

Strukturierung von Fertigungssystemen mittels Genetischer Algorithmen

JENS ARNOLD

Fakultät für Informatik
Technische Universität Chemnitz
Lehrstuhl Modellierung und Simulation
09107 Chemnitz
E-Mail: jarn@informatik.tu-chemnitz.de

ANJA NESTLER

Institut für Betriebswissenschaften und
Fabriksysteme
Technische Universität Chemnitz
Lehrstuhl Fabrikplanung und -betrieb
09107 Chemnitz

1. Motivation - Das Problem der Fertigungsstrukturierung

Für eine hohe Wirtschaftlichkeit und Flexibilität von Unternehmen des Maschinenbaus ist die Optimierung der Fertigungsgestaltung elementare Voraussetzung. Ideales Ziel einer Strukturverbesserung sind „Gegenstandsspezialisierte Fertigungssysteme“, in denen die Teilearten möglichst vollständig bearbeitet werden können. Transporte zwischen Fertigungssystemen verursachen Kosten, erhöhen die Durchlaufzeiten und sind daher so gering wie möglich zu halten. Dem gegenüber stehen Steuerungs- und Logistikprobleme mit wachsender Größe und Komplexität der Fertigungssysteme. Die entscheidende Fragestellung bei der Fertigungsstrukturierung ist demnach die, zwischen welchen beiden Extrema »alle Maschinen bilden *ein* Fertigungssystem« bzw. »jede Maschine bildet ein separates Fertigungssystem« die optimale Anzahl und Größe der Fertigungssysteme für ein gegebenes Produktionsprogramm (Menge von Teilearten) liegt? In Abbildung 1 seien einige bisher angewandte und daher als „klassisch“ bezeichnete Methoden und Verfahren zur Strukturierung von Fertigungssystemen aufgezeigt und systematisiert.

