Experiences with Algebraic Multigrid for a 2D and 3D Biological Respiration-Diffusion Model

Dominik Smits, Stefan Vandewalle

Katholieke Universiteit Leuven, Dept Computerscience, Belgium

Nico Scheerlinck, Bart Nicolaï

Katholieke Universiteit Leuven, PostHarvest Lab., Belgium

1. **Introduction**
   - Motivation
2. **Mathematical model**
   - Respiration-diffusion
3. **Numerical simulation procedure**
   - Finite element method
4. **Our AMG experience**
   - Steady-state and time-dependent
5. **Concluding remarks**

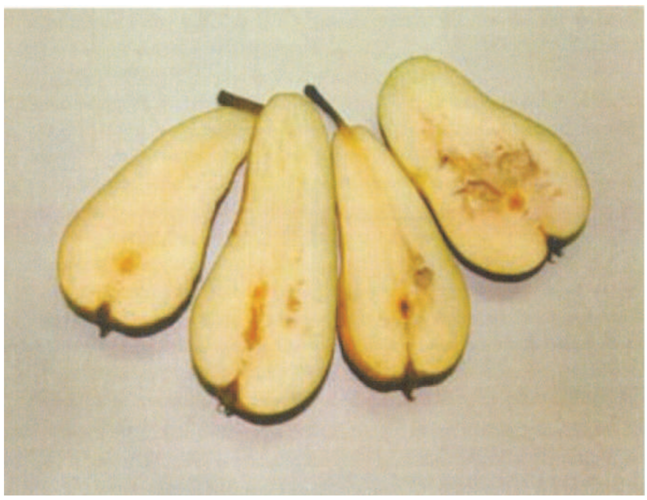
1. INTRODUCTION

- collaboration with Lab. of PostHarvest Technol.

- **their mission:** to develop postharvest storage techniques, in order to extend storage life of harvested fruit (great economical importance for Belgium).

  e.g. control of atmospheric storage conditions (micro atmosphere and cooling rooms)

- **Conference pear** ("brown and hollow" disease)



- **needed:** math model for respiratory activity of fruit

- **our goal:** enable fast simulation through use of advanced multilevel solvers

2. MATHEMATICAL MODEL

- respiration-diffusion model
- O_2 and CO_2 metabolism in the pear
- two coupled non-linear partial differential equations

$$\frac{\partial C_{O_2}}{\partial t} = \nabla \cdot (D_{O_2} \nabla C_{O_2}) - V_{O_2}$$
$$\frac{\partial C_{CO_2}}{\partial t} = \nabla \cdot (D_{CO_2} \nabla C_{CO_2}) + V_{CO_2}$$

with reaction terms

$$V_{O_2} = \frac{V_{max} C_{O_2}}{(K_m + C_{O_2})(1 + \frac{C_{CO_2}}{K_{mCO_2}})}$$
$$V_{CO_2} = RQ_{OX} V_{O_2} + \frac{V_{mfCO_2}}{(1 + \frac{C_{O_2}}{K_{mfO_2}})}$$

- mixed type of boundary conditions

$$D_{O_2} \frac{\partial C_{O_2}}{\partial n} = h_{O_2}(C_{O_2}^{\infty} - C_{O_2})$$
$$D_{CO_2} \frac{\partial C_{CO_2}}{\partial n} = h_{CO_2}(C_{CO_2}^{\infty} - C_{CO_2})$$

Future models

- **extend** with temperature, ethanol, vitamine C, more advanced metabolic equations

- couple with **fermentation model** for brown region and model for cavity growth

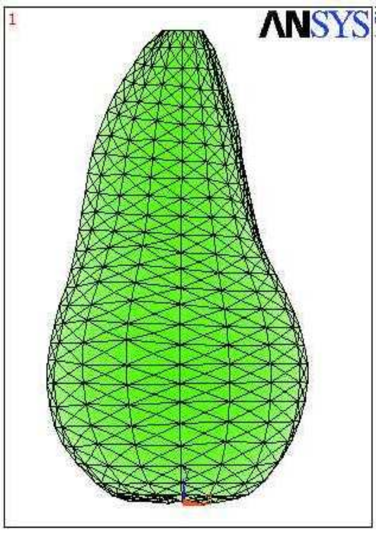
- inclusion of **stochastic aspects**

  "given the uncertainty of the model parameters, what is the uncertainty on the model solution?"

  e.g. Monte Carlo, perturbation, variance propagation, stochastic expansion

3. NUMERICAL SIMULATION PROCEDURE

- geometry scan
- 2D triangular / 3D tetrahedral mesh generation (ANSYS, GiD)



