorithms*, en. Chichester: Wiley-Blackwell, 2013, 1 online resource (1 volume), Simon, Dan, (author.), ISBN: 9781118659502.

[21] F. Rothlauf, *Design of modern heuristics, Principles and application*, Ser. Natural computing series. Heidelberg und New York: Springer, 2011, xi, 267, ISBN: 978-3-540-72961-7. DOI: 10.1007/978-3-540-72962-4.

[22] K. Weicker, *Evolutionäre Algorithmen*, ger, 2., überarb. und erw. Aufl., Ser. Lehrbuch Informatik. Wiesbaden: Teubner, 2007, 313 S., ISBN: 3835102192.

[23] N. Kliewer, „Komplexitätstheorie," in *Gronau, Norbert ; Becker, Jörg ; Kliewer, Natalia ; Leimeister, Jan Marco ; Overhage, Sven (Herausgeber): Enzyklopädie der Wirtschaftsinformatik – Online-Lexikon*, [Online; Stand 27. April 2022], Berlin : GITO, 2019. Adresse: https://wi-lex.de/index.php/lexikon/technologische-und-methodische-grundlagen/informatik-grundlagen/komplexitaetstheorie/.

[24] Frankfurter Allgemeine Zeitung GmbH. „Eine der härtesten Nüsse der Mathematik: Was es mit P $\neq$ NP auf sich hat. - Bild 2 von 3." Copyright: Frankfurter Allgemeine Zeitung GmbH 2001 - 2022, Frankfurter Allgemeine Zeitung GmbH. (2017), Adresse: https://www.faz.net/aktuell/wissen/computer-mathematik/mathematik-was-es-mit-p-np-auf-sich-hat-15168441/mathematik-was-es-mit-p-np-15168507.html (besucht am 09. 05. 2022).

[25] O. Kramer, *Genetic Algorithm Essentials*, Ser. Studies in Computational Intelligence. Cham: Springer International Publishing, Imprint und Springer, 2017, Bd. 679, 92 S., ISBN: 978-3-319-52155-8. DOI: 10.1007/978-3-319-52156-5.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[26] U. Aickelin und K. A. Dowsland, „An indirect Genetic Algorithm for a nurse-scheduling problem," *Computers & Operations Research*, Jg. 31, Nr. 5, S. 761–778, 2004, PII: S0305054803000340, ISSN: 0305-0548. DOI: 10.1016/S0305-0548(03)00034-0.

[27] Jose Allen Lima, Nuno Gracias, Henrique Pereira und Agostinho Rosa, „Fitness function design for genetic algorithms in cost evaluation based problems," *Proceedings of the IEEE Conference on Evolutionary Computation*, S. 207–212, 1996. DOI: 10.1109/ICEC.1996.542362.

[28] Broos Maenhout und Mario Vanhoucke, „Comparison and hybridization of crossover operators for the nurse scheduling problem," *Annals of Operations Research 2007 159:1*, Jg. 159, Nr. 1, S. 333–353, 2007, 12, ISSN: 1572-9338. DOI: 10.1007/S10479-007-0268-Z. Adresse: https://link.springer.com/article/10.1007/s10479-007-0268-z.

[29] D. H. Wolpert und W. G. Macready, „No free lunch theorems for optimization," *IEEE Transactions on Evolutionary Computation*, Jg. 1, Nr. 1, S. 67–82, 1997, ISSN: 1089778X. DOI: 10.1109/4235.585893.

[30] D. Simon, *Evolutionary optimization algorithms*, en. Chichester: Wiley-Blackwell, 2013, 1 online resource (1 volume), Simon, Dan, (author.), ISBN: 9781118659502.

[31] B. Heinzl, „Methods for hybrid modeling and simulation-based optimization in energy-aware production planning," en, TU Wien, 2020. Adresse: 10.34726/HSS.2020.74400.

