AFRL-AFOSR-VA-TR-2018-0361

Next-generation Chemical-kinetics, Transport, and Reacting-flow Software Tools

Robert Kee
COLORADO SCHOOL OF MINES
1500 ILLINOIS ST
GOLDEN, CO 80401-1887

10/04/2018
Final Report

Air Force Research Laboratory
AF Office Of Scientific Research (AFOSR)/RTA1
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From - To)*
10/08/2018	Final Report	01-09-2016 - 31-08-2018

4. TITLE AND SUBTITLE
Next-generation chemical-kinetics, transport, and reacting-flow software tools

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA9550-16-1-0349,

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Kee, Robert J.; DeCaluwe, Steven C.; Jackson, Gregory S.

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Colorado School of Mines
1600 Illinois St.
Golden, CO 80401

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Air Force Office of Scientific Research
AFOSR/RTE
875 N Randolph St., Ste. 325
Arlington, VA 22203
Dr. Chiping Li

10. SPONSOR/MONITOR'S ACRONYM(S)
AFOSR

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution A -- Approved for Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
The purpose of this grant was to develop and articulate a vision for designing and implementing advanced chemically reacting flow modeling software. The focus is on chemistry and transport in three important technology areas: 1) high pressure gas-phase processes, 2) heterogeneous catalysis, and 3) electrochemistry. The general thrust is concerned with energy-conversion processes, such as high-pressure combustion, catalytic fuel conversion, fuel cells, and batteries. The results of the study are communicated in three documents: 1) A roadmap to guide future development, 2) A proposal to bring together a nationwide implementation team, and 3) An archival paper that articulates a vision for electrochemistry.

15. SUBJECT TERMS
Modeling software, high-pressure combustion, heterogeneous catalysis, electrochemistry, open-source software, object-oriented software, Development roadmap

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			Robert J. Kee
U	U	U	UU	99	**19b. TELEPHONE NUMBER** *(Include area code)* (303) 273-3379

INSTRUCTIONS FOR COMPLETING SF 298

1. REPORT DATE. Full publication date, including day, month, if available. Must cite at least the year and be Year 2000 compliant, e.g. 30-06-1998; xx-06-1998; xx-xx-1998.

2. REPORT TYPE. State the type of report, such as final, technical, interim, memorandum, master's thesis, progress, quarterly, research, special, group study, etc.

3. DATE COVERED. Indicate the time during which the work was performed and the report was written, e.g., Jun 1997 - Jun 1998; 1-10 Jun 1996; May - Nov 1998; Nov 1998.

4. TITLE. Enter title and subtitle with volume number and part number, if applicable. On classified documents, enter the title classification in parentheses.

5a. CONTRACT NUMBER. Enter all contract numbers as they appear in the report, e.g. F33315-86-C-5169.

5b. GRANT NUMBER. Enter all grant numbers as they appear in the report. e.g. AFOSR-82-1234.

5c. PROGRAM ELEMENT NUMBER. Enter all program element numbers as they appear in the report, e.g. 61101A.

5e. TASK NUMBER. Enter all task numbers as they appear in the report, e.g. 05; RF0330201; T4112.

5f. WORK UNIT NUMBER. Enter all work unit numbers as they appear in the report, e.g. 001; AFAPL30480105.

6. AUTHOR(S). Enter name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. The form of entry is the last name, first name, middle initial, and additional qualifiers separated by commas, e.g. Smith, Richard, J, Jr.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES). Self-explanatory.

8. PERFORMING ORGANIZATION REPORT NUMBER. Enter all unique alphanumeric report numbers assigned by the performing organization, e.g. BRL-1234; AFWL-TR-85-4017-Vol-21-PT-2.

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES). Enter the name and address of the organization(s) financially responsible for and monitoring the work.

10. SPONSOR/MONITOR'S ACRONYM(S). Enter, if available, e.g. BRL, ARDEC, NADC.

11. SPONSOR/MONITOR'S REPORT NUMBER(S). Enter report number as assigned by the sponsoring/ monitoring agency, if available, e.g. BRL-TR-829; -215.

12. DISTRIBUTION/AVAILABILITY STATEMENT. Use agency-mandated availability statements to indicate the public availability or distribution limitations of the report. If additional limitations/ restrictions or special markings are indicated, follow agency authorization procedures, e.g. RD/FRD, PROPIN, ITAR, etc. Include copyright information.

13. SUPPLEMENTARY NOTES. Enter information not included elsewhere such as: prepared in cooperation with; translation of; report supersedes; old edition number, etc.

14. ABSTRACT. A brief (approximately 200 words) factual summary of the most significant information.

15. SUBJECT TERMS. Key words or phrases identifying major concepts in the report.

16. SECURITY CLASSIFICATION. Enter security classification in accordance with security classification regulations, e.g. U, C, S, etc. If this form contains classified information, stamp classification level on the top and bottom of this page.

17. LIMITATION OF ABSTRACT. This block must be completed to assign a distribution limitation to the abstract. Enter UU (Unclassified Unlimited) or SAR (Same as Report). An entry in this block is necessary if the abstract is to be limited.

Final Report
AFOSR Grant FA9550-16-1-0349

Next-generation chemical-kinetics, transport, and reacting-flow software tools

Robert J. Kee, Steven DeCaluwe, and Gregory S. Jackson
Mechanical Engineering Dept.
Colorado School of Mines
Golden, CO 80401

rjkee@mines.edu
(303) 273-3379

AFOSR Program Manager
Dr. Chiping Li

Abstract

The purpose of this grant was to develop and articulate a vision for designing and implementing advanced chemically reacting flow modeling software. The effort focuses on chemistry and transport for three important technology areas: 1) high pressure (including supercritical) processes, 2) heterogeneous catalysis, and 3) electrochemistry. The general thrust is concerned with energy-conversion processes, such as high-pressure combustion, catalytic fuel conversion, fuel cells, and batteries. The results of the study are communicated in three documents: 1) A roadmap to guide future software development, 2) A proposal to bring together a nationwide team to implement the roadmap, and 3) An archival paper that articulates the vision for electrochemistry modeling. The vision discusses the development and maintenance of user-accessible, open-source, software. The software implementation will be based on extensions of Cantera. Cantera, initially designed and developed by David Goodwin (Caltech), is an object-oriented, open-source, framework that serves as an excellent foundation for further development. The anticipated outcome is the present planning project is the initiation of future programs to implement the vision.

Report content

This final report is comprised of three sections, each documenting aspects of the program accomplishments.

1. A planning Roadmap that articulates and discusses future technical needs and opportunities.
2. A submitted archival paper that documents a vision for future software focused on electrochemical transport and kinetics.
3. A draft proposal to bring together a nationwide group of experts with the objective of implementing and maintaining advanced modeling software.

Roadmap: Next-generation chemical-kinetics, transport, and reacting-flow software tools

Robert J. Kee[a,*], Steven C. DeCaluwe[a], Gregory S. Jackson[a], Huayang Zhu[a], Canan Karakaya[a], Peter Weddle[a], Benjamin Kee[a], Gandhali Kogekar[a]

[a]Mechanical Engineering, Colorado School of Mines, Golden, CO 80401, USA

Abstract

This project articulates a roadmap for developing next-generation, open-source software to model chemically reacting flow processes. The CHEMKIN software, which was developed initially in the 1980s, is still the leading software package worldwide. However, advances in the scientific underpinnings and object-oriented programming create opportunities to substantially further the state of the art in widely available, open-source software as exemplified by CANTERA. New initiatives proposed herein are based on extending capabilities beyond those of the current-day CANTERA software. This roadmap exercise gathers and evaluates input from existing and potential user communities. In broad terms, the initial scope addresses new capabilities in modeling non-ideal chemical processes (e.g., high-pressure or supercritical combustion), heterogeneous catalysis (e.g., complex bifunctional catalysts), and electrochemistry (e.g., batteries and fuel cells). In addition to new capabilities, the roadmap addresses usability (e.g., documentation, user interfaces), extensibility (e.g., incorporating new chemical theories), compatibility (e.g., interfacing with ab initio models and computational fluid dynamics), and maintainability (e.g., version control, bug fixes, community support).

Contents

*Corresponding author. Tel: +1-303-273-3379.
Email address: rjkee@mines.edu (Robert J. Kee)

1. Introduction

Modeling and simulation play critical roles in the research, development, and deployment of chemical processes. Energy-conversion processes in the context of primary propulsion (e.g., combustion) or auxiliary power (e.g., batteries or fuel cells) are topics of particular interest to the Air Force. The objective of the present project is to identify critical needs and establish a *Roadmap* that can serve as a guide in the development of advanced modeling capabilities. The Roadmap's scope ranges from fundamental physical chemistry to software development and support. The present project focuses on long-term planning, setting the stage for implementation in future development and deployment projects. This report documents the first year of a two-year planning process.

By way of background, the widespread availability and use of CHEMKIN for over 30 years has made major contributions to enable the modeling of complex combustion processes and coupling into reactive flow simulations. The present project builds on the foundation of these successes, planning for the next generation of modeling tools with major advances in non-ideal chemical kinetics, heterogeneous catalysis, and electrochemistry.

The Roadmap is expected to lay the foundation for next-generation computational simulation software that enables fundamental scientific advances in thermodynamics, transport, chemical kinetics, and electrochemistry. The project considers three stakeholder groups and their diverse different needs. The stakeholders include:

- Scientists and engineers who develop and implement new theories and approaches, such as for combustion kinetics or electrochemistry,
- Software developers who design and write new applications, such as in chemically reacting computational fluid dynamics,
- Practitioners who apply models to assist fundamental research and technology development.

New software for chemically reacting processes will facilitate significant new scientific functionality. However, to be of great value to diverse scientific and engineering communities, the software must be designed and implemented to be understandable and flexible for use in a broad range of applications.

The planning is based on extending and supporting the CANTERA software (cf., Appendix A). The object-oriented approach and the core C++ kernel in CANTERA provide an excellent foundation for building and supporting next-generation software. Moreover, especially in the academic combustion community, there is already a substantial user base and active community support. Because CANTERA is *open-source* and will remain so, the planning must consider unique aspects of extending, distributing, maintaining, and supporting a large and complex software package in the open-source environment.

2. Roadmap structure

The Roadmap is intended to chart a course for substantially advancing the state of the art in reactive flow modeling. The planning is based on long-term experience in developing and applying high-performance modeling software (i.e., CHEMKIN) and on soliciting input from other stakeholders. Although advancing scientific capabilities is the paramount objective, other software-related objectives are important to successful long-term outcomes and impact.

For a variety of reasons, the CANTERA software forms the foundation onto which new capabilities will be designed, developed, and implemented. Developed initially by Prof. David Goodwin (Caltech), CANTERA is designed and implemented in C++ using object-oriented software paradigms. The long-term intent is to develop and maintain new capabilities as *open-source software*. Thus, important elements of the Roadmap are concerned with managing large, scientifically complex, software capabilities with community-based development and support. As Fig. 1 illustrates, the Roadmap is organized around five complementary elements — Capability, Usability, Extensibility, Compatibility, and Maintainability.

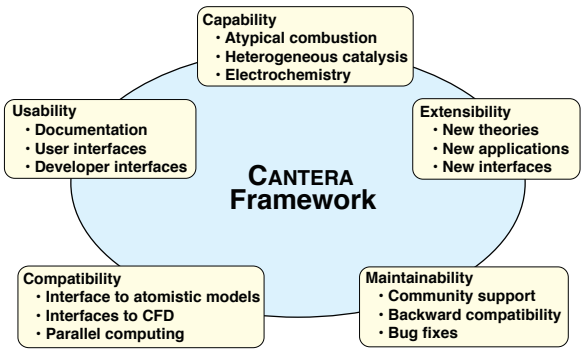

Figure 1: Contributing elements to the software Roadmap.

Accomplishing the Roadmap objectives will require financial support, which is expected to be partially from government sponsorship and partially from community service. As it stands today, CANTERA is community supported, mostly by academics, without specifically targeted government funds. Contributors are independently supported on research grants where modeling and software advances play central roles. This model has certainly advanced the state of the art. Nevertheless, managing and structuring large scientific efforts usually benefit from some degree of formal coordination, thus benefiting the broader user community.

2.1. Capability

Although there are numerous opportunities for extending modeling capabilities for reactive flow generally, the present planning concentrates on three areas that are expected to be priorities in energy conversion technologies. These are:

- Non-ideal behaviors associated with high pressure and supercritical flows,
- Heterogeneous catalysis with multifunctional catalysts and supports,
- Electrochemical processes that are relevant to topics such as rechargeable batteries and fuel cells.

The current capabilities in CANTERA (and CHEMKIN) are largely oriented around gas-phase chemistry and transport, with the dominant applications being in aspects of combustion research. Both CHEMKIN and CANTERA are concerned primarily with ideal-gas mixtures. However, CANTERA has incorporated some non-ideal-gas capabilities in the form of the Redlich–Kwong equation of state and associated thermodynamics.

Both CHEMKIN and CANTERA have capabilities to represent surface chemistry, such as in heterogeneous catalysis and chemical vapor deposition. However, these capabilities are based on mean-field approximations and are limited in their ability to handle complex multicomponent surfaces, such as bifunctional catalysts and chemically participating supports.

Electrochemistry plays essential roles in energy technologies such as batteries and fuel cells. Electrochemistry is also important in processes such as corrosion. CANTERA has some ability to handle electrochemistry, but was largely designed to support applications such as solid-oxide fuel cells. CHEMKIN simply does not consider electrochemistry. Thus, there are good reasons to significantly expand electrochemical modeling capabilities.

2.2. Usability

Extraordinary scientific capabilities do not assure widespread use of specific software tools. Software usability is concerned with making the capabilities widely accessible, in large part through comfortable user interfaces and clearly written documentation. An important objective of general-purpose open-source software is to enable development of new applications by users. Lowering user barriers to modify or extend existing capabilities accelerates the pace of new development. For example, numerous “applications” have been written to model different types of laboratory flames using the CHEMKIN or CANTERA software. Well documented, object-oriented, open-source software greatly facilitates writing new applications that follow logically from foregoing models. New applications typically involve the derivation of representative conservation equations and possibly solution algorithms, but do not involve new representations of the fundamental underpinning chemistry and transport capabilities.

2.3. Extensibility

As envisioned here, extensibility is understood to mean advancing the core software by incorporating new fundamental theories or alternative representations of existing models. This objective is conceptually different from writing new applications that use existing core capabilities. For example, an extension could involve reformulating chemical rate expressions to include processes such as non-thermalized transition states, or developing new equations of state to represent non-ideal

behaviors of supercritical fluids. Assuming good object-oriented software implementations, existing applications could make direct use of extended core capabilities. For example, an existing flame model could easily be used to study effects of supercritical circumstances, where ideal-gas approximations are no longer valid. Broadly speaking, writing new applications will be much more common than incorporating fundamentally new underpinning scientific concepts. Nevertheless, the new scientific concepts are essential components of next-generation chemical kinetics software, serving as a tool to incorporate, validate, and disseminate new theories and ideas.

2.4. Compatibility

In some sense, chemical kinetics software occupies a middle ground between fundamental *ab initio* chemistry on one hand and computational fluid dynamics (CFD) on the other hand. Thus, it must have sufficient compatibility to reach in both directions. The kinetics software should be capable of incorporating thermodynamic, transport, and chemistry information from fundamental models. At the same time, incorporating high-performance chemical kinetics models into complex CFD software is also a priority. For example, new chemistry models should be accessible from Direct Numerical Simulation (DNS) turbulence models. Practical combustor or chemical reactor design and development often uses geometrically complex CFD models. Most commercial CFD offerings already have Chemkin interfaces. The Cantera development must also be cognizant of the need to interface comfortably with CFD codes employed by academic and industrial user bases.

2.5. Maintainability

Cantera is already a large and complex body of software, which is expected to continue growing. Much of the growth will come as the result of contributions from scientifically and geographically diverse contributors. Some level of coordination and control is needed to maintain long-term coherence and viability of the capabilities. Currently, to a very large extent, the control and coordination is handled by Dr. Ray Speth (MIT), who is not directly compensated for his efforts. Additionally, there is group of approximately ten experts who contribute greatly by answering on-line questions from a much larger, and growing, group of users around the world. Not to diminish the great value of volunteer community support, but core aspects of the maintenance and coordination functions should be officially codified and supported. Of course, that requires some level of resource (including financial) commitment.

3. Non-ideal Thermochemistry

With the objective of improving efficiency, there is a trend in combustion technologies (e.g., gas turbines, diesel engines) to increase operating pressures. A variety of important chemical processes operate at very high pressure and low temperature (e.g., Haber–Bosch ammonia synthesis). Such processes may operate in trans-critical and supercritical regions. Thus, there is a growing need to develop predictive models that accurately represent non-ideal thermodynamics, transport, and chemistry phenomena. While current software tools (including Cantera and Chemkin) provide considerable support for gas-phase thermochemistry (e.g., homogeneous combustion), current models are concerned primarily with ideal-gas formulations. There is a major unmet need in chemical kinetics software for robust thermodynamic phase models that can accurately model thermodynamic properties and chemical kinetic rates across a wide array of non-ideal states.

3.1. Compressibility

The ideal-gas law, which is accurate at high temperatures and low pressures, assumes equal molar volumes for all species and no interaction potentials between molecules. However, thermodynamic properties near the critical point violate both assumptions and as such depart significantly from ideal-gas behavior. Deviation from ideal volumetric behavior is typically represented in terms of the compressibility factor

$$Z = \frac{pv}{RT}, \tag{1}$$

where p is the pressure (Pa), R the gas constant (J mol^{-1} K^{-1}), T the temperature (K), and v the molar volume (m^3 mol^{-1}).

Figure 2 is a generalized compressibility chart, which plots Z as a function of the reduced temperature T_r and pressure p_r,

$$T_\mathrm{r} = \frac{T}{T_\mathrm{c}}, \qquad p_\mathrm{r} = \frac{p}{p_\mathrm{c}}, \tag{2}$$

where T_c and p_c are the critical temperature and pressure, respectively.

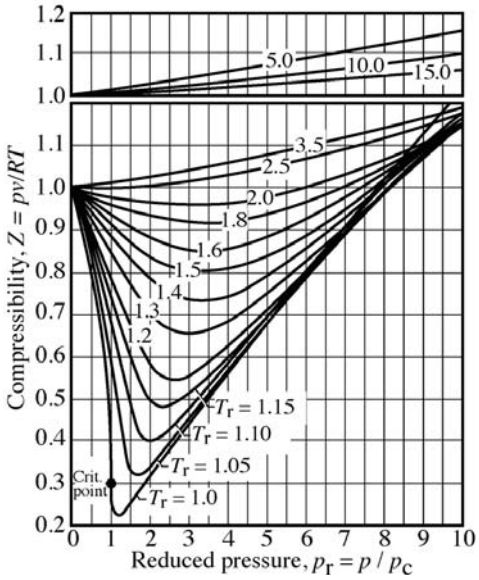

Figure 2: Generalized compressibility chart. The upper panel isolates $T_r > 3.5$ data, for ease of viewing.

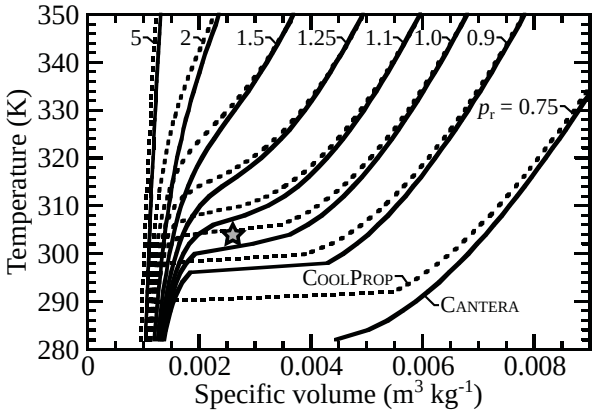

Figure 3: Comparison of predicted T–v relationships for CO_2 for a range of isobars at the specified reduced pressures. Comparisons between CoolProp and Cantera's R–K implementation demonstrate the challenge of accurate thermodynamic calculations near the critical point, which is labeled with the star.

3.2. Redlich–Kwong equation of state

Cantera currently represents real-gas effects using a mixture-averaged, multi-species implementation of the cubic Redlich–Kwong (R–K) equation of state [1], which is stated as

$$p = \frac{RT}{v - b^*} - \frac{a^*}{v\sqrt{T}(v + b^*)}. \tag{3}$$

The species-specific Van der Waals attraction parameter a^* and repulsive volume-correction parameter b^* represent molecular interaction effects [2]. For mixtures, a^* and b^* are evaluated from the pure-species parameters a_k and b_k, using well-known mixture-averaging rules [3, 4]. At low pressure and high temperature, the specific volume v is large relative to a^* and b^*. Thus, Eq. 3 is very well approximated by the ideal-gas law.

3.3. Thermodynamic properties

Especially near the critical point, it is well known that thermochemical properties are highly sensitive functions of temperature and pressure. Accurate property evaluation depends on real-gas behavior, as represented by an accurate equation of state (EoS) and consistent thermodynamic properties. Although a variety of real-gas models are available [1, 5, 6], their application to complex combustion and other reactive-flow simulations remains limited [7, 8]. In 1994, Prof. Barry Butler (Embry-Riddle Aeronautical University) and colleagues implemented a real-gas version of Chemkin [9]. However, it found limited use and is certainly no longer readily available.

In broad-brush terms, the cubic equations of state (e.g., Redlich–Kwong) permit evaluation of the Helmholz free energy A from its definition as

$$p = -\left(\frac{\partial A}{\partial v}\right)_{n_k,T}, \tag{4}$$

where v is the molar volume and the composition and temperature are fixed [3]. Once the Helmholtz free energy is established, all the other thermodynamic properties follow from their definitions [4]. Although the details are not spelled out in the present document, they have been implemented into Cantera.

Evaluating non-ideal thermodynamic properties presents numerous challenges. For example, some thermodynamic properties can be nearly singular near the critical point and liquid-vapor equilibrium. Such behavior is illustrated by comparing Cantera's multi-component R–K implementation to predictions using a more accurate, multi-parameter EoS (i.e., NIST's CoolProp, which implements a Helmholtz EoS). Although multi-parameter EoS are not currently practical for incorporation into large-scale CFD, they provide valuable references for evaluating the accuracy of more-approximate models. Figure 3 shows temperature as a function of specific volume for CO_2 for a range of isobars, while Figs. 4 and 5 show the constant-volume specific heat for CO for an elevated temperature (Fig. 4)

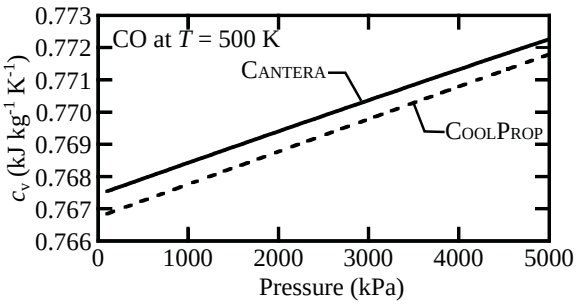

Figure 4: Comparison of predicted c_v for CO for a range of pressures at $T = 500$ K. Comparisons between CoolProp and Cantera's Redlich–Kwong implementation demonstrate close agreement at temperatures well above the critical point.

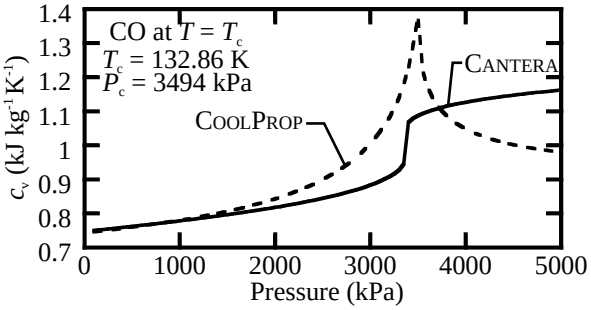

Figure 5: Comparison of predicted c_v for CO for a range of pressures at $T = T_c$. Comparisons between CoolProp and Cantera's Redlich–Kwong implementation demonstrate qualitative disagreements in the vicinity of the critical point.

and at the critical temperature (Fig. 5).

These figures illustrate the difficulty in accurate thermodynamic property evaluation near the critical point. Figure 3 shows that the Cantera predictions are qualitatively correct, but differ significantly from the CoolProp predictions near the critical point (starred in Fig. 3). Similarly, Fig. 4 shows that the Cantera predictions are accurate far from the critical point (the error in c_v is 0.1% across the range of pressures). However, Fig. 5 shows that Cantera's c_v predictions are not only highly inaccurate near the critical point, but do not even reproduce qualitative trends in the data. The CoolProp data shows a maximum in c_v at the critical point, while the Redlich–Kwong EoS predicts a discontinuity and a monotonic increase in c_v with increasing pressure p. Not only will future software tools require increased accuracy in the vicinity of the critical point, but the singularity at the critical point, such as that in c_v, makes some thermodynamic properties intractable for calculations.

3.4. Chemical kinetics

The rate of progress $\dot{q}_i$ for a reversible reaction i may be written according to mass-action kinetics as [10, 11]

$$\dot{q}_i = \frac{1}{\gamma_i^*}\left(k_{\mathrm{f}i}\prod C_{\mathrm{ac},k}^{\nu'_{ki}} - k_{\mathrm{r}i}\prod C_{\mathrm{ac},k}^{\nu''_{ki}}\right), \tag{5}$$

where ν'_{ki} and ν''_{ki} are the forward and reverse stoichiometric coefficients, respectively, $k_{\mathrm{f}i}$ and $k_{\mathrm{r}i}$ are the forward and reverse rate constants for species k, and γ_i^* is the activity coefficient of the activated complex in the transition state for reaction i. The activity concentration of species k is written as

$$C_{\mathrm{ac},k} = \alpha_k C_k^\circ, \tag{6}$$

where α_k is the species activity and C_k° is the species molar concentration ($\mathrm{mol}_k\ \mathrm{m}^{-3}$) at a thermodynamic reference state. Using the activity coefficient γ_k, the activity concentration may be written as

$$C_{\mathrm{ac},k} = \gamma_k C_k, \tag{7}$$

where C_k is the actual species molar concentration ($\mathrm{mol}_k\ \mathrm{m}^{-3}$) at the current state, evaluated as:

$$C_k = \frac{X_k}{v}, \tag{8}$$

where X_k are the mole fractions. The activity concentration can be converted into a number of equivalent forms to represent departure from ideality (such as the species fugacity f_k). For example, the molar concentrations can be written as

$$C_k = \frac{p}{ZRT}X_k, \tag{9}$$

which results in

$$C_{\mathrm{ac},k} = \frac{p}{ZRT}\gamma_k X_k. \tag{10}$$

For an ideal gas, $Z = 1$ and $\gamma_k = 1$, leading to

$$C_{\mathrm{ac},k,\,\mathrm{Ideal\ gas}} = \frac{p}{RT}X_k. \tag{11}$$

Although the required mathematical manipulations are tedious, the activity coefficients γ_k can be derived from the Redlich–Kwong equation of state as [3]

$$\begin{aligned} RT\ln(\gamma_k) = {} & RT\ln\left(\frac{v}{v-b^*}\right) + RT\left(\frac{b_k}{v-b^*}\right) \\ & + \frac{a^* b_k - 2b^*\sum_j a_{jk}X_j}{b^{*2}\sqrt{T}}\ln\left(\frac{v+b^*}{v}\right) \\ & - \frac{a^*}{b^*\sqrt{T}}\left(\frac{b_k}{v+b^*}\right), \end{aligned} \tag{12}$$

where a_k and b_k are the pure-species Van der Waals parameters, a^* and b^* are the mixture-averaged parameters, and $a_{ij} = a_i^{0.5} a_j^{0.5}$.

The variables γ_i^*, k_{fi}, and k_{ri} (Eq. 5) all depend on the thermodynamics of the transition state, which are not always readily available. As such, these real-gas effects are frequently neglected. When they are addressed, they are typically limited to the evaluation of γ_k and C_k. The transition-state thermodynamic properties could, in theory, be accessed via electronic structure calculations, but such an approach rapidly becomes untenable for mechanisms with hundreds of reactions. Instead, new software extensions are needed that efficiently leverage the capabilities of electronic structure software programs. User-friendly interfaces with these programs will make it such that transition-state thermodynamics can be efficiently calculated and automatically imported into CANTERA mechanism input files, for a seamless user experience.

However, even the approach above has limitations, in that it implicitly assumes that the reactants and products are equilibrated with the transition state according to a Maxwell–Boltzmann (i.e., "thermalized") distribution. Near the critical point, the collision rate changes substantially, and the activated complex may not be in a thermalized state. As a result, the branching fractions between various competing pathways will vary as the critical point is approached, or in some cases entirely different products may be formed, relative to thermalized reactions. Conventional modeling tools are currently not equipped to capture the functional dependence for such reactions.

3.5. Ignition delay prediction

The ignition-delay times (IDTs) measured in shock-tube experiments are frequently used to characterize hydrocarbon fuel combustion. Typically, IDTs decrease with increasing temperature. However, at intermediate temperatures, IDTs actually *increase* with increasing temperature, a phenomena referred to as negative temperature coefficient (NTC) behavior (cf., Fig. 6). It is interesting to investigate the effects of non-ideal behavior within this so-called negative temperature coefficient (NTC) region, to assess the impact of high-pressure on combustion dynamics.

The reflected-shock conditions may be represented simply as a constant volume, adiabatic reactor. Stated

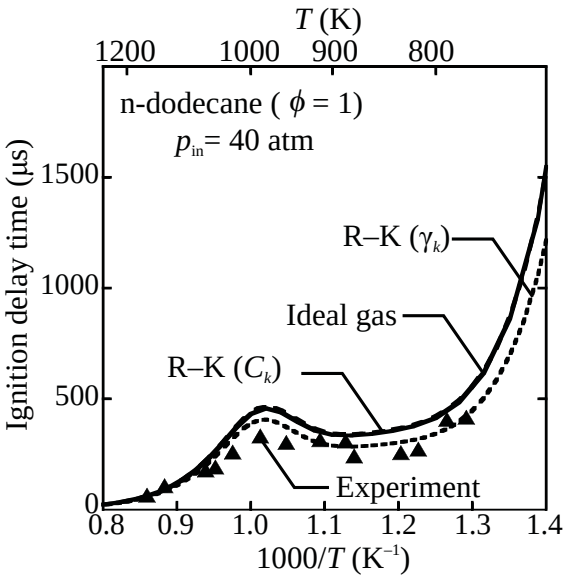

Figure 6: Ignition delay time as a function of temperature for different EoS at 40 atm.

mathematically,

$$\frac{du}{dt} = 0, \qquad \frac{dC_k}{dt} = \dot{\omega}_k, \tag{13}$$

where u is internal energy, t is time, C_k are the molar concentrations ($\mathrm{mol}_k\ \mathrm{m}^{-3}$), and $\dot{\omega}_k$ is the net molar production rate of species k due to chemical reactions ($\mathrm{mol}_k\ \mathrm{m}^{-3}\ \mathrm{s}^{-1}$). The problem represented by Eq. 13 is reformulated with the species mass fractions Y_k and temperature T as the dependent variables, and the system of stiff ordinary differential equations is solved computationally. Throughout the solution process, appropriate CANTERA functions are used to evaluate thermodynamic properties and chemical reaction rates.

To demonstrate the importance of real-gas effects in high-pressure combustion, we examine the extent of non-ideality in predicted IDTs using three different thermodynamic and chemical kinetic implementations:

1. *ideal-gas thermodynamics and kinetics*: The ideal-gas EoS is used to evaluate all thermodynamic properties and γ_k=1 for all species. Results are labeled "Ideal gas."

2. *Real-gas thermodynamics, ideal-gas kinetics*: The R–K EoS (Eq. 3) is used for *p-v-T* behavior and associated thermodynamic calculations, but the mass-action kinetics uses an activity coefficient of unity for all species (i.e., the species activity con-

centration is replaced by the species concentration, $C_{\mathrm{ac},k} = C_k$). Results are labeled "R–K (C_k)."

3. *Real-gas thermodynamics and kinetics*: The R–K EoS (Eq. 3) is used for *p-v-T* behavior and associated thermodynamic calculations, and the activity coefficients γ_k are derived explicitly from the R–K EoS (Eq. 12). Results for this implementation are labeled "R–K (γ_k)."

Figure 6 shows predicted IDTs for the three implementations at an initial pressure of $p_{\mathrm{in}} = 40$ atm, along with the experimental shock-tube data. Simulated IDTs are compared to experimental data for high-pressure combustion of n-dodecane/air mixtures as measured by Prof. Matt Oehlschlaeger's group (Rensselaer Polytechnic Institute). The simulated IDTs for all three implementations show the same qualitative trends, including NTC behavior at intermediate temperatures, a sharp increase toward very high IDTs at low temperature, and convergence toward very short delay times for $T >$ 1000 K. The "R–K (γ_k)" EoS, with consistent activity coefficients, predicts the experimental data most accurately. In the NTC region, depending on the assumptions, the ignition delay differs by roughly 50 μs, which is sufficient to influence design decisions for some combustion applications. Simulations at 60 atm and 80 atm show similar behaviors. Real-gas effects become more prominent with decreasing T (< 1000 K), as fuel species approach their critical properties. Conversely, real-gas effects become insignificant for $T > 1000$ K.

Predictions resulting from the second case "R–K (C_k)" are not thermodynamically consistent with the EoS. The reverse rate constant and the species concentrations C_k are calculated as $k_{\mathrm{fi}}/K_{\mathrm{c}}$ and X_k/v respectively, instead of using the non-ideal EoS and γ_k. However, the activity concentrations in the mass-action expression are evaluated using $\gamma_k = 1$. However, in the absence of clearly established best practices for such calculations, the implementation might reasonably be applied by researchers to incorporate real-gas effects, and is therefore worth considering. Comparing the two real-gas implementations (R–K (C_k) and R–K (γ_k)) reveals that neglecting the real-gas influence on the activity coefficient "R–K (C_k)" creates significant error at these conditions. Although the thermodynamically consistent implementation predicts lower IDTs than the ideal-gas case, the thermodynamically inconsistent model predicts a slight *increase* in IDTs, due to real-gas effects. Great care must be taken to develop thermodynamically consistent EoS implementations, particularly as they relate to species activity concentrations for mass-action kinetics.

The trends for the three thermo-kinetic implementations can be interpreted in terms of the mixture compressibility Z and the species activity coefficients γ_k. For the "R–K (C_k)" model, the only real-gas effect in the kinetics comes from the impact of Z on the molar concentrations (Eq. 9). Because of the high-concentration bath gas (dominantly N_2), the *mixture* behaves essentially as an ideal gas and Z is very close to the ideal-gas value, $Z = 1$. In reality, some of the hydrocarbon species are much closer to their critical conditions, which affect the mass-action kinetics for individual reactions, via activity coefficients that diverge from the $\gamma_k = 1$ case. This effect is neglected entirely in the "R–K (C_k)" model, but is captured in the "R–K (γ_k)" model.

The influence of activity coefficients on chemical rates of progress is often neglected in mass-action kinetics, as many studies adopt some form of the ideal-gas standard. Because the form of the mass-action law affects how the kinetic parameters k_{f} are fit to experimental data, correctly capturing real-gas effects on chemical kinetics is more than an academic consideration.

The ignition-delay results illustrate the influence of formulating and implementing thermodynamically consistent models. Although the impact may be small for some problems (e.g., high-temperature flames), the impact for high-pressure low-temperature processes (e.g., catalytic fuel processing reactors) and combustion without significant N_2 bath gas (e.g. pyrolysis, oxy-combustion) may be much greater. Thus, providing high-fidelity and general-purpose capabilities in the Cantera software is expected to have great value.

4. Heterogeneous catalysis

Both Chemkin and Cantera have capabilities to include surface chemistry and heterogeneous catalysis. However, most current catalysis models consider simple crystal surfaces (platinum, rhodium, nickel, etc.) and/or idealized oxide surfaces on chemically passive supports or idealized bulk phases. There is great value in extending current capabilities to represent the more complex catalyst structures that see wide industrial use. The models must incorporate technologically significant effects such as localized spillover, bulk-surface interactions, and dynamic surface structures.

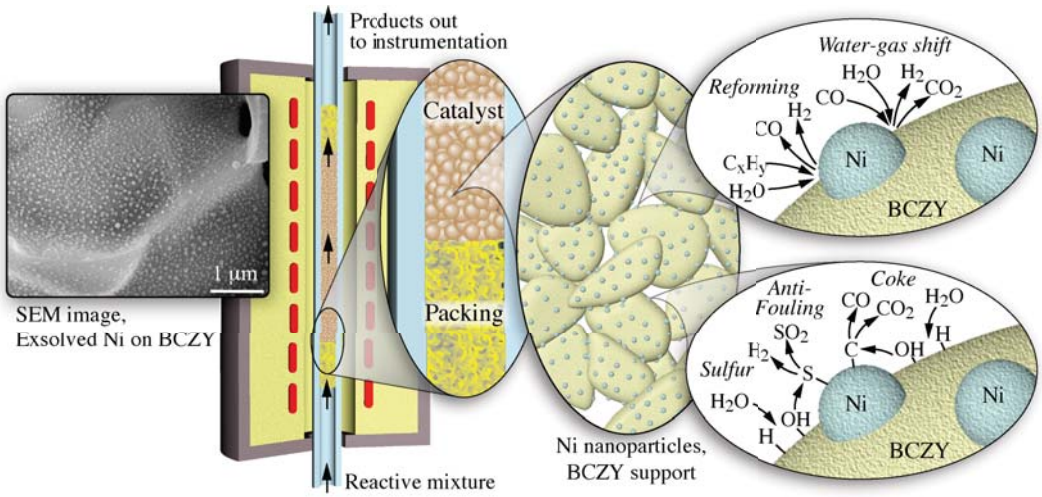

Figure 7: Typical packed-bed reactor using a catalyst with redox-active supports. The scanning electron micrograph on the left shows nano-scale exsolved Ni particles on a yttrium-doped barium zirconate support.

4.1. Redox-active supports

Figure 7 illustrates a laboratory-scale packed-bed reactor, showing at different length scales Ni catalyst particles on a reactive support of BCZY (a doped perovskite oxide with bulk-phase conductivity of protons and other ions). These bifunctional structures have demonstrated excellent reforming performance, while resisting deleterious poisoning by carbon or sulfur deposits [12]. Although not yet fully understood at the atomistic level, the reaction mechanisms that govern the catalyst-support interactions are illustrated in the right-hand balloons of Fig. 7.

