

Spatial Analysis and Developing Tool - a new software for landscape analysis

R. Wieland ^{*}, G.A. Korn [†], X. Holtmann [‡]

Abstract

Es wird eine Software vorgestellt, die eine räumliche Simulationsumgebung (SAMT) mit einer dynamischen Simulationsumgebung (DESIRE) kombiniert. SAMT/DESIRE zielt auf Anwendungen in denen räumliche Daten mit dynamischen Simulationen verbunden werden müssen. DESIRE erlaubt das schnelle Prototyping dynamischer Problemstellungen am Punkt. SAMT liefert die Daten an den zu untersuchenden Raumpunkten. Das erlaubt die Lösung eindimensionaler Prozesse an n-Punkten im Raum, wie sie z.B. bei der Agroökosystemmodellierung oder der Modellierung von linienhaften Fließgewässern vorkommen.

1 Einführung

Bisherige Simulationssysteme wie Matlab [1] oder das freie Tool scilab [2] sind darauf spezialisiert technische Modellierungsprobleme zu lösen. In einem klar definiertem technischen System, wie einer chemischen Anlage, gibt es z.B. Rohrleitungen die Strömungsverhältnisse festlegen. Innhalb dieser Simulationsumgebungen wurden eine Reihe von Modellen entwickelt, die auch für Fragen der Umweltsimulationen interessant erscheinen [3]. Daraus ergab sich, dass viele Problemstellungen der Umweltsimulation in Matlab implementiert wurden [4]. Nachteilig bei diesen Lösungen sind die notwendigen Konvertierungen vom Geographischen Informationssystem (GIS) nach Matlab und zurück zum GIS. Weiterhin nachteilig ist, dass Matlab für die Fuzzy-Modellierung interpretativ arbeitet und damit sehr langsam bei großen Problemen (mehrere 10.000 bis 10.000.000 Rechnungen) ist. Aus diesem und dem Grund eine erweiterbare Simulationsplattform als Open Source zu besitzen, entstand SAMT [5]. SAMT konnte einen großen Teil der räumlichen Simulation, einschließlich der Fuzzy-Modellierung und der Neuronalen Netze, übernehmen. Für die dynamischen Problemstellungen wurden C++ Modelle geschrieben, die unterstützt durch die Gnu Scientific Library (GSL) [6] sehr leistungsfähig, aber schwer zu programmieren waren. Die Implementierung der Modelle setzte einen Informatiker voraus. Dieser Nachteil soll nun durch die Integration des dynamischen Simulationssystem DESIRE [7] in SAMT überwunden werden.

^{*}Leibniz-Centre for Agricultural Landscape and Land Use Research (ZALF) e. V email: rwieland@zalf.de

[†]G. A. AND T. M. KORN INDUSTRIAL CONSULTANTS, email: gatmkorn@aol.com

[‡]ZALF email: holtmann@zalf.de

2 Spatial Analysis and Modeling Tool

Bevor DESIRE vorgestellt wird, sollen hier die Eigenschaften von SAMT [11] zusammengefaßt werden:

- SAMT ist ein einfaches gridbasiertes GIS, mit Schnittstellen zu ARCGIS und zum Datenbanksystem MySQL
- Die grundlegenden Gridfunktionen, wie cut, mul, add, norm etc. und auch Funktionen zum Verschneiden, Distanzrechnung, Moving Window Technologie etc.
- Integriert wurden: Analysefunktionen (Histogramm, Statistik), Analyse von Transekten (Profile, Semivariogramm, Powerspektrum), 2D-Analysefunktionen (Contouring, Korrelation von Karten), 3D-Analysefunktionen (Splatter-Technik, NURBS)
- Als eigenständige Tools wurden integriert: Fuzzy-Entwicklungstool und Neuronales Netzwerk Entwicklungstool
- Ein Sensitivitätsanalyse Modul kann zur Analyse des Modellverhaltens eingesetzt werden.