[32] T. Sobottka, „Eine anwendungsorientierte simulationsbasierte Methode, unter Berücksichtigung von Energieeffizienz, in der optimierenden Planung von Produktion und Logistik, Eine anwendungsorientierte simulationsbasierte Methode, unter Berücksichtigung von Energieeffizienz, in der optimierenden Planung von Produktion und Logistik," de, TU Wien, 2017. Adresse: 10.34726/hss.2017.50381.

[33] K. Evers, *Simulationsgestützte Belegungsplanung in der Multiressourcen-Montage*, ger, 2002. DOI: 10.15488/5982.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[34] X.-S. Yang, *Engineering Optimization*. Hoboken, NJ, USA: John Wiley & Sons, Inc, 2010, ISBN: 9780470640425. DOI: 10.1002/9780470640425.

[35] J. Shi und Q. Zhang, „A new cooperative framework for parallel trajectory-based metaheuristics," *Applied Soft Computing*, Jg. 65, S. 374–386, 2018, ISSN: 1568-4946. DOI: 10.1016/j.asoc.2018.01.022. Adresse: https://www.sciencedirect.com/science/article/pii/S1568494618300280.

[36] C. Blum und A. Roli, „Hybrid Metaheuristics: An Introduction," in *Hybrid metaheuristics, An emerging approach to optimization / Christian Blum (eds.) ... [et al.]* Ser. Studies in computational intelligence, 1860-949X v. 114, C. Blum, Hrsg., Bd. 114, Berlin: Springer, 2008, S. 1–30, ISBN: 978-3-540-78294-0. DOI: 10.1007/978-3-540-78295-7_1.

[37] S. Nesmachnow, „An overview of metaheuristics: accurate and efficient methods for optimisation," *International Journal of Metaheuristics*, Jg. 3, Nr. 4, S. 320, 2014, ISSN: 1755-2176. DOI: 10.1504/IJMHEUR.2014.068914.

[38] M. Gerdts und F. Lempio, *Mathematische Optimierungsverfahren des Operations Research*, ger, unter Mitarb. von F. Lempio, Ser. De-Gruyter-Studium. Berlin und New York, NY: De Gruyter, 2011, 527 S., Gerdts, Matthias (VerfasserIn) Lempio, Frank (MitwirkendeR) Gerdts, Matthias (Verfasser) Lempio, Frank (Verfasser), ISBN: 9783110249989. Adresse: https://ebookcentral.proquest.com/lib/kxp/detail.action?docID=765878.

[39] J.-P. Thommen und M. Siepermann. „Gabler Wirtschaftslexikon - Heuristik." (), Adresse: https://wirtschaftslexikon.gabler.de/definition/heuristik-34474/version-257976.

[40] Volker Nissen, *Volker Nissen Einführung in Evolutionäre Algorithmen Einführung in Evolutionäre Algorithmen*. 1997, ISBN: 978-3-322-93861-9. DOI: 10.1007/978-3-322-93861-9.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[41] R. Klein, *Planung und Entscheidung, Konzepte, Modelle und Methoden einer modernen betriebswirtschaftlichen Entscheidungsanalyse*, ger, 2nd ed., unter Mitarb. von A. Scholl, Ser. Vahlens Handbücher der Wirtschafts- und Sozialwissenschaften. München: Franz Vahlen, 2012, 1557 S., Klein, Robert (VerfasserIn) Scholl, Armin (MitwirkendeR), ISBN: 9783800638857.

[42] W. Domschke und A. Scholl, *Heuristische verfahren*, 2006. Adresse: https://core.ac.uk/download/pdf/7188814.pdf.

[43] M. Lübecke. „Gabler Wirtschaftslexikon - Metaheuristik." (), Adresse: https://wirtschaftslexikon.gabler.de/definition/metaheuristik-53837/version-276902.

[44] F. Glover, „Future paths for integer programming and links to artificial intelligence," *Computers & Operations Research*, Jg. 13, Nr. 5, S. 533–549, 1986, PII: 0305054886900481, ISSN: 0305-0548. DOI: 10.1016/0305-0548(86)90048-1.

