

Die rasante Entwicklung digitaler Technologien führt zu immer neuen Anwendungsmöglichkeiten im Bereich der Architektur. Inwiefern verändern sich dadurch die Entwürfe – und die Rolle des Entwerfers? Ein kurzer Überblick über neuartige technische Raffinessen und ein Planungsbeispiel, das eine konkrete Anwendung illustriert: Mittels einer Software handeln eine Vielzahl von Grundstücken in einem dynamischen Prozess ihre Koordinaten untereinander aus, um die spezifischen Wünsche möglichst vieler zukünftiger Bewohner zu erfüllen.

# Programmieren statt zeichnen?

Vom Einfluss digitaler Technologie auf den architektonischen Entwurf Oliver Fritz

Diskutiert man den Einfluss des Computers auf den architektonischen Entwurf, hat man schnell Echtzeit-Begehungen von virtuellen Räumen, 3-D-Interfaces und High-End-Rendings vor dem geistigen Auge. Tatsächlich waren vor wenigen Jahren diese Bereiche die wichtigsten Aspekte der CAAD-Forschung (*computer aided architectural design*). Bei nüchterner Betrachtungsweise stellt man jedoch fest, dass die Hollywood-Industrie diese Technologien heute mit grösster Selbstverständlichkeit benutzt – mit einer Geschwindigkeit und einem finanziellen Einsatz, mit denen die Hochschulforschung nicht mehr konkurrieren kann. Viele dieser Technologien gehören inzwischen zum Mainstream und haben erfolgreich Einzug in die meisten Architekturbüros gehalten. In Wettbewerbsausstellungen sind häufig übernatürlich wirkende Rendings zu sehen, welche die Architektur in einem Detaillierungsgrad und einer Materialität darstellen, die im eigentlichen Entwurf noch gar nicht erreicht sein können. Dieser sich auf Darstellungsmethodiken beschränkte Eklektizismus hat jedoch nur mittelbar mit den Inhalten oder der Gestaltung des Architekturentwurfs zu tun.

Eine weitaus bedeutendere Rolle für den Entwurfsprozess spielt die Verbreitung des Internets. Die Vernetzung und Internationalisierung in der Architekturszene wird durch die Möglichkeit, Originaldaten unmittelbar und verlustfrei weltweit zu kommunizieren, stark gefördert.

Gegenstand dieses Artikels ist die Auseinandersetzung mit der Frage, inwieweit der Computer durch seine Möglichkeit, viele Varianten zur Lösung eines definierten Problems zu liefern, Einfluss auf die Architektur hat. Oder – anders formuliert: Kann das Programm einer Architektur mit einer Software beschrieben werden? Welchen Einfluss hat diese Perspektive auf das gegenwärtige und zukünftige Entwerfen, auf das Bauen und auf die Architekturtheorie?

## Aktuelle Strömungen in der Architektur

*Entwerfen nach Zahlen:* Betrachtet man die aktuellen Veröffentlichungen von MVRDV oder von OMA, so fällt auf, dass statistische Analysen in der Bewältigung architektonischer Fragestellungen eine wichtige Rolle spielen. In unterschiedlicher Manier werden Statistiken zur Architektur in Verhältnis gesetzt: Sie dienen dazu, Entwürfe zu legitimieren (Stadtpla-

