Revolutionizing Catalysis Industry through Automated Multiscale Modeling and Active Exploration of Chemical Space

Nestor F. Aguirre
Scientific Software Developer
aguirre@scm.com
Background: SCM, ADF & AMS

- AMOL/ADF: first DFT code for chemistry (1970s)
  Baerends@VU (>’73), Ziegler@Calgary(+) (>’75)

- 1980s: Mitsui, Shell, Akzo, Unilever: training@Amsterdam
  - Tom Ziegler => Ziegler-Natta catalysts, hydroformulation
  - BAND for periodic systems

- SCM: Spin-off company 1995
  - To keep supporting users

- 2010s: DFTB, ReaxFF, COSMO-RS (Albemarle, DSM)

- 2019: Multi-scale: ReaxPro (BASF, Dow, Shell, JM)

- 2022: 24 people (19 senior PhD’s) + 3 EU fellows
  Many academic collaborators & EU networks
  SCM: development, debug, port, optimize, & support

Papers with “DFT+chemistry”
The SCM team celebrating 25 years (2020)
Amsterdam Modeling Suite

- Integrated GUI
- Python scripting (workflows)
- Tools: parametrization, reaction mapping & analysis
- AMSdriver: MD, MC, PES exploration
  - Hybrid: multi-layer, QM/MM, QM/QM'
  - Automated reaction
  - Network mapping
  - Analysis, properties
Joint Code Development

• We collaborate with academic groups developing code.

• They provide code and/or know-how (algorithms, test sets, etc).

• We focus on usability (e.g. GUI), performance (optimization, parallelization, etc.), long-term maintenance and support.

• We integrate and distribute code.

• Academic developers get increased visibility and outreach, citations, software licenses, etc.

• And they keep their intellectual property.
AMS driver is a powerful central tool in the Amsterdam Modeling Suite for complex potential energy tasks, such as molecular dynamics, Monte Carlo, PES scans, and finding transition states.

The AMS driver provides efficient and reliable implementations for common tasks like geometry optimization or transition state searches as well as more advanced modeling and simulation options like Grand Canonical Monte Carlo, force bias Monte Carlo, and the molecule gun.
Revolutionizing Catalysis Industry through Automated Multiscale Modeling and Active Exploration of Chemical Space

Nestor F. Aguirre
Scientific Software Developer
aguirre@scm.com
The ReaxPro Project

http://www.reaxpro.eu
AMS driver is a powerful tool for potential energy tasks, such as molecular dynamics, Monte Carlo, and finding transition states with our atomistic modules (engines) such as ADF, BAND, DFTB, MOPAC, ReaxFF, Machine Learning Potentials, and Force Field.

https://www.scm.com/product/ams/
The EON software package contains a set of algorithms used primarily to model the evolution of atomic scale systems over long time scales. The algorithms currently implemented include parallel replica dynamics, hyperdyamics, adaptive kinetic Monte Carlo, and basin hopping. The EON software is a collaboration between the Henkelman and Jónsson research groups.

https://theory.cm.utexas.edu/eon/

Zacros is a Kinetic Monte Carlo (KMC) software package, for simulating molecular phenomena on catalytic surfaces. Zacros enables to perform dynamic modeling of adsorption, desorption, surface diffusion, and reaction processes on heterogeneous catalysts. Zacros is developed by a team of scientists at UCL, led by Dr Michail Stamatakis,

https://zacros.org/

CatalyticFOAM allows for the solution of Navier-Stokes equations for complex and general geometries for reacting flows at surfaces, based on microkinetic descriptions of the surface reactivity. CatalyticFOAM has been developed in the Multiscale Catalysis Group of the Laboratory of Catalysis and Catalytic Processes of Politecnico di Milano.

https://www.catalyticfoam.polimi.it/
https://github.com/multiscale-catalysis-polimi/catalyticFoam

The ReaxPro Project

http://www.reaxpro.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the grant agreement No 814416
This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the grant agreement No 814416
The ReaxPro Project

http://www.reaxpro.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the grant agreement No 814416.
The ReaxPro Project

http://www.reaxpro.eu

This project has received funding from the European Union’s Horizon 2020 research and innovation programme under the grant agreement No 814416
Automated Exploration of Energy Landscapes

Determination of accurate rate coefficients remains a key challenge for a wide variety of chemical research areas.

Main problems:

- The complexity of mapping the possible reaction channels.
- The traditionally used strategies often incorporate several manual actions and need chemical expert knowledge.
- It is expensive to calculate accurate ab initio parameters (typically energies, frequencies, and geometries)

It is necessary to create tools that automatically search for reaction pathways and calculate the rate coefficients.