[45] G. Jaradat, M. Ayob und I. Almarashdeh, „The effect of elite pool in hybrid population-based meta-heuristics for solving combinatorial optimization problems," *Applied Soft Computing*, Jg. 44, S. 45–56, 2016, PII: S1568494616000041, ISSN: 1568-4946. DOI: 10.1016/j.asoc.2016.01.002.

[46] C. Voudouris, E. P. Tsang und A. Alsheddy, „Guided Local Search," in *Handbook of metaheuristics*, Ser. International Series in Operations Research & Management Science 1460, M. Gendreau und J.-Y. Potvin, Hrsg., 2nd ed., Bd. 146, New York: Springer, 2010, S. 321–361, ISBN: 978-1-4419-1663-1. DOI: 10.1007/978-1-4419-1665-5_11.

[47] S. Kirkpatrick, C. D. Gelatt und M. P. Vecchi, „Optimization by simulated annealing," eng, *Science (New York, N.Y.)*, Jg. 220, Nr. 4598, S. 671–680, 1983, Journal Article, ISSN: 0036-8075. DOI: 10.1126/science.220.4598.671. eprint: 17813860.

[48] N. Mladenović und P. Hansen, „Variable neighborhood search," *Computers & Operations Research*, Jg. 24, Nr. 11, S. 1097–1100, 1997, PII: S0305054897000312, ISSN: 0305-0548. DOI: 10.1016/S0305-0548(97)00031-2.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[49] H. R. Lourenço, O. C. Martin und T. Stützle, „Iterated Local Search," in *Handbook of metaheuristics*, Ser. International Series in Operations Research & Management Science 57, F. Glover und G. A. Kochenberger, Hrsg., Bd. 57, Boston: Kluwer Academic Publishers, 2003, S. 320–353, ISBN: 1-4020-7263-5. DOI: 10.1007/0-306-48056-5_11.

[50] J. Kennedy und R. Eberhart, „Particle swarm optimization," in *IEEE International Conference on Neural Networks, 1995. Proceedings*, (Perth, WA, Australia), IEEE / Institute of Electrical and Electronics Engineers Incorporated, 1995, S. 1942–1948, ISBN: 0-7803-2768-3. DOI: 10.1109/ICNN.1995.488968.

[51] M. Dorigo und G. Di Caro, „Ant colony optimization: a new meta-heuristic," in *Proceedings of the 1999 Congress on Evolutionary Computation, CEC99, July 6-9, 1999, Mayflower Hotel, Washington, D.C., USA*, (Washington, DC, USA), IEEE Neural Networks Council, Evolutionary Programming Society und Institution of Electrical Engineers, New York: IEEE, 1999, S. 1470–1477, ISBN: 0-7803-5536-9. DOI: 10.1109/CEC.1999.782657.

[52] M. Dorigo, V. Maniezzo und A. Colorni, „Ant system: optimization by a colony of cooperating agents," eng, *IEEE transactions on systems, man, and cybernetics. Part B, Cybernetics : a publication of the IEEE Systems, Man, and Cybernetics Society*, Jg. 26, Nr. 1, S. 29–41, 1996, Journal Article, ISSN: 1083-4419. DOI: 10.1109/3477.484436. eprint: 18263004.

[53] M. Dorigo, M. Birattari und T. Stutzle, „Ant colony optimization," *IEEE Computational Intelligence Magazine*, Jg. 1, Nr. 4, S. 28–39, 2006, ISSN: 1556-603X. DOI: 10.1109/MCI.2006.329691.

[54] J. H. Holland, *Adaptation in natural and artificial systems, An introductory analysis with applications to biology, control, and artificial intelligence*, en, 2. print, Ser. Complex adaptive systems. Cambridge, Mass.: MIT Press, 1993, ISBN: 9780262581110.