2.1 DESIRE

Das von G.A. Korn entwickelte Simulationstool DESIRE (Direct Executing SIMulation in REal Time) [7] wurde als Open Source unter GPL [8] freigegeben. DESIRE hat eine 25 jährige erfolgreiche Entwicklung hinter sich [13]. Dabei hat es sich in einer Reihe von Industrieprojekten bewährt und gilt als ausgereifte Software. Bei einer so langen Geschichte eines Softwaretools ergab sich die Notwendigkeit einer teilweisen Überarbeitung und Anpassung an moderne graphische Nutzerbedingungen. Vor allem aber sollte DESIRE in der SAMT Umgebung laufen und Daten aus dieser übernehmen können. Das machte eine Erweiterung von DESIRE und SAMT notwendig. Bevor auf die Interaktion zwischen SAMT und DESIRE detailliert eingegangen werden soll, auch hier wieder eine kurze Charakteristik von DESIRE:

- DESIRE trennt zwischen einem interpretativ arbeitenden Teil und einem compilierten Teil. Im ersteren werden die Initialisierung, die Simulationssteuerung und die Übergabe von Daten zu und von SAMT realisiert. Im letztern die Simulations selbst und die graphische Ausgabe vorgenommen. Durch die Compilerung des dynamischen Teils erfolgt eine sehr schnelle Realisierung, so dass selbst 40.000 Differentialgleichungen in wenigen Sekunden gelöst werden können.
- DESIRE besitzt eine Vector-Verarbeitung [14] die auf Ideen aus dem Supercomputing stammt und die Formulierung komplexer Problemstellungen vereinfacht und beschleunigt.
- DESIRE wurde mit einer modernen graphischen Oberfläche versehen. Sie enthält einen Editor mit Syntaxhighlighting, eine Modellverwaltung, ein neues Fehlermanagement und vor allem eine leistungsfähige Graphik. Letztere kann zwischen den Simulationen weitergenutzt werden und ermöglicht so einen Vergleich zwischen den Simulationen.

Aus dieser Charakteristik ergeben sich folgende Einsatzgebiete von DESIRE im Zusammenhang mit der räumlichen Simulation:

- Nutzung von DESIRE als Werkzeug zur Entwicklung, Parametrisierung und Testung von Modellen. Diese Modelle können dann in C++ Code überführt und konventionell in SAMT genutzt werden.
- Nutzung von DESIRE als Simulationstool eindimensionaler räumlicher Fragestellungen. SAMT liefert in diesem Einsatzfall Werte für n räumliche Punkte. DESIRE berechnet an diesen Punkten den Prozess zu einem Zeitpunkt t_1 und liefert die Werte zurück zu SAMT. SAMT übernimmt die räumliche Darstellung und eventuell eine räumliche Interpolation. Gegebenenfalls ist diese Rechnung zu anderen Zeitpunkten t_2, t_3, \dots zu wiederholen.
- SAMT liefert Daten aller Gridzellen und DESIRE berechnet die zeitliche Veränderung. Hierbei sind Restriktionen zu beachten. DESIRE kann “nur” maximal 40.000 Differentialgleichungen auf einmal rechnen. Damit muss häufig eine Auswahl von Gridzellen im Vorfeld getroffen werden.
- SAMT berechnet räumlich verteilt unter Nutzung numerischer Näherungsverfahren einen Prozess. An einigen Messpunkten ermittelt DESIRE die Werte nach einem genaueren, aber langsameren Verfahren. Diese Punkte dienen als Referenz für die räumliche Rechnung.

Im weiteren soll ein Verfahren zur Lösung partieller Differentialgleichungen vorgestellt werden, das für eine Reihe von Problemstellungen in der Ökosystemmodellierung als Basis eingesetzt werden kann. Durch den Einsatz von DESIRE kann die Problemstellung prinzipiell untersucht, an die realen Erfordernisse angepasst und gegebenenfalls implementiert werden. Der letzte Schritt ist hier offengelassen, da die technische Realisierung den Rahmen sprengen würde.