The unique coke tolerance of the Ni/BCZY catalyst is related to its bifunctional properties and the redox behavior of the BCZY support. The BCZY support provides rapid H_2O dissociation to form surface-adsorbed hydroxyl OH(BCZY) and hydrogen H(BCZY) compounds, which, upon spillover to the Ni surface, promote carbon removal via oxidation and water-gas shift processes. Temperature and gas-phase H_2O concentration variations may result in different catalytic pathways to form surface intermediates such as formyl COH, carboxyl COOH or formate HCOO species. The following surface reactions describe plausible pathways where COOH(Ni) and COH(Ni) are formed and consumed:

$$H_2O + BCZY \rightleftharpoons OH(BCZY) + H(BCZY) \quad (14)$$

$$C(Ni) + OH(BCZY) \rightleftharpoons COH(Ni) + BCZY \quad (15)$$

$$COH(Ni) + O(Ni) \rightleftharpoons COOH(Ni) + Ni \quad (16)$$

$$COOH(Ni) + OH(Ni) \rightleftharpoons CO_2(Ni) + H_2O(Ni) \quad (17)$$

$$COOH(Ni) + OH(BCZY) \rightleftharpoons CO_2(Ni) + H_2O(BCZY) \quad (18)$$

$$CO(Ni) + OH(BCZY) \rightleftharpoons COOH(Ni) + BCZY \quad (19)$$

Reactions between the adsorbates on the two surfaces, as illustrated in Fig. 7, are driven by spillover as well as surface and bulk-phase species transport, which are not captured by the mean-field approximations that are used in most models. These limitations point to the value of enhanced modeling capabilities for heterogeneous surface reactions that can represent the transport and thermodynamics of adsorbates and bulk species near surface-phase interfaces.

The CANTERA architecture is designed to accommodate multiple phases, as illustrated in Fig. 7 with a single gas phase, two bulk-solid phases with two unique surfaces, and surface interfaces. CANTERA can represent so-called surface "interfaces" of different bulk phases as "edges" between surfaces. Reactions may occur on interfaces or at edges, but the capabilities for representing the interplay between these constructs and bulk-phase species transport are limited. Thus, critical phenomena in common catalyst structures, particularly with active supports as in Fig. 7 cannot be directly modeled. Extensions for capabilities to model multiphase heterogeneous catalysis systems have been identified and will likely evolve as models for particular catalyst structures are developed.

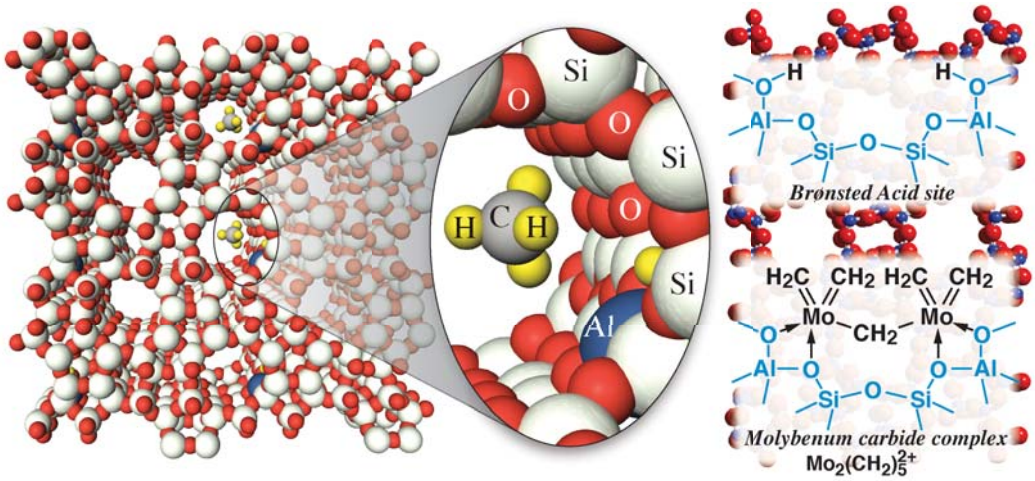

Figure 8: Zeolite catalyst with a molybdenum-carbide complex at the Brønsted acid site.

4.2. Zeolite catalysts

Other types of catalyst-support architectures are widely employed for bifunctional behavior. For example, Fig. 8 illustrates a zeolite-based catalyst with a molybdenum-carbide (MoC) complex at a Brønsted acid site that is used in methane dehydroaromatization (MDA) to convert natural gas to benzene [13]. Development of similar bifunctional catalysts have supported processes such as endothermic fuel pyrolysis in hypersonic combustion applications.

In MDA, activated CH_4 on MoC sites reacts to form hydrogen and ethylene. The H_2 and C_2H_4 are transported to open Brønsted acid sites where protonation and deprotonation reactions ultimately produce benzene and other cyclic aromatic compounds. As with the Ni/BCZY catalyst, the MoC/zeolite bifunctional catalyst depends on adsorbates transport between surface phases. However, for the MoC/zeolite catalyst, the phase interfaces are at the molecular scale and further nanoscale channels within the zeolite crystal directly affect the species transport and catalytic selectivity. Current catalytic reactor models have limited capability to address such issues, motivating further extension of heterogeneous surface chemistry models.

Essentially all current chemical kinetics modeling software uses the mean-field approximation, which assumes that adsorbates are uniformly distributed on surfaces at constant site fractions [14]. However, adsorbates are known to form clusters on some catalyst surfaces. In such cases, the mean-field approximation is invalid. CHEMKIN and CANTERA provide interfaces to specify coverage-dependent activation energies that approximate the effects of nonuniform site fractions, but significant opportunities to move beyond the mean-field approximation can advance fundamental understanding of adsorbate dispersion and develop corresponding software extensions.

CHEMKIN and CANTERA provide the ability to represent surface reactions as being microscopically reversible. However, as a practical matter, thermodynamic properties must be available for all the adsorbates. Unlike gas-phase species, thermodynamic properties for surface species are not readily available. For surface adsorbates, computationally complex ab initio models can provide quantitative guidance, providing needed properties to the heterogeneous reaction models. Developing thermodynamic property databases is not directly the role of chemical kinetics software. Nevertheless, it is important for next-generation catalysis models to maintain compatibility with atomistic modeling capabilities.

5. Electrochemistry

Computational simulations play important roles in understanding and improving electrochemical devices such as batteries and fuel cells. However, the commonly used chemistry models usually rely on a few global reactions. Unlike gas-phase combustion mechanisms, which can often involve hundreds or thousands of reactions, large electrochemical reaction mechanisms typically include no more than 5–10 reactions. Thus, as electrochemical knowledge expands, new modeling ca-

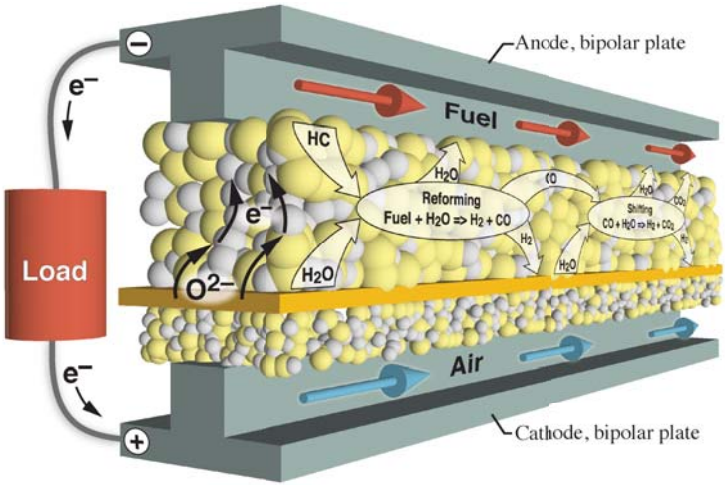

Figure 9: Planar solid-oxide fuel cell (SOFC) membrane-electrode assembly (MEA) with a hydrocarbon-reforming composite anode. The reforming is represented in global terms with a steam reforming reaction ($CH_4 + H_2O \rightleftharpoons 3H_2 + CO$) and water-gas-shift reaction ($CO + H_2O \rightleftharpoons H_2 + CO_2$).

pabilities are needed to conveniently handle more extensive sets of electrochemical reactions and charge-transfer processes.

5.1. Solid-oxide fuel cells

Figure 9 shows the salient features of a planar SOFC operating on a hydrocarbon (e.g., methane) fuel [17]. Fuel enters via the upper flow channel and reaction products leave via the same channel. The oxidizer, typically air, flows through the lower channel. The channels typically have characteristic dimensions on the order of a millimeter and the flow is typically laminar. As illustrated, the composite electrodes are porous structures composed of two solid phases, one being an oxygen-ion conductor (electrolyte) and the other being an electron conductor (electrode). Gases fill the pore volume. A dense electrolyte membrane is sandwiched between the electrode structures. In the case of an SOFC, the dense electrolyte membrane is typically a material such as yttrium-stabilized zirconia (YSZ), which is an oxygen-ion conductor with essentially no electronic conduction.

The composite anode (negative electrode) structure serves several important functions. For one, the relatively thick composite serves as a structural support for thin (order tens of microns) dense electrolyte membrane. Second, the electron-conducting phase (e.g., Ni) serves as a current collector to conduct electrons to the bipolar plate that forms the flow channel. Charge-transfer chemistry (i.e., fuel electro-oxidation) proceeds at the *three-phase boundaries* (TPB) formed at the intersections between electrode particles, electrolyte particles, and the gas. Finally, the electron-conducting phase (e.g., Ni), acts as a reforming catalyst, promoting the reaction between H_2O and CO_2 (electrochemical products) and the fuel (e.g., CH_4) to produce the more electrochemically active H_2.

The composite cathode (positive electrode) is structurally similar to the anode. Its role, however, is to electrochemically reduce molecular oxygen and deliver oxygen ions into the electrolyte phase. The strongly oxidizing environment in the cathode prevents use of electron-conducting materials such as Ni, which would oxidize to NiO and lose electronic conductivity. The electron-conducting phase for SOFC cathodes is typically an oxide ceramic such as strontium-doped lanthanum manganate (LSM).

Electrochemical charge-transfer reactions take place at the interface between an electron-conducting electrode and an ion-conducting electrolyte. All electrochemical charge-transfer processes are heterogeneous in the sense that they involve transferring electronic charge between two material phases. Electrochemical reactions involve charged species whose activity depends on the electrostatic potential of the phase in which the species resides, and the charge-transfer rate therefore depends on the electrostatic potential difference between the two participating phases. The species from which an electron is stripped is said to have been oxidized, while the species to which the electron is added is said to have been reduced.

An electrochemical cell, such as a fuel cell (Fig. 9), is composed of two half-cells, each of which consists of an

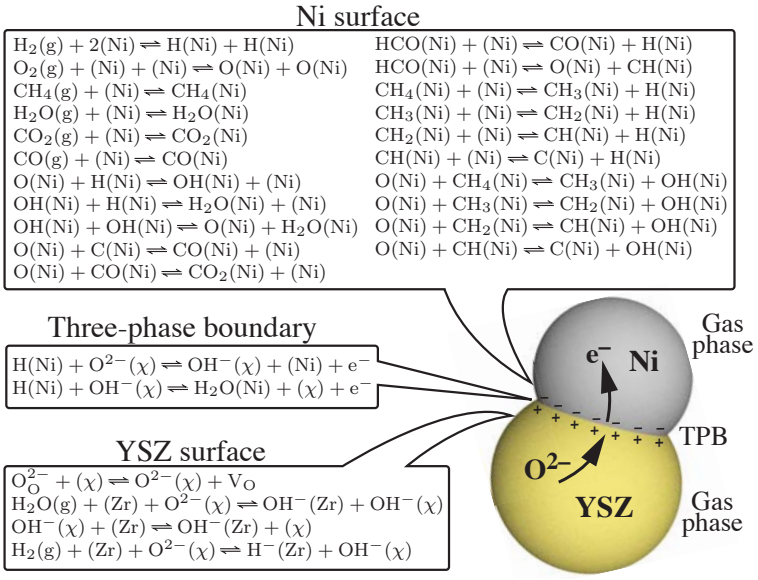

Figure 10: Possible thermal and electrochemical reactions on the electrode (Ni) and electrolyte (YSZ) surfaces and charge-transfer reactions at the particle interface [15]. The Ni thermal chemistry includes the possibility of methane reforming [16].

electrode and an electrolyte. The anode is the electrode where oxidation occurs and the cathode is the electrode where the reduction takes place. In principle, electrodes can be made from any sufficiently conductive material, including metals, semiconductors, graphite, or conductive polymers. Of course, the material must be stable in its operating environment. An ion-conducting electrolyte (with negligible electronic conductivity) must be situated between the electrodes. The two electrodes (anode and cathode) must be connected electrically via an external circuit such that electrons can conduct between the electrodes and through some load. The electrochemical processes depend on oxidizing chemical species in one half-cell (delivering electrons from the anode) and reducing chemical species (consuming electrons within the cathode) in the other half-cell.

Electrochemical cells can be characterized as being either *galvanic* or *electrolytic*. Galvanic cells, such as fuel cells, derive electrical energy from spontaneous oxidation and reduction (redox) reactions taking place within the electrochemical cell. Electrolytic cells, such as steam electrolyzers, require an imposed electric potential (polarization) in excess of the open-circuit potential to drive the electrochemical redox processes. Galvanic cells produce electricity by oxidizing a fuel. Electrolytic cells consume electricity to activate a chemical process, such as reducing H_2O and/or CO_2 to produce fuel such as H_2 and CO. Rechargeable batteries operate in both electrolytic (charging) and galvanic (discharging) modes. Electrochemical kinetics therefore must be implemented in a robust and general manner, in order to accommodate the variety of applications and operating modes where electrochemistry plays a role.

5.2. *Electrochemical kinetics*

Thermal and electrochemical reversible reactions can be represented generally as

$$\sum_{k=1}^{K} \nu'_{ki} \mathrm{X}_k^{z_k} \rightleftharpoons \sum_{k=1}^{K} \nu''_{ki} \mathrm{X}_k^{z_k}, \tag{20}$$

where $\mathrm{X}_k^{z_k}$ is the chemical symbol for the kth species with charge z_k. The indices represent species k and reaction i, with K being the number of species involved in the reaction. The forward and backward stoichiometric coefficients are represented as ν'_{ki} and ν''_{ki}, respectively. Reactions may involve species that reside in different phases, such as electrode and electrolyte phases. By convention, reversible charge-transfer reactions are usually written such that the forward direction is the *anodic direction* (i.e., producing electrons). The backward direction is called *cathodic*, meaning that electrons are consumed.

Consider the following elementary charge-transfer reaction (cf., Fig. 10), which serves as an example for developing the theory to evaluate kinetics rates,

$$\mathrm{H(Ni)} + \mathrm{OH}^-(\chi) \rightleftharpoons \mathrm{H_2O(Ni)} + (\chi) + \mathrm{e}^-. \tag{21}$$

Two surface sites are involved—one is the Ni surface and the other is a χ site on the YSZ surface [15]. The anodic (forward) and cathodic (backward) stoichiometric coefficients are

$$\nu'_{\mathrm{H(Ni)}} = 1,\ \nu'_{\mathrm{OH^-}(\chi)} = 1,\ \nu''_{\mathrm{H_2O(Ni)}} = 1,\ \nu''_{\mathrm{e^-}} = 1,\ \nu''_{(\chi)} = 1. \tag{22}$$

The electrostatic potential at the Ni surface (i.e., electronic-conducting electrode phase) is different from that at the YSZ surface (i.e., ionic-conducting electrolyte phase), which affects the activities of charged species residing therein. The H(Ni), H_2O(Ni), and (χ) are assumed not to carry a charge (i.e., $z = 0$). The adsorbed hydroxyl $OH^-(\chi)$ and the electron (within the Ni phase) both carry a single negative charge (i.e., $z = -1$).

5.3. Rates of Progress

Considering the general reaction (Eq. 20), a reaction's rate of progress can be written in terms of the difference between forward (anodic) and backward (cathodic) reaction rates of progress q_i as

$$q_i = q_{\mathrm{f}i} - q_{\mathrm{r}i} = k_{\mathrm{f}i} \prod_{k=1}^{K} C_{\mathrm{ac},k}^{\nu'_{ki}} - k_{\mathrm{r}i} \prod_{k=1}^{K} C_{\mathrm{ac},k}^{\nu''_{ki}}, \tag{23}$$

The activity concentrations $C_{\mathrm{ac},k}$ of gas-phase species are the molar concentrations C_k, under typical conditions. The activity concentrations of the surface-adsorbed species are the surface coverages $\Gamma_m \theta_{k,m}$, where $\theta_{k,m}$ is the site fraction for species k on the surface of phase m and Γ_m is the total available surface site density on phase m.

The forward and backward rate expressions for each reaction i are written as

$$k_{\mathrm{f}i} = k_{\mathrm{f}i}^{\mathrm{t}} \exp\left[-\beta_{\mathrm{f}i} \sum_{k=1}^{K} \frac{\nu_{ki} z_k F \Phi_k}{RT}\right], \tag{24}$$

$$k_{\mathrm{r}i} = k_{\mathrm{r}i}^{\mathrm{t}} \exp\left[+\beta_{\mathrm{r}i} \sum_{k=1}^{K} \frac{\nu_{ki} z_k F \Phi_k}{RT}\right], \tag{25}$$

where $\nu_{ki} = \nu''_{ki} - \nu'_{ki}$. $\beta_{\mathrm{f}i}$ and $\beta_{\mathrm{r}i}$ are the forward and backward symmetry factors, with $\beta_{\mathrm{f}i} + \beta_{\mathrm{r}i} = 1$ for elementary (i.e., single-electron transfer) reactions. The forward and backward thermal rate coefficients (i.e., at zero electric-potential difference) are represented as $k_{\mathrm{f}i}^{\mathrm{t}}$ and $k_{\mathrm{r}i}^{\mathrm{t}}$.

Each phase m is assumed to be at an electric potential Φ_m. However, as a matter of convenience in computational implementation, each species can be assigned the potential associated with its phase. In other words, in writing Eqs. 24 and 25, each species k is assigned with an electric potential Φ_k, not the phase Φ_m directly. Generally, each species assumes the electric potential of phase in which it exists. The gas phase is usually considered to be electrically neutral.

If all the charged species in an electrochemical reaction are in the same phase, the electric potentials do not affect the reaction rate (i.e., the exponential factors in Eqs. 24 and 25 are exactly unity). For example, consider the reaction $I_2 + I^- \leftrightharpoons I_3^-$, which may occur within an aqueous electrolyte (i.e., a single phase). In this case, assuming Φ_k is the same for all species, $\sum_k \nu_k z_k = \nu_{\mathrm{I_2}} z_{\mathrm{I_2}} + \nu_{\mathrm{I^-}} z_{\mathrm{I^-}} + \nu_{\mathrm{I_3^-}} z_{\mathrm{I_3^-}} = (-1) \times (0) + (-1) \times (-1) + (+1) \times (-1) = 0$. However, when there is a transfer of charge between phases at different electric potentials, the exponential factors in Eqs. 24 and 25 are not equal to unity, and the charge transfer reaction rate is modified by the electric potentials.

The thermal reaction rate expressions k_i^{t} ($k_{\mathrm{f}i}^{\mathrm{t}}$ or $k_{\mathrm{r}i}^{\mathrm{t}}$) are usually represented using the modified Arrhenius expression as

$$k_i^{\mathrm{t}} = A_i T^{n_i} \exp\left(-\frac{E_i}{RT}\right), \tag{26}$$

where E_i represents the activation energy, A_i the pre-exponential factor, and n_i the temperature exponent. To satisfy microscopic reversibility and maintain thermodynamic consistency, the thermal component of the backward (cathodic) rate $k_{\mathrm{r}i}^{\mathrm{t}}$ is related to the forward (anodic) rate $k_{\mathrm{f}i}^{\mathrm{t}}$ via the reaction equilibrium constant K_i as

$$K_i = \frac{k_{\mathrm{f}i}^{\mathrm{t}}}{k_{\mathrm{r}i}^{\mathrm{t}}} = \exp\left(-\frac{\Delta G_i^\circ}{RT}\right), \tag{27}$$

where ΔG_i° is the change of the standard-state Gibbs free energy for the reaction. Evaluating ΔG_i°, and hence the equilibrium constant, requires quantitative thermochemical properties for all species in the reaction.

Return attention to a charge-transfer reaction in the electrochemical hydrogen-oxidation mechanism (Eq. 21). Figure 11 illustrates potential-energy surfaces that assist understanding the influence of electric potentials on charge-transfer rates. The potential-energy surface on the left represents the reactants and the one on the right represents the products. The electric-potential difference between the electrode (here, the anode Ni) and the electrolyte (here, the YSZ) is written as $E_{\mathrm{a}} = \Phi_{\mathrm{a}} - \Phi_{\mathrm{e}}$. The equilibrium electric-potential difference $E_{\mathrm{a}}^{\mathrm{eq}}$ is the electric-potential difference at which the reaction proceeds at equal and opposite rates in the an-

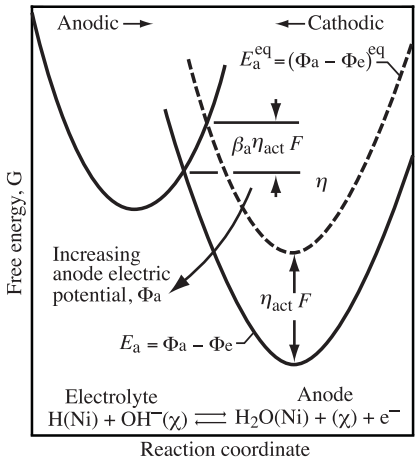

Figure 11: Potential energy surfaces to assist visualizing the effect of electric-potential difference on charge-transfer reaction rates.

odic (forward) and cathodic (backward) directions (illustrated as the dashed line).

There is a potential-energy barrier between the reactant and product states, which tends to be cusp-like for charge-transfer reactions. When proceeding in the anodic direction, the charge-transfer reaction illustrated in Fig. 11 delivers electrons to the anode, which is at a lower electric potential than the electrolyte. The negatively charged electron is naturally repelled from the negative electrode. Some of the chemical energy stored as chemical bonds in the reactants is therefore converted to electric current as electrons are delivered into the Ni anode. As the electric potential of the anode increases relative to the electrolyte (i.e., E_a increases), the barrier to the electron transfer decreases. The symmetry factors β are related to the slopes of the potential-energy surfaces at their crossing point. Because the slopes are typically similar, the symmetry factors for elementary reactions are usually near $\beta \approx 1/2$. When the anode electric potential Φ_a is increased relative to the adjoining electrolyte electric potential Φ_e, the *activation overpotential* η_{act} is increased by the same amount. As illustrated in Fig. 11, the product-side potential energy surface is lowered by $\eta_{act}F$ and the anodic energy barrier is lowered by $\beta_a \eta_{act} F$.

To fully support electrochemical modeling, capabilities must exist to evaluate electrochemical reaction rates and provide functionality to specify needed physical and electrochemical chemical parameters (e.g., symmetry factors β, species charges z_k,...). Moreover, capabilities will likely be needed to support alternative charge-transfer rate expressions and reaction types. The Butler–Volmer formulation is one such example.

5.4. Butler–Volmer form

The charge-transfer processes at the electrode and electrolyte interfaces normally involves several thermal and electrochemical reaction steps. In other words, the overall charge-transfer process is not the result of a single elementary reaction. The charge-transfer rates of the electrochemical reactions, such as electrochemical H_2 oxidation within the SOFC anode and O_2 reduction within the SOFC cathode, can be represented globally in terms of Butler–Volmer formulation as

$$i_{e,BV} = i^\circ \left[\exp\left(\frac{\alpha_a F \eta_{act}}{RT} \right) - \exp\left(-\frac{\alpha_c F \eta_{act}}{RT} \right) \right], \quad (28)$$

where α_a and α_c are the anodic and cathodic symmetric factors, i_0 the exchange current density, and η_{act} the activation overpotential. The symmetry parameters α (Eq. 28) indicate a *global reaction process*. Unlike the symmetry parameters β used for elementary (i.e., single-electron transfer) reactions (Eqs. 24 and 25), it is generally the case that $\alpha_a + \alpha_c \neq 1$. The activation overpotential η_{act} is defined as

$$\eta_{act} = E_{ed} - E_{ed}^{eq} = (\Phi_{ed} - \Phi_{el}) - (\Phi_{ed} - \Phi_{el})^{eq}, \quad (29)$$

where $E_{ed} = (\Phi_{ed} - \Phi_{el})$ is the electric-potential difference between the electrode phase and the electrolyte phase, and $E_{ed}^{eq} = (\Phi_{ed} - \Phi_{el})^{eq}$ is the electric-potential difference that causes the charge-transfer reaction to be equilibrated (i.e., proceeding in the anodic and cathodic directions at equal and opposite rates).

Figure 12 illustrates the functional behavior of the Butler–Volmer equation. Figures 12a and 12b represent an elementary reaction with $\beta_a = \beta_c = 0.5$. The negative overpotentials represent the cathodic branch, indicating that the reaction proceeds in the reverse (cathodic) direction, consuming electrons. The positive overpotentials indicate forward (anodic) rate of progress, producing electrons. Figure 12b plots the absolute value of the dimensionless current density. This form is called a Tafel plot. At relatively large overpotentials the slopes of the Tafel plot approach the symmetry factors, which are called Tafel slopes. Figures 12c and 12d show the behaviors for a global reaction with $\alpha_a = 1.5$ and $\alpha_c = 0.5$. Clearly, there is great asymmetry in the anodic and cathodic branches. As in the symmetric case, the Tafel slopes approach the symmetry factors α.

While current Cantera capabilities support the basic Marcus theory presented in Eqs. 23–27, functionality does not currently exist to support Butler–Volmer

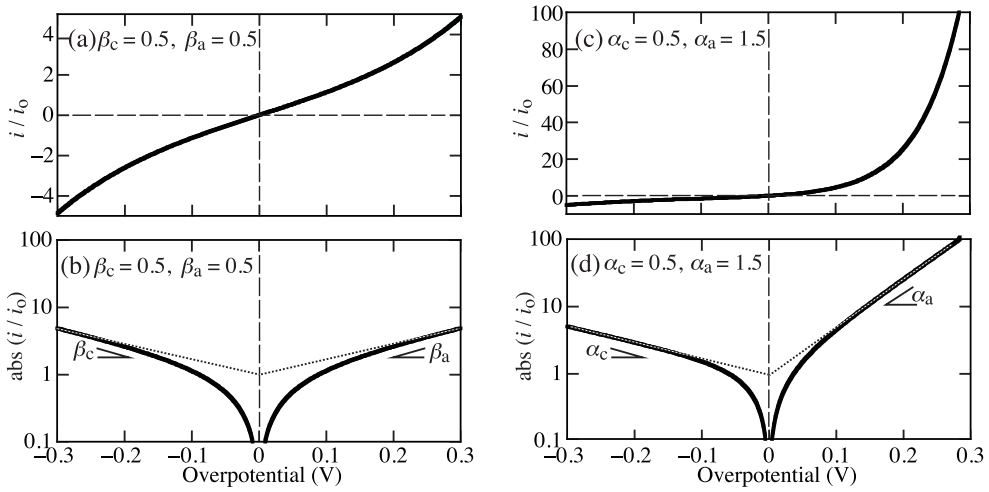

Figure 12: Representative functional forms of the Butler–Volmer equation. (a) Dimensionless exchange current density with equal symmetry factors, $\beta_a = \beta_c = 0.5$. (b) Tafel plot of the dimensionless exchange current density with equal symmetry factors, $\beta_a = \beta_c = 0.5$. (c) Dimensionless exchange current density with unequal symmetry factors, $\alpha_c = 0.5, \alpha_a = 1.5$. (d) Tafel plot of the dimensionless exchange current density with unequal symmetry factors, $\alpha_c = 0.5, \alpha_a = 1.5$. The dotted lines show projections of the Tafel slopes.

formulations. Given the prominence of Butler–Volmer rates presented in electrochemical literature, extensions to support these rate expressions will have great value.

5.5. Li-ion batteries

Figure 13 illustrates the range of relevant length scales within a Li-ion battery. Although fuel cells and batteries are both electrochemical devices that share similarities in the electrochemistry, there are significant differences that affect chemical kinetics and hence the needed software tools.

Many batteries, but not all, rely on *intercalation* electrodes. Consider, for example, as illustrated in Fig. 14, a charge-transfer reaction at the interface between the electrolyte solvent and a graphite anode. The reversible reaction may be written as

$$\mathrm{LiC_6(ed)} \rightleftharpoons \mathrm{Li^+(el) + C_6(ed) + e^-(ed)}, \qquad (30)$$

where the nomenclature (ed) and (el) refer to the electrode and electrolyte phases, respectively. The graphite electrode has electronic conductivity. The Li intercalates into the anode, forming the $\mathrm{LiC_6(ed)}$. Reaction 30 is written with the forward direction being anodic, meaning that electrons are produced within the electrode as the battery discharges. The reaction proceeds in the reverse (cathodic) direction upon charging.

The rates of charge-transfer reactions (e.g., Eq. 30) may be expressed using either the elementary Marcus theory or the Butler–Volmer form. However, essentially all battery literature uses the Butler–Volmer form. One reason is that it is relatively easy to measure reversible (open-circuit) potentials and represent the reaction rates in terms of overpotentials. In any case, the CANTERA software must be extended to handle charge-transfer reaction rate expressions in either form.

In typical Li-ion batteries, a solid electrolyte interface (SEI) forms as a passivation layer at the anode surface due to electrolyte instability at the low-voltage anode surface. Formation of a stable, conductive SEI beneficially prevents further electrolyte degradation, but also deleteriously increases the anode charge-transfer resistance. Chemical breakdown of the SEI at elevated temperatures is also a significant factor in the "thermal runaway" process that can lead to catastrophic battery failure and is a serious safety concern that limits Li battery adoption. Despite the importance of SEI chemistry for battery performance and safety, existing models employ only the simplest of chemical mechanisms, some of which consider only one SEI species and one reaction [18–24].

The reaction mechanisms illustrated in Fig. 15 represent one of the more complete SEI models in the literature [18, 21], but even this is only a partial representation of the SEI chemical complexity. Planned extensions of the CANTERA software are needed to represent the chemical and electrochemical complexity of SEI formation, growth, and decomposition.

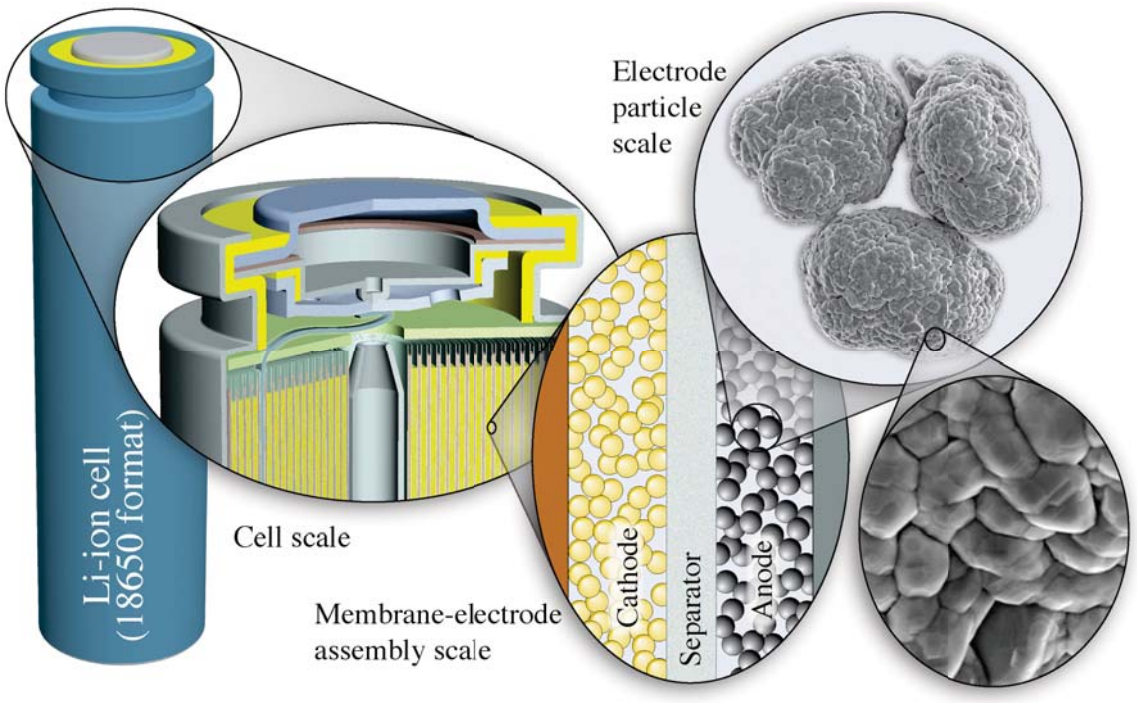

Figure 13: Range of scales in a Li-ion battery. The microscopic images for lithium-ion-phosphate anode particles are courtesy of Prof. William Cheuh (Stanford University).

The overall paucity of detailed electrochemical reaction mechanisms is due in part to the central role of materials discovery, where performance advances are generally associated with new materials and chemistries. Developing new detailed reaction mechanisms can be a challenging and lengthy process that struggles to keep pace with materials development. Thus, advanced modeling tools that facilitate mechanism development for new materials are extraordinarily valuable, especially in environments where the materials and chemistries themselves are works in progress and change frequently.

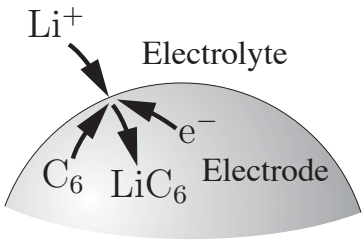

Figure 14: Illustration of a charge-transfer reaction at the graphite-electrolyte interface in the Li-ion battery anode. As illustrated, the battery is charging. Reaction 30 is a reversible reaction, but written with the forward direction being anodic (i.e., battery discharge).

In addition, new physical models and capabilities are required to accurately describe some charge-transfer processes. Figure 16 illustrates the range of phases and charge transfer processes that characterize a lithium-iron-phosphate $LiFePO_4$ cathode. Electrochemical reactions involve a wide array of condensed phases—liquid electrolytes and solids—with thermodynamic behaviors that are not accurately described by current models. Because electrochemical models require accurate species thermodynamic expressions for equilibrium voltage predictions, new thermodynamic classes are required in Cantera for these phases. In the lithium iron phosphate phase, new capabilities are also required to describe the phase transformation front that accompanies lithiation of the cathode (Fig. 16). This will require advanced models not currently found in either Cantera or Chemkin, which go beyond mean-field theory. Finally, in the electrolyte solvent phase, the high concentration of the lithium salts mean that typical "dilute solution" approximations (as in common Fick's Law expressions) are no longer valid. Instead, concentrated solution theory needs to be supported to enable accurate transport calculations.

The preceding text lays out some of the challenges for electrochemistry modeling, but a wide array of new

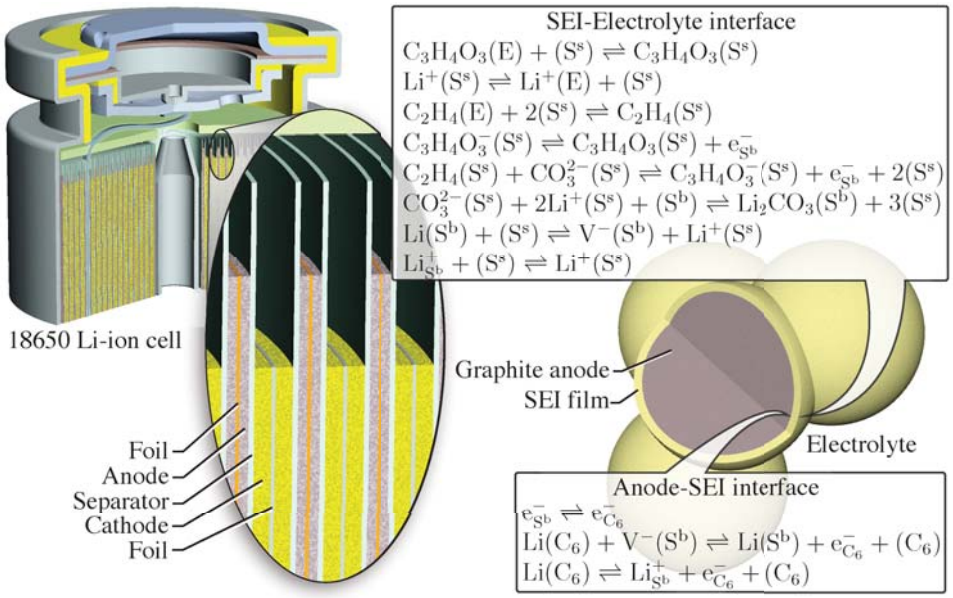

Figure 15: Representation of SEI chemistry in a Li-ion battery. The SEI is a passivation layer that forms at the surface of the graphite anode. Understanding the complex chemistry therein is critical to improving battery performance, durability, and safety.

phase models and alternative charge-transfer pathways, beyond those in Fig. 16 for modeling lithium iron phosphate cathodes, will require novel software solutions beyond current capabilities. For example, in lithium-sulfur and lithium-O_2 batteries—two promising "beyond Li-ion" batteries capable of storing significantly more energy per unit weight, relative to current technology, have charge transfer reactions via heterogeneous nucleation and growth of insoluble precipitates. Modeling such reactions will require support for nucleation and growth theory for kinetic rates and for adequate quantitative descriptions of the precipitate phases beyond current mean-field theory approaches.

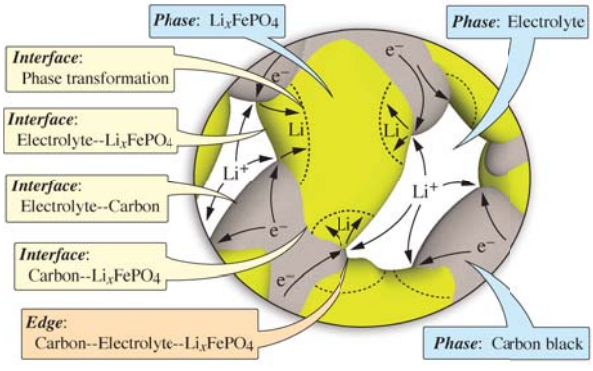

Figure 16: Illustration of the complex array of charge transfer processes and thermodynamic phase models required for lithium iron phosphate electrodes.

5.6. Electrochemical properties

As is evident from the foregoing discussion, the evaluation of charge-transfer rates depends on electrostatic-potential differences at phase interfaces. Thus, models must incorporate charge-conservation equations whose solution yields the electrostatic-potential distributions within electrode and electrolyte phases. In some sense this is analogous to reacting flow models (e.g., combustion) solving mass, momentum and energy equations to yield distributions of temperature, pressure, velocity and species concentrations. The role of software such as CANTERA is to provide the functionalities that are needed to evaluate terms in the conservation equations. Functions must be available to evaluate properties such as temperature-dependent ionic and electronic conductivities as well as charged-defect mobilities in the participating phases. As with thermodynamic and transport properties for the fluid phases, parameters to define the basic electrochemical material properties must be provided by the user or be available via databases.