- discretisation with linear finite elements

$$\frac{\partial u}{\partial t} - Lu = f(u) \Rightarrow C\frac{du}{dt} + Ku = f(u)$$

- time-discretisation: BDF1, BDF2 / IRK
- modified Newton linearization
- sparse Gauss elimination (LU)

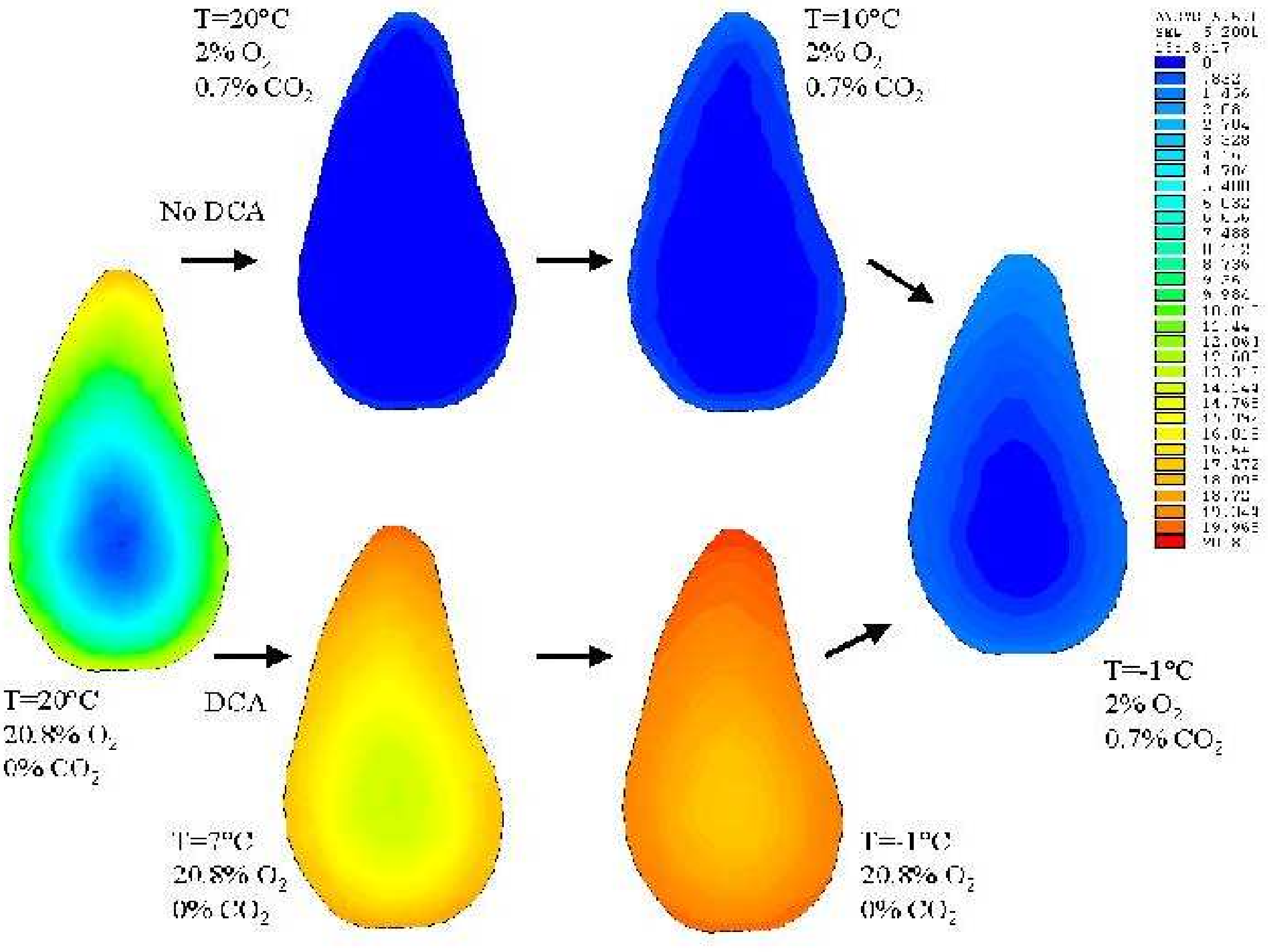
Problems

- computational complexity of Gauss elimination (3D)
  - large computing time
  - problematic memory consumption
- convergence to unphysical (negative) solutions
  - solved through parameter continuation strategy