[55] Y. Bengio, A. Lodi und A. Prouvost, „Machine learning for combinatorial optimization: A methodological tour d'horizon," *European Journal of Operational Research*, Jg. 290, Nr. 2, S. 405–421, 2021, PII: S0377221720306895, ISSN: 03772217. DOI: 10.1016/j.ejor.2020.07.063.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[56] N. Mazyavkina, S. Sviridov, S. Ivanov und E. Burnaev, „Reinforcement learning for combinatorial optimization: A survey," *Computers & Operations Research*, Jg. 134, S. 105 400, 2021, PII: S0305054821001660, ISSN: 0305-0548. DOI: 10.1016/j.cor.2021.105400.

[57] R. S. Sutton und A. Barto, *Reinforcement learning, second edition, An introduction*, eng, Second edition, Ser. Adaptive computation and machine learning. Cambridge, Massachusetts und London, England: The MIT Press, 2018, 1590 S., Sutton, Richard S. (VerfasserIn) Barto, Andrew (VerfasserIn), ISBN: 9780262352703.

[58] R. Zhang, A. Prokhorchuk und J. Dauwels, „Deep Reinforcement Learning for Traveling Salesman Problem with Time Windows and Rejections," in *2020 International Joint Conference on Neural Networks (IJCNN), 2020 conference proceedings*, (Glasgow, United Kingdom, 2020), Institute of Electrical and Electronics Engineers, IEEE Computational Intelligence Society und International Neural Network Society, Piscataway, NJ, USA: IEEE, 2020, S. 1–8, ISBN: 978-1-7281-6926-2. DOI: 10.1109/IJCNN48605.2020.9207026.

[59] F. Ansari und K. Schenkelberg, *Problem Solving in the Digital World: Synoptic Formalism, Incrementalism and Heuristics*, unter Mitarb. von P. A. Laplante. Taylor & Francis und New York. Adresse: https://repositum.tuwien.at/handle/20.500.12708/29501.

[60] Chi Way Wang, Lei Ming Sun, Ming Hui Jin u. a., „A genetic algorithm for resident physician scheduling problem," *Proceedings of GECCO 2007: Genetic and Evolutionary Computation Conference*, S. 2203–2210, 2007. DOI: 10.1145/1276958.1277380.

[61] M. Erhard, J. Schoenfelder, A. Fügener und J. O. Brunner, „State of the art in physician scheduling," *European Journal of Operational Research*, Jg. 265, Nr. 1, S. 1–18, 2018, PII: S0377221717305787, ISSN: 03772217. DOI: 10.1016/j.ejor.2017.06.037.

[62] Amy Cohn, Sarah Root, Carisa Kymissis, Justin Esses und Niesha Westmoreland, „Scheduling Medical Residents at Boston University School of Medicine," *Interfaces*, Jg. 39, Nr. 3, S. 186–195, 2009, ISSN: 00922102. Adresse: http://www.jstor.org/stable/25622796 (besucht am 06. 04. 2022).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[63] J. Beliën und E. Demeulemeester, „Scheduling trainees at a hospital department using a branch-and-price approach," *European Journal of Operational Research*, Jg. 175, Nr. 1, S. 258–278, 2006, PII: S0377221705004248, ISSN: 03772217. DOI: 10.1016/j.ejor.2005.04.028.

[64] Ruben A. Proano und Akshit Agarwal, „Scheduling internal medicine resident rotations to ensure fairness and facilitate continuity of care," *Health care management science*, Jg. 21, Nr. 4, S. 461–474, 2018, 12, ISSN: 1386-9620. DOI: 10.1007/S10729-017-9403-9. eprint: 28500408. Adresse: https://pubmed.ncbi.nlm.nih.gov/28500408/.

[65] M. Tamiz, D. Jones und C. Romero, „Goal programming for decision making: An overview of the current state-of-the-art," *European Journal of Operational Research*, Jg. 111, Nr. 3, S. 569–581, 1998, PII: S0377221797003172, ISSN: 03772217. DOI: 10.1016/S0377-2217(97)00317-2.

