UNIVERSITY OF CAMBRIDGE

CoMo GROUP

Detailed population balance modelling of soot in flames

Prof. Markus Kraft

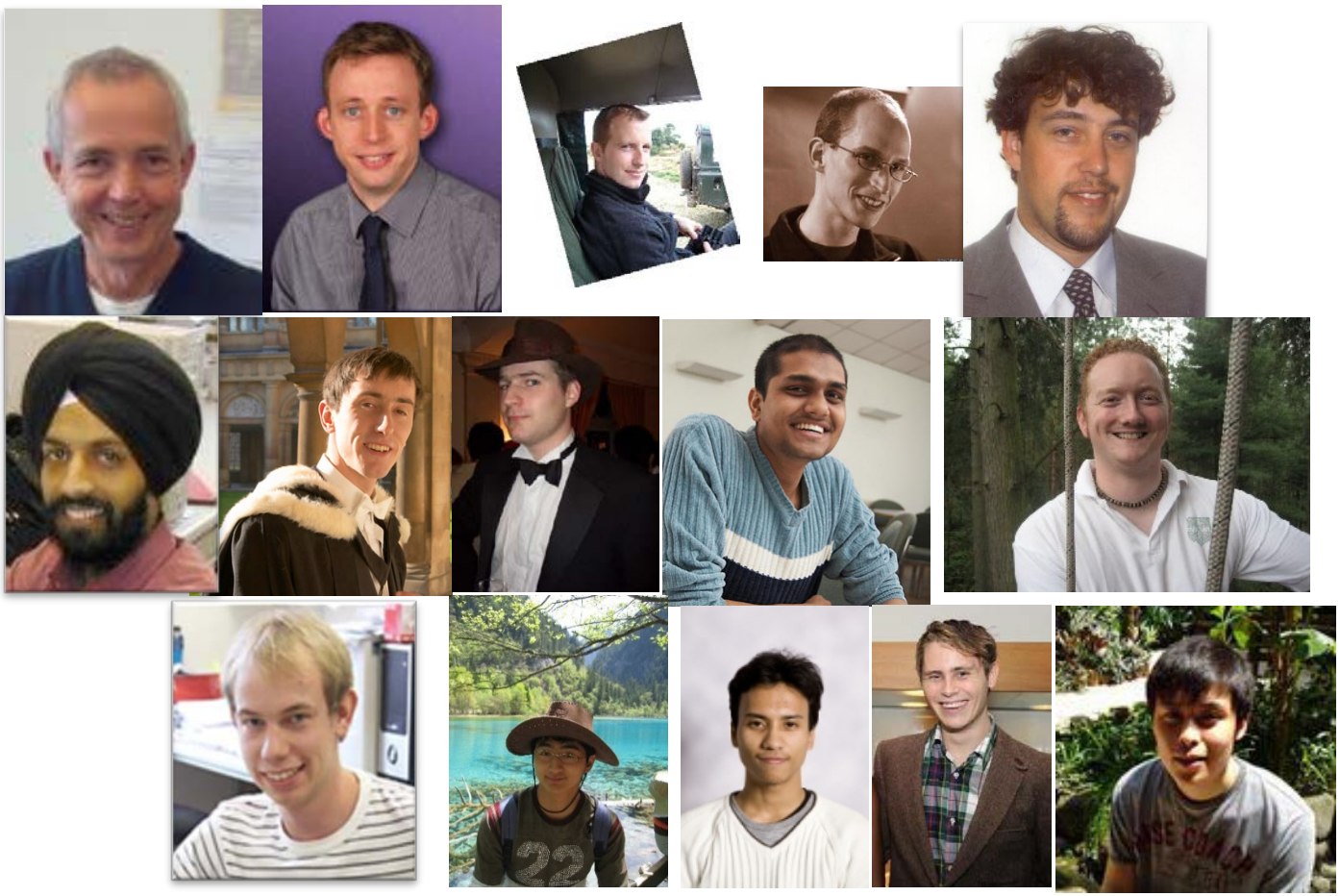
10th IMACS Seminar on Monte Carlo Methods

Linz/Austria July 6-10 2015

Chemical Engineering and Biotechnology

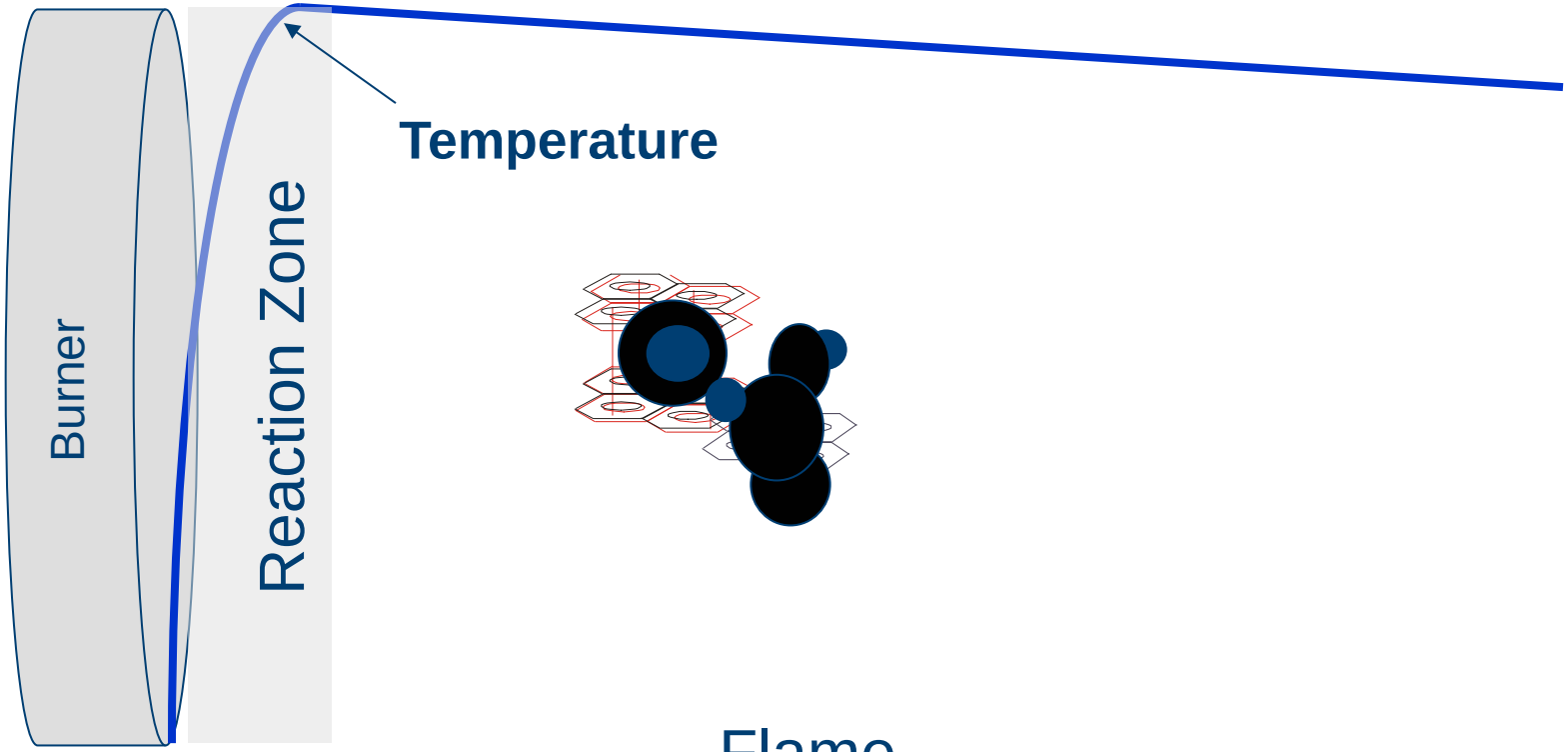
Co-authors

- Wolfgang Wagner
- Mike Goodson
- Jethro Akroyd
- Sebastian Mosbach
- Michael Balthasar
- Jasdeep Singh
- Robert Patterson
- Matt Celnik
- Abhijeet Raj
- Neal Morgan
- Markus Sander
- Dongping Chen
- Zackwan Zainuddin
- Will Menz
- Edward Yapp



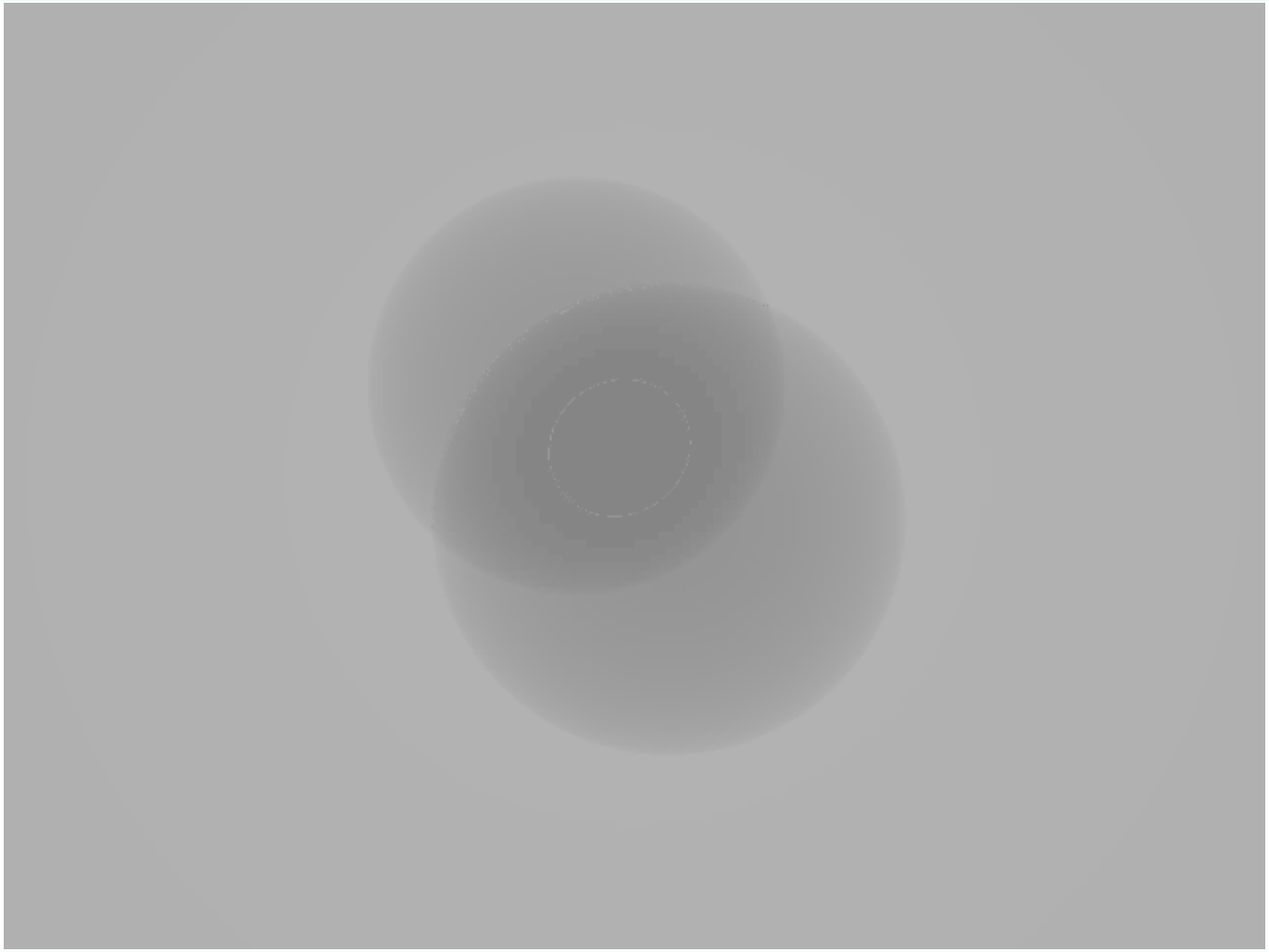
Laminar premixed stabalised flame

Soot Formation

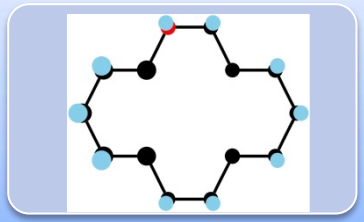


Trajectory of a soot particle

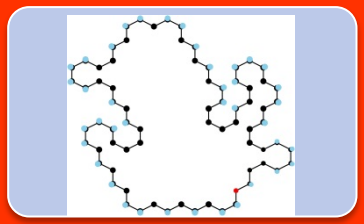
Trajectory of a soot particle



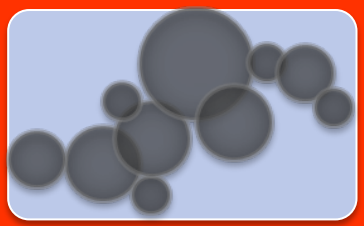
Detailed population balance model



Gas-phase chemistry up to pyrene

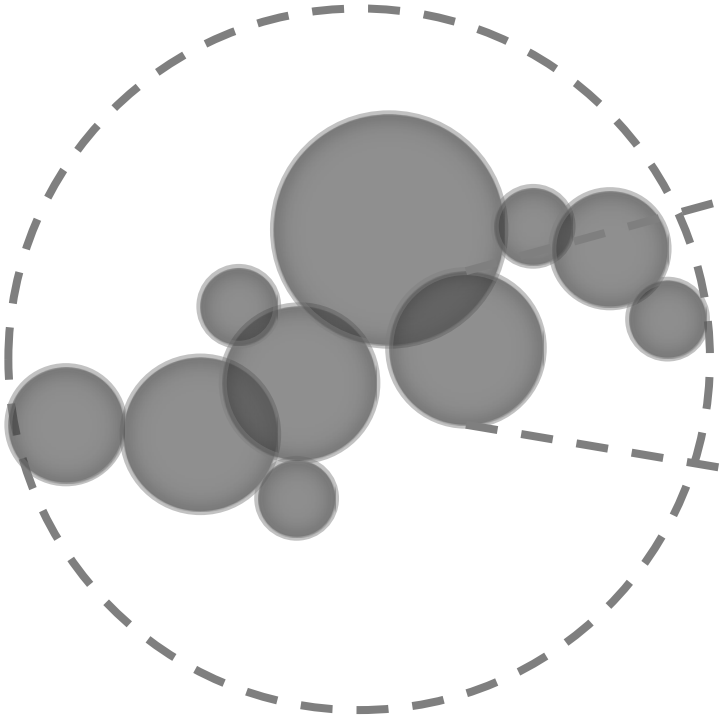


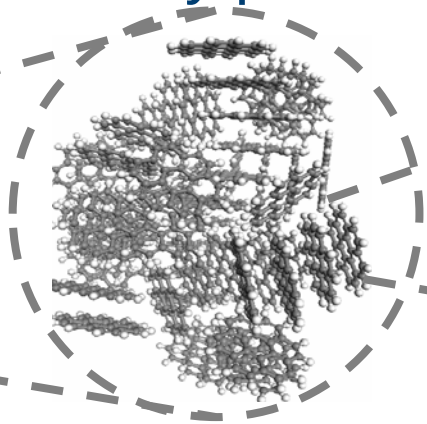
PAH dynamics beyond pyrene

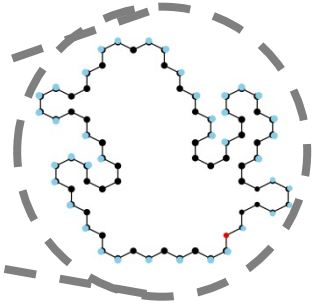


Soot and PAH dynamics

Particle representation







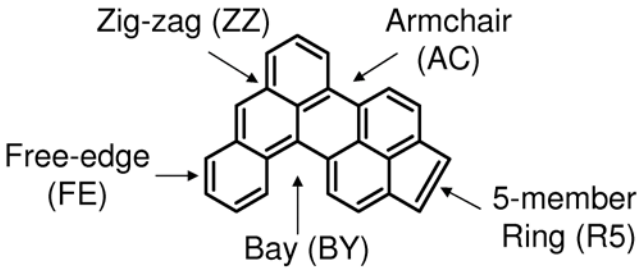
- PAHs rigidly stick

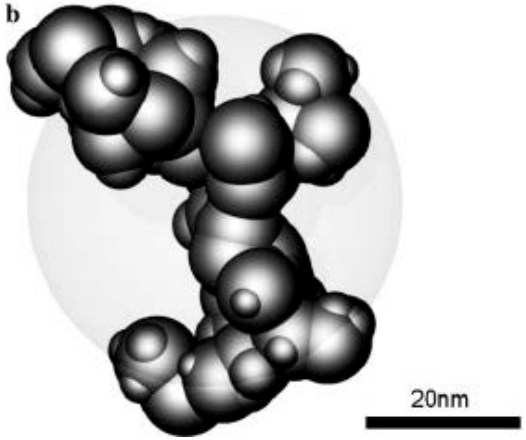
- Edge C
- Site types
- Fringe length

- Connectivity
- Joint surface area

} Sintering level

Detailed population balance model





PAH + PAH

Kinetic Monte-Carlo Aromatic Site (KMC-ARS)