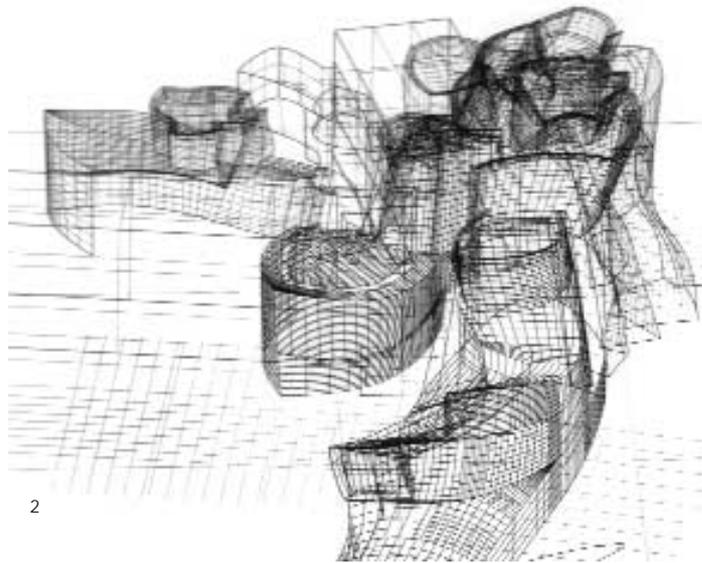
nung Euralille, OMA), sie werden unmittelbar in Architektur übertragen (Functionmixer, MVRDV), oder sie werden durch Architektur veranschaulicht (MEGACITY, DATATOWN, MVRDV, s. *archithese 3.02*). Das Ergebnis ist ein reizvolles Spiel, resultierend aus einem mehr oder weniger analytischen Umgang mit Zahlen und Diagrammen, deren Präsentation ästhetische Gesichtspunkte zu Grunde gelegt werden. Der Weg von der Analyse zum Gebäude wird vorexerziert, mit ihm wird für das Produkt geworben und dessen zukünftige Nutzungen werden beschrieben. Raum und Excelsheets rücken näher zusammen, überschneiden oder bedingen sich zunehmend. Die Entwürfe zeigen ihre funktionalen und relationalen Zusammenhänge.

*Mathematische, physikalische oder biologische Experimente für die Formfindung:* Durch die rasante Entwicklung der Computer (nach dem Moor'schen Gesetz verdoppelt sich deren Geschwindigkeit alle 18 Monate!) und der CAD-Systeme entstanden neue Möglichkeiten zur Generierung von Formen. Musste Frei Otto für den Entwurf der Überdachung des Münchener Olympiastadions noch Experimente mit Seifenblasen durchführen, um wissenschaftliche Erkenntnisse für die Findung einer guten Form zu erhalten, lassen sich diese Dinge mittlerweile in CAD-Programmen simulieren. Oberflächen hoher Komplexität können spätestens seit der Erfindung der Bezierkurve durch mathematische Zusammenhänge beschrieben werden. Prominenter Vertreter hierfür ist Greg Lynn, der auf CAD-Systemen Formen programmiert oder mit Programmen modifiziert, um diese sehr unmittelbar in computergesteuerter so genannter CNC-Fertigung herzustellen (*archithese 2.02*). Seine nichteuklidischen Geometrien sind auf keinem anderen Weg herstellbar – sie sind generiert.

*Freie Formen werden gerastert und beschrieben:* Ungeachtet der modernen Fertigungsmöglichkeiten werden handwerklich Modelle hergestellt. Der nicht zu unterschätzende Einsatz des Computers liegt in der Strukturierung bisher nicht beschriebener mathematischer Formen: Ein 3-D-Scan des Modells digitalisiert den haptischen Entwurf und macht ihn in einem präzisen Transformationsprozess maschinenlesbar – und so mit computergesteuerten Maschinen (CNC) produzierbar. Der Computer gibt die Möglichkeit, Formen frei von strukturellen und konstruktiven Überlegungen zu entwerfen: Die ange-



1



2

1 Entwerfen nach Zahlen: MVRDV, Metacity Datatown

2 Freie Formen werden gerastert und beschrieben: Frank O. Gehry, Guggenheim Museum Bilbao, Computermodell

wandte Technologie ermöglicht erst die Formenvielfalt und befreit von den Beschränkungen der Rasterung.