[66] K. A. Dowsland, „Nurse scheduling with tabu search and strategic oscillation," *European Journal of Operational Research*, Jg. 106, Nr. 2-3, S. 393–407, 1998, PII: S0377221797002816, ISSN: 03772217. DOI: 10.1016/S0377-2217(97)00281-6.

[67] E. K. Burke, P. de Causmaecker, G. V. Berghe und H. van Landeghem, „The State of the Art of Nurse Rostering," En;en, *Journal of Scheduling*, Jg. 7, Nr. 6, S. 441–499, 2004, ISSN: 1099-1425. DOI: 10.1023/B:JOSH.0000046076.75950.0b. Adresse: https://link.springer.com/article/10.1023/b:josh.0000046076.75950.0b.

[68] Ahmad Jan, Masahito Yamamoto und Azuma Ohuchi, „Evolutionary Algorithms for Nurse Scheduling Problem,"

[69] H. Kawanaka, K. Yamamoto, T. Yoshikawa, T. Shinogi und S. Tsuruoka, „Genetic algorithm with the constraints for nurse scheduling problem," in *Proceedings of the 2001 Congress on Evolutionary Computation (IEEE Cat. No.01TH8546)*, (Seoul, South Korea, 2001), IEEE, 2001, S. 1123–1130, ISBN: 0-7803-6657-3. DOI: 10.1109/CEC.2001.934317.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[70] Margarida Moz und Margarida Vaz Pato, „A genetic algorithm approach to a nurse rerostering problem," *Computers & Operations Research*, Jg. 34, Nr. 3, S. 667–691, 2007, 3, ISSN: 0305-0548. DOI: 10.1016/J.COR.2005.03.019.

[71] Kundu S., Mahato M., Mahanty B., Acharyya S., *Comparative performance of simulated annealing and genetic algorithm in solving nurse scheduling problem*. 2008. Adresse: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.149.343&rep=rep1&type=pdf.

[72] C.-C. Tsai und S. H. Li, „A two-stage modeling with genetic algorithms for the nurse scheduling problem," *Expert Systems with Applications*, Jg. 36, Nr. 5, S. 9506–9512, 2009, PII: S0957417408008348, ISSN: 09574174. DOI: 10.1016/j.eswa.2008.11.049.

[73] Ruibin Bai, Edmund K. Burke, Graham Kendall, Jingpeng Li und Barry McCollum, „A hybrid evolutionary approach to the nurse rostering problem," *IEEE Transactions on Evolutionary Computation*, Jg. 14, Nr. 4, S. 580–590, 2010, 8, ISSN: 1089778X. DOI: 10.1109/TEVC.2009.2033583.

[74] W. J. Gutjahr und M. S. Rauner, „An ACO algorithm for a dynamic regional nurse-scheduling problem in Austria," *Computers & Operations Research*, Jg. 34, Nr. 3, S. 642–666, 2007, PII: S0305054805001139, ISSN: 0305-0548. DOI: 10.1016/j.cor.2005.03.018.

[75] F. Kamhuber, T. Sobottka, B. Heinzl, J. Henjes und W. Sihn, „An efficient hybrid multi-criteria optimization approach for rolling production smoothing of a European food manufacturer," *Computers & Industrial Engineering*, Jg. 147, S. 106 620, 2020, PII: S0360835220303545, ISSN: 0360-8352. DOI: 10.1016/j.cie.2020.106620. Adresse: https://www.sciencedirect.com/science/article/pii/S0360835220303545.

[76] T. Bäck, D. B. Fogel und Z. Michalewicz, *Basic algorithms and operators*, Ser. Evolutionary computation. Bristol: Institute of Physics Publishing, 2000, Bd. 1, ISBN: 0750306645.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[77] S. N. Sivanandam und S. N. Deepa, Hrsg., *Introduction to genetic algorithms*, Berlin und New York: Springer, 2007, xix, 442, ISBN: 978-3-540-73189-4. DOI: 10.1007/978-3-540-73190-0.