5.7. Standard electrochemical models

There is great value in making available relatively straightforward "standard" user-friendly modeling frameworks for electrochemistry. In gas-phase

chemistry research, models such as continuously stirred tank reactors (CSTR), plug-flow reactors (PFR), premixed flames, and other settings are readily available and widely used. There are very few equivalents in electrochemistry. The so-called Newman models (named for Prof. John Newman, University of California, Berkeley) are widely used for modeling Li-ion batteries. However, the chemistry capabilities are very limited. Within the CANTERA framework a number of low-dimensional electrochemical models could be developed. Examples could include laboratory experiments such as rotating-disk electrodes, cyclic voltammetry, or button-cell fuel cells. There are also opportunities for modeling technology applications such fuel cells or batteries. Working within the CANTERA framework would enable the incorporation of much more complex chemistry and electrochemistry than is currently available.

6. Databases

Property databases play critically important roles in supporting research communities generally and modeling initiatives particularly. For example, CHEMKIN users rely heavily on the electronic availability of thermodynamics- and transport-property databases as well as readily available reaction mechanisms. Developers of new reaction mechanisms, particularly for gas phase mechanisms, begin with previously published reaction mechanisms and supporting properties. Some measure of format standardization is important to the widespread and convenient use of databases.

As new capabilities are developed, there will be continuing needs to establish and validate the physical parameters used therein. For example, species-specific parameters and mixing coefficients in non-ideal equations of state (such as the Redlich–Kwong EoS) are not included in current thermodynamic databases for modeling reactive flows. To incorporate these parameters into complex reaction mechanisms involving numerous species, databases of these non-ideal EoS parameters must be sustained and based on fundamental principles. These non-ideal EoS parameters will be required for intermediate and highly reactive species where measured data will be unavailable.

Heterogeneous reaction processes usually involve bulk-phase materials (e.g., catalysts and supports). Solid-phase thermodynamic and transport properties play essential roles in modeling such processes. CALPHAD [25] and FactSage [26] are software tools that provide capabilities for establishing and developing databases for material properties that can be readily integrated into heterogeneous reactor models. The CANTERA software can benefit by establishing electronic access to these modeling capabilities and associated databases.

For electrochemical models, significant new data about charge-transfer reactions, ionic species transport properties, and thermodynamics of intermediate species will be critical. The thermodynamics and transport properties for many, if not most, charged species, have not been measured and are certainly not readily available. Assuming that the properties can be established (theoretically and/or experimentally), validated databases must be developed and supported. The availability of species and materials property databases as well as reaction mechanisms will be of vital importance for accelerating adoption of these modeling tools by academic, industrial, and government researchers.

7. Software interoperability

As with the CHEMKIN capabilities, the CANTERA software is not designed or intended to stand alone. Rather, it plays valuable intermediate and enabling roles. Thus, compatibility with other software is essential. The majority of users interact with CANTERA via writing or using MATLAB or PYTHON scripts that access core CANTERA capabilities to evaluate thermodynamic, transport, or kinetics functions. In these cases, it is the role of the user's program to implement the formulation and solution of governing conservation equations. Thus, as new core CANTERA capabilities are developed (i.e., written in C++), it is important that the interface wrappers are written to enable convenient access from higher-level programming languages.

7.1. Fundamental or ab initio models

The CANTERA software must *use* thermodynamic and transport properties as well as chemical kinetics rate expressions and parameters. However, it is *not the role* of CANTERA to develop such properties and parameters. Rather, fundamental chemical properties and rate theories must come from experiment or from *ab initio* models. As such, it is essential that the CANTERA capabilities be written and supported in ways that facilitate effective and convenient communication with more fundamental supporting software.

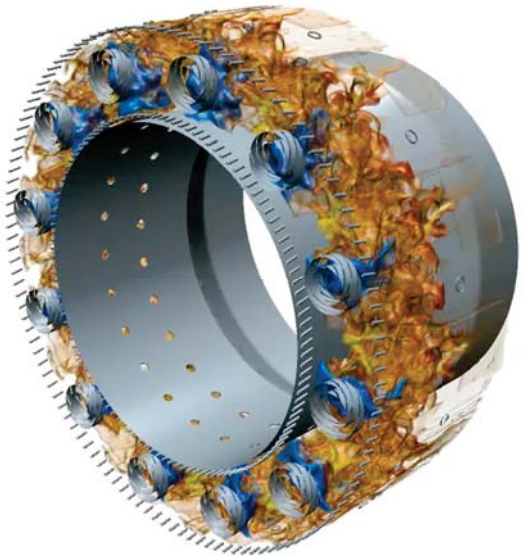

Figure 17: Computational Fluid Dynamics model of a gas-turbine combustor. Image courtesy of CD-adapco.

7.2. Computational fluid dynamics

The CANTERA software must also communicate with "higher-level" software such as computational fluid dynamics and multiphysics simulation tools, including commercial and open-source software tools. Such software interfaces will enable the incorporation of latest chemical kinetics theories and models into practical reactor or combustor development. For example, Fig. 17 illustrates a relatively complex CFD simulation of a combustor. Assuming that the CFD software accesses chemistry functions via CANTERA, the simulation could easily take advantage of new formulations in non-ideal equations of state, thermodynamics, and combustion kinetics. Such a capability enables the rapid and effective incorporation of new academic research into practical technology development.

In principle, because the core CANTERA functions are written as object-oriented C++, the incorporation into CFD models should be possible. Nevertheless, given the inherent complexity of both software sets, numerous practical challenges will need to be addressed and resolved. Some of these challenges are technical and some may well be connected with legitimate and pragmatic business interests of the CFD developers.

7.3. Algorithms and mathematics

The CANTERA software is designed to facilitate the simulation of chemically reacting processes across a broad range of potential applications. The CANTERA code itself is designed to represent aspects of physical chemistry and molecular transport in very general settings. However, to have practical utility in developing models for particular processes of interest, the chemical functions must integrate with higher level programming languages and computational mathematics.

For a majority of users, MATLAB or PYTHON serve as the programming language. Most CANTERA functions can be conveniently accessed from these programming platforms. Most users use the embedded computational mathematics within these packages.

The CANTERA software is distributed with a number of applications, such as low-dimensional flame models. These applications are typically written in C++, with the computational mathematics implemented using the open-source SUNDIALS software.

High level modeling software, including commercial CFD software, may use special-purpose algorithms or propriety solvers. However, beyond accessible interfaces, the core CANTERA functions should be independent of the application model and solution algorithms.

7.4. Parallel computing

As scientific problems increase in size and complexity, massively parallel computing hardware has emerged as an important enabling resource. Especially with large reaction mechanisms, solving CFD problems on serial computers can be prohibitively expensive. Thus, CANTERA must be configured to work in a parallel computing environment. An interesting aspect of parallel computing involves the dispatch of certain small, but computationally intensive, tasks to graphical processing units (GPU). To take advantage of these computing resources, CANTERA must be at least compatible with standard GPU-based languages such as CUDA and OpenCL. Prof. James Sutherland (University of Utah) and colleagues have developed a GPU-based library PoKiTT (Portable Kinetics, Thermodynamics, and Transport) that achieves significant efficiency improvements in chemical kinetics models [27]. Large-scale CFD simulations will benefit greatly by integrating CANTERA with software tools that work in the parallel and GPU environments.

8. CANTERA Community

While still in early development stages, Prof. David Goodwin began releasing versions of his CANTERA

software in the early 2000s. The potential value of the new software paradigm was recognized, especially among academics. Open-source, well-structured, object-oriented software formed the basis on which to grow and share future developments. Goodwin established a website from which the software could be downloaded. However, the installation process was complicated and there was very little documentation. Nevertheless, despite practical impediments, a Developer community and a User community began to grow.

The CANTERA enterprise operates without any officially sanctioned structure. Through extraordinary volunteer efforts, Dr. Ray Speth (MIT) has emerged as the de facto coordinator. Dr. Speth essentially maintains the core software and controls the public releases. A number of other capable individuals should also be identified as playing important roles in developing and supporting CANTERA. Identifying individuals can be thorny, because deserving individuals may be inadvertently not acknowledged. Nevertheless, a few of the active developers and support contributors include Profs. Bryan Weber (University of Connecticut), Steven DeCaluwe (Colorado School of Mines), Kyle Neimeyer (Oregon State), and Richard West (Northeastern).

8.1. Developer community

The CANTERA core developer community is relatively small. The low-level C++ software is complex, with serious production-scale development demanding the combination of in-depth knowledge in chemical physics and object-oriented programming. Today, there are fewer than ten core developers.

Application software is developed using the core capabilities. For example, a premixed flame simulation would be considered an application. There are many more application developers than there are core developers. Broadly speaking, application developers need to derive problem-specific conservation equations and implement computational solution algorithms. For applications with broad applicability, user interfaces and documentation need to be written as well.

Since 2004, 27 individuals have contributed code that has been incorporated into the released versions. Of these 27 contributors, ten contributors have made more than 10 commits. Since the beginning of 2016, CANTERA has received 634 commits from 15 contributors. To date in 2017, there have been 107 commits from 9 contributors. These contributions include new code and software revisions, corrections, etc.

Figure 18: CANTERA Workshop that was held in conjunction with the National Combustion meeting in Washington, DC in April 2017. The workshop was organized and facilitated by Drs. Ray Speth (MIT), Steven DeCaluwe (Colorado School of Mines), and Bryan Weber (University of Connecticut).

8.2. Support community

There is an active on-line volunteer support community. The Cantera Users' Group [cantera-users@googlegroups.com] hosts a very busy website where user's questions are posed and mostly answered. As a practical matter, a few individuals, who are generally also developers, answer most of the queries. Nevertheless, this is a highly valuable resource that facilitates rapid communication among the CANTERA community.

Since 2009, 1519 topics/threads have been posted on the users' group; in the first six months of 2017, 176 new threads were created, totaling 832 individual posts and 8304 views.

Dr. Ray Speth (MIT) plays a very major and extraordinarily valuable role in supporting CANTERA. Anyone who monitors the Users' Group email traffic will know that Dr. Speth handles a majority of the queries, with great care and patience. Calling specific attention to Dr. Speth is in no way meant to detract from the numerous other contributors who volunteer valuable assistance and advice via the Users' Group.

8.3. User community

Although the precise number of active users is difficult ascertain, there are certainly a few thousand users. Most are in academia. Since CANTERA v2.3.0 was released in January 2017, the binary installer packages have been downloaded over 5,500 times by users on Windows, macOS, and Linux platforms. Statistics are unavailable for clones of the source repository or installs using Homebrew on macOS.

Most users interact with CANTERA in two ways. One is to use applications (e.g., a flame model) to study aspects of combustion chemistry, interpret laboratory measurements, etc. The other is to develop simulations for particular circumstances. In most cases, the user is writing in MATLAB or PYTHON and accessing CANTERA functions. There are thousands of such programs being written, ranging from homework exercises to research initiatives. Most such programs do not have the generality needed to develop them further into supported and distributed applications.

9. Software Management

CANTERA software is already a large and complex body of software, which is expected to grow in technical scope and user acceptance. However, sustaining the growth and vitality of such an enterprise demands some supporting infrastructure. This can be especially challenging for *open-source* software where a continuing resource base (financial and staffing) must be established. Certainly there is a legitimate government role in supporting the development of advanced software tools that support a wider research base. However, most government funding agencies are not interested in maintaining, distributing, and supporting established software. Thus, practically speaking, some of enduring support must be identified.

9.1. Documentation

Good documentation is critically important to the widespread use and success of software packages. This is especially true for software that embodies complexity in the underpinning science. Certainly the extensive documentation that supports CHEMKIN is an important contributing factor in its long-standing success.

There are at least two important aspects to software documentation. One involves explaining the underlying science. Especially in research, successfully using software depends on the user fully understanding what the software is doing. For example, users of software that deal with supercritical fluids should really understand the supporting equations of state and thermodynamics. By contrast, users of software such as PHOTOSHOP really do not need to understand the theory of image manipulation. Thus, the documentation for CANTERA must explain, or at least state clearly, the theory that certain functions implement.

The other important aspect of documentation involves using the software itself. One of the first tasks that a new user encounters is the installation procedure. Software such as CHEMKIN or CANTERA are constructed with hundreds of low-level functions with particular purposes. To be practically useful, the user needs to know how to find and interact with the functions.

Historically, software was supported by written documentation in the form of user manuals. Today, very few commercial software packages ship with printed user manuals. For users who prefer a physical book, there are numerous user guides that are written privately and are available for sale. Such a model might be viable for high-level scientific software. In other words, there may be a sufficiently viable market to entice authors to write and sell documentation.

Without a doubt, electronic versions of documentation must be available. However, there are numerous alternatives. Perhaps the easiest is simply to provide pdf representations of a written document. Interactive documentation is perhaps more useful, but likely requires more effort to develop and support.

To some extent, open-source code can be self-documented using software tools such as Doxygen. However, the code's author must exercise the self-discipline to incorporate the appropriate in-line documentation. Then, the user must be able to locate and use the resulting documentation. So, while in-line documentation is surely a good idea and should be strongly encouraged, most users will find more traditional narrative-style documentation to be more friendly.

Clearly, any viable software package needs easily accessible documentation. Whether or not it is physically printed, the content must be written. As anyone who has written such documentation knows well, producing such documentation is a large and tedious task. Many, even most, software developers are less than enthusiastic about writing high-quality documen-

tation. The extensions to CANTERA will likely be written by a diverse set of developers with different sources and mechanisms of support. So, especially for open-source software without explicit commercial incentives, some means must be found to produce documentation.

9.1.1. Coordination and control

Especially with software that is developed and supported by a large and diverse community, coordination and control can be difficult. As with the software itself, some organization must be established with the responsibility for maintaining the core code and ensuring that updates and extensions are sufficiently verified and validated. Furthermore, documentation standards will require developers of new capabilities to deliver compliant documentation before those capabilities are included into formal software releases.

9.2. Open-source and licensing

Without any doubt, the core CANTERA software will be developed and maintained as open-source code. However, the particulars of how that happens need some attention. The current software is licensed using Berkeley Software Distribution (BSD-3), which has minimal use restrictions. All extensions that are adopted into the core capability will be licensed similarly using BSD-3. Future development is expected to proceed with a range of developers, who work for a variety of institutions. Thus, there could potentially be institutional issues that must be resolved concerning intellectual property agreements between employees (e.g., faculty) and institutions (e.g., universities). The author and the author's institution may hold copyright, and must surely be recognized for their contributions.

9.2.1. Berkeley Software Distribution licensing

The BSD license is widely used and accepted in open-source software. The license has legal standing and cannot be revoked. The BSD-3 license reads simply as:

> Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
>
> 1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the disclaimer below.
> 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the disclaimer (as noted below) in the documentation and/or other materials provided with the distribution.
> 3. Neither the name of [the Copyright holder] nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

9.2.2. Relevant precedents

The SUNDIALS (SUite of Nonlinear and DIfferential/ALgebraic Equation Solvers) software was developed and is maintained by the Lawrence Livermore National Laboratory (LLNL). The open-source code is widely used in the research and academic communities. Among other capabilities, SUNDIALS includes high-level, well-documented capabilities to solve stiff systems of ordinary differential equations. The software is copyrighted by LLNL and distributed as open-source code under a BSD license.

The CANTERA software uses the SUNDIALS software in several of its applications. This is a good example of the leverage associated with one high-level open-source software package (i.e., CANTERA) taking advantage of other supporting high-level software (i.e., SUNDIALS). Because both packages are written as object-oriented C++, integrating the packages is relatively straightforward.

9.2.3. Commercial use

There may be good reasons for the core open-source code to be embedded into commercial software, which is entirely acceptable. The SUNDIALS computational mathematics software, serves a good example.

The STAR-CCM+ software is a leading commercial CFD offering that is developed by CD-adapco. When STAR-CCM+ is used for combustion and chemical kinetics applications, solving stiff differential equations is central to the overall computational algorithms. Rather than develop new stiff differential equation software, CD-adapco incorporated the SUNDIALS solvers. Incorporating BSD-licensed software in no way compromises the proprietary nature of the CFD software that uses the open-source software.

Because the CANTERA core software is written in

object-oriented C++, it can interface comfortably with commercial CFD software (e.g., FLUENT, STAR-CCM+, or CONVERGE) or with open-source CFD software (e.g., OPENFOAM). Because of the BSD licensing, there are no practical restrictions to its incorporation into commercial highly capable CFD offerings. In this way, users can benefit from state of the art chemistry capabilities in the context of geometrically complex CFD simulation. For example, a gas-turbine designer could include CANTERA-based supercritical-fluid capabilities into the simulation of a combustor can using FLUENT. Of course, commercial CFD developers would need to enable and support such interfaces. There is precedent with commercial CFD developers for incorporating CHEMKIN capabilities.

9.3. Maintenance and control

CANTERA represents a large and complex body of software. As the software capabilities and user base grow, there is increasing need for formal mechanisms to maintain and support the enterprise. With open-source, noncommercial, software there are challenges in establishing and sustaining an effective support structure. In addition to technical challenges, some form of staffing and financial support is likely to be needed.

The viability of software packages depends on a user base that is confident that underpinning formulations are scientifically correct and that the implementations are bug-free. If errors are found, they must be repaired quickly. As new capabilities are implemented, some degree of backward compatibility is needed. Although recognizing the need for achieving such objectives is easy, practical aspects of realizing the objectives can be more difficult.

Looking ahead, new capabilities will likely be developed by a scientifically diverse and geographically separated set of researchers. Some control must be exercised over the process by which new contributions are incorporated into the core capability. There must be rigorous evaluation of the scientific correctness. Testing protocols must be established to assure that the software is correctly implemented. Accompanying documentation must be written, possibly complying with some formatting standards. As software evolves, formal protocols are followed for version control and public releases. In principle, all these functions are well understood and are critical aspects of commercial software development and maintenance. Especially as it grows, some formal organization is likely needed to manage the CANTERA initiative.

Currently, Dr. Ray Speth serves voluntarily, and very competently, as the de facto manager of the CANTERA software. With long-term viability in mind, there are good reasons to formalize and probably broaden the support structure. This in no way diminishes the value of uncompensated community support. Nevertheless, a compensated and dedicated support staff with well defined roles and responsibilities would benefit the overall growth, adoption, and viability of the CANTERA capabilities. Of course, some means of secure funds for compensation is required.

There are certainly successful precedents for supporting open-source software. Examples include LINUX and PYTHON. Foundations can be established, with different mechanisms for acquiring the funding needed to maintain a staff. Nonprofit organizations such as NumFOCUS (http://www.numfocus.org/about.html) serve as a vehicle to support financial aspects of open-source software projects. The NumFOCUS organization is not directly involved with the software per se, but rather it serves as a neutral third party for handling financial transactions such as donations. Prof. Kyle Niemeyer (Oregon State) has approached the NumFOCUS group and they have expressed interest in working with the CANTERA initiative.

10. Identification and prioritization of needs

Broadly speaking, developing "next-generation" capabilities and software is open-ended, potentially leading in many different directions. Thus, assuming limited availability of resources, some prioritization is appropriate. To the extent possible, understanding needs of the broader community and achieving some consensus among the stakeholders is valuable in establishing priorities and allocating resources.

The Theoretical Chemistry group and collaborators at Argonne National Laboratory (ANL) represent a good cross section of interested stakeholders. Following a meeting with that group in October 2016, Dr. Al Wagner and colleagues offered a written list of topics and initiatives that would be valuable in their research. Quoting directly, that list follows as:

- *Effectively parallelized code*: We believe the complete market for CHEMKIN would want parallelized code. We may want to run 10000 coupled runs for flame propagation. Currently even for small mechanisms it takes multiple days to reach full convergence in CHEMKINPRO with best parameters. This

leads to people making assumptions that are probably not warranted for some situations. CANTERA is not parallelized.

- *Fast Pre-conditioned ODE solvers*: Matt McNenly has shown the orders of magnitude speedup possible with the solution of pre-conditioned coupled kinetics equation solvers. Fast solvers would enable rapid and repeated solutions of kinetics equations that in turn would enable global sensitive analysis (GSA) and uncertainty quantification (UQ) studies to become more routine for a variety of studies such as ignition delay.

- *UQ hooks*: The ability to input in flexible ways the uncertainties in all input data would allow GSA and UQ studies. Such ways could include entering the total rate (often with low uncertainty) along with separate branching fractions (often with higher uncertainty). At a higher level of functionality, CHEMKIN could be aware of common UQ software packages (e.g., DAKOTA) and develop input/output options that would allow easy communication between CHEMKIN and these packages.

- *More flexible collision efficiency expressions*: Currently CHEMKIN does not accept many appropriate expressions for the pressure dependence of multiple colliders, nonlinear mixture rules, temperature dependence colliders.

- *Flexibility beyond Lennard-Jones in describing buffer gas effects*: The Lennard-Jones form is more convenient than accurate.

- *More general kinetics expressions*: Non-thermal kinetics and other kinetic phenomena require expressions that are currently not supported.

- *Improved polynomial expressions for thermodynamics*: Currently polynomials with correct asymptotics and improved accuracy are not supported.

- *Checks on impossible rate constants*: In the forward or reverse direction, rates can obviously not exceed the collision frequency. The user should be alerted to input rates that violate such limits and be given options on how to proceed.

- *Multiple optional representations of equation of state:* The current version of CANTERA with the ideal-gas and Redlich–Kwong equations of state must adopt additional analytical equations of state to capture high-pressure behavior across fluid phases. Adoption of quantitative equations of state for real fluids can enable new features associated with phase separation and transformations that are critical in multiphase reaction systems

Thoughtful and specific input from groups and individuals, such as the ANL group, will be extraordinarily valuable in setting priorities and directing future development. As the planning process continues and matures in the coming year, additional such dialog and feedback will be sought.

11. Outlook and recommendations

There are clearly needs and opportunities to significantly expand capabilities to model chemically reacting processes. The present roadmap identifies needs in three broad areas — non-ideal fluids, heterogeneous catalysis, and electrochemistry. The CANTERA software provides a sound foundation on which to build and maintain these and other new capabilities.

CANTERA's open-source and object-oriented architecture, developed initially by Prof. David Goodwin (Caltech), provides an excellent platform for continued growth. Over the past decade, a significant and active user community (over 1000 users) has grown. There is also an active developer community, with approximately ten individuals working on supporting the core code. Although there is no formal support organization, Dr. Ray Speth (MIT) plays a major role in coordinating software maintenance. The user community is currently supported via voluntarily internet communication.

Looking ahead, the CANTERA development and maintenance would benefit from a more formal organizational structure. As the capabilities grow and the user base increases, the complexity of managing the enterprise increases. While in no way diminishing the value of volunteer community support, managing and controlling large and complex software packages benefits from a dedicated staff. Among the needed functions are validation and verification of new capabilities, version control, formalizing new releases, backward compatibility, bug fixes, installer protocols on diverse computing platforms, producing documentation, user help functions, etc. Of course, all these functions are supported in commercial software. Establishing the sustaining resources to manage widely used open-source software is more challenging, but can be done.

One can imagine numerous opportunities to enhance modeling capabilities and software implementations. Thus, keeping practical resource limitations in mind, it is important to prioritize the research and development efforts. At least in the short term, development likely needs to be government supported. Thus, needs of the particular funding agents help to establish priorities. The present planning is greatly influenced by energy-conversion processes, as are perceived to be important to the Department of Defense and the Department of Energy.

In addition to funding-agency priorities, there needs to be broader input from the scientific communities. For the enterprise to grow and remain vital, there needs to be active "buy-in" from the user community. The stakeholders need to see that the generalized capabilities are directly responsive to their particular needs. Thus, it is imperative to keep communication channels open, such that the developers are cognizant of the broader scientific needs.

The needed research and development efforts can be accomplished within academia and at National Laboratory facilities. In some cases, one can imagine that software development falls within the scope of ongoing projects and grants. In other cases, one can imagine a solicitation and proposal process that seeks to identify creative new ideas and directions.

12. Acknowledgements

This research is supported by the Air Force Office of Scientific Research via grant FA9550-16-1-0349. We gratefully acknowledge the insightful encouragement and support of Dr. Chiping Li (AFOSR). Underpinning research on approaches to electrochemistry modeling are supported by the Office of Naval Research (Dr. Michele Anderson) via grant N00014-16-1-2780.

13. References

[1] O. Redlich and J.N.S. Kwong. On the thermodynamics of solutions. V. An equation of state. Fugacities of gaseous solutions. *Chem. Rev.*, 44:233–244, 1949.

[2] R.C. Reid, J.M. Prausnitz, and B.E. Poling. *Properties of Gases and Liquids*. Mc.Graw–Hill Inc., 4th edition, 1987.

[3] G.M. Kogekar, C. Karakaya, G.J. Liskovich, M.A. Oehlschlaeger, S.C. DeCaluwe, and R.J. Kee. Impact of non-ideal behavior on ignition delay and chemical kinetics in high-pressure shock tube reactors. *Under review at Comb. Flame*, 2017.

[4] R.J. Kee, M.E. Coltrin, P. Glarborg, and H. Zhu. *Chemically Reacting Flow: Theory, Modeling and Simulation*. Wiley, Hoboken, NJ, 2nd edition, 2018.

[5] G. Soave. Equilibrium constants from a modified Redlich–Kwong equation of state. *Chem. Eng. Sci.*, 27:1197–1203, 1972.

[6] D.Y. Peng and D.B. Robinson. A new two-constant equation of state. *Ind. Eng. Chem. Fund.*, 15:59–64, 1976.

[7] G. Lacaze and J.C. Oefelein. A non-premixed combustion model based on flame structure analysis at supercritical pressures. *Combust. Flame*, 159:2087–2103, 2012.

[8] W. Tang and K. Brezinsky. Chemical kinetic simulations behind reflected shock waves. *Intl. J. Chem. Kinet.*, 75-97:38, 2005.

[9] R.G.Schmitt, P.B.Butler, and N.B.French. *Chemkin Real Gas: A Fortran Package for Analysis of Thermodynamic Properties and Chemical Kinetics in Nonideal Systems*. University of Iowa, 1994.

[10] G.F. Froment and K.B. Bischoff. *Chemical Reactor Analysis and Design*. Wiley, NewYork, 1979.

[11] C.A. Eckert and M. Boudart. On the use of fugacities in gas kinetics. *Chem. Eng. Sci.*, 18:144–147, 1963.

[12] L. Yang, S. Wang, K. Blinn, M. Liu, Z. Liu, Z. Cheng, and M. Liu. Enhanced sulfur and coking tolerance of a mixed ion conductor for sofcs: $BaZr_{0.1}Ce_{0.7}Y_{0.2-x}Yb_xO_{3-\delta}$. *Science*, 326:126–129, 2009.

[13] C. Karakaya, H. Zhu, and R.J. Kee. Kinetic modeling of methane dehydroaromatization chemistry on Mo/Zeolite catalysts in packed-bed reactors. *Chem. Engr. Sci.*, 123:474–486, 2015.

[14] O. Deutschmann. *Modeling and Simulation of Heterogeneous Catalytic Reactions: From the Molecular Process to the Technical System*. Wiley-VCH, Weinheim, Germany, 2012.

[15] D.G. Goodwin, H. Zhu, A.M. Colclasure, and R.J. Kee. Modeling electrochemical oxidation of hydrogen on Ni-YSZ pattern anodes. *J. Electrochem. Soc.*, 156:B1004–B1021, 2009.

[16] E.S. Hecht, G.K. Gupta, H. Zhu, A.M. Dean, R.J. Kee, L. Maier, and O. Deutschmann. Methane reforming kinetics within a Ni-YSZ SOFC anode. *Appl. Catal. A*, 295:40–51, 2005.

[17] H. Zhu, R.J. Kee, V.M. Janardhanan, O. Deutschmann, and D.G. Goodwin. Modeling elementary heterogeneous chemistry and electrochemistry in solid-oxide fuel cells. *J. Electrochem. Soc.*, 152:A2427–A2440, 2005.

[18] D. Aurbach, B. Markovsky, I. Weissman, E. Levi, and Y. Ein-Eli. On the correlation between surface chemistry and performance of graphite negative electrodes for Li ion batteries. *Electrochim. Acta*, 45(1-2):67–86, 1999.

[19] G.-H. Kim, A. Pesaran, and R. Spotnitz. A three-dimensional thermal abuse model for lithium-ion cells. *J. Power Sources*, 170:476–489, 2007.

[20] S.-P. Kim, A.C.T. van Duin, and V.B. Shenoy. Effect of electrolytes on the structure and evolution of the solid electrolyte interphase (SEI) in li-ion batteries: A molecular dynamics study. *J. Power Sources*, 196:8590–8597, 2011.

[21] A.M. Colclasure, K.A. Smith, and R.J. Kee. Modeling detailed chemistry and transport for solid-electrolyte-interface (SEI) films in li-ion batteries. *Electrochim Acta*, 58:33–43, 2011.

[22] N. Tanaka and W.G. Bessler. Numerical investigation of kinetic mechanism for runaway thermo-electrochemistry in lithium-ion cells. *Solid State Ionics*, 262:70–73, 2014.

[23] Y. Xie, J. Li, and C. Yuan. Multiphysics modeling of lithium ion battery capacity fading process with solid-electrolyte interphase growth by elementary reaction kinetics. *J. Power Sources*, 248:172–179, 2014.

[24] P. Guan, L. Liu, and X. Lin. Simulation and experiment on solid electrolyte interphase (SEI) morphology evolution and lithium-

ion diffusion. *J. Electrochem. Soc.*, 162:A1798–A1808, 2015.

[25] P.J. Spencer. A brief history of CALPHAD. *CALPHAD-Computer Coupling of Phase Diagrams and Thermochemistry*, 32:1–8, 2008.

[26] C.W. Bale, E. Belisle, P. Chartrand, S.A. Decterov, G. Eriksson, A.E. Gheribi, K. Hack, I.-H. Jung, Y.-B. Kang, J. Melancon, A.D. Pelton, S. Petersen, C. Robelin, J. Sangster, P. Spencer, and M-A. Van Ende. FactSage thermochemical software and databases, 2010-2016. *CALPHAD-Computer coupling of phase diagrams and thermochemistry*, 54:35–53, 2016.

[27] N. Yonkee and J. Sutherland. PoKiTT: Exposing task and data parallelism on heterogeneous architectures for detailed chemical kinetics, transport, and thermodynamics calculations. *SIAM J Sci. Comp.*, 38:S264–S281, 2016.

[28] G.P. Smith, D.M. Golden, M. Frenklach, N.W. Moriarty, B. Eiteneer, M. Goldenberg, C.T. Bowman, R.K. Hanson, S. Song, W.C. Gardiner, V. Lissianski, and Z. Qin. Gri-mech—an optimized detailed chemical reaction mechanism for methane combustion. Technical report, Gas Research Institute, 1999.

[29] C. Karakaya, B. Zohour, H. Zhu, S. Senkan, and R.J. Kee. Detailed reaction mechanisms for oxidative coupling of methane (OCM) over La_2O_3/CeO_2 nanofiber fabric catalysts. *ChemCatChem.*, Accepted, 2017. doi: 10.1002/cctc.201701172.

A. General overview of Cantera

Cantera is a suite of object-oriented computational tools that incorporate "basic" thermodynamics, transport, and kinetics functionalities. The software is designed to represent aspects of physical chemistry and molecular transport in very general settings, meaning that different chemical mixtures and reactions can be easily accommodated. In broad-brush terms, the basic functionalities are intended to be used to evaluate physical and chemical attributes (e.g., equations of state, enthalpies, entropies, conductivities, diffusion coefficients, chemical production rates,...) as functions of thermodynamic states (e.g., pressure, temperature, composition,...).

Consider, for example, the use of Cantera in modeling a chemical vapor deposition (CVD) reactor. The developer of the CVD model begins by deriving appropriate conservation equations, typically represented as partial differential equations (PDE). Assuming computational solution, the PDEs are discretized spatially on a finite-volume mesh network. The dependent variables typically represent the system state at each of the discretized finite volumes. Terms in the conservation equations involve thermodynamic and transport properties as well as chemical reaction rates. The model developer implements a solution algorithm, which is typically iterative. At each stage of the iteration, the local state is used to evaluate the conservation equations. The generalized Cantera functions effectively and conveniently assist evaluating terms as functions of state, providing great flexibility and efficiency in formulating models.

A particular CVD reactor may be used for very different chemical processes. For example, a stagnation-flow reactor might be used to grow doped epitaxial silicon for CMOS semiconductor applications or it may be used to grow doped gallium nitride for solid-state lighting applications. Of course, the chemistries for these processes are very different. However, there is great value in writing predictive models that can switch easily between different chemical processes, but in the same reactor configuration. Cantera is designed and written to serve this need.

Cantera is written in C++, which is a widely used object-oriented language. As such, there is great flexibility in applying, adapting, and maintaining the models. However, understanding Cantera's architecture requires some familiarity with aspects of object-oriented programming (OOP). Capabilities such as Chemkin are written in Fortran, which is a *procedural* language. Even in circumstances where the chemical functionality is equivalent, the Chemkin and Cantera code can appear to be quite different.

A.1. Object-oriented structure

Cantera contains numerous *classes*, representing elements, species, phases of matter, equations of state, interfaces between phases, molecular transport, chemical kinetics, etc. Each class contains parameters (e.g., the species class contains atomic compositions, molecular weights, etc.) and operations or methods. For example, the role of the thermodynamics *class* is to represent the state of a phase (i.e., pressure, temperature, density, composition,...) and enable the evaluation of thermodynamic properties (e.g., heat capacity, enthalpy, Gibbs free energy,...) that are consistent with the state. *Methods* are the functions and mathematical operations within the thermodynamics *class* that enable a user to set the state and evaluate thermodynamic properties. An *object* is a particular *instantiation* of the *class*, meaning the class has been made particular to a specific chemical system. In other words, an *object* for air (i.e., N_2, O_2, CO_2, Ar,...) can be created from the ideal-gas *class*.

Cantera uses object-oriented concepts to create a modular computational framework that facilitates information sharing between *classes*. For example, a species *class* contains parameters and functions that are required by the thermodynamics *class* to evaluate thermodynamic properties. Thermodynamic properties

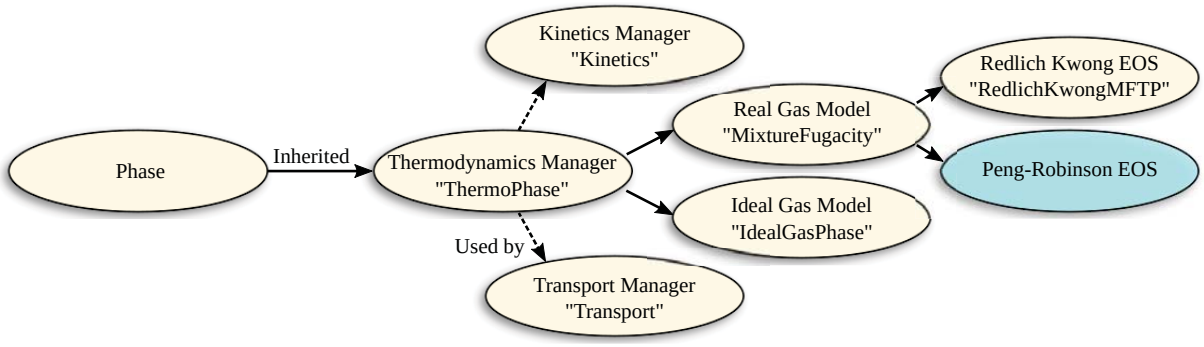

Figure 19: Inheritance diagram between real-gas equation-of-state *classes* and the main phase *class*. The solid arrows indicate inheritance from the *parent* class to the *child* class. The dashed arrows indicate that the thermodynamics manager is used by the kinetics and transport managers.

from the thermodynamics *class* are then needed to evaluate reaction rates, which is accomplished in the kinetics *class*. C++ enables information exchange between *classes* by allowing one *class* to access directly the methods of another.

Using *inheritance* between *classes* enables new models to be implemented while preserving common *methods* and ensuring consistent syntax. When one class (the *child*) is derived from another (the *parent*), it *inherits* all of the parameters and *methods* of the *parent class*. These parameters and *methods* can be overwritten in the *child class* to enable new functionality. For example, while various reaction types (e.g., bimolecular reactions, three-body reactions, charge-transfer reactions) have unique *methods* to evaluate rates of progress, the *method* to evaluate net production rates based on the rate of progress is common to all reaction types. Therefore *methods* to evaluate net production rates should be written in a *parent class* such that they are automatically incorporated in all *child* reaction types.

A.2. Cantera managers

Cantera's source code is organized into three manager *classes* that represent phase behavior. The thermodynamic, transport, and kinetic managers are the *classes* that form a basis to represent a main "Phase" *class*. These managers contain functions that can evaluate the state of phases, chemical production rates, and species transport.

A phase *object* is created for each user-specified bulk, surface, or edge element. For example, in a heterogeneous reactor the user would specify a minimum of two phase *objects* (i.e., a bulk phase *object* and a surface phase *object*). Each of these phase *objects* have thermodynamic, kinetic, and transport managers that model the phase's physics.

A.2.1. Thermodynamics manager

The thermodynamics manager ("ThermoPhase" *class*) handles the thermodynamic state of the phase. Attributes of this class include species composition, elements, and thermodynamic properties such as temperature, composition, and density. Other common properties such as Gibb's free energy and enthalpy are also handled by this class. Specific thermodynamics models are implemented as *classes* that inherit from the thermodynamic manager. Kinetic and transport managers communicate with thermodynamics manager to calculate and update state properties.

Figure 19 illustrates a diagram of *parent-child* relationships between the main *parent* "Phase" *class* and real-gas models. The hierarchical structure available in C++ enables the ideal ("IdealGasPhase" *class*) and non-ideal gas phases ("MixtureFugacity" *class*) to be defined as extensions to the *parent* ThermoPhase *class*. The MixtureFugacity *class* can be further extended to specify different ideal-gas models. Illustrated in blue is the Peng–Robinson EoS *class* that is not yet implemented in Cantera. This class would inherit all *public attributes* and *methods* from the Phase, ThermoPhase, and MixtureFugacity *parent classes*.

It is illustrative to analyze the thermodynamics manager's hierarchy through a simple example. Suppose the user defined a bulk phase *object* named "Air." This *object* is an *instance* of the ThermoPhase *class*, which is inherited from the *instance* of the "Phase" *class*. The user can then invoke *methods* from the ThermoPhase *class* that extract the thermodynamic properties or com-

pute equilibrium state (e.g., accessing the mole fractions of "Air" would show 21% O_2, 78% N_2, and 1% Ar). The user must also define the equation of state that represents the system. If, for example, the user defines the "Air" *object* as being an ideal gas, then the "IdealGasPhase" *class* is used to compute the equation of state relations. As shown in Fig. 19, "Air's" IdealGasPhase *object* is a *child* of the thermodynamics manager. Because IdealGasPhase is a *child* of the thermodynamics manager, the IdealGasPhase automatically incorporates all of the methods and parameters in the thermodynamics manager. In addition to the *methods* and parameters *inherited* through the thermodynamics manager, the "IdealGasPhase" has specific *methods* to handle the equation of state.