### Aktuelle Strömungen in der Informatik

*Neue Programmiersprachen:* Aus den ursprünglichen, prozeduralen Programmiersprachen haben sich objektorientierte Beschreibungsformen entwickelt. Objekte sind informationstechnische Konstrukte, die sowohl Eigenschaften als auch Verhalten (Methoden) umfassen. Einige Programmierungsumgebungen haben grafische Benutzerführungen, in denen die gezeichneten Elemente als Objekte formuliert sind und ihre Methoden in einem Menü anbieten. Ihre Grösse, Farbe oder Position kann zum Beispiel programmiert werden. Ein wesentliches Merkmal von Objekten ist die Möglichkeit der *Typisierung*. Die gemeinsamen Eigenschaften und Verhalten verschiedener Objekte werden in hierarchischen Typenbeschreibungen zusammengefasst und von den Individuen (Instanzen) geerbt. Dabei können die Individuen von ihrem Typ abweichende Eigenschaften und Verhalten durch «Überschreiben» ausprägen. Auf diese Art und Weise ist es relativ leicht, sehr kompakte und dennoch anpassungsfähige Strukturen aufzubauen, die sowohl beschreibend als auch operational genutzt werden können. Diese Art des Programmierens kommt dem architektonischen Entwurf entgegen, da Strukturen im Detail beschrieben werden können, ohne dass ein übergeordnetes Problem bereits endgültig gelöst zu sein braucht.

*Software-immanente Skriptsprachen:* Inzwischen bieten alle guten CAD- und Multimediaprogramme leicht erlernbare und gut dokumentierte Skript- oder Programmiersprachen an. Sie ermöglichen es, Zeichnungselemente mit einer «Intelligenz» oder einem «Verhalten» zu versehen. Während bei professionellen Multimediaprogrammen (z.B. *Macromedia Flash* oder *Macromedia Director*) die User selbstverständlich Grafiken programmieren und auf diese Weise interaktiv machen, wird diese Möglichkeit bei CAD bisher kaum genutzt. In diesem Bereich steht ein weites Feld für neue Fantasien bislang ungenutzt offen.

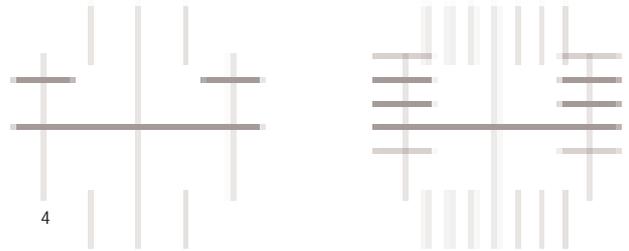
### Programmierte Architektur

Im Spannungsfeld dieser architektonischen und informationstechnischen Phänomene stellt die neue Professur für CAAD an der Eidgenössischen Technischen Hochschule in Zürich unter der Leitung von Ludger Hovestadt eine Produktionskette für Architekten auf und erforscht die einzelnen Elemente sowie deren Zusammenspiel systematisch. Ziel ist es, Architektur mit Informationstechnologie zu beschreiben, zu generieren, zu verwalten, zu bewerten und zu produzieren.

Bei den bisherigen Versuchen, Software zu programmieren, die Architektur einigermassen selbstständig generiert, stiess man auf ein sehr allgemeines Problem von Architektur-entwürfen: die Objektivierbarkeit. Inwiefern können die generierten Lösungen objektiv beurteilt werden? Fragen funk-



3



4

tionaler Natur, beispielsweise nach möglichst kurzen Wegen innerhalb eines Gebäudes, können zwar noch gut analysiert werden; nicht aber, ob die Fenster einer Fassade quadratisch oder rechteckig sein sollen. Das Problem, welches sich üblicherweise beim Bewerten von Architektur ergibt, ist, dass es für Entscheidungen häufig kein eindeutiges «richtig» oder «falsch» gibt, sondern nur «gefällt» oder «gefällt nicht»: Die Bewertung kann nicht immer auf objektive Kriterien gestützt werden und ist oft vom subjektiven Geschmack geleitet.

