

Lehrstuhl für Systemsimulation

Prof. Dr. Christoph Pflaum¹

¹Department Informatik
Universität Erlangen-Nürnberg



Outline

- 1 Forschung
- 2 Anwendungen
 - Strömungsmechanik
 - Optik, Photonik
- 3 Softwareentwicklung
- 4 Hochleistungsrechnen
- 5 Kenntnisse
- 6 Lehre
- 7 Personen

Anwendungen

- Strömungsmechanik
- Optik (Laser-, Solarzellensimulation)

Softwareentwicklung

- Expression Templates
- Bibliotheken zur numerischen Simulation

Numerische Methoden

- Iterative Löser, Mehrgitterverfahren

Hochleistungsrechnen

- Spezielle Hardware (Graphikkarten, Cell-Prozessor)
- Paralleles Rechnen auf Höchstleistungsrechnern

Strömungsmechanik

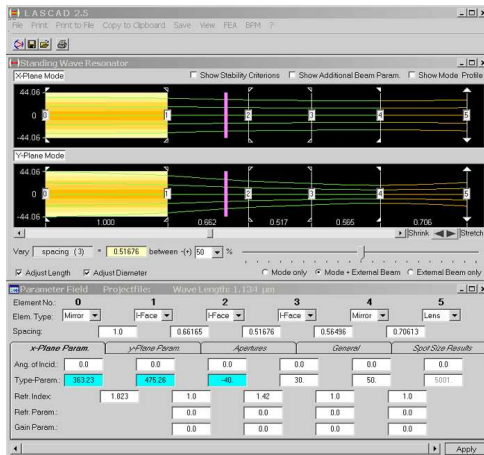
Lattice Boltzmann

- Blasenentstehung in porösen Medien
- Simulation um die Blasenbildung zu verstehen und zu optimieren



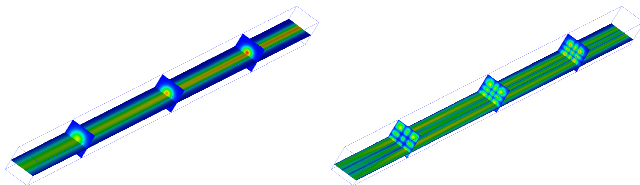
Simulation von Dioden gepumpten Festkörperlaserern

LASCAD:

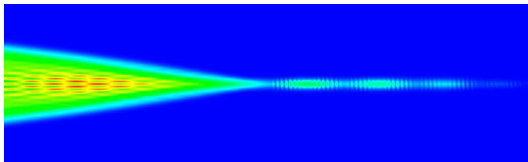


Simulation von Laser Resonatoren

- Festkörperlaser: Zwei-Wellen-Ansatz



- Halbleiterlaser: Neue Finite Elemente



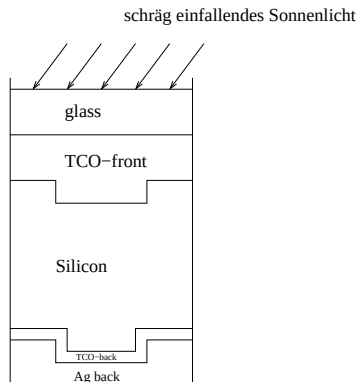
Simulation von optischen Wellen in Dünnschichtsolarzellen

Dünnschichtsolarzellen sind eine neue Generation von Solarzellen.

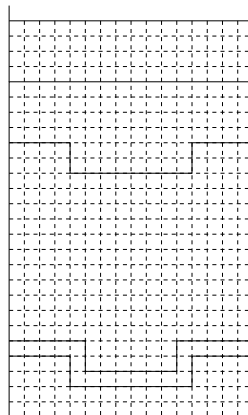
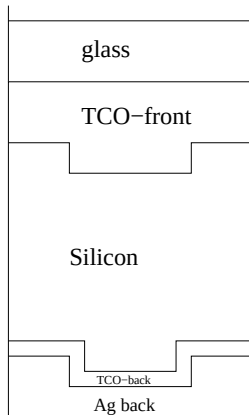
- Deren Dicke ist nur ein paar Mikrometer.
- Geringe Herstellungskosten.
- Aber: Die Effizienz von Dünnschichtsolarzellen ist geringer als die von klassischen Solarzellen.