[78] M. V. E. Massone, *Cross-Sections for Transient Analyses: Development of a Genetic Algorithm for the Energy Meshing*, en. Karlsruhe, 2018. DOI: 10.5445/IR/1000083680. Adresse: https://www.researchgate.net/publication/326356402_Cross-Sections_for_Transient_Analyses_Development_of_a_Genetic_Algorithm_for_the_Energy_Meshing.

[79] A. E. Eiben und S. K. Smit, „Parameter tuning for configuring and analyzing evolutionary algorithms," *Swarm and Evolutionary Computation*, Jg. 1, Nr. 1, S. 19–31, 2011, PII: S2210650211000022, ISSN: 22106502. DOI: 10.1016/j.swevo.2011.02.001.

[80] I. Rechenberg, „Evolutionsstrategien," de, in *Simulationsmethoden in der Medizin und Biologie*, Springer, Berlin, Heidelberg, 1978, S. 83–114. DOI: 10.1007/978-3-642-81283-5_8. Adresse: https://link.springer.com/chapter/10.1007/978-3-642-81283-5_8.

[81] Wikipedia, Hrsg. „Fitness proportionate selection." en. (2020), Adresse: https://en.wikipedia.org/w/index.php?title=Fitness_proportionate_selection&oldid=979505220 (besucht am 23. 08. 2022).

[82] Wikipedia, Hrsg. „Stochastic universal sampling." en. (2021), (besucht am 23. 08. 2022).

[83] K. Deb, „Multi-objective Optimisation Using Evolutionary Algorithms: An Introduction," en, in *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing*, Springer, London, 2011, S. 3–34. DOI: 10.1007/978-0-85729-652-8_1. Adresse: https://link.springer.com/chapter/10.1007/978-0-85729-652-8_1.

[84] U. Mehboob, J. Qadir, S. Ali und A. Vasilakos, „Genetic algorithms in wireless networking: techniques, applications, and issues," *Soft Computing*, Jg. 20, Nr. 6, S. 2467–2501, 2016, PII: 2070, ISSN: 1432-7643. DOI: 10.1007/s00500-016-2070-9.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[85] M. Tenemaza, S. Lujan-Mora, A. de Antonio und J. Ramirez, „Improving Itinerary Recommendations for Tourists Through Metaheuristic Algorithms: An Optimization Proposal," *IEEE Access*, Jg. 8, S. 79 003–79 023, 2020. DOI: 10.1109/ACCESS.2020.2990348.

[86] L. Davis, Hrsg., *Handbook of genetic algorithms*, eng, 2. [print.], Davis, Lawrence (Hrsg.), New York: Van Nostrand Reinhold, 1991, 385 S., ISBN: 0442001738.

[87] SYSWERDA G., „Scheduling Optimization using Genetic Algorithms," *Handbook of Genetic Algorithms*, 1991. Adresse: https://ci.nii.ac.jp/naid/10021131794/.

[88] N. J. Radcliffe, „The algebra of genetic algorithms," *Annals of Mathematics and Artificial Intelligence*, Jg. 10, Nr. 4, S. 339–384, 1994, PII: BF01531276, ISSN: 1012-2443. DOI: 10.1007/BF01531276.

[89] A. M. Al-Shayea, M. Saleh, M. Alatefi und M. Ghaleb, „Scheduling Two Identical Parallel Machines Subjected to Release Times, Delivery Times and Unavailability Constraints," *Processes*, Jg. 8, Nr. 9, S. 1025, 2020, PII: pr8091025. DOI: 10.3390/pr8091025.

[90] M. Ehrgott, *Multicriteria optimization, 12 Tables*, eng, 2. ed. Berlin und Heidelberg: Springer, 2005, 323 S., ISBN: 9783540213987.