Bemühungen, Programme mit Lernalgorithmen zu versehen – wie genetische Algorithmen oder neuronale Netze –, sind bisher noch nicht überzeugend gelungen. Offenbar sind die Zusammenhänge zwischen Aspekten wie Umfeld, Analyse, Idee, Stil, Raumprogramm und Materialwahl zu kompliziert, als dass sie sich zusammenfassen liessen. Und jeder Entwerfer beurteilt Hierarchie und Wichtigkeit der einzelnen Aspekte vollkommen unterschiedlich. Ähnliche Probleme finden sich in der computergenerierten Musik beim Versuch, klassische Komponisten zu simulieren, während neue Kompositionen ohne diese Einschränkungen sehr einfach programmierbar sind. Folgerichtig kann es keine allgemeingültige Software oder universelle Maschine geben, die Architektur entwerfen kann.

Wie also könnte der Computer als ein weiter gehendes Werkzeug – über die elektronische Zeichenmaschine, die Kommunikationszentrale und das Archivierungssystem hinweg – als Entwurfsunterstützung dienen, und welche Auswirkungen hätte sein Einsatz auf die Architektur?

Eines der ersten «Generierungsprogramme» für Architektur beschreibt Jean Nicholas Louis Durand in seinem 1819 erschienenen Buch *Marche à suivre. Précis des Leçons d'Architecture données à l'Ecole Royale Polytechnique*. Durand leitet aus wenigen einfachen und mehrstufigen Regeln die Konstruktion eines detaillierten Gebäudes ab. Er formuliert die Spiegelachsen

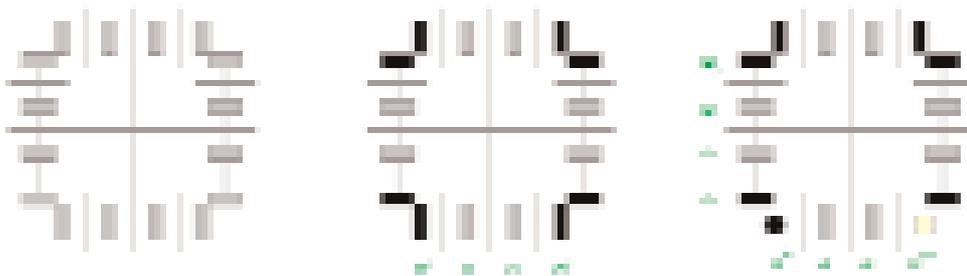
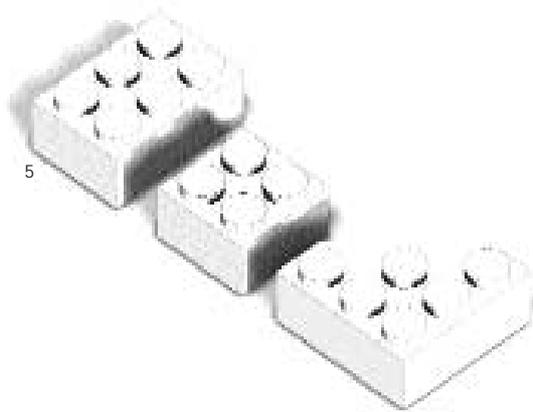
und kleinsten Gemeinsamkeiten in seinem Programm und beschreibt auf diese Weise unmissverständlich die Architektur. Als Ergebnis finden wir einen Plan, bei dem es zum einen nicht beliebig ist, wo welches Element angeordnet ist, und es zum anderen sehr leicht möglich ist, Fehler zu finden. Es handelt sich hier nicht um ein Programm, das Architektur generiert, sondern vielmehr um einen programmierten Entwurf – und um eine Beschreibung des Gebäudes, die kompakter nicht sein könnte und Redundanzen vermeidet.

Unter dem Thema «Programmieren statt zeichnen» forscht die Professur für CAAD nach neuen, aus der inneren Struktur heraus entwickelten Formulierungen von Architektur.