Optimization for the solver

- **AMG**
  - AMG1R5 for scalar equations (K. Stüben)
  - system AMG code (K. Stüben)
    - "unknown-based" approach
    - "point-based" approach
  - called from Matlab (MEX-file)
- **inexact** modified Newton
- steady-state and time-evolution problem

Oxygen concentration profiles



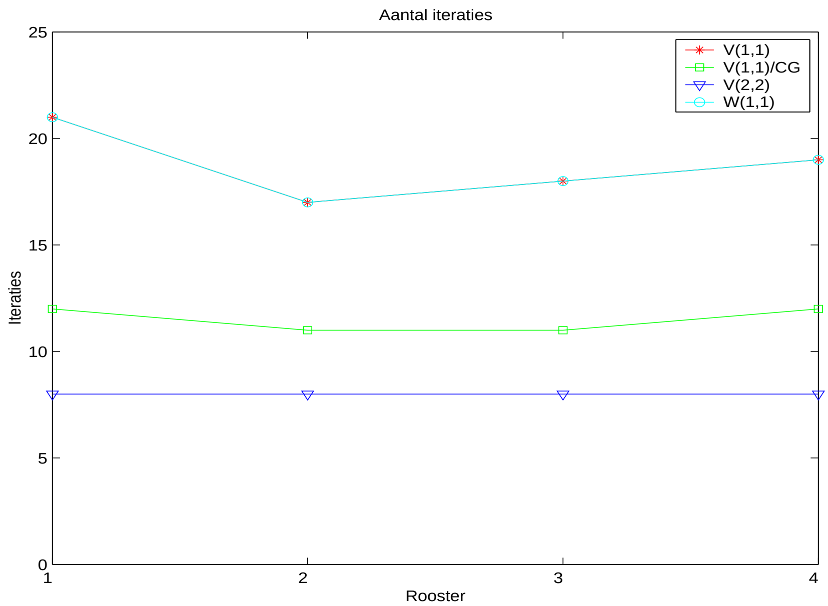
Different storage atmospheres
temperature, percentage O_2, CO_2

(source: PhD J. Lammertyn, 2002)