Simulation der optischen Welle in Dünnschichtsolarzellen

- Das Licht koppelt besser in die Solarzellen ein, wenn die Schichten eine bestimmte Rauigkeit besitzen.
- Was ist die optimale Rauigkeit?
- Lösung der Maxwell'schen Gleichungen!



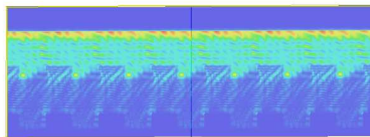
Diskretisierungsgitter



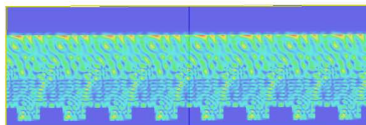
Simulationsergebnisse

Einfallswinkel: 60°

Wellenlänge: $\lambda = 505nm$:

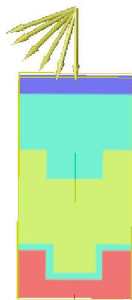


Wellenlänge: $\lambda = 700nm$:

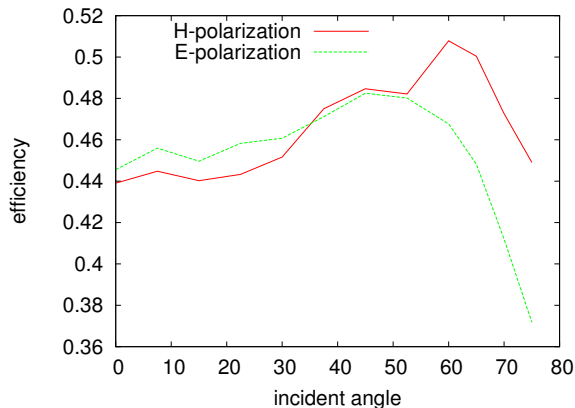


Optische Effizienz: Rechteck-Struktur

Modell der
Solarzelle:

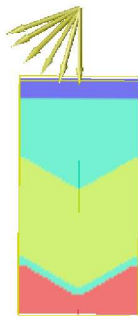


Effizienz:

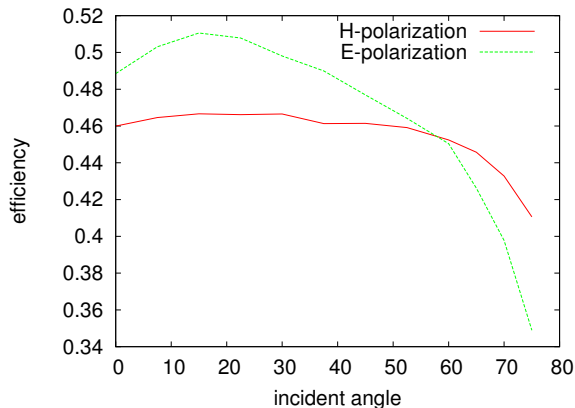


Optische Effizienz: Motten-Struktur

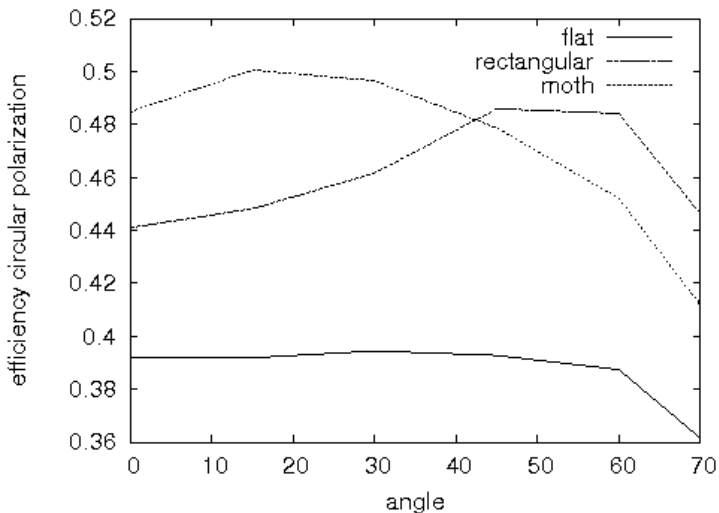
Modell der
Solarzelle:



Effizienz:



Optische Effizienz



Probleme bei der Simulation optischer Wellen

- Hohe Rechenzeit: ein Winkel, 160 Prozessoren, 4 Stunden
- .
- Software muss weiterentwickelbar sein.