3 Lösung partieller Differentialgleichungen mittels “Method of Lines”

Die sogenannte “Method of Lines” ist eine allgemeine Technik zur Lösung partieller Differentialgleichungen [9]. Partielle Differentialgleichungen spielen eine Rolle, wenn Prozesse sowohl von der Zeit als auch vom Raum abhängen. Einfache Beispiele dafür bilden die Wärmeleitungs- und Diffusionsprozesse. Die Aufgabe lässt sich wie folgt darstellen:

$$\frac{\partial u}{\partial t} = a * \frac{\partial^2 u}{\partial x^2} \text{ with } u(t, 0) = u(0)u(t, L) = u(L) \ t > 0 \ u(0, x) = f(x) \ x \in (0, L) \quad (1)$$

Für einen isolierten homogenen Stab der Länge L beschreibt obige Gleichung den Wärmetransport. Die Randbedingungen $u(t, 0)$ und $u(t, x)$ bestimmen die Temperatur am Rand des Stabes, die Anfangsbedingungen bestimmen die Temperatur innerhalb des Stabes zur

Zeit $t = 0$. Gesucht ist eine Temperaturverteilung $u(t, x)$ zu beliebigen Zeiten t innerhalb des Stabes. Fragestellungen wie die Erwärmung von Böden, der Diffusion von Stoffen im Erdreich etc führen auf die gleiche partielle Differentialgleichung. Solche Gleichungen können auf unterschiedliche Weise gelöst werden. Für einfache Beispiele, wie dem obigen lassen sie sich geschlossen lösen. In komplizierteren Fällen bleibt aber nur eine numerische Lösung. Dazu wird der Stab diskretisiert, d.h in k kleine Fraktionen unterteilt.

$$\frac{\partial u_n}{\partial t} = \frac{a}{(\Delta x)^2} * (u_{n+1} - 2 * u_n + u_{n-1}) \text{ for } 2 \leq n \leq L \quad (2)$$

Zu beachten sind die Randbedingungen die im Beispiel als $u_1 = 0$ und $\frac{\partial u_{n+1}}{\partial x} = 0$ angenommen wurden. Randbedingung an der Oberfläche $n = 1$ meint, dass der Stab plötzlich an Eiswasser gehalten wird. Die andere Randbedingung meint anschaulich, dass das Stabende wärmeisoliert wurde. Die erste Randbedingung kann direkt verwendet werden, wogegen die zweite zur Abschätzung der Temperatur an der Stelle $n + 1$ angewandt wird.

$$\frac{\partial u_{n+1}}{\partial t} = \frac{3 * u_{n+1} - 4 * u_n + u_{n-1}}{2 * \Delta x} = 0 \text{ mit } n = k \quad (3)$$

Daraus kann u_{n+1} ermittelt werden. Umgesetzt in ein DESIRE Programm sieht das folgendermaßen aus:

```

DYNAMIC
u[1]=0
u[11]=(4*u[10]-u[9])/3
d/dt u[2]=alpha/(DX*DX)*(u[3]-2*u[2]+u[1])
d/dt u[3]=alpha/(DX*DX)*(u[4]-2*u[3]+u[2])
d/dt u[4]=alpha/(DX*DX)*(u[5]-2*u[4]+u[3])
d/dt u[5]=alpha/(DX*DX)*(u[6]-2*u[5]+u[4])
d/dt u[6]=alpha/(DX*DX)*(u[7]-2*u[6]+u[5])
d/dt u[7]=alpha/(DX*DX)*(u[8]-2*u[7]+u[6])
d/dt u[8]=alpha/(DX*DX)*(u[9]-2*u[8]+u[7])
d/dt u[9]=alpha/(DX*DX)*(u[10]-2*u[9]+u[8])
d/dt u[10]=alpha/(DX*DX)*(u[11]-2*u[10]+u[9])
dispt u[1],u[3],u[5],u[7],u[9],u[10]