→

Soot

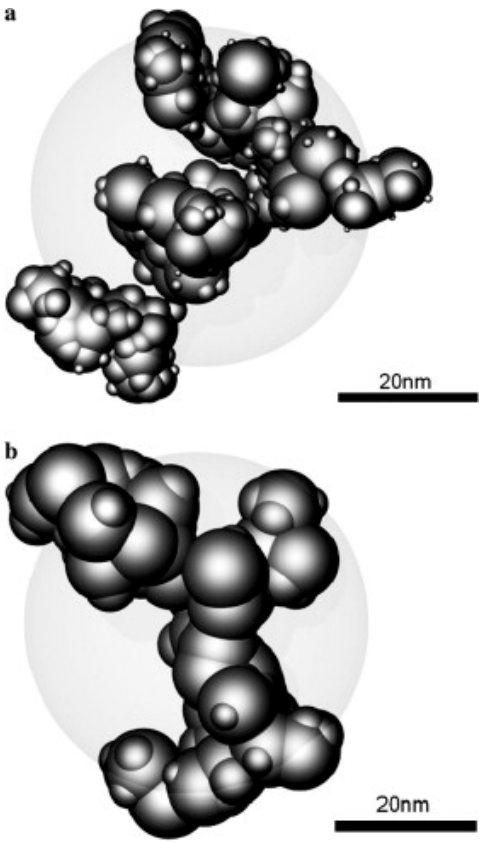
PAH-Primary Particle (PAH-PP)

Type space: PAH-PP model

- Each particle is represented as:

$$P_q = P_q\left(p_1, \cdots, p_{n(P_q)}, \boldsymbol{C}\right)$$

where $\boldsymbol{C}$ is the connectivity matrix



N. Morgan, M. Kraft, M. Balthasar, D. Wong, M. Frenklach, P. Mitchell, Proc. Combust. Inst. 31 (2007) 693–700

Type space: PAH-PP model

- Connectivity matrix $\boldsymbol{C}$ stores the common surface area between two primary particles:

$$\boldsymbol{C}(P_q) = \begin{pmatrix} 0 & \cdots & 0 & \cdots & 0 \\ C_{21} & \ddots & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \cdots & \vdots \\ C_{i1} & \cdots & C_{ij} & \ddots & \vdots \\ \vdots & \cdots & \vdots & \cdots & \vdots \end{pmatrix}$$

- Element C_{ij} of matrix $\boldsymbol{C}$ has the following properties:

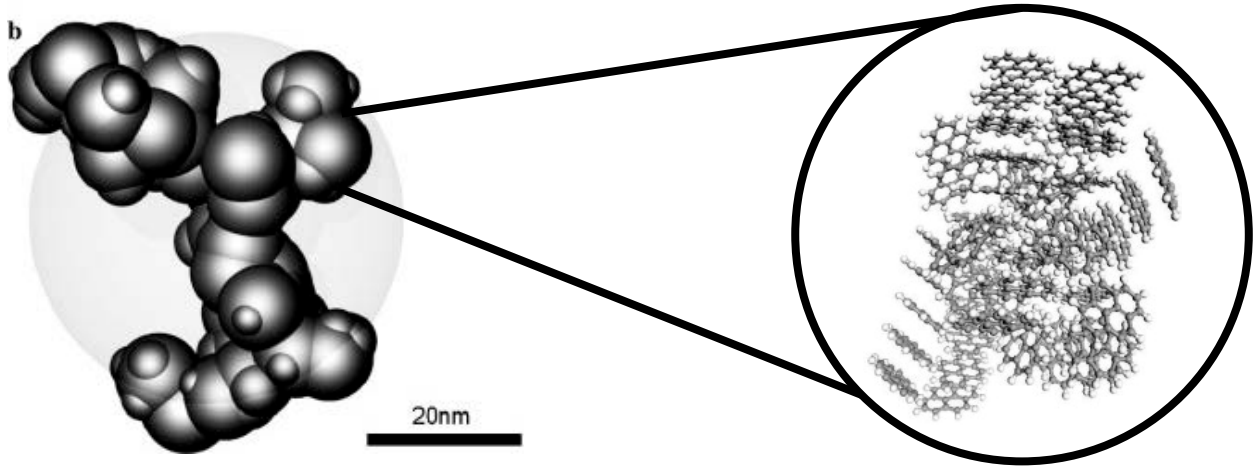
$$C_{ij} = \begin{cases} 0, & \text{non-neighbouring} \\ S_{\text{sph}}(p_i, p_j) \leq C_{ij} \leq s(p_i) + s(p_j), & \text{neighbouring} \end{cases}$$

Type space: KMC-ARS model

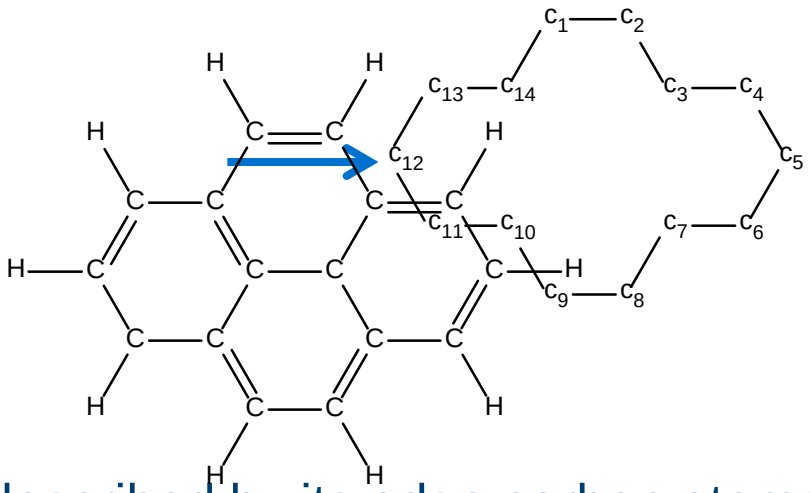
- Each primary p_i is represented as:

$$p_i = p_i\left(m_1, \cdots, m_{n(p_i)}\right)$$

where m is the exact structure of a PAH



Type space: KMC-ARS model

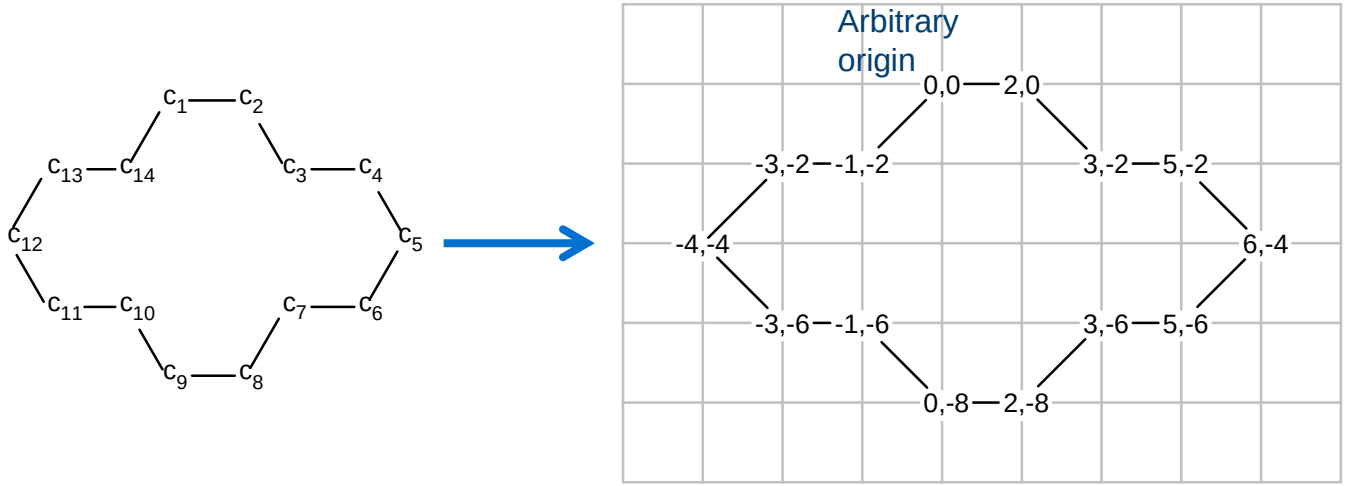


- Each PAH m_a is fully described by its edge carbon atoms:

$$m_a = m_a(c_1, \cdots, c_{n(m_a)})$$

- Surface reactions on edge carbon atoms

Type space: KMC-ARS model



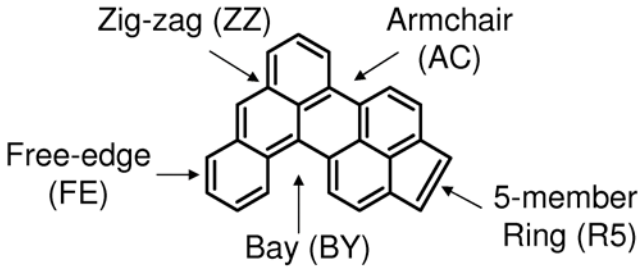
- Each carbon atom c_b has spatial coordinates:

$$c_b = c_b(\eta_i, \eta_j)$$

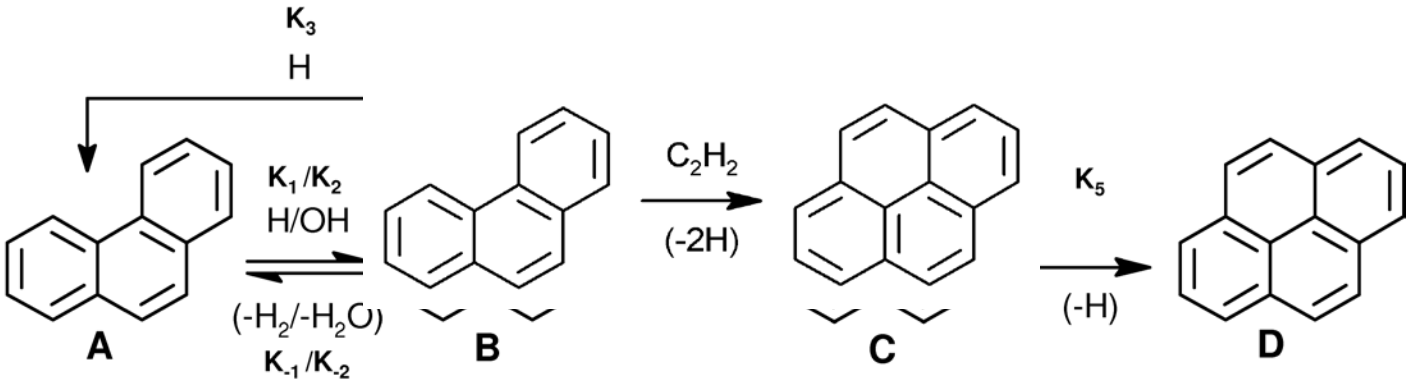
where η are units in the horizontal and vertical direction

KMC-ARS model

- Surface reactions happen when gas-phase molecules react with sites on a PAH

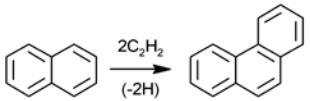


- Armchair growth reaction

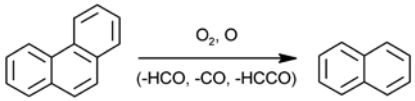


KMC-ARS transformations

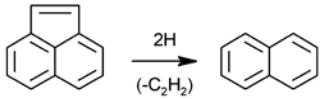
- Surface reactions are described by a set of 20 jump processes:



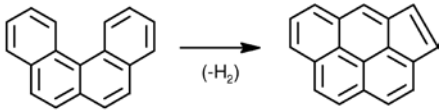
S1 Free-edge ring growth



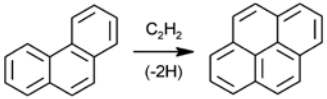
S6 Free-edge oxidation by O2



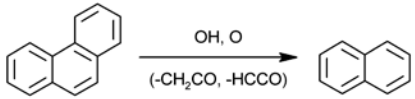
S11 5-member ring desorption



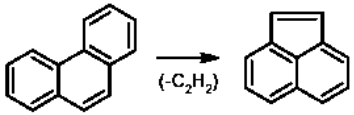
S16 6-member ring rearrangement at bay



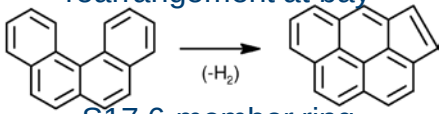
S2 Armchair ring growth



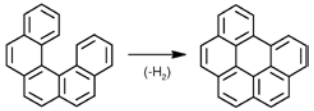
S7 Free-edge oxidation by OH



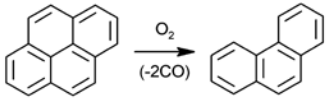
S12 Armchair 6- to 5-member ring conversion



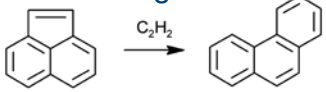
S17 6-member ring rearrangement at bay



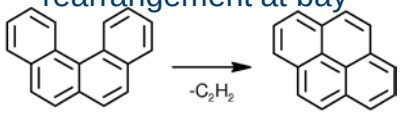
S3 6-member bay closure



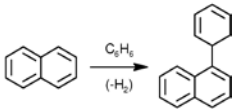
S8 Armchair oxidation by O2



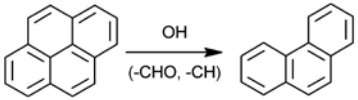
S13 Free-edge 5- to 6-member ring conversion