4. OUR AMG EXPERIENCE

Steady-state, scalar, linearized model problem

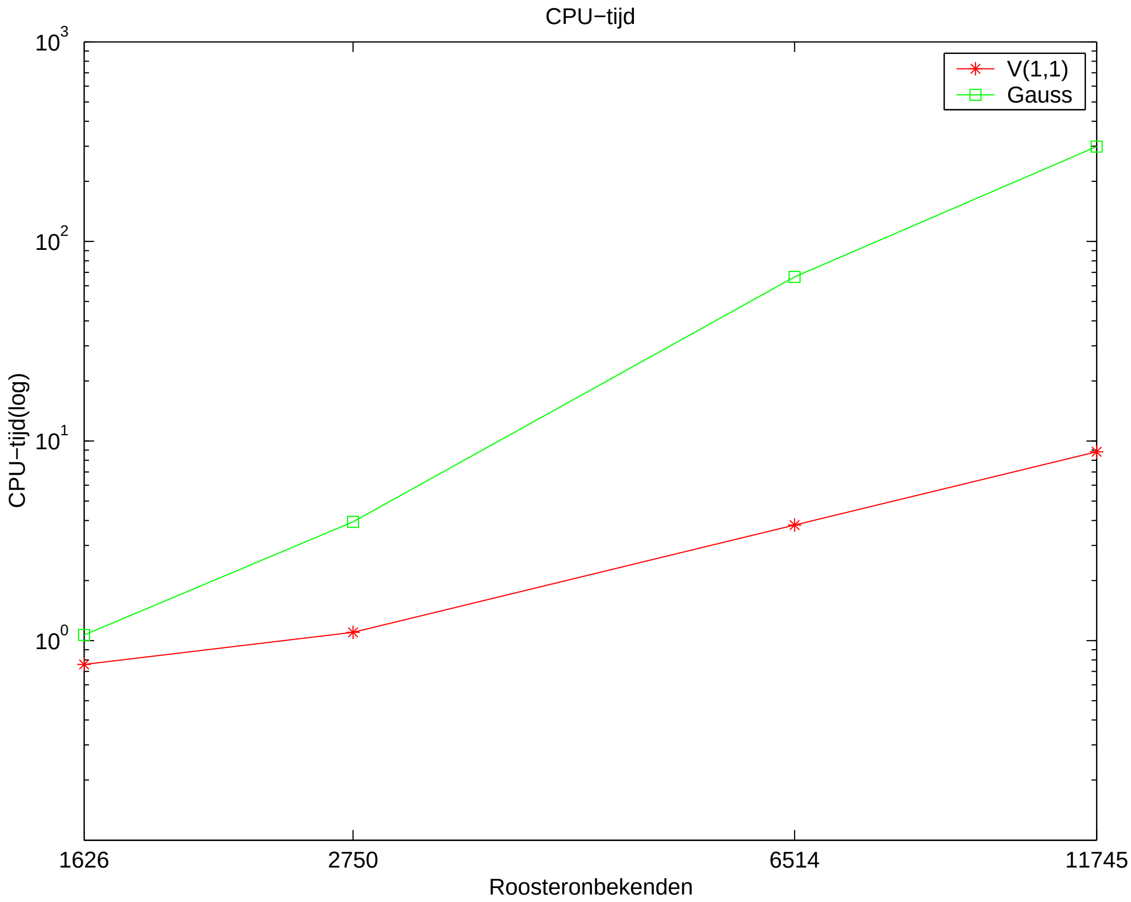
- Number of iterations



- Convergencefactors

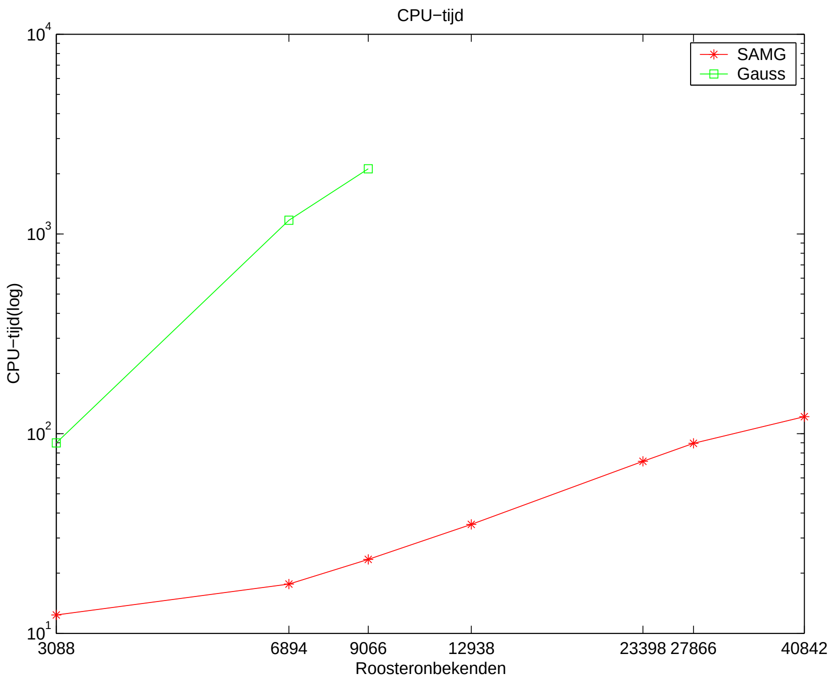
Variables	V(1,1)	V(1,1)/CG	V(2,2)	W(1,1)
2720	0.13	0.04	0.01	0.13
6514	0.15	0.04	0.01	0.15
11745	0.17	0.05	0.01	0.17

- CPU-time comparison



Steady-state, full model

- starting value important for realistic convergence
- coupled system, 2 unknowns: "unknown-based" AMG
- standard coarsening, on the level of variables, treated independently for the unknowns
- smoother: "variable-wise" Gauss-Seidel (first all O_2, then all CO_2)
- V(2,1)-cycle as a preconditioner for CG
- coarsest-level solver: sparse Gauss-elimination
- CPU-time comparison