#### Parametrisierte Objekte oder: Ausnahmen wie Regeln beschreiben

Ein kleines, aber verständliches Beispiel soll Idee und Produktionskette eines parametrisierten Objektes verdeutlichen. Eine Kleiderleiste wurde in einem Programm beschrieben; durch Eingabe von entsprechenden Parametern wie beispielsweise der Länge oder des Mindestabstandes der einzelnen zufällig erzeugten Einhängen wird eine Instanz – ein individuelles Objekt – erzeugt, das direkt mit einer Laserschneidemaschine produziert werden kann. Ob drei gleiche oder drei verschiedene Objekte hergestellt werden sollen, macht sich im Preis unwesentlich bemerkbar: Die Leiste kostet etwa 100.– CHF pro Laufmeter. Hier handelt es sich um einen programmierten Entwurf, von dem in Sekunden beliebig viele unterschiedliche Instanzen erzeugt werden können.

Die Kenntnisse, die für die Programmierung eines solchen Objektes notwendig sind, kann sich ein erfahrener CAD-User in ca. einer Woche aneignen. Mit ein wenig Übung programmiert man einen solchen Entwurf in einer Stunde. Das heisst: Wenn mehr als drei unterschiedliche Ausprägungen eines Elementes innerhalb eines Entwurfs auftauchen, lohnt sich



3 Parametrisiertes Objekt Kleiderleiste: Entwurf von Christoph Schindler, Professur für CAAD, ETH Zürich, Ansicht und Schnittplan

4 Sowohl Ausnahmen als auch Regeln beschreiben: Funktionsweise, demonstriert an einem abstrakten Beispiel

a vererbt an a';  
a' vererbt an a'' usw.;  
a ist rechteckig und hellgrau;  
a' ist wie a, jedoch dunkelgrau;  
a'' ist wie a', jedoch rund

5 Informationstechnische Prototypen: Auch nichtvordefinierte Parameter können modifiziert werden. Obwohl der rechte Legostein nicht an den vorgegebenen Parametern verändert wurde, ist er vom linken Prototypen erzeugt worden

eine Parametrisierung. Oder noch weiter gedacht: Die Wirtschaftlichkeit einer Architektur ist nicht mehr proportional zu ihrer Regelmässigkeit. Wenn ein Entwurf, der wie die Kleiderleiste aus vielen unterschiedlichen Objekten gleicher Art besteht, auf diese Art und Weise beschrieben und entworfen werden kann, hat das weitreichende Konsequenzen auf die Gestaltung von Architektur.

Die Kleiderleiste versinnbildlicht eine Idee. Beschreibt man jedes Zeichnungselement separat, entspricht die Planung einem handwerklichen Prozess; bei der Verwendung von Symbolen, Kopien oder Zeichnungsblöcken befindet sich die Planung auf dem Entwicklungsstand des industrialisierten Bauens; durch das Bilden von Klassen und Instanzen erreicht sie den Stand programmierter und variabler CAAD-Objekte. Durch die Modifizierung des prozeduralen Programmschemas von Durand in ein objektorientiertes steigert sich die Reichhaltigkeit der Varianz bei annähernd gleicher Datenmenge enorm. Das formulierte Grundelement (Klasse) vererbt die Informationen an seine Instanzen, die wiederum nur die Veränderungen zur Klasse speichern müssen.

### Informationstechnische Prototypen, Analyse und Simulation mit Agenten, Raumlabor

Von einem gezeichneten oder programmierten Prototypen lassen sich modifizierte Ableger erzeugen, die zwar von dem ursprünglichen Original abstammen, jedoch auch eine nicht vordefinierte mögliche Abweichung haben können. Dadurch, dass die Variablen eines Objektes nicht von vornherein bekannt sein müssen, unterscheiden sich gezeichnete oder programmierte Prototypen wesentlich von den parametrisierten Objekten.