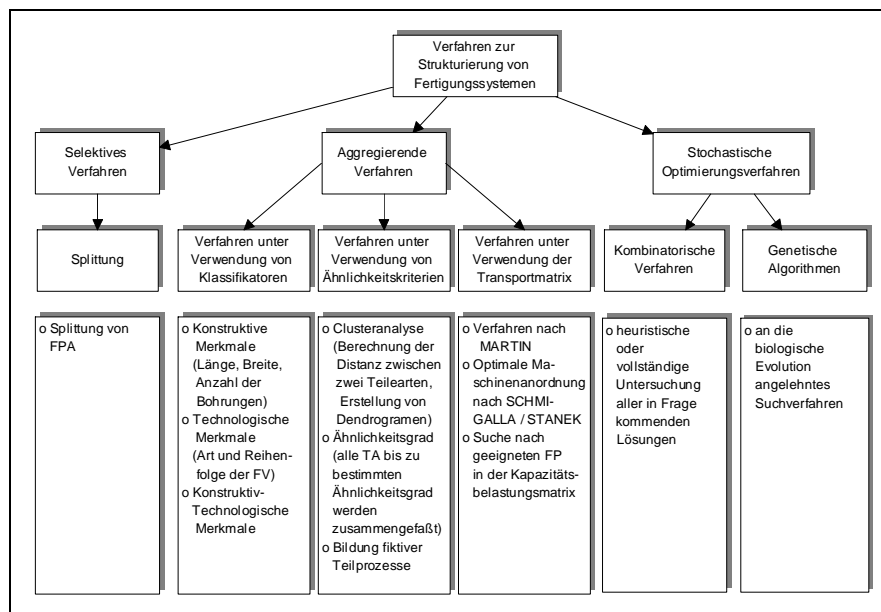


Abbildung 1: Verfahren zur Strukturierung von Fertigungssystemen

2. Ein Genetischer Algorithmus zur Fertigungsstrukturierung

Mit jedem der in Abschnitt 1 aufgeführten (klassischen) Verfahren der Analytik und Kombinatorik wird bei praxisrelevanten Dimensionen der frei veränderbaren Parameter jeweils nur ein Teilproblem (Reihenfolge, Gruppierung, Zuordnung) gelöst und der Widerspruch zwischen hoher Kooperation der Fertigungssysteme und der Maximierung ihrer Autonomie dabei stets auf eine Seite zuungunsten der anderen verschoben, ohne mathematisch begründbare Optimalitätskriterien definieren zu können. Da die Anzahl der möglichen Lösungen exponentiell zu den vorhandenen Fertigungsplätzen und Teilearten ansteigt, ist deren vollständige Durchmusterung nicht möglich. „Gewünscht sind daher Optimierungsalgorithmen, die in Abhängigkeit von einem vom Benutzer zu definierenden Ziel die Ergebnisse eines Simulationslaufes interpretieren und daraus automatisch Schlußfolgerungen für die Veränderung der Eingangsparameter des nächsten Simulationslaufes ableiten“ (SCHULTE und BECKER: „Optimierung in der Werkstattsteuerung: Simulation und genetische Algorithmen“ in [SYDO93]).

Die stochastische Optimierung mittels Genetischer Algorithmen (GA) beruht auf dem biologischen Prinzip der Evolution: »Gute« Individuen (Lösungen) überleben und geben ihre Gene (Variablen) an die nächste Generation (Lösungsmenge) durch Rekombination und Mutation weiter. Die theoretische Fundierung der Konvergenz eines solchen Iterationsprozesses wurde bereits in den siebziger Jahren gezeigt ([HOLL75]) und in zahlreichen Anwendungen als praktikabel anerkannt ([SCHA89], [FORR93], [ARNO95]). „Aus der Sicht des Mathematikers, Informatikers und Ingenieurs stellt die Evolution ein extrem leistungstarkes Optimierungsverfahren dar.“ [SCHÖ94]. Im folgenden soll ein Verfahren zur Strukturierung von Fertigungssystemen auf der Basis Genetischer Algorithmen nach [NEST96] näher beschrieben werden.

2.1. Codierung der Individuen

$$P = \{l \in S : l \text{ ist eine gültige Lösung in } S\} \quad (1)$$

bezeichnet die Teilmenge P (*Population*), der alle gültigen Lösungen l (*Individuen*) aus der Menge S (*Suchraum*) angehören. Der Suchraum S wird von den Parametervektoren (*Chromosomensatz*) des Problems aufgespannt und kann deshalb auch als Parameterraum bezeichnet werden. Die Komponenten eines Vektors werden als *Gene* bezeichnet. Ihre möglichen Wertebelagungen sind die *Allele*. Eine Fertigungsstruktur kann durch folgende drei Vektoren beschrieben werden, die somit als Chromosomen ein Individuum codieren:

- Reihenfolgevektor:

In diesem Chromosom wird die topologische Anordnung der Maschinen codiert. Es repräsentiert also das Reihenfolgeproblem. Steht der Fertigungsplatz A vor dem Fertigungsplatz B im Chromosom, so kommt der Fertigungsplatz A vor dem Fertigungsplatz B in Flußrichtung der Fertigungsstruktur. Ein Gen besteht aus der Art und der Nummer des Fertigungsplatzes, die Gruppennummer bestimmt die eventuelle Zugehörigkeit zu einem zuvor festgelegten Fertigungssystem. *First* und *Last* sind Verweise auf die diesem Fertigungsplatz zugewiesenen Fertigungsvorgänge. Die *Auslastung* ist eine reelle Zahl zwischen 0 (kein Fertigungsvorgang zugeordnet) und 1 (Fertigungsplatz ist voll ausgelastet). Die Länge dieses Vektors (Anzahl der Gene) entspricht der durch die statische Dimensionierung bestimmten minimal zur Fertigung des gegebenen Produktionsprogrammes notwendigen Anzahl Fertigungsplätze plus ein Viertel dieser Anzahl. Damit kann die Optimierung 25% mehr Fertigungsplätze hinzufügen, falls die damit verbundene Einsparung an Transportaufwand den Gesamtaufwand senkt.