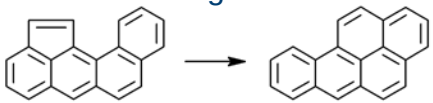
S18 6-member ring desorption at bay

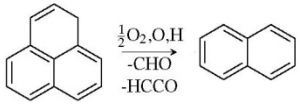


S4 Benzene addition

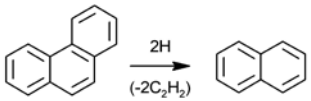


S9 Armchair oxidation by OH

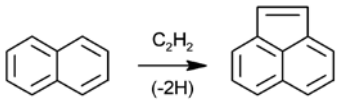

S14 Armchair 5- to 6-member ring conversion



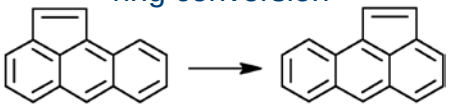
S19 Zig-zag oxidation by O2

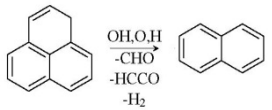


S5 Free-edge ring desorption



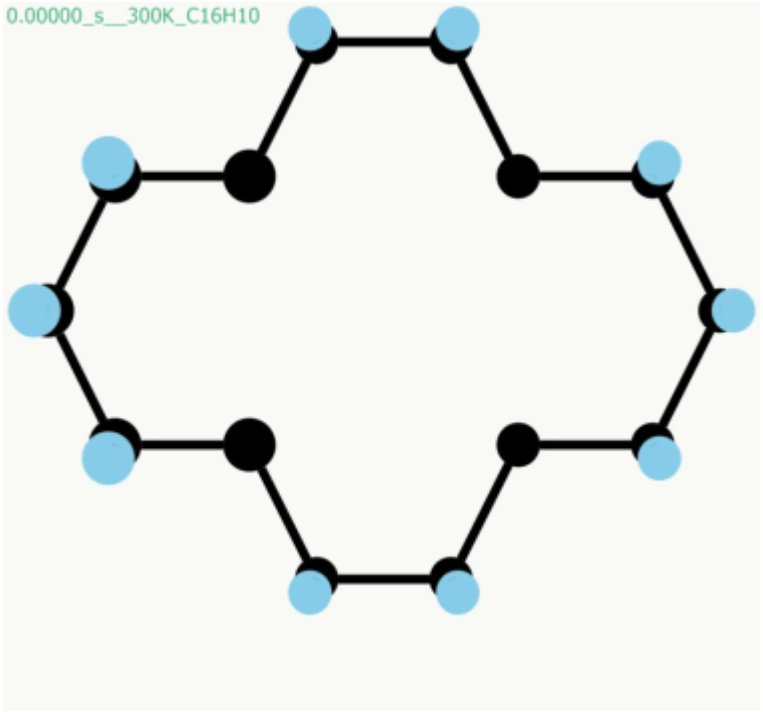
S10 Zig-zag 5-member ring addition


S15 5-member ring migration



S20 Zig-zag oxidation by OH

KMC-ARS transformation

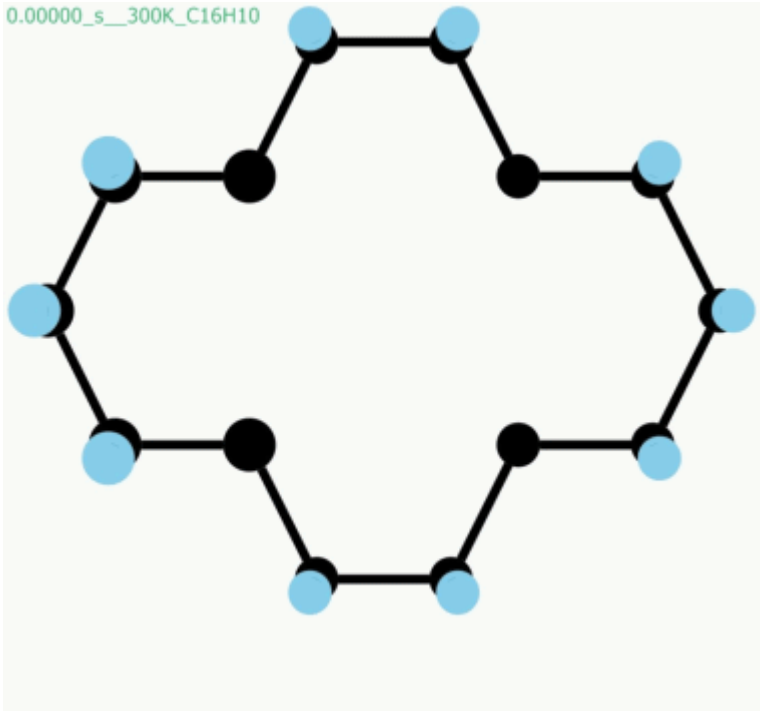


D. Chen, Z. Zainuddin, E. Yapp, J. Akroyd, S. Mosbach, M. Kraft, Proc. Combust. Inst. 34 (2013) 1827–1835

	Values
Fuel composition (mol%)	50 C_2H_4, 50 O_2
Velocity (cm/s)	54
Pressure (bar)	0.15
Equivalence ratio (-)	3.0

J. Happold, Geschichtete polyzyklische aromatische Kohlenwasserstoffe als Bausteine der Rußbildung, Universität Stuttgart, Stuttgart, 2008

KMC-ARS transformation



D. Chen, Z. Zainuddin, E. Yapp, J. Akroyd, S. Mosbach, M. Kraft, Proc. Combust. Inst. 34 (2013) 1827–1835

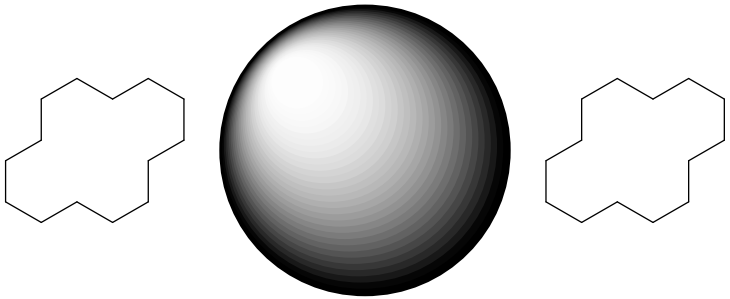
	Values
Fuel composition (mol%)	50 C_2H_4, 50 O_2
Velocity (cm/s)	54
Pressure (bar)	0.15
Equivalence ratio (-)	3.0

J. Happold, Geschichtete polyzyklische aromatische Kohlenwasserstoffe als Bausteine der Rußbildung, Universität Stuttgart, Stuttgart, 2008

Particle processes: Inception

- Two PAHs in the gas-phase stick together introducing a new particle into the system consisting of one primary
- Inception is implemented as:

$$P_q(p_1(m_1), \boldsymbol{C}) + P_r(p_1(m_1), \boldsymbol{C}) \rightarrow P_s(p_1(m_1, m_2), \boldsymbol{C})$$



Particle processes: Coagulation

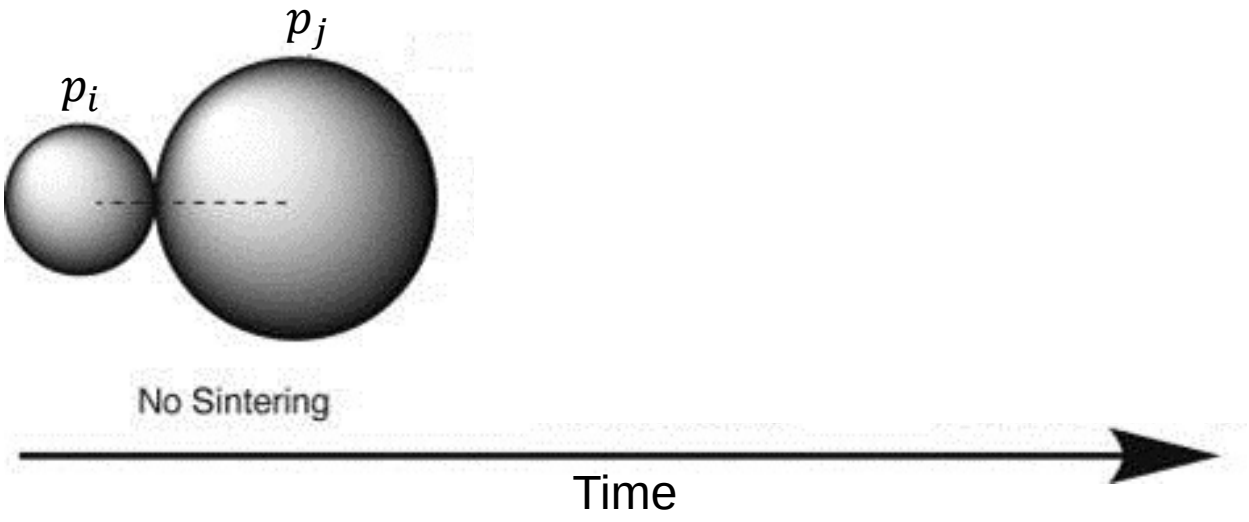
- Coagulation occurs when two particles stick to each other and assume point contact
- The coagulation of particles P_q and P_r is implemented as:

$$P_q\left(p_1,\cdots,p_{n(P_q)},\boldsymbol{C}\right)+P_r\left(p_1,\cdots,p_{n(P_r)},\boldsymbol{C}\right)\rightarrow$$
$$P_s\left(p_1,\cdots,p_{n(P_q)},p_{(n(P_q)+1)},\cdots,p_{n(P_q)+n(P_r)},\boldsymbol{C'}\right)$$

- Primary particle p_i from P_q and a primary particle p_j from P_r are in point contact

Particle processes: Sintering

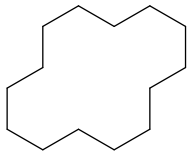
- Rearrangement of PAHs to minimise the surface area of the particle

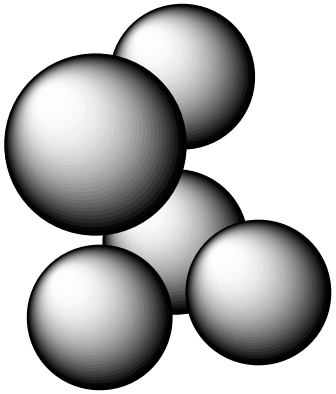


Particle processes: Condensation

- Particles can grow by the deposition of PAHs on the particle surface
- One of the particles is a single PAH species:

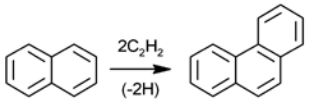
$$P_q(p_1(m_1), \boldsymbol{C}) + P_r\left(\cdots, p_k\left(m_1, \cdots, m_{n(p_k)}\right), \cdots, \boldsymbol{C}\right)$$
$$\rightarrow P_s\left(\cdots, p_k\left(m_1, \cdots, m_{n(p_k)}, m_{(n(p_k)+1)}\right), \cdots, \boldsymbol{C}'\right)$$



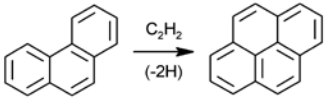


Particle processes: Surface reaction

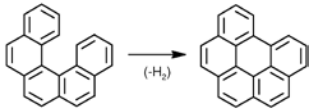
- Surface reactions are described by a set of 20 jump processes:



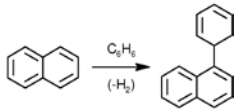
S1 Free-edge ring growth



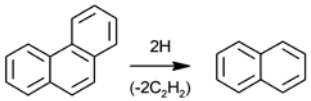
S2 Armchair ring growth



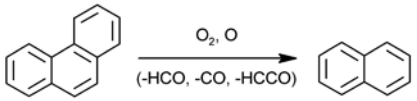
S3 6-member bay closure



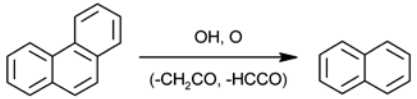
S4 Benzene addition



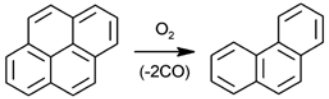
S5 Free-edge ring desorption



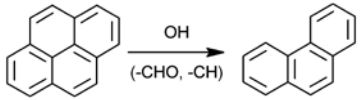
S6 Free-edge oxidation by O2



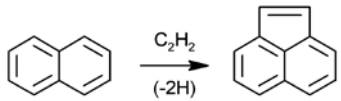
S7 Free-edge oxidation by OH



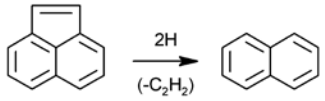
S8 Armchair oxidation by O2



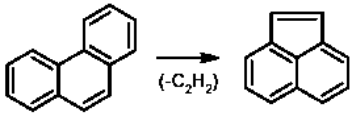
S9 Armchair oxidation by OH



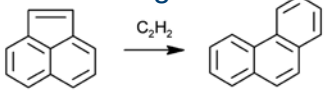
S10 Zig-zag 5-member ring addition



S11 5-member ring desorption

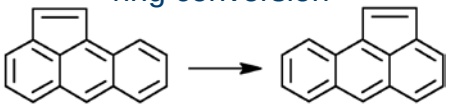


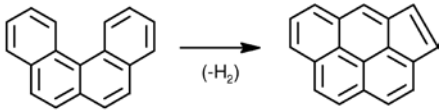
S12 Armchair 6- to 5-member ring conversion



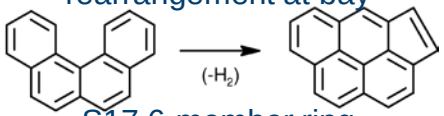
S13 Free-edge 5- to 6-member ring conversion