A.2.2. Kinetics manager

The kinetics manager ("Kinetics" *class*) handles both homogeneous and heterogeneous chemical reactions. The kinetics manager communicates between the Kinetics and "Reaction" *classes* to access reaction rate parameters and evaluate reaction rates. *Derived classes* (*children*) of the Reaction *class*, such as "ElectrochemicalReaction" and "ThreeBodyReaction," handle specific types of reactions and their associated rate expressions. The kinetics manager obtains thermodynamic parameters needed in reaction kinetics from the thermodynamics manager.

A.2.3. Transport manager

The transport manager ("Transport" *class*) handles transport properties such as viscosity, conductivities, and diffusion coefficients. The *child classes* are primarily divided on the transport medium (gas, liquid, porous media, etc.). *Classes* such as "DustyGasTransport" are derived from the Transport *class* to calculate transport properties using particular transport models. The transport manager calculates mass and molar fluxes between two different thermodynamic states. The transport manager communicates with the thermodynamics manager to determine transport rates based on thermodynamic states.

A.3. Acquiring and installing Cantera

Cantera installation is supported for the major operating systems, including Mac OS, Windows, and Linux. The Cantera source code may be downloaded from Github, which is an internet hosting service for open-source software. Installers are available for the aforementioned operating systems, or Cantera can be built manually from source code.

The build and installation process for Cantera is preceded by installation of several additional dependencies, including Python, Numpy, and Scons (for Mac OS installations). After subsequent installation of the Cantera modules, setting relevant environment variables and file paths (for Matlab compatibility) completes the installation process. Pre-compiled binaries exist for Windows installations, while Mac OS installations can use programs such as Homebrew or MacPorts. For all operating systems (Windows, Mac OS, and Linux) Conda Python distributions represent the simplest means to install Canteras Python interface. Cantera and all required dependencies are built using just two command-line entries.

A.4. Using Cantera

Broadly speaking, a typical Cantera user interacts with the software in two ways. The first requires preparing an input file that specifies information about species thermodynamics and transport and reaction chemistry. The so-called "cti" (Cantera input) file is an editable text file that complies with certain formatting and keyword requirements. There are also provisions to input the information as as "xml" file. The second interaction requires writing a program (e.g., a Matlab script) that accesses functions within the core Cantera software. The user program (an Application) typically implements the formulation and solution of some physical problem, often represented as the computational solution of some set of conservation equations.

This input file (cti or xml format) describes a particular chemical system, including thermodynamics, transport, and kinetics models, together with associated parameters. Quite often, users begin with a previous file that may be used as is or modified. For example, among many others, Cantera provides two frequently used files—GRI30.cti and Air.cti. The GRI30.cti file specifies the GRI-3.0 reaction mechanism [28] and Air.cti specifies the composition of air.

Figure 20 illustrates the overall information flow in a Cantera-based flame model. The left-hand side illustrates aspects of a cti file for a flammable gas. The top-most box defines the chemical species used in the model. Each species has user-defined name, which in this illustration is "H2." The next line specifies the

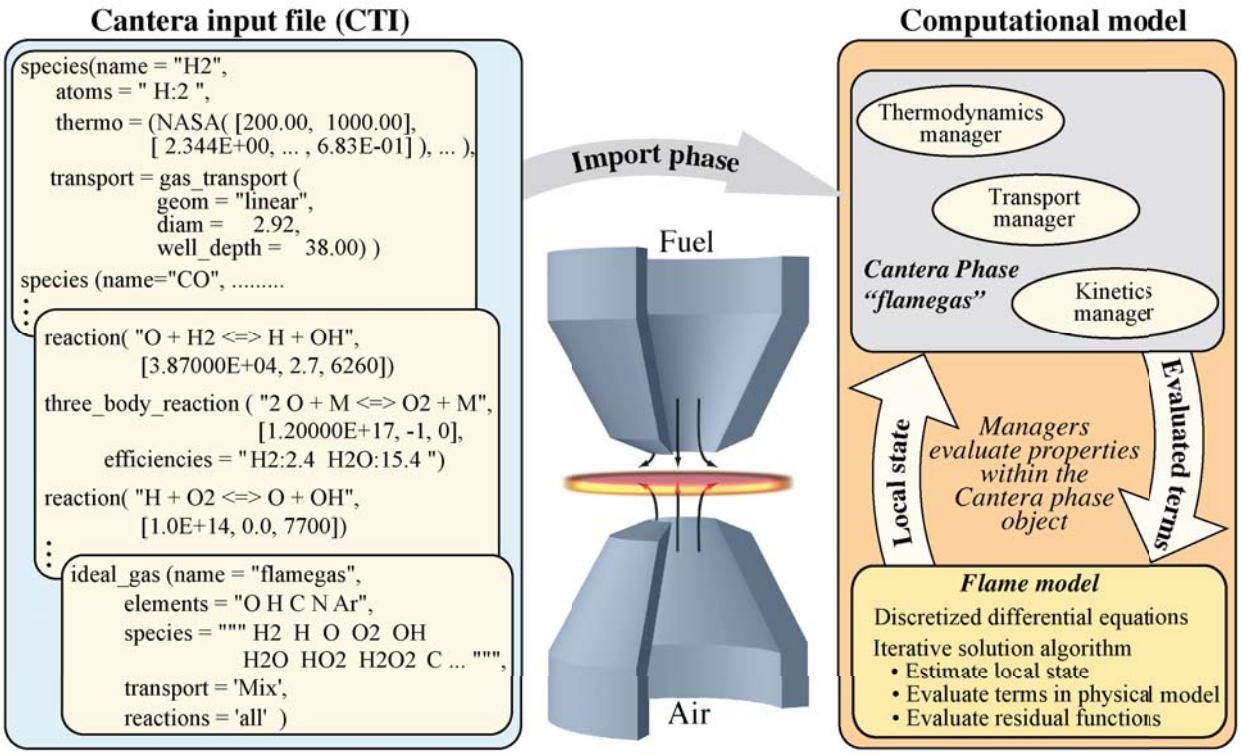

Figure 20: Example of CANTERA's "cti" file format importing to CANTERA's computational model. The pseudo-code (shown on the left) illustrates modular species, reaction, and phase specifications. The model structure (shown on the right) illustrates CANTERA's three main class managers along with a description of CANTERA's one-dimensional flame model.

atomic composition, showing 2 hydrogen atoms. The following lines specify thermodynamic properties for each species. In this case, the thermodynamics (heat capacity, enthalpy, and entropy) are specified in terms of the widely used NASA polynomials with 14 coefficients. The species transport properties are specified in terms of parameters, including the Lennard-Jones well depth and collision diameter.

The middle box on the left-hand side of Fig. 20 illustrates how chemical reactions are specified. Each reaction is described as a character string involving the species, followed by three numbers that represent parameters in the modified Arrhenius rate expression. In the case of a three-body reaction, the Arrhenius parameters must be augmented by species third-body efficiencies.

The lower box on the left-hand side of Fig. 20 illustrates a phase definition. In this illustration, an "ideal_gas" phase is named "flamegas." The phase definition must identify all the elements and species in the phase. The transport model in this illustration is defined as 'Mix,' indicating that the mixture-averaged approximation will be used evaluating transport properties. The notation 'all' reactions indicates that all reactions in the cti file will be included in the phase. More complex cti files may involve multiple phases, with different phases involving different species, transport models, reactions, etc.

The right-hand side of Fig. 20 illustrates how CANTERA functionality is used in an application, such as a flame model. The right-hand box is meant to represent a user-written model, such as would be implemented in a programming language (e.g., C++, MATLAB, or PYTHON). The upper box on the right-hand side of Fig. 20 illustrates the CANTERA functionality in terms of class Managers that interact with the Phase *object*. As discussed in previous sections, CANTERA's three main class managers consider thermodynamics, transport, and kinetics. The class managers are used to evaluate properties and terms that appear in the conservation equations. Typical solution algorithms involve iteratively updating states (e.g., pressure, temperature, composition) and evaluating residual forms of the conservation equations. The

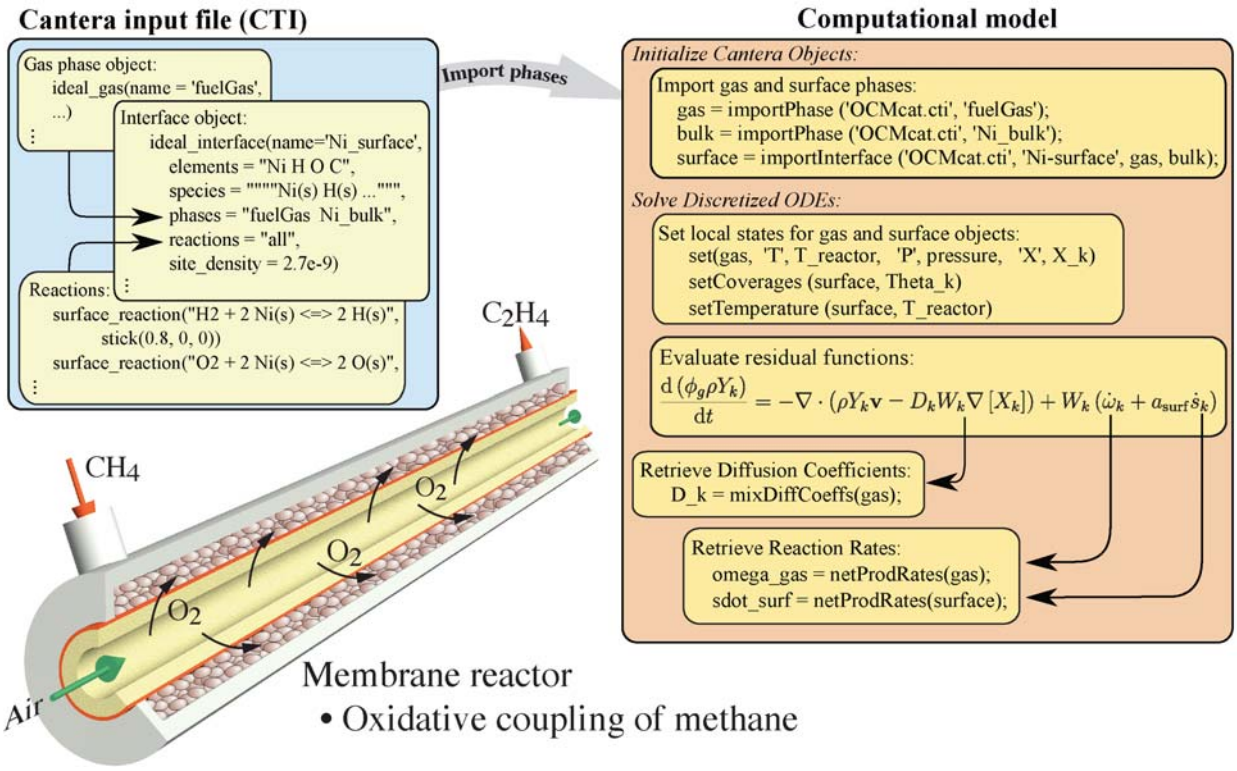

Figure 21: Membrane reactor example implemented using CANTERA. The top left "cti" file illustrates phase, interface, and reaction object pseudo-code. The right side illustrates pseudo-code for a computational model that implements CANTERA.

role of CANTERA is to evaluate thermodynamic, transport, and kinetics as functions of varying problem states as the iteration proceeds.

Figure 21 illustrates the model of a catalytic membrane-reactor for oxidative-coupling of methane [29], which is more complex than the flame problem represented in Fig. 20. The left-hand side summarizes aspects of the cti input file. This problem involves a gas-phase *object* and a Ni-surface *object*. Of course, the surface and bulk phases involve models and parameters that are different from the gas-phase *object*. Heterogeneous surface reactions on the Ni surface are quite different from the gas-phase reactions.

The right-hand side of Fig. 21 illustrates aspects of the user-written application that interacts with the CANTERA functionality. The first step is to import user-specified chemistry information (i.e., represented in the cti file) into CANTERA and generate phase objects gas, bulk, and surface. The "set" functions interact with the Managers to adjust the phase *objects* to the local states. Once the states are set, the phase *objects* can be exercised to evaluate relevant aspects of the *object* (e.g., diffusion coefficients, species production rates via chemical reactions), which, in turn, are used for evaluating terms in the conservation equations.

Because of its object-oriented structure, CANTERA enables the relatively straightforward development of new applications. As an example, consider a model problem in which a fuel-air mixture is combusted at constant volume and then expanded isentropically. For the purpose of this illustration, assume that the chemistry remains in equilibrium. Figure 22 shows a MATLAB function called "isentropicExpansion" that interacts with CANTERA to solve this problem. The first line of the function imports the phase, which here is the GRI30 phase. The next few lines define an initial state (in SI units). The pressure is set to one atmosphere, using the predefined CANTERA variable "oneatm." The mixture composition is set to be lean methane-oxygen at an equivalence ratio of $\phi = 0.8$. Even though the specified mole fractions don't sum to unity, CANTERA will do the renormalization. The gas phase is then defined using the "set" method with the property-value pairs syntax. The composition can also be set using a column vector of mole

```
function isentropicExpansion()
     % Import phase
     gas = GRI30;

     % Set initial state
     T_in = 25+273.15;      % Kelvin
     P_in = oneatm;         % Pascal
     X_in = 'CH4:1,O2:2.5'; % Moles
     set(gas,'T',T_in,...
             'P',P_in,...
             'X',X_in    );

     % Equilibrate at constant
     % internal energy and pressure
     equilibrate(gas,'UV');

     % Retrieve post combustion properties
     S_combusted = entropy_mass(gas);
     X_combusted = moleFractions(gas);

     % Expand to exhaust pressure
     P_exhaust = oneatm; % Pascal
     set(gas,'S',S_combusted,...
             'P',P_exhaust  ,...
             'X',X_combusted    );

     % Retrieve properties at exhaust
     T_exhaust = temperature(gas);
end
```

Figure 22: MATLAB code for defining a phase, operating on it, then retrieving the new state.

fractions instead of the string input. The gas mixture is then equilibrated using the "equilibrate" method. The keyword 'UV' specifies that the equilibration is constrained at fixed volume and constant internal energy (other choices are also available, such as 'TP' to specify temperature and pressure). The gas object has now been updated with the new equilibrium state. The thermodynamic properties can be evaluated with the "temperature", "pressure", "moleFractions", and "entropy_mass" methods. To expand the gas at constant entropy, the gas is specified to maintain same entropy but with an exhaust pressure of atmospheric pressure, once again using the "set" method. Finally, the exhaust state can be examined using the methods such as "temperature".

On the fundamental and practical aspects of modeling complex electrochemical kinetics and transport

Steven C. DeCaluwe[a], Peter J. Weddle[a], Huayang Zhu[a], Andrew M. Colclasure[b], Wolfgang Bessler[c], Gregory S. Jackson[a], Robert J. Kee[a,*]

[a]*Department of Mechanical Engineering, Colorado School of Mines, Golden, CO 80401, United States*
[b]*National Renewable Energy Laboratory, Golden CO 80401, United States*
[c]*Institut für Energiesystemtechnik, Hochschule Offenburg, 77652 Offenburg, Germany*

Abstract

Numerous technologies, such as batteries and fuel cells, depend on electrochemical kinetics. In some cases, the responsible electrochemistry and charged-species transport is complex. However, to date, there are essentially no general-purpose modeling capabilities that facilitate the incorporation of thermodynamic, kinetic, and transport complexities into the simulation of electrochemical processes. A vast majority of the modeling literature uses only a few (often only one) global charge-transfer reactions, with the rates expressed using Butler–Volmer approximations. The objective of the present paper is to identify common aspects of electrochemistry, seeking a foundational basis for designing and implementing software with general applicability across a wide range of materials sets and applications. The development of new technologies should be accelerated and improved by enabling the incorporation of electrochemical complexity (e.g., multi-step, elementary charge-transfer reactions and as well as supporting ionic and electronic transport) into the analysis and interpretation of scientific results. The spirit of the approach is analogous to the role that CHEMKIN has played in homogeneous chemistry modeling, especially combustion. The CANTERA software, which already has some electrochemistry capabilities, forms the foundation for future capabilities expansion.

Keywords: Marcus theory, Butler–Volmer kinetics, CANTERA, Ion and electron transport, Faradaic and Ohmic heating, General-purpose software

Contents

*Corresponding author: Tel: (303) 273-3379
Email address: rjkee@mines.edu (Robert J. Kee)

1. Introduction

Electrochemistry plays a central role for technologies in energy conversion, energy storage, and material/chemical processing. Corrosion processes and their mitigation are also grounded in electrochemistry. Electrochemical systems such as batteries and fuel cells can provide primary or auxiliary power with high efficiency and low environmental impact for a range of vehicular or stationary applications. However, improved energy density, durability, and safety are needed to enable broader deployment. Such technological advances must build on fundamental understanding of electrochemical processes over many scales. Flexible and robust computational tools can play a key supporting role in accelerating that understanding and promoting technology breakthroughs. Advances in atomistic computational modeling have enabled the identification of materials with finely tuned properties at the nanoscale [1–5], but the lack of concurrent development of modeling frameworks at the device scale impedes the translation of these materials into practical electrochemical devices and systems.

With only a few exceptions, the electrochemistry modeling literature is based on using a very few (often one per electrode) global charge-transfer reaction steps [6–9]. In reality, the heterogeneous chemistry and electrochemistry is far more complex. For example, in Li-ion batteries, the formation of solid-electrolyte-interface (SEI) films is known to involve complex chemistry and electrochemistry [3, 10, 11]. Solid-oxide fuel cells operating on hydrocarbon fuels are typically modeled with two global reforming reactions and one charge-transfer reaction. The actual chemistry is significantly more complex. Beyond these, numerous other technology examples involving electrochemistry can benefit from more complete electrochemical modeling and simulation tools. However, new software frameworks are needed to handle the neccessary complexity.

The lack of general software for expressing electrochemical complexity limits the practical ability to introduce such complexity into experimental interpretation and technology development. This is not to be critical of current modeling efforts, which are certainly effective and useful. Rather, it expresses the opportunities for significant and beneficial improvements. The primary objective of the present paper is to identify the scientific needs and to explore opportunities for computational implementation. New modeling capabilities should be closely coupled with functional relationships, properties, and parameters that can be readily extracted from atomistic modeling or basic analytical measurements, rather than from empirical parameter fits that are difficult to generalize.

Fundamental and theoretical underpinnings for electrochemistry, which are generally documented in textbooks [12–15] and a vast archival literature, certainly cannot be fully explained in a few pages. Thus, the intent of the present writing is to highlight areas where new modeling and software capabilities can offer substantial value. Drawing from illustrative applications in batteries, fuel cells, electrolyzers, and membrane reactors, the discussion shows how development and deployment of new generalized software capabilities can enable modeling of complex electrochemical processes, significantly assisting the design of new high-performance electrochemical systems and thereby accelerating the pace of technology development.

2. Illustrative examples

Before discussing the important quantitative relationships needed for electrochemical modeling, it is useful to consider illustrative examples of electrochemical devices. The example devices presented here vary in chemical and electrochemical complexity. By analyzing how these example devices are modeled, overarching modeling needs and capabilities for next-generation software become evident.

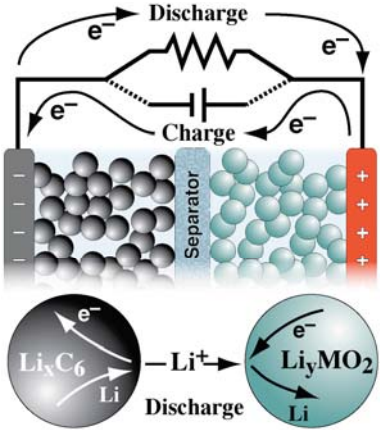

Figure 1: Illustrative aspects of a Newman Li-ion battery model.

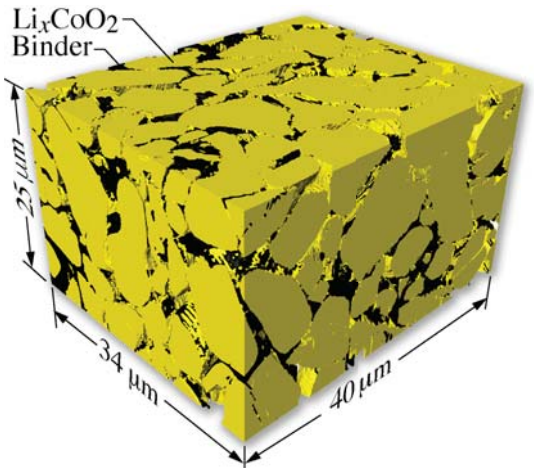

Figure 2: Three-dimensional reconstruction of a commercial Li_xCoO_2 cathode. The microscale reconstruction was created using Focused-Ion-Beam–Scanning-Electron-Microscopy (FIB-SEM). This image was produced by Prof. Scott Barnett, Northwestern University.

2.1. Lithium-ion batteries

Lithium-ion batteries represent a widely used and rapidly changing electrochemistry technology. With only a few exceptions, the majority of the modeling literature is based on an approach developed by Newman and colleagues [12, 16, 17]. Figure 1 illustrates the central tenets of the Newman model. During discharge, an electrochemical charge-transfer reaction removes Li from the anode phase, delivering Li^+ ions into an electrolyte solvent and electrons into the anode phase. The Li^+ ions are transported within the electrolyte solvent by diffusion and migration through the separator into the cathode side of the cell. Charge transfer chemistry reacts a Li^+ from the electrolyte phase with an electron from the electrode phase (coming from an external circuit) to deliver charge-neutral Li into the cathode phase. The electrodes themselves are assumed to be composed of spherical graphite particles for the anode and spherical metal-oxides for the cathode. The Newman model is frequently referred to as a 'pseudo-2D' model: electronic and ionic charge transport (in the electrode and electrolyte phases, respectively) are modeled in the 'through-cell' direction, while charge-neutral lithium intercalates within the particles via diffusion.

Charge-transfer chemistry at the electrode-electrolyte interfaces is usually modeled using a global reaction as

$$\mathrm{Li(ed)} \rightleftharpoons \mathrm{Li^+(el)} + \mathrm{e^-(ed)}, \qquad (1)$$

where Li(ed) represents a charge-neutral Li incorporated in the electrode, Li^+(el) represents a Li ion within the electrolyte, and e^-(ed) represents an electron within the electrode phase. As written, the forward direction is anodic, meaning that the reaction produces electrons. This would be the case for a battery anode during discharge or a battery cathode during charging. Although this reaction may be a reasonable approximation under many circumstances, it is certainly a simplification of the actual chemistry.

An alternative representation of the charge-transfer chemistry employs a two-step process, which may be expressed as [9, 18]

$$\mathrm{Li(ed)} + \mathrm{(s)} \leftrightarrows \mathrm{Li(s)} + \mathrm{V_{Li}(ed)}, \qquad (2)$$

$$\mathrm{Li(s)} \leftrightarrows \mathrm{Li^+(el)} + \mathrm{e^-(ed)} + \mathrm{(s)}, \qquad (3)$$

where (s) is a vacant surface site, Li(s) is the lithium on the electrode surface, and V_{Li}(ed) is a lithium vacancy within the graphite electrode lattice structure. In this representation, Reaction 3 is the charge-transfer step, whose rate depends on concentrations and electrostatic-potential differences. Reaction 2 is a heterogeneous surface reaction that depends on species concentrations, but not directly on electrostatic potentials. Even this relatively straightforward two-step process introduces electrochemical complexity that is not typically practiced in Li-ion battery models.

Although Fig. 1 illustrates the electrodes as spherical particles, and typical models represent the electrode particles as spheres, the actual electrodes are far more complex. Figure 2 illustrates a microscale reconstruction of a commercial Li_xCoO_2 cathode. The "particles" have irregular shapes and are overlapping. The carbon-based binder is seen as black in Fig. 2. The open pore space would be filled with a Li-ion-conducting solvent. Models that deal with electrochemistry at the electrode microscale must be concerned with geometrical complexity as well as electrochemical complexity [19].

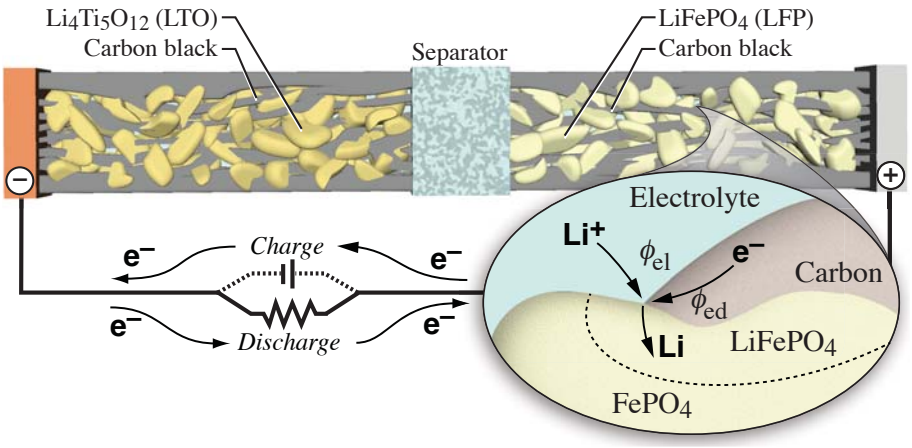

Figure 3: Illustration of processes in a Li-ion battery with a $Li_4Ti_5O_{12}$ anode, liquid electrolyte, and $LiFePO_4$ cathode.

Figure 3 illustrates a Li-ion battery based on a lithium-titanate ($Li_4Ti_5O_{12}$, LTO) anode and a lithium-iron-phosphate ($LiFePO_4$, LFP) cathode. There are several qualitative differences from the system illustrated in Fig. 1. The particles are not spherical and an electronically conducting binder layer is shown. The Li-ion-conducting electrolyte solvent is still assumed to be an organic liquid. However, the charge-transfer chemistry is different.

Figure 3 shows an expanded view at the cathode-particle scale. As illustrated, the charge transfer involves three phases — Li^+ in the electrolyte phase, Li in the LFP phase, and electrons in the carbon-based binder phase. As illustrated in the expanded balloon, the charge-transfer process and the Li reactions within the lithium-iron-phosphate cathode phase involve several chemical and physical processes. Thus, unlike Reaction 1 which involves two phases, the cathode chemistry in Fig. 3 involves three phases. As discussed subsequently, this difference has significant ramifications on representing the net charge-transfer process.

An LFP cathode particle behaves quite differently from a metal-oxide (e.g., Li_xCoO_2) cathode particle. The metal-oxide particles react with Li in a diffusive intercalation process, but the LFP particle is a phase-transformation electrode. As the Li is transported within the particle, a sharp phase-transformation front proceeds within the particle. So, rather than thinking about a spatially distributed Li fraction (e.g., Li_xCoO_2) within the particle, locations within the cathode particles are either $FePO_4$ or $LiFePO_4$. Predicting the phase-front speed depends upon complex thermodynamic, transport, and electrochemical factors, which have substantial impacts on modeling [6, 20–24].

2.2. *Sodium batteries*

Figure 4 illustrates a sodium-based battery, where the anode is molten Na, which also serves as the anode electronic conductor. There is no electrolyte within the anode structure. The separator, which is a solid ceramic Na^+ conductor (e.g., Nasicon), is thus also an electrolyte. The cathode compartment is composed of an electron-conducting carbon foam (electrode phase) and an aqueous iodine solution (electrolyte phase). The aqueous solution contains I_2, I^-, I^{3-}, and Na^+; the presence of multiple charge carriers represent complexity that is not well handled by standard transport models. At the interfaces between the carbon and the aqueous solution, charge transfer proceeds as

$$2I^-(\text{el}) \rightleftharpoons I_2(\text{el}) + 2e^-(\text{ed}). \tag{4}$$

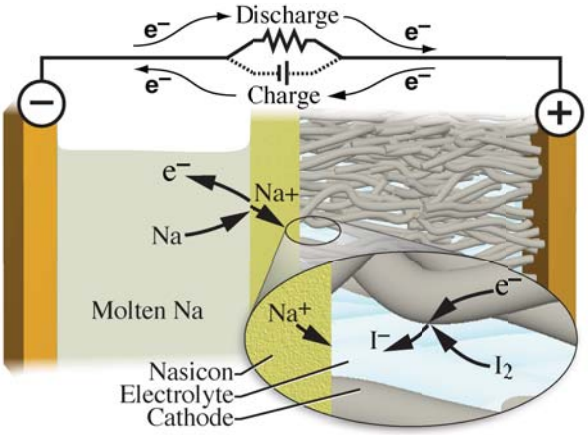

Figure 4: Illustration of a Na-based battery with a sodium-iodine cathode [25].

Within the aqueous solution, homogeneous reaction can proceed as

$$I_2(\text{el}) + I^-(\text{el}) \rightleftharpoons I_3^-(\text{el}). \quad (5)$$

At the interface between the separator and the aqueous solution, Na^+ enters the aqueous solution at a rate that is needed to maintain charge neutrality within the cathode compartment. The Na^+ concentrations must be maintained below a precipitation limit where solid-phase NaI would be formed. Overall, the global chemistry can be represented as

$$2Na^+(\text{el}) + 2I^-(\text{el}) \rightleftharpoons 2Na(\text{ed}) + I_2(\text{el}). \quad (6)$$

However, it is evident that more complex chemistry and transport are present which support the global behavior.

2.3. Lithium-based conversion batteries

So-called "beyond lithium-ion' batteries include lithium-air (Li/O_2) and lithium-sulfur (Li/S) cells, which provide theoretical specific energies an order of magnitude above those of lithium-ion cells [26]. These batteries have conversion chemistries, that is, discharge/charge involves the formation and dissolution of bulk solid phases (lithium oxides, lithium sulfide, metallic lithium). The reaction mechanisms consist of multiple intermediate solid or dissolved species and a combination of parallel and sequential reaction pathways [27]. These multi-phase reactions can result in substantial volume changes within the electrode, which are not captured by standard modeling approaches [28]. Moreover, the associated kinetics are affected by nucleation and growth mechanisms [29], resulting in complex dependencies between reaction rate and morphology including particle size distributions [30].

Unlike most Li-ion battery models, which accommodate binary electrolyte salts such as Li^+ and PF_6^+, Li/S electrolytes have multiple polysulfide anion species whose reactive transport can result in cross-talk between the electrodes (so-called polysulfide shuttle) [31]. Many transport models for electrochemical cells, as discussed in Section 5.2.1, rely on significant simplifications that are only suited for binary soltions [12, 16]. Obviously, the modeling of transport in Li/S electrolytes (or similarly in the Na battery catholyte) requires models with the additional complexity of homogeneous reactions within the liquid catholyte.

The foregoing examples illustrate some differences and commonalities for different battery types. There are numerous other types of batteries, all of which have unique features and associated chemical and electrochemical complexities. The intent here is not to explore all battery architectures and chemistries, but rather to call attention to the need for generalized, flexible, electrochemistry modeling capabilities.

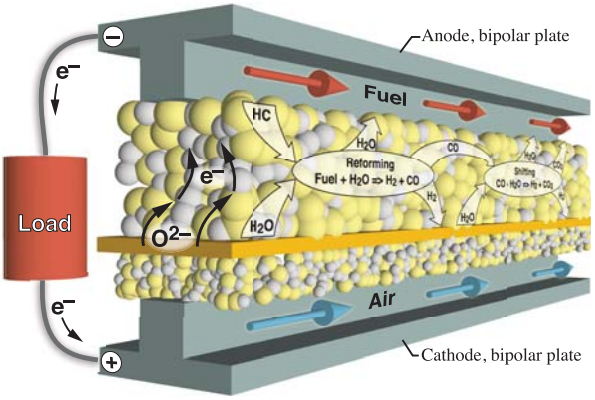

Figure 5: A solid-oxide fuel-cell channel operating on a hydrocarbon fuel, with internal reforming within the anode support structure.

2.4. Solid-oxide fuel cells

As with batteries, there are numerous types of fuel cells. Figure 5 illustrates aspects of a channel layout in a solid-oxide fuel cell. In this case, the dense electrolyte membrane is often a yttrium-doped zirconate (YSZ), which is an oxygen-ion O^{2-} conductor [33]. The anode structure is typically a Ni-YSZ cermet. For hydrocarbon fuel feeds to the anode, catalytic fuel reforming takes place within the porous anode support structure. Near the anode-membrane interface, charge-transfer chemistry produces H_2O as adsorbed H atoms react with the O^{2-} emerging from the dense membrane on three-phase boundaries, delivering electrons to the Ni and ultimately the external circuit. As discussed by Goodwin, et al. [34], the charge-transfer process involves several elementary steps. Gas-phase O_2 is dissociately adsorbed onto the cathode surface and then electrochemically reduced with electrons returning from the external circuit. The resulting O^{2-} ions are incorporated from the cathode surface into the dense membrane.

Along the length of the fuel flow channel in the anode, the fuel is depleted and diluted with product H_2O and CO_2. Along the length of the air flow channel in the cathode, O_2 is removed from the air. The decreases in anode fuel and cathode O_2 concentrations along the length of the channel reduce the chemical potential driving force across the oxide-ion conducting membrane that provides for the cell voltage and facilitates current via electron production at the anode and consumption at the cathode. At operating conditions along the flow path where cell voltages fall below about 0.6 V, Ni in the

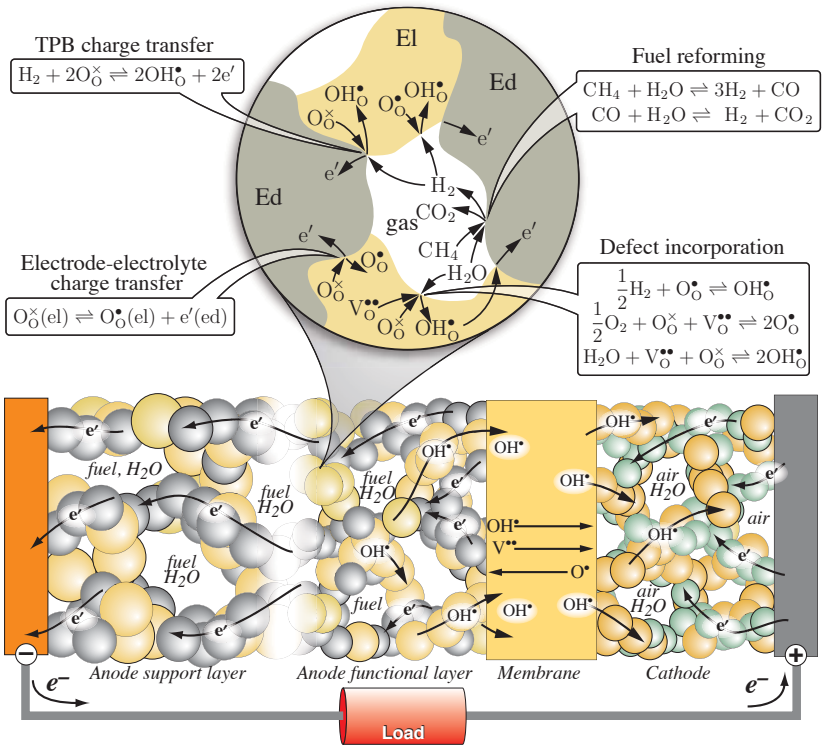

Figure 6: Illustration of the membrane-electrode assembly of a protonic-ceramic fuel cell [32].

porous anode structure begins to oxidize to NiO, leading to irreversible damage. Modeling Ni oxidation represents an example of electrochemical complexity that requires modeling structures that are not readily demonstrated in past modeling studies.

2.5. *Protonic-ceramic fuel cells*

Protonic-ceramic fuel cells (PCFC) [32, 35], as illustrated in the membrane-electrode assembly image in Figure 6, incorporate solid-oxide electrolytes made of materials such as yttrium-doped barium zirconate (e.g., $BaZr_{0.8}Y_{0.2}O_{3-\delta}$, BZY20). The BZY20 predominantly conducts protons, but is a mixed ionic-electronic conductor (MIEC) that has at least three mobile charged defects – protons $OH_O^{\bullet}$, oxide vacancies $V_O^{\bullet\bullet}$, and small polarons $O_O^{\bullet}$ [36]. The composite electrodes are porous structures that usually involve two solid phases and the gas phase flows of reactants. Like SOFCs, the anode is a cermet composite, with Ni being the electrode phase and BZY20 being the electrolyte phase. Such fuel cells are designed to operate on fuel streams of a hydrocarbon (e.g., natural gas) and steam.

In the anode of a hydrocarbon-fueled PCFC, steam reforming produces H_2, which reacts at the anode-electrolyte interface to deliver protons into the dense electrolyte membrane. The protons are transported across the dense membrane, where they react with adsorbed oxygen on the cathode to produce H_2O. PCFCs can convert hydrocarbons to electricity with high efficiency at temperatures in the range of 600 °C. [37–40]

The expanded balloon of Fig. 6 illustrates several heterogeneous and electrochemical reactions in a PCFC anode. Charge transfer can proceed at the three-phase boundaries formed at the intersections of the Ni, BZY20, and gas. Charge transfer can also proceed at the two-phase boundaries between the Ni and BZY20 phases. Charged defects (protons, vacancies, and polarons) from the charge-transfer reactions are incorporated into the MIEC electrolyte at the interfaces. The defect-incorporation reactions are not considered as charge-transfer reactions because they do not transfer charge between two phases at different electrostatic potentials. In other words, all the charged defects are within the electrolyte phase; the gas phase species are charge-neutral. Finally, in addition to acting as the electron-conducting electrode phase, the Ni acts as a steam-reforming catalyst that facilitates the reactions of H_2O and hydrocarbons to form H_2, CO, and CO_2.

This PCFC example highlights a variety of chemical and electrochemical kinetics responsible for fuel-

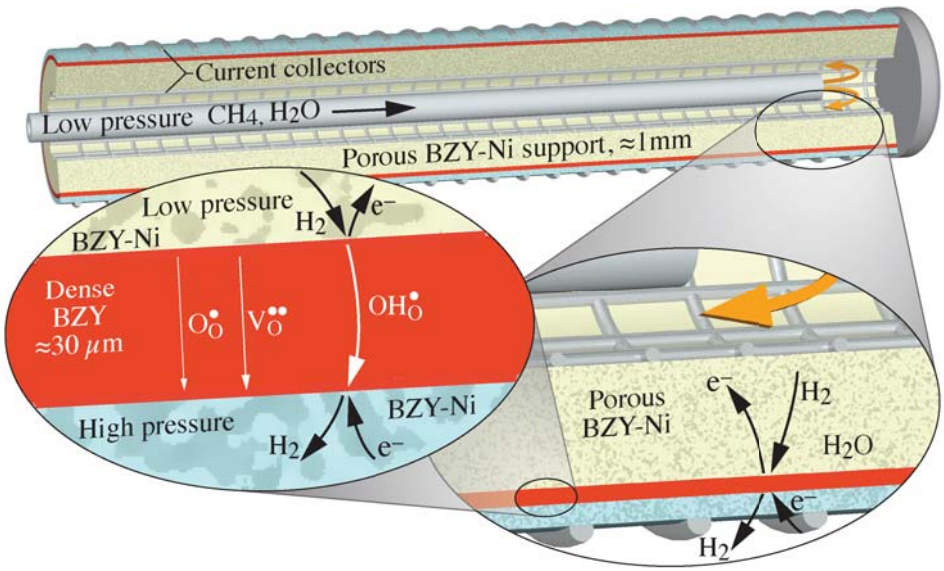

Figure 7: A tubular protonic-ceramic electrochemical cell that integrates steam reforming, hydrogen separation, and hydrogen compression.

cell functionality. The MIEC nature of the electrolyte adds complexity to the chemistry and the transport that encourages the development for more robust modeling frameworks for fuel cell and electrochemical device modeling.