The reaction path network for the HCo(CO)$_3$ + CO + H$_2$ + C$_2$H$_4$ system obtained by the SC-AFIR search. 2601 local minima.

Figure taken from: https://www.icredd.hokudai.ac.jp/research/1591asdasd
Hatanaka, M., Yoshimura, T., Maeda, S.
New Directions in the Modeling of Organometallic Reactions 2020, 57-80
Automated Exploration of Energy Landscapes for Catalytic Processes

The reaction path network for CO oxidation on Pt(111).

CO + O₂ → CO₂
139 MIN + 572 TS

Additional problems:

- The structural variety and size of the systems
- The vast amount of small transformation steps that need to be considered.
- Structural changes of the catalyst occur during the reaction.
- Structural rearrangements during reactions on surfaces significantly differ from their gas-phase counterparts.
PESEExploration in AMS

PESEExploration

Job ProcessSearch
NumExpeditions 25
NumExplorers 3
End

EON Jobs available from AMS:
- ProcessSearch
- BasinHopping
- SaddleSearch
- NudgedElasticBand
- MolecularDynamics
- ParallelReplica

Jobs AMS original:
- LandscapeRefinement
- BindingSites

ProcessSearch uses the dimer method (min-mode following) for finding nearby saddle points without knowledge of the final or transition state. The method only makes use of first derivatives of the potential energy.

PESEExploration in AMS

PESEExploration
Job ProcessSearch
DynamicSeedStates T
NumExpeditions 25
NumExplorers 3
CalculateFragments T
End
PESEExploration

Job ProcessSearch
DynamicSeedStates T
NumExpeditions 25
NumExplorers 3
CalculateFragments T
End
PESEExploration in AMS

PESEExploration
Job ProcessSearch
DynamicSeedStates T
NumExpeditions 25
NumExplorers 3
CalculateFragments T
End

Energy Landscape

MIN1
MIN2
MIN3
TS1
TS2
FS1
Isomerization of CHNO

Task PESEExploration

System
Atoms
H  -0.58691625  1.79642617  0.79110081
O  -0.53563869  1.37408080 -0.08531432
N  -0.54146683 -0.09761965  0.18986664
C   0.58077340  0.40111238 -0.05052313
End

PESEExploration
  RandomSeed 100
  Job ProcessSearch
  NumExpeditions 500
  NumExplorers 4
End

Engine MOPAC
EndEngine
Isomerization of CHNO

Splitting Water on rutile TiO$_2$(110)


Splitting Water on Metal Oxide Surfaces


Engine: ReaxFF (TiO2bio.ff)
Binding Sites

A binding site is a small fraction of the catalyst surface which is catalytically active and might be composed of an atom or an ensemble of atoms. They are generally situated at surface defects such as corners, edges, and other crystalline discontinuities.
Binding Sites: Case Co₈ +Co

mol = scm.plams.Molecule('CO-Pt111.xyz')

sett = scm.plams.Settings()
sett.input.ams.Task = 'PESExploration'
sett.input.ams.PESExploration.Job = 'ProcessSearch'
sett.input.ams.PESExploration.NumExpeditions = 10
sett.input.ams.PESExploration.NumExplorers = 4

job = scm.plams.AMSJob(molecule=molO, settings=sett)
results = job.run()

for state in results.get_energy_landscape():
    print("Energy = ", state.energy)
    print("Geometry = ", state.molecule)
    if state.isTS:
        print("Forward barrier = ", state.energy - state.reactants.energy)
        print("Backward barrier = ", state.energy - state.products.energy)

...
PLAMS: python interface
https://github.com/SCM-NV/PLAMS

Single-Body Mechanism

- Adsorption:
  - PESExploration
  - ProcessSearch
  - +DynamicSeedStates = F
  - +CheckSymmetry = T
  - +Restraints

- First-Neighbor Reactions:
  - PESExploration
  - ProcessSearch
  - +DynamicSeedStates = F
  - +CheckSymmetry = T

- Lattice:
  - PESExploration
  - LandscapeRefinement
  - +GenerateSymmetryImages = T
  - +RelaxFromSaddlePoint = T
  - +CheckSymmetry = F
AMS workflows

https://github.com/SCM-NV/PLAMS

2024?
PLAMS: python interface
https://github.com/SCM-NV/PLAMS
Single-Body Mechanism for O$_2$-Pt(111)

Irreducible-Symmetry Energy Landscape

Engine: ReaxFF (CHONSFPtClNi.fff)
Zacros enables to perform dynamical modeling of adsorption, desorption, surface diffusion, and reaction processes on heterogeneous catalysts.
This project has received funding from the European Union's Horizon 2020 research and innovation programme under the grant agreement No 814416
pyZacros is a collection of tools that aims to provide a powerful, flexible, and easily extendable Python interface to Zacros. It is designed as an extension of the python library PLAMS.

pyZacros inherits from PLAMS the robust way of managing the inputs file preparation, job execution, file management, and output file processing. Above and above that, it also offers the possibility of postprocessing the results and building very advanced data workflows.

Base components

Extended components
```python
loader = scm.pyzacros.RKFLoader( results_ads )
loader.lattice.set_repeat_cell( (10,10) )

settings = scm.pyzacros.Settings()
settings.temperature = 273.15
settings.pressure = 1.01325
settings.molar_fraction.CO = 0.1

dt = 1e-8
settings.max_time = 1000*dt
settings.snapshots = ('logtime', dt, 3.5)
settings.species_numbers = ('time', dt)

job = scm.pyzacros.ZacrosJob( lattice=loader.lattice, mechanism=loader.mechanism,
                               cluster_expansion=loader.clusterExpansion, settings=settings )
results_pz = job.run()
results_pz.plot_lattice_states( results_pz.lattice_states() )
```
PyZacros: Graphical User Interface
Poisoning of Pt(111) by CO

https://www.scm.com/doc/pyzacros/examples/COPt111.html

Engine: ReaxFF (CHONSFPtClNi.fff)
Poisoning of Pt(111) by CO
https://www.scm.com/doc/pyzacros/examples/COPt111.html

CO Adsorption

\[ \text{CO(g)} + \overset{*}{\text{Pt}} \rightleftharpoons \overset{*}{\text{CO}} \]

CO Diffusion

\[ \overset{*}{\text{CO}} \rightleftharpoons \overset{*}{\text{Pt}} \rightarrow \overset{*}{\text{CO}} \]

Engine: ReaxFF (CHONSFPtClNi.fff)
PyZacros Examples
https://github.com/SCM-NV/pyZacros

Simple examples

- Water-gas shift reaction on Pt(111)
- Ziff-Gulari-Barshad model

Intermediate examples

- Ziff-Gulari-Barshad model: Phase Transitions
- Ziff-Gulari-Barshad model: Steady State Conditions

Advanced examples

- Poisoning of Pt(111) by CO: From atomistic to mesoscopic modeling
- Ziff-Gulari-Barshad model: Phase Transitions and ML-based Surrogate Model
- Ziff-Gulari-Barshad model: Phase Transitions under Steady State Conditions
- Langmuir-Hinshelwood model: Acceleration by Automated Rescaling of the Rate Constants
PyZacros Examples
https://github.com/SCM-NV/pyZacros

Simple examples
- Water-gas shift reaction on Pt(111)

Advanced examples
- Poisoning of Pt(111) by CO: From atomistic to mesoscopic modeling

Ziff-Gulari-Barshad model:
Phase Transitions and ML-based Surrogate Model
CO oxidation on Pt(111)

CatalyticFOAM allows for the solution of Navier-Stokes equations for complex and general geometries for reacting flows at surfaces, based on microkinetic descriptions of the surface reactivity.

T = 673.15 K, P = 1 atm, gas-velocity 0.1 m/s
CO oxidation on Pt(111)

CatalyticFOAM allows for the solution of Navier-Stokes equations for complex and general geometries for reacting flows at surfaces, based on microkinetic descriptions of the surface reactivity.

T = 673.15 K, P = 1 atm, gas-velocity 0.1 m/s
The ReaxPro Project

http://www.reaxpro.eu

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the grant agreement No 814416
CO(gas) + \( \frac{1}{2} \) O\(_2\) (gas) \rightarrow CO\(_2\) (gas)

Pablo Lopez-Tarifa
Matthias Büschelberger
Joana Francisco Morgado
This project has received funding from the European Union's Horizon 2020 research and innovation programme under the grant agreement No 814426.

This document and all information contained herein is in the sole property of the ReaxPro Consortium. It may contain information subject to intellectual property rights. No intellectual property rights are granted by the delivery of this document or the disclosure of its content. Reproduction or circulation of this document to any third party is prohibited without the consent of the author(s).

The statement made herein do not necessarily have the consent or agreement of the ReaxPro consortium and represent the opinion and findings of the author(s).
Thanks!