FDTD zur Lösung der Maxwell'schen Gleichungen

$$E_x|_M^{t+\frac{\tau}{2}} = E_x|_M^{t-\frac{\tau}{2}} + \frac{\tau}{\epsilon} \left(\frac{H_z|_N^t - H_z|_S^t}{h_y} - \frac{H_y|_T^t - H_y|_D^t}{h_z} \right)$$

Wie implementiert man solche Formeln einfach und sehr effizient?

Überladen von Operatoren

Betrachte die Vektor Klasse:

```
class vector {  
    public:  
        vector(int l);  
        double operator[](int i) { return p[i]; }  
        ...  
    private:  
        int length;  
        double *p;  
};
```

Wie kann man den Operator

vector operator+(vector a, vector b)
effizient implementieren?

Expression Templates

Klassische Implementierung des + Operators:

```
template<class A, class B>
DExpr<DExprSum<Expr<A>, Expr<B> > >
operator+(const Expr<A>& a,const Expr<B>& b) {
    typedef ExprSum<Expr<A>, Expr<B> > ExprT;
    return Expr<ExprT>(ExprT(a,b));
}
```

FDTD zur Lösung der Maxwell'schen Gleichungen

Die Gleichung

$$E_x|_M^{t+\frac{\tau}{2}} = E_x|_M^{t-\frac{\tau}{2}} + \frac{\tau}{\epsilon} \left(\frac{H_z|_N^t - H_z|_S^t}{h_y} - \frac{H_y|_T^t - H_y|_D^t}{h_z} \right)$$

wird wie folgt implementiert:

```
Ex = Ex + tau / epsilon
( ( N(Hz) - S(Hz) ) / hy - ( T(Hy) - D(Hy) ) / hz );
```

Hochleistungsrechnen

- Spezielle Hardware (Graphikkarten, Cell-Prozessor)
- Paralleles Rechnen auf Höchstleistungsrechnern

Wie kann man Bibliotheken entwickeln, die effizientes Rechnen auf Hochleistungsrechnern oder spezieller Hardware erleichtern?

- Interesse in vielen Disziplinen zu arbeiten.
 - Informatik
 - Ingenieurwissenschaft
 - Mathematik
 - Physik
 - Medizin
- Man sollte mit der Mathematik nicht auf Kriegsfuß leben.

Englische Vorlesung, 7.5 ECTS, 2 SWS Vorlesung,
2 SWS Seminar, 2 SWS Übungen (Programmieraufgaben)

Simulation und wissenschaftliches Rechnen I

- performance optimization
- Finite Difference discretization
- basic solvers
- software development
- parallelization

Simulation und wissenschaftliches Rechnen II

- multigrid method
- Finite Elements
- type of grids, visualization and interpolation
- linear elasticity
- computational fluid dynamics and Maxwell's equations

Grundvorlesung

- Simulation und wissenschaftliches Rechnen I
- Simulation und wissenschaftliches Rechnen II

Spezielle Vorlesungen

- Numerische Simulation von Strömungen
- Simulation in der Optik (Computational Optics)
- Iterative Löser (Mehrgitterverfahren)
- Special Topics in Scientific Computing: Numerical Computation of Electromagnetic Fields
- Funktionalanalysis für Ingenieure
- Vorlesung, Seminar über spezielle Hardware: Playstation 3 Programming

Professoren

- Uli Rüde
- Christoph Pflaum

Mitarbeiter

- Stefan Donath
- Christian Feichtinger
- Tobias Gradl
- Jan Götz
- Klaus Iglberger
- Harald Köstler
- Cherif Mihoubi
- Markus Stürmer

Mitarbeiter

- Daniel Ritter
- Tobias Preclik
- Britta Heubeck
- Chrstine Jandl
- Johannes Werner
- Kai Hertel
- Matthias Wohlmuth
- Wolfgang Degen