[91] A. E. Smith und D. W. Coit, „Penalty functions," in *Handbook of Evolutionary Computation*, IOP Publishing Ltd, 1995. DOI: 10.1887/0750308958/b386c48.

[92] Ö. Yeniay, „Penalty Function Methods for Constrained Optimization with Genetic Algorithms," *Mathematical and Computational Applications*, Jg. 10, Nr. 1, S. 45–56, 2005, PII: mca10010045. DOI: 10.3390/mca10010045.

[93] Richardson, J. T., Palmer, M. R., Liepins, G. E., Hilliard, M. R., „Some guidelines for genetic algorithms with penalty functions," in *Proceedings of the 3rd international conference on genetic algorithms*, S. 191–197.

[94] Python.org. „Welcome to Python.org." (2022), Adresse: https://www.python.org/ (besucht am 06. 09. 2022).

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[95] J. D. Hunter, „Matplotlib: A 2D Graphics Environment," *Computing in Science & Engineering*, Jg. 9, Nr. 3, S. 90–95, 2007, ISSN: 1521-9615. DOI: 10.1109/MCSE.2007.55.

[96] Plotly Technologies Inc. „Collaborative data science." Plotly Technologies Inc., Hrsg. (2015), Adresse: https://plot.ly (besucht am 07. 09. 2022).

[97] C. R. Harris, K. J. Millman, S. J. van der Walt u. a., „Array programming with NumPy," eng, *Nature*, Jg. 585, Nr. 7825, S. 357–362, 2020, Journal Article Research Support, Non-U.S. Gov't Review Journal Article Research Support, Non-U.S. Gov't Review. DOI: 10.1038/s41586-020-2649-2. eprint: 32939066.

[98] Jeff Reback, jbrockmendel, Wes McKinney u. a., *pandas-dev/pandas: Pandas 1.4.4*, Zenodo, 2022. DOI: 10.5281/ZENODO.7037953.

[99] „DEAP: Evolutionary algorithms made easy," 2012. Adresse: https://www.jmlr.org/papers/volume13/fortin12a/fortin12a.pdf.

[100] J. Blank und K. Deb, „Pymoo: Multi-Objective Optimization in Python," *IEEE Access*, Jg. 8, S. 89 497–89 509, 2020. DOI: 10.1109/ACCESS.2020.2990567.

[101] K. Deb, A. Pratap, S. Agarwal und T. Meyarivan, „A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, Jg. 6, Nr. 2, S. 182–197, 2002, ISSN: 1089-778X. DOI: 10.1109/4235.996017.

[102] Julian Blank. „pymoo - NSGA-II: Non-dominated Sorting Genetic Algorithm." (2022), Adresse: https://pymoo.org/algorithms/moo/nsga2.html (besucht am 15. 09. 2022).

[103] K. Deb und H. Jain, „An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems With Box Constraints," *IEEE Transactions on Evolutionary Computation*, Jg. 18, Nr. 4, S. 577–601, 2014, ISSN: 1089-778X. DOI: 10.1109/TEVC.2013.2281535.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

[104] D. Orvosh und L. Davis, „Using a genetic algorithm to optimize problems with feasibility constraints," in *Proceedings of the First IEEE Conference on Evolutionary Computation, June 27 - June 29, 1994, Walt Disney World Dolphin Hotel, Orlando, Florida*, (Orlando, FL, USA), World Congress on Computational Intelligence, Piscataway, NJ: IEEE, 1994, S. 548–553, ISBN: 0-7803-1899-4. DOI: 10.1109/ICEC.1994.350001.

[105] Q. Zhang und H. Li, „MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition," *IEEE Transactions on Evolutionary Computation*, Jg. 11, Nr. 6, S. 712–731, 2007, ISSN: 1089-778X. DOI: 10.1109/TEVC.2007.892759.