2.6. Electrochemical membrane reactors

Figure 7 illustrates a tubular cell that combines steam reforming, hydrogen separation, and hydrogen electrochemical compression. Such a system has been reported recently by Malerød-Fjeld, et al. [41]. In this system, methane and steam are introduced via a central feed tube, with reactive gases flowing through the annular space between the feed-tube exterior and the inner surface of a porous ceramic-metallic (cermet) tube support structure. The cermet here is similar to the PCFC and composed of Ni and BZY, with the Ni serving as both a reforming catalyst and an electron conductor. In this regard, the porous cermet anode functions similarly to the fuel-cell electrode illustrated in Fig. 6.

The catalytic reforming chemistry involves multiple surface reaction steps. Although much of the fuel-cell literature uses only two global reaction steps (one reforming and one water-gas shift), higher fidelity models use over 20 elementary reactions [33, 42, 43]. Thus, software that handles complex electrochemistry must interact effectively with comparable modeling capabilities that deal with complex homogeneous and heterogeneous thermal chemistry.

A thin (order tens of microns) and dense BZY mixed-conducting membrane is applied to the outside of the relatively thick (order few hundreds of microns) cermet support structure. The BZY membrane predominantly conducts protons ($OH^{\bullet}_{O}$), but other mobile charged defects include oxide vacancies ($V^{\bullet\bullet}_{O}$) and O-site small polarons ($O^{\bullet}_{O}$) [36]. Upon electrical polarization (typically, a few volts), protons are transported through the membrane. If the tube assembly is within a pressure vessel, the H_2 emerging from the membrane can be compressed.

As an ion conductor, the BZY membrane has relatively high resistance and thus ohmic heating must be dissipated within the membrane when operating at high current densities. The conductivity varies exponentially with temperature. With operating temperatures around 600 °C, practical devices must incorporate thermal balances for operational strategies that control membrane temperatures.

Other chemical processes can be accomplished using proton-conducting tubular assemblies such as illustrated in Fig. 7. For example, methane dehydroaromatization (MDA) is a pyrolytic process that synthesizes benzene from methane (i.e., $6CH_4 \rightleftharpoons 9H_2 + C_6H_6$). Because the process is equilibrium limited, removing H_2 through the membrane should, in theory, increase benzene yield [42–45]. Of course, the catalytic MDA chemistry differs from steam-reforming chemistry with a higher propensity for solid carbon deposition and bifunctional catalysts with multiple reactive site types.

Such complex chemistry coupled to the MIEC behavior of the proton-ceramic membrane strains the capabilities of most electrochemical and catalysis modeling frameworks at the reactor scale. As such, this indicates the need for new software tools with generalized data structures to enable modeling of electrochemical membrane reactors for chemical manufacturing.

3. Phases and phase interfaces

All electrochemical cells depend on the reactive transfer of charged species between electronically conductive electrodes and an ion-conducting electrolyte. The anode produces electrons by extracting charge from the conductive ion in the electrolyte, and the cathode consumes electrons to produce ions via a reduction reaction, often either of an oxide material (as in a Li-ion battery discharging) or of oxygen gas flow (as in a fuel cell cathode). Charge-transfer reactions occur at the electrode-electrolyte interface where the two phases are at distinct electrostatic potentials (i.e., voltage Φ_m where m represents the phase). The charge transfer rate is related to the difference in Φ_m across phase interfaces and to the reactive species chemical potentials (μ_k were k represents the species). Since the electrodes and electrolytes are often dense solid or liquid phases, effective electrochemical modeling must incorporate non-ideal thermodynamics to properly account for μ_k. Depending on the direction of charge transfer current across the phase interface and the sign of the voltage change $\Delta\Phi$ across the phase interface, the cell either produces electrical work from the chemical free energy or the cell requires electrical work to increase chemical free energy.

Modeling electrochemical kinetics necessarily involves attention to *phase* thermodynamics, in particular of the mobile and reactive species. Electrochemically neutral reactions may also proceed within each phase or at phase interfaces. However, electrochemical *charge-transfer* reactions must proceed at phase interfaces. The rates of charge-transfer reactions depend on electrostatic potentials of the participating phases at the phase interfaces. Moreover, as is the case for thermal reactions, charge-transfer rates also depend on temperature activation and the activities of participating species.

3.1. Thermodynamics

The relationship between phase voltages Φ_m and species chemical potential μ_k can be derived from equilibrium thermodynamics. Both reaction rates and species transport fluxes depend on differences in species electrochemical potentials $\tilde{\mu}_k$ which combine μ_k and Φ_m as follows:

$$\begin{aligned}\tilde{\mu}_k &= \mu_k + z_k F \Phi_m, \\ &= \mu_k^\circ + RT \ln\left(\gamma_k \frac{[X_k]}{\left[X_k^\circ\right]}\right) + z_k F \Phi_m,\end{aligned} \tag{7}$$

μ_k° are the standard-state chemical potentials at temperature T and a fixed reference species concentrations $\left[X_k^\circ\right]$, and $[X_k]$ are the actual species molar concentrations (mol–m^{-3}). The gas and Faraday constants are represented as R and F, respectively. The activity coefficients for the k species are represented as γ_k, which capture any non-ideal species interactions associated with a given phase, relative to the fixed reference state. The species charges and the phase electrostatic potentials for each of the k species are represented as $\Phi_{m,k}$ and z_k, respectively.

For a given electrode-electrolyte interface reaction i under equilibrium conditions with no net current across the interface, the equilibrium condition relates the $\tilde{\mu}_k$.

$$\sum_{k=1}^{K} \nu'_{ki} \tilde{\mu}_k = \sum_{k=1}^{K} \nu''_{ki} \tilde{\mu}_k, \tag{8}$$

where ν'_{ki} and ν''_{ki} are the forward and backward stoichiometric coefficients for reaction i, respectively. Substituting Eq. 7 into Eq. 8 provides a basis for relating equilibrium chemical potential and voltage differences across electrode-electrolyte interfaces. The equilibrium $\Delta\Phi_m$ across the phase interface is relatively straightforward when there is only one charge transfer reaction across the phase interface. When multiple charge transfer reactions are active at an interface, multiple equilbria equations must be solved simultaneously to find the equilibrium $\Delta\Phi_m$. The equilibrium $\Delta\Phi_m$ is referred to as the reversible half-cell voltage. The sum of the anode-electrolyte and cathode-electrolyte reversible half cell voltages gives the so-called reversible cell voltage.

The relationship between thermodynamics and cell voltages, chemical reaction rates, and transport-driving gradients requires software with effective models and data structures for calculating both μ_k° and γ_k for the various materials/phases in relevant electrochemical cells. The chemical potentials can be evaluated in terms of the local phase Gibbs free energy G as

$$\mu_k = \left.\frac{\partial G}{\partial n_k}\right|_{T,p,n_{j\neq k}}, \tag{9}$$

where n_k are the number of moles of species k used in evaluating G. The activity coefficients γ_k in solid, liq-

uid, or high-pressure gas mixtures are related to the molar excess Gibbs free energy g^{E} due to species interactions or mixing as

$$n_{\mathrm{T}} g^{\mathrm{E}} = G - \sum_{k=1}^{K} n_k \left[\mu_k^{\circ} + RT \ln \left(\frac{[X_k]}{[X_k^{\circ}]} \right) \right], \tag{10}$$

where $n_{\mathrm{T}} = \sum_k n_k$. The excess free energy g^{E} can be simplified to reveal the relationship to the activity coefficients γ_k as

$$n_{\mathrm{T}} g^{\mathrm{E}} = \sum_{k=1}^{K} n_k RT \ln(\gamma_k). \tag{11}$$

Individual species γ_k are now defined as the partial derivative of the excess free energy with respect to n_k.

$$RT \ln(\gamma_k) = \left. \frac{\partial (n_{\mathrm{T}} g^{\mathrm{E}})}{\partial n_k} \right|_{T,p,n_{j \neq k}}, \tag{12}$$

For ideal gas conditions as expected in SOFC gas flows, $g^{\mathrm{E}} = 0$ and thus all $\gamma_k = 1$. However, for high-pressure flows and condensed phases (such as solid ceramic or organic liquid electrolytes), accurate theoretical models for γ_k are required.

A challenge for generalized electrochemical device modeling tools is the development of robust, quantitative models for g^{E} and self-consistent databases for different solid materials and fluids derived from experiments and/or molecular material models. Many thermodynamic models (some empirical [9, 46] and some more fundamental) have been developed for liquid electrolytes, solid oxides, and other non-ideal materials. Some of these models are referenced in subsequent sections, but the importance of developing user-accessible material and fluid models for calculating γ_k remains a critical step for developing next-generation electrochemistry modeling tools.

Further complications to thermodynamics related to mechanical stress can impact electrochemical reaction rates and driving forces for species transport. Stress induced diffusion can become significant for a variety of solid materials [47] (e.g. phase transformation battery electrodes [48], intercalation electrodes [49], and MIEC ceramic membranes [50–52]). Stress induced diffusion is accommodated by adding a term to the species electrochemical potential [48–50]. For example, the species electrochemical potential for a solid accounting for hydrostatic stresses can be expressed as [50, 53]

$$\tilde{\mu}_k = \mu_k + z_k F \Phi_m - \Omega_k \sigma_{\mathrm{h}}, \tag{13}$$

where Ω_k is the species partial molar volume, and σ_{h} is the hydrostatic stress. To resolve the hydrostatic stress, an additional stress-strain relation can be imposed on the system. One constraint proposed by Yang for systems in quasi-static equilibrium with no external force can be expressed as

$$\nabla^2 \left(\sigma_{\mathrm{h}} + \frac{2E\Omega_k}{9(1-\nu)} [X_k] \right) = 0 \tag{14}$$

where E and ν are the Young's modulus and Poisson's ratio of the material, respectively [54]. The importance of mechanical stress on electrochemical cell voltages and reaction rates depends strongly on the volumetric expansion or contraction of phases with changes in point defect concentrations $[X_k]$. This issue has received increased attention, recently, for both batteries and high-temperature fuel cells.

3.2. Interfaces

Electrochemical processes may proceed at the so-called three-phase boundaries (intersections between electrode, electrolyte, and an electrically neutral phase), or at two-phase interfaces (intersections between electrode and electrolyte phases). For example, consider a composite solid-oxide fuel cell anode, typically a porous structure comprised of the electrode phase (e.g., Ni), electrolyte phase (e.g., YSZ), and the gas phase containing the fuel. Li-ion battery electrodes can also have three-phase boundaries. Such a cathode may be structured with the electrode phase (e.g., $LiFePO_4$), an electrolyte phase (e.g., liquid ethylene carbonate), and an electrically conductive carbon additive phase (see, e.g. the expanded view in Fig. 3. In such cases, species (possibly including electrons) in three phases, as well as the electrostatic potentials of the phases, contribute to the charge-transfer chemistry. Li-ion batteries are also modeled assuming charge transfer reactions proceed at two-phase interfaces. In such cases, the charge-transfer reactions proceed on the surface between the electrode and electrolyte. The rates depend on the species concentrations and phase electrostatic potentials. Addtionally, rates depend on the local electrode microstructure (i.e., total amount of surface area or three-phase boundary available). As noted in section 2.1, accurate microstructural representations are required to accurately predict charge transfer rates.

3.3. Phase management

Electrochemical cells are often modeled using a homogenization approach, where the components of an individual subdomain (e.g., a composite electrode) are not spatially resolved on a microscopic scale, but are described as continuum of superimposed phases. In such

a setting, the phases are characterized by their respective volume fractions ε_m and densities ρ_m. In standard modeling approaches these properties are often (implicitly) assumed constant. However, this is generally not the case. In lithium-ion batteries, active materials change their density and expand during intercalation. During aging, secondary phases (e.g., SEI) grow, gases are formed, and liquid electrolyte is consumed. In Li/O_2 batteries, different lithium oxides are formed, taking up the space of gas-filled or liquid-electrolyte-filled pore space, eventually causing pore clogging. In PEM fuel cells, liquid water saturation dynamically depends on operating conditions. A generalized continuity equation for all phases in a continuum setting may be formulated as [28]

$$\frac{\partial(\rho_m \varepsilon_m)}{\partial t} = \sum_{k,m} \dot{\omega}_k W_k, \tag{15}$$

where the sum runs over all species of phase m and $\dot{\omega}_k$ is the rate of formation (mol m^{-3} s^{-1}).

The density ρ_m generally depends on phase composition, and constitutive equations including their parameters are required to describe this dependency. For gas phases, the assumption of the ideal gas law,

$$\rho_m = \frac{p}{RT} \frac{1}{\sum_k Y_k / W_k}, \tag{16}$$

is valid for the conditions met in most electrochemical cells. Here ρ_m is the phase density, p the pressure, and Y_k and W_k the mass fraction and molecular weight of species k. The density of an ideal condensed phase can be calculated from species partial molar volumes Ω_k,

$$\rho_m = \frac{\sum_k X_k W_k}{\sum_k X_k \Omega_k}, \tag{17}$$

where X_k is the species mole fraction. Using these constitutive relationships, Eq. 15 predicts the dynamic change of volume fractions of each phase. The necessary condition $\sum_m \varepsilon_m = 1$ requires the assumption of either the presence of a compressible gas phase that accommodates volume changes, or the presence of a mobile fluid phase that is displaced upon volume changes, or the coupling to a mechanical model describing overall volume expansion.

4. Charge-transfer chemistry

Electrochemical reactions occur at phase interfaces where charge is transferred between phases that are at different electrostatic potentials Φ. In Li-ion batteries, where the chemical potential driving forces across the cell are derived from Li stored in the electrode structures, these phase interfaces typically involve two phases (e.g., the electrolyte and the electrode as in Reaction 1 for a battery anode). In fuel cells, where the chemical-potential driving forces are derived from gas or liquid flows in contact with electrode/electrolyte interfaces, the charge-transfer occurs at so-called three-phase boundaries at the intersection between the electrode, electrolyte, and the electrically neutral fluid phase (cf., Fig. 6). For example, consider a solid-oxide fuel cell composite anode that is a porous structure comprised of the electron conducting electrode (e.g., Ni), the ion-conducting electrolyte (e.g., YSZ), and the gas phase containing the fuel (e.g., H_2). The charge-transfer chemistry may be represented as

$$H_2(g) + O^{2-}(el) \rightleftharpoons H_2O(g) + V_O(el) + 2e^-(an), \tag{18}$$

where $H_2(g)$ is in the gas phase, $O^{2-}(el)$ is in the electrolyte phase, $H_2O(g)$ is in the gas phase, $V_O(el)$ is an oxide ion vacancy in the electrolyte phase, and $e^-(an)$ is in the Ni phase. Reaction 18 is a global reaction that simplifies a more complex multi-step reaction sequence [33, 55].

The rate of a charge-transfer reaction at a multi-phase interface, according to fundamental Marcus theory, depends on the electrostatic potential difference between the participating electrode and electrolyte phases. However, nearly all battery and fuel-cell models are based on the Butler–Volmer formulation using "overpotentials."

4.1. Marcus theory

The fundamental Marcus formulation is general and does not depend on simplifying assumptions. Consider a general reaction, including a charge-transfer reaction, as represented by

$$\sum_{k=1}^{K} \nu'_{ki} \chi_k^{z_k} \rightleftharpoons \sum_{k=1}^{K} \nu''_{ki} \chi_k^{z_k}, \tag{19}$$

where $\chi_k^{z_k}$ is the chemical symbol for the kth species with charge z_k. The subindices represent species k and reaction i, with K being the number of species involved in the reaction. One of the "species" may be an electron. Reactions may involve species that reside in different phases, such as electrode and electrolyte phases. Electrochemical charge-transfer reactions transfer charge between phases. By convention, reversible charge-transfer reactions are usually written such that the forward direction is the *anodic direction* (i.e., producing electrons). The backward direction is called *cathodic*, meaning that electrons are consumed.

A reaction's rate of progress can be written in terms of the difference between forward (anodic) and backward (cathodic) rates of progress q_i as

$$q_i = q_{\mathrm{f}i} - q_{\mathrm{b}i} = k_{\mathrm{f}i} \prod_{k=1}^{K} a_k^{\nu'_{ki}} - k_{\mathrm{b}i} \prod_{k=1}^{K} a_k^{\nu''_{ki}}, \tag{20}$$

where a_k are the activity concentrations of the participating species. Generally speaking, the activity concentration for the species in a particular phase are equal to the activity coefficient γ_k multiplied by a dimensionally-appropriate concentration. For bulk, three-dimensional fluids (gases, liquids, solids), the concentration is simply the molar concentration $[X_k]$. For surface-adsorbed species, the concentration equals the surface coverage $\Gamma_m \theta_{k,m}$, where $\theta_{k,m}$ is the site fraction for species k on the surface of phase m and Γ_m is the total available surface site density on phase m.

For a given rate of progress q_i for a charge transfer reaction i, the associated charge transfer current $i_{e,i}$ is:

$$i_{e,i} = q_i n_{e,i} F, \tag{21}$$

where $n_{e,i}$ is the number of electrons (or equivalent charge) transferred to the electrode by the reaction:

$$n_{e,i} = -\sum_{k=1}^{K_{\mathrm{ed}}} \nu_k z_k, \tag{22}$$

summed over all the K_{ed} species in the electrode phase, including electrons. The sign convention is such that an anodic rate of progress (produces negative charge in the electrode) is considered a positive current, while a cathodic rate of progress results in $i_{i,e} < 0$.

The forward and backward rate expressions for each reaction i are written as

$$k_{\mathrm{f}i} = k_{\mathrm{f}i}^{\mathrm{t}} \exp\left[-\beta_{\mathrm{a}i} \sum_{k=1}^{K} \frac{\nu_{ki} z_k F \Phi_k}{RT}\right], \tag{23}$$

$$k_{\mathrm{b}i} = k_{\mathrm{b}i}^{\mathrm{t}} \exp\left[+\beta_{\mathrm{c}i} \sum_{k=1}^{K} \frac{\nu_{ki} z_k F \Phi_k}{RT}\right], \tag{24}$$

where $\nu_{ki} = \nu''_{ki} - \nu'_{ki}$, and the forward and backward thermal rate coefficients (i.e., at zero electric-potential difference) are represented as $k_{\mathrm{f}i}^{\mathrm{t}}$ and $k_{\mathrm{b}i}^{\mathrm{t}}$. Assuming an elementary single-electron-transfer reaction, the anodic and cathodic symmetry factors $\beta_{\mathrm{a}i}$ and $\beta_{\mathrm{c}i}$ respectively are related by $\beta_{\mathrm{a}i} + \beta_{\mathrm{c}i} = 1$. There can be confusion between the anodic and cathodic symmetry factors (β_{a} and β_{c}) and the anodic and cathodic transfer coefficients (α_{a} and α_{c}). These coefficients are interchangeable for elementary, single-charge transfer reactions. For global reactions and reactions involving the transfer of more than one charge, the anodic and cathodic transfer coefficients (α_{a} and α_{c}) should be used because they are not constrained to sum to unity [33, 56, 57].

When there is a transfer of charge between phases at different electric potential, the term $\sum_{k=1}^{K} \nu_{ki} z_k F \Phi_k$ is non-zero, and the charge transfer rate is modified by the electric-potential difference between the two participating phases. Each phase m is assumed to be at an electric potential Φ_m. However, as a matter of convenience in computational implementation, each species can be assigned the charge associated with its phase. In other words, in writing Eqs. 23 and 24, each species k is assigned with an electric potential Φ_k, not the phase Φ_m directly. Generally, each species assumes the electric potential of phase in which it exists. The gas-phase is usually considered to be electrically neutral ($\Phi_m = 0$).

If all the charged species in an electrochemical reaction are in the same phase, the electric potentials do not affect the reaction rate (i.e., the exponential factors in Eqs. 23 and 24 are exactly unity). For example, consider the reaction $\mathrm{I_2} + \mathrm{I^-} \leftrightharpoons \mathrm{I_3^-}$, which may occur within a sodium battery aqueous electrolyte (i.e., a single phase). In this case, assuming Φ_k is the same for all species, $\sum_k \nu_k z_k = \nu_{\mathrm{I_2}} z_{\mathrm{I_2}} + \nu_{\mathrm{I^-}} z_{\mathrm{I^-}} + \nu_{\mathrm{I_3^-}} z_{\mathrm{I_3^-}} = (-1) \times (0) + (-1) \times (-1) + (+1) \times (-1) = 0$.

The thermal reaction rate expressions k_i^{t} ($k_{\mathrm{f}i}^{\mathrm{t}}$ or $k_{\mathrm{b}i}^{\mathrm{t}}$) are usually represented using the modified Arrhenius expression as

$$k_i^{\mathrm{t}} = A_i T^{n_i} \exp\left(-\frac{E_i}{RT}\right), \tag{25}$$

where E_i represents the activation energy, A_i the pre-exponential factor, and n_i the temperature exponent. To satisfy microscopic reversibility and maintain thermodynamic consistency, the thermal component of the backward (cathodic) rate $k_{\mathrm{b}i}^{\mathrm{t}}$ is related to the forward (anodic) rate $k_{\mathrm{f}i}^{\mathrm{t}}$ via the reaction equilibrium constant K_i as

$$K_i = \frac{k_{\mathrm{f}i}^{\mathrm{t}}}{k_{\mathrm{b}i}^{\mathrm{t}}} = \exp\left(-\frac{\Delta G_i^\circ}{RT}\right) \prod_{k=1}^{K} \left[X_k^\circ\right]^{-\nu_{ki}}, \tag{26}$$

where ΔG_i° is the change of the standard-state Gibbs free energy for the reaction. Evaluating ΔG_i°, and hence the equilibrium constant, requires quantitative thermochemical properties for all species in the reaction.

4.2. Charge-transfer transition state

For the sake of illustration, consider a charge-transfer reaction representing behavior at a solid-oxide fuel cell

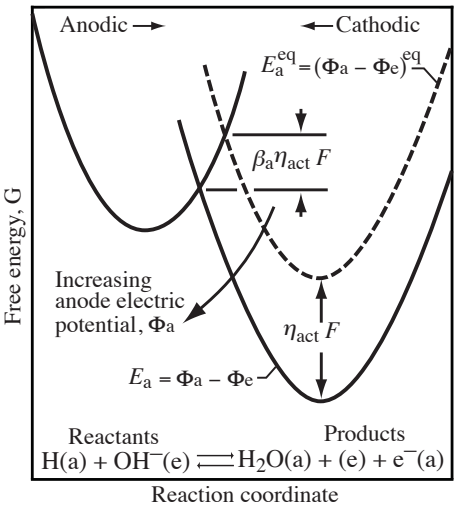

Figure 8: Potential energy surfaces to assist visualizing the effect of electric-potential difference on charge-transfer reaction rates.

anode,

$$H(a) + OH^-(e) \rightleftharpoons H_2O(a) + (e) + e^-(a). \qquad (27)$$

The nomenclature (a) and (e) indicate the anode and electrolyte phases, respectively. In this reaction, H(a) and $H_2O(a)$ are assumed to be adsorbates on the anode (e.g., Ni) surface. A hydroxyl ion on the electrolyte surface (e.g., YSZ) is represented as $OH^-(e)$ and (e) represents an open site on the electrolyte surface. The electron $e^-(a)$ is in the electrode phase. Figure 8 illustrates potential-energy surfaces that assist understanding the influence of electric potentials on charge-transfer rates. The potential-energy surface on the left represents the reactants and the one on the right represents the products. The electric-potential difference at the interface between anode (a) and electrolyte (e) is written as $E_a = \Phi_a - \Phi_e$ (where the subscript 'a' represents that it is for the anode half-cell). The equilibrium electric-potential difference E_a^{eq} is the electric-potential difference at which the reaction proceeds at equal and opposite rates in the anodic (forward) and cathodic (backward) directions (illustrated as the dashed line). There is a potential-energy barrier between the reactant and product states, which tends to be cusp-like for charge-transfer reactions. When proceeding in the anodic direction, the charge-transfer reaction illustrated in Fig. 8 delivers electrons into the anode, which is at a lower electric potential than is the electrode. The electric potential difference therefore opposes the charge transfer reaction, in this state: the negatively charged electron is naturally repelled from the negative electrode. Some of the chemical energy stored as chemical bonds in the reactants is converted to electric current as electrons are delivered into the anode (i.e., the conduction band of the Ni). When the electrostatic potential of the anode increases relative to the electrolyte (i.e., E_a increases), the electrostatic barrier to charge transfer decreases.

The symmetry factors β are related to the magnitudes of the slopes of the potential-energy surfaces at their crossing point. Because the slopes are typically similar, the symmetry factors for elementary reactions are usually near $\beta = 1/2$. When the anode electric potential Φ_a is increased relative to the adjoining electrolyte electric potential Φ_e, the *activation overpotential* η_{act} is increased by the same amount. As illustrated in Fig. 8, the product-side potential energy surface is lowered by $\eta_{act}F$ and the anodic energy barrier is lowered by $\beta_a \eta_{act} F$. The energy barrier in the cathodic direction, meanwhile, increases by $(1 - \beta_a)\,\eta_{act} F$.

4.3. Butler–Volmer kinetics

The Bulter–Volmer formulation is an alternative approach to modeling charge transfer reactions. Fundamentally, the Bulter–Volmer expression can be derived from elementary Marcus Theory if there is a single rate-limiting charge-transfer reaction [8, 57]. Although the Butler–Volmer formulation is very widely used and has some significant advantages, it also has some very significant limitations. If the chemistry is represented simply as a single reaction (e.g., Reaction 1), then the limitations are not demanding. In fact, for such simple chemistry, there is relatively little need for generalized software. However, if the charge-transfer chemistry is a multi-step process, then the Butler–Volmer formulation is much less appropriate and more difficult to implement.

In the Butler–Volmer form, the charge-transfer current can be expressed as

$$\begin{aligned} i_e &= q_i n_{e,i} F \\ &= i_0 \left[\exp\left(\frac{\alpha_a F \eta_{act}}{RT} \right) - \exp\left(\frac{-\alpha_c F \eta_{act}}{RT} \right) \right], \end{aligned} \qquad (28)$$

where i_e is the current at the electrode/electrolyte interface, $n_{e,i}$ is as in Eq. 22, i_0 is the *exchange current density*, η_{act} is the *activation overpotential*, and α_a and α_c are the anodic and cathodic transfer coefficients, respectively. The activation overpotential η_{act} is defined as

$$\eta_{act} = (\Phi_{ed} - \Phi_{el}) - (\Phi_{ed} - \Phi_{el})^{eq}, \qquad (29)$$

where Φ_{ed} and Φ_{el} are the electrostatic potentials in the electrode and electrolyte phases, respectively. The term $E^{eq} = (\Phi_{ed} - \Phi_{el})^{eq}$ represents the equilibrium electrostatic-potential difference (cf., Section 4.4).

As expressed by Eq. 29, the Butler–Volmer formulation references electrostatic potentials to equilibrium reversible potential differences. At first, this may seem indirect and cumbersome. However, the advantage is that open-circuit voltages can be easily measured and, in many cases, evaluated theoretically. However, as discussed subsequently, as the chemistry becomes more complex, there are limitations and complications associated with the Butler–Volmer formulation.

The Butler–Volmer exchange current density i_0 includes the activities of the participating species. For example, the exchange current density for Reaction 1 at a graphite anode is typically represented as

$$i_0 = k_{ct} F [\mathrm{Li^+(el)}]^{\alpha_a} [\mathrm{Li_{max}(ed) - Li(ed)}]^{\alpha_a} [\mathrm{Li(ed)}]^{\alpha_c}. \tag{30}$$

Several important points can be noted about Eq. 30. The factor k_{ct} represents an Arrhenius rate expression that captures the temperature dependence and F is the Faraday constant. The exponents α_a and α_c are the anodic and cathodic symmetry factors, multiplied by the stoichiometric coefficients for the reaction, which are equal to unity in Reaction 1. The concentration $[\mathrm{Li_{max}(ed)}]$ represents the maximum concentration of intercalated Li within the graphite anode, which must be independently specified.

Strictly speaking, as stated, Reaction 1 means that the anode is Li metal. Although a bulk Li metal anode is a viable possibility, the common intent is that the anode is graphite with intercalated Li. So, the more correct way to state the reaction is

$$\mathrm{LiC_6(ed) \rightleftharpoons Li^+(el) + e^-(ed) + C_6(ed)}, \tag{31}$$

which explicitly identifies the fact that the anode is graphite and that the Li is intercalated. In this case, the exchange current density can be evaluated directly in the dilute solution approximation as

$$i_0 = k_{ct} F [\mathrm{Li^+(el)}]^{\alpha_a} [\mathrm{C_6(ed)}]^{\alpha_a} [\mathrm{LiC_6(ed)}]^{\alpha_c}. \tag{32}$$

The concentration $[\mathrm{LiC_6(ed)}]$ represents the intercalated Li and $[\mathrm{C_6(ed)}]$ represents the unoccupied graphite that is available to accept Li. The activity coefficients here are assumed to be unity and do not include complex concentrated solution effects [9, 58] . In this form, the identity of the electrode phase and its associated thermodynamics are clear in the reaction statement itself. Thus, these parameters do not need to be specified separately and software can be written to establish the form of the exchange current density.

4.3.1. Multi-step charge-transfer processes

Assuming there is a single rate-limiting reaction step, a multi-step charge transfer reactions can be written in Butler–Volmer form. However, the resulting exchange current density can become complex and not easily derived from the original reaction mechanism. Consider, for example, oxygen reduction at a protonic-ceramic fuel cell (PCFC) cathode. With very few exceptions in the literature, fuel cell models represent the cathode chemistry globally (in Kröger–Vink notation) as

$$\frac{1}{2}\mathrm{O_2(g)} + 2\mathrm{OH_O^\bullet(el)} + 2\mathrm{e'(ed)} \rightleftharpoons 2\mathrm{O_O^\times(el)} + \mathrm{H_2O(g)}, \tag{33}$$

where $\mathrm{OH_O^\bullet}$ is a proton in the electrolyte phase, $\mathrm{O_O^\times(el)}$ is an oxide ion in the electrolyte phase, $\mathrm{e'}$ is an electron in the electrode (cathode) phase, and $\mathrm{O_2(g)}$ and $\mathrm{H_2O(g)}$ are gas-phase oxygen and steam respectively. Such a reaction is easily represented in Butler–Volmer form. However, assume that the actual chemistry is a multi-step process as

$$\mathrm{O_2(g) + 2(s) \rightleftharpoons 2O(s)}, \tag{34}$$

$$\mathrm{OH_O^\bullet(el) + O(s) + e'(ed) = OH(s) + O_O^\times(el)}, \tag{35}$$

$$\mathrm{OH_O^\bullet(el) + OH(s) + e'(ed) = H_2O(s) + O_O^\times(el)}, \tag{36}$$

$$\mathrm{H_2O(s) = H_2O(g) + (s)}, \tag{37}$$

where O(s), OH(s), and $\mathrm{H_2O(s)}$ are the adsorbed oxygen, hydroxyl and water molecules on the cathode surface, respectively, and (s) is an empty cathode surface site. The rates of such elementary reactions can be represented according to Eqs. 20–24. Assuming a single rate-limiting charge-transfer step (e.g., Reaction 35), and with a great deal of tedious algebra, a Butler–Volmer rate expression can be derived [32]. Without going into details here, the resulting expression for the net current density is

$$i_e = i_0 \left[\exp\left(\frac{(1+\beta_{a,(35)}) F \eta_{act}}{RT} \right) - \exp\left(-\frac{\beta_{c,(35)} F \eta_{act}}{RT} \right) \right], \tag{38}$$

where the exchange current density may be expressed as

$$i_0 = i_0^* \frac{(p_{O_2}/p_{O_2}^*)^{(1/2-\beta_{c,(35)}/4)} (p_{H_2O}/p_{H_2O}^*)^{\beta_{c,(35)}/2}}{1 + (p_{O_2}/p_{O_2}^*)^{1/2} + (p_{H_2O}/p_{H_2O}^*)} \times [\mathrm{O_O^\times}]^{\beta_{c,(35)}} [\mathrm{OH_O^\bullet}]^{\beta_{a,(35)}}. \tag{39}$$

In these expressions, $\beta_{a,(35)}$ and $\beta_{c,(35)}$ are the symmetry factors associated with the rate-limiting charge-transfer reaction (Reaction 35). Note that the effective Butler–Volmer symmetry factors $(1+\beta_a)+\beta_c \neq 1$. The variables $p_{O_2}^* = 1/K_{34}$, $p_{H_2O}^* = 1/K_{37}$, are derived from equilibrium constants of the adsorption reactions. The variable

Table 1: Proposed SEI film growth reaction mechanism for a graphite particle [11].

Graphite-SEI interface
$C_3H_4O_3(E) + (S^s) \rightleftharpoons C_3H_4O_3(S^s)$
$Li^+(S^s) \rightleftharpoons Li^+(E) + (S^s)$
$C_2H_4(E) + 2(S^s) \rightleftharpoons C_2H_4(S^s)$
$C_3H_4O_3^-(S^s) \rightleftharpoons C_3H_4O_3(S^s) + e^-_{S^b}$
$C_2H_4(S^s) + CO_3^{2-}(S^s) \rightleftharpoons C_3H_4O_3^-(S^s) + e^-_{S^b} + 2(S^s)$
$CO_3^{2-}(S^s) + 2Li^+(S^s) + (S^b) \rightleftharpoons Li_2CO_3(S^b) + 3(S^s)$
$Li(S^b) + (S^s) \rightleftharpoons V^-(S^b) + Li^+(S^s)$
$Li^+_{S^b} + (S^s) \rightleftharpoons Li^+(S^s)$
SEI-electrolyte interface
$e^-_{S^b} \rightleftharpoons e^-_{C_6}$
$Li(C_6) + V^-(S^b) \rightleftharpoons Li(S^b) + e^-_{C_6} + (C_6)$
$Li(C_6) \rightleftharpoons Li^+_{S^b} + e^-_{C_6} + (C_6)$

i_0^* is an empirical parameter, which contains information such as three-phase-boundary length. Clearly, the resulting Butler–Volmer expressions are not easily derived from the reactions themselves.

To pick another example, consider the important problem solid-electrolye-interface (SEI) chemistry in Li-ion battery anodes. Although details of this chemistry are not well understood, there is certainly active numerical and experimental research to understand it [3, 11, 59–61]. Again, without developing the details here, Table 1 shows the reactions mechanisms analyzed by Colclasure, et al. [3, 11]. The only purpose for showing such reactions here is to emphasize the need for handling increasingly complex electrochemical kinetics.

4.4. Reversible potentials

The *reversible potential* for any charge-transfer reaction i can be thermodynamically evaluated as [57]

$$E_i^{\mathrm{REV}} = \frac{1}{n_{\mathrm{e},i}F}\sum_{k=1}^{K}\nu_{ki}\,\mu_k, \tag{40}$$

where the signs of stoichiometric coefficients ν_{ki} assume that the reaction is written in the anodic direction (i.e., producing electrons). Assuming that there is a single global charge-transfer reaction for both the anode and cathode half-cells, the full-cell reversible potential can be expressed as

$$E^{\mathrm{REV}} = E_{\mathrm{c}}^{\mathrm{REV}} - E_{\mathrm{a}}^{\mathrm{REV}}. \tag{41}$$

To be more explicit, for example, consider a polymer-electrolyte-membrane (PEM) fuel cell. Assuming that the cathode half-cell reaction (written in the anodic direction) can be expressed as

$$H_2O(\mathrm{g,c}) \rightleftharpoons 2H^+(\mathrm{el}) + \frac{1}{2}O_2(\mathrm{g,c}) + 2e^-(\mathrm{c}), \tag{42}$$

the reversible potential can be expressed as

$$E_{\mathrm{c}}^{\mathrm{REV}} = \frac{1}{2F}\left(-\mu_{H_2O(\mathrm{g,c})} + 2\mu_{H^+(\mathrm{el})} + \frac{1}{2}\mu_{O_2(\mathrm{g,c})}\right). \tag{43}$$

Similarly, assuming the anode half-cell reaction is expressed as

$$H_2(\mathrm{g,a}) \rightleftharpoons 2H^+(\mathrm{el}) + 2e^-(\mathrm{a}), \tag{44}$$

the reversible potential on the anode can be expressed as

$$E_{\mathrm{a}}^{\mathrm{REV}} = \frac{1}{2F}\left(-\mu_{H_2(\mathrm{g,a})} + 2\mu_{H^+(\mathrm{el})}\right). \tag{45}$$

The full-cell reversible potential is then

$$\begin{aligned} E^{\mathrm{REV}} &= \frac{1}{2F}\left(-\mu_{H_2O(\mathrm{g,c})} + \frac{1}{2}\mu_{O_2(\mathrm{g,c})} + \mu_{H_2(\mathrm{g,a})}\right), \\ &= -\frac{\Delta G^\circ}{2F} + \frac{RT}{2F}\ln\frac{p_{H_2(\mathrm{g,a})}p_{O_2,\mathrm{c}}^{1/2}}{p_{H_2O(\mathrm{g,c})}}, \end{aligned} \tag{46}$$

with

$$\Delta G^\circ = \mu^\circ_{H_2O} - \frac{1}{2}\mu^\circ_{O_2} - \mu^\circ_{H_2}. \tag{47}$$

In these expressions, $p_{k,\mathrm{c}}$ and $p_{k,\mathrm{a}}$ are the partial pressure of species k on the cathode and anode side, respectively. The change in standard-state Gibbs free energy ΔG° can be expressed using the equilibrium cathode-side gas composition,

$$\exp\left(\frac{-\Delta G^\circ}{RT}\right) = \frac{p_{H_2O(\mathrm{g,c})}}{p_{H_2(\mathrm{g,c})}p_{O_2(\mathrm{g,c})}^{1/2}}. \tag{48}$$

Keep in mind that the reversible potential is an inherently equilibrium concept. That is, the gas on both sides of the membrane is assumed to be in thermodynamic equilibrium.