```

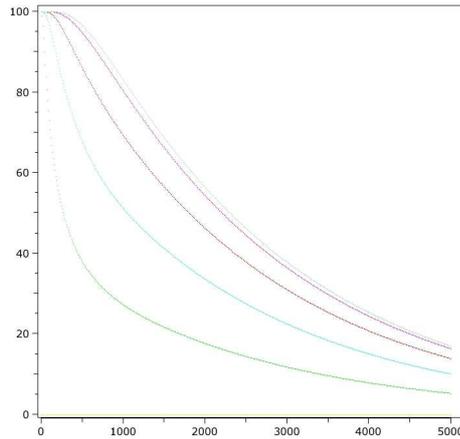


Figure 1: Simulation der Abkühlung

Obwohl das Programm anstandslos läuft ist es nicht gerade elegant. Es müssen alle Gleichungen einzeln notiert werden. Wesentlich besser kann das mittels des Vektorcompilers von DESIRE erledigt werden.

Im Programm sieht das wie folgt aus:

```

u[1]=0
u[n+1]=(4*u[n]-u[n-1])/3
Vectr d/dt u=[vv] alpha/(DX*DX)*(u{1}-2*u{0}+u{-1})
dispt u[1],u[3],u[5],u[7],u[9],u[10]

```

Diese Art der Formulierung von Problemen ist elegant und vor allem hoher räumlicher Diskretisierung vorteilhaft. ¹ Eine Verallgemeinerung der diskreten räumlichen Ableitung als Operator kann die Nutzung der “Method of Lines” in DESIRE weiter vereinfachen [9].

Bevor praktische Problem mit dieser Methode in Angriff genommen werden können, ist eine wichtige Frage nach dem Verhältnis der räumlichen zur zeitlichen Auflösung zu klären. Wie bei allen numerischen Verfahren kommt es bei ungeeigneter Wahl der Diskretisierung zu numerischen Instabilitäten, die das Ergebnis oft unbrauchbar machen. Aus der Literatur [10] konnte folgende Stabilitätsbedingung für die Wärmeleitgleichung gefunden werden:

$$\frac{\Delta t}{(\Delta x)^2} = \frac{1}{2} \quad (4)$$

Das bedeutet in praktischen Lösungen sehr kleine Zeitschritte. Oft wird deshalb versucht das hier vorgestellte explizite Verfahren durch ein implizites Verfahren, das für alle Diskretisierungen numerisch stabil arbeitet zu ersetzen. Nachteilig bei den impliziten Verfahren ist, ds ein Gleichungssystem gelöst werden muss. Auch bedeuten numerische Stabilität nicht automatisch eine gute Approximation der Lösung an die Wirklichkeit. ²

¹[vv] dient als Maskierungsoperator um unnötige Berechnungen der Gleichung für $i=1$ und $i=n+1$ zu vermeiden

²Die Wärmeleitgleichung kann geschlossen gelöst werden, so dass sie als Vergleich zur Güte des numerischen Verfahrens herangezogen werden kann.

4 Beispiel von DESIRE zur Lösung einer partiellen Differentialgleichung mittels der “Method of Lines”

Um ein Beispiel zu zeigen, wie DESIRE zur Lösung eines eindimensionalen Problems verwendet werden kann, wurde eine einfache Fragestellung genutzt. Es soll untersucht werden, wie sich die Bodentemperatur in einer Tiefe von 0.1m bis 1m während einer Hitzeperiode im Sommer verhält. Eine solche Fragestellung tritt z.B. im Zusammenhang mit Umsetzungsprozessen im Boden auf, die gewöhnlich temperaturabhängig sind. Im Beispiel wird nur das prinzipielle Vorgehen beschrieben um die Wirksamkeit von SAMT/DESIRE zu zeigen.³ Die Oberflächentemperatur des Bodens wird in der Simulation als Input verwendet. Die Bodendichte und die Bodenfeuchte kommen aus dem räumlichen Modell, also SAMT. Eine Rückkopplung der Temperatur auf die Bodenfeuchte wird nicht beachtet. Die folgende partielle Differentialgleichung beschreibt das Übergangsverhalten der Temperatur in einer Dimension (Bodensäule)

$$\frac{\partial T}{\partial t} = a(z, t) * \frac{\partial^2 T}{\partial z^2} \quad (5)$$