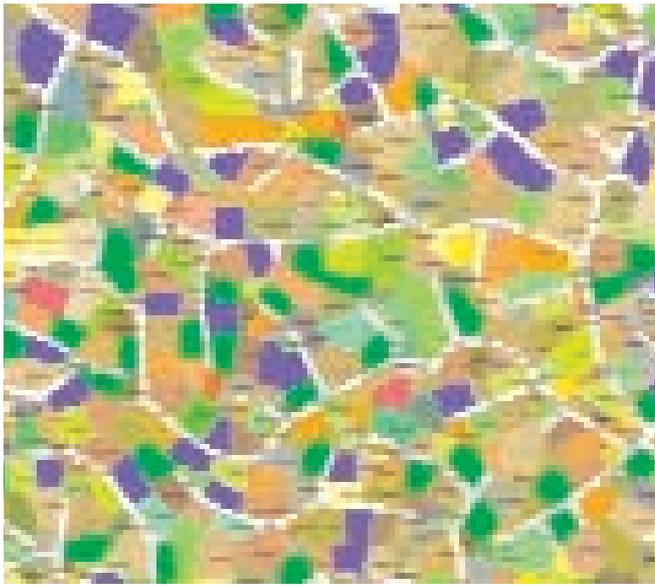
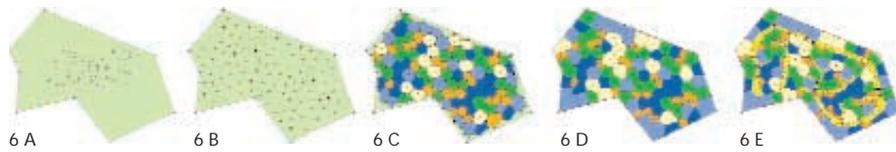
Agenten sind aus informationstechnischer Sicht viele eigenständig kommunizierende «Arbeiter» eines funktionierenden Systems. Jede Ameise innerhalb eines Stammes ist bei-

spielsweise ein Agent. Das Insekt kennt kein gemeinschaftliches Ziel, hat nur eine begrenzte Wahrnehmung und kommuniziert mit den anderen durch die Aussendung von Interferenzen. In dem gemeinschaftlichen Gebilde «Stamm» gibt das einzelne Verhalten durchaus Sinn (zum Beispiel Ameisenstrasse und Ameisenhaufen).

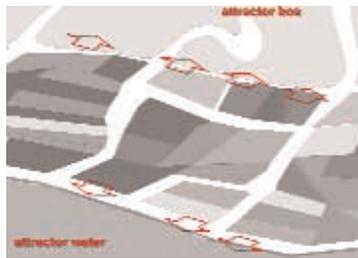
In vergleichbarer Weise können Agenten gestaltet werden, die mit individuellem Verhalten Informationen über einen Entwurf sammeln. Man kann ebenso Agenten definieren, die ein geplantes Gebäude bewohnen oder in ihm arbeiten und es nach ihren «Vorstellungen» bewerten oder sogar zu verändern versuchen. Mit Agentensystemen werden unter anderem auch Verkehrssimulationen und -bewertungen programmiert.

Analyse und Synthese können zu einem iterativen Prozess zusammengeführt werden; das bedeutet, dass die generierten Strukturen bewertet werden, um danach unter korrigierten Parametern neu erzeugt zu werden. Diese Wechselwirkungen bewirken bei den sich permanent regenerierenden Ergebnissen eine zunehmende Optimierung.

Die Professur für CAAD baut in mehreren sich ergänzenden Forschungsprojekten ein Team auf, welches sich mit computergenerierten Strukturen und deren Analyse beschäftigt. Da die bisher erarbeiteten Lösungen weitgehend auf Akzeptanz unter Architekten und Laien gestossen sind, sollen die bestehenden Ansätze erweitert werden. Neben der Grundlagenforschung sollen Praxisnähe erzeugt und die gewonnenen Erkenntnisse schnell in die Realität übertragen werden. Durch die Kombination unterschiedlicher algorithmischer Bausteine sollen Erkenntnisse darüber gewonnen werden, wie ein Raum, ein Gebäude oder eine Siedlung effektiv programmiert werden können. Das Projekt «Raumlabor» basiert auf der Studie «KaisersRot», die im vergangenen Jahr in Kooperation mit dem holländischen Architekturbüro Kees Christiaanse Architects and Planners (KCAP) entwickelt wurde.



