Entwicklung eines mehrkriteriellen Ant Colony Optimization-Verfahrens für die Lösung eines stochastischen und iterativen Projektplanungsproblems

Sebastian TERSTEGEN

Institut für Arbeitswissenschaft, RWTH Aachen University
Bergdriesch 27, D-52062 Aachen

Kurzfassung: In seinem Dissertationsvorhaben beschäftigt sich der Autor mit der Entwicklung einer auf metaheuristischen Optimierungsverfahren basierenden Methode für die Planung komplexer Projekte. Die zu entwickelnde Methode ergänzt und erweitert bestehende Methoden des Projektmanagements und Operations Research und ermöglicht die Lösung eines stochastischen Projektplanungsproblems. In diesem Beitrag werden die theoretischen Grundlagen sowie erste Ansätze der Modellbildung dargestellt.

Schlüsselwörter: Projektplanung, RCPSP, Ameisenalgorithmus

1. Herausforderungen der Projektplanung

Zunehmend werden Forschungs-, Entwicklungs- und Konstruktionsaufgaben in Unternehmen in Projektform durchgeführt. Das heißt, Vorhaben der Produkt-entwicklung oder Innovationsforschung werden unter sowohl zeitlich, finanziell als auch personell begrenzten Bedingungen strukturiert und bedürfen einer projekt-spezifischen Organisation (vgl. DIN 69901-5). Die Aufgabe des Projektmanagements besteht in der Definition, Planung und Steuerung dieser Projekte. Eine besonders gewichtige Stellung in dem Projektmanagement-Gesamtprozess nimmt dabei die Projektplanung ein. Die Projektplanung beinhaltet die Festlegung der zu erbringenden Leistungen und Projektergebnisse (Leistungsplanung), die Festlegung der Ausführungszeiten und Termine der Projektaufgaben (Ablauf- und Terminplanung) sowie die Planung von Aufwänden, Zuordnung begrenzter Arbeitspersonen auf Projektaufgaben und Ermittlung des Bedarfs an Finanzmitteln (Einsatz-, Kosten- und Finanzmittelplanung). Die Projektplanung schafft somit die strukturellen und organisatorischen Voraussetzungen für geplante Projekte eines Unternehmens und nimmt die Weichenstellungen für die erfolgreiche Projektdurchführung vor.

Aufgrund vielschichtiger wechselseitiger Abhängigkeiten zwischen Projektaufgaben, Arbeitspersonen sowie zeitlichen und finanziellen Ressourcen eines Innovationsprojekts sind Projektmanager immer weniger in der Lage, die komplexen und latent unsicheren Planungsprozesse zu bewältigen. Gerade die Ermittlung des Fertigstellungstermins eines Projekts – eine der wichtigsten Zielgrößen bei der Projektplanung – ist ein hochkomplexes Planungsproblem und stellt Projektmanager vor erhebliche Herausforderungen. Eine operative Unterstützung steht Projektmanagern in Form klassischer Methoden und Werkzeuge, wie Gantt-Charts und Netzplantechniken, zur Verfügung. Gantt-Charts lassen sich zwar intuitiv anwenden und geben eine schnelle Übersicht über den Ablaufplan. Sie erfüllen aber nicht die Anforderungen einer umfassenden Projektplanung und lassen sich bestenfalls für Projekte mit geringer Komplexität einsetzen. Netzplantechniken ermöglichen zwar die Ermittlung von Ablauf- und Terminplänen und die Identifizierung der kritischen und nichtkritischen Projektaufgaben (Reichert 2009). Allerdings werden dabei sehr restriktive Annahmen vorausgesetzt, sodass sich Netzplantechniken meist nur für azyklische Projektstrukturen mit statischen, fest determinierten Durchlaufzeiten und Anordnungsbeziehungen, die zwischen Projektaufgaben bestehen, anwenden lassen.

Klassische Methoden und Werkzeuge des Projektmanagements bieten Projektmanagern also nur eine geringe Unterstützung bei der Planung und operativen Steuerung komplexer Projekte unter Unsicherheit. Mithilfe von Project Scheduling Ansätzen aus dem Operations Research und entsprechenden Lösungs- und Optimierungsverfahren kann hingegen die stochastische Natur von Aufgaben in komplexen Projekten abgebildet werden.