Time-dependent: Backward Differentiation

- set of equations

$$C\frac{du}{dt} + Ku = f(u)$$

- stiff implicit time-discretisation methods
- BDF1: Implicit Euler method

$$C\frac{u_{n+1} - u_n}{\Delta t} + Ku_{n+1} = f(u_{n+1})$$

- BDF2

$$C\frac{3u_{n+1} - 4u_n + u_{n-1}}{2\Delta t} + Ku_{n+1} = f(u_{n+1})$$

- solve in each time-step FEM discretisation of stationary PDE

$$\frac{\alpha}{\Delta t}u - Lu = f(u)$$

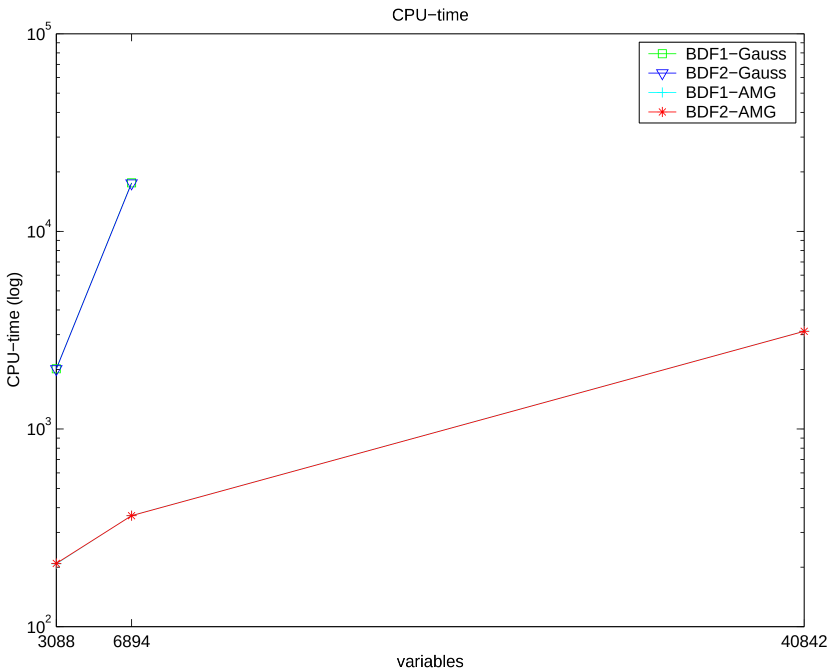
Results

- "unknown-based" approach
  "variable wise" Gauss-Seidel relaxation

- numerical results
  3D model, 3088 variables, $\Delta t = 300s$

	BDF1		BDF2	
	Gauss	AMG	Gauss	AMG
time(s)	1.8	0.14	1.8	0.14
ρ		0.015		0.019

- CPU-time comparison



Time-dependent: Implicit Runge-Kutta

- **IRK method**

$$Cu_{n+1} = Cu_n + \Delta t \sum_{j=1}^{s} b_j(-KU_j + f(\cdot))$$

with

$$CU_i = Cu_n + \Delta t \sum_{j=1}^{s} a_{ij}(-KU_j + f(\cdot))$$

characterized by

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array}$$

- Radau IIA method of order 5

$$\begin{array}{c|ccc} \frac{4-\sqrt{6}}{10} & \frac{88-7\sqrt{6}}{360} & \frac{296-169\sqrt{6}}{1800} & \frac{-2+3\sqrt{6}}{225} \\ \frac{4+\sqrt{6}}{10} & \frac{296+169\sqrt{6}}{1800} & \frac{88+7\sqrt{6}}{360} & \frac{-2-3\sqrt{6}}{225} \\ 1 & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \\ \hline & \frac{16-\sqrt{6}}{36} & \frac{16+\sqrt{6}}{36} & \frac{1}{9} \end{array}$$

- solve in each time-step FEM discretisation of system of stationary PDEs for the stage unknowns

- simplified Newton iterations

$$(C - \Delta t A \otimes J)\Delta U^k = -CU^k + \Delta t(A \otimes I)F(U^k)$$

$$U^{k+1} = U^k + \Delta U^k$$

Results

- 2 unknowns (O_2, CO_2), 3-stage IRK method: 6-dimensional nonlinear system to solve

- SAMG: coupled system with 6 unknowns "point-based" approach

- standard point-coarsening, on the level of points

- smoother: collective (block) Gauss-Seidel relaxation

- V(1,1)-cycle as a preconditioner for CG

- coarsest-level solver: sparse Gauss-elimination

Numerical results

- 2D model, 1932 variables, 6 unknowns

$\Delta t = 600s$	Gauss	AMG
Time(s)	0.3	0.08
CPU-Time(s)	147.9	39.4
# Newton iter	493	493

- 3D model, 9264 variables, 6 unknowns

$\Delta t = 600s$	Gauss	AMG
Time(s)	34	0.8
CPU-Time(s)	12376	338.4
# Newton iter	364	423
ρ		0.025

5. CONCLUDING REMARKS

- **AMG**
  - "black-box" solver (MEX-file)
  - improvement in computing time
    * 2D: factor 2
    * 3D: factor 50
  - improvement in memory consumption

- **time-stepping**
  - BDF1/BDF2: good results with AMG (as expected)
  - IRK: solution of implicit system for stage values no problem for AMG (not as expected)

    "...efficient solution of the system is the main problem..."

    (Solving ODEs II, Hairer and Wanner, 2002)