7



8



9

**6–7 «KaisersRot»,  
Parzellenplan, Ent-  
stehungsprozess  
und Ergebnis**

Der Plan berücksichtigt die Wünsche der zukünftigen Bewohner bezüglich Orientierung, Nachbarschaft, etc. ihres Grundstücks. Er beruht auf einem dynamischen Prozess, in dessen Verlauf die Grundstücke ihre Form, Position und Grösse untereinander aushandeln, bis ein Gleichgewicht erreicht ist.

- A Bürgerwünsche
- B Verteilung unter Berücksichtigung der individuellen Bedürfnisse
- C Triangulation und Polygonbildung
- D Straffung der Polygone
- E Strassenbildung

**8 «KaisersRot»,  
Einflussnahme  
externer Attrak-  
toren auf die  
Parzellenform**

**9 «KaisersRot»,  
Einflussnahme des  
Bewohners auf  
verschiedene auf  
das Grundstück be-  
zogene Faktoren**

**10 «KaisersRot»,  
Ausstellung am NAI  
Rotterdam, 2001**

Die Software formte und platzierte die Parzellen der Besucher gemäss den Wünschen, die diese in einem Formular notiert hatten. Die Ergebnisse verschiedener Durchläufe sind unterschiedlich, erfüllen aber weitgehend die Vorgaben.

**Ein Anwendungsbeispiel aus dem Planungsbereich:  
«KaisersRot»**

Die Autoren der Studie «KaisersRot» waren bei den ersten Diskussionen und Entwürfen zu diesem Projekt noch in dem Forschungsprojekt «Medienexperimentelles Entwerfen» der Universität Kaiserslautern involviert. Der Name «KaisersRot» entstand aus der Kombination der Namen Kaiserslautern und Rotterdam.

«KaisersRot» reagiert auf die konkrete baupolitische Entwicklung in den Niederlanden. Dort ist im staatlichen VINEX-Programm 1993 festgelegt worden, dass in Holland bis 2005 unter festgelegten Rahmenbedingungen 635 000 neue Wohnungen gebaut werden sollen. Die Vorgaben definieren für die ausgewählten Baugebiete exakt den Prozentsatz und die Grösse der frei stehenden Häuser, der Reihenhäuser, der Freiflächen, der Erschliessungen und so weiter. Eine Konsequenz dieser politisch motivierten Festlegungen ist, dass Stadtplanung in Holland zunehmend zu einer von Generalplanern gesteuerten Prozedur wird – mit dem Ergebnis, dass häufig gleichförmige und anonyme Siedlungen ohne Individualität und Wiedererkennungswert entstehen.

Gegen diese Tendenzen entwickelte das Büro KCAP in der Folge eigene Entwurfsstrategien, die auf die Vorgaben des VINEX-Programms mit unverwechselbaren städtebaulichen Strukturen reagieren. Zu den gesetzten Vorgaben schafft KCAP weitere formale Regeln und räumliche Anforderungen. Dieses Vorgehen beim Entwurf von Parzellen- und Bebau-

ungsplänen lässt eine hohe Komplexität entstehen, die manuell nur sehr aufwändig zu bewältigen wäre. Das Entwerfen wird zu einer Denksportaufgabe, bei der nach dem Verfahren «Trial and Error» Lösungsmöglichkeiten eingekreist werden. Die Software «KaisersRot» automatisiert diesen Prozess nach den von KCAP vorgegebenen Parametern und generiert innerhalb weniger Sekunden eine Lösung, die sich den eingegebenen Anforderungen weitestgehend annähert. Ein vergleichbares Ergebnis würde mit «manuellem Vorgehen» ungefähr 300 Stunden Arbeitszeit erfordern.