S14 Armchair 5- to 6-member ring conversion

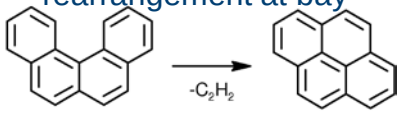

S15 5-member ring migration



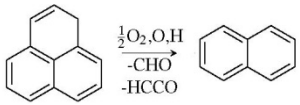
S16 6-member ring rearrangement at bay



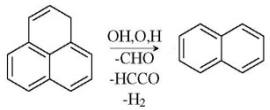
S17 6-member ring rearrangement at bay



S18 6-member ring desorption at bay



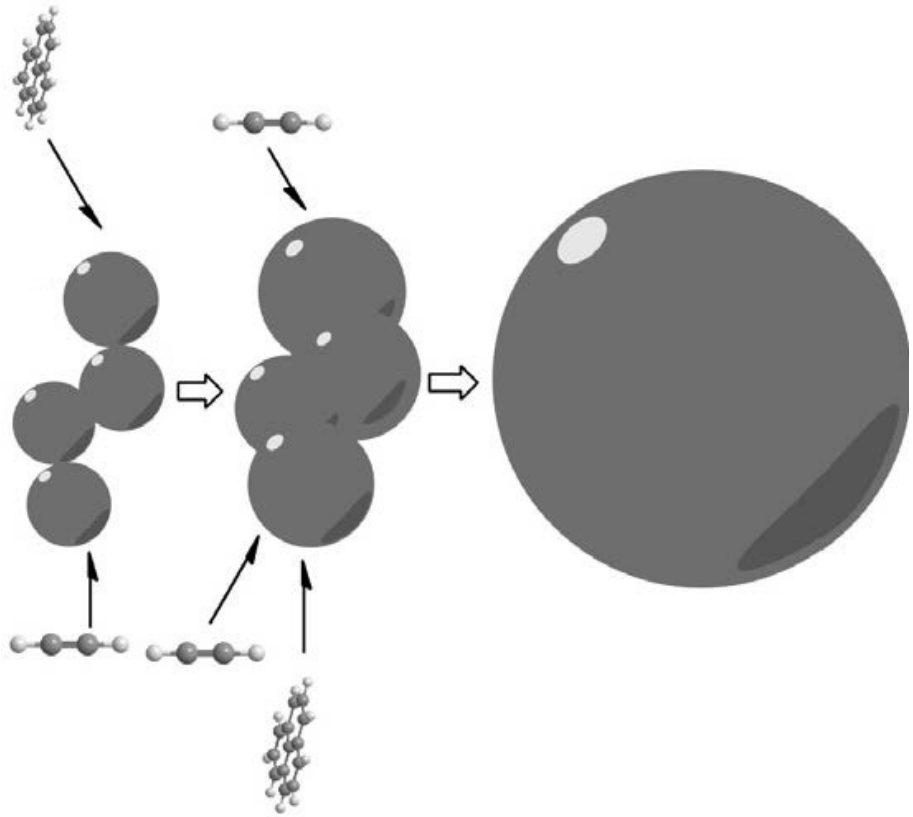
S19 Zig-zag oxidation by O2



S20 Zig-zag oxidation by OH

Particle process: Surface rounding

- Surface rounding by mass addition



Particle process: Surface rounding

- Condensation or surface reaction increases its mass and volume.
- The change in volume results in the change in net common surface area of the primary particle p_i with all its neighbouring primaries:

$$\Delta s(p_i) = \Delta v(p_i) \frac{2s}{d_{\mathrm{pri}}(p_i)}$$

Where s is the smoothing factor such that $0 \leq s \leq 2$

- The corresponding change in $\boldsymbol{C}'$ is given by:

$$C_{ij} = \begin{cases} 0, & \text{non-neighbouring} \\ C_{ij} + \Delta s(p_i), & \text{neighbouring} \end{cases}$$

Algorithms

- The overall model consists of the kinetic model and the stochastic particle model
- These models can be coupled using operator splitting techniques:
  - Strang splitting
  - Predictor-Corrector algorithm
- Simplest algorithm to solve the population balance is the Direct Simulation Algorithm (DSA)
- Numerical issues with DSA addressed by Stochastic Weighted Algorithm (SWA)
- Techniques to improve computational efficiency

Operator splitting technique

- Divide and conquer strategy

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = f(x,y) \equiv f_1(x,y) + f_2(x,y)$$

- First-order splitting: first solve for

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = f_1(x,y)$$

followed by

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = f_2(x,y)$$

Operator splitting technique

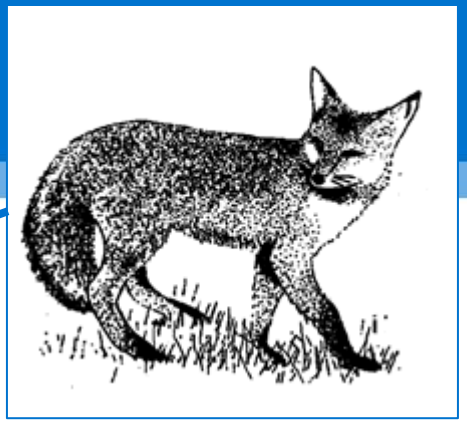
- Second-order splitting: Strang splitting

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{2} f_1(x, y)$$

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = f_2(x, y)$$

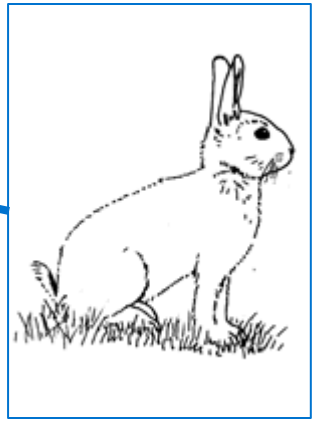
$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{2} f_1(x, y)$$

Operator splitting: Examples



- Lotka-Volterra predator-prey equations

$$\frac{d}{dt}\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x(y-2) \\ y(1-x) \end{pmatrix}$$



- With $z = (x, y)^T$, a simple splitting is

$$f_1(x,y) = \begin{pmatrix} x(y-2) \\ 0 \end{pmatrix}, f_2(x,y) = \begin{pmatrix} 0 \\ y(1-x) \end{pmatrix}$$

Operator splitting: Predictor-Corrector

- Consider the differential equation

$$\frac{dz}{dt} = f(z)$$

- We approximate $f(z)$ by a polynomial so that

$$z_{n+1} = z_n + h(\beta_0 z'_{n+1} + \beta_1 z'_n + \beta_2 z'_{n-1} + \beta_3 z'_{n-2} + \cdots)$$

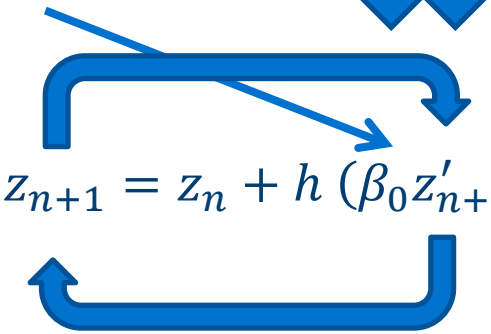
Where $z'_n = f(z_n), z'_{n-1} = f(z_{n-1}), z'_{n-2} = f(z_{n-2}), \cdots$

Operator splitting: Predictor-Corrector

1. Predictor step: explicit method to predict z_{n+1}
2. Corrector step: implicit method to use predicted z_{n+1} to correct z_{n+1}

$$z_{n+1} = z_n + h\,(\beta_0 z'_{n+1} + \beta_1 z'_n + \beta_2 z'_{n-1} + \beta_3 z'_{n-2} + \cdots)$$

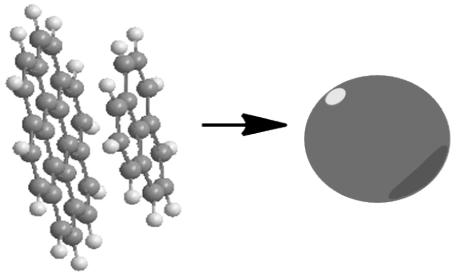
$$z_{n+1} = z_n + h\,(\beta_0 z'_{n+1} + \beta_1 z'_n + \beta_2 z'_{n-1} + \beta_3 z'_{n-2} + \cdots)$$



Operator splitting technique

- The rates of certain particle processes depend on the concentration of certain gas-phase species.

Inception
(collision efficiency model):



- The gas-phase chemistry and the particle processes are coupled
- Technique to couple a stochastic particle model to gas-phase chemistry

State space

The state of the system (Q) at any time consists of two components

1. The first component (Q_1) contains the concentration of the chemical species C:

$$Q_1 = \{C_k : k \in \{1, \cdots, N_g\}\}$$

2. The second component (Q_2) is the stochastic particle system:

$$Q_2 = \{P_q : q \in \{1, \ldots, N\}\}$$

Operator splitting technique

- Evolution of the state of the system Q:

$$\frac{d}{dt}\begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix} = \begin{pmatrix} G_1(Q_1) \\ G_2(Q_1, Q_2) \end{pmatrix} + \begin{pmatrix} P_1(Q_1, Q_2) \\ P_2(Q_1, Q_2) \end{pmatrix}$$

- Operator *G* represents the effects of gas-phase chemical reactions on the system and operator *P* indicates the effects of the particle processes on the system.
- Subscripts 1 and 2 denote gas-phase and particle-phase, respectively.
- The gas-phase chemistry is best solved using an implicit ODE solution technique, as there are only a finite number of gas-phase variables.

M. Celnik, R. Patterson, M. Kraft, W. Wagner, Combust. Flame 148 (2007) 158–176

Operator splitting: Strang splitting

Input: State of the system $Q_0 = Q_{1,0} + Q_{2,0}$ at initial time t_0; Final time t_f.**Output**: State of the system Q_f at final time t_f.

$t_i \leftarrow t_0$, $Q_i \leftarrow Q_0$;

while $t_i < t_f$ **do**

Integrate over time interval $\left[t_i, t_i + \frac{h}{2}\right]$

$$\frac{d}{dt}\begin{pmatrix} Q_1^1 \\ Q_2^1 \end{pmatrix} = \begin{pmatrix} G_1(Q_1^1) \\ G_2(Q_1^1, Q_2^1) \end{pmatrix}$$

With initial conditions

$$\begin{pmatrix} Q_1^1(t_i) \\ Q_2^1(t_i) \end{pmatrix} = \begin{pmatrix} Q_{1,i} \\ Q_{2,i} \end{pmatrix}$$

ODE solver

Solve over time interval $[t_i, t_i + h]$

$$\frac{d}{dt}\begin{pmatrix} Q_1^2 \\ Q_2^2 \end{pmatrix} = \begin{pmatrix} P_1(Q_1^2, Q_2^2) \\ P_2(Q_1^2, Q_2^2) \end{pmatrix}$$

With initial conditions

$$\begin{pmatrix} Q_1^2(t_i) \\ Q_2^2(t_i) \end{pmatrix} = \begin{pmatrix} Q_1^1(t_i + h/2) \\ Q_2^1(t_i + h/2) \end{pmatrix}$$

DSA

Integrate over time interval $\left[t_i + \frac{h}{2}, t_i + h\right]$

$$\frac{d}{dt}\begin{pmatrix} Q_1^3 \\ Q_2^3 \end{pmatrix} = \begin{pmatrix} G_1(Q_1^3) \\ G_2(Q_1^3, Q_2^3) \end{pmatrix}$$

With initial conditions

$$\begin{pmatrix} Q_1^3(t_i) \\ Q_2^3(t_i) \end{pmatrix} = \begin{pmatrix} Q_1^2(t_i + h) \\ Q_2^2(t_i + h) \end{pmatrix}$$

ODE solver

Assign solution at $t_{i+1} = t_i + h$

$$\begin{pmatrix} Q_{1,i+1} \\ Q_{2,i+1} \end{pmatrix} \leftarrow \begin{pmatrix} Q_1^3(t_i + h) \\ Q_2^3(t_i + h) \end{pmatrix}$$

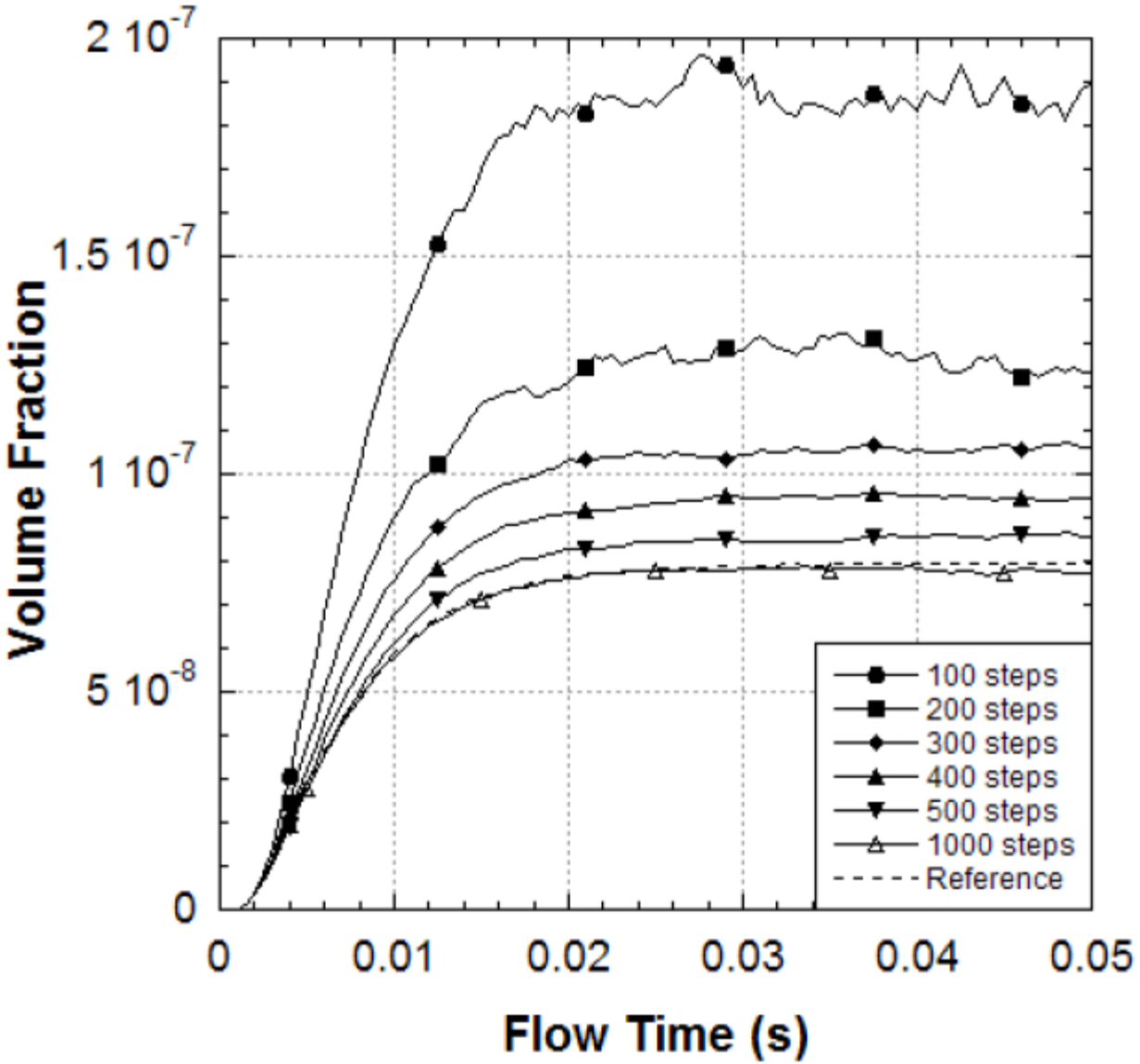
$i \leftarrow i + 1$

end

Numerical issues

- Large magnitude source term in ODE system competing with a similarly sized sink term in the population balance
- Splitting step size very small to keep numerical error sufficiently low

M. Celnik, R. Patterson, M. Kraft, W. Wagner, Combust. Flame 148 (2007) 158–176



Numerical issues

- The system of chemical reactions is inevitably stiff and therefore has to be treated with implicit methods, which involve the inversion of Jacobian matrices
- DSA leads to a discontinuous change in the particle system which makes the reuse of intermediate results impossible
- Reinitialize the ODE solver every splitting step

M. Celnik, R. Patterson, M. Kraft, W. Wagner, J. Comput. Phys. 228 (2009) 2758–2769

Predictor-Corrector algorithm: Predictor step

Input: Approximate solution $Q_1, Q_2, Q_3, \cdots, Q_i$ at times $t_1 < t_2 < t_3, \cdots, t_i$ where $Q_t = Q_{1,t} + Q_{2,t}$

Output: State of the system Q_{i+1} at time t_{i+1}.

