

Universelle Modelle und ausgewählte Algorithmen des DNA-Computing

Kurzfassung der Dissertation

zur Erlangung des akademischen Grades
Doktoringenieur (Dr.-Ing.)

vorgelegt an der
Technischen Universität Dresden
Fakultät Informatik

eingereicht von
Dipl.-Inform. Thomas Hinze
geboren am 20. Juli 1971 in Halle/Saale

Betreuender Hochschullehrer: Prof. Dr.-Ing. habil. E.P. Stoschek

Dresden, den 22. Februar 2002

Danksagung

Meinem Doktorvater, Prof. Dr.-Ing. habil. E.P. Stoschek am Institut für Theoretische Informatik der Fakultät Informatik an der TU Dresden möchte ich herzlich für die mehrjährige hervorragende Zusammenarbeit danken. In einer sehr guten Arbeitsatmosphäre, geprägt von gegenseitigem Vertrauen, Achtung und Menschlichkeit hat er mir ermöglicht, daß ich mich auf meine Promotion konzentrieren konnte und mir – innerhalb der globalen Aufgabenstellung „DNA-Computing“ – bezüglich der Ausgestaltung des Themas sowie der Ausrichtung des Forschungsschwerpunktes weitgehende Freiheit gewährt.

Dem Leiter der Abteilung Chirurgische Forschung an der Medizinischen Fakultät der TU Dresden, Prof. Dr. med. H.K. Schackert, sowie den Mitarbeiterinnen und Mitarbeitern dieser Abteilung danke ich für die Erlaubnis, das molekularbiologische Labor mitbenutzen zu dürfen sowie für die Einweisung in Grundlagen entsprechender Labor- und Arbeitstechniken und die zuteil gewordene Unterstützung. Insbesondere bin ich Dipl.-Biol. M.Sc. Olga-Niki Koufaki für die Plasmidpräparation und die hilfreichen laborpraktischen Tips sehr dankbar, die eine effiziente Durchführung der experimentellen Studien erleichtert haben.

Für die vortreffliche Kooperation bei der gemeinsam realisierten und fachübergreifenden Entwicklung eines Simulationssystems für molekularbiologische Prozesse möchte ich mich herzlich bei Dipl.-Inform. Uwe Hatnik von der Arbeitsgruppe Modellierung und Simulation am Fraunhofer Institut für Integrierte Schaltungen Dresden bedanken.

Sehr dankbar bin ich Prof. Dr.-Ing. habil. R.G. Spallek und dem gesamten Team des Institutes für Technische Informatik für die unbürokratisch geleistete Unterstützung bei der Bereitstellung technischer Ressourcen und Ausrüstungen.

Mein besonderer Dank gebührt Frau Dr.-Ing. M. Sturm am Institut für Theoretische Informatik für die exzellente und konstruktive langjährige Zusammenarbeit, ihr hohes persönliches Engagement für die Etablierung und Förderung des Forschungsgebietes am Lehrstuhl sowie ihre umfassende administrative Unterstützung bei der Verwirklichung und Durchsetzung von Forschungsvorhaben. Sehr hilfreich waren die geführten wissenschaftlichen Diskussionen in einer sehr offenen, motivationsfördernden, von gegenseitiger Wertschätzung geprägten und auf die Gewinnung innovativer Ideen gerichteten Atmosphäre.

Bei der Studienstiftung des Deutschen Volkes möchte ich mich für die Aufnahme in die Förderung sowie die Vergabe meines Doktorandenstipendiums ebenso bedanken wie bei meinem Vertrauensdozenten, Prof. Dr. rer. nat. habil. C. Laubschat, für die gute Betreuung.

Meinen Eltern bin ich in tiefer Dankbarkeit und Hochachtung dafür verbunden, daß ich eine sorgenfreie Kindheit und Jugend erleben konnte. Vor allem möchte ich meiner Mutter großen Dank dafür aussprechen, daß sie mir, stets aufopferungsvoll um mein Wohl bemüht, nach dem Tod meines Vaters ein Hochschulstudium ermöglicht und mich in jeder Hinsicht nach Kräften unterstützt hat. Nicht zuletzt danke ich all meinen Freunden für den Rückhalt, die gegenseitige Inspiration sowie für den kreativitätsfördernden und ideenfreisetzenden Ausgleich bei Sport und Kulturerlebnissen. Allen akribischen Leserinnen und Lesern des Manuskripts für diese Arbeit danke ich für die aufgewendete Mühe und die eingebrachten Anregungen.

1 Einführung

Motivation

Elektronische Rechentechnik hat in den zurückliegenden Jahrzehnten Technik, Wirtschaft, Wissenschaft und Alltag nachhaltig verändert. Sie löste eine Welle der Automatisierung und Beschleunigung vieler Arbeitsabläufe aus. Computertechnik, die nach dem Prinzip des von-Neumann-Rechners arbeitet, flankiert nahezu alle Bereiche des Lebens. In einer Reihe von anspruchsvollen Anwendungsgebieten dominiert der Einsatz von Supercomputern. Integrationsgrad, Rechengeschwindigkeit und Speicherkapazität elektronischer Computer konnten im Ergebnis intensiver Forschungen kontinuierlich gesteigert werden. Einhergehend mit der Erhöhung der hardwareseitigen Leistungsfähigkeit gelang die Entwicklung effizienter Algorithmen zur Lösung zahlreicher Aufgabenstellungen und die Erschließung eines breiten Spektrums von Anwendungsfeldern. Trotz der erreichten Fortschritte ist bereits heute absehbar, daß Leistungsparameter elektronischer Rechentechnik wie beispielsweise Taktfrequenz und Speicherdichte nicht beliebig erhöht werden können. Bei der Bearbeitung extrem rechenintensiver Aufgaben und beim Einsatz unter außergewöhnlichen Umgebungsbedingungen stößt konventionelle Rechentechnik zunehmend an ihre physikalisch wie auch technologisch bedingten Grenzen. Insbesondere kombinatorische Suchprobleme (NP-Probleme) erweisen sich als bedeutende mathematische Problemstellungen hoher Praxisrelevanz, deren uneingeschränkt exakte Lösung konventionelle Rechentechnik infolge des exponentiellen Ressourcenverbrauchs vor neue Herausforderungen stellt.

In diesem Zusammenhang untersucht die Forschung weltweit alternative Computingkonzepte, auf deren Basis die Konstruktion entsprechender Hard- und Software mit favorisierten Eigenschaften aufsetzt. Unter Computingkonzepten werden nutzbare Prinzipien aus der Natur mit beherrschbarer, reproduzierbarer und analysierbarer Wirkung verstanden, die gezielt zur steuerbaren Ausführung von Rechengängen dienen können. Als unkonventionelle Computingkonzepte stehen derzeit die Ausprägungen *Quantum Computing*, *Neural Computing*, *Evolutionary Computing* und *Molecular Computing* im Fokus des wissenschaftlichen Interesses, die zusammenfassend mit dem Oberbegriff *Future Computing* bezeichnet werden ([CaCD_98]).

Forciert durch rasante Fortschritte in der Molekularbiologie, der Biochemie sowie der Gentechnik gilt aus heutiger Sicht das Molecular Computing als dasjenige Computingkonzept, dessen Potential mittelfristig als am aussichtsreichsten im Hinblick auf eine tatsächliche praktische Anwendung und vorteilhafte Nutzbarkeit erscheint. Molecular Computing verfolgt die Idee, organische Moleküle als Datenträger (Speichermedium) einzusetzen, mittels chemischer Reaktionen und physikalischer Prozesse auf diesen Molekülen die Ausführung von Operationen nachzubilden und dadurch Rechengänge zu realisieren. Dieser Ansatz begründet eine neuartige Rechnerarchitektur, bei der die Eingabedaten in Form geeigneter Moleküle bereitgestellt werden und die Ergebnisausgabe durch Analyse der aus der Bearbeitung hervorgegangenen Moleküle erfolgt. Innerhalb des Molecular Computing unterscheidet man derzeit je nach gewähltem Datenträger zwischen *DNA*-, *RNA*- und *Protein*-Computing. Prominentester Vertreter ist das DNA-Computing, bei dem Desoxyribonucleinsäure (DNA) als Datenträger sowie moderne Rekombinationstechniken aus der Molekularbiologie im Einklang mit durchdachten, für den speziellen Einsatzzweck konstruierten Versuchsaufbauten zur Anwendung gelangen. Das Potential des DNA-Computing liegt sowohl in der *massiven Datenparallelität* – denn ei-

ne Operation wirkt gleichzeitig auf eine Vielzahl von DNA-Strängen – als auch in der *hohen Speicherkapazität*. Hochleistungsrechner auf Basis des DNA-Computing mit einer Rechengeschwindigkeit von bis zu $1,2 \cdot 10^{18}$ molekularen Operationen pro Sekunde, einer Speicherkapazität von bis zu 10^{21} bit und einer Speicherdichte von $1 \frac{\text{bit}}{\text{nm}^3}$ gelten als praktisch realisierbar ([Kari_97]). Diese Leistungsparameter übertreffen Vergleichswerte elektronischer Rechner um mehrere Zehnerpotenzen. Darüber hinaus zeichnen sich DNA-Computer durch einen sehr niedrigen Energieverbrauch (10^{19} molekulare Operationen pro Joule), eine Arbeitsweise frei von mechanischem Verschleiß und durch eine weitgehende Recyclingfähigkeit aus ([RWBC_99]). In Konsequenz verkörpert das DNA-Computing einen vielversprechenden Ansatz, sich den Herausforderungen bei der zeiteffizienten Lösung von NP-Problemen zu stellen und darüber hinaus zusätzliche rechen- und speicherintensive Anwendungsgebiete zu erschließen, die gegenwärtig noch als realitätsfern gelten. Als Beispiel seien biokompatible Minicomputer genannt, die Spezialaufgaben im menschlichen Körper übernehmen können.

Die ersten praktischen Arbeiten zum DNA-Computing gehen auf das Jahr 1994 zurück, als Leonard M. Adleman (einer der Begründer des Kryptostandards RSA) erfolgreich im Labor eine Instanz des Hamiltonkreis-Problems (Größe $n = 7$) löste, bekannt unter dem Namen *Adleman-Experiment* ([Adle_94]). Dieser erste Erfolg gab den entscheidenden Anstoß für zahlreiche Forschergruppen, die Idee des DNA-Computing aufzugreifen und theoretisch wie auch experimentell weiterzuentwickeln. Die weltweit zusammengesetzte Forschergemeinde zählt heute etwa 400 Mitglieder. Begleitet werden die Forschungsaktivitäten von speziellen internationalen Workshops (DNA Based Computers) sowie der Integration des Forschungsgebietes in eine Disziplin der Theoretischen Informatik, die sich mit unkonventionellen Modellen der Berechenbarkeit befaßt.

In der Folge des Adleman-Experimentes entwickelten sich Arbeiten auf dem Gebiet des DNA-Computing weitgehend getrennt voneinander in zwei verschiedene Richtungen. Einerseits entstanden mehrere Modelle des DNA-Computing, die sowohl den Datenträger DNA als auch das Repertoire molekularbiologischer Operationen stark abstrahiert und idealisiert betrachten und das DNA-Computing unter dem Aspekt der Berechenbarkeitstheorie untersuchen. Modelle des DNA-Computing werden hierbei als spezielle Modelle der Berechenbarkeit aufgefaßt und hinsichtlich ihrer Berechnungsstärke klassifiziert. Andererseits beschäftigen sich Laborstudien mit der Aufgabe, ausgewählte Instanzen zum meist NP-vollständiger Probleme experimentell zu lösen. Von Anfang an zeigte sich die Seiteneffektanfälligkeit von molekularbiologischen Operationen als Hemmnis für große Anwendungen und damit als Herausforderung für die Schaffung entsprechend optimierter Operationsfolgen und für die Auswahl einsatzfähiger DNA-Sequenzen. Seiteneffekte sind unerwünschte Wirkungen der Operationen, die sporadisch auftreten und sich nicht gezielt reproduzieren lassen. Gegenwärtige Experimente zur laborpraktischen Lösung von NP-Problemen erreichen Problemgrößen von $n \approx 20$ ([BJRH_00]). Seit etwa 1997 trat die Idee eines *universellen DNA-Computers* in den Vordergrund, der über einen Satz exakt definierter Basisoperationen verfügt, die lediglich in ihrer Abfolge variiert werden können ([PaRS_98]). Ein universeller DNA-Computer soll hierbei die Berechnungsstärke der Turingmaschine erreichen.

Das globale Forschungsziel des DNA-Computing besteht im Aufbau eines universellen und darüber hinaus stark miniaturisierten DNA-Computers, wobei zum gegenwärtigen Zeitpunkt noch keine klare Aussage getroffen werden kann, wann dieser Computer Anwendern zur Verfügung stehen wird. Eine ähnliche Situation konnte man auch im Vorfeld der elektronischen Rechentechnik beobachten: Während die Turingmaschine als forma-

les Modell der Berechenbarkeit bereits in den 30er Jahren in der Theorie anerkannt war, stammt der erste darauf basierende von-Neumann-Rechner aus dem Jahr 1952.

Zielstellung und thematische Abgrenzung

Die Arbeit hat zum Ziel, die Eingliederung des DNA-Computing in die Theoretische Informatik zu festigen sowie einen Beitrag zum besseren Verständnis von Rechenvorgängen, ihrer mathematischen Beschreibung und von daraus ableitbaren Modellen der Berechenbarkeit zu leisten. Mit dieser Zielstellung ist verbunden, den Begriff der Turing-Berechenbarkeit fundiert auf das DNA-Computing und seine zugrundeliegenden Prinzipien zu adaptieren.

Das erste Teilziel besteht in einer Analyse molekularbiologischer Prozesse auf DNA, die mittels geeigneter Labortechniken *in vitro* ausführbar sind, im Hinblick auf eine abstrakt formale, aber dennoch labornahe Beschreibung dieser Prozesse. Sie werden jeweils als Operationen auf dem Datenträger DNA angesehen und bilden die Grundlage eines zu schaffenden labornahen Modells des DNA-Computing. Bisher bekannte Modelle des DNA-Computing weisen ein im Vergleich sehr hohes Abstraktionsniveau der Modelloperationen und der Darstellung DNA-basierter Daten auf, wodurch diese Modelle der Berechenbarkeit als stark idealisiert gelten. Zur Erzielung ihrer Universalität werden Voraussetzungen herangezogen, die einer praktischen Implementierbarkeit grundsätzlich entgegenstehen. Beispiele hierfür sind die Annahme von unendlich vielen DNA-Strangexemplaren, die zu jedem Zeitpunkt während der Ausführung von Berechnungen zur Verfügung stehen, die Forderung nach uneingeschränkter Duplizierbarkeit beliebiger DNA-Pools ohne Kenntnis der Anfangs- und Endstücke der enthaltenen DNA-Stränge sowie die Hinzunahme von Modelloperationen ohne Laborbezug.

Eine vergleichende Untersuchung bisher bekannter Modelle des DNA-Computing kennzeichnet das zweite Teilziel. Im Ergebnis dieser Betrachtungen kristallisieren sich genau diejenigen DNA-Computing-Modelle heraus, die universell oder zumindest platzbeschränkt universell sind und von linearer DNA als Datenträger ausgehen. Diese Modelle sollen hinsichtlich wesentlicher Modelleigenschaften (wie z.B. Restriktivität, Multimengenbasiertheit) klassifiziert und durchgängig formal beschrieben werden. Dies beinhaltet, teilweise vorliegende informale Modellbeschreibungen semantikäquivalent zu formalisieren und entsprechend zu ersetzen. Für alle untersuchten Modelle des DNA-Computing wird einheitlich jeweils die Konstruktion eines Algorithmus zur Lösung des Rucksackproblems angestrebt. Dieses NP-vollständige Problem erfordert eine Verarbeitung natürlicher Zahlen und unterscheidet sich diesbezüglich von bisher im DNA-Computing zumeist favorisierten selektionsorientierten algorithmischen Beispielen.

Ausgehend von einem geeigneten Repertoire molekularbiologischer Prozesse und der Beschreibung linearer DNA-Einzel- und -Doppelstränge auf der Ebene von Nucleotiden und Strangendenmarkierungen fokussiert die Arbeit die Erstellung eines labornahen abstrakten Modells des DNA-Computing, das formal beschreibbar ist, Seiteneffekte der zugrundeliegenden molekularbiologischen Prozesse geeignet in die Modellbeschreibung einbezieht und dessen uneingeschränkte Universalität durch Anwendung von Methoden der Berechenbarkeitstheorie nachgewiesen wird. Ein solches DNA-Computing-Modell stellt ein Modell der Berechenbarkeit dar, das die Berechnungsstärke der Turingmaschine erreicht und darüber hinaus als Basis für den Aufbau universeller DNA-Computer dienen kann. Ein historisches Analogon ist unmittelbar durch die Turingmaschine gegeben, auf deren Grundlage universelle elektronische Computer wie der PC beruhen.

Die Anwendungsorientierung des zu erstellenden DNA-Computing-Modells soll ebenfalls durch Konstruktion eines Algorithmus zur Lösung des Rucksackproblems auf natürlichen Zahlen untersetzt sowie anhand einer Probleminstanz und eines weiteren signifikanten Beispiels laborpraktisch verifiziert werden.

DNA-Computing vereint innovativ die Molekularbiologie mit der Informatik. Das junge und interdisziplinär geprägte Arbeitsgebiet besitzt international den Status einer Grundlagenforschung. Es berührt in seiner Gesamtheit jedoch viele wissenschaftliche Aspekte der Informatik und integriert sich schrittweise in historisch gewachsene Strukturen der umgebenden Fachgebiete. Die vorliegende Arbeit konzentriert sich auf eine Betrachtung aus Sicht der Berechenbarkeitstheorie sowie Algorithmenkonstruktion und bezieht sich stets auf entsprechende abstrakte Modelle der Berechenbarkeit. Etablierte Labortechniken der Molekularbiologie werden – vergleichbar mit einem Werkzeugkasten – als gegeben vorausgesetzt. Ihre laborpraktisch-experimentelle Modifikation und Optimierung (beispielsweise zur Eindämmung und/oder Kompensation von Seiteneffekten) wird nicht als Gegenstand dieser Arbeit angesehen. Ebenso wird der Aspekt des *in vivo* DNA-Computing nicht weiterführend berücksichtigt, bei dessen praktischer Anwendung molekularbiologische Prozesse vom Reagenzglas in lebende Zellen verlagert werden.

Gliederung der Arbeit

Die vorliegende Arbeit resultiert aus einem interdisziplinären Ansatz und gliedert sich in sechs thematische Kapitel, die fortlaufend aufeinander aufbauen. Das Kapitel 2 behandelt die mathematischen Grundlagen des DNA-Computing, definiert ausgewählte konventionelle universelle Modelle der Berechenbarkeit und zeigt konstruktiv, wie eine gegenseitige Simulation dieser Modelle möglich ist. Ausführungen zur Turing-Berechenbarkeit und Grundbegriffe zur Komplexitätstheorie beschließen das Kapitel. Gegenstand des Kapitels 3 sind die molekularbiologischen Grundlagen des DNA-Computing. Aufbauend auf der Vorstellung von DNA-Einzel- und -Doppelstrang sowie ihrer Eigenschaften bezüglich einer Anwendung als Datenträger erfolgt eine Beschreibung und Analyse der molekularbiologischen Prozesse, die als Operationen auf DNA dienen: Oligonucleotidsynthese, Hybridisierung, Denaturierung, Mischen und Aliquotieren, Ligation, Restriktionspaltung, Strangendenmodifikation, Polymerisation, Polymerase-Kettenreaktion (PCR), Avidin-Biotin-Separation, Gel-Elektrophorese und Sequenzierung. Im Kapitel 4 werden universelle und platzbeschränkt universelle, auf linearer DNA basierende Modelle des DNA-Computing klassifiziert und formal beschrieben. Die Universalitätsnachweise werden konstruktiv durch Angabe von Transformationen konventioneller universeller Modelle der Berechenbarkeit geführt und einheitlich Algorithmen zur Lösung des Rucksackproblems konstruiert. Die Kapitel 5 und 6 führen neue Modelle des DNA-Computing ein, die den gesetzten Zielen der Modellentwicklung genügen. Mit dem endlichkomponentigen verteilten Splicing-System TT6 (Test Tube 6) wird ein anwendungsorientiertes universelles Modell des DNA-Computing beschrieben, das gegenüber Vergleichssystemen mehrere Vorteile bietet, wie beispielsweise ein separates Ergebnisreagenzglas und einen mittels PCR realisierbaren Filtermechanismus. Das Modell Sisyphus verkörpert ein labornahes, restriktives und multimengenbasiertes DNA-Computing-Modell in Form eines Simulationssystems, das auch Seiteneffekte der zugrundeliegenden molekularbiologischen Prozesse durch geeignete statistische Parameter in die Modellbeschreibung einbezieht. Die Universalität der Modelle TT6 und Sisyphus wird konstruktiv nachgewiesen, und es werden Beispielalgorithmen zur Lösung des Rucksackproblems entwickelt und verifiziert.

2 Mathematische Grundlagen des DNA-Computing

In diesem Kapitel werden die berechenbarkeits- und komplexitätstheoretischen Grundlagen beschrieben, auf denen formale Modelle des DNA-Computing aufsetzen. Damit einhergehend werden erforderliche mathematische Beschreibungsmittel, Zusammenhänge und Methoden eingeführt sowie eine entsprechende Symbolnotation festgelegt.

Basierend auf den dargestellten mathematischen Grundbegriffen erfolgt die Vorstellung ausgewählter konventioneller Berechenbarkeitsmodelle, die als universell und mithin gleichmächtig im Sinne des Turing-Berechenbarkeitsbegriffs gelten: deterministische und nichtdeterministische Turingmaschine, die Klasse der μ -rekursiven Funktionen, die Klasse der WHILE-Programme sowie Chomsky-Grammatiken vom Typ 0 zur Beschreibung rekursiv aufzählbarer formaler Sprachen. Das Repertoire bekannter Berechenbarkeitsmodelle ist damit bei weitem nicht erschöpft. Es wurden diejenigen Modelle aufgenommen, die mittels weiterführend betrachteter Modelle des DNA-Computing simuliert werden können und deshalb für den Universalitätsnachweis von DNA-Computing-Modellen von Bedeutung sind. DNA-Computing-Modelle stellen spezielle Modelle der Berechenbarkeit dar. Es wird ferner gezeigt, wie sich ausgewählte konventionelle Berechenbarkeitsmodelle gegenseitig simulieren können. Die hierzu verwendeten Modelltransformationen beinhalten konstruktive Methoden, um auch Algorithmen, die in den Beschreibungsmitteln der jeweiligen Modelle notiert sind, geeignet zwischen Berechenbarkeitsmodellen zu überführen. Dies gestattet es, Algorithmen vorteilhaft mit genau den Beschreibungsmitteln zu notieren, die für die jeweilige Problemklasse am effizientesten handhabbar sind und sie anschließend in das gewünschte Berechenbarkeitsmodell abzubilden. Die Definition der Turing-Berechenbarkeit sowie Ausführungen zum intuitiven Berechenbarkeitsbegriff und zur Churchschen These sind im Zusammenhang mit der Universalität von Modellen der Berechenbarkeit von Interesse und werden weiterführend bei der Untersuchung von Modellen des DNA-Computing herangezogen. Ausgewählte Grundbegriffe der Komplexitätstheorie, die im Zusammenhang mit der Betrachtung von NP-Problemen und Effizienzbewertungen entsprechender Lösungsalgorithmen stehen, beschließen dieses Kapitel.

Die dargelegten mathematischen Grundlagen des DNA-Computing basieren auf [Börg_92], [BrSM_00], [CrKa_99], [EMCG_98], [HoUl_00], [KBL_94], [Loec_76], [PaRS_98], [RoSa_97], [Schö_97] und [Yage_86].

3 Molekularbiologische Grundlagen des DNA-Computing

Das experimentelle DNA-Computing ist eng mit den Erkenntnissen, Entwicklungen und Entdeckungen der Biochemie, der Molekularbiologie und der Gentechnik verbunden.

Die Desoxyribonucleinsäure (DNA) wird im Hinblick auf ihre Funktion als Datenträger vorgestellt. Ihr Aufbau, ihre chemische Struktur und daraus resultierende, für das DNA-Computing wesentliche Eigenschaften werden erläutert. Molekularbiologische Prozesse und Reaktionen werden beim DNA-Computing als Operationen auf DNA genutzt. Viele dieser Abläufe vollziehen sich auch auf natürliche Weise in der lebenden Zelle (in vivo) und können im Reagenzglas (in vitro) nachgebildet werden. Es wird darauf eingegangen, wie beim DNA-Computing die Dateneingabe, ihre Verarbeitung durch ein Repertoire an Operationen auf DNA und die Ergebnisausgabe laborpraktisch realisierbar sind. Dies

bedeutet auf der molekularbiologischen Ebene, DNA zu gewinnen, gezielt miteinander zu rekombinieren und spezifische Merkmale von DNA zu bestimmen und auszuwerten. Das Kapitel gibt einen Überblick über die Wirkung, die Ausführung und signifikante Seiteneffekte von molekularbiologischen Prozessen und daraus resultierenden Labortechniken. Im einzelnen werden betrachtet:

- Oligonucleotidsynthese (Synthesis)
- Hybridisierung (Annealing)
- Denaturierung (Melting)
- Mischen (Union)
- Aufteilen (Split)
- Ligation
- Restriktionsspaltung (Digestion)
- Strangendenmodifikation (Labeling)
- Polymerisation
- Polymerase-Kettenreaktion (PCR)
- Avidin-Biotin-Separation (Affinity Purification)
- Gel-Elektrophorese (Gel Electrophoresis)
- Sequenzierung (Sequencing)

Dieses Kapitel basiert im wesentlichen auf [AlBu_94], [Alph_98], [BeSi_93], [Bugg_97], [DrPo_92], [HaBa_90], [JoDa_95], [KaDK_94], [Knip_97], [LoZo_98], [Mart_96], [NeGr_97], [Tren_94] und [Wats_93].

4 Ausgewählte Modelle und formale Sprachen des DNA-Computing

Universelle sowie platzbeschränkt universelle Modelle und formale Sprachen des DNA-Computing werden untersucht, deren DNA-basierte Daten auf einer Abbildung linearer DNA beruhen. Eigene Arbeiten zeigen die formale Beschreibung von DNA-Computing-Modellen, die imperative Programmiersprachen begründen, den Nachweis ihrer Universalität sowie einheitlich Algorithmen zur Lösung des Rucksackproblems auf natürlichen Zahlen für alle betrachteten Modelle des DNA-Computing.

Jedes DNA-Computing-Modell verkörpert eine abstrakte Beschreibung von Berechnungsprozessen, bei denen durch geeignete mathematische Beschreibung kodierte DNA-basierte Daten mittels einer Abfolge darauf einwirkender Operationen gezielt verändert werden. Operationen sind hierbei idealisierte (abstrahierte) Abbilder molekularbiologischer Prozesse auf DNA.

Wichtige Referenzen für Modelle des DNA-Computing sind: [Ade_96], [Amos_97], [DaEr_98], [Head_87], [KaTh_96], [Lipt_95], [MiSa_97], [MVPS_98], [PaRS_96], [PaRS_98], [Paun_96b], [Reif_95], [Reif_99], [RoWa_95], [RoWa_96], [Sosi_01], [YoKo_99].

Klassifikation

Modelle des DNA-Computing werden mit dem Ziel betrachtet, dieses Computingkonzept hinsichtlich seiner Berechnungsstärke, Praktikabilität und der Effizienz von Algorithmen zur Lösung ausgewählter Probleme einzuschätzen. In der vergleichsweise kurzen Geschichte des DNA-Computing wurde eine Vielzahl entsprechender Modelle und formaler Sprachen eingeführt, die jeweils spezifische Aspekte des Rechnens auf DNA hervorheben

oder ignorieren. Faßt man diese Aspekte zusammen, so erhält man Klassifikationsmerkmale für Modelle des DNA-Computing:

- Berechnungsstärke (Klasse der durch das Modell generierbaren formalen Sprachen)
- zugrundeliegendes Paradigma (imperativ, funktional, logisch) der Modellsprache
- Ablaufsteuerung bei Abarbeitung von Algorithmen (deterministisch, nichtdeterministisch)
- Einordnung des Modells in das Klassifikationsschema für Rechnerarchitekturen nach Flynn (SISD, SIMD, MISD, MIMD)
- Art der Beschreibung der im Modell definierten Operationen (formal, informal)
- Vorliegen einer laborpraktischen Implementierung der im Modell definierten Operationen und Kontrollstruktur sowie von Beispielalgorithmen
- Verfügbarkeit der Eingangsoperanden nach Operationsausführung als Annahme im Modell
- geforderte DNA-Struktur zur Kodierung der im Modell verarbeiteten Daten, z.B.
 - einzel- bzw. doppelsträngige lineare DNA, kodiert durch Wörter
 - Berücksichtigung von Strangendenmarkierungen im Modell
 - Hairpins, im Modell erzeugt durch gezielt mit sich selbst annealende DNA-Einzelstränge
 - nichtlineare räumliche DNA-Strukturen, die im Modell aus gezielt miteinander annealenden DNA-Einzelsträngen aufgebaut sind
 - Verbindungen von DNA mit anderen modelldatenkodierenden Molekülen
- Berücksichtigung der Exemplaranzahl identischer DNA-Moleküle in der Beschreibung der Modelloperationen (Multimengenbasiertheit)
- Abstraktionsniveau der Operationen hinsichtlich laborpraktischer Implementierbarkeit
- Ressourcenbedarf bei laborpraktischer Implementierung von Modellalgorithmen, z.B.
 - Anzahl benötigter Reagenzgläser
 - Anzahl, Art und Verfügbarkeit benötigter Enzyme
 - Anforderungen an die als Eingangsoperanden benötigte DNA (Sequenzen, Stranglängen, Mindestmengen)

Allen Modellen des DNA-Computing ist gemeinsam, daß verarbeitete Daten durch abstrahierte Abbildungen von DNA modelliert sind (DNA-basierte Daten), die durch einen spezifischen Satz definiert darauf einwirkender Operationen modifiziert werden. Der Operationssatz umfaßt Methoden zur Generierung und selektiven Modifikation DNA-basierter Daten. Selektive Modifikation bedeutet, daß DNA-basierte Daten mit bestimmten definierten Merkmalen gezielt ausgewählt und nur diese zu modifizierten DNA-basierten Daten umgeformt (rekombiniert) werden. Generierung und selektive Modifikation können im Verbund bei der Abarbeitung einer Einzeloperation auftreten. Jedes DNA-Computing-Modell beinhaltet ferner eine Kontrollstruktur, die es ermöglicht, Abfolgen von Operationen zu bestimmen und auszuführen. Alle bisher bekannten Modelle des DNA-Computing abstrahieren von möglichen Seiteneffekten molekularbiologischer Prozesse, die bei einer laborpraktischen Implementierung als Bestandteile der Operationenwirkungen auftreten können.

Forschungsarbeiten zum DNA-Computing werden motiviert von der Vision, ein möglichst labornahes Modell eines uneingeschränkt in vitro implementierbaren DNA-basierten Universalcomputers zu schaffen, der zuverlässig für eine effiziente Lösung rechen- und speicherintensiver algorithmischer Probleme wirtschaftlicher und wissenschaftlicher Relevanz vorteilhaft eingesetzt werden kann. Bisher bekannte DNA-Computing-Modelle sind von

diesem Ideal entfernt, vor allem begründet durch das hohe Abstraktionsniveau der verwendeten Modelloperationen, das einer unmittelbaren Laborimplementierung entgegensteht.

In diesem Kapitel werden DNA-Computing-Modelle beschrieben und untersucht, die universell oder platzbeschränkt universell sind, die gemäß dem natürlichen Vorbild auf der ausschließlichen Nutzung linearer DNA, modelliert durch geeignete Wortkodierungen, basieren und die als wissenschaftlich etabliert gelten. Es werden betrachtet:

- die Filtering-Modelle nach Adleman, Lipton und Amos in einer überarbeiteten Fassung
- das Modell Parallel Associative Memory (PAM)
- DNA-Pascal
- DNA-Equality Checking (DNA-EC)
- Insertion-Deletion-Systeme
- Watson-Crick D0L-Systeme
- Splicing-Systeme (H-Systeme, EH-Systeme)

Die genannten DNA-Computing-Modelle werden eingeführt, begleitet von einem kurzen geschichtlichen Abriß der Modellentwicklung. Der jeweilige Universalitätsnachweis wird konstruktiv geführt durch Angabe einer Transformation eines universellen konventionellen Berechenbarkeitsmodells aus Kapitel 2 in das entsprechende Modell des DNA-Computing. Als Beispielalgorithmus wird einheitlich für alle betrachteten Modelle ein Algorithmus zur Lösung des Rucksackproblems auf natürlichen Zahlen konstruiert.

Eigenschaften von Modellen des DNA-Computing

Restriktivität: Ein Modell des DNA-Computing ist *restriktiv*, wenn die Eingangsoperanden jeder Operation nach deren Ausführung nicht mehr zur Verfügung stehen.

Restriktive DNA-Computing-Modelle sind der Vorgehensweise bei laborpraktischer Ausführung von Algorithmen *in vitro* nachempfunden: DNA modifizierende molekularbiologische Prozesse laufen in einem Reagenzglas ab, dessen ursprünglicher Inhalt nach Prozeßende im allgemeinen verändert ist. Restriktivität trägt auch der Tatsache Rechnung, daß beliebige Reagenzglasinhalte, bestehend aus Mengen von DNA-Strängen unterschiedlicher unbekannter Sequenzen, laborpraktisch nicht uneingeschränkt beliebig oft dupliziert werden können. Nichtrestriktive DNA-Computing-Modelle vernachlässigen diesen Aspekt.

Multimengenbasiertheit: Ein Modell des DNA-Computing ist *multimengenbasiert*, wenn im Modell abgebildete und verarbeitete Daten durch Multimengen beschrieben werden und die Operationen auf diesen Daten als Multimengenoperationen definiert sind.

Multimengenbasierte DNA-Computing-Modelle berücksichtigen die Exemplaranzahl identischer DNA-Stränge bei der Spezifikation der Operationswirkungen. Sie berücksichtigen somit, daß sich die beobachtbare Wirkung molekularbiologischer Prozesse mit zunehmendem Ungleichgewicht der Exemplaranzahlen eingehender DNA-Stränge signifikant verändern kann.

Vorliegen von Determinismus: Ein Modell des DNA-Computing ist *deterministisch*, wenn sich bei Ausführung jedes in diesem Modell beschreibbaren Algorithmus jede Operation einer Operationsfolge einschließlich ihrer Eingangsoperanden nur aus den

zuvor abgearbeiteten Operationen dieses Algorithmus und seinen Eingangsdaten ergibt.

Diese Eigenschaft charakterisiert die Ablaufsteuerung bei Modellen des DNA-Computing. Bei nichtdeterministischen Modellen kann es aufgrund der innewohnenden Zufallskomponente vorkommen, daß keine uneingeschränkte Wiederholbarkeit algorithmischer Abläufe möglich ist.

Zugrundeliegendes Paradigma: Eine durch ein Modell des DNA-Computing definierte Programmiersprache unterliegt dem *imperativen Paradigma*, wenn die Kontrollstruktur der in diesem Modell beschreibbaren Algorithmen durch eine Abfolge von Steueranweisungen und Wertzuweisungen vorgegeben ist.

Eine durch ein Modell des DNA-Computing definierte Programmiersprache unterliegt dem *funktionalen Paradigma*, wenn jeder in diesem Modell beschreibbare Algorithmus ausschließlich durch eine Menge von Funktionsdefinitionen gegeben ist und die Berechnung von Funktionswerten die Prinzipien Funktionsabstraktion und Funktionsapplikation nutzt.

Eine durch ein Modell des DNA-Computing definierte Programmiersprache unterliegt dem *logischen Paradigma*, wenn ein Algorithmus in Form einer Problemspezifikation durch Axiome beschrieben wird und die Abarbeitung des Algorithmus der Berechnung einer logischen Folgerung aus diesen Axiomen gemäß einer gegebenen Anfrage entspricht.

Multiple-Instruction-Fähigkeit: Ein Modell des DNA-Computing ist *Multiple-Instruction-fähig*, wenn damit beschreibbare Algorithmen zeitgleich unterschiedliche Operationen im Sinne einer parallelen oder verteilten Berechnung ausführen können.

Multiple-Data-Fähigkeit: Ein Modell des DNA-Computing ist *Multiple-Data-fähig*, wenn darin eine Operation definiert ist, die zeitgleich auf unterschiedliche DNA-basierte Daten zugreifen und diese im Sinne einer datenparallelen Berechnung verarbeiten kann.

5 Splicing-Systeme als Modelle für DNA-basierte Universalcomputer

Es wird ein anwendungsorientiertes verteiltes universelles Splicing-System (TT6-EH-System, kurz TT6) eingeführt, das als Resultat eigener Arbeiten ein optimiertes Splicing-System im Hinblick auf eine praktische Realisierung DNA-basierter Universalcomputer darstellt. Beschreibungen dieses Systems sind in [HiSt_00b] und [StHi_01] veröffentlicht. Ein wesentliches Ziel der Entwicklung von Modellen des DNA-Computing besteht darin, einen DNA-basierten Universalcomputer möglichst labornah zu beschreiben. Ein entsprechend optimiertes Modell des DNA-Computing dient der Vorbereitung einer laborpraktischen Umsetzung und verkörpert einen wichtigen Meilenstein auf dem Weg zum Aufbau biologischer Rechner auf Basis geeignet konstruierter Biohardware. Unter dem Begriff Biohardware wird eine Anordnung von Reagenzgläsern (Tubesystem) in Verbindung mit Prozeßapparaturen einschließlich ihrer Befüllung mit DNA, weiteren Reagenzien, Katalysatoren und Hilfsstoffen verstanden, mit der alle im zugrundeliegenden

DNA-Computing-Modell zulässigen Berechnungen ohne Verwendung zusätzlicher externer Hilfsmittel ausgeführt werden können. DNA-Algorithmen bilden die Biosoftware und sind auf niedrigster Ebene durch die Summe aller auf die Biohardware abgestimmten Prozeßparameter beschrieben. Die schrittweise Weiterentwicklung von DNA-Computing-Modellen zu DNA-basierten Universalcomputern läßt sich mit dem historischen Analogon der technischen Entwicklung vom Modell der deterministischen Turingmaschine zu modernen elektronischen Rechnern vergleichen.

Es stellt sich die Frage, welches universelle DNA-Computing-Modell ausgehend vom derzeitigen Stand der Technik am besten für eine Weiterentwicklung zu einem DNA-basierten Universalcomputer geeignet ist. Das wichtigste Auswahlkriterium besteht darin, ob jeder Ableitungsschritt und mithin jede Berechnung mittels des verfügbaren Repertoires molekularbiologischer Operationen zuverlässig und wiederholbar simuliert werden kann. Das heißt, das Abstraktionsniveau der Modelloperationen sollte möglichst niedrig sein. Um die praktische Ausführbarkeit beliebiger Modellalgorithmen sicherzustellen, bedarf es der Existenz einer einheitlichen Kodierung zwischen DNA-basierten Daten (Wörtern formaler Sprachen) im Modell und korrespondierenden DNA-Strängen. Unterzieht man die in Kapitel 4 betrachteten Modelle des DNA-Computing einem Vergleich hinsichtlich der genannten Auswahlkriterien, so werden Splicing-Systeme favorisiert. Die Splicing-Operation setzt unmittelbar auf dem Vorbild der DNA-Rekombination auf und wird durch Erkennungssequenzen parametrisiert, die sich konkreten Restriktionsenzymen zuordnen lassen. Die Modellbeschreibung erfaßt im Gegensatz zu anderen DNA-Computing-Modellen unmittelbar molekularbiologische Prozesse. Abstrahiert wird jedoch von den Seiteneffekten, mit denen diese Prozesse (Digestion und Ligation) behaftet sind. Der unmittelbaren Verwendung von H-Systemen wie auch EH-Systemen zur Konstruktion DNA-basierter Universalcomputer steht entgegen, daß sie eine universelle Berechnungsstärke nur unter Einbeziehung unendlich vieler Axiome (DNA-Stränge) oder Splicing-Regeln (Restriktionsenzyme) erreichen. Es wurden verschiedene Ansätze diskutiert, um durch geeignete Systemerweiterungen universelle Splicing-Systeme mit endlichen Komponenten zu erhalten (z.B. [CFKP_96], [CsKP_96], [DaMi_96], [FrKP_99], [FrMF_98], [MaPa_97], [MaRo_98], [PaRS_97], [PaRS_98], [Paun_98], [SaFe_99], [YoKF_97]) und weiterführend zu optimieren. Diese Ansätze ergeben eine Klassifizierung universeller Splicing-Systeme mit endlichen Komponenten:

- Modifikation der Splicing-Operation
- Einbeziehung gezielt erzeugter nichtlinearer DNA-Strukturen
- Verwendung von Multimengen
- Verteilte Splicing-Systeme (Mehrtubesysteme)

Im Hinblick auf eine Modellnutzung als Grundlage für DNA-basierte Universalcomputer sind jene Systemerweiterungen von Interesse, die Auswahlkriterien für eine mögliche laborpraktische Implementierung genügen.

Nachfolgend werden Ziele der Modellentwicklung definiert, an denen sich die Konstruktion und Optimierung universeller Splicing-Systeme mit endlichen Komponenten orientiert in bezug auf den ersten Entwicklungsschritt zum DNA-basierten Universalcomputer im Vorfeld einer Laborimplementierung. Bekannte Systeme entsprechend der Klassifikation werden bezüglich des Erfüllungsgrades dieser Ziele gegenübergestellt und verglichen. Mit dem speziellen Mehrtubesystem TT6 wird ein anwendungsorientiertes verteiltes Splicing-System eingeführt, welches alle gesetzten Ziele der Modellentwicklung erfüllt. Der Universalitätsnachweis erfolgt konstruktiv durch Angabe einer Transformation von

Chomsky-Grammatiken des Typs 0, die in Kuroda-Normalform vorliegen, in das Mehrtubesystem TT6. Eine Simulationsmöglichkeit des TT6 durch eine definierte Abfolge molekularbiologischer Operationen aus Kapitel 3 wird angegeben.

Ziele der Modellentwicklung

Uneingeschränkte Universalität: Das Splicing-System soll in der Lage sein, jede formale Sprache aus RE zu generieren bzw. zu akzeptieren, konstruktiv nachgewiesen durch Transformation eines als universell bekannten Berechenbarkeitsmodells in das Splicing-System.

Endlichkeit aller Systemkomponenten: Alle im Splicing-System enthaltenen Komponenten (z.B. Axiome, Splicing-Regeln) sollen sich durch endliche Mengen beschreiben lassen.

Statischer Systemaufbau: Die Anzahl benötigter Reaktionsgefäße und Prozeßapparaturen sowie ihre Vernetzung untereinander durch eine geeignete Pipelinestruktur sollen für alle ausführbaren Algorithmen konstant sein. Modellalgorithmen dürfen lediglich die Befüllung von Reaktionsgefäßen und Prozeßapparaturen sowie die Summe aller Prozeßparameter (z.B. Temperatur-Zeit-Verläufe, pH-Wert-Zeit-Verläufe, Zugaben von Reagenzien) auf der niedrigsten Beschreibungsebene enthalten, jedoch keine konstruktiven Änderungen des Systemaufbaus bedingen.

Kein systeminhärenter Nichtdeterminismus: Das Verhalten des Splicing-Systems soll durch eine feste Operationsfolge, die zur Ausführung jedes Ableitungsschrittes durchlaufen wird, beschreibbar sein. Ist das zugrundeliegende Modell der Berechenbarkeit deterministisch, dann auch das daraus entsprechend dem konstruktiven Universalitätsnachweis resultierende Splicing-System.

Multiple-Data-Fähigkeit: Das Splicing-System soll eine möglichst hohe Datenparallelität zur Steigerung der Zeiteffizienz von Berechnungen nutzen.

Einheitliche Kodierfähigkeit DNA-basierter Daten des Modells in DNA:

Gleiche DNA-basierte Daten (z.B. Alphabetsymbole) sollen im gleichen Modellalgorithmus auf gleiche Weise eindeutig in DNA abgebildet werden. Dies bedeutet, daß jede bei Ausführung eines Algorithmus einmal gewählte DNA-Kodierung bis zur Terminierung beibehalten wird.

Separation der Berechnungsergebnisse von Zwischenprodukten: Das Splicing-System soll derart definiert sein, daß alle Berechnungsergebnisse in einer oder mehreren separaten Systemkomponenten bereitgestellt werden und daß diese ausgewiesenen Systemkomponenten außer Berechnungsergebnissen keine Zwischenprodukte der Berechnung enthalten.

Auslesbarkeit jedes Berechnungsergebnisses: Jedes vom Splicing-System ermittelte Berechnungsergebnis soll in einer Form vorliegen, die eine analytische Auswertung zuläßt sowie die Dekodierung des Berechnungsergebnisses in die durch die Berechnung erzielte Problemlösung ermöglicht.

Ressourcenminimierung: Das Splicing-System soll die Erzeugung und Verarbeitung von Zwischenprodukten minimieren, die nicht zum Berechnungsergebnis führen.

Möglichst niedriges Abstraktionsniveau der Modelloperationen: Jede Modelloperation, die als Bestandteil der Ausführung von Ableitungsschritten herangezogen wird, soll formal derart definiert sein, daß die zugrundeliegenden molekularbiologischen Prozesse, Methoden und Techniken nicht übermäßig stark idealisiert, generalisiert und abstrahiert werden.

Simulierbarkeit des Systems: Die Beschreibung des Gesamtsystems soll auf einer simulierbaren Abfolge bekannter molekularbiologischer Prozesse, Methoden und Techniken basieren.

6 Sisyphus – ein labornahes, seiteneffektberücksichtigendes Modell des DNA-Computing

In diesem Kapitel wird das im Ergebnis eigener Arbeiten labornah konzipierte DNA-Computing-Modell Sisyphus (Simulation System for Phenomena undergoing Side Effects) beschrieben. Dieses DNA-Computing-Modell verkörpert ein abstraktes Simulationssystem für beliebige Abfolgen der im Kapitel 3 definierten DNA-Operationen. Es zeichnet sich insbesondere durch die Aufnahme ausgewählter Seiteneffekte der den DNA-Operationen zugrundeliegenden molekularbiologischen Prozesse in die Modellbeschreibung aus. Das DNA-Computing-Modell Sisyphus ist in [HiHS_01] und [HaHS_01] einschließlich ausgewählter laborpraktischer Studien veröffentlicht. Ausgehend von der Modellbeschreibung wird die Universalität von Sisyphus konstruktiv durch Transformation des TT6-EH-Systems in determinierte Abfolgen entsprechender DNA-Operationen gezeigt, so daß über die Systembeschreibung hinaus eine labornah Simulation des TT6-Systemverhaltens vorliegt ([StHi_01]). Weiterhin werden ein effizienter Algorithmus zur Lösung des Rucksackproblems für das Modell Sisyphus konstruiert und notiert sowie Simulationsergebnisse für ausgewählte Operationsfolgen durch entsprechende Laborexperimente verifiziert. In diesem Zusammenhang entstand unter anderem eine laborpraktische Lösung einer Instanz des Rucksackproblems mit der Problemgröße $n = 3$, die besonders als molekularbiologisches Verfahren patentiert und veröffentlicht ist ([SSHK_00], [StSH_01], [HiSt_00a]). Die Modellbeschreibung von Sisyphus, der Universalitätsnachweis, der Beispielalgorithmus zur Lösung des Rucksackproblems sowie die beschriebenen Laborexperimente einschließlich der Lösung einer Instanz des Rucksackproblems widerspiegeln ebenfalls Resultate eigener Arbeiten.

Zur Durchführung von Laborexperimenten mit dem Ziel der Verifikation von Simulationsergebnissen für ausgewählte Operationsfolgen wurde nach Einweisung in Grundlagen molekularbiologischer Arbeitstechniken die Mitnutzung des Labors der Abteilung Chirurgische Forschung an der Medizinischen Fakultät der TU Dresden ermöglicht. Eine Implementierung von Sisyphus in Form eines objektorientierten Simulationssystems mittels der Programmiersprache Java auf Basis der Modellbeschreibung erfolgt am Fraunhofer Institut für Integrierte Schaltungen, Außenstelle für Automatisierung des Schaltkreis- und Systementwurfs Dresden im Rahmen einer wissenschaftlichen Kooperation mit der dort ansässigen Arbeitsgruppe Modellierung und Simulation.

In den vergangenen Jahren haben sich auf dem Gebiet des DNA-Computing zwei grundlegende Forschungsrichtungen herausgebildet. Der Entwicklung, Analyse und Optimierung berechenbarkeitstheoretischer Modelle des DNA-Computing nach rein formalen Aspekten unter Nutzung von Methoden der Theoretischen Informatik stehen laborpraktische

Implementierungen von DNA-Algorithmen zur Lösung spezieller Probleminstanzen gegenüber, wobei die Laborstudien häufig den Status von Pionierexperimenten besitzen. Bereits die Absicherung einer uneingeschränkten Wiederholbarkeit solcher Experimente gilt als große Herausforderung des praktischen DNA-Computing ([KaCL_95]). Beide Forschungsrichtungen – die auf formal-mathematischen Ansätzen beruhende berechenbarkeitstheoretische Modellierung einerseits und pragmatisch-experimentelle laborpraktische Studien andererseits – entwickeln sich weitgehend getrennt voneinander.

Bisher bekannte formale universelle Modelle des DNA-Computing weisen ein hohes Abstraktionsniveau auf. DNA-basierte Daten, Operationen und Mechanismen der Ablaufsteuerung sind stark idealisiert gegenüber realen molekularbiologischen Prozessen. Insbesondere unerwünschte Seiteneffekte dieser Prozesse fanden bisher keine Aufnahme in universelle Modelle des DNA-Computing. Umgekehrt gelang bisher auch keine uneingeschränkte, beliebig reproduzierbare und in jedem möglichen Verarbeitungsschritt modellkorrespondierende laborpraktische Implementierung eines formalen universellen Modells des DNA-Computing.

Mit Sisyphus wird der Versuch unternommen, ein abstraktes, formal beschreibbares und universelles DNA-Computing-Modell zu schaffen, das durch ein niedrigeres Abstraktionsniveau der DNA-basierten Daten wie auch Operationen stärker an die Laborrealität angenähert ist als die im Kapitel 4 untersuchten Vergleichsmodelle. Sisyphus nutzt als Beschreibungsebene für die Operationen auf DNA unmittelbar die Definitionen aus Kapitel 3 in Ergänzung mit ausgewählten, geeignet parametrisierten Seiteneffekten als Bestandteile der Operationswirkungen.

Motiviert durch die nucleotid- und labelgenaue Abbildung von DNA-Strängen liegt das Hauptanwendungsgebiet von Sisyphus in der Konstruktion von DNA-Algorithmen im Vorfeld einer laborpraktischen Implementierung. Insbesondere wird die Suche nach seiteneffektkompensierenden (error-resistenten) Operationsfolgen und DNA-Kodierungen für gewünschte algorithmische Teilaufgaben unterstützt, und es sind Prognosen über erwartete Fortpflanzungen und Einflüsse von Seiteneffekten auf Folgeoperationen möglich. Die Ziele der Modellentwicklung, einhergehend mit resultierenden Eigenschaften von Sisyphus als universelles Modell des DNA-Computing, werden weiterführend benannt und in den Kontext bekannter Arbeiten auf dem Gebiet der Simulation von Prozessen des DNA-Computing eingeordnet.

Ziele der Modellentwicklung

Verwaltung mehrerer Reagenzgläser: Eine endliche Multimenge von DNA-Strängen, die resultierend aus einer definierten Abfolge von Modelloperationen hervorgegangen ist, wird als Reagenzglas bezeichnet. Das Modell Sisyphus soll in der Lage sein, mehrere Reagenzgläser dynamisch nach Anwendungsvorgabe zu verwalten.

Restriktivität: Modelloperationen können Reagenzgläser definiert modifizieren. Sisyphus soll ein restriktives Modell des DNA-Computing verkörpern, so daß auf den Inhalt eines konkreten, als Eingangsoperand dienenden Reagenzglases nur von genau einer Operation zugegriffen werden kann, nicht aber unabhängig von mehreren Operationen parallel.

Multimengenbasiertheit: Es soll ein multimengenbasiertes DNA-Computing-Modell geschaffen werden. Sisyphus führt zu jedem in einem Reagenzglas enthaltenen DNA-

Strang die Anzahl seiner Exemplare. Die Wirkungen der Modelloperationen sind von den jeweils vorliegenden Strangexemplaranzahlen in den einbezogenen Reagenzgläsern abhängig.

Universalität ermöglichender Satz von Modelloperationen: Der Satz von Modelloperationen soll derart gewählt sein, daß ein universelles Modell des DNA-Computing entsteht, jede Modelloperation unmittelbar einen molekularbiologischen Prozeß abstrakt simuliert und zusätzlich formal beschreibbar ist. Sisyphus stellt in Konsequenz folgende Modelloperationen bereit: Synthesis, Annealing, Melting, Union, Split (Aufteilen eines Reagenzglasinhaltes auf mehrere Reagenzgläser), Ligation, Digestion, Labeling (5'-Phosphorylierung, 5'-Dephosphorylierung, 5'-Biotinylierung), Polymerisation, Affinity Purification, Gel Electrophoresis und Separation by Length (Gel Electrophoresis und anschließende Isolation von DNA-Strängen vorgegebener Länge). Aus determinierten Abfolgen der genannten Operationen lassen sich die Makrooperationen PCR und Extraction (Separation nach Subsequenz) zusammensetzen, die ebenfalls Eingang in den Operationsatz finden. Jeder Aufruf einer Modelloperation soll die Vorgabe aller jeweils spezifischen Operations- und Seiteneffektparameter erfordern.

Seiteneffektberücksichtigung: Das Modell Sisyphus soll Wirkungen ausgewählter Seiteneffekte der molekularbiologischen Prozesse, welche durch die Modelloperationen simuliert werden, in den Operationsbeschreibungen berücksichtigen. Zu diesem Zweck erfolgt eine geeignete statistische Parametrisierung der entsprechenden Seiteneffekte. Resultierende Seiteneffektparameter bilden zusätzliche spezifische Eingangsoperanden jeder Modelloperation. Durch jeden erfaßten Seiteneffektparameter kann die Intensität des zugrundeliegenden Seiteneffektes feinkörnig zwischen den Extremen „kein Einfluß“ und „absolute Dominanz“ variiert werden.

Nichtdeterministischer Ansatz: Die Steuerung von Interaktionen zwischen verschiedenen DNA-Strängen innerhalb eines Reagenzglases wie auch die Auswahl von DNA-Strängen für Seiteneffekteinflüsse und ihre konkrete Wirkung unterliegen Zufallsentscheidungen. Die Steuerung von Interaktionen betrifft die Festlegung, welche DNA-Stränge zu welchem Zeitpunkt einer Operationsausführung zur abstrakten und diskreten Simulation der jeweiligen molekularen Reaktion zusammengeführt werden.

Nutzung des imperativen Paradigmas: Algorithmen im Modell Sisyphus sollen durch determinierte Abfolgen (Sequenzen) von Modelloperationen und den jeweiligen Parameterbelegungen dieser Operationen vollständig beschreibbar sein, wobei voneinander unabhängige Teilfolgen von Modelloperationen in getrennten Reagenzgläsern auch nebenläufig notiert werden können. Die Abfolge der auszuführenden Modelloperationen ist durch den Algorithmus fest vorgegeben.

Visualisierung von Reagenzgläsern: Im Modell Sisyphus soll jedes entsprechend der Anwendervorgabe definierte Reagenzglas nach Ausführung jeder einzelnen Modelloperation visualisiert werden können. Visualisierung eines Reagenzglases bedeutet in diesem Zusammenhang die Ausgabe derjenigen Multimenge von DNA-Strängen, die das betreffende Reagenzglas bildet.

Protokollierung von Reagenzgläsern über Operationsfolgen: Im Sinne eines Simulationssystems soll das Modell Sisyphus in der Lage sein, nach Anwendervorgabe die jedem erfaßten Reagenzglas zugrundeliegende Multimenge von DNA-Strängen vor

und nach Ausführung jeder Modelloperation gemeinsam mit den ggf. herangezogenen Zufallszahlen geeignet zu protokollieren, so daß Strangmodifikationen infolge von Operationswirkungen über endliche Operationsfolgen rekonstruiert und gezielt reproduziert werden können.

7 Zusammenfassung und Ausblick

Die vorliegende Arbeit beleuchtet das Forschungsgebiet des DNA-Computing vordergründig aus dem Blickwinkel der Berechenbarkeitstheorie. Universelle sowie platzbeschränkt universelle Modelle des DNA-Computing, deren DNA-basierte Daten auf der Abbildung linearer DNA beruhen, wurden ausführlich untersucht, klassifiziert und als Beschreibungssysteme für Algorithmen angewendet. Im einzelnen erfüllen folgende DNA-Computing-Modelle die vorgenannten Auswahlkriterien: die Filtering-Modelle nach Adleman, Lipton und Amos, das Modell DNA-Pascal, das Modell Parallel Associative Memory, das Modell DNA-Equality Checking, Watson-Crick D0L-Systeme, Insertion-Deletion-Systeme und Splicing-Systeme (H-Systeme wie auch EH-Systeme). Die Klassifizierung bezieht sich auf folgende Modelleigenschaften: Restriktivität, Multimengenbasiertheit, Determinismus, innewohnendes Paradigma (imperativ, funktional, logisch) und Fähigkeiten zur Parallelverarbeitung (Multiple-Instruction-Fähigkeit, Multiple-Data-Fähigkeit). Es gelang sowohl die einheitlich formale Beschreibung dieser Modelle im Sinne

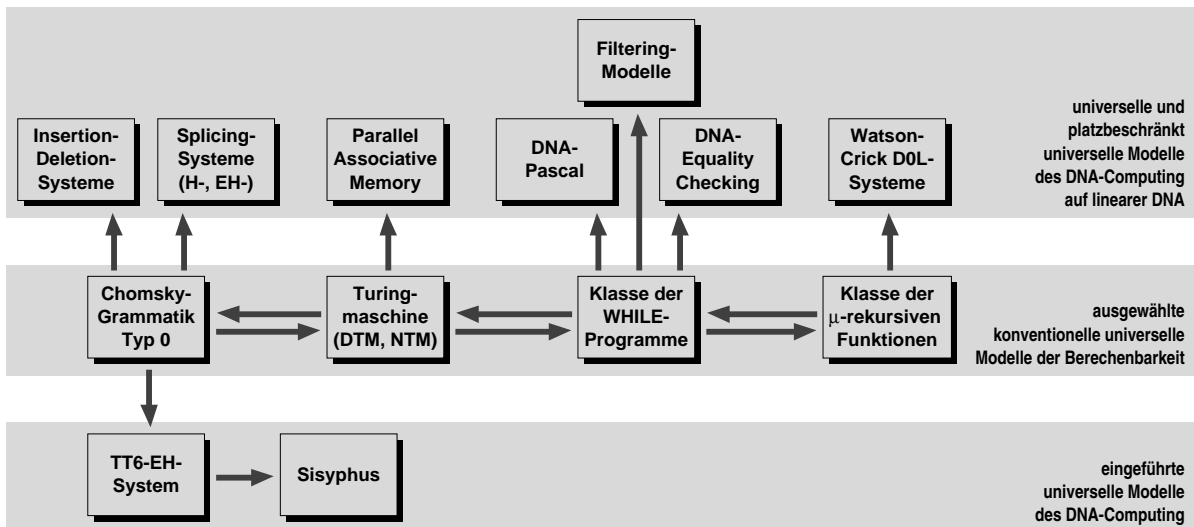


Abbildung 1: Beschriebene Transformationen von Modellen der Berechenbarkeit

von Modellen der Berechenbarkeit als auch die Konstruktion effizienter Modellalgorithmen zur Lösung des als NP-vollständig bekannten Rucksackproblems. Mit Ausnahme der Watson-Crick D0L-Systeme konnten durchgängig Algorithmen entwickelt werden, die sich durch eine bezüglich der Problemgröße lineare Zeitkomplexität auszeichnen. Die Algorithmen sind hierbei unter Verwendung der Beschreibungsmittel der jeweiligen Modelle notiert und ihre Abarbeitung beispielhaft anhand von Probleminstanzen untersetzt. Die Universalität der betrachteten DNA-Computing-Modelle konnte in jedem Fall durch Angabe einer Transformation eines konventionellen universellen Modells der Berechenbarkeit in das jeweilige DNA-Computing-Modell konstruktiv gezeigt werden, siehe Abbil-

dung 1. Für Watson-Crick D0L-Systeme, Insertion-Deletion-Systeme und EH-Systeme waren die entsprechenden Nachweise bereits bekannt, für die restlichen untersuchten Modelle sind sie im Rahmen dieser Arbeit entwickelt worden. Als Ausgangspunkt für die Modelltransformationen dienten folgende konventionelle universelle Modelle der Berechenbarkeit: deterministische und nichtdeterministische Turingmaschine, die Klasse der WHILE-Programme, die Klasse der μ -rekursiven Funktionen sowie Chomsky-Grammatiken vom Typ 0 zur Erzeugung rekursiv aufzählbarer formaler Sprachen. Diese Modelle können sich gegenseitig simulieren, wobei die zugrundeliegenden Modelltransformationen als Bestandteil der mathematischen Grundlagen des DNA-Computing angegeben wurden.

Bei der Analyse der untersuchten bekannten DNA-Computing-Modelle hinsichtlich einer laborpraktischen Implementierbarkeit mittels Abfolgen wohluntersuchter molekularbiologischer Prozesse fällt auf, daß die Modelle ein sehr hohes Abstraktionsniveau aufweisen und mithin stark idealisiert sind. Jedes dieser Modelle ist entweder nicht restriktiv, nicht multimengenbasiert, nicht endlichkomponentig oder enthält Modelloperationen, für die es derzeit keine geeignete praktische Implementierungsmöglichkeit gibt. Diese Eigenschaften stehen der Verwendung dieser Modelle als Basis für den Aufbau universeller DNA-Computer grundsätzlich entgegen.

Zur Beseitigung dieser Unzulänglichkeit wurden mit dem TT6-EH-System und dem Simulationssystem Sisyphus zwei universelle DNA-Computing-Modelle eingeführt, deren Modelleigenschaften labornah ausgerichtet sind und den interdisziplinären Charakter des DNA-Computing unterstreichen. Das TT6-EH-System stellt ein endlichkomponentiges verteiltes Splicing-System dar, das sich durch einen statischen Systemaufbau, eine Minimierung der in die Verarbeitung einbezogenen Ressourcen und ein niedriges Abstraktionsniveau der Modelloperationen auszeichnet. Berechnungsergebnisse werden im TT6-EH-System konsequent von Zwischenprodukten des Berechnungsprozesses getrennt und in einer separaten Systemkomponente angereichert. Das Modell impliziert keinen systeminhärenten Nichtdeterminismus und ist durch eine feste Operationsfolge, die in jedem Ableitungsschritt (Arbeitsschritt) durchlaufen wird, beschreibbar. Alle Modelloperationen haben wohluntersuchte molekularbiologische Prozesse zum Vorbild, die nur geringfügig abstrahiert sind. Die Universalität des Modells ist konstruktiv nachgewiesen worden durch Angabe einer Transformation beliebiger Chomsky-Grammatiken vom Typ 0, die in Kuroda-Normalform vorliegen, in ergebnisäquivalente TT6-EH-Systeme. Die Arbeitsweise des Systems wurde anhand eines Beispielalgorithmus zur Lösung des Rucksackproblems demonstriert. Eine laborpraktische Implementierbarkeit von TT6-EH-Systemen setzt eine seiteneffektfreie Ausführung aller Modelloperationen voraus.

Diesem Umstand Rechnung tragend, wurde mit dem Simulationssystem Sisyphus ein formal beschreibbares universelles Modell des DNA-Computing geschaffen, das erstmals Seiteneffekte der den Modelloperationen zugrundeliegenden molekularbiologischen Prozesse in der Modellbeschreibung berücksichtigt. Zusätzlich besitzt das Modell die Eigenschaften „restriktiv“ und „multimengenbasiert“. Die Modelloperationen wurden auf der Beschreibungsebene von Nucleotiden und Strangendenmarkierungen, aus denen sich die DNA-basierten Daten zusammensetzen, unmittelbar aus molekularbiologischen Prozessen abgeleitet und spezifiziert. Folgender Universalität ermöglichender Satz von Modelloperationen steht zur Verfügung: Synthesis, Annealing, Melting, Union, Split, Ligation, Digestion, Labeling, Polymerisation, PCR, Affinity Purification, Extraction, Gel Electrophoresis und Separation by Length. Dieser Operationssatz korrespondiert mit genau dem Repertoire an molekularbiologischen Prozessen, die in bisherigen bekannten labor-

praktischen Studien zum DNA-Computing Verwendung fanden. Die Wirkungsintensität ausgewählter Seiteneffekte ist in Sisyphus durch geeignete statistische Parameter erfaßt. Folgende Seiteneffekte wurden berücksichtigt: Auftreten von Punktmutationen und Deletions, Basenfehlpaarungsrate, Rate unverarbeiteter DNA-Stränge sowie Strangverluste. Jeder Modelloperation ist eine zutreffende Auswahl aus diesen Seiteneffekten spezifisch zugeordnet. Der Universalitätsnachweis von Sisyphus erfolgte durch Simulation von TT6-EH-Systemen. Ein Beispielalgorithmus zur Lösung des Rucksackproblems, der weitgehend gegenüber erfaßten Seiteneffekteinflüssen robust ist, konnte konstruiert werden. Anhand einer Problem Instanz der Größe $n = 3$ sowie eines unabhängigen Beispiels einer PCR von synthetisierter Template-DNA mit darauf abgestimmten Primern erfolgte eine laborpraktische Verifikation.

Die Thematik der vorliegenden Arbeit bietet mehrere Ansatzpunkte für weiterführende Überlegungen. Das Simulationssystem Sisyphus eignet sich zur Anwendung in der Informatik als Tool für die Algorithmenkonstruktion im weiteren Sinne. Sisyphus verkörpert ein abstraktes Modell der Berechenbarkeit. Es ist denkbar, die Beschreibungsebene DNA-basierter Daten wie auch der Modelloperationen noch weiter an die Laborrealität anzunähern, indem auch chemische und physikalische Prozeßparameter (wie beispielsweise Temperatur, pH-Wert) erfaßt werden und in eine Simulation auf atomarer Beschreibungsebene einfließen. Für die Hybridisierung von DNA sind entsprechende mathematische Ansätze bereits bekannt, die die Notwendigkeit einer Abbildung nichtlinearer DNA nach sich ziehen.

Betrachtungen zum in-vivo-DNA-Computing erlangen in der aktuellen internationalen Forschung eine wachsende Bedeutung. Die lebende Zelle, in die die DNA eingeschlossen ist, gestattet zusätzliche Wechselwirkungen, die sich in-vitro-beschränkten Betrachtungen entziehen. Die abstrakte Beschreibung entsprechender Vorgänge im Sinne eines Computingkonzepts mündet ebenfalls in die Beschreibung von Modellen der Berechenbarkeit, die mit dem Begriff *P-Systeme* bezeichnet werden. P-Systeme begründen ein Spezialgebiet des DNA-Computing, für das sich gegenwärtig die Bezeichnung *Membrane Computing* herausbildet. Aus Sicht der Theoretischen Informatik liegen attraktive Aufgaben in der berechenbarkeits- und komplexitätstheoretischen Untersuchung von P-Systemen sowie in ihrer Optimierung hinsichtlich ihres minimalen Ressourcenbedarfs in Abhängigkeit erzielbarer Berechnungsstärken.

Literatur

- [Adle_94] L.M. Adleman. *Molecular computation of solutions to combinatorial problems*. Science **266**:1021–1024, 1994
- [Adle_96] L.M. Adleman. *On constructing a molecular computer*. In R.J. Lipton, E. Baum, editors, DNA Based Computers, American Mathematical Society, DIMACS Vol. 27, pp. 1–22, 1996
- [AlBu_94] R.C. Allen, B. Budowle. *Gel electrophoresis of proteins and nucleic acids: selected techniques*. Verlag de Gruyter Berlin, 1994
- [Alph_98] L. Alphey. *DNA-Sequenzierung*. Spektrum Akademischer Verlag Heidelberg, Berlin, 1998

- [Amos_97] M. Amos. *DNA Computation*. Thesis for the degree of Doctor of Philosophy, University of Warwick, 1997
- [BeSi_93] P. Berg, M. Singer. *Die Sprache der Gene. Grundlagen der Molekulargenetik*. Spektrum Akademischer Verlag Heidelberg, Berlin, Oxford, 1993
- [BJRH_00] R.S. Braich, C. Johnson, P.W.K. Rothemund, D. Hwang, N. Chelyapov, L.M. Adleman. *Solution of a Satisfiability Problem on a Gel-Based DNA Computer*. In A. Condon, G. Rozenberg, editors, *DNA Computing, Proceedings Sixth International Meeting on DNA Based Computers*, University of Leiden, The Netherlands, Series Lecture Notes in Computer Science, Vol. 2054, Springer Verlag, pp. 27–42, 2000
- [Börg_92] E. Börger. *Berechenbarkeit, Komplexität, Logik*. Vieweg Verlag, Wiesbaden, 1992
- [BrSM_00] I.N. Bronstein, K.A. Semendjajew, G. Musiol. *Taschenbuch der Mathematik*. Verlag Harri Deutsch, Frankfurt/Main, 2000
- [Bugg_97] T. Bugg. *An introduction to enzyme and coenzyme chemistry*. Blackwell Science, Oxford, 1997
- [CaCD_98] C. Calude, J. Casti, M.J. Dinnen. *Unconventional Models of Computation*. Springer Verlag, 1998
- [CFKP_96] E. Csuhaj-Varju, R. Freund, L. Kari, G. Păun. *DNA computing based on splicing: universality results*. In L. Hunter T.E. Klein, editors, *Proceedings First Annual Pacific Symposium on Biocomputing*, Hawaii, World Scientific, pp. 179–190, 1996
- [CrKa_99] P. Crescenzi, V. Kann. *A compendium of NP optimization problems*. In G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, M. Protasi. *Complexity and Approximation. Combinatorial optimization problems and their approximability properties*. Springer Verlag, 1999
- [CsKP_96] E. Csuhaj-Varju, L. Kari, G. Păun. *Test tube distributed systems based on splicing*. *Computers and AI* **15(2–3)**:211–232, 1996
- [DaEr_98] M.J. Daley, M.G. Eramian. *Models of DNA Computation*. CS881b Term Report, University of Western Ontario, Canada, 1998
- [DaMi_96] J. Dassow, V. Mitrana. *Splicing grammar systems*. *Computers and AI* **15(2–3)**:109–122, 1996
- [DrPo_92] D. Dressler, H. Potter. *Katalysatoren des Lebens: Struktur und Wirkung von Enzymen*. Spektrum Akademischer Verlag Heidelberg, Berlin, New York, 1992
- [EMCG_98] H. Ehrig, B. Mahr, F. Cornelius, M. Große-Rhode, P. Zeitz. *Mathematisch-strukturelle Grundlagen der Informatik (Springer-Lehrbuch)*. Springer Verlag, 1998
- [FrKP_99] R. Freund, L. Kari, G. Păun. *DNA computing based on splicing: The existence of universal computers*. *Theory of Computing Systems* **32(1)**:69–112, 1999
- [FrMF_98] P. Frisco, G. Mauri, C. Ferretti. *Simulating Turing machines through extended mH systems*. In G. Păun, editor, *Computing with Bio-Molecules. Theory and Experiments*, Springer Verlag Singapore, pp. 221–238, 1998
- [HaBa_90] R. Hagemann, F. Baldauf. *Gentechnologische Arbeitsmethoden: ein Handbuch experimenteller Techniken und Verfahren*. Akademie-Verlag Berlin, 1990

- [HaHS_01] U. Hatnik, T. Hinze, M. Sturm. *A Probabilistic Approach to Description of Molecular Biological Processes on DNA and Their Object Oriented Simulation*. In V.V. Kluev, N.E. Mastorakis, editors, Proceedings WSES International Conference on Simulation 2001, ISBN 960-8052-40-8, Malta, 2001
- [Head_87] T. Head. *Formal language theory and DNA: An analysis of the generative capacity of specific recombinant behaviors*. Bulletin of Mathematical Biology **49(6)**:737-759, 1987
- [HiHS_01] T. Hinze, U. Hatnik, M. Sturm. *An object-oriented simulation of real occurring molecular biological processes for DNA computing and its experimental verification*. In N. Jonoska, N. Seeman, editors, Proceedings Seventh International Meeting on DNA Based Computers, University of South Florida, Tampa, USA, 2001, Series Lecture Notes in Computer Science, Springer Verlag, in press
- [HiSt_00a] T. Hinze, M. Sturm. *A universal functional approach to DNA computing and its experimental practicability*. In A. Condon, G. Rozenberg, editors, Preliminary Proceedings Sixth International Meeting on DNA Based Computers, University of Leiden, Leiden, The Netherlands, p. 257, 2000 und Technical Report TUD-FI00-05, ISSN 1430-211X, Technische Universität Dresden, 2000
- [HiSt_00b] T. Hinze, M. Sturm. *Towards an in-vitro Implementation of a Universal Distributed Splicing Model for DNA Computation*. In R. Freund, editor, Proceedings Theorietag 2000, ISBN 3-85028-325-9, Technische Universität Wien, Austria, pp. 185-189, 2000
- [HoUl_00] J.E. Hopcroft, J.D. Ullman. *Einführung in die Automatentheorie, Formale Sprachen und Komplexitätstheorie*. Oldenbourg Verlag München, 2000
- [JoDa_95] J. Jones, S.G. Davies. *Synthese von Aminosäuren und Peptiden*. VCH Weinheim, 1995
- [KaCL_95] P.D. Kaplan, G. Cecchi, A. Libchaber. *Molecular computation: Adleman's experiment repeated*. Technical report No. 95-120, NEC Research Institute, 1995
- [KaDK_94] P. Karlson, D. Doenecke, J. Koolman. *Kurzes Lehrbuch der Biochemie für Mediziner und Naturwissenschaftler*. Georg Thieme Verlag Stuttgart, New York, 1994
- [Kari_97] L. Kari. *From Micro-soft to Bio-soft: Computing with DNA*. In D. Lundh, B. Olsson, A. Narayanan, editors, Proceedings BioComputing and Emergent Computation (BCEC97), University of Skovde, Sweden, World Scientific, 1997
- [KaTh_96] L. Kari, G. Thierrin. *Contextual insertion/deletion and computability*. Information and Computation **131(1)**:47-61, 1996
- [KBL_94] H. Kleine-Büning, T. Lettmann. *Aussagenlogik, Deduktion und Algorithmen*. Teubner Verlag Stuttgart, 1994
- [Knip_97] R. Knippers. *Molekulare Genetik*. Georg Thieme Verlag Stuttgart, 1997
- [Lipt_95] R.J. Lipton. *DNA solution of hard computational problems*. Science **268(5210)**:542-545, 1995
- [Loec_76] J. Loeckx. *Algorithmtheorie*. Springer Verlag, 1976
- [LoZo_98] F. Lottspeich, H. Zorbas. *Bioanalytik*. Spektrum Akademischer Verlag Heidelberg, 1998

- [MaPa_97] C. Martin-Vide, G. Păun. *Cooperating Distributed Splicing Systems*. CDMTCS Research Report No. 71, Centre for Discrete Mathematics and Theoretical Computer Science, University of Auckland, New Zealand, 1997
- [MaRo_98] M. Margenstern, Y. Rogozhin. *Time-varying distributed H systems of degree 2 generate all RE languages*. In Preliminary Proceedings Workshop on Frontiers of Universality at 23rd International Symposium Mathematical Foundations of Computer Science (MFCS98), Brno, Czech Republic, 1998
- [Mart_96] R. Martin. *Elektrophorese von Nucleinsäuren*. Spektrum Akademischer Verlag Heidelberg, 1996
- [MiSa_97] V. Mihalache, A. Salomaa. *Lindenmayer and DNA: Watson-Crick DOL systems*. Bulletin of the European Association for Theoretical Computer Science **62**:160–175, 1997
- [MVPS_98] C. Martin-Vide, G. Păun, A. Salomaa. *Characterizations of recursively enumerable languages by means of insertion grammars*. Theoretical Computer Science **205(1–2)**:195–205, 1998
- [NeGr_97] C.R. Newton, A. Graham. *PCR*. Spektrum Akademischer Verlag Heidelberg, Berlin, 1997
- [PaRS_96] G. Păun, G. Rozenberg, A. Salomaa. *Computing by splicing*. Theoretical Computer Science **168(2)**:321–336, 1996
- [PaRS_97] G. Păun, G. Rozenberg, A. Salomaa. *Computing by splicing: Programmed and evolving systems*. Proceedings 1997 IEEE International Conference on Evolutionary Computation (ICEC97), Indianapolis, USA, IEEE Press, pp. 273–277, 1997
- [PaRS_98] G. Păun, G. Rozenberg, A. Salomaa. *DNA Computing. New Computing Paradigms*. Springer Verlag Berlin, Heidelberg, New York, 1998
- [Paun_96b] G. Păun. *Regular Extended H Systems are Computationally Universal*. Journal of Automata, Languages and Combinatorics **1(1)**:27–36, 1996
- [Paun_98] G. Păun. *DNA computing based on splicing: universality results*. In M. Margenstern, editor, Proceedings Second International Colloquium Universal Machines of Computations (MCU98), Metz, France, Vol. 1, pp. 67–91, 1998
- [Reif_95] J.H. Reif. *Parallel molecular computation: Models and simulations*. In Proceedings Seventh Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA95), Santa Barbara, USA, Association for Computing Machinery, pp. 213–223, 1995
- [Reif_99] J.H. Reif. *Parallel Biomolecular Computation: Models and Simulations*. Algorithmica **25**:142–175, 1999
- [RoSa_97] G. Rozenberg, A. Salomaa. *Handbook of Formal Languages*. Volumes 1 – 3. Springer Verlag Berlin, Heidelberg, New York, 1997
- [RoWa_95] D. Rooß, K.W. Wagner. *On the power of DNA-computers*. Technical Report, Universität Würzburg, 1995
- [RoWa_96] D. Rooß, K.W. Wagner. *On the power of DNA computing*. Information and Computation **131(2)**:95–109, 1996

- [RWBC_99] S. Roweis, E. Winfree, R. Burgoyne, N.V. Chelyapov, M.F. Goodman, P.W.K. Rothemund, L.M. Adleman. *A sticker based model for DNA computation*. In L. Landweber, E. Baum, editors, DNA Based Computers II, Proceedings Second Annual Meeting on DNA Based Computers, Princeton University, USA, American Mathematical Society, DIMACS Vol. 44, pp. 1–30, 1999
- [SaFe_99] Y. Sakakibara, C. Ferretti. *Splicing on tree-like structures*. In H. Rubín, D.H. Wood, editors, DNA Based Computers III, Proceedings 3rd DIMACS Workshop on DNA Based Computers, University of Pennsylvania, USA, American Mathematical Society, DIMACS Vol. 48, pp. 348–358, 1999
- [Schö_97] U. Schöning. *Theoretische Informatik – kurzgefaßt*. Spektrum Akademischer Verlag Heidelberg, 1997
- [Sosi_01] P. Sosik. *DOL System + Watson-Crick Complementarity = Universal Computation*. In M. Margenstern, Y. Rogozhin, editors, Machines, Computations, and Universality, Proceedings Third International Conference MCU2001, Chisinau, Moldavia, Series Lecture Notes in Computer Science, Vol. 2055, Springer Verlag, pp. 308–319, 2001
- [SSHK_00] E.P. Stoschek, M. Sturm, T. Hinze, O.N. Koufaki, M. Hauses, H.K. Schackert. *Molekularbiologisches Verfahren zur Lösung von NP-Problemen*. Deutsches Patent DE 198 53 726 A1, IPC C12N 15/10, Deutsches Patentamt München, 2000
- [StHi_01] M. Sturm, T. Hinze. *Distributed Splicing of RE with 6 Test Tubes*. In C.S. Calude, M.J. Dinnen, G. Păun, editors, Preliminary Proceedings Workshop on Multiset Processing 2000, Curtea de Arges, Romania, CDMTCS Research Report No. 140, pp. 236–248, 2000 und Romanian Journal of Information Science and Technology, ISSN 1453–8245, Editura Academiei Romane 4(1–2):211–234, 2001
- [StSH_01] E.P. Stoschek, M. Sturm, T. Hinze. *DNA-Computing – ein funktionales Modell im laborpraktischen Experiment*. Informatik Forschung und Entwicklung, Springer Verlag 16(1):35–52, 2001
- [Tren_94] R.J. Trent. *Molekulare Medizin: eine Einführung*. Spektrum Akademischer Verlag Heidelberg, 1994
- [Wats_93] J.D. Watson. *Rekombinierte DNA*. Spektrum Akademischer Verlag Heidelberg, 1993
- [Yage_86] R.R. Yager. *On the theory of bags*. International Journal of General Systems 13:23–37, 1986
- [YoKF_97] T. Yokomori, S. Kobayashi, C. Ferretti. *On the Power of Circular Splicing Systems and DNA Computability*. Proceedings 1997 IEEE International Conference on Evolutionary Computation (ICEC97), Indianapolis, USA, IEEE Press, pp. 219–227, 1997
- [YoKo_99] T. Yokomori, S. Kobayashi. *DNA-EC: A Model of DNA-Computing Based on Equality Checking*. In H. Rubín, D.H. Wood, editors, DNA Based Computers III, Proceedings 3rd DIMACS Workshop on DNA Based Computers, University of Pennsylvania, USA, American Mathematical Society, DIMACS Vol. 48, pp. 347–360, 1999