[106] Pyhton Software Foundation. „GlobalInterpreterLock - Python Wiki." (), Adresse: https://wiki.python.org/moin/GlobalInterpreterLock (besucht am 30. 09. 2022).

[107] Rust Foundation. „Rust Programming Language." (), Adresse: https://www.rust-lang.org/ (besucht am 30. 09. 2022).

[108] C. Helfferich, „Leitfaden- und Experteninterviews," in *Handbuch Methoden der empirischen Sozialforschung*, Ser. Handbuch, N. Baur und J. Blasius, Hrsg., Wiesbaden: Springer VS, 2014, S. 559–574, ISBN: 978-3-531-17809-7. DOI: 10.1007/978-3-531-18939-0_39.

[109] Volker Turau, *Algorithmische Graphentheorie*. Oldenbourg Wissenschaftsverlag, 2009, ISBN: 9783486593778. DOI: 10.1524/9783486593778.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Abbildungsverzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Tabellenverzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Formelverzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Abkürzungsverzeichnis

$\mathcal{NP}$	nichtdeterministisch-polynomiell
$\mathcal{P}$	polynomiell
ACO	Ant Colony Optimization (Ameisenalgorithmus)
AS	Ant System (Ameisensystem)
bzw.	beziehungsweise
CG	Column Generation (Spaltengenerierung)
DSRM	Design Science Research Methodology
EA	Evolutionary Algorithm (Evolutionärer Algorithmus)
ES	Evolution Strategy (Evolutionsstrategie)
GA	Genetic Algorithm (Genetischer Algorithmus)
GLS	Guided Local Search (Geführte Lokale Suche)
i.A.	im Allgemeinen
i.d.R	in der Regel
ILS	Iterated Local Search (Iterierte Lokale Suche)
IP	ganzzahlige Optimierung (Integer Programming)
KP	Rucksackproblem (Knapsack Problem)
LS	Local Search (Lokale Suche)
MIP	gemischt-ganzzahlige Optimierung (Mixed Integer Programming)
MOP	Mehrzieloptimierungsproblem (Multi-Objective Optimization Problem)
MS	Multi Start Search (Multi-Start Suche)

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

NFL	No Free Lunch Theorem (nichts-ist-umsonst-Theorem)
NSP	Nurse Scheduling Problem
OR	Operations Research (Unternehmensforschung)
OX	Order Crossover (Ordnungsrekombination)
PMX	Partially Mapped Crossover (Abbildungsrekombination)
PS	Pattern Search (Mustersuche)
PSO	Particle Swarm Optimization (Partikelschwarmoptimierung)
PX	Position Crossover (Positionsrekombination)
RL	Reinforcement Learning (Bestärkendes Lernen)
RSP	Resident Scheduling Problem
RWS	Roulette Wheel Selection (Fitnessproportionale Selektion)
SA	Simualted Annealing (Simulierte Abkühlung)
SOP	Einzieloptimierungsproblem (Single-Objective Optimization Problem)
SUS	Stochastisch Universelles Sampling (Stochastic Universal Sampling)
TS	Tabu Search (Tabu Suche)
TSP	Handlungsreisendenproblem (Traveling Salesman Problem)
VNS	Variable Neighborhood Search (Variable Nachbarschaftssuche)
vs.	versus
z.B.	zum Beispiel
ÄAM	Ärzt*innenausbildungsmanagement
ÄAO 2015	Ärztinnen-/Ärzte- Ausbildungsordnung 2015
ÄrzteG 1998	Bundesgesetz über die Ausübung des ärztlichen Berufes und die Standesvertretung der Ärzte
ÖÄK	Österreichische Ärztekammer

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.
TU Bibliothek Your knowledge hub WIEN

Algorithmen-Verzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.

Code-Verzeichnis

Die approbierte gedruckte Originalversion dieser Diplomarbeit ist an der TU Wien Bibliothek verfügbar
The approved original version of this thesis is available in print at TU Wien Bibliothek.