- Gruppierungsvektor:

Die Gene dieses Chromosoms korrespondieren in ihrer Position mit denen des Reihenfolgevektors und können nur die diskreten Allele 0 und 1 annehmen. Eine 0 bedeutet, dieser und der im Reihenfolgevektor unmittelbar nachfolgende Fertigungsplatz gehören zu einem Fertigungssystem. Eine 1 gibt an, daß dieser Fertigungsplatz der letzte Fertigungsplatz eines Fertigungssystems (in Flußrichtung) ist und der im Reihenfolgevektor unmittelbar folgende Fertigungsplatz als erster (in Flußrichtung) in einem neuen Fertigungssystem steht.

- Fertigungsvorgangsvektor:

Die Länge dieses Vektors entspricht der Anzahl aller Fertigungsvorgänge. Jeder Fertigungsvorgang wird durch eine eindeutige Nummer (Allel) bestimmt. Fertigungsvorgänge des gleichen Typs stehen hintereinander, ebenso innerhalb dieser „Typgruppen“ die Fertigungsvorgänge, welche dem selben Fertigungsplatz zugeordnet sind. Das Gen des ersten und des letzten Fertigungsvorganges einer so zugeordneten Menge wird durch die im Reihenfolgevektor beschriebenen Verweise **First** und **Last** referenziert. Um bei der Erzeugung neuer Individuen das Einhalten bestimmter Nebenbedingungen garantieren zu können, müssen jedem Individuum noch ein Vektor für die Speicherung der maximal zulässigen und aktuellen Anzahl der Fertigungsplätze einer jeden Art sowie die für zusätzlich aufgenommene Fertigungsplätze getätigte Investitionssumme zugeordnet werden.

Der Aufwand enthält den Zielfunktionswert (*Fitneß*) für eine konkrete Lösung (*Individuum*). Wie bei jeder Optimierungsaufgabe kann die Zielfunktion

$$f : S \rightarrow R^1 \quad (2)$$

als Abbildung aus dem Suchraum S in die Menge der reellen Zahlen R^1 beschrieben werden. Generiert der evolutionäre Optimierungsprozeß ungültige Lösungen, also Individuen, die nicht zur Population P gehören, so erhalten diese Individuen einen überproportional hohen Aufwandswert (*Strafenfitneß*). Auf die Berechnung der Fitneßwerte wird im folgenden Abschnitt 3 eingegangen. Eine Teilmenge $G_i \subset P$ wird als *i-te Generation* bezeichnet. Alle in ihr enthaltenen Individuen werden gewissermaßen parallel untersucht und konkurrieren direkt.