**Funktionsweise von «KaisersRot»**

Während beim klassischen städtebaulichen Entwurf *top-down*, also vom grossen Massstab zum kleinen hin, entworfen wird, arbeitet «KaisersRot» *bottom-up*, also vom Detail zum Gesamten.

In einem dynamischen Prozess verhandelt zeitgleich jedes Grundstück mit seiner Umgebung, deren Attraktoren und den anderen Grundstücken seine Position und Grösse aus. Diese Interaktion der einzelnen Elemente, die sich wie verschieden gepolte Magneten anziehen oder abstossen, visualisiert sich am Bildschirm, bis letztendlich die gesamte Struktur in ein Gleichgewicht kommt. Innerhalb dieses Prozesses werden die «Bedürfnisse» jedes einzelnen Grundstücks gleichberechtigt behandelt – die entstehende Struktur kann deshalb als demokratisch bezeichnet werden. Danach findet



10

eine Kristallisierung statt, die die Topologie dieses «Entwurfes» festlegt. In den folgenden Schritten werden Erschliessungen eingefügt sowie die Formen der Grundstücke und Strassen aufeinander abgestimmt. Das sich einstellende Ergebnis ist als Ausgleich der eingegebenen Faktoren zu verstehen und zeigt sich als eine natürlich anmutende Struktur. Sie entsteht ausschliesslich aus den Wechselwirkungen der Strukturelemente und ist ohne zusätzliche formale Hilfestellungen entstanden. Durch die Möglichkeit, eine grosse Anzahl unterschiedlicher Variationen mit verschiedenen Parametern zu erzeugen, kann sich der Entwerfer eine aussuchen und manuell weiter bearbeiten.

«KaisersRot» ist also keine Entwurfsmaschine, sondern eine Software, die Architekten in ihrer Entwurfsmethodik und bei einer speziellen Aufgabe unterstützt.

### Ein Bürgerbeteiligungsmodell?

Mit «KaisersRot» ist eine völlig neue Form der Beteiligung der Bewohner am Entwurfsprozess möglich. In der Ausstellung «heilige huisjes» simulierte das «KaisersRot»-Team im Netherlands Architecture Institute in Rotterdam am 15. und 16. September 2001 diese Art von Bürgerbeteiligung. Die Besucher konnten ihre Vorstellungen zum Wunschgrundstück, zu den Lieblingsnachbarn, die Nähe zu Wald oder Wasser usw. in ein Formular eintragen. Wenige Momente später platzierte die Software das neue, individuell gewählte Grundstück an einer möglichst günstigen Stelle. Lässt man den Prozess mehrmals

laufen, sind die Ergebnisse jeweils unterschiedlich, erfüllen aber dennoch zu einem hohen Prozentsatz die erklärten Bedürfnisse. Diese Ansätze will KCAP in die Realität umsetzen – das Büro ist momentan auf der Suche nach einer Gemeinde, die ein grösseres Areal für diese Ansätze zur Verfügung stellt.

Am Beispiel «KaisersRot» wird sehr deutlich, welche Chancen für die Architektur im gezielten Einsatz von Informationstechnologie stecken. Der Computer vollführt keinen kreativen Prozess – dieser fand beim Programmieren des Entwurfes statt –, sondern er organisiert komplexe Zusammenhänge, die helfen können, Szenarien aufzustellen und zu überprüfen, die manuell nicht möglich gewesen wären.

Weitere Informationen unter  
<http://www.caad.arch.ethz.ch> und  
<http://kaisersrot.com>

Oliver Fritz ist Architekt und wissenschaftlicher Mitarbeiter an der Eidgenössischen Technischen Hochschule Zürich, Lehrstuhl CAAD, Ludger Hovestadt.