Worin $a(z, t)$ die Temperaturleitfähigkeit; T die Temperatur; z die Tiefe bedeuten.

$$a(z, t) = \frac{\lambda(z, t)}{c(z, t) * d(z, t)} \quad (6)$$

mit $\lambda(z, t)$ ist die thermische Leitfähigkeit des Bodens der Tiefe z und $c(z, t)$ ist die Wärmekapazität; $d(z, t)$ meint die Dichte des Bodens. Die thermische Leitfähigkeit kann über die folgende empirische Gleichung

$$\lambda(z, t) = \frac{(3 * rd - 1.7) * 1.0E - 03}{1 + (11.5 - 5 * rd) * \exp(-50 * (\theta(z, t)/rd)^{1.5})} \quad (7)$$

Mit rd als Dichte des trockenen Bodens; $\theta(z, t)$ ist die Feuchtigkeit in einer Tiefe z . Die Wärmeleitfähigkeit wird als gewichtete Summe der Leitfähigkeit des trockenen Bodens und der Leitfähigkeit des Wassers berechnet:

$$c(z, t) = r_s(z) * rd * c_s(z) + r_w(z) * d_w * c_w \quad (8)$$

Mit $r_s(z)$ als volumetrischer Teil des Bodens; $r_w(z)$ als volumetrischer Teil des Wasser; d_w der Dichte des Wassers; $c_s(z)$ als Wärmekapazität des trocken Bodens der Tiefe z , c_w und der Wärmekapazität von Wasser. In Abbildung 2, wird die Temperaturleitfähigkeit a in Abhängigkeit der Feuchte und der Dichte berechnet.

³Für eine wirklichkeitsnahe Simulation müßte: die Bodenoberflächentemperatur, die Strahlung, der Wind und der Einfluß der Pflanzendecke beachtet werden. Außerdem hängt die Temperaturleitfähigkeit im Boden von der Bodentextur, der Bodendichte und der Bodenfeuchtigkeit ab.

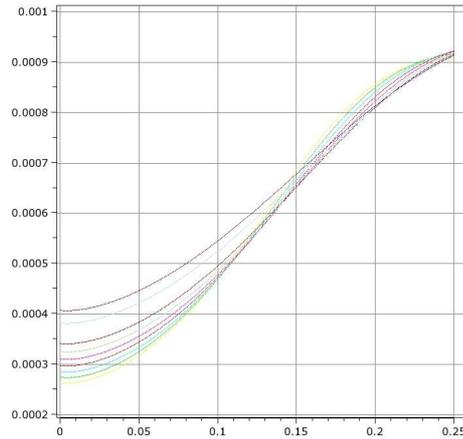


Figure 2: x zeigt die Feuchte in mm , y ist der Wert in m^2/s ; die unterschiedlichen Kurven präsentieren eine Dichte im Bereich von $[1.26kg/l..1.80kg/l]$

Um die Varianz zu bestimmen, die zwischen trockenen und feuchten Boden besteht, wurde für beide Extrema eine Simulation durchgeführt, deren Werte in Tabelle 1 aufgelistet sind.

| soil deep in cm | 0 | 40 | 80 | 120 | 160 | 200 |
|-----------------|--------|--------|---------|--------|---------|--------|
| wet in m^2/s | 0.0005 | 0.0006 | 0.00075 | 0.0008 | 0.00085 | 0.0009 |
| dry in m^2/s | 0.0003 | 0.0004 | 0.0006 | 0.0008 | 0.00085 | 0.0009 |