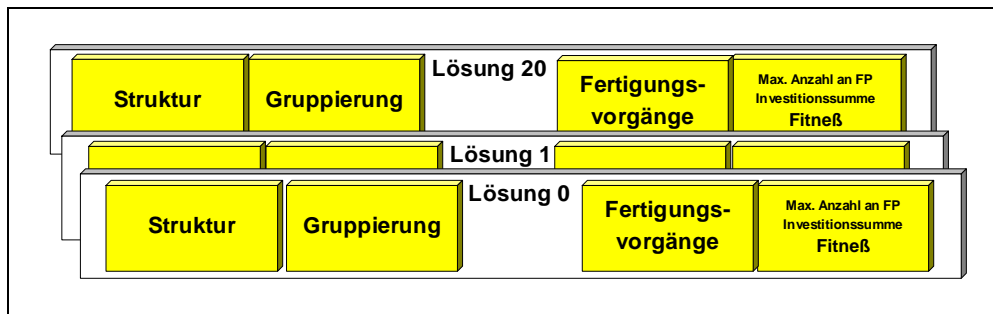


Abbildung 2: Bestandteile einer Generation

Die in der Literatur als sinnvoll angegebenen Werte für die Wahl der Generationsgröße (Anzahl der Individuen in einer Generation) haben sich in vielen Experimenten als Durchschnittswerte bewährt ($N=30$ [SCHA89], $N=50$ [SCHÖ94], $N>50$ [KINN94]), sind aber für praxisrelevante Dimensionen der Fertigungsstrukturierung aus Gründen des Speicher- und Rechenaufwandes nicht praktikabel. Mit der Beschränkung der Generationsgröße auf $N=21$ Individuen ist ein guter Kompromiß zwischen Konvergenzgeschwindigkeit der Fitneßwerte gegen das Optimum und dem Speicher- und Rechenaufwand für Personalcomputer gefunden worden.

Die 21 Individuen einer Generation setzen sich aus 6 Eltern und den von ihnen erzeugten 15 Kindern zusammen. Das Verhältnis 6/15 von Eltern zu Kindern hat sich als besonders günstig für den *Selektionsdruck* erwiesen. Ein hoher Selektionsdruck vergrößert die Konvergenzgeschwindigkeit, senkt aber die Wahrscheinlichkeit, das globale Optimum zu finden und vergrößert die Gefahr, in einem lokalen Optimum „hängenzubleiben“.

2.2. Der evolutionäre Optimierungsalgorithmus

Der iterativ ablaufende, evolutionäre Optimierungsalgorithmus ist in Abbildung 3 dargestellt. Zur Beschleunigung der Konvergenz des Evolutionsprozesses wird im Schritt 1 eine (suboptimale) Anfangslösung mit dem Aggregationsverfahren aus Abschnitt 1 berechnet und an den Algorithmus übergeben. Die Auswahl der 5 Eltern im Schritt 6 erfolgt nach dem *Roulette-Verfahren*. Da eine Minimierungsaufgabe vorliegt (Aufwand für den Transport und die Fertigungsplätze sollen minimal werden), wird der schlechteste Fitneßwert der Generation (höchster Aufwand) von den Fitneßwerten der anderen Individuen dieser Generation subtrahiert. Die „neuen“ Fitneßwerte der einzelnen Lösungen bestimmen die Sektorgrößen der Individuen auf dem Roulette-Rad. Die Wahrscheinlichkeit, daß eine zufällig geworfene „Kugel“ in einen „großen“ Sektor fällt, also ein „gutes“ Individuum überlebt, ist demzufolge höher, als daß ein „schlechtes“ Individuum seine Gene weitervererben kann. Ist die absolute Differenz der Fitneßwerte zwischen dem besten und dem schlechtesten Individuum einer Generation kleiner als 20, wird jedem Individuum, unabhängig von seiner tatsächlichen Fitneß, eine Fitneß zugeordnet, die direkt proportional zu seiner Rangordnung in der Generation ist. Dieses *Lineare Ranking-Verfahren* erhöht den Selektionsdruck und damit die Konvergenzgeschwindigkeit gegen das tatsächliche Optimum in seiner vermutlich bereits gefundenen, suboptimalen Umgebung.