2. Project Scheduling

Schon seit den 1950er Jahren wurden im Operations Research auf dem Gebiet der Projektplanung zahlreiche Modelle und Methoden entwickelt, die Projektmanager bei der effizienten Projektplanung großer, praxisrelevanter Projekte unterstützen (Zimmermann et al. 2010).

2.1 Resource-Constrained Project Scheduling Problem

Mit kombinatorischen Optimierungsproblemen wie dem sogenannten Ressourcen-beschränkten Projektplanungsproblem, engl. Resource-Constrained Project Scheduling Problem (RCPSP), werden Projektvorgänge, zeitliche Abhängigkeiten zwischen Vorgängen, Projektressourcen und zu lösende Zielfunktionen beschrieben. Ziele von Lösungsalgorithmen sind, eine möglichst effiziente Zuordnung der Ressourcen zu Vorgängen sowie einen optimalen terminlichen Ablauf der Vorgänge zu finden. Meist soll dabei die Gesamtprojektdauer minimiert werden. Zudem muss berücksichtigt werden, dass die Bearbeitungsreihenfolge der Vorgänge nicht beliebig erfolgen kann und dass Ressourcen nur in begrenzter Kapazität und in bestimmten Zeiträumen zum Einsatz zur Verfügung stehen.

Die grundlegenden Bestandteile des mathematischen RCPSP-Modells werden meist wie folgt definiert (Yang et al. 2001):

$\mathcal{J}$	Menge aller Vorgänge $j = 1,\ldots,J = \lvert\mathcal{J}\rvert$
$\mathcal{J}^+$	Menge $\mathcal{J}$ und zwei Dummy-Vorgänge 0 (Quelle) und J+1 (Senke)
$\mathcal{T}$	Zeithorizont unterteilt in gleichmäßig lange Zeitintervalle
d_j	Dauer des Vorgangs j, $d_0 = d_{J+1} = 0$
$\mathcal{P}_j$	Menge aller Vorgänger des Vorgangs j
$\mathcal{S}_j$	Menge aller Nachfolger des Vorgangs j
EST_j	Früheste mögliche Startzeit des Vorgangs j
LST_j	Späteste mögliche Endzeit des Vorgangs j
EFT_j	Früheste mögliche Startzeit des Vorgangs j
LFT_j	Späteste mögliche Endzeit des Vorgangs j
$\mathcal{R}$	Menge aller Ressourcen k

R_k Kapazität der Ressource k pro Zeitintervall

$r_{j,k}$ Bedarf des Vorgangs j an Ressource k, $r_{0,k} = r_{J+1,k} = 0\ \forall\ k \in \mathcal{R}$

Ein häufig verwendetes RCPSP-Modell wurde 1969 von Pritsker definiert (Hartmann 1999; Demeulemeester & Herroelen 2001):

$$min \sum_{t=EFT_{J+1}}^{LFT_{J+1}} t \cdot x_{J+1,t} \tag{1}$$

sodass

$$\sum_{t=EFT_j}^{LFT_j} x_{j,t} = 1 \qquad \forall\, j \in \mathcal{J}^+ \tag{2}$$

$$\sum_{t=EFT_h}^{LFT_h} t \cdot x_{h,t} \leq \sum_{t=EFT_j}^{LFT_j} (t - d_j) \cdot x_{j,t} \qquad \forall\, j \in \mathcal{J}^+, h \in \mathcal{P}_j \tag{3}$$

$$\sum_{t=1}^{J} r_{j,k} \cdot \sum_{b=min\{t,EFT_j\}}^{min\{t+d_j-1,LFT_j\}} x_{j,b} \leq R_k \qquad \forall\, k \in \mathcal{R}, t \in \mathcal{T} \tag{4}$$

$$x_{j,t} \in \{0,1\} \qquad \forall\, j \in \mathcal{J}^+, t \in \mathcal{T} \tag{5}$$

2.2 Erweitertes RCPSP-Modell mit stochastischen Einflussfaktoren

Bei der Entwicklung des mehrkriteriellen Optimierungsverfahrens wird das zur Verkürzung der Projektdauer weit verbreitete Organisationskonzept Concurrent Engineering berücksichtigt. Bei dieser Vorgehensweise, die insbesondere bei komplexen Produktentwicklungsprojekten angewendet wird, werden Projektvorgänge integriert und hochgradig parallel ausgeführt. Häufig müssen Projektvorgänge dann in Iterationsschleifen wiederholt bearbeitet werden, bis durch das iterative Vorgehen eine zufriedenstellende Qualität des Arbeitsergebnisses erreicht wurde.