Substituting Eq. 48 into Eq. 46 results in a simplified expression for the full-cell reversible potential as

$$E^{\mathrm{REV}} = \frac{RT}{2F}\ln\frac{p_{H_2(\mathrm{g,a})}}{p_{H_2(\mathrm{g,c})}}. \tag{49}$$

Importantly, for PEM fuel cells, assuming the membraned is a pure proton conductor, the reversible potential may be evaluated using only the gas-phase compositions in the anode and cathode chambers.

If the electrolyte membrane is single-ion-conducting, the reversible potential is equivalent to the open-circuit potential

$$E^{\mathrm{REV}} = E^{\mathrm{OCV}}. \tag{50}$$

The open-circuit potential is the (measurable) electric potential difference between the cathode and anode current collectors,

$$E^{\mathrm{OCV}} = \Phi_{\mathrm{c}} - \Phi_{\mathrm{a}}, \tag{51}$$

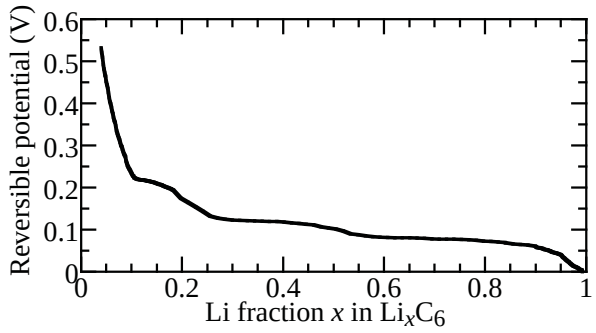

Figure 9: Reversible half-cell potential between an intercalating graphite electrode and Li metal.

when no external circuit is attached (i.e., an infinite-resistance external circuit). In most, if not all, electrochemical devices (e.g., batteries, fuel cells, membrane reactors,...) the *open-circuit voltage* is easily measured. Thus, any practical model must be capable of accurately predicting open-circuit voltage.

If the reversible potential is known, either by theory or measurement, then including that information in modeling is valuable. In cases such as the PEM fuel cell, evaluating E^{REV} is straightforward in terms of gas-phase composition (cf. Eq. 49). However, in Li-ion batteries, for example, the situation is somewhat more complex. Figure 9 shows measurable half-cell reversible potential between a graphite anode and Li metal. The reversible potential varies as a function of intercalated Li (i.e., x in Li_xC_6). Such information can be communicated to a model via data fitting or theoretical expressions [9].

Expressing charge-transfer kinetics in the Butler–Volmer setting takes advantage of *a priori* knowledge of reversible potentials. Although the Butler–Volmer formulation depends on some very significant limiting assumptions, it does benefit from the inherent ability to predict measurable open-circuit voltages.

For membranes that are mixed ionic-electronic conductors (MIEC), in general $E^{\mathrm{OCV}} \neq E^{\mathrm{REV}}$. This is the case, for example, for protonic-ceramic fuel cells (PCFC) that are based on doped-perovskite membranes [32, 35]. In such cases, establishing the relationships between E^{OCV} and E^{REV} is more difficult. The doped-perovskite membranes typically have three or more charged defects, including small polarons that lead to so-called electronic leakage. Even at open-circuit (i.e., an infinite resistance external circuit), there is ionic and electronic transport through the membrane. In general, because of electronic leakage, $E^{\mathrm{OCV}} < E^{\mathrm{REV}}$. Models must be able to represent such behavior, which depends on mixed-conducting transport within the electrolyte membrane phase [32, 62].

4.5. Microscopic reversibility

Maintaining thermodynamic consistency and microscopic reversibility are important considerations in most, if not all, chemical kinetics modeling. It is important that the long-time (infinite time) result of kinetics approaches the thermodynamic equilibrium state. Achieving thermodynamic consistency requires knowing thermodynamic properties for all species participating in the kinetics. With known thermodynamic properties, the equilibrium state can be evaluated independently of any kinetics behaviors by minimizing the Gibbs free energy. With gas-phase mixtures, the equilibrium composition is easily evaluated because the thermodynamics for most gas-phase species are readily available. In electrochemistry, the situation is quite different.

Consider an elementary electrochemical reaction mechanism, with the rates being expressed in the form of Marcus theory. Such mechanisms (e.g., Reactions 34–37) typically involve ionic species and surface adsorbates, the thermodynamic properties for which are largely unknown. If the thermodynamics are not known, then equilibrium state cannot be independently established from the thermodynamics (cf, Eq. 40). In electrochemistry, the equilibrium electrostatic potential differences at phase interfaces is an important aspect of the thermodynamic equilibrium. For applications such as batteries or fuel cells, predicting correct open-circuit potentials is a critically important aspect of a model. Thus, it is essential that a model using the Marcus formalism must have complete thermodynamic properties for all species.

In the Butler–Volmer setting, the reversible half-cell potentials can be specified independently. Then, the activation overpotentials are measured relative to the reversible potentials (Eq. 29). Thus, a model based on the Butler–Volmer formulation is inherently suited to respecting known reversible potentials and open-circuit potentials. However, the Butler–Volmer formulation depends on simplifying assumptions, some of which may not be consistent with the intent of the multi-step reaction mechanism. As reaction mechanisms become more complex, the Butler–Volmer formulation becomes increasingly cumbersome. Moreover, depending on details of the multi-step reaction mechanism, thermodynamic properties for individual species may be required to derive the Butler–Volmer exchange current densities. Thus, the potential value of the Butler–Volmer framework is diminished, and may be inappropriate.

The Marcus framework is certainly the most appropriate as the electrochemistry complexity increases. However, because the Marcus framework demands thermodynamic properties, there is a need to establish and validate the needed properties. Direct measurements, if possible, are difficult. Thus, there are opportunities for developing new atomistic-scale and *ab initio* simulation capabilities that are specifically targeted toward ionic species and surface adsorbates.

4.6. Macroscopic irreversibility

There are numerous examples where electrochemical cells show macroscopically irreversible features. In lithium-ion batteries, the SEI is observed to form at low anodic half-cell potentials, but does not completely decompose when going back to high potentials. Simple Tafel kinetics have been used to describe this behavior [63], which, however, clearly violate microscopic reversibility. In Li/O_2 batteries, charging typically proceeds at considerably higher overpotentials than discharge, leading to a strong asymmetry in the charge/discharge behavior [27]. In PEM fuel cells, observed open-circuit voltage is considerably lower than the value predicted by thermodynamics [64]. In all these cases, Butler-Volmer kinetics fail to describe the experimental observations. The understanding and prediction of macroscopic irreversibility remains a challenge for kinetic modeling.

5. Charge transport

As demonstrated, predicting charge-transfer reaction rates requires knowledge about the electrostatic potentials and species concentrations in participating phases. Both quantities are affected by the need to transport charged species to the relevant electrode-electrolyte interfaces. Transport of charged species may be modeled using dilute or concentrated solution theory. Phase electrostatic potentials may be modeled by solving a Poisson equation or by assuming strict electroneutrality.

As a practical matter, the much-simpler dilute approximation is the most widely used transport model. The dilute approximation usually leads to formulating problems in the context of Nernst–Planck fluxes. The concentrated-solution theory is often applied in ways that are limited to binary systems, meaning only one cation and one anion. Transport within crystalline lattice structures typically must constrain the mobilities of charged defects to preserve site and charge balances at the lattice scale.

Generally, charged-species fluxes $\mathbf{J}_k$ can be represented as

$$\mathbf{J}_k = -u_k[X_k]\nabla\tilde{\mu}_k + [X_k]\,\mathbf{v}, \tag{52}$$

where u_k and $\mathbf{v}$ represent the ion mobilities and bulk phase velocity, respectively. Often, the bulk velocity is negligibly small, and certainly vanishes for solid electrolytes. Yet, in systems with strong volume changes within the composite electrode (e.g., Li/O_2 cells), bulk velocity may become important. Given a set of fluxes $\mathbf{J}_k$, the species concentrations can be expressed generally as a conserved quantity and integrated as a function of time:

$$\frac{\partial\,[X_k]}{\partial t} = -\nabla\cdot\mathbf{J}_k + \dot{\omega}_k W_k. \tag{53}$$

Here $\dot{\omega}_k$ is a volumetric production rate and W_k are the species molecular weights. Additional relationships are needed to determine the electrostatic potentials Φ (cf., Section 5.1) and convective velocities $\mathbf{v}$. The current density within the phase can be expressed in terms of the charge-species fluxes as

$$\mathbf{i} = F\sum_k z_k\mathbf{J}_k. \tag{54}$$

5.1. Electroneutrality

Predicting the electrostatic potential fields is an essential element of all electrochemistry models. Requiring electroneutrality within a phase can be a practical and reasonably accurate assumption. Strict electroneutrality means that

$$\sum_k F z_k\,[X_k] = 0, \tag{55}$$

everywhere within the bulk phase. Strict charge neutrality can also be enforced as

$$\nabla\cdot\mathbf{i} = 0. \tag{56}$$

Equation 56 is typically referred to as *conservation of charge*. Although strict electroneutrality (or *conservation of charge*) can be a reasonably accurate approximation, it is not entirely correct. Rather, the electrostatic potential is more accurately governed by a Poisson equation as

$$\nabla\cdot(\varepsilon\nabla\Phi) = -\rho = -\sum_k F z_k\,[X_k]\,, \tag{57}$$

where ε is the phase's dielectric constant and ρ is the local charge density [12]. Interestingly, while Eq. 55 is roughly correct (i.e., $\sum_k F z_k\,[X_k] \approx 0$), it also is the case that

$$\nabla\cdot(\varepsilon\nabla\Phi) \neq 0. \tag{58}$$

In other words, assuming that $\nabla \cdot (\varepsilon \nabla \Phi) = 0$ is *incorrect*! As discussed by Bazant and colleagues [65], because typically $\varepsilon \ll 1$, small (second-order) variations in ρ lead to large (leading order) variations in Φ. When Eq. 57 is solved coupled together with species-conservation equations (e.g., Eq. 53), the resulting models are called Nernst–Planck–Poisson (NPP) models [66].

Most electrochemistry models, at least those at the technology scale (e.g., batteries, fuel cells, etc.) assume a sharp "jump" in the electrostatic potential across phase boundaries. This jump $\Delta\Phi$ is used to evaluate electrochemical charge-transfer rates at the phase interfaces. However, by solving Eq. 57 on very small length scales near the phase interfaces, it is possible to resolve the electrical *double layers* at the phase boundaries. In other words, it is possible to resolve the electrostatic potential and species profiles within the extremely thin double layers. Nevertheless, at the technology level, there is usually no practical value to resolving the double-layer structure. Moreover, because of the enormous scale disparity, it is entirely impractical to resolve double-layer structure together with predicting larger-scale electrochemical performance.

5.2. Dilute solution theory

Dilute solution theory assumes that within an ionic solution, the charged species do not directly interact. The species flux can be expanded using the Nernst–Einstein equation ($D_k = RTu_k$) to

$$\mathbf{J}_k = -D_k \frac{\nabla(\gamma_k [X_k])}{\gamma_k} - D_k \frac{z_k F}{RT} [X_k] \nabla\Phi + [X_k]\mathbf{v}, \quad (59)$$

where D_k are species diffusion coefficients. For dilute solutions, the activity coefficients γ_k approach unity for all species, which results in [12]

$$\mathbf{J}_k = \underbrace{-D_k \nabla [X_k]}_{\text{diffusion}} \underbrace{-D_k \frac{z_k F}{RT} [X_k] \nabla\Phi}_{\text{migration}} + \underbrace{[X_k]\mathbf{v}}_{\text{convection}}. \quad (60)$$

The species fluxes are driven by diffusion and migration, and possibly convection.

Using Eq. 60, the current density, ionic conductivity, and transference numbers t_k can be derived [12]. The current density can be expressed as

$$\begin{aligned}\mathbf{i} = &- F \sum_k z_k D_k \nabla [X_k] - F^2 \nabla\Phi \sum_k z_k^2 u_k [X_k] \\ &+ F\mathbf{v} \sum_k z_k [X_k]. \end{aligned} \quad (61)$$

The ionic conductivity σ is obtained in the limit of no concentration gradients and no bulk velocity,

$$\sigma = F^2 \sum_k z_k^2 u_k [X_k]. \quad (62)$$

The transference number t_k is defined as the fraction of current carried by species k within an electrolyte. Continuing to assume negligible concentration gradients and bulk velocity, the transference number can be expressed as

$$t_k \mathbf{i} = -F^2 z_k^2 u_k [X_k] \nabla\Phi = \frac{z_k^2 u_k [X_k]}{\sum_k z_k^2 u_k [X_k]} \mathbf{i}, \quad (63)$$

$$t_k = \frac{z_k^2 u_k [X_k]}{\sum_k z_k^2 u_k [X_k]}. \quad (64)$$

By definition, the transference numbers, when summed over all mobile ions in a phase, must sum to unity.

A central tenet of the dilute solution theory is that the flux of species k depends on diffusion coefficients and mobilities that are associated with species k alone. In the more general case, the fluxes depend on diffusion coefficient matrices that represent the interactions between species (i.e., off-diagonal Onsager contributions).

5.2.1. Dilute solution binary electrolyte

Significant simplifications are possible for electrolytes that carry only two charged species. Assuming, for example, a liquid electrolyte solution composed of a salt that dissociates into one cation and one anion. Let ν_+ and ν_- be the number of cations and anions produced by dissolving electrolyte salt[1]. Assuming strict electroneutrality, the anion and cation concentrations ($[X_-]$ and $[X_+]$) within the electrolyte are related to the original salt concentration C as

$$C = \frac{[X_+]}{\nu_+} = \frac{[X_-]}{\nu_-}. \quad (65)$$

Assuming no homogeneous reactions within the solvent and no bulk velocity, the species continuity equation can be expressed as

$$\frac{\partial C}{\partial t} = \mathcal{D}\nabla^2 C, \quad (66)$$

where

$$\mathcal{D} = \frac{z_+ u_+ D_- - z_- u_- D_+}{z_+ u_+ - z_- u_-} \quad (67)$$

[1] Newman introduced a ± nonclature that is widely used. However, it is necessarily restricted to binary mixtures

is called the *ambipolar* diffusion coefficient [12]. The diffusion coefficient can be simplified using the Nernst–Einstein equation ($D_k = RTu_k$) to

$$\mathcal{D} = \frac{D_+ D_- (z_+ - z_-)}{z_+ D_+ - z_- D_-}. \tag{68}$$

The ambipolar diffusion coefficient is used frequently in modeling solid-state ceramic electrolytes, where it is also called the *chemical diffusion coefficient*. Although the application is limited to dilute binary systems, the explicit consideration of the electrostatic potential is eliminated. The resulting mathematical problem can be treated as a relatively straightforward diffusion problem [66–68].

5.3. Concentrated solution theory

The multicomponent transport for concentrated solution theory can be represented generally as

$$\begin{aligned}[X_k]\,\nabla\tilde{\mu}_k &= \sum_j K_{kj}\left(\mathbf{v}_j - \mathbf{v}_k\right) \\ &= \frac{RT\,[X_k]}{[X_T]} \sum_j \frac{\left[X_j\right]}{\mathcal{D}_{kj}}\left(\mathbf{v}_j - \mathbf{v}_k\right),\end{aligned} \tag{69}$$

where K_{kj} are the *friction coefficients* (also referred to as interaction coefficients), $[X_T] = \sum_k [X_k]$ is the total concentration, and $\mathcal{D}_{kj}$ are binary diffusion coefficients. By definition, the binary-diffusion-coefficient matrix is symmetric ($\mathcal{D}_{kj} = \mathcal{D}_{jk}$). The so-called self-diffusion coefficient $\mathcal{D}_{kk}$ can be evaluated, but has little or no physical utility. Equation 69 is analogous to the Stefan–Maxwell equations [69] and equivalent to those developed by Onsager [12, 70].

As discussed by Newman and Thomas-Alyea [12], concentrated solution theory (Eq. 69) is typically inverted to evaluate the diffusion velocities. The concentrated solution theory can be expressed as

$$[X_k]\,\nabla\tilde{\mu}_k = \sum_j M_{kj}\left(\mathbf{v}_j - \mathbf{v}_0\right), \tag{70}$$

where

$$M_{kj} = \begin{cases} K_{kj} & \text{if } k \neq j \\ K_{kj} - \sum_i K_{ki}, & \text{if } k = j. \end{cases} \tag{71}$$

The diffusion velocities can then be expressed as

$$\mathbf{v}_j - \mathbf{v}_0 = -\sum_{k\neq 0} L^0_{kj}\,[X_k]\,\nabla\tilde{\mu}_k, \quad j \neq 0, \tag{72}$$

where L^0_{kj} is evaluated as the inverse of the M^0_{kj} matrix as

$$L^0_{kj} = -\left(M^0_{kj}\right)^{-1}. \tag{73}$$

The submatrix M^0_{kj} is the matrix M_{kj} with the row and column of a reference species '0' removed. The species fluxes can be expressed as

$$\mathbf{J}_k = [X_k]\,\mathbf{v}_0 - \sum_{j\neq 0} L^0_{kj}\left[X_j\right][X_k]\,\nabla\tilde{\mu}_j. \tag{74}$$

The velocity of species '0' is such that the net diffusive flux vanishes. That is,

$$\sum_k \mathbf{J}_k = \mathbf{v}, \tag{75}$$

where $\mathbf{v}$ is the bulk velocity. The current density can then be evaluated as

$$\begin{aligned}\mathbf{i} &= F\sum_k z_k\,[X_k]\,\mathbf{v}_k = F\sum_k z_k\,[X_k]\,(\mathbf{v}_k - \mathbf{v}_0), \\ &= -F\sum_{k\neq 0} z_k\,[X_k] \sum_{j\neq 0} L^0_{kj}\left[X_j\right]\nabla\tilde{\mu}_j.\end{aligned} \tag{76}$$

In a solution with spatially uniform species concentrations (i.e., $\nabla\tilde{\mu}_k = z_k F\nabla\phi$), the conductivity of the solution is

$$\sigma = -F^2 \sum_{k\neq 0}\sum_{j\neq 0} L^0_{jk} z_j z_k [X_k][X_j], \tag{77}$$

and the transference number with respect to the reference species '0' is

$$t^0_k = \frac{z_k[X_k]F^2}{\sigma} \sum_{j\neq 0} L^0_{kj} z_j [X_j]. \tag{78}$$

It should be noted that the transference number for a concentrated solution *depends* on species '0'. Thus, unlike dilute solution theory, the transference number t^0_k is no longer analogous to the fraction of current carried by species k. The sum of the transference numbers must be unity, independent of the reference species '0' [12].

5.3.1. Concentrated solution binary electrolyte

A binary electrolyte is composed of a single anion specie, a single cation specie, and the solvent. The following analysis uses subscripts '+', '−', '0' to represent the anion, cation, and solvent, respectively. Concentrated solution theory for a binary electrolyte can be expressed as

$$\begin{aligned}[X_+]\,\nabla\mu_+ = {} & RT\frac{[X_+]\,[X_0]}{[X_T]\,\mathcal{D}_{+0}}\,(\mathbf{v_0} - \mathbf{v}_+) \\ & + RT\frac{[X_+]\,[X_-]}{[X_T]\,\mathcal{D}_{+-}}\,(\mathbf{v}_- - \mathbf{v}_+),\end{aligned} \tag{79}$$

$$[X_-]\nabla\mu_- = RT\frac{[X_-][X_0]}{[X_T]\mathcal{D}_{-0}}(\mathbf{v_0}-\mathbf{v}_-) + RT\frac{[X_-][X_+]}{[X_T]\mathcal{D}_{-+}}(\mathbf{v}_+-\mathbf{v}_-), \tag{80}$$

The cation flux can be expressed as

$$\begin{aligned}\mathbf{J}_+ &= [X_+]\mathbf{v}_+ \\ &= \frac{1}{\mathcal{D}_{+-}[X_0]-\mathcal{D}_{+0}[X_-]}\Big(\frac{[X_+][X_T]\mathcal{D}_{+-}\mathcal{D}_{+0}}{RT}\nabla\mu_+ \\ &\quad + [X_0][X_+]\mathcal{D}_{+-}\mathbf{v}_0 + [X_+][X_-]\mathcal{D}_{+0}\mathbf{v}_-\Big).\end{aligned} \tag{81}$$

It can be shown that the anion flux, cation flux, current density, and transference number (Eqs. 54, 69 and 78) can be combined such that [12, 71]

$$\mathbf{J}_+ = \frac{[X_+][X_T]\mathcal{D}}{[X_0]RT(\nu_++\nu_-)}\nabla(\nu_+\mu_++\nu_-\mu_-) + \frac{\mathbf{i}t_+^0}{z_+F}+[X_+]\mathbf{v}_0, \tag{82}$$

$$\mathbf{J}_- = \frac{[X_-][X_T]\mathcal{D}}{[X_0]RT(\nu_++\nu_-)}\nabla(\nu_+\mu_++\nu_-\mu_-) + \frac{\mathbf{i}t_-^0}{z_-F}+[X_-]\mathbf{v}_0, \tag{83}$$

where $\mathcal{D}$ is the diffusion coefficient 'based on the thermodynamic driving force' and is similar to the dilute-solution diffusion coefficient (see Eq. 68)

$$\mathcal{D} = \frac{\mathcal{D}_{0+}\mathcal{D}_{0-}(z_+-z_-)}{z_+\mathcal{D}_{0+}-z_-\mathcal{D}_{0-}}. \tag{84}$$

The cation transference number for a binary concentrated solution is expressed as

$$t_+^0 = \frac{z_+\mathcal{D}_{0+}}{z_+\mathcal{D}_{0+}-z_-\mathcal{D}_{0-}}. \tag{85}$$

The diffusion coefficient for concentrated solutions is usually measured based on the gradient of concentration

$$D = \mathcal{D}\frac{[X_T]}{[X_0]}\left(1+\frac{d\ln\gamma_\pm}{d\ln m}\right), \tag{86}$$

where $\gamma_\pm$ is the mean molal activity coefficient and m is the molality [12, 71, 72]. The species continuity equations (assuming no bulk velocity or homogeneous reactions) can then be combined using strict electroneutrality as

$$\frac{\partial C}{\partial t} = \nabla\cdot\left[D\left(1-\frac{d\ln[X_0]}{d\ln C}\right)\nabla C - \frac{\mathbf{i}\,t_+^0}{z_+\nu_+F}\right]. \tag{87}$$

5.4. Binary diffusion coefficients in liquids

There are several approaches to approximating the binary diffusion coefficients in liquids, with the Stokes–Einstein relationship being widely used. However, the Stokes–Einstein relation is most accurate for infinitively dilute systems. Empirical relationships to modify the Stokes–Einstein relationship have been developed to more accurately predict the binary diffusion coefficients in dilute systems. In concentrated solutions, experimental studies have shown that the binary diffusion coefficients depend on concentrations, temperatures, and viscosities [69].

In concentrated solutions, electrical interactions strongly influence ion mobility [73, 74]. In accordance with the Debye–Hückle theory, ions tend to form "ion atmospheres" that can increase or decrease mobility of charged species due to positive and negative charge interactions.

5.4.1. Stokes–Einstein binary diffusion coefficients

The Einstein diffusion equation can be stated generally as

$$\mathcal{D}_{kj} = \frac{k_\mathrm{B}T}{\zeta}, \tag{88}$$

where k_B is the Boltzmann constant, T is temperature, and ζ is a friction coefficient [73, 75]. If the fluid is isotropic, incompressible, and Newtonian, and the boundary between the particles is fixed, high-shear, and nonslip, then Stokes's law for creeping flow states

$$\zeta = 6\pi\eta_j r_k, \tag{89}$$

where η_j is the fluid viscosity of species j and r_k is the molecular radius of species k. If the fluid is Newtonian and there is zero-shear between the particles, then the friction coefficient can be expressed as

$$\zeta = 4\pi\eta_j r_k. \tag{90}$$

If there is shear and slip between the particles, the friction coefficient can be expressed as

$$\zeta = 6\pi\eta_j r_k\left(\frac{2\eta_j+\upsilon r_k}{3\mu_j+\upsilon r_k}\right), \tag{91}$$

where υ is the sliding coefficient [69, 75, 76]. If there is high shear and no slip between the particles ($\upsilon\to\infty$), Eq. 91 simplifies to Eq. 89. If there is zero shear and high slip ($\upsilon\to 0$), Eq. 91 simplifies to Eq. 90.

The Stokes–Einstein relationships are most accurate for infinitively dilute concentrations. Theoretical adjustments for polar and non-circular molecules have

been proposed to modify the friction coefficient ζ [73]. Empirical relations modifying the Stokes–Einstein relation for infinitely dilute solutions have also been developed to add concentration, activity, and viscosity dependence [77]. In addition, the product of the binary diffusion coefficient involving the solvent and the viscosity η have been shown to be reasonably constant over moderate concentration ranges (e.g. $\eta\mathcal{D}_{k0}$ is constant) [69]. The binary diffusion coefficients between ions, according to Debye-Hückel-Onsager theory, is expected to be proportional to $\sqrt{C}$ [72, 74]. Additional theories for concentrated solutions have been developed to better describe concentrated solution performance [78–80]. Complex electrochemistry software must able to accommodate concentrated solution theories with varying levels of complexity, to reduce the computational burden placed on the software user.

5.5. Charged transport in solids

Broadly speaking, there are three approaches to model charge and species transport within solids. These approaches, in ascending order of complexity, are

- Solid phases with effective properties
- Dilute defect solids (negligible Onsager cross-diagonal coefficients)
- Concentrated defect solids (significant Onsager cross-diagonal coefficients)

Battery electrodes are commonly modeled using solid phase models with effective properties [7, 9, 20, 48, 81, 82]. Such models assume that the mobile solid-phase species is charge-neutral. Thus, the species transport and charge transport are no longer explicitly coupled. These models represent species fluxes as

$$\mathbf{J}_k = -D_{k,\text{eff}} \nabla \left(\gamma_k \left[X_k \right] \right), \tag{92}$$

and current density as

$$\mathbf{i} = -\sigma_{\text{eff}} \nabla \Phi. \tag{93}$$

In these expressions, $D_{k,\text{eff}}$ and σ_{eff} are the effective diffusion coefficients and the effective conductivity, respectively. The effective parameters can be functions of concentration, thus coupling the charge and species equations.

The differences between the dilute defect and concentrated solid methods can be viewed in the context of the Onsager matrix. The Onsager coefficients $L_{k,i}$ relate the electrochemical potentials of mobile species and their respective fluxes. Mathematically, this relationship can be expressed as

$$\begin{bmatrix} \mathbf{J}_1 \\ \vdots \\ \mathbf{J}_k \\ \vdots \\ \mathbf{J}_K \end{bmatrix} = \begin{bmatrix} L_{1,1} & L_{1,2} & \dots & L_{1,K} \\ \vdots & \vdots & \ddots & \vdots \\ L_{k,1} & L_{k,2} & \dots & L_{k,K} \\ \vdots & \vdots & \ddots & \vdots \\ L_{K,1} & L_{K,2} & \dots & L_{K,K} \end{bmatrix} \begin{bmatrix} \nabla\tilde{\mu}_1 \\ \vdots \\ \nabla\tilde{\mu}_k \\ \vdots \\ \nabla\tilde{\mu}_K \end{bmatrix}. \tag{94}$$

As derived by Osager and Fuoss [74], the Onsager coefficient matrix is symmetric. If the flux of species k is independent of other species, the off-diagonal Onsager coefficients $L_{k,i}$ vanish. The off-diagonal coefficients are negligible if the short-range and long-range interactions between charged species are negligible [83]. In this text the *dilute defect solids* are models that assume that all off-diagonal Onsager coefficients are negligible and conversely the *concentrated defect solids* assume that there are significant off-diagonal coefficients.

Dilute defect models are used when the solid-phase species are charged, but their mobilities are independent of other charged species. For *dilute defect solids* the flux of species k is the same as dilute solute solution theory with the bulk velocity being zero (cf., Eq. 60) [32, 36, 50, 55, 57, 66, 68].

Concentrated defect models are used when the solid-phase species are charged, and the species' mobilities depend on other charged species [83]. The resultant species fluxes, which are analogous to concentrated solution theory, can be expressed as

$$\mathbf{J}_k = \sum_i L_{k,i} \nabla\tilde{\mu}_i. \tag{95}$$

Equation 95 shows that the flux of species k depends on the electrochemical-potential gradients of all other species that have non-zero Onsager coefficients. Thus, the species transport equations are tightly coupled. The off-diagonal Onsager coefficients $L_{i,k}$ must be estimated or measured for every species pairing. In principle, these parameters can be measured using electron blocking electrodes and material blocking electrodes [84, 85]. However, because there are numerous potential species combinations, many Onsager coefficients need to be estimated.

5.5.1. Lattice-scale site and charge balances

In solid-phase crystalline electrolytes, models must be concerned with lattice-scale constraints. For example, consider strontium- and iron-doped lanthanum cobaltites, which can be generally represented as $La_{1-x}Sr_xCo_{1-y}Fe_yO_{3-\delta}$ [50]. It is assumed that three

cobalt oxidation states (Co^{2+}, Co^{3+}, and Co^{4+}) may all be present simultaneously, but that the iron remains as Fe^{4+} [86]. Using Kröger–Vink notation, "incorporation" and "disproportionation" reactions may be written, respectively, as

$$\frac{1}{2}O_2 + V_O^{\bullet\bullet} + 2Co_{Co}^{\times} \rightleftharpoons O_O^{\times} + 2Co_{Co}^{\bullet}, \tag{96}$$

$$2Co_{Co}^{\times} \rightleftharpoons Co_{Co}' + Co_{Co}^{\bullet}. \tag{97}$$

The superscripts "×", "•", and "′" indicate charge relative to the fully occupied crystal lattice, with "×" being neutral, "•" meaning $z_k = +1$, and "′" meaning $z_k = -1$. An oxygen vacancy $V_O^{\bullet\bullet}$ has a $z_k = +2$ charge, relative to the oxygen anion that would ordinarily occupy the site. The reduced cobalt Co_{Co}' and oxidized cobalt $Co_{Co}^{\bullet}$ are both expected to behave as small polarons [86].

The equilibrium composition must be constrained by site and charge balances. The charge balance (electroneutrality) is stated as

$$[Co_{Co}^{\bullet}]_L - [Co_{Co}']_L + 2[V_O^{\bullet\bullet}]_L + [Fe_{Co}^{\bullet}]_L - [Sr_{La}']_L = 0. \tag{98}$$

In this expression, the subscript "L" in $[X]_L$ indicates lattice-scale concentrations. In the doped perovskite crystal structure (ABO_3), site balances must be enforced for the A and B sites. The cobalt (B-site) site balance is stated as

$$[Co_{Co}^{\bullet}]_L + [Co_{Co}']_L + [Co_{Co}^{\times}]_L + [Fe_{Co}^{\bullet}]_L = 1. \tag{99}$$

The oxygen site balance requires that

$$[O_O^{\times}]_L + [V_O^{\bullet\bullet}]_L = 3. \tag{100}$$

In this formulation, it is assumed that $[Sr_{La}']_L$ and $[Fe_{Co}^{\bullet}]_L$ are fixed by the doping levels ($La_{1-x}Sr_xCo_{1-y}Fe_yO_{3-\delta}$) as

$$[Sr_{La}']_L = x, \quad [Fe_{Co}^{\bullet}]_L = y. \tag{101}$$

The oxygen non-stoichiometry is assumed to be $\delta = [V_O^{\bullet\bullet}]_L$.

6. Thermal considerations

The design and control of electrochemical devices can be substantially influenced by thermal behaviors. Li-ion battery safety and lifetime, for example, are directly affected by internal heating and thermal runaway. In applications such as fuel cells and electrolyzers, the management of significant heating within the ion-conducting membranes directly affects design and implementation.

The thermal behaviors within electrodes and electrolytes are intimately coupled with the electrochemical processes. Broadly speaking, electrochemical processes produce heat in two ways. *Faradaic* heating is the result of inefficiencies (irreversibilities) in charge-transfer chemistry at the electrode-electrolyte interfaces. *Ohmic* heating (Joule heating) is the result of ionic and electronic fluxes through electrode and electrolyte phases. Additionally, heat release of *thermochemical* (non-charge-transfer) main or side reactions in complex reaction systems contribute to the overall thermal behavior.

6.1. Faradaic heating

The Faradaic heat-generation rate $\dot{q}_{F,i}$ that is associated with a charge-transfer reaction at electrode and electrolyte interfaces can be evaluated as the difference of the enthalpy change and produced electric power from the ith electrochemical reaction. Stated mathematically,

$$\dot{q}_{F,i} = -i_{e,i}\frac{\Delta H_i}{n_{e,i}F} + i_{e,i}\left(\Phi_{ed} - \Phi_{el}\right), \tag{102}$$

where ΔH_i is the enthalpy change associated with the ith charge-transfer reaction.

Considering the general thermodynamic relationships,

$$\Delta H = \Delta G + T\Delta S, \tag{103}$$

with

$$\Delta S = -\left(\frac{\partial(\Delta G)}{\partial T}\right)_p, \tag{104}$$

the enthalpy difference associated with a charge-transfer reaction can be expressed as

$$\begin{aligned}\frac{\Delta H}{n_e F} &= \frac{\Delta G}{n_e F} + T\left[\frac{\partial}{\partial T}\left(-\frac{\Delta G}{n_e F}\right)\right]_p \\ &= E^{eq} - T\left(\frac{\partial E^{eq}}{\partial T}\right)_p.\end{aligned} \tag{105}$$

In this expression, E^{eq} is the equilibrium electrostatic-potential difference across the electrode-electrolyte interface. The equalibrium electrostatic potential E^{eq} is equivalent to the reversal potential E^{REV}.

The heat release associated with charge-transfer chemistry can be rewritten as

$$\begin{aligned}\dot{q}_{F,i} &= i_{e,i}\left(\left(\Phi_{ed} - \Phi_{el}\right) - E_i^{eq}\right) + i_{e,i}T\left(\frac{\partial E_i^{eq}}{\partial T}\right)_p \\ &= i_{e,i}\eta_{act,i} + i_{e,i}T\left(\frac{\partial E_i^{eq}}{\partial T}\right)_p,\end{aligned} \tag{106}$$

which indicates that there are two contributions to the Faradaic heat generation. One is the heat generation due to the inefficiency or activation polarization of the charge-transfer reactions. The other is the reversible entropic heat generation.

6.2. Ohmic heating

Although both electrode and electrolyte materials are conductors, they often offer significant resistances (i.e., relatively low conductivities). Electrodes are usually dominantly electronic conductors and usually have higher conductivities than the electrolytes. The electrolytes are dominantly ion conductors and usually have lower conductivities than the electrolytes. There can be significant Ohmic or Joule heating in both electrodes and electrolytes.

The local ohmic heat production due to ion and electron transport within an electrode or electrolyte can be expressed as

$$\dot{q}_{\text{ohm}} = \frac{\mathbf{i} \cdot \mathbf{i}}{\sigma}, \tag{107}$$

where $\mathbf{i}$ and σ are the current density and conductivity, respectively.

6.3. Heating due to non-charge-transfer reactions

Thermochemical reactions (i.e., reactions without charge transfer) can dominate thermal behavor of electrochemical cells. Examples include the (endothermal) steam reforming in hydrocarbon-fueled SOFCs and the (exothermal) decomposition chemistry taking place in lithium-ion batteries at high temperature, leading to thermal runaway. In a generalized expression, the heat release of an arbitrary reaction is given by

$$\dot{q}_{\text{r},i} = q_i \Big[-\Delta H_i + n_{e,i} F \left(\Phi_{\text{ed}} - \Phi_{\text{el}} \right) \Big], \tag{108}$$

where q_i the reaction rate of progress (cf. Eq. 20). This expression is valid for both electrochemical and thermochemical reactions, as $n_{e,i} = 0$ for the latter. While Eq. 106 shows reversible and irreversible heating as separate terms, the equivalent Eq. 108 is more directly related to the fundamental thermodynamic property ΔH_i and therefore easier to implement in a generalized Marcus formalism.

6.4. Energy balances

The Faradaic and Ohmic heating are ultimately incorporated as terms in energy-conservation equations, with the particular form of the energy equations depending on the geometry being modeled. In all cases, the ohmic heating appears as a source term within a phase-specific conservation equation.

Because the Faradaic heating appears at interfaces between phases, it can participate differently in different models. On one hand, the Faradaic heating could be viewed in the context of a boundary condition for a model that is focused on an electrolyte phase. On the other hand, consider a geometrically complex porous electrode model such as illustrated in Fig. 2. In this case, phase interfaces are distributed throughout the entire domain. If the porous structure is represented as a continuum with phase fractions (e.g., porosity), then the Faradaic heating is usually most easily represented as a source term based on interfacial area per unit volume.

The role of the electrochemistry software is to make functions available that can be used to evaluate the Faradaic and Ohmic heating rates. However, the results of these functions may be used differently in different models of electrochemical systems.

7. Software implementation

As a practical matter, general software implementations are needed. The user should be able to describe the chemistry in terms of clearly written and understandable material properties and reactions, with the software providing the needed functionality to pose and solve complex electrochemistry problems. The challenge for developing such software is much more complex than is the case for homogeneous gas-phase chemistry. At this time, no such *general purpose* electrochemistry is available to assist the research and development of critical electrochemistry applications and technologies.

Broadly speaking, the software must be designed and implemented to fulfill two functions:

- A user interface is needed within which to describe electrochemical kinetics and transport. Meeting this need requires defining a syntax that is sufficiently general to describe relevant complexity, but sufficiently structured so as to make rule-based communication efficient and comfortable. On one hand, if the syntax is too restrictive, then expressing the needed complexity may be impeded. If, on the other hand, the syntax is too general then it effectively becomes a programming language without the practical benefits of a convenient user interface. Finding the appropriate middle ground can be challenging.

- Once the user has communicated the electrochemical kinetics, such as via a reaction mechanism,

the software must make functions available that can be accessed by higher-level code to evaluate properties and rates as functions of state variables. This general software structure is already in place within the CANTERA software [87]. The challenge is to extend to embrace increasingly complex electrochemistry.

7.1. User interfaces

As in CHEMKIN and CANTERA, the user initially interacts with a file-based interface to describe a particular electrochemical system and associated thermodynamic, reaction, and transport properties. The general-purpose software makes functions available for use in evaluating terms in conservation equations. The input must specify phases and describe all species within each phase. Input syntax rules must be sufficiently flexible so as to accept any range of commonly available inputs which may be used to specify thermodynamic or transport parameters. For example, species transport properties can be given as either diffusion coefficients or as mobilities. Species activities can be defined by parameters specific to an equation of state, but these parameters can frequently be derived, in part, from fundamental physical parameters such as species critical properties [88].

7.1.1. Phases

Phases play essential roles in electrochemistry, with potentially great differences between the types of phases. For example, liquid-phase electrolytes are very different from solid-phase polycrystalline electrolytes. Many applications involve gas phases (e.g., solid-oxide fuel cells), which have no electrochemically specific attributes within the phase. Similarly, some applications involve heterogeneous reactions that proceed on a catalyst phase.