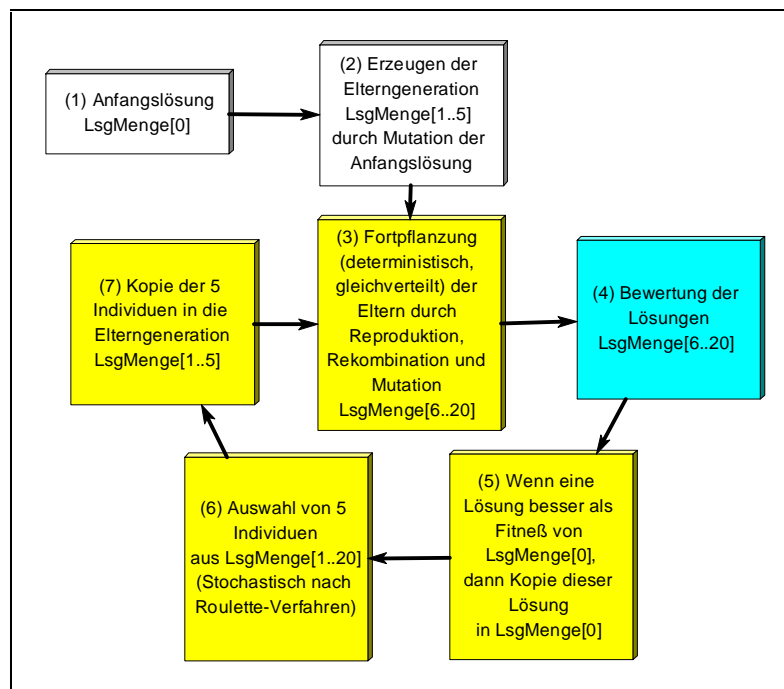


Abbildung 3: Der evolutionäre Optimierungsalgorithmus

Für die Wahl des Abbruchzeitpunktes liefert uns die Natur keinen Hinweis. Die biologische Evolution ist ein (möglicherweise unendlich) fortdauernder Prozeß ohne erkennbares *End-Ziel*. Der Grund liegt im Sinn der Evolution: Anpassung der Lebewesen an sich permanent ändernde Umweltbedingungen. Da unsere zu optimierenden künstlichen Systeme aber ein konkret definiertes, funktionales Verhalten in einer mehr oder weniger bekannten „Umwelt“ zeigen, sollte der Suchprozeß nach dem Auffinden eines/des Optimums terminieren. Deshalb kann der Nutzer jederzeit den Evolutionsprozeß unterbrechen. Ansonsten beendet das Programm nach einer vorgebbaren maximalen Anzahl Generationen seit der letzten Fitneßverbesserung automatisch die Optimierung.

2.3. Die genetischen Operatoren

Für jedes der drei Chromosomen müssen eigene genetische Operatoren entworfen werden. Die *Reproduktion* ist die ständige Erneuerung und Wiederholung der genetischen Informationen. Die Chromosomen werden einfach von den Eltern übernommen. Die Reproduktion läuft deshalb bei allen Chromosomen gleich ab. Die Reproduktion garantiert die Stabilität der Lösungen, damit der Algorithmus nicht zu sehr im Suchraum „springt“. *Rekombinationen* sind Prozesse, die zu einer Umgruppierung der genetischen Informationen führen. Durch die Rekombination werden genetische Informationen zwischen den Individuen (sexuelle Rekombination) und innerhalb der Chromosomen eines Individuums (Selbstrekombination) ausgetauscht. Die *Mutation* ist der Zufall der Evolution und für die Anpassungsfähigkeit einer Population an sich ändernde Umweltbedingungen unverzichtbar. Die Mutation ist dafür verantwortlich, daß lokale Optima (suboptimal angepaßte Individuen) wieder verlassen werden können und in andere Bereiche des Suchraumes „gesprungen“ wird.

Die Problematik der genetischen Operatoren besteht darin, daß durch ihre Ausführung oft die Konsistenz der neuen Lösung zerstört wird und ein relativ hoher Aufwand getätigt werden muß, um die Menge der ungültigen Lösungen gering zu halten. Dabei mußte ein „sinnvoller“ Kompromiß zwischen der für die Evolution unverzichtbaren Stochastik und den stark gekoppelten Nebenbedingungen gefunden werden. Insbesondere die effiziente Implementation der Rekombinationsoperatoren und die Hinzu- bzw. Wegnahme von Fertigungsplätzen durch Mutationen erforderte relativ komplexe Algorithmen.

3. Die Bewertung

Vom Genetischen Algorithmus werden weitestgehend konsistente Lösungen zur Bewertung übergeben. Trotzdem ist es möglich, daß bestimmte Restriktionen nicht eingehalten wurden. Diese Lösungen erhalten einen extrem hohen Aufwandswert (*Strafenfitneß*). Die Evolution erkennt dadurch solche ungültigen Lösungen und läßt sie im weiteren Verlauf „aussterben“.

Wurden alle Restriktionen eingehalten, werden über das gesamte Produktionsprogramm die Anzahl der Transporte in Flußrichtung, entgegen der Flußrichtung und einen Systemwechsel erfordernde Transporte gezählt. Weiterhin werden noch die Anzahl der nichtbelegten Fertigungsplätze gezählt. Diese Transporte werden nun mit den jeweiligen, vom Nutzer definierten Aufwänden multipliziert. Die Summe des Aufwandes für die Transporte und die Summe des Aufwandes der nichtbelegten Fertigungsplätze ergeben die strukturbedingten Kosten. Zu diesen Kosten werden die Abschreibungen für die Fertigungsplätze addiert. Diese Gesamtkosten ergeben nun den Aufwand der Lösung (*Fitneß*). Dieser Aufwand wird für jede Lösung berechnet und danach erneut die Optimierung aufgerufen.

4. Ergebnisse

Die Versuche haben gezeigt, daß mit den Genetischen Algorithmen bessere Strukturen erzielt werden als mit den herkömmlichen Verfahren. Nach 60.000 bewerteten Lösungen liegt die Verbesserung des Gesamtaufwandes für große Produktionsprogramme (ca. 100 Fertigungsplätze, über 100 Fertigungslose) im Mittel bei etwa 20%. Es hat sich als günstig erwiesen, die Individuen der ersten Generation (Urgeneration) mit Hilfe eines oder mehrerer „klassischer“ Verfahren zu erzeugen. Weitere Untersuchungen sollen Aufschluß über die Wirkung der Aufwandsfaktoren und deren Relationen, insbesondere auf das Strukturierungsziel bringen. Mit einer verbesserten Feineinstellung der Ausführungswahrscheinlichkeiten für die genetischen Operatoren ist eine schnellere Konvergenz des evolutionären Optimierungsprozesses zu erwarten.

Literatur

- [ARNO95] Arnold, J.:
Die Verwendung von Evolutionären Algorithmen bei der Optimierung von Fertigungssystemen. Diplomarbeit an der TU Chemnitz-Zwickau, Fakultät für Informatik, Chemnitz 1995
- [FORR93] Forrest, S. (Hrsg.):
Proceedings of the fifth international conference on Genetic Algorithms. Morgan Kaufmann Publishers, Inc., San Mateo 1993
- [HOLL75] Holland, J.H.:
Adaptation in natural and artificial systems. The University of Michigan Press, Ann Arbor 1975
- [KINN94] Kinnebrock, W.:
Optimierung mit genetischen und selektiven Algorithmen. R. Oldenbourg Verlag GmbH, München 1994
- [NEST96] Nestler, A.:
Rechnergestützte Strukturierung von Fertigungssystemen auf der Basis ähnlicher Vorgangsfolgen mittels Genetischer Algorithmen. Diplomarbeit an der TU Chemnitz-Zwickau, Institut für Betriebswissenschaften und Fabrikssysteme, Chemnitz 1996
- [SCHA89] Schaffer, J.D. (Hrsg.):
Proceedings of the 3rd International Conference of Genetic Algorithms & Applications, Arlington 1989
- [SCHÖ94] Schöneburg, E., Heinzmann, F. und Feddersen, S.:
Genetische Algorithmen und Evolutionsstrategien. Addison-Wesley, Bonn, Paris 1994
- [SYDO93] Sydow, A. (Hrsg.):
Simulationstechnik - 8. Symposium in Berlin, September 1993. Tagungsband, Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig, Wiesbaden 1993