Das in der Praxis häufig auftretende Concurrent Engineering wird erstmalig durch die Weiterentwicklung des mathematischen RCPSP-Modells nach Pritsker berücksichtigt. Dabei müssen verschiedene Iterationstypen unterschieden werden (vgl. Abbildung 1): Bei einer einfachen Iteration wird ein Vorgang wiederholt durchgeführt mit dem Ziel der Ergebnisverbesserung. Das Ergebnis eines Arbeitsschrittes dient als Ausgangspunkt für den nächsten Arbeitsschritt. Auf diese Weise nähert man sich schrittweise einer optimalen bzw. ausreichend guten Lösung, wobei die iterative Bearbeitung eines Vorgangs in diesem Fall keine Auswirkungen auf andere Vorgänge hat. Bei der Iteration mit einem zusätzlichen Vorgang muss während einer Iteration ein zusätzlicher Vorgang durchgeführt werden, der ohne Iteration nicht notwendig wäre. Bei einer Iteration zweiter Ordnung werden zunächst Vorgänger und Nachfolger sequentiell oder zeitlich überlappend durchgeführt und anschließend in einer Iterationsschleife in dergleichen Reihenfolge erneut bearbeitet.

Darüber hinaus werden weitere Fälle berücksichtigt, die an dieser Stelle aus

Gründen der Übersichtlichkeit nicht im Einzelnen dargestellt werden: Kombination aus einfacher Iteration und Iteration zweiter Ordnung, einfache Iterationen und Iterationen zweiter Ordnung mit vorzeitigem Abbruch der Bearbeitung der Vorgänge, parallele Bearbeitung von Vorgängen mit indirekter und direkter Beziehung zu einer Iteration.

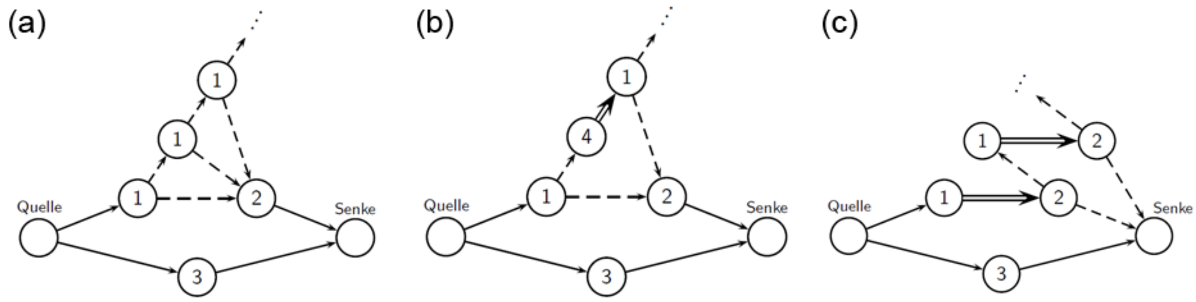

***Abbildung 1**: Darstellung verschiedener Iterationstypen in einem zyklenfreien (Netz-)Graphen: (a) Einfache Iteration, (b) Iteration mit einem zusätzlichen Vorgang, (c) Iteration zweiter Ordnung über zwei Vorgänge.*

Die informatorischen Abhängigkeiten zwischen Vorgängen werden in Form der vom Projektmanager festzulegenden Iterationswahrscheinlichkeiten erster und zweiter Ordnung abgebildet. Die Iterationswahrscheinlichkeit erster Ordnung $p_{2,1}$ entspricht der Wahrscheinlichkeit, dass ein Nachfolger die erneute Bearbeitung eines Vorgängers auslöst. Die Iterationswahrscheinlichkeit zweiter Ordnung $p_{1,2}$ entspricht der Wahrscheinlichkeit, dass der Nachfolger durch diese Iterationsschleife ebenfalls erneut bearbeitet werden muss (vgl. Abbildung 2).