Table 1: Simulation Parameter

Diese Tabelle bildet die Grundlage für die Simulation unter Nutzung der "Method of Line" zur Lösung der partiellen Differentialgleichung. Durchgeführt wurde die Simulation für eine Hitzeperiode im Sommer (vom 20.7.2001 bis 30.7.2001). Für diesen Zeitraum lagen uns Bodentemperaturwerte und Oberflächentemperatur vor aus dem Versuchsfeld Müncheberg vor. Für die Simulation wurde eine Bodentemperatur von $UA = 25^\circ C$ am Tag und von $UB = 17.5^\circ C$ in der Nacht gewählt. Als Randbedingung wurde eingesetzt:

$$u_0 = (UA - UB) * 0.5 * (1 + \sin(1/24 * 3.141 * 2 * t)) + UB ; u_n = 15^\circ C \quad (9)$$

Die gemessenen Bodentemperaturen in Tabelle 2 wurden als Anfangswerte genutzt. Um auch für Tiefen Werte verwenden zu können, wo keine Messfühler lagen, wurde eine quadratische Interpolation verwendet:

| | | | | | |
|-----------------|------|------|------|------|------|
| soil deep in cm | 100 | 50 | 20 | 10 | 5 |
| T in $^\circ C$ | 17.9 | 19.5 | 20.7 | 21.5 | 21.8 |

Table 2: Gemessene Temperatur

$$T(z) = 2.1816E - 04 * z^2 - 0.0629 * z + 22.02 \text{ with } r^2 = 0.996 \quad (10)$$

Die Interpolationsfunktion wurde bis in eine Tiefe von 1.5m genutzt. Für die Tiefe von 1.5m bis 2m wurde von einer konstanten Temperatur von $17.5^\circ C$ ausgegangen. Die Simulation wurde für den stabilen Zustand, also keinerlei Temperatursprung durchgeführt. In diesem Fall war der Einfluß der Bodenfeuchte vernachlässigbar. Sichtbar wird dieser Einfluß, wenn ein Temperatursprung von $25^\circ C$ to $30^\circ C$ angesetzt wurde. Die Ergebnisse können in Abbildung 3 verglichen werden.

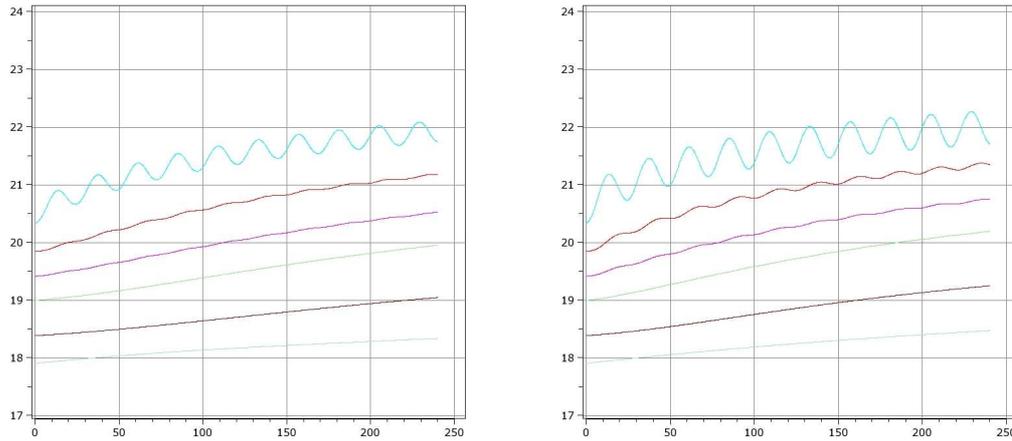


Figure 3: Simulations Ergebnisse des Wärmetransports; 30 cm, 40 cm, 50 cm, 60 cm, 80 cm 100 cm Tiefe über eine Zeitspanne von 10 Tagen

Der Einfluß des Tagesganges der Temperatur ist in 30 cm Tiefe noch gut sichtbar. Der Einfluß unterschiedlicher Bodenfeuchten ist nicht sehr hoch (das ändert sich aber, wenn durch Staunässen ein Wärmeaustausch einsetzen würde). In Beispiel kann daher der Einfluß der unterschiedlichen Bodenfeuchte vernachlässigt werden. Für bestimmte Anwendungen, wo sandige Böden dominieren, kann das zu einer signifikanten Vereinfachung der Modellrechnungen führen. In einer anspruchsvolleren Modellierung müssen die Bodenfeuchte durch den Wasserentzug durch Pflanzen und Abschattungseffekte durch diese beachtet werden. Hier ging es nur darum zu zeigen, wie die prinzipielle Herangehensweise mit DESIRE möglich ist. Ausgehend von diesem einfachen Ansatz können schrittweise komplizierte Einflüsse mit berücksichtigt werden. DESIRE kann helfen, wichtige Einflüsse von weniger bedeutsamen zu trennen. Ist ein befriedigendes Modell gefunden, so kann es mittels der Kopplung zwischen SAMT und DESIRE auch im räumlichen Kontext genutzt werden. SAMT liefert die räumlichen Daten, DESIRE führt die Rechnung aus und legt die Daten in einer für SAMT zugänglichen Form ab. Getestet wurden als Übergabe, ein einfaches Schreiben in ein binäres File, aber auch die Interaktion über shared Memory und die Synchronisation mittels Semaphore.

5 Zusammenfassung

Es konnte gezeigt werden, dass die Kopplung eines räumlichen Simulationsmodells mit einem dynamischen Simulationstool nützlich ist. Beide ergänzen sich, liefern eine Synergie bei der Modellentwicklung und Parametrisierung und können durch den direkten Datenaustausch in dynamischen Fragestellungen im Raum eingesetzt werden. Die hohe Rechenleistung von DESIRE bietet sich an um auch anspruchsvolle Probleme zu lösen. Randbedingung für den Einsatz von DESIRE war immer, dass die Probleme ohne räumliche Kopplung vorlagen, also im wesentlichen auf eine Dimension beschränkt sind. Das trifft die Klasse der traditionellen Agroökosystemmodelle, aber auch die Modellierung von Abbau und Auswaschungsprozessen in Fließgewässern.

Bei Problemstellungen, die eine Interaktion lateral zu den Gridzellen beinhalten, kann DESIRE nicht mehr direkt eingesetzt werden. Aber auch hier kann ein dynamisches Simulationssystem für die Untersuchung der Modellmethodik, Parametrisierung und Optimierung der Simulation wertvolle Hilfe leisten.

References

- [1] <http://www.mathworks.com/>
- [2] <http://www.scilab.org/>
- [3] <http://www.asim-gi.org/>
- [4] Gnauck, A.: "Modellentscheidung oder Entscheidungsmodelle" in: A. Gnauck "Theorie und Modellierung von Ökosystemen" Workshop Koelpinsee 2001; Shaker Verlag
- [5] <http://www.samt-lsa.org>
- [6] <http://www.gnu.org/software/gsl>
- [7] <http://members.aol.com/gatmkorn/>
- [8] <http://www.gnu.org/copyleft/gpl.html>
- [9] Schiesser, W. E.: "The numerical Method of Lines", San Diego, CA: Academic Press, 1991
- [10] Tveito, A.; Winther, R.: "Einführung in partielle Differentialgleichungen"; Springer 1998
- [11] Wieland, R.; Voss, M.: "Spatial Analysis and Modeling Tool - structure possibilities and applications" in IASTED: "Environmental Modeling and Simulation", Nov. 2004
- [12] Korn, G.A.: "Interactive Dynamicsystem Simulation", Gordon and Breach, London, 1998.
- [13] Korn, G.A.: "A Simulation-Model Compiler For All Seasons", Simulation Practice and Theory, 9:21-25, 2001.

- [14] Korn, G.A.: "Model-replication Techniques for Parameter-influence Studies and Monte Carlo Simulation with Random Parameters", *Math. and Computers in Simulation*, 67/6, 501-513, 2004.
- [15] Korn, G.A.: "Fast Monte Carlo Simulation of Noisy Dynamic Systems on Small Digital Computers", *Simulation News Europe*, 2002.