Fit a vector of polynomials $P^0(t)$ to m previous points $P_1(Q_{1,i-m}, Q_{2,i-m}), P_1(Q_{1,i-m+1}, Q_{2,i-m+1}), \cdots, P_1(Q_{1,i}, Q_{2,i})$

Solve over time interval $[t_i, t_{i+1}]$

$$\frac{d}{dt}\begin{pmatrix}\tilde{Q}_1^0 \\ \tilde{Q}_2^0\end{pmatrix} = \begin{pmatrix}G_1(\tilde{Q}_1^0) \\ G_2(\tilde{Q}_1^0, \tilde{Q}_2^0)\end{pmatrix} + \begin{pmatrix}P^0(t) \\ 0\end{pmatrix}$$

With initial condition

$$\begin{pmatrix}\tilde{Q}_1^0(t_i) \\ \tilde{Q}_2^0(t_i)\end{pmatrix} = \begin{pmatrix}Q_{1,i} \\ Q_{2,i}\end{pmatrix}$$

Fit a vector of polynomials $Q^0(t)$ to $Q_{1,i-m+1}, Q_{1,i-m+2}, Q_{1,i}, \tilde{Q}_1^0(t_{i+1})$

Solve over time interval $[t_i, t_{i+1}]$

$$\frac{d}{dt}\hat{Q}_2^0 = B_2(Q^0(t), \hat{Q}_2^0)$$

With initial condition

$$\hat{Q}_2^0(t_i) = \tilde{Q}_2^0(t_{i+1})$$

$\left(\tilde{Q}_1^0(t_{i+1}), \hat{Q}_2^0(t_{i+1})\right)^T$ for t_{i+1}

Predictor-Corrector algorithm: Corrector step

Input: Approximate solution $Q_1, Q_2, Q_3, \cdots, Q_i$ at times $t_1 < t_2 < t_3, \cdots, t_i$ where $Q_t = Q_{1,t} + Q_{2,t}$

Output: State of the system Q_{i+1} at time t_{i+1}

$j \leftarrow 1$

while $j < J$ **do**

Fit a vector of polynomials $P^j(t)$ to $m+1$ points $P_1(Q_{1,i-m+1}, Q_{2,i-m+1}), P_1(Q_{1,i-m+2}, Q_{2,i-m+1}), \cdots, P_1(Q_{1,i}, Q_{2,i}), P_1\left(\tilde{Q}_1^{j-1}(t_{i+1}), \hat{Q}_2^{j-1}(t_{i+1})\right)$

Solve over time interval $[t_i, t_{i+1}]$

$$\frac{d}{dt}\begin{pmatrix}\tilde{Q}_1^j \\ \tilde{Q}_2^j\end{pmatrix} = \begin{pmatrix}G_1(\tilde{Q}_1^j) \\ G_2(\tilde{Q}_1^j, \tilde{Q}_2^j)\end{pmatrix} + \begin{pmatrix}P^j(t) \\ 0\end{pmatrix}$$

With initial condition

$$\begin{pmatrix}\tilde{Q}_1^j(t_i) \\ \tilde{Q}_2^j(t_i)\end{pmatrix} = \begin{pmatrix}Q_{1,i} \\ Q_{2,i}\end{pmatrix}$$

Fit a vector of polynomials $Q^j(t)$ to $Q_{1,i-m+1}, Q_{1,i-m+2}, Q_{1,i}, \tilde{Q}_1^j(t_{i+1})$

Solve over time interval $[t_i, t_{i+1}]$

$$\frac{d}{dt}\hat{Q}_2^j = B_2(Q^j(t), \hat{Q}_2^j)$$

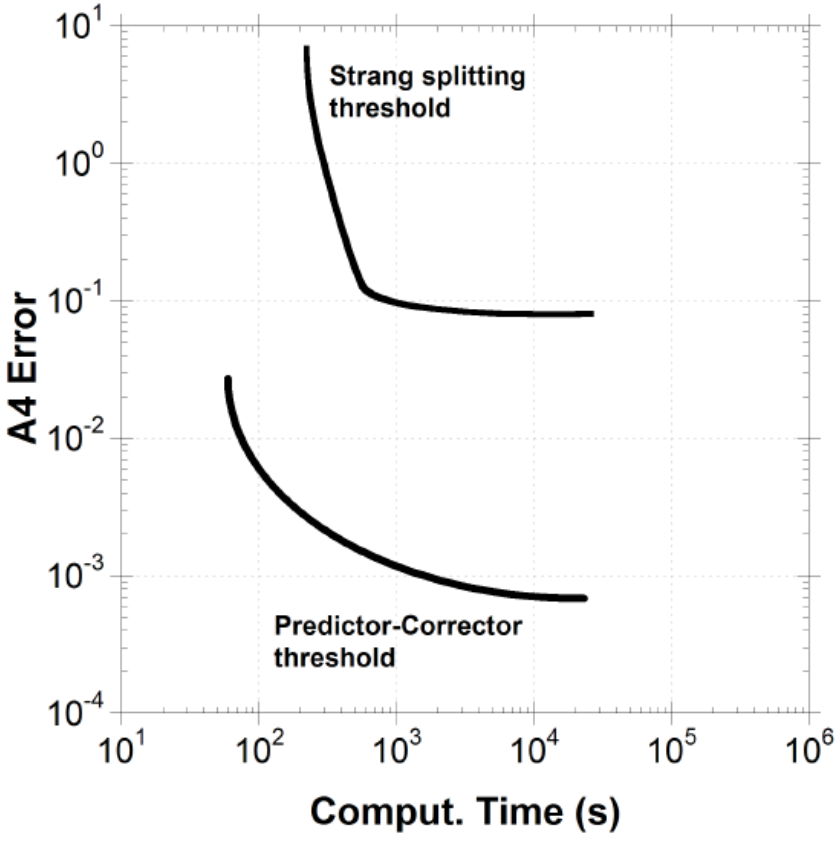
With initial condition

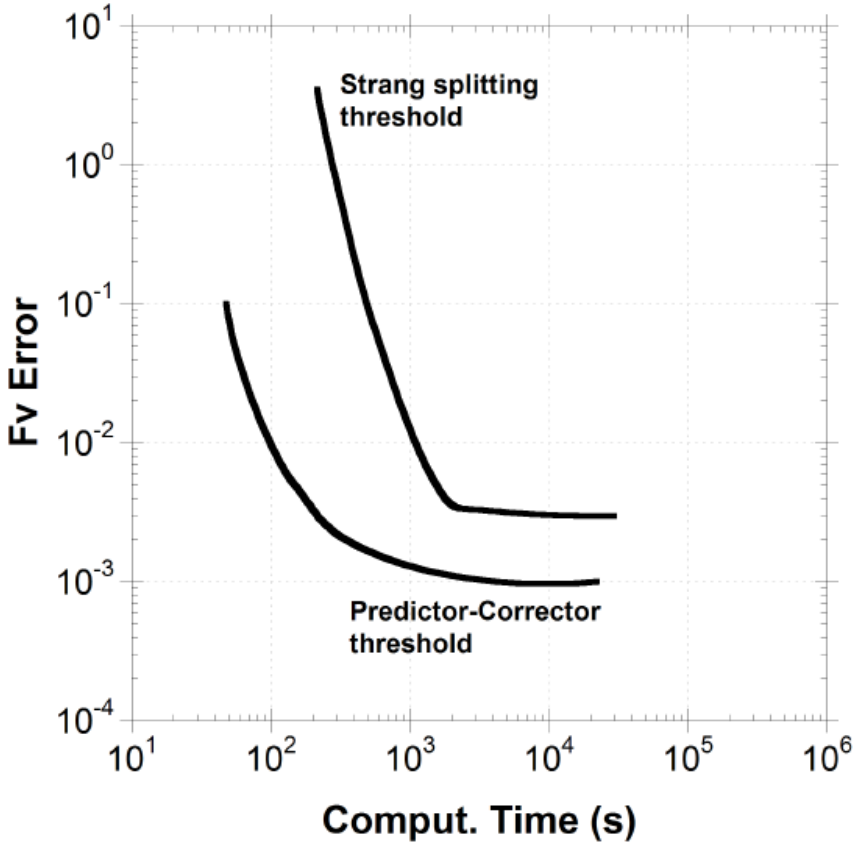
$$\hat{Q}_2^j(t_i) = \tilde{Q}_2^j(t_{i+1})$$

$j \leftarrow j+1$

end

Numerical efficiency





Direct Simulation Algorithm (DSA)

- Solve the population balance equations
- Every process is directly included in the Monte-Carlo algorithm

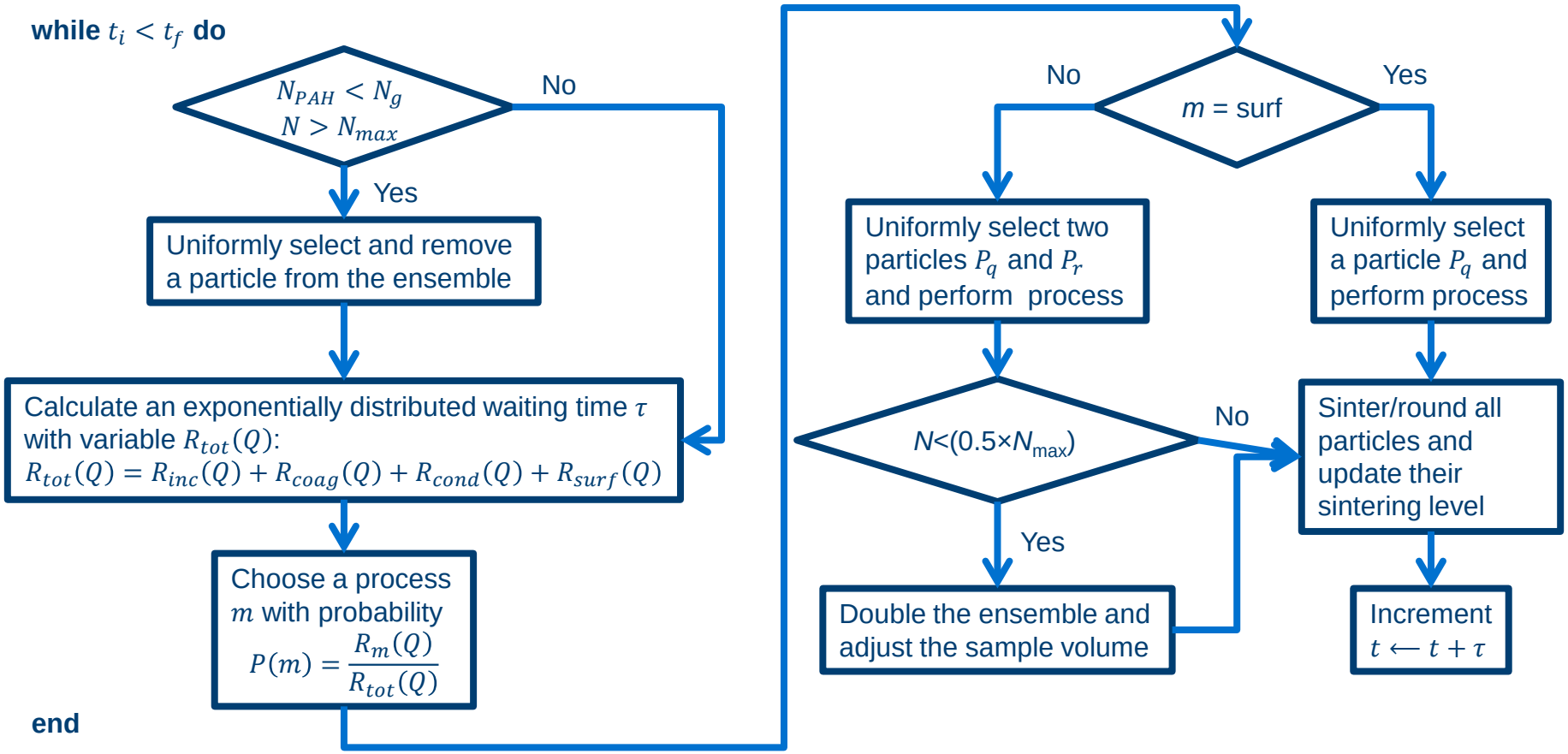
A. Eibeck, W. Wagner, Ann. Appl. Probab. 11 (2003) 845–889

Direct Simulation Algorithm (DSA)

Input: State of the system Q_0 at time t_0; Final time t_f. **Output**: State of the system Q_f at final time t_f.

$t_i \leftarrow t_0$, $Q_i \leftarrow Q_0$;

while $t_i < t_f$ **do**

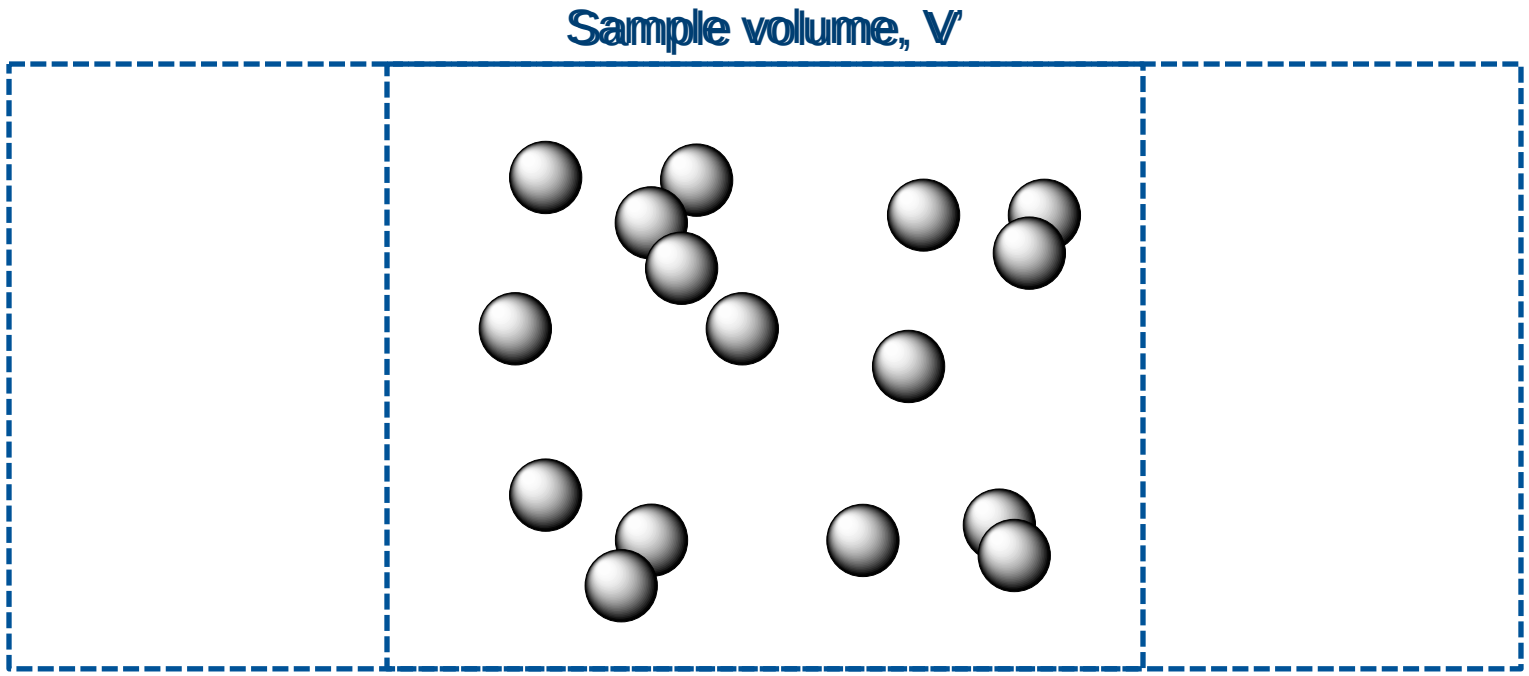


end

Direct Simulation Algorithm (DSA)

- Problems with a variable size particle ensemble:
  - Removing particles until no particles remain
  - Attempting to add a particle when there is insufficient space

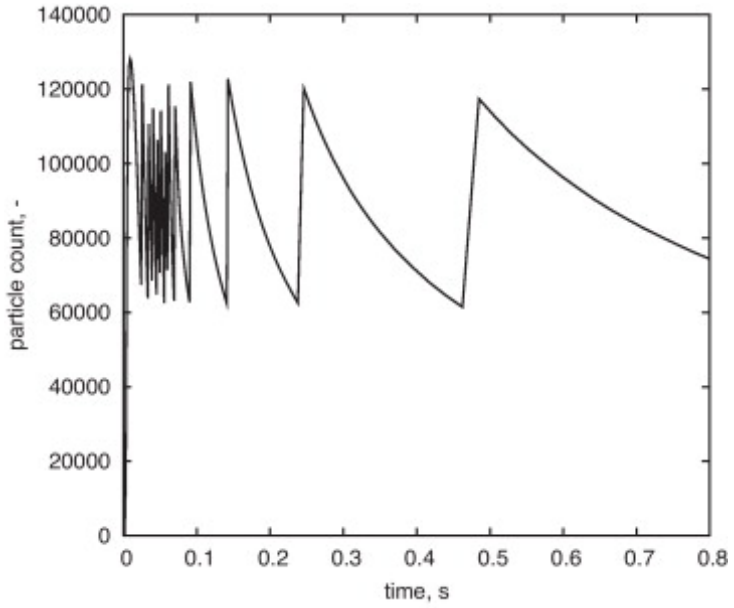
Doubling algorithm



Maximum capacity, N_{max} = 8 particles

Doubling algorithm

- Maintain statistically significant number of particles



. Shekar, W. J. Menz, A. J. Smith, M. Kraft, W. Wagner, Comput. Chem. Eng. 43 (2012) 130–14
Sabelfeld, et al. Monte Carlo Methods and Appl., 2(1):41–87, 1996.

Problems with DSA

- Each stochastic particle represents the same number of physical particles
- DSA yields relatively little information about the rarest particles
  - $N_{\mathrm{benzene}} \sim 10^{19}\ \mathrm{cm}^{-3}$ and $N_{\mathrm{soot}} \sim 10^{11}\ \mathrm{cm}^{-3}$
- Large amounts of computational effort is spent on very common PAHs/particles in order to obtain useful estimates concerning the concentrations of rarer, physically significant large particles

Stochastic weighted algorithm (SWA)

- A pair of particles coagulate such that:

$$(x_i, w_i), (x_j, w_j) \rightarrow (x_i + x_j, \gamma(x_i, w_i, x_j, w_j)), (x_j, w_j)$$

Where w is the statistical weight and γ is a weight transfer function

- Coagulation events do not reduce the number of stochastic particles
- Adjust the weight that is attached to each stochastic particle
- This weight is proportional to the number of 'real' particles represented by the stochastic particle

R. I. A. Patterson, W. Wagner, M. Kraft, J. Comput. Phys. 230 (2011) 7456–7472

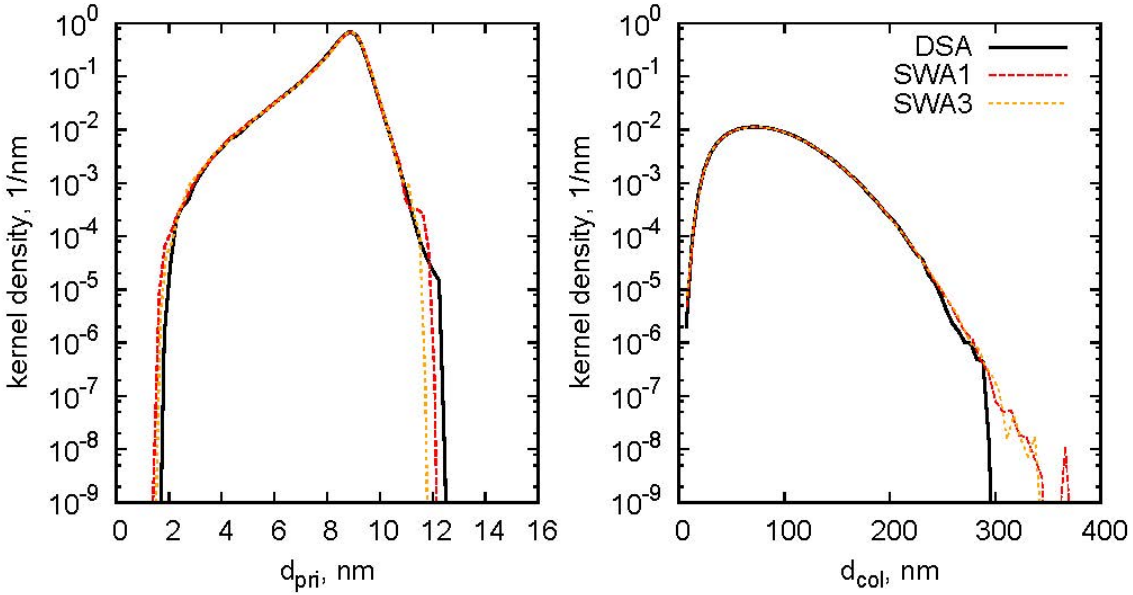
Stochastic weighted algorithm (SWA) – Case study

$$\underset{\text{TEOS}}{Si(OC_2H_5)_4} \rightarrow \underset{\text{Silicic acid}}{Si(OH)_4} \rightarrow \text{Silica}$$

Model parameters	Values
$M_{0,\max}$ (m^{-3})	1.55×10^{18}
A_s (sm^{-1})	1.1×10^{16}
E_s (K)	1.2×10^{5}
$d_{\text{pri,crit}}$ (nm)	4.4
s	1.0

Process settings	Values
Initial temperature (°C)	900
Residence time (s)	0.8
Initial TEOS fraction	250 ppm
Initial total pressure (atm)	1.0

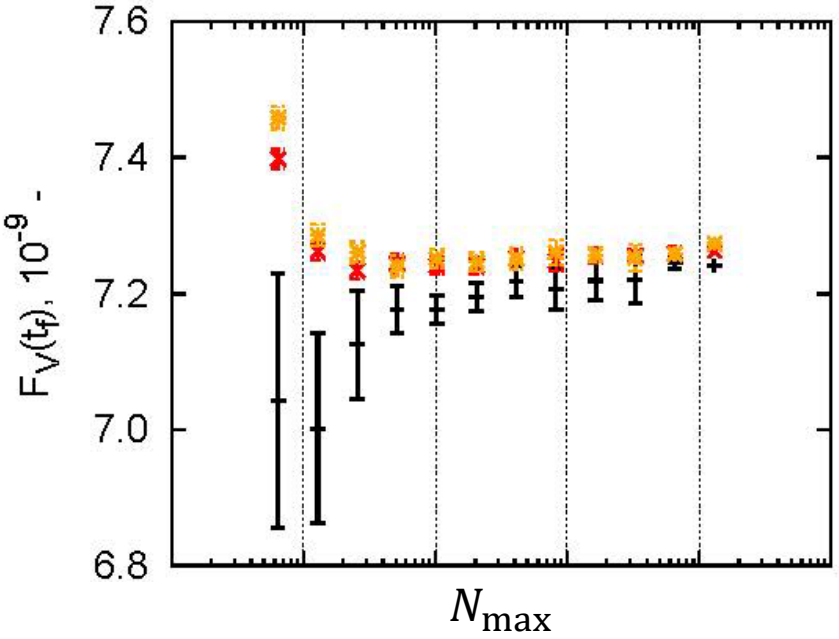
- SWA solution should converge in the limit to DSA solution
- SWA1 = conservation of weight; SWA3 = conservation of mass



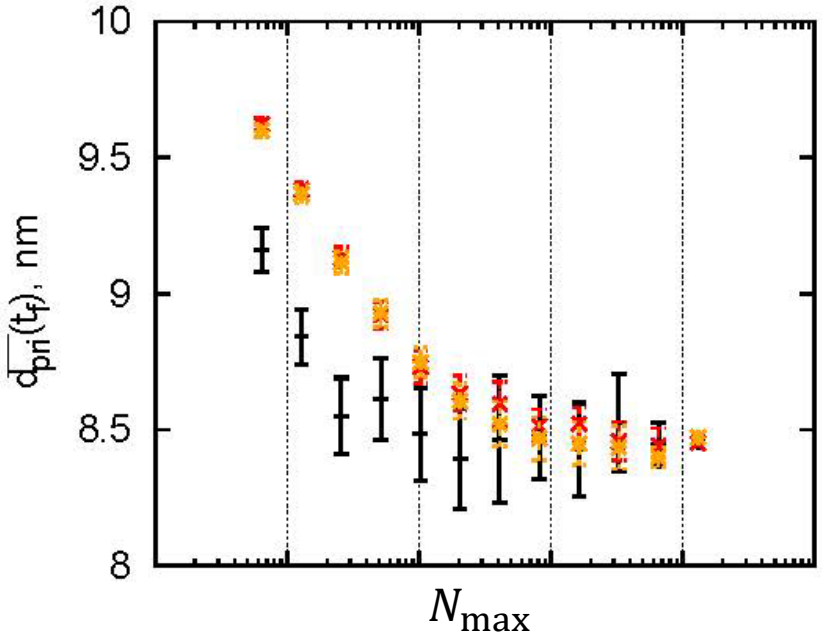
W.J. Menz, R.I.A. Patterson, W. Wagner, M. Kraft, J. Comput. Phys. 248 (2013) 221–234

CoMo GROUP

- SWA shows convergence for fewer stochastic particles
- SWA has a smaller confidence interval than DSA

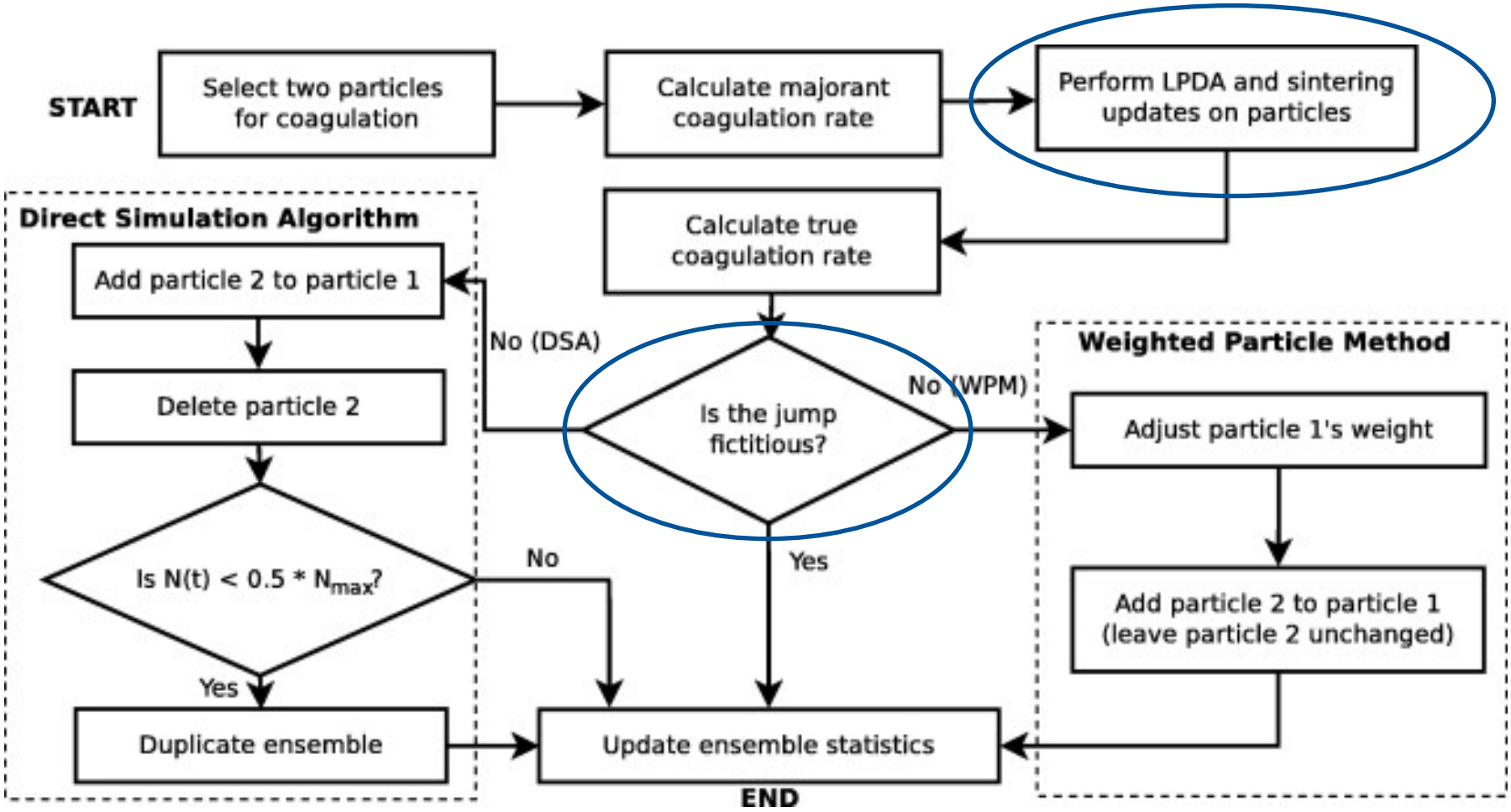


Particle volume fraction

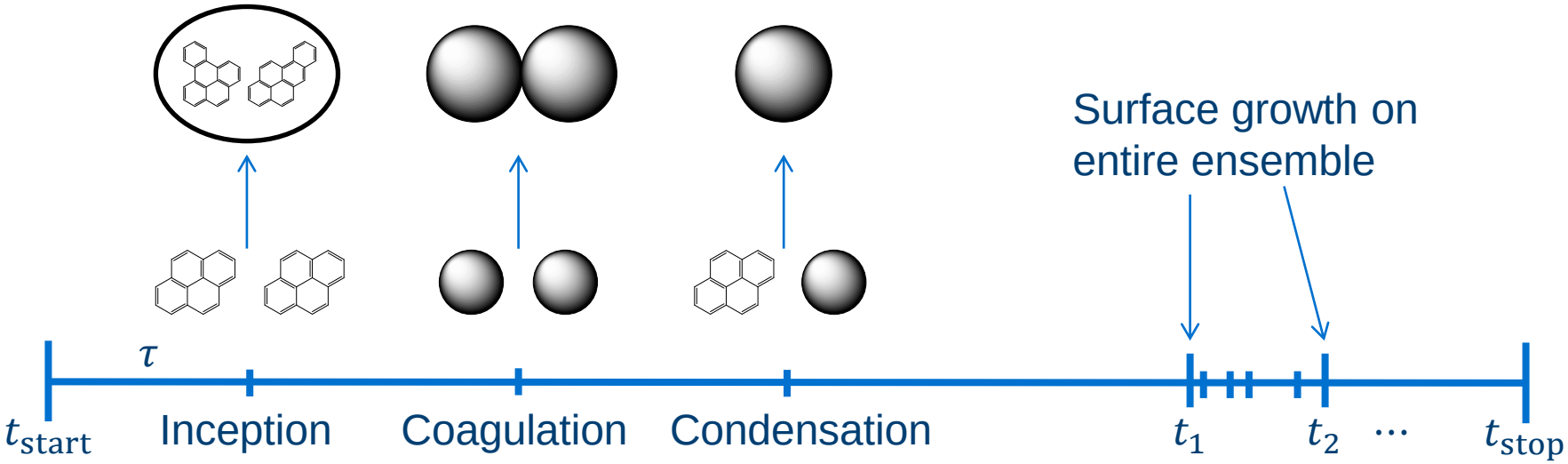


Mean primary diameter

CoMo GROUP


START
Select two particles for coagulation
Calculate majorant coagulation rate
Perform LPDA and sintering updates on particles
Calculate true coagulation rate
Direct Simulation Algorithm
Add particle 2 to particle 1
Delete particle 2
Is N(t) < 0.5 * N_max?
Yes
Duplicate ensemble
No
No (DSA)
Is the jump fictitious?
No (WPM)
Yes
Weighted Particle Method
Adjust particle 1's weight
Add particle 2 to particle 1 (leave particle 2 unchanged)
Update ensemble statistics
END

- Nonlinear processes cannot be neglected but are dominated in rate by linear ones

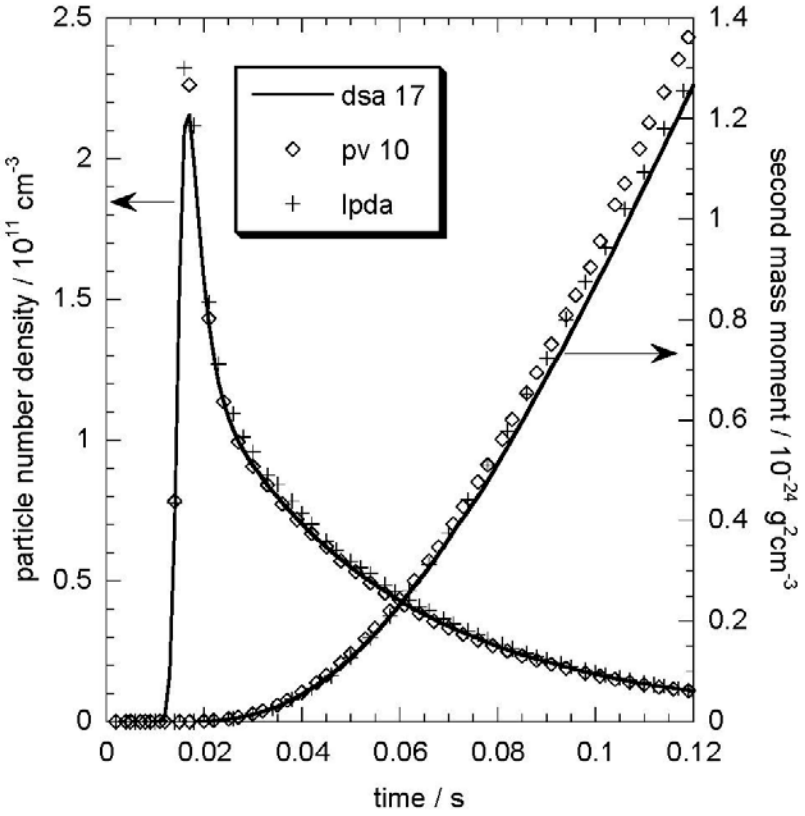


R.I.A. Patterson, J. Singh, M. Balthasar, M. Kraft, J.R. Norris, SIAM J. Sci. Comput. 28 (2006) 303–320

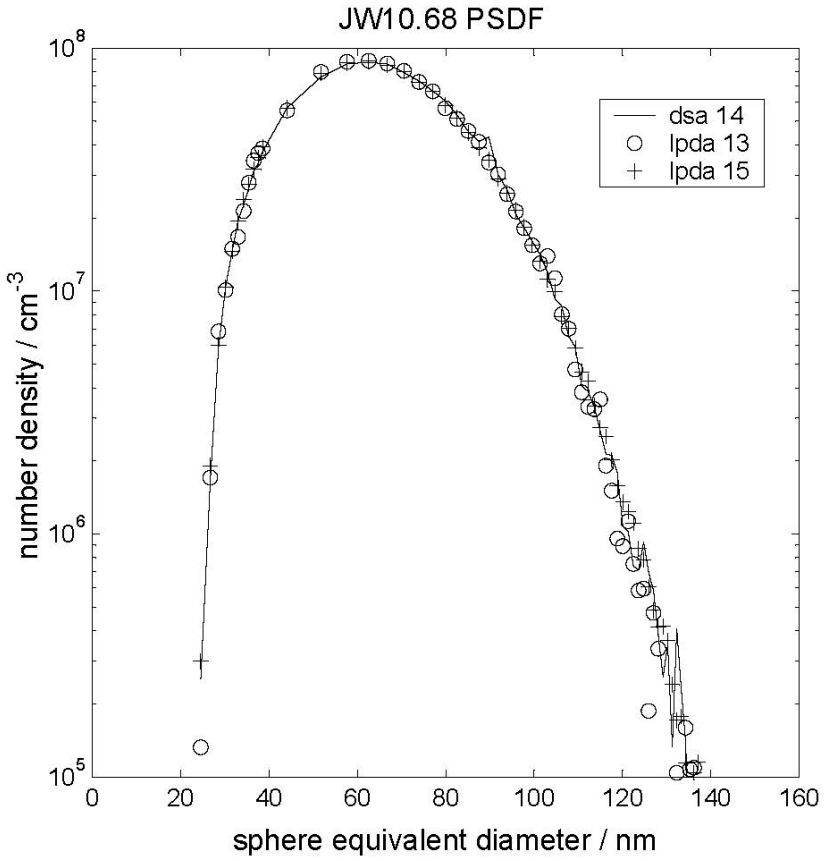
CoMo GROUP

- Soot formation in laminar premixed ethylene flames

Flame no.	Pressure (bar)	C/O	Cold gas velocity (cm/s)	T_{max}
1	10	0.68	6.0	1880
4	1	0.69	5.9	1711



	Number of tree levels	Time (mins)
DSA	13	50
LPDA	13	4.2



	Number of tree levels	Time (mins)
DSA	13	8497
LPDA	13	6.33

- The total jump rate R for coagulation is:

$$R = \frac{1}{2}\sum_{i\neq j}^{N(t)} \frac{K(x_i, x_j)}{V_{\mathrm{smp}}}$$

- The transition kernel is a popular choice:

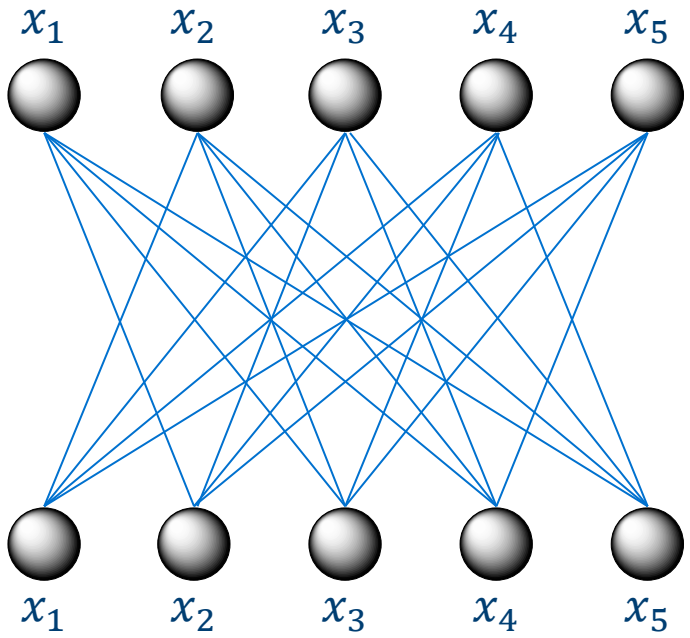
$$K^{\mathrm{tr}}(x_i, x_j) = \left(\frac{1}{K^{\mathrm{fm}}(x_i, x_j)} + \frac{1}{K^{\mathrm{sf}}(x_i, x_j)}\right)^{-1}$$

- Repeated selection of pairs of particles $x_i, x_j, i \neq j$ from a population $\{x_1, x_2, \cdots, x_n\}$
- Selection done so that the probability of a pair being selected is:

$$\frac{K(x_i, x_j)}{R}, R = \sum_{i \neq j}^{N(t)} K(x_i, x_j)$$

- Particles of size x_i and x_j are removed and a particle of size $x_i + x_j$ is added

- Generation of the joint probability distribution of i and j requires summing over $O(N^2)$ terms
- Run time proportional to N^2



- A majorant is a function $\widehat{K} \geq K$ for which the computational of

$$\hat{R} = \sum_{i \neq j}^{N(t)} \widehat{K}(x_i, x_j)$$

and the particle selection of the distribution

$$\frac{\widehat{K}(x_i, x_j)}{\hat{R}}$$

are relatively fast.

A. Eibeck, W. Wagner, SIAM J. Sci. Comput. 22 (2000) 802–821

CoMo GROUP

- Possible to recover the distribution and rate defined by K by rejecting the selection of a pair of particles x_i, x_j with probability:

$$1 - \frac{K(x_i, x_j)}{\widehat{K}(x_i, x_j)}$$

- The full free-molecular kernel:

$$K^{\mathrm{fm}}(x_i, x_j) = 2.2\left(\frac{\pi k_B T}{2}\right)^{\frac{1}{2}}\left(\frac{1}{m_i} + \frac{1}{m_j}\right)^{\frac{1}{2}}(d_i + d_j)^2$$

- The free-molecular majorant kernel:

$$\widehat{K}^{\mathrm{fm}}(x_i, x_j) = 2.2k_{\mathrm{maj}}\left(\frac{\pi k_B T}{2}\right)^{\frac{1}{2}}\left(\frac{1}{\sqrt{m_i}} + \frac{1}{\sqrt{m_j}}\right)(d_i^2 + d_j^2)$$

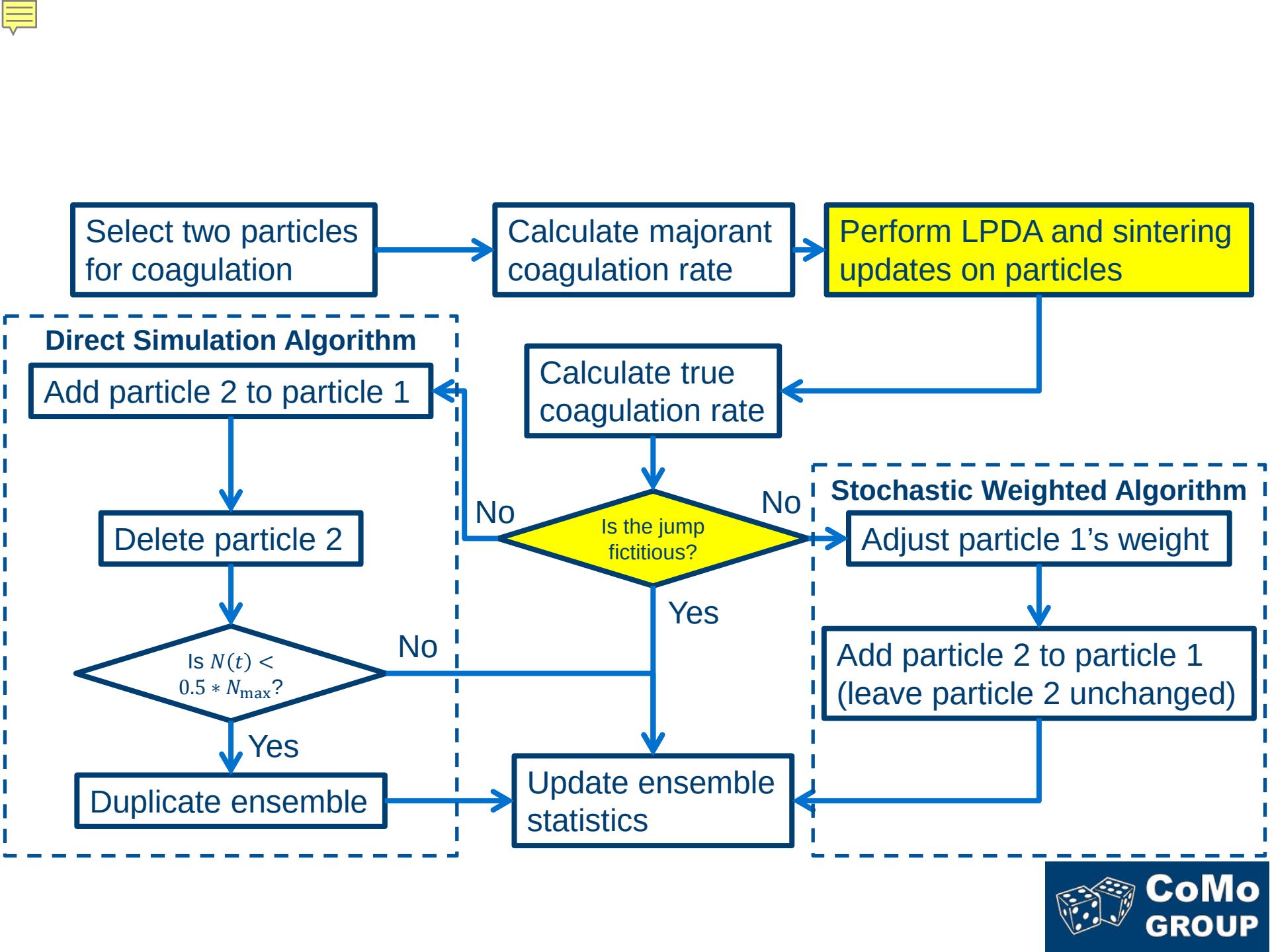
- The total majorant rate:

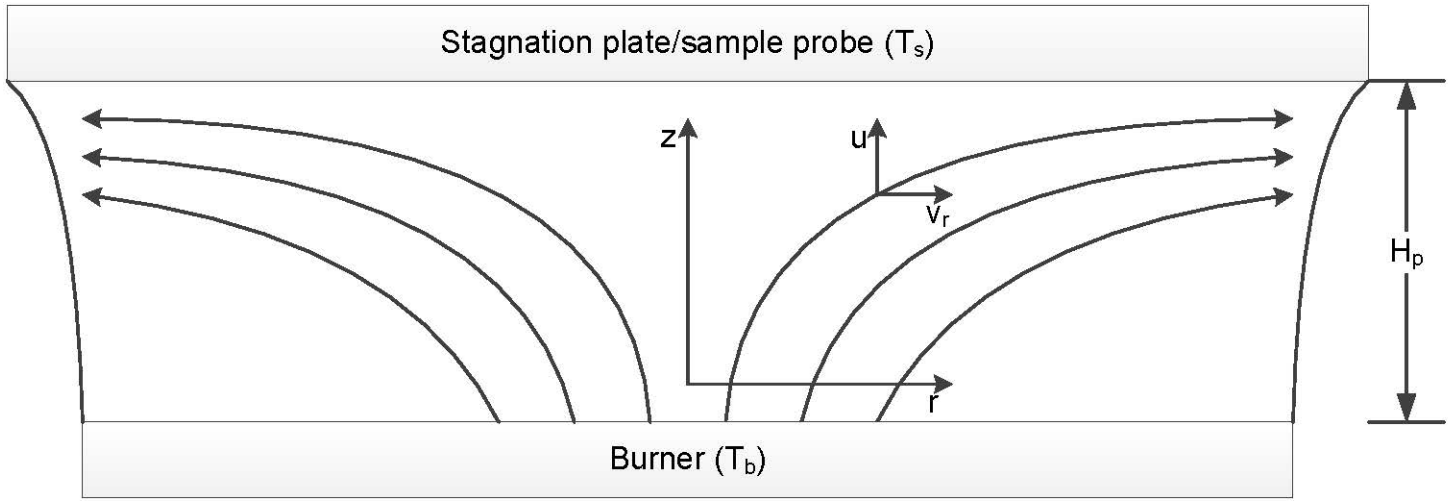
$$\sum_{i\neq j}^{N(t)} \hat{K}^{\mathrm{fm}}(x_i, x_j) = 2.2 k_{\mathrm{maj}} \left(\frac{\pi k_B T}{2}\right)^{\frac{1}{2}} \left\{ (N(t)-1) \sum_{i=1}^{N(t)} d_i^2 m_i^{-\frac{1}{2}} + \sum_{i=1}^{N(t)} m_i^{-\frac{1}{2}} \sum_{i=1}^{N(t)} d^2 + \sum_{i=1}^{N(t)} d_i^2 m_i^{-\frac{1}{2}} \right\}$$

- Independent generation of the indices i and j reduces the computational time to $O(N \log N)$

A. Eibeck, W. Wagner, SIAM J. Sci. Comput. 22 (2000) 802–821

CoMo GROUP

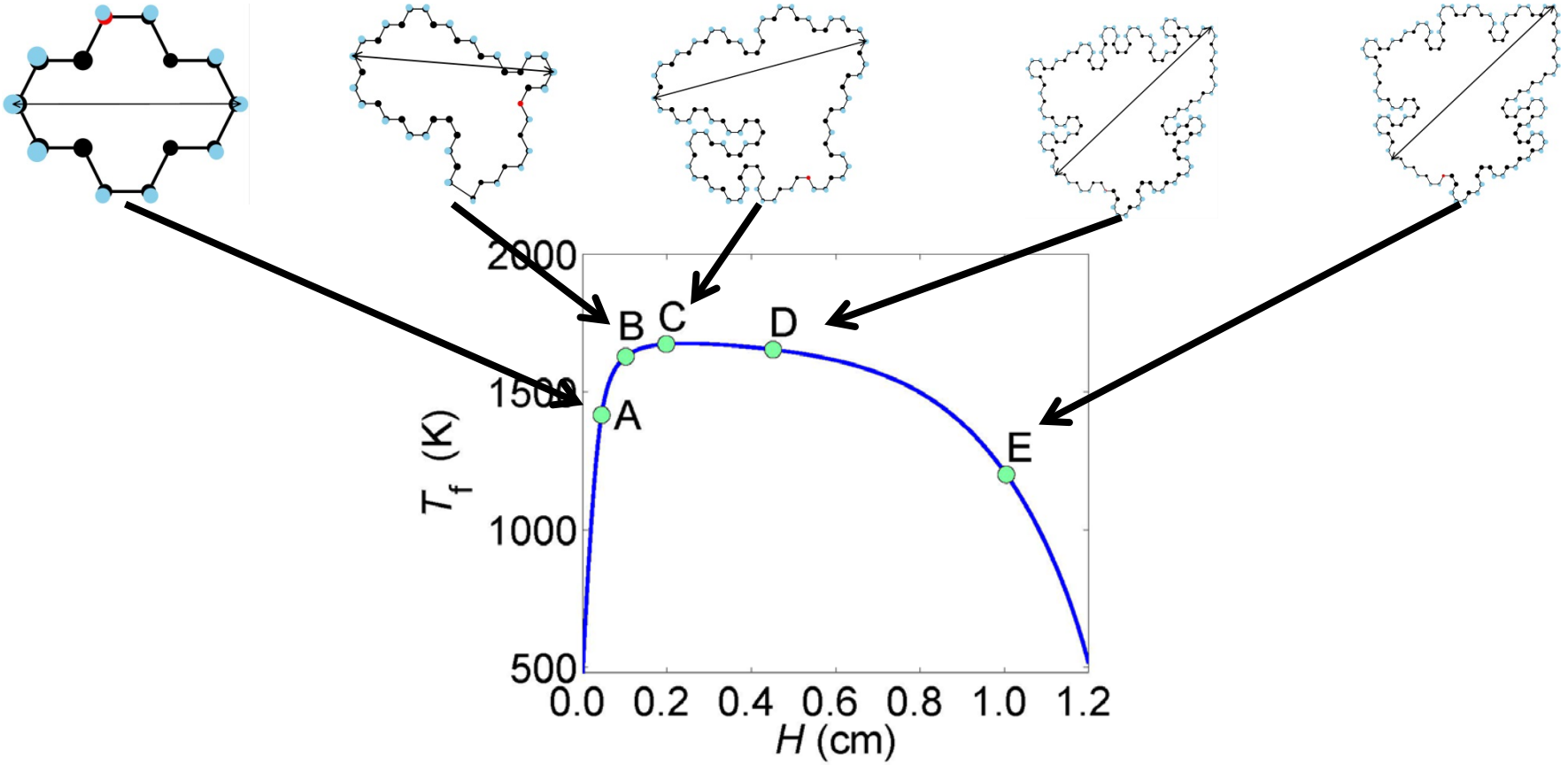

Select two particles for coagulation
Calculate majorant coagulation rate
Perform LPDA and sintering updates on particles
Direct Simulation Algorithm
Add particle 2 to particle 1
Delete particle 2
Is $N(t) < 0.5 * N_{max}$?
No
Yes
Duplicate ensemble
Calculate true coagulation rate
No
Is the jump fictitious?
No
Yes
Stochastic Weighted Algorithm
Adjust particle 1's weight
Add particle 2 to particle 1 (leave particle 2 unchanged)
Update ensemble statistics



- Sample probe integrated into plate
- Temperature profiles and PSDs measured by Wang & co-workers

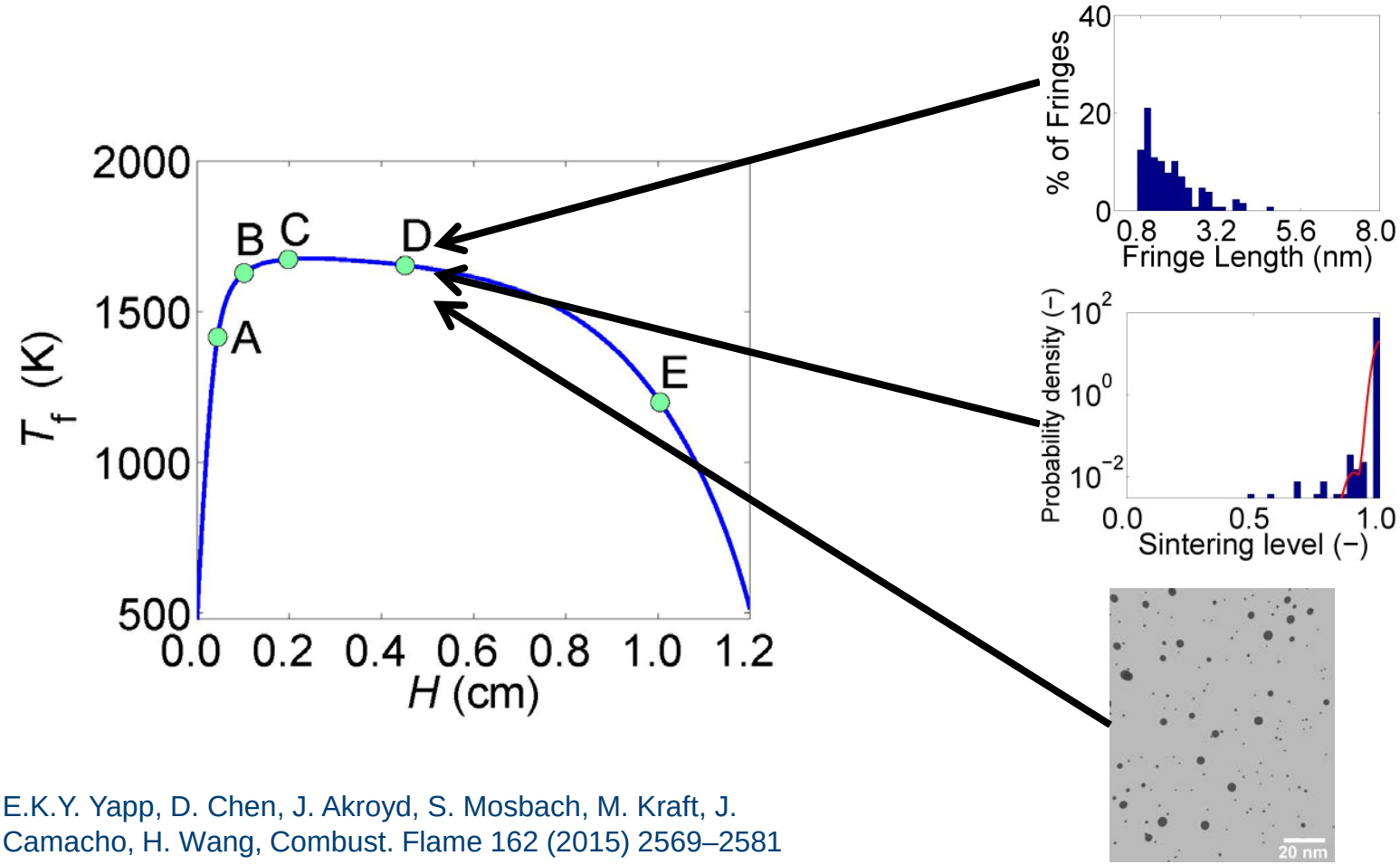
A.D. Abid, J. Camacho, D.A. Sheen, H. Wang, Combust. Flame 156 (2009) 1862–1870

CoMo GROUP



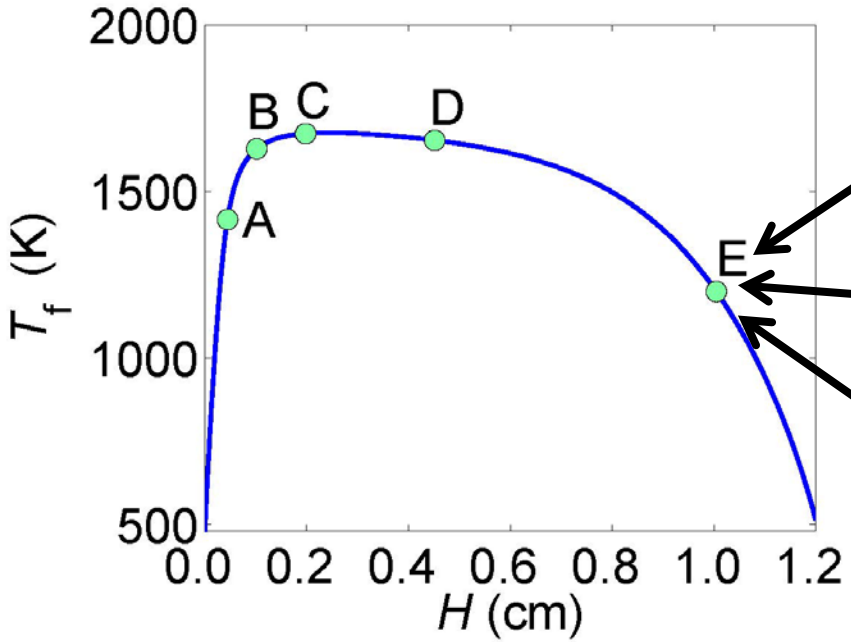
E.K.Y. Yapp, D. Chen, J. Akroyd, S. Mosbach, M. Kraft, J. Camacho, H. Wang, Combust. Flame 162 (2015) 2569–2581

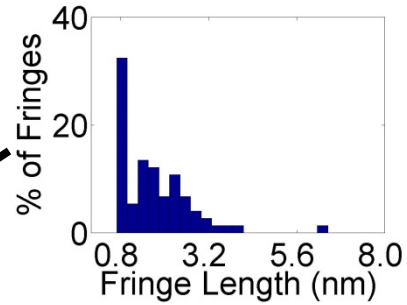
CoMo GROUP

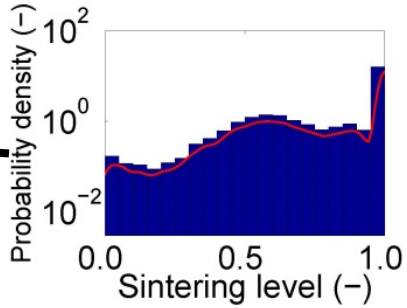


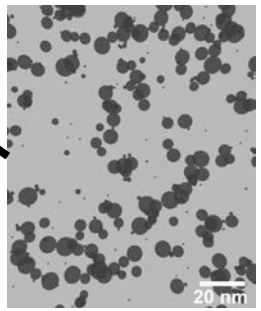
E.K.Y. Yapp, D. Chen, J. Akroyd, S. Mosbach, M. Kraft, J. Camacho, H. Wang, Combust. Flame 162 (2015) 2569–2581

CoMo GROUP









E.K.Y. Yapp, D. Chen, J. Akroyd, S. Mosbach, M. Kraft, J. Camacho, H. Wang, Combust. Flame 162 (2015) 2569–2581

CoMo GROUP

CoMo GROUP

PRIME MINISTER'S OFFICE
SINGAPORE

CREATE
Campus for Research Excellence And Technological Enterprise

WIAS

THE ROYAL SOCIETY

EPSRC
Engineering and Physical Sciences Research Council

- Detailed population balance model describes the evolution of aggregates composed of primary particles which are in turn composed of individual polycyclic aromatic hydrocarbons
- Particles are subjected to inception, coagulation, condensation, sintering and a detailed set of surface growth jump processes
- Gas-phase chemistry and particle processes may be coupled through an operator splitting technique
- The simplest stochastic particle method for kinetic population balance equations is the direct simulation Monte Carlo method but has its flaws

CoMo GROUP

- Stochastic weighted algorithm is shown to offer computational advantage over direct simulation algorithm in situations where interest is focused on the larger particles in a system
- Where nonlinear processes cannot be neglected but are dominated in rate by linear ones, deferment of the linear processes is shown to reduce run time by a factor of up to 1000
- Replacement of the actual coagulation kernel by a majorant kernel reduces run time proportionality from N^2 to N
- The detailed particle model was applied to a burner-stabilised stagnation flame and used to model quantities which may be directly compared with experiments

CoMo GROUP