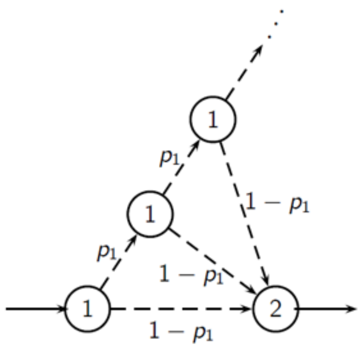

***Abbildung 2**: Darstellung der Iterationswahrscheinlichkeiten in einem zyklenfreien (Netz-)Graphen am Beispiel einer einfachen Iteration.*

Die erwartete Bearbeitungsdauer eines Vorgangs j ergibt sich aus den Bearbeitungsdauern der möglichen Iterationsschleifen, gewichtet mit ihren Wahrscheinlichkeiten, falls die Anzahl der Iterationen durch $N \in \mathbb{N}$ beschränkt ist:

$$d_{j,N}^{*} = \sum_{i=0}^{N} p_j(i) \cdot d_j(i) \tag{6}$$

Das RCPSP-Modell nach Pritsker (siehe Kapitel 2.1) wird dahingehend erweitert, dass die Dauer d_j in den Nebenbedingungen (3) und (4) durch die erwartete Bearbeitungsdauer $d_{j,N}^{*}$ ersetzt wird:

$$\sum_{t=EFT_h}^{LFT_h} t \cdot x_{h,t} \leq \sum_{t=EFT_j}^{LFT_j} \left(t - d_j^*\right) \cdot x_{j,t} \qquad \forall\, j \in \mathcal{J}^+, h \in \mathcal{P}_j \tag{7}$$

$$\sum_{t=1}^{J} r_{j,k} \cdot \sum_{b=min\{t,EFT_j\}}^{min\{t+d_j^*-1,LFT_j\}} x_{j,b} \leq R_k \qquad \forall\, k \in \mathcal{R}, t \in \mathcal{T} \tag{8}$$

Neben informatorischen Abhängigkeiten zwischen Vorgängen in Form von Iterationswahrscheinlichkeiten besteht ein weiterer wesentlicher Aspekt des Concurrent Engineering in der parallelen bzw. zeitlich überlappenden Bearbeitung von Vorgängen. Der vom Projektmanager festzulegende Grad der Überlappung $\alpha_{i,j}$ bestimmt den Anteil der Dauer des Vorgangs i, der von Vorgang j überlappt werden darf. Die Bearbeitung des Vorgangs i muss somit zu einem Grad von $1 - \alpha_{i,j}$ abgeschlossen sein, bevor Vorgang j begonnen werden kann. Die Nebenbedingung (3) im RCPSP-Modell nach Pritsker (siehe Kapitel 2.1) wird durch die erweiterte Nebenbedingung (9) ersetzt:

$$\sum_{t=EFT_h}^{LFT_h} \left(t - \alpha_{h,j} \cdot d_h\right) \cdot x_{h,t} \leq \sum_{t=EFT_j}^{LFT_j} \left(t - d_j\right) \cdot x_{j,t} \tag{9}$$

3. Metaheuristischer Lösungsalgorithmus

Die optimale Lösung des Problems ist für größere Probleminstanzen oft nicht mit angemessenem Rechenaufwand möglich, da das RCPSP der Problemklasse $\mathcal{NP}$ zugeordnet wird. Zur Klasse $\mathcal{NP}$ gehören Algorithmen, deren Lösungsaufwand mit der Problemgröße exponential ansteigt; meist sind dies ganzzahlige oder kombinatorische Probleme (Zimmermann 2008). Für diese Klasse von Optimierungsproblemen haben sich heuristische und metaheuristische Lösungsalgorithmen bewährt.

Die Metaheuristik Ant Colony Optimization (ACO) gehört zu den erfolgreichsten Verfahren der sogenannten Schwarmintelligenz-Verfahren. Dorigo und Di Caro (1999) entwickelten den Algorithmus in Anlehnung an die von Goss et al. (1989) im sogenannten Doppelbrückenexperiment beobachtete und untersuchte Futtersuche-Strategie einer Ameisenkolonie, bei der Ameisen aus Sicht des Operation Research ein Kürzester-Weg-Problem in einem Graphen lösen. Das Kürzester-Weg-Problem lässt sich formalisieren als Graph $G = (N, A)$ mit $|n| = N$ Knoten und einem Startknoten f und Zielknoten d sowie der Kantenlänge p_{ij} zwischen Knoten i und j. Mithilfe künstlicher Ameisen wird das Verhalten natürlicher Ameisen nachgebildet und dementsprechend versucht, einen möglichst kurzen Weg zwischen Start- und Zielknoten zu finden. In jedem Knoten fällen die Ameisen eine zufällige Entscheidung, welche der von diesen Knoten abgehenden Kanten aus der Knotennachbarschaft N_i gewählt wird, wobei die Wahrscheinlichkeit von der Pheromonmarkierung τ_{ij} der Kante zwischen Knoten i und j abhängt:

$$p_{ij}^{k} = \begin{cases} \frac{\tau_{ij}}{\sum_{j \in N_i} \tau_{ij}} & falls\ j \in N_i \\ 0 & sonst \end{cases} \quad (10)$$

Die Pheromonmarkierung der Kanten im Graph erlaubt eine indirekte Kommunikation zwischen virtuellen Ameisen, sodass im Vergleich zu anderen metaheuristischen Optimierungsalgorithmen eine schnelle Konvergenz des globalen Optimums erzielt wird. Die Pheromonverdunstung wiederum verhindert eine zu schnelle Konvergenz gegen lokale Optima.

Indem das Projektplanungsproblem RCPSP auf ein Kürzester-Weg-Problem überführt wird, kann das RCPSP mithilfe des ACO gelöst werden.

4. Ausblick auf die weitere Forschungsarbeit

Im Rahmen des Dissertationsvorhabens soll eine prototypische Software für die Planung komplexer Projekte unter Unsicherheit entwickelt werden. Dazu werden aufbauend auf den hier dargestellten Vorarbeiten zum einen die theoretischen Grundlagen weiterentwickelt sowie ein ganzheitliches RCPSP-Modell erstellt. Anschließend wird ein Lösungsalgorithmus auf dem Konzept des metaheuristischen Optimierungverfahrens Ant Colony Optimization implementiert und im Rahmen von Fallstudien, in denen reale Projektszenarien verwendet werden sollen, validiert und erprobt.

Der praktische Nutzen für Anwender, wie z.B. Projektmanager, wird darin bestehen, dass sie basierend auf den Optimierungsergebnissen bereits vor Beginn eines Projektes unterschiedliche Planungsszenarien erstellen können und ihnen somit valide Entscheidungshilfen für eine effiziente Projektplanung zur Verfügung stehen. Das Dissertationsvorhaben soll somit einen wichtigen Beitrag zur Forschung in der Arbeitswissenschaft sowie im Operations Research und Projektmanagement leisten. Es soll gleichzeitig eine praxisnahe Lösung für die im Projektmanagement tätigen Anwender bieten.

5. Literatur

Demeulemeester EL, Herroelen WS (2001) Project Scheduling – A Research Handbook. Dordrecht: Kluwer Academic Publishers.

DIN, Deutsches Institut für Normung (2009) Projektmanagement – Projektmanagementsysteme – Teil 5: Begriffe. DIN 69901-5.

Dorigo M, Di Caro G (1999) The Ant Colony Optimization Meta-Heuristic. In: Waller A (Ed) New Ideas in Optimization. London: McGraw-Hill, 9-32.

Goss S, Aron S, Deneubourg J, Pasteels J (1989) Self-organized Shortcuts in the Argentine Ant. Naturwissenschaften 76:579-581.

Hartmann S (1999) Project Scheduling under Limited Resources – Models, Methods, and Applications. Lecture Notes in Economics and Mathematical Systems No. 478. Berlin: Springer.

Reichert T (2011) Projektmanagement: die häufigsten Fehler, die wichtigsten Erfolgsfaktoren. 2. Aufl., Freiburg im Breisgau: Haufe.

Yang B, Geunes J, O'Brien WJ (2001) Resource-Constrained Project Scheduling – Past Work and New Directions. Tech. Rep. Gainesville, FL: University of Florida, Dept. Industrial and Systems Engineering.

Zimmermann H-J (2008) Operations Research – Methoden und Modelle. 2. Aufl., Wiesbaden: Vieweg & Sohn.