# Zur Programmierung raumzeitlicher diskreter Systeme

Hermann von Issendorff
Institut für Netzwerkprogrammierung
21745 Hemmoor

**Zusammenfassung:**

Die belebte Natur macht uns die Programmierung raumzeitlicher Systeme vor: Sie erzeugt Ketten von Aminosäuren durch sequentielles Übersetzen von Gencode. Die fertige Aminosäurenkette faltet sich in geeigneter Umgebung zu einem Protein zusammen, d.h. ein eindimensionales Gebilde wird zu einem dreidimensionalen. Der Vorgang kann durch Änderung der Umgebung umgekehrt werden, d.h. zwischen Aminosäurenkette und Protein besteht eine bijektive und bikontinuierliche Abbildung, ein Homöomorphismus. Mit anderen Worten: Die Aminosäurenkette stellt ein Programm dar, in dem die Information über die dreidimensionale Struktur des Proteins vollständig enthalten ist.

Die räumliche Struktur eines Proteins entsteht chemisch/physikalisch dadurch, dass die Aminosäuren neben der Kettenbindung untereinander zusätzliche schwächere Bindungen enthalten. Deren Bindungsstärke hängt vom Abstand der verbundenen Aminosäuren ab und kann daher als analoges Signal interpretiert werden. Durch äusseren Einfluss, z.B. durch ein sich anlagerndes weiteres Molekül, kann das Protein in eine andere metastabile homöomorphe Struktur übergehen. Mit anderen Worten: Ein Protein kann induziert verschiedene Zustände annehmen und ist damit zur Informationsspeicherung und -verarbeitung in der Lage.

Der Vorgehensweise der Natur folgend führen wir in dieser Arbeit eine mehrsortige Termalgebra ein, die die Sprache der Natur enthält, d.h. sowohl diskrete räumliche Strukturen als auch darauf ablaufende Datenverarbeitung linear darzustellen erlaubt.

Ein reales diskretes System besteht aus einer Menge materieller Komponenten, die aktiv oder statisch sein können. Sind die Komponenten aktiv, d.h. produzieren sie reale Objekte oder werten sie Funktionen aus, dann sind sie zeitlich gerichtet und werden in einer partiellen zeitlichen Ordnung aktiviert. Sind sie statisch, dann kann man ihnen eine partielle Aufbauordnung zuweisen, die ebenfalls eine zeitliche Richtung induziert.

Abstrahiert man ein reales diskretes System, beispielsweise einen Rechner, von seiner Metrik, d.h. von den räumlichen Abmessungen der Komponenten, dann schrumpft das System zu einem dreidimensionalen gerichteten Netz von ausführbaren Aktivitäten, die in den Komponenten realisiert sind. Abstrahiert man ein reales diskretes System, beispielsweise wieder einen Rechner, von seiner Funktionalität, dann bleibt ein dreidimensionales Netz von Bausteinen übrig, die die Abmessungen der Komponenten haben. Abstrahiert man zugleich von der Metrik und der Funktionalität, dann bleibt ein dreidimensionales Netz von Knoten übrig, das nur noch die Abhängigkeiten der Knoten zeigt. Zwei beliebige Knoten dieses Netzes können abhängig sein oder nicht, und jeder abhängige Knoten kann mit einer endlichen Menge von Vorgänger- bzw. Nachfolgerknoten verbunden sein.

Gibt man den Netzknoten ihre ursprünglichen funktionalen und metrischen Eigenschaften zurück, dann erhält man wieder das ursprüngliche System. Das Knotennetz ist daher die strukturelle Grundlage aller realen diskreten Systeme. Wenn daher eine formale Sprache für die vollständige Beschreibung eines Knotennetzes existiert, dann ist diese die gemeinsame Grundsprache aller diskreten physikalischen Systeme. Akton-Algebra hat diese Eigenschaft.

Die Abstraktion von Metrik und Funktionalität bedeutet nicht etwa Abstraktion von Raum und Zeit, was ein Knotennetz auf einen gerichteten Graph reduzieren würde. Die Raum- und Zeit-Beziehungen zwischen Netzknoten sind vielmehr genau die Eigenschaften, auf der die Akton-Algebra aufbaut. Ein Knoten, d.h. die metrische und funktionale Abstraktion einer Komponente, hat eine endliche räumliche Ausdehnung, und die Querung eines Knoten von der Vorderseite zur Rückseite braucht eine endliche Zeit. Die Aktivität einer Komponente verschwindet also nicht bei metrischer und funktionaler Abstraktion, sondern reduziert sich lediglich auf die rudimentäre Aktivität einer Signalübertragung. Um diese Eigenschaft zu hervorzuheben, wurde das Wort "Akton" als allgemeiner Bezeichner für konkrete Komponenten und deren funktionalen oder metrischen Abstraktionen gewählt.

Im ersten Teil des Vortrags wird zunächst gezeigt, dass sich die Akton-Algebra systematisch aus einer Hierarchie von Aktonsorten aufbauen lässt. Bedeutsam ist, dass sich die Planarisierung und Linearisierung einer räumlichen Struktur durch Einführung von je einem topologischen Cut erreichen lassen. Sodann wird gezeigt, dass sich mit dieser abstrakten Akton-Algebra bereits genetische Grundelemente wie der genetische 4er-Code, DNA- und RNA-Strukturen, und ebenso

Aminosäureketten, und deren strukturelle Proteinvorstufen wie α-Helices und ß-Sheets darstellen lassen.

Der zweite Teil des Vortrags hat die funktionale und metrische Konkretisierung der Akton-Algebra zum Thema. Als Beispiel dient die digitale Datenverarbeitung, insbesondere Speicherung, Codierung und Dekodierung. Dabei wird auch auf die Zerlegung eines Aktonausdrucks in die adressgesteuerte Statement-Folge klassischer von-Neumann-Programme eingegangen. Die metrische Konkretisierung von Aktonausdrücken wird sodann am Beispiel eines planaren Layouts bei Einbettung in ein rechtwinkliges Raster behandelt.