A phase definition must necessarily identify the physical form of the phase. Then, depending on the phase type, the required attributes can be quite different. For example, in a crystalline phase information is needed about crystal-lattice structure, dopants, etc. Issues such as site-balance and charge-balance constraints are relevant. Liquid phases need no such information, but properties such as phase density and viscosity are required. Additionally, the number of mobile ions or charged defects is very important. Significant simplifications are available for modeling phases with only two mobile ions (i.e., binary systems). Phases with three or more mobile ions require more information and are more complicated to model.

7.1.2. Species and charged defects

Each species or charged defect must be associated with a phase. Moreover, each species requires thermodynamic and transport properties. Thermodynamic properties are usually specified in terms of temperature-dependent fits to heat capacity, enthalpy, and entropy. Unlike the gas phase, the needed properties are often not known. Thus, as new materials and chemistries are developed, an important task is to establish properties for the participating species. This can be a lengthy and difficult task, with the resulting properties often being not entirely unique [36, 67, 68]. As new experimental methods and computational chemistry capabilities are developed to establish these parameters, it is important that general-purpose software can flexibly incorporate input parameters in a variety of forms, such as from atomic-scale *ab initio* simulations.

Transport properties may be represented variously as Onsager coefficients, mobilities, diffusion coefficients, or conductivities, which are generally interconnected. Transport properties for binary systems can benefit from significant simplifications. In addition to binary or multi-component considerations, transport properties may be represented in a dilute limit or may need to be represented using concentrated-solution theories. Of course, more information and material properties are required to represent the concentrated-solution theory. Thus, the user interface must be designed to accommodate different phase identities and the levels of transport theory.

7.1.3. Electrochemical reactions

Electrochemical reactions must be represented in a syntax that is easily understood. Representing electrochemistry in the framework of fundamental Marcus theory is most general. However, any general-purpose software must also be capable of representing charge-transfer reactions in the Butler–Volmer form. The Butler–Volmer approach depends on assumptions and approximations, but is very widely used in the electrochemical literature. As a practical matter, the Marcus and Butler–Volmer representations cannot be easily mixed within a particular reaction mechanism. Depending on which approach is used, the user may need to provide different information and properties. For example, expressions for reversible potentials are needed in any Butler–Volmer implementation. However, in the Marcus approach accurate thermodynamic properties are required to assure the accurate prediction of measurable open-circuit voltages.

Another important consideration involves whether or not Kröger–Vink notation is used. With systems in-

volving polycrystalline electrolytes, Kröger–Vink notation is the most natural. However, for systems such as those involving liquid electrolytes, Kröger–Vink notation is usually not appropriate. The Kröger–Vink notation is quite different from the notation commonly used in liquid-phase electrolytes. For example, a proton in a PEM fuel-cell membrane is usually represented as H^+. In the equivalent protonic-ceramic fuel cell with a polycrystalline proton-conducting membrane, the proton is represented as $OH_O^\bullet$, with the leading "O" indicating that the proton is associated with an oxygen site. Additionally, the immobile lattice vacancies are typically modeled as a mobile charged defect in Kröger-Vink notation.

General-purpose electrochemistry software must be capable of using the Kröger–Vink notation, where appropriate, and must be configured to work in a generalized manner with reactions in complex lattice phases that are common in solid-state conductors. Chemical kinetics software (such as CANTERA) verify that user-defined reactions balance elements and charge. In addition, for complex lattice phases the software must also be able to verify that lattice sites are conserved. In other words, a reaction must create or destroy species within the constraints of lattice characteristics. Lattice sites themselves cannot be created or destroyed.

7.2. User Input Example

A user-defined set of inputs in the CANTERA format provides an example of the required structure for a user input file for generalized electrochemical kinetics software calculations. For any generalized software program, the user is required to input information to communicate the following information to the software:

1. *Phase information* –For each phase, the user must specify the following:

   - Phase type
   - Elements involved
   - Species involved
   - Thermodynamic model
   - Transport model

   The thermodynamic model typically consists of an equation of state which provides a framework for calculating both phase-level thermodynamic quantities (e.g. Gibbs energy G and enthalpy H) and $p - v - T$ relationships. Moreover, the thermodynamic and transport models rely on individual species properties, as described below, but any information describing species interactions for either model must be provided at the 'phase' level. Partial molar thermodynamic quantities, for example, are calculated at the 'phase' level.

2. *Species information* – For every species included in a phase declaration, the user must provide:

   - Elements included
   - Species charge
   - Species thermodynamic parameters
   - Species transport parameters

   These parameters pertain to the individual species properties. These are provided to the respective phase's thermodynamic and transport models, which determine how the individual species interact to determine phase and species properties.

3. *Reaction information* – For each reaction, the user must provide the following:

   - Reaction type
   - Reaction equation
   - Reaction rate parameters

Figures 10–12 illustrate examples of how required information is provided to describe lithium intercalation at the interface between a graphite anode phase and a carbonate-based electrolyte. The text from all three figures is entered into a single input file, but are broken out separately here for discussion purposes. This input is framed within the context of a CANTERA input (`CTI`) file. Some of the functionality shown in this example already exists in CANTERA, while other capabilities are yet to be implemented. The model includes binary electrolyte transport to and from the anode-electrolyte interface according to concentrated solution theory, plus a two-step intercalation reaction (Reactions 2 and 3) wherein the lithium first adsorbs/plates on the anode surface (a charge-transfer step), and is subsequently incorporated into the graphite anode (a non-electrochemical interface reaction). More complex chemistry can be incorporated, particularly as related to SEI formation and evolution, but are omitted at present for clarity's sake. Regardless, any additional information will be input in a format similar to that currently shown in the figures.

7.3. Phase information

Figure 10 shows the required inputs related to phase information. Three phases are required to describe the intercalation process: a solid electrode phase (named

```
# CTI - Cantera Input file for Lithium intercalation at the interface between
#         a graphite anode and liquid carbonate electrolyte.

#-------------------------------------------------------------------------------
#  Phase data
#
#-------------------------------------------------------------------------------
# Bulk anode phase:
intercalation_electrode(name = "graphite",
   elements = " C Li E ",
   species = " C6(C_b) LiC6(C_b) electron(C_b) ",
   transport = "dilute_solid",
   thermo = RedlichKister(s1="LiC6(C_b)", s2="C6(C_b)",
     activity_coefficients=(binarySpeciesParameters(
       excess_enthalpy = [-3.268E6, 3.955E6,  -4.573E6, 6.147E6, -3.339E6,
              1.117E7,  2.997E5, -4.866E7,  1.362E5,  1.373E8,
                    -2.129E7, -1.722E8,  3.956E7,  9.302E7, -3.280E7],
       excess_entropy = 0.0))),
   molar_density = (3.15e-2, 'mol/cm3'))

# Bulk electrolyte phase:
liquid_electrolyte(name = "electrolyte",
   elements = " C H O Li E F P ",
   species = " Li+(e) PF6-(e) solvent(e) ",
   density = (1.208, 'g/cm3'),
   transport = binary_concentrated_solution(
     cation(name="Li+(e)", stoich=1),
     anion(name="PF6-(e)", stoich=1),
     solvent(name="solvent(e)"),
     viscosity_ref = (0.7, 'cP'),T_ref=293),
   thermo = RedlichKister(s1="Li+(e)", s2="PF6-(e)", s3="solvent(e)",
       activity_coefficients=(multiComponentSpeciesParameters(
         A = [[-1.287E6, 1.723E5,  1.353E6],
              [ 1.417E4, -7.319E7, -3.117E5],
              [-3.216E3,  6.215E5,  1.174E8]]))))

# Anode-electrolyte interface:
ideal_interface(name = "graphite_surf",
    elements = " Li ",
    species = "Li(C_s) (C_s)",
    site_density = (1.07e-5, 'mol/m2'),
    phases = " graphite electrolyte ")
```

Figure 10: Example CANTERA input (cti) file describing the necessary phase information for intercalation at the interface between a liquid carbonate electrolyte and solid graphite anode. The physical parameters should be considered as illustrative and some of the functionality may not yet be implemented in the released version of CANTERA.

'graphite,' here), a liquid electrolyte phase ('electrolyte'), and a phase to represent the interface between the two ('graphite_surf').

Graphite electrode – The graphite electrode phase is specified as an 'intercalation_electrode,' which would be linked to a set of governing equations for a solid phase with a lattice-like structure that can host a single intercalating species and which can also conduct electrons. The phase declaration contains lists of the elements and species included in the phase. This particular phase has three species: a C_6 graphite unit cell (a defacto vacancy), LiC_6, a graphite unit cell with intercalated lithium, and a delocalized electron. Each species name includes a reference to its phase ('C_b' = 'Carbon bulk'), thus differentiate it from a similarly-named species in another phase. The graphite phase declaration must also include a 'molar_density' input, which specifies the total molar concentration (kmol m^{-3}) of the available lattices sites. The molar concentration of intercalated lithium sites and vacancies must always sum to the number of available sites. This phase type is unique in that it contains some species bound to lattice sites and an electron which is not. Well defined rules must be established to differentiate the two types of species.

The phase description also specifies transport and thermodynamic models. In this example, transport calculations for species other than electrons use a dilute defect solid model (Section 5.5). Thermodynamic calculations use the Redlich–Kister model to evaluate the excess Gibbs free energy for the intercalation reaction in the following form [9]

$$n_{\mathrm{T}} g^{\mathrm{E}} = [X_{\mathrm{s}1}]\,[X_{\mathrm{s}2}] \sum_{\ell} A_\ell \left([X_{\mathrm{s}1}] - [X_{\mathrm{s}2}]\right)^\ell , \qquad (109)$$

where the bulk lattice only has two species, and A is the vector input for 'excess_enthalpy'. The excess free energies are labeled "s1" and "s2" to associate the species the excess free energies.

Liquid electrolyte – The electrolyte phase declaration proceeds in a manner similar to the electrode phase. It is declared as a 'liquid_electrolyte', which is an incompressible phase, similar to the 'intercalation_electrode', but without any restrictions related to lattice sites and without electrical conductivity. The phase declares three species: a Li^+ cation, a PF_6^- anion, and a solvent. The suffix '(e)' is given to each species name, both for clarity and to differentiate from potential similarly-named species in other phases. A constant mass density is also provided.

The input file instructs the electrolyte transport calculations to use concentrated solution theory for a binary salt in a liquid solvent. At the phase level, the user must indicate which species names correspond to the anion, cation, and solvent, and must specify the stoichiometry of the cation and anion (ν_+ and ν_-, respectively). For the solvent, the user must also provide the viscosity. The phase thermodynamic quantities here will also be calculated using a Redlich–Kister model for the Gibbs excess energy. Because the electrolyte has three interacting species, the 'binarySpeciesParameters' from the graphite electrode cannot be used, here. As described by O'Connell and Haile [46], the binary Redlich–Kister expansion can be extended to a multicomponent form

$$n_{\mathrm{T}} g^{\mathrm{E}} = g_{12} + g_{13} + g_{23}, \qquad (110)$$

$$\frac{g_{ik}}{RT} = [X_i]\,[X_k]\Big(A_{ik} + B_{ik}\left([X_i] + [X_k]\right) + \ldots\Big) \qquad (111)$$

Each term in the expansion requires a series of input matrices of $N_{\mathrm{species}} \times N_{\mathrm{species}}$ species interaction parameters

to calculate the excess energy. The example in Fig. 10 is restricted to just a single input matrix A_{ik}.

Anode-electrolyte interface – Finally, the interface between the anode and electrolyte must be defined as a phase. It is specified here as an 'ideal_interface', which dictates the form of the chemical potential calculations. The interface phase must provide lists of elements and species. It must also specify a 'site_density' (i.e., the total number of available sites to host surface species, similar to the 'molar_density' in the graphite phase). Additionally, the interface specification must provide a list of the phases that can participate in any interface reactions. The current mechanism includes an empty surface site (a vacancy, '(C_s)') and an adsorbed lithium species at the graphite–electrolyte interface. However, an interface phase is required even for interfacial reaction mechanisms where no intermediate surface phases are defined (e.g. where the Li ion from the electrolyte is transferred directly to the bulk graphite). This informs the software that the participating bulk phases are in direct physical contact, and to create a separate kinetics manager to calculate relevant terms for reactions involving the participating phases.

7.4. Species information

Figure 11 shows an example of the species input format for the phases described in Figure 10. Each species listed in a phase declaration must have its own species declaration in the input file. The species entry must give the species name (which matches that provided in the phase declaration), a chemical formula (which must utilize only those elements listed in the phase declaration), a species thermodynamic entry and a species transport entry. The charge of the species is indicated by the number of electrons in the species (the number of electrons < 0 for a positive charge). Because the Li-ion battery will operate within a relatively narrow temperature range, a relatively simple species thermodynamic model, which assumes a constant specific heat c_p, is specified. Other forms (such as NASA polynomials) are available for incorporating complex temperature dependence. Note also that these terms are used to calculate the ideal, reference-state thermodynamic terms, to which activity and excess energy terms are added, using the information in the phase declaration. Additionally, the thermodynamic entry for the electron species in the carbon bulk specifies a 'metal_electron' model, where the activity is equal to the electrostatic potential times the species charge.

The species transport parameters reflect a range of different possible models. The C_6 lattice species ('C6(C_b)'), represents an immobile vacancy. Even though it is immobile, lattice site conservation implies that the vacancy flux must always be equal and opposite to that of the intercalating species. Thus, a transport model ('intercalation') is specified, but no diffusion coefficient is given, indicating the immobile lattice vacancy. In contrast, the 'LiC6(C_s)' species is given the same 'intercalation' transport model, but with

```
#-----------------------------------------------------------------------
#  Species data
#
#-----------------------------------------------------------------------
# Anode species:
species( name = "C6(C_b)",
  atoms = " C:6 ",
  thermo = const_cp(h0 = (0.0, 'kcal/mol')),
  transport = intercalation(diffCoeff='None'))

species( name = "LiC6(C_b)",
  atoms = " C:6 Li:1 E:-1",
  thermo = const_cp(h0 = (-11.65, 'kJ/mol'), s0 = (0, 'kJ/mol')),
  transport = intercalation(diffCoeff=(2.0e-16,'m2/s'))

species( name = "electron(C_b)",
  atoms = "E:1",
  thermo = metal_electron,
  transport = metal_electron(conductivity=(1.0e4, 'S/m'))

# Electrolyte species:
species( name = "Li+(e)",
  atoms = " Li:1 E:-1 ",
  thermo = const_cp(h0 = (0.0, 'kJ/mol')),
  transport = Stokes_Einstein(geom='spherical',
          radius = (0.182, 'nm'),
          sliding_factor = 0))

species( name = "PF6-(e)",
  atoms = "P:1, F:6, E:1",
  thermo = const_cp(h0 = (0.0, 'kJ/mol'), s0 = (0.0, 'J/mol/K')),
  transport = Stokes_Einstein(geom='spherical',
          radius = (0.242, 'nm'),
          sliding_factor = 0))

species( name = "Solvent(e)",
  atoms = "C:3, H:4, O:3",
  thermo = const_cp(h0 = (0.0, 'kJ/mol'), s0 = (0.0, 'J/mol/K')),
  transport = 'None')

# Graphite surface species:
species( name = "Li(C_s)",
  atoms = " Li:1 ",
  thermo = const_cp(h0 = (0.0, 'kJ/mol'), s0 = (0.0, 'J/mol/K'),
    cp0 = (3.56, 'kJ/kg/K'), T0 = (298, 'K')),
  transport = 'None')

species( name = "(C_s)",
  atoms = "  ",
  thermo = const_cp(h0 = (0.0, 'kJ/mol'), s0 = (0.0, 'J/mol/K')),
  transport = 'None')
```

Figure 11: Example CANTERA input ('cti') file describing the necessary species information for intercalation at the interface between a liquid carbonate electrolyte and solid graphite anode. The physical parameters should be considered as illustrative and some of the functionality may not yet be implemented in the released version of CANTERA.

a non-zero diffusion coefficient. The electron species in the graphite is given a 'metal_electron' transport model, where the electronic current density is calculated according to Ohm's law (Eq. 93).

For the cation and anion in the electrolyte, parameters (a geometry label, the ionic radius, and a sliding factor) are provided to evaluate binary diffusion coefficients using the Stokes-Einstein relationship. Again, this is provided simply as an example (cf., Section 5.4). Other more accurate modifications have been developed and can be implemented. Because the solvent self-diffusion coefficient is never explicitly used in the concentrated solution theory outlined in Section 5.3, no species transport information is required. Similarly, because the surface species 'Li(C_s)' and '(C_s)' are considered to be immobile, transport properties are not needed.

7.5. Reaction information

Figure 12 shows the necessary reaction inputs to describe the intercalation process. The mechanism includes two steps: Li adsorption/plating onto the graphite surface, which is a charge-transfer reaction, followed by Li incorporation from the surface into the graphite bulk, which is a thermal reaction (i.e. the surface and bulk species involved are all charge-neutral, so the reaction does not transfer any charge between phases). This is slightly more complex than the one-step mechanism in Reaction 1, but additional chemical complexity would certainly be required to incorporate processes such as Li plating and dendrite growth or SEI formation and evolution.

Each reaction entry requires, at minimum, a reaction equation string and inputs to evaluate reaction rate parameters. For the charge-transfer reaction, the rate parameters are given in a Butler-Volmer formulation, which includes Arrhenius parameters for the exchange current density prefactor k_{ct} and the forward symmetry factor β_f. The three Arrhenius parameters for k_{ct} correspond to $[A_i, n_i, E_i]$ respectively (see Eq. 25). Omitting the backward symmetry factor β_b triggers the default assumption that $\beta_f + \beta_b = 1$. Furthermore, while the foregoing equations (e.g. Sec. 4.1) were written using β_a and β_c and assuming the reaction is written such that the forward reaction is anodic in nature, for generalized software no such assumptions should be made, allowing for greater user flexibility. For the incorporation reaction, Arrhenius parameters are given for the thermally activated rate coefficient $k_{\mathrm{f}}^{\mathrm{t}}$ (Eq. 25).

```
#-------------------------------------------------------------------------------
#  Reaction data
#-------------------------------------------------------------------------------
# Surface reaction 1: charge transfer of Li+(e) to Li(C_s):
surface_reactions("Li+(e) + electron(C_b) + (C_s) <=> Li(C_s)",
   exchangecurrentdensity([4.3611, 0.0, 0.0],beta_f=0.5))

# Surface reaction 2: Li(C_s) incorporation into graphite:
surface_reaction("Li(C_s) + C6(C_b) <=> (C_s) + LiC6(C_b)",
   Arrhenius([2.3e3, 0.0, 0.0]))
```

Figure 12: Example Cantera input ('cti') file describing the necessary reaction information for intercalation at the interface between a liquid carbonate electrolyte and solid graphite anode. The physical parameters should be considered as illustrative and some of the functionality may not yet be implemented in the released version of Cantera.

8. Summary and conclusions

Although numerous important technologies depend on electrochemistry, to date there are no general-purpose modeling tools available to handle complexity in electrochemical transport and kinetics. The present paper seeks to articulate a vision about the scientific needs for modeling electrochemical processes. In fact, the underpinning theories are well known and generally well documented. However, the software implementations have not been fully developed and documented.

The general approach follows the lead of Chemkin and Cantera software. That is, enable the user to define an electrochemical system using a convenient file-format syntax. In this context, "define" means to specify the characteristics of phases, interfaces between phases, species that exist within phases, and reactions that proceed within phases and a phase interfaces. The specifications involve thermodynamic and transport properties as well as rate expressions.

Building on the foundation of the Cantera capabilities, the software can be extended to incorporate increasingly complex aspects of electrochemical kinetics and transport. In fact, Cantera already has some significant electrochemical capabilities. However, because electrochemistry is inherently complicated, compared, for example, to gas-phase chemistry, developing and validating the needed capabilities is a substantial undertaking.

The software discussed in this paper can play a major beneficial role in the development of a wide range of new electrochemical technologies. Advanced predictive modeling capabilities can substantially impact the interpretation of experimental findings and improve scientific understanding. Such capabilities also assist to improve and accelerate the pace of technology development.

Acknowledgments

This research was supported by the Air Force Office of Scientific Research via grant FA9550-16-1-0349 and by the Office of Naval Research via grant N00014-16-1-2780.

References

[1] K. Kang, Y. Meng, J. Bréger, C. Grey, G. Ceder, Electrodes with high power and high capacity for rechargeable lithium batteries, Science 311 (2006) 977–980.

[2] B. Kang, G. Ceder, Battery materials for ultrafast charging and discharging, Nature 458 (2009) 190–193.

[3] D. Aurbach, B. Markovsky, I. Weissman, E. Levi, Y. Ein-Eli, On the correlation between surface chemistry and performance of graphite negative electrodes for Li-ion batteries, Electrochim. Acta 45 (1999) 67–86.

[4] S.-W. Kim, D.-H. Seo, X. Ma, G. Ceder, K. Kang, Electrode materials for rechargeable sodium-ion batteries: Potential alternatives to current lithium-ion batteries, Adv. Energy Mats. 2 (2012) 710–721.

[5] A. Jain, S. Ong, G. Hautier, W. Chen, W. Richards, S. Dacek, S. Cholia, D. Gunter, D. S. G., C. K. Persson, Commentary: The materials project: A materials genome approach to accelerating materials innovation, Apl. Mater. 1.

[6] V. Srinivasan, J. Newman, Discharge modelf for lithium iron-phosphate electrode, J. Electrochem. Soc. 151 (2004) A1517–A1529.

[7] T. Fuller, M. Doyle, J. Newman, Simulation and optimization of the dual lithium insertion cell, J. Electrochem. Soc. 141 (1994) 1–9.

[8] M. Guo, G. Sikha, R. White, Single-particle model for a lithium-ion cell: Thermal behavior, J. Electrochem. Soc. 158 (2011) A122–A132.

[9] A. Colclasure, R. Kee, Thermodynamically consistent modeling of elementary electrochemistry in lithium-ion batteries, Electrochim. Acta 55 (2010) 8960–8973.

[10] J. Christensen, J. Newman, A mathematical model for the lithium-ion negative electrode solid electrolyte interphase, J. Electrochem. Soc. 151 (2004) A1977–A1988.

[11] A. Colclasure, K. Smith, R. Kee, Modeling detailed chemistry and transport for solid-electrolyte-interface (SEI) films in Li-ion batteries, Electrochim. Acta 58 (2011) 33–43.

[12] J. Newman, K. Thomas-Alyea, Electrochemical Systems, 3rd Edition, John Wiley & Sons Inc., Hoboken, New Jersey, 2004.

[13] A. Bard, L. Faulkner, Electrochemical Methods Fundamentals and Applications, Wiley, Hoboken, NJ, 2001.

[14] J. Bockris, A. Reddy, M. Gamboa-Aldeco, Modern electrochemistry: Fundamentals of electrodics, 2nd Edition, Kluver Academic/Plenum Publishers, New York, 2000.

[15] T. Fuller, J. Harb, Electrochemical engineering, Wiley, Hoboken, NJ, 2018.

[16] M. Doyle, T. Fuller, J. Newman, Modeling of galvanostatic charge and discharge of the lithium/polymer/insertion cell, J. Electrochem. Soc. 140 (1993) 1526–1533.

[17] M. Doyle, T. Fuller, J. Newman, Relaxation phenomena in lithium-ion-insertion cells, J. Electrochem. Soc. 141 (1994) 982–990.

[18] F. Hall, S. Wußler, H. Buqa, W. Bessler, Asymmetry of discharge/charge curves of lithium-ion battery intercalation electrodes, J. Phys. Chem. C 120 (41) (2016) 23407–23414.

[19] A. Wiedemann, G. Goldin, S. Barnett, H. Zhu, R. Kee, Effects of three-dimensional cathode microstructure on the performance of lithium-ion battery cathodes, Electrochim. Acta 88 (2013) 1580–1588.

[20] M. Farkhondeh, M. Safari, M. Pritzker, M. Fowler, T. Han, J. Wang, C. Delacourt, Full-range simulation of a commercial $LiFePO_4$ electrode accounting for bulk and surface effects: A comparative analysis, J. Electrochem. Soc. 161 (2014) A201–A212.

[21] M. Farkhondeh, M. Pritzker, M. Fowler, C. Delacourt, Mesoscopic modeling of a $LiFePO_4$ electrode: Experimental validation under continuous and intermittent operating conditions, J. Electrochem. Soc. 164 (2017) E3040–E3053.

[22] B. Orvananos, T. Ferguson, H.-C. Yu, M. Bazant, K. Thornton, Particle-level modeling of the charge-discharge behavior of nanoparticulate phase-spearating Li-ion battery electrodes, J. Electrochem. Soc. 161 (2014) A535–A546.

[23] W. Dreyer, J. Jamnik, C. Guhlke, R. Huth, J. Moškon, M. Gaberšček, The thermodynamic origin of hysteresis in insertion batteries, Nat. Mater. 9 (2010) 448–453.

[24] R. Malik, A. Abdellahi, G. Ceder, A critical review of the Li insertion mechanisms in $LiFePO_4$ electrodes, J. Electrochem. Soc. 160 (2013) A3179–A3197.

[25] H. Zhu, R. Kee, Computational modeling of sodium-iodine secondary batteries, Electrochim. Acta 219 (2016) 70–81.

[26] M. Armand, J.-M. Tarascon, Building better batteries, Nature 451 (2008) 652–657.

[27] D. Grübl, B. Bergner, D. Schröder, J. Janek, W. Bessler, Multistep reaction mechanisms in nonaqueous lithium–oxygen batteries with redox mediator: A model-based study, J. Phys. Chem. C 120 (43) (2016) 24623–24636.

[28] J. Neidhardt, D. Fronczek, T. Jahnke, T. Danner, B. Horstmann, W. Bessler, A flexible framework for modeling multiple solid, liquid and gaseous phases in batteries and fuel cells, J. Electrochem. Soc. 159 (2012) A1528–A1542.

[29] B. Horstmann, T. Danner, W. Bessler, Precipitation in aqueous lithium–oxygen batteries: A model-based analysis, Energy Environ. Sci. 6 (4) (2013) 1299.

[30] Y. Yin, C. Gaya, A. Torayev, V. Thangavel, A. Franco, Impact of Li_2O_2 particle size on Li-O_2 battery charge process: Insights from a multiscale modeling perspective, J. Phys. Chem. Lett. 7 (19) (2016) 3897–3902.

[31] A. Hofmann, D. Fronczek, W. Bessler, Mechanistic modeling of polysulfide shuttle and capacity loss in lithium–sulfur batteries, J. Power Sources 259 (2014) 300–310.

[32] H. Zhu, R. Kee, Modeling protonic-ceramic fuel cells with porous composite electrodes in a button-cell configuration, J. Electrochem. Soc. 164 (2017) F1400–F1411.

[33] H. Zhu, R. Kee, V. Janardhanan, O. Deutschmann, D. Goodwin, Modeling elementary heterogeneous chemistry and electrochemistry in solid-oxide fuel cells, J. Electrochem. Soc. 152 (2005) A2427–A2440.

[34] D. Goodwin, H. Zhu, A. Colclasure, R. Kee, Modeling electrochemical oxidation of hydrogen on Ni-YSZ pattern anodes, J. Electrochem. Soc. 156 (2009) B1004–B1021.

[35] C. Duan, J. Tong, M. Shang, S. Nikodemski, M. Sanders, S. Ricote, A. Almansoori, R. O'Hayre, Readily processed protonic ceramic fuel cells with high performance at low temperature, Science 349 (2015) 1321–1326.

[36] H. Zhu, S. Ricote, W. Coors, R. Kee, Interpreting equilibrium-conductivity and conductivity-relaxation measurements to establish thermodynamic and transport properties for multiple charged defect conducting ceramics, Farad. Discuss. 182 (2015) 49–74.

[37] M. Shang, J. Tong, R. O'Hayre, A promising cathode

for intermediate temperature protonic ceramic fuel cells: $BaCo_{0.4}Fe_{0.4}Zr_{0.2}O_{3-\delta}$, RSC Adv. 3 (36) (2013) 15769–15775.
[38] C. Duan, J. Tong, M. Shang, S. Nikodemski, M. Sanders, S. Ricote, A. Almansoori, R. O'Hayre, Readily processed protonic ceramic fuel cells with high performance at low temperatures, Science 349 (6254) (2015) 1321–1326.
[39] C. Duan, D. Hook, Y. Chen, J. Tong, R. O'Hayre, Zr and Y co-doped perovskite as a stable, high performance cathode for solid oxide fuel cells operating below 500°C, Energy Environ. Sci. 10 (1) (2017) 176–182.
[40] C. Duan, R. Kee, H. Zhu, C. Karakaya, Y. Chen, S. Ricote, A. Jarry, E. Crumlin, D. Hook, R. Braun, N. Sullivan, R. O'Hayre, Highly durable, coking and sulfur tolerant, fuel-flexible protonic ceramic fuel cells, Nature 557 (7704) (2018) 217–222.
[41] H. Malerød-Fjeld, D. Clark, I. Yuste-Tirados, R. Zanón, D. Catalán-Martinez, D. Beeaff, S. Morejudo, P. Vestre, T. Norby, R. Haugsrud, J. Serra, C. Kjølseth, Thermo-electrochemical production of compressed hydrogen from methane with near-zero energy loss, Nat. Energy 2 (2017) 923–931.
[42] C. Karakaya, R. Kee, Progress in the direct catalytic conversion of methane to fuels and chemicals, Prog. Energy Combust. Sci. 55 (2016) 60–97.
[43] R. Kee, C. Karakaya, H. Zhu, Process intensification in the catalytic conversion of natural gas to fuels and chemicals, Proc. Combust. Inst. 36 (2017) 51–76.
[44] S. Morejudo, R. Zanón, S. Escolástico, I. Yuste-Tirados, H. Malerød-Fjeld, P. Vestre, W. Coors, A. Martínez, T. Norby, J. Serra, C. Kjølseth, Direct conversion of methane to aromatics in a catalytic co-ionic membrane reactor, Science 353 (2016) 563–566.
[45] C. Karakaya, S. Hernández-Morejudo, H. Zhu, R. Kee, Catalytic chemistry for methane dehydroaromatization (MDA) on a bifunctional Mo/HZSM-5 catalyst in a packed bed, Ind. Eng. Chem. Res. 55 (2016) 9895–9906.
[46] J. O'Connell, J. M. Haile, Thermodynamics: Fundamentals for applications, Cambridge University Press, Cambridge, NY, 2005.
[47] S. Bishop, D. Marrocchelli, C. Chatzichristodoulou, N. Perry, M. Mogensen, H. Tuller, E. Wachsman, Chemical expansion: Implications for electrochemical energy storage and conversion devices, Annu. Rev. Mater. Res. 44 (2014) 205–239.
[48] T. Ferguson, M. Bazant, Nonequilibrium thermodynamics of porous electrodes, J. Electrochem. Soc. 159 (2012) A1967–A1985.
[49] V. Malavé, J. R. Berger, P. A. Martin, Concentration-dependent chemical expansion in lithium-ion battery cathode particles, J. Appl. Mech. 81 (2014) 091005.
[50] B. Euser, J. Berger, H. Zhu, R. Kee, Defect-transport-induced stress in mixed ionic-electronic conducting (MIEC) ceramic membranes, J. Electrochem. Soc. 163 (2016) F264–F271.
[51] D. Marrocchelli, N. Perry, S. Bishop, Understanding chemical expansion in perovskite-structured oxides, Phys. Chem. Chem. Phys. 17 (2015) 10028–10039.
[52] S. Adler, Chemical expansivity of electrochemical ceramics, J. Am. Ceram. Soc. 84 (2001) 2117–2119.
[53] J. Li, Physical chemistry of some microstructural phenomena, Metall. Trans. A 9A (1978) 1350–1380.
[54] F. Yang, Interaction between diffusion and chemical stresses, Mater. Sci. Eng. A. 409 (2005) 153–159.
[55] H. Zhu, R. Kee, Modeling distributed charge-transfer processes in SOFC membrane electrode assemblies, J. Electrochem. Soc. 155 (2008) B715–B729.
[56] W. Bessler, J. Warnatz, D. Goodwin, The influence of equilibrium potential on the hydrogen oxidation kinetics of SOFC anodes, Solid State Ion. 177 (2007) 3371–3383.
[57] R. Kee, M. Coltrin, P. Glarborg, H. Zhu, Chemically reacting flow: Theory, modeling and simulation, 2nd Edition, Wiley, Hoboken, NJ, 2018.
[58] M. Bazant, Theory of chemical kinetics and charge transfer based on nonequalibrium thermodynamics, Acc. Chem. Res. 46 (2012) 1144–1160.
[59] C. Kupper, W. Bessler, Multi-scale thermo-electrochemical modeling of performance and aging of a $LiFePO_4$/graphite lithium-ion cell, J. Electrochem. Soc. 164 (2) (2017) A304–A320.
[60] J. Owejan, J. Owejan, S. DeCaluwe, J. Dura, Solid electrolyte interphase in li-ion batteries: Evolving structures measured in situ by neutron reflectometry, Chem. Mater. 24 (11) (2012) 2133–2140.
[61] A. Cresce, S. Russell, D. Baker, K. Gaskell, K. Xu, In situ and quantitative characterization of solid electrolyte interphases, Nano Lett. 14 (3) (2014) 1405–1412.
[62] H. Zhu, R. Kee, Membrane polarization in mixed-conducting ceramic fuel cells and electrolyzers, Int. J. Hydrogen Energy 41 (2016) 2931–2943.
[63] M. Safari, M. Morcrette, A. Teyssot, C. Delacourt, Multimodal physics-based aging model for life prediction of Li-ion batteries, J. Electrochem. Soc. 156 (3) (2009) A145–A153.
[64] S. Vilekar, R. Datta, The effect of hydrogen crossover on open-circuit voltage in polymer electrolyte membrane fuel cells, J. Power Sources 195 (8) (2010) 2241–2247.
[65] M. Bazant, K. Thornton, A. Ajdari, Diffuse-charge dynamics in electrochemical systems, Phys. Rev. E 70 (2004) 021506.
[66] E. Vøllestad, H. Zhu, R. Kee, Interpretation of defect and gas-phase fluxes through mixed-conducting ceramics using Nernst–Planck–Poisson and integral formulations, J. Electrochem. Soc. 161 (2014) F114–F124.
[67] S. Ricote, H. Zhu, W. Coors, C. Chatzichristodoulou, R. Kee, Equilibrium and transient conductivity for gadolium-doped ceria under large perturbations: I. Experiments, Solid State Ion. 265 (2014) 22–28.
[68] H. Zhu, S. Ricote, W. Coors, C. Chatzichristodoulou, R. Kee, Equilibrium and transient conductivity for gadolium-doped ceria under large perturbations: II. Modeling, Solid State Ion. 268 (2014) 198–207.
[69] R. Bird, W. Steward, E. Lightfoot, Transport phenomena, John Wiley & Sons, 1960.
[70] L. Onsager, Theories and problems of liquid diffusion, Ann. N.Y. Acad. Sci. 46 (1945) 241–265.
[71] R. Pollard, J. Newman, Transport equations for a mixture of two binary molten salts in a porous electrode, J. Electrochem. Soc. 126 (1979) 1713–1717.
[72] J. Newman, D. Bennion, C. Tobias, Mass transfer in concentrated binary electrolytes, Ber. Bunsenges. Phys. Chem. 69 (1965) 608–612.
[73] H. Tyrrell, K. Harris, Diffusion in liquids: A theoretical and experimental study, Butterworth & Co, Oxford, UK, 1984.
[74] L. Onsager, R. Fuoss, Irreversible processes in electrolytes. Diffusion, conductance, and viscous flow in arbitrary mixtures of strong electrolytes, J. Phys. Chem. 36 (1932) 2689–2778.
[75] L. Belfiore, Transport phenomenal for chemical reactor design, John Wiley & Sons, Hoboken, New Jersey, 2003.
[76] J. Happel, H. Brenner, Low Reynolds number hydrodynamics: with special application to particulate media, Prentice-Hall, Upper Saddle River, NJ, 1965.
[77] B. Poling, J. Prausnitz, J. O'Connell, The properties of gases and liquids, McGraw-Hill, 1977.
[78] J.-F. Dufrêche, O. Bernard, S. Durand-Vidal, P. Turq, Analyti-

cal theories of transport in concentrated solutions from MSA, J. Phys. Chem. B 109 (2005) 9873–9884.
[79] N. Shaffer, S. Baalrud, J. Daligault, Effective potential theory for diffusion in binary ionic mixtures, Phys. Rev. E 95 (2017) 013206.
[80] T. Thacher, J.-L. Lin, C. Mou, Theory of Onsager phenomenological coefficients for isothermal linear transport processes in electrolyte solutions, J. Chem. Phys. 81 (1984) 2053–2063.
[81] E. Lai, F. Ciucci, Mathematical modeling of porous batter electrodes – Revisit of Newman's model, Electrochim. Acta 56 (2011) 4369–4377.
[82] S. Hein, A. Latz, Influence of local lithium metal deposition in 3D microstructures on local and global behavior of Lithium-ion batteries, Electrochim. Acta 201 (2016) 354–365.
[83] H.-I. Yoo, M. Martin, J. Janek, Comment on "How to interpret Onsager cross terms in mixed ionic electronic conductors" by I. riess, Phys. Chem. Chem. Phys. 17 (2015) 11103–111106.
[84] D.-K. Lee, H.-I. Yoo, Electron-ion interface and Onsager reciprocity in mixed ionic-electronic transport in TiO_2, Phys. Rev. Lett. 97 (2006) 255901.
[85] C. Chatzichristodoulou, W.-S. Park, H.-S. Kim, P. Hendriksen, H.-I. Yoo, Experimental determination of the Onsager coefficients of transport for $Ce_{0.8}Pr_{0.2}O_{2-\delta}$, Phys. Chem. Chem. Phys. 12 (2010) 9637–9649.
[86] A. Zuev, V. Sereda, D. Tsvetkov, Defect structure and defect-induced expansion of doped perovskite $La_{0.7}Sr_{0.3}Co_{0.9}Fe_{0.1}O_{3-\delta}$, Int. J. Hydrog. Energy 39 (2014) 21553–21560.
[87] D. G. Goodwin, H. K. Moffat, R. L. Speth, Cantera: An object-oriented software toolkit for chemical kinetics, thermodynamics, and transport processes, `http://www.cantera.org`, version 2.3.0 (2017).
[88] G. Kogekar, C. Karakaya, G. Liskovich, M. Oehlschlaeger, S. DeCaluwe, R. Kee, Impact of non-ideal behavior on ignition delay and chemical kinetics in high-pressure shock tube reactors, Comb. Flame 189 (2018) 1–11.

Nomenclature

Variable	Description
a_k	Activity concentrations of species k
A_i	Pre-exponential factor of reaction i
A_ℓ	Prefactor for excess gibbs energy
A_{ik}, B_{ik}	Matrix inputs for Redlich–Kister expression
c_p	Specific heat
C	Concentration of binary salt
D	Diffusion coeff. in binary concentrated solutions
D_k	Diffusion coeff. of species k
$D_{k,\mathrm{eff}}$	Effective diffusion coeff. of species k
$\mathcal{D}$	Ambipolar diffusion coeff.
$\mathcal{D}_{kj}$	Binary diffusion coeff.
E	Young's modulus
E_a	Electric-potential difference between an and el
E_a^eq	Equilibrium potential difference between an and el
E_c	Electric-potential difference between ca and el
E_c^eq	Equilibrium potential difference between ca and el
E^eq	Equilibrium potential difference between ed and el
E_i^eq	Equilibrium potential difference between ed and el for reaction i
E_i	Activation energy of reaction i
E^OCV	Open-circuit potential
E_i^REV	Reversible potential of reaction i
F	Faraday constant
g^E	Excess Gibbs free energy due to mixing
g_{ik}	Gibbs energy for mixture of species i and k
G	Gibbs free energy
ΔG_i°	Change in standard-state Gibbs free energy
ΔH_i	Change in enthalpy for reaction i
$\mathbf{i}$	Current density
i_0	Exchange current density
i_0^*	Exchange current density prefactor
i_e	Butler–Volmer current density
$\mathbf{J}_k$	Flux of species k
k_B	Boltzmann constant
k_bi	Backward rate constant for reaction i
k_bi^t	Thermal backward rate constant for reaction i
k_ct	Charge transfer rate coefficient
$k_{\mathrm{f}i}$	Forward rate constant for reaction i
$k_{\mathrm{f}i}^\mathrm{t}$	Thermal forward rate constant for reaction i
K_i	Reaction equilibrium constant for reaction i
K_{kj}	Friction coeff. matrix
L_{kj}	Onsager coeff. matrix
L_{kj}^0	Negative inverted matrix of M_{kj}^0
m	Molality and phase index
M_{kj}	Modified friction coefficient matrix
M_{kj}^0	Modified M_{kj} matrix
$n_\mathrm{e,i}$	Electrons transfered in reaction i
n_i	Temperature exponent
n_k	Moles of species k
n_T	Total moles in solution
N_species	Number of species in solution
p	Pressure
p_i^*	Inverse equilibrium constants for reaction i
p_k	Partial pressure of species k
$q_{\mathrm{b}i}$	Backward rate of progress for reaction i
$q_{\mathrm{f}i}$	Forward rate of progress for reaction i
$\dot{q}_{\mathrm{F},i}$	Faradaic heat generation rate for reaction i
q_i	Rate of progress for reaction i
$\dot{q}_\mathrm{ohm}$	Ohmic heat generation rate
r_k	Radius of species k
R	Universal gas constant
S	Entropy
t	Time
t_k	Transference number of species k
t_k^0	Transference number of species k based on species 0
T	Temperature
u_k	Mobility of species k
$\mathbf{v}$	Bulk velocity
$\mathbf{v}_k$	Diffusive velocity of species k
W_k	Molecular weight of species k
x, y	Intercalation fraction
X_k	Mole fraction of species k
$[X_k]$	Concentration of species k
$\left[X_k^\circ\right]$	Standard state concentration of species k
Y_k	Mass fraction of species k
z_k	Charge of species k
α_a	Anodic transfer coeff.
α_c	Cathodic transfer coeff.
β	Symmetry factor
β_a	Anodic symmetry factor
$\beta_{\mathrm{a}i}$	Anodic symmetry factor for reaction i
β_c	Cathodic symmetry factor
$\beta_{\mathrm{c}i}$	Backward symmetry factor for reaction i
β_f	Forward symmetry factor
ϵ	Phase dielectric constant
ϵ_m	Volume fraction for phase m
ζ	Friction coeff.
γ_k	Activity coeff. of species k
$\gamma_\pm$	Binary concentrated solution mean molal activity coeff.
Γ_m	Total available surface site density of phase m
η_act	Activation overpotential
η_j	Viscosity of solvent j
$\theta_{k,m}$	Site fraction of species k on surface of phase m
μ_k	Chemical potential of species k
μ_k°	Standard-state chemical potential of species k
$\tilde{\mu}_k$	Electrochemical potential of species k
ν	Poisson's ratio
ν_+	Cation stoichiometric coefficient in binary mixture
ν_-	Anion stoichiometric coefficient in binary mixture
ν'_{ki}	Forward stoichiometric coeff. for kth species in ith reaction
ν''_{ki}	Reverse stoichiometric coeff. for kth species in ith reaction
ν_{ki}	Net stoichiometric coeff. for kth species in ith reaction
ρ	Local charge density
ρ_m	Density of phase m
σ	Ionic/electric conductivity

σ_{h}	Hydrostatic stress
σ_{eff}	Effective conductivity
υ	Sliding coeff.
Φ_k	Electrostatic potential of species k
Φ_m	Electrostatic potential of phase m
χ_k	Chemical symbol for the species k
$\dot{\omega}_k$	Volumetric production rate of species k
Ω_k	Partial molar volume of species k

Next-generation Chemical-kinetics, Transport, and Reacting-flow Software Tools: Non-ideal gases, heterogeneous catalysis, electrochemistry

Robert J. Kee[a], Steven C. DeCaluwe[a], C. Franklin Goldsmith[b], Gregory S. Jackson[a], Kyle E. Niemeyer[c], Raymond L. Speth[d], Bryan W. Weber[e], Richard H. West[f]

[a]*Department of Mechanical Engineering, Colorado School of Mines, Golden, CO 80401, United States*
[b]*Department of Chemical Engineering, Brown University, Providence, RI 02912, United States*
[c]*Department of Mechanical Engineering, Oregon State University, Corvallis, OR 97331, United States*
[d]*Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Cambridge, MA 02139, United States*
[e]*Department of Mechanical Engineering, University of Connecticut, Storrs, CT 06269, United States*
[f]*Department of Chemical Engineering, Northeastern University, Boston, MA 02115, United States*

Abstract

With programmatic leadership from the Colorado School of Mines, a nationwide team of experts in the chemically reacting flow modeling proposes to develop and support next-generation modeling tools. The technical scope considers significant advances in three areas—non-ideal gas phase kinetics, heterogeneous catalysis, and electrochemistry. The software structure follows notionally in the footsteps of Chemkin, but the proposed effort builds on extensions to Cantera. The resulting software will be developed and maintained as open-source, enabling unfettered access and collaborative interactions between researchers in academia, National Laboratories, and industry. Successfully accomplishing the proposed research and development requires a substantial, sustained, and coordinated effort among subject-matter experts, including chemistry and transport theory as well as software development. The proposed effort is structures in the spirit of a MURI. That is, the multidisciplinary team is drawn from several universities where the team members are recognized as experts and leaders. Although detailed budgets would need to be negotiated, the nominal plan calls for roughly $1M annual over a five-year project period.

Contents

Introduction

Modeling and simulation play critical roles in the research, development, and deployment of a diverse array of chemical processes. Energy-conversion processes in the context of primary propulsion (e.g., combustion) or auxiliary power (e.g., batteries or fuel cells) are topics of particular interest to the Air Force. Numerical models can be an efficient and effective method to identify limitations, optimize device designs and control strategies, and explore the impact of new materials. As such, robust, generalized, and widely-available chemical simulation tools are essential to accelerate the pace of scientific discovery and associated technology development. For example, the widespread availability and use of Chemkin for over 30 years has made major contributions to enable the modeling of complex combustion processes and coupling into reactive flow simulations. However, widely available, generalized chemical kinetic software tools have not, in recent years, kept pace with the increasing chemical complexity and interdisciplinarity of advanced technology solutions. The present project builds on the foundation of the successes brought about by Chemkin's widespread promulgation, and will establish next generation modeling tools with major advances in non-ideal chemical kinetics, heterogeneous catalysis, and electrochemistry.

The proposed software development effort will substantially advance the state of the art in modeling chemical kinetics and transport processes relevant for

- homogeneous reacting flows of real gases and fluids,
- heterogeneous reactions for catalysis, bulk deposition, and solid decomposition,
- electrochemical processes in energy storage, energy conversion, and corrosion.

The new theoretical formulations derived from this program will be implemented in an open-source software platform based on the existing Cantera framework. The open-source platform will accelerate the implementation of these new capabilities in non-ideal homogeneous chemistry, heterogeneous chemistry, and electrochemistry into a wide-array of computational tools to facilitate broad fundamental scientific impact and practical technology design modeling and analysis.

Recognizing the diverse scope and the magnitude of the undertaking, we propose to form a multidisciplinary, nationwide team of experts. The proposed approach follows the spirit of Multidisciplinary University Research Initiatives (MURIs). The technical approach is based on the planning study and roadmap (Appendix A) that was led by Colorado School of Mines (Mines) and supported by the Air Force Office of Scientific Research (AFOSR). The proposed software development will build upon the current capabilities of Cantera. Cantera represents a logical extension of the Chemkin capabilities. Cantera is written in C++, providing a modern, object-oriented, extensible, programming environment in which to develop flexible modeling structures for handling complex multi-species and/or multi-phase thermodynamics, chemical kinetics, and transport. Moreover, as discussed in Appendix A, Cantera already has a vibrant and growing developer and user community representing academia, government, and industry.

As articulated in the Roadmap planning, the proposed approach will not only address the core scientific modeling capabilities for real fluids, heterogeneous interfaces, and electrochemical processes, but also establish critical abilities for software management and integration with other modeling tools and frameworks to facilitate broad deployment for the relevant research and development communities. The Roadmap identified four areas in software management and integration, notably usability, extensibility, compatibility, and maintainability, that are critical to ensure that the developed scientific capabilities are implemented broadly and supported sustainably for

the relevant research and development communities to benefit for decades to come.

Proposing Team

Although there are great benefits to working as a multidisciplinary cross-national team, effectively coordinating and managing such a team can be challenging. Prof. Robert Kee is proposed to lead the team. Prof. Kee was the principal architect of the CHEMKIN software and he has successfully led an ONR MURI team involving several universities and an ONR Research Tools Consortium (RTI) involving two universities and an industry partner. In broad terms, the proposed research will be coordinated by Team leaders who are subject-matter experts in six functional areas. Of the six Teams, three are concerned primarily with new chemistry and transport capabilities. One Team is concerned primarily with numerical mathematics and advanced computing architectures (e.g., parallel computing, graphical processing units, etc.). The remaining two Teams are concerned primarily with software integration, interoperability, and quality (i.e., the user experience).

In addition to Prof. Kee, the team is composed of seven members at six universities. The team has a wealth of experience in a diverse array of content areas and in developing open source, generalized chemical kinetic software, either within CANTERA or in other applications. The team members, in alphabetic order, are:

- Steven C. DeCaluwe, Colorado School of Mines. Expertise in electrochemical and combustion simulations, with a focus on mechanism development for electrochemistry and and the influence of non-ideal equations of state. Contributing developer to CANTERA.
- C. Franklin Goldsmith, Brown University. Expertise in simulation and mechanism generation for combustion and catalytic processes. Non-ideal effects on chemical kinetics and electronic structure methods. Contributing developer of RMG and primary developer of RMG-CAT.
- Gregory S. Jackson, Colorado School of Mines. Expertise in thermochemical and electrochemical energy storage, catalytic chemical processes.
- Kyle E. Niemeyer, Oregon State University. Expertise in advanced numerical methods for modeling of combustion and reactive flows, including mechanism reduction and algorithms for GPUs. Contributing developer at CANTERA, and primary developer of PYJAC.
- Raymond L. Speth, Massachusetts Institute of Technology. Expertise in combustion modeling with a focus on flame simulations, environmental impact, and multi-phase combustion (soot and droplet dynamics). Primary developer at CANTERA.
- Bryan W. Weber, University of Connecticut. Expertise in combustion simulations and open-source software tools for combustion research and thermodynamics. Primary developer at CANTERA.
- Richard H. West, Northeastern University. Expertise in development of detailed microkinetic models for complex reacting systems. Contributing developer of RMG and primary developer of RMG-CAT.

Together with Prof. Kee serving as the project leader, the Team leaders will form a de-facto executive council. Each Team will be composed of individuals who are experts both in the relevant chemistry and in software development. Team leaders for each topic area are specified in bold text. Of course, numerous students and postdoctoral fellows will also be actively engaged. Team activities will be coordinated using both real-time virtual communications (via SLACK and web-conferencing), plus annual in-person meetings. The following sections of this report provide summaries of the Team scopes and objectives.

Draft budget

The requested budget is $915,000 per year, as outlined in Table 1, which primarily funds salary and fringe to staff the project, including graduate students, postdoctoral associates, plus limited funds for travel (outreach and in-person team meetings) and computational resources to demonstrate new capabilities to leverage high-performance computing.

New Physical Modeling Capabilities

1. Homogeneous phase thermochemistry

Team: **Goldsmith**, Kee, DeCaluwe, West

1.1. Overview and objectives

Homogeneous thermochemistry and transport processes play central roles in technology with wide-ranging

Table 1: Proposed yearly project budget

Colorado School of Mines	
Salaries, Wages, and Fringe:	$246,000
(1 month Summer Salary per PI, grad student and post-doc staffing.)	
Other direct costs – travel and computational resources.	$12,000
Indirect Costs.	$94,000
Mines total:	**$352,000**
Massachusetts Institute of Technology	
Salaries, Wages, and Fringe:	$101,000
(PI Salary, at 25% FTE, grad student staffing.)	
Other direct costs – travel and computational resources.	$7,000
Indirect Costs.	$47,000
MIT total:	**$155,000**
Oregon State University	
Salaries, Wages, and Fringe:	$78,500
(1 month Summer Salary, post-doc staffing.)	
Other direct costs – travel and computational resources.	$2,500
Indirect Costs.	$38,500
OSU total:	**$119,500**
Brown University	
Salaries, Wages, and Fringe:	$71,000
(1 month Summer Salary, grad student staffing.)	
Other direct costs – travel and computational resources.	$5,000
Indirect Costs.	$38,000
Brown total:	**$114,000**
Northeastern University	
Salaries, Wages, and Fringe:	$59,000
(1 month Summer Salary, grad student staffing.)	
Other direct costs – travel and computational resources.	$4,000
Indirect Costs.	$36,000
NEU total:	**$99,000**
University of Connecticut	
Salaries, Wages, and Fringe:	$40,000
(3 month Summer Salary, student staffing.)	
Other direct costs – travel and computational resources.	$3,000
Indirect Costs.	$26,000
UConn total:	**$69,000**
Other direct costs: computational resources.	**$20,000**
Total:	**$928,500**

societal impacts, ranging from combustion to chemical processing to atmospheric chemistry. While robust and mature chemical kinetic modeling tools (e.g., CHEMKIN) have been available for decades, these tools are based almost exclusively on ideal gas applications. However, with the increasing relevance of high operating pressures in combustion and condensed liquid- and solid-phases in electrochemistry, it is crucial that chemical kinetic modeling frameworks enable and support modeling of non-ideal effects. This support includes implementing non-ideal (aka 'real gas') equations of state and posing thermodynamic, kinetic, and transport calculations in a sufficiently generalized manner such that non-ideal effects can be represented robustly and flexibly. State-of-the-art theory in non-ideal thermodynamic, kinetic, and transport phenomena most certainly should be implemented, but beyond this non-ideal capabilities should be developed in such a way that can easily be extended and modified as new theories develop.

Thermodynamically-consistent treatment of non-ideal effects must be centered on the equation of state (EoS). Consider, for example the Peng-Robinson EoS [1], which can be represented as:

$$p = \frac{RT}{v - b^*} - \frac{a^*\alpha}{v^2 + 2b^*v - b^{*2}}. \tag{1}$$

Here, p, T, and v are the phase pressure, temperature, and average molar volume, respectively, R is the universal gas constant. a^* and b^* are composition-dependent interaction parameters, and α is a temperature- and composition-dependent interaction parameter. The cubic EoS can be solved to determine the phase p–v–T behavior. Additionally, integration of the EoS provides an entry point to determining thermodynamic parameters, which are required for accurate chemical kinetic and transport rate calculations.

In particular, integrating the pressure at a fixed temperature and composition, from some reference state volume v° to the present molar volume v provides the departure function for the Helmholtz energy a, relative to the reference state:

$$a - a^\circ = -\int_{v^\circ}^{v} p\, dv. \quad (2)$$

Substituting Eq. (1) into Eq. (2) and integrating provides an expression for the Helmholtz energy, relative to some known reference state, which in turn provides a means to calculate other required thermodynamic properties such as enthalpies, chemical potentials, and activity coefficients.

Once thermodynamic properties have been defined according to the EoS, the net rate of progress for a reaction, q_i, can be written as the difference between forward and backward rates of progress:

$$q_i = k_{\mathrm{f}i} \prod_{k=1}^{K} (\gamma_k [X_k])^{\nu'_{ki}} - k_{\mathrm{r}i} \prod_{k=1}^{K} (\gamma_k [X_k])^{\nu''_{ki}}, \quad (3)$$

where $k_{\mathrm{f}i}$ and $k_{\mathrm{b}i}$ are the forward and reverse rate constants, $[X_k]$ and γ_k are the molar concentrations (mol m^{-3}) and activity coefficients of the participating species, respectively, and ν'_{ki} and ν''_{ki} are the forward and reverse stoichiometric coefficients. Under ideal-gas conditions, the activity coefficients are assumed equal to unity. Under non-ideal conditions, however, the activity coefficients are calculated directly from the EoS:

$$RT \ln\left(\frac{\gamma_k [X_k]}{\left[X_k^\circ\right]}\right) = \left(\frac{\partial \left(n_T \left(a - a^\circ\right)\right)}{\partial n_k}\right)_{T,V,n_{j\neq k}}, \quad (4)$$

where $\left[X_k^\circ\right]$ is the molar concentration at the reference state, and where n_T and n_k are the total number of moles and the moles of species k, respectively. The Helmholtz departure function comes from Eq. (2), directly linking the non-ideal EoS to the resulting chemical kinetics.

As a general rule, support for a greater range of EoS is required in chemical kinetic modeling frameworks. Cantera currently supports a multi-component Redlich-Kwong EoS, and additional EoS implementations (such as multi-component Peng–Robinson EoS and greater support for liquids and solids) will be developed and implemented as part of the proposed work. Additionally, in this section we highlight four broad areas where next-generation software is essential to enable quantitative predictions under extreme chemical environments:

- Represent real-gas effects on rate coefficients,
- Quantify reactions involving non-thermalized species,
- Represent collider effects in a physically meaningful manner,
- Provide increased flexibility for representing intermolecular potentials and their effects on transport properties.

1.2. High-pressure effects on rate constants

Although considerable advances have been made regarding real-gas equations of state for thermodynamic properties, the same cannot be said of chemical kinetics under extreme pressures. The current state-of-the-art in computational kinetics for homogeneous chemistry implicitly assumes that the transition state behaves ideally, even when the rest of the fluid is assumed to behave non-ideally. Real-gas behavior can have a profound effect on the chemical source terms in reactive flow simulations, and general purpose software must be able to account for these effects in a thermodynamically consistent manner.

The rate constants in Eq. (3) should be consistent with the fluid equation of state (i.e., the kinetic and thermodynamic equilibrium states are equal). In virtually all models to date, however, the rate constant actually used is that of an ideal gas: $k_{\mathrm{f}i} \approx k^0_{\mathrm{f}i}$. The rate constant under real-gas conditions can be related to the ideal-gas rate constant by including the effect of the non-idealities consistently for both the reactant and the transition state: $k_{\mathrm{f}i} = k^0_{\mathrm{f}i}\gamma_k/\gamma^*$, where γ^* is the effective activity coefficient of the transition state complex. As reviewed recently by Kogekar et al. [2], one can rewrite the rate of progress for the i^{th} reaction as:

$$q_i = \frac{1}{\gamma^*}\left[k^0_{\mathrm{f}i} \prod_{k=1}^{K} (\gamma_k [X_k])^{\nu'_{ki}} - k^0_{\mathrm{r}i} \prod_{k=1}^{K} (\gamma_k [X_k])^{\nu''_{ki}}\right], \quad (5)$$

For some reactions, the transition state activity coefficient may be close to that of the reactant, $\gamma_k/\gamma^* \approx 1$, and thus real-gas effects may cancel. For other reactions (particularly reactions for which the transition state is structurally dissimilar from the reactant), $\gamma^* \neq \gamma_k$, and the effects of real-gas behavior on kinetics become essential. Contemporary reactive flow simulations *may* include the γ_k's in the rate of progress, but they never include γ^*, in part because there is no systematic method for representing the activity coefficient for transition states.

Proposed Technical Solution: We propose to include ability to provide transition state activity coefficients for elementary reactions. The syntax likely will be modeled

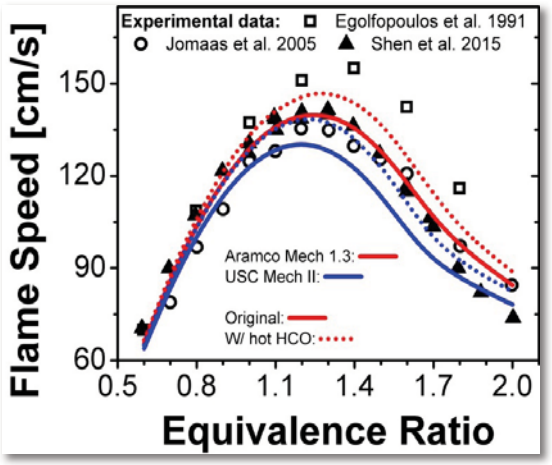

Figure 1: Effect of prompt dissociation of hot HCO on laminar flame speeds of acetylene in air at standard conditions.

on the analogous syntax for chemical species for the same fluid.

1.3. Non-Boltzmann effects

An implicit assumption in all reactive flow simulations is that the chemical species are thermalized, *i.e.,* that the distribution of internal states follows a Boltzmann distribution, which is determined by the local bath temperature. In other words, conventional phenomenological rate theory assumes that the products in an exothermic reaction may be formed in a ro-vibrationally hot state, but they will be completely thermalized via a sequence of non-reactive, inelastic collisions before successive chemical reactions can occur.

However, a growing body of evidence suggests that many reactions violate this paradigm, and that elementary reactions involving non-thermalized species can have a significant impact on the observable kinetic properties of a system. These "non-Boltzmann" reactions exhibit markedly different behavior, including the formation of entirely different products. Furthermore, the elementary rate coefficients that result from non-Boltzmann reactions exhibit a complex functional dependence on temperature, pressure, *and* composition.

Together, this combination of multiple product channels and concentration-dependent rate coefficients constitute a new class of reactions that cannot be modeled using current software. In some instances, the ro-vibrationally hot products will isomerize and/or decompose prior to thermalization. In other instances, ro-vibrationally hot products can react other species to unlock new reactions. These two types of effects are described separately below.

1.3.1. Prompt dissociation

Figure 1 presents recent work on the effect of prompt dissociation of a single radical, HCO. At high temperatures, a fraction of the nascent, vibrationally hot HCO^* will decompose to H + CO prior to thermalization. For reactions that produce HCO, such as $OH + CH_2O \longrightarrow H_2O + HCO$, a fraction of the total rate will produce $H_2O + H + CO$ instead. When this effect is included in a mechanism, the laminar flame speed increases by up to 15%. The modeling results in Figure 1 were obtained through a cumbersome *ad hoc* process that would be difficult to perform consistently for arbitrary radicals in larger systems. The results in Figure 1 suggest that non-Boltzmann reactions contribute meaningfully to macroscopic combustion properties, but current modeling paradigms cannot capture the essential chemical physics.

1.3.2. Reacting while hot

To illustrate the effect of bimolecular reactions involving non-thermalized species, it is helpful to consider the example of low-temperature auto-ignition. The ignition delay time is governed by a sequence of reactions that begins with the formation of an alkyl radical R, progresses by the successive addition of two O_2 molecules, and concludes with the decomposition of a keto-hydroperoxide to form a keto-alkyloxy radical, OQ′O. The net reaction can be written as $RH + OH + 2O_2 \longrightarrow OQ'O + 2OH + H_2O$. This reaction is chain branching, since it starts with a single radical on the left and yields three radicals on the right. Figure 2 presents a simplified potential energy diagram for propane, heptane, dimethyl-ether, and n-butanol. The net reaction for the oxygenated fuels is considerably more exothermic, which is a critical consideration for non-Boltzmann effects.

The first step in this sequence produces a vibrationally hot alkyl radical, R^*. R^* must undergo a succession of non-reactive collisions to quench it to a thermalized R. However, these radicals are reactive with one of the main colliders, O_2. Thus, a small percentage of the nacent R^* will react with O_2 to form vibrationally hot $RO_2{}^*$, which isomerizes to hot $QOOH^*$. If this hot $QOOH^*$ reacts with the second O_2, then additional excess energy is imparted into the nacent O_2QOOH^*, which is carried

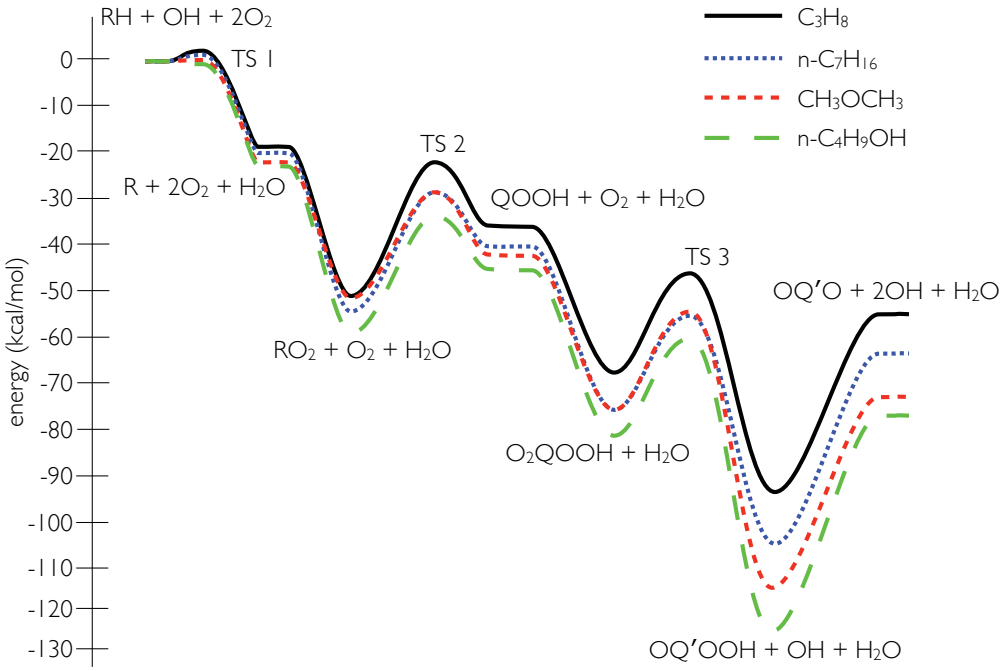

Figure 2: Potential energy diagram illustrating the six key reactions that govern low-temperature auto-ignition for four different fuels.

all the way through to the final step, the decomposition of OQ′OOH to OQ′O + OH.

As with the prompt dissociation of hot species in Figure 1, there is no systematic way to include hot bimolecular reactions in current kinetic models, and these effects must be implemented and tested in a highly laborious, *ad hoc* process. Figure 3 illustrates the particular case of low-temperature ignition of n-heptane under fuel lean conditions, $\phi = 0.5$. The solid black line is the conventional model (in this case, the LLNL mechanism [3, 4]). The three colored lines represent what would happen if a small percentage of the reaction $RH + OH(+2O_2) \longrightarrow R + H_2O(+2\,O_2)$ were replaced with $RH + OH + 2O_2 \longrightarrow OQ'O + 2OH + H_2O$. The 4% (short-dash red) is approximately the upper limit. Even for the smaller cases, these reactions have a pronounced effect. The results in Figure 3 are likely to be an under-estimate, because additional non-Boltzmann reactions should be included, which would accelerate the ignition. Moreover, these results are likely to be even more pronounced for the two oxygenated fuels in Figure 2, since they are considerably more exothermic and thus require more non-reactive collisions to maintain thermalization. Indeed, this cascade of non-Boltzmann reactions could be the root cause of the unexplained dependence of n-butanol ingition delays on oxygen partial pressures.

Proposed Technical Solution: To capture non-Boltzmann effects, we will devise and implement one or more strategies for representing elementary rate coefficients as functions of temperature, pressure, and fluid composition.

1.3.3. Collider effects

Reactions that proceed through one or more unimolecular intermediates are ubiquitous in gas-phase chemistry, and the rate constants for these reactions are functions of pressure. This pressure dependence results from the competition between collisional energy transfer and chemical transformation. Importantly, the rate of collisional energy transfer itself depends strongly on the bath-gas composition. Most kinetic studies are performed in a single bath, *e.g.*, excess N_2 or Ar, but real-world systems involve vastly more diverse bath gases. Combustion by-products such as H_2O, CO_2, and CO have markedly different rates of collisional energy transfer than N_2 and Ar. Although laboratory experiments (*e.g.*, shock tubes) can measure rate constants with high accuracy, it is unrealistic (and in many cases unfeasible) to measure them with a suitable range of colliders. In principle, theoretical calculations can "fill in" all the missing information by providing rigorously computed rate constants for differ-

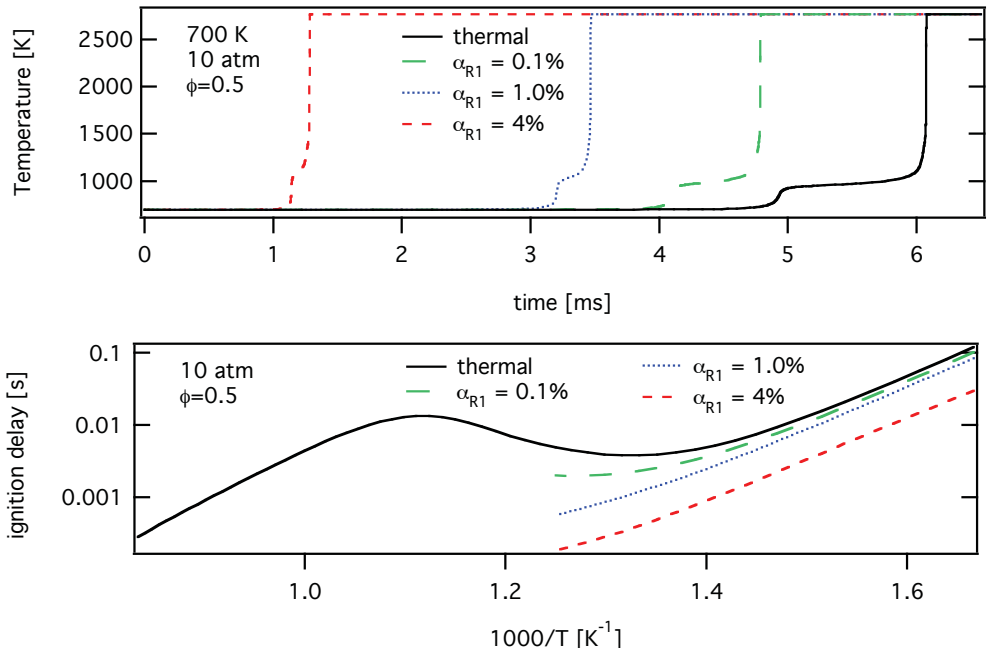

Figure 3: illustration of the impact that non-Boltzmann reactions could have on the ignition delay time of n-heptane. The solid black line is the literature mechanism. The dashed red line is the theoretical maximum percentation of non-Boltzmann effects. The dotted blue and dashed green lines represent a fraction of the theoretical maximum.

ent colliders. The current approach is to represent these collider-specific results independently using the PLOG formalism.

The challenge is that current software cannot represent how multi-component mixtures influence pressure-dependent rate constants in a physically realistic manner. The current state of the art assumes a linear mixture rule. This rule, while convenient, is not physically justifiable (at a minimum, it consistently over-predicts the rate constants). Instead, accurate extrapolation from detailed laboratory studies to engine-relevant conditions requires the ability to consider non-linear mixture rules, but current software lacks the flexibility to consider these effects. Preliminary evidence suggests that errors induced by the standard linear mixture rule can in some circumstances exceed the errors in laboratory measurements and in theoretical predictions. In other words, the considerable effort that is expended to obtain accurate measurements and predictions is then wasted when it comes to calculations involving multi-component mixtures.

Proposed Technical Solution: Advanced non-linear mixture rules will be implemented, in collaboration with Prof. Michael Burke at Columbia University.

1.3.4. Transport properties

The explosion of computational chemistry, from accurate electronic structure theory to large-scale molecular dynamics simulations, has provided a wealth of information on intermolecular interactions. These atomistic simulations can be used to compute transport properties, such as diffusion coefficients, with impressive accuracy. However, the standard approach to computing transport properties in reactive flow simulations assumes that intermolecular interactions are described by the Lennard-Jones(12,6) potential. The LJ(12,6) model is qualitatively correct, but it is now well-established that it is not an accurate representation of intermolecular interactions, particularly at small separation distances. Thus, despite enormous advances in computational chemistry, the standard approach to estimating transport properties still relies on a century-old approximation that was born out of a mathematical convenience. Species-specific binary diffusion coefficients can be computed accurately using routine methods, but there is no agreed upon formalism or syntax for including these results in simulations. Alternative parameterizations of interaction potentials, such as the Buckingham model, are known to be more accurate than the LJ(12,6) model, but it is currently impossible to test the sensitivity of combustion properties

to alternative transport models.

The importance of accurate diffusion coefficients, and the sensitivity of global models to them, will increase as the density approaches that of a condensed phase. Once the density approaches that of a liquid, the rate is frequently diffusion controlled; in other words, the dynamical bottleneck is no longer the intrinsic activation in the reaction, but rather is the ability of reactants and products to diffuse through the solvent medium. Moreover, condensed phases with mobile charged species are pervasive in electrochemical energy devices, such that new thermodynamic and transport models must be implemented to enable accurate transport predictions.

Proposed Technical Solution: Provide more user flexibility to define alternative parameterizations for intermolecular potentials, as well as user-defined binary diffusion coefficients.

1.4. Homogeneous Chemistry Software Needs

The homogeneous chemistry field has reached a state where the theory is well ahead of the current capabilities in generalized and widely available chemical kinetics software. As described above, new thermodynamic and chemical kinetic theories are currently developed and implemented in an ad hoc manner, at great cost to individual researchers. Creating software frameworks in a generalized, open-source software platform (e.g., CANTERA) which enable the development, evaluation, and implementation of new theory would greatly accelerate the pace, efficiency, and promulgation of scientific advances in homogeneous chemistry. Moreover, such computational frameworks would allow researchers to evaluate new kinetic theories, for example, in the context of relevant transport and thermodynamic phenomena.

Implementing capabilities in CANTERA to reflect state-of-the-art theory would represent a significant advance in homogeneous chemistry modeling. Furthermore, while it is of course not possible to anticipate future scientific advances, implementing non-ideal capabilities in a generalized and flexible manner (for example, accepting inputs as a generalized polynomial, where appropriate, rather than assuming a particular functional form) will facilitate incorporation of new theories, as they develop. Specific capabilities to be implemented as part of this content area include:

- Implement a wider array of non-ideal equations of state. This includes multi-component Peng Robinson and Soave-Redlich-Kwong, as well as linking to and implementing currently-available models such as CALPHAD [5–7] and COOLPROP [8] for solid and liquid phases.
- Implement software frameworks to incorporate the influence of transition state activity coefficients γ^* on kinetic rate constants (e.g., Eq. (5)).
- Develop software routines to incorporate non-Boltzmann effects. Represent families of non-thermalized reactions where ro-vibrationally hot species may undergo prompt dissociation or otherwise react prior to thermalization.
- Moving beyond Arrhenius rate coefficients: Enable rate constant formulations with complex functional dependence on temperature, pressure, and chemical composition.
- Create software structures to represent collider effects on rate constants. Reflect non-linear mixing rules via generalized, higher-order polynomial expressions.
- Implement generalized binary diffusion coefficient routines, to enable models other than Lennard–Jones(12,6), such as the Buckingham Model.

2. Heterogeneous Chemistry

Team: **Jackson**, Speth, Goldsmith, West, DeCaluwe

2.1. Overview and objectives

Many chemical processes are mediated through surfaces (i.e., heterogeneous phase interfaces). Reactions at interfaces between solid catalyst and gas or liquid flows describe innumerable heterogeneous chemical processes to enhance desired reactions such as reforming of hydrocarbon fuels or impact deleterious reactions such as surface carbon formation during fuel decomposition. Beyond catalysis, numerous heterogeneous chemical processes drive bulk material deposition, such as soot particle growth in combustion environments, or bulk consumption such as ablation of protective surfaces in hypersonic flight. Generalized software tools for modeling heterogeneous surface chemistry at the continuum-level, such as Surface CHEMKIN or CANTERA, have limited capability beyond ideal lattice-site models. These models assume that surface reactions take place on fixed sites with local thermodynamic properties independent of neighboring species or bulk-phase support. Such ideal lattice models have been the basis for linking atomistic modeling of surfaces via density functional theory (DFT) to continuum-level CFD modeling with surface chemistry

[9], but they fail to capture critical phenomena related to surface-species interactions and bulk-surface exchange relevant to many technologies. Establishing and implementing a generalized software framework for interfacial surface chemistry that captures critical non-idealities and the complexities of multi-phase surfaces represents a principal goal for this software development effort.

Ideal lattice surface thermochemistry models such as Langmuir–Hinshelwood (L-H) assume a fixed number of reactive sites that are occupied or not by adsorbed species. The adsorption rate of gas-species, which is often a rate-limiting step in heterogeneous catalysis or growth processes, is the product of: (*i*) the fraction of free sites available for adsorption θ_s, raised to the power of the number of sites ν_{si} needed for a single adsorption event, (*ii*) the collision rate of molecules on the surface Z_k, and (*iii*) the probability that a collision results in an adsorption reaction or 'sticking' on the surface, i.e., a sticking coefficient σ_k. The resulting adsorption reaction rate can be expressed as:

$$\dot{s}_i = \sigma_i Z_k \theta_{si}^{\nu_{si}} = \left[\sigma_i^\circ \left(\frac{T}{T^0}\right)^{n_{\mathrm{Ti}}} \exp\left(-\frac{E_{\mathrm{ai}}}{RT}\right)\right]\left[\frac{P_k}{\sqrt{2\pi W_k RT}}\right]\theta_{si}^{\nu_{si}} \tag{6}$$

where the square brackets capture general expressions, respectively, for the activated adsorption sticking coefficient and for the ideal gas collision flux with the surface. In typical surface chemistry (often referred to as microkinetic) modeling, adsorption reactions are coupled to reactions involving surface species reacting with each other and desorption reactions that create open lattice surface sites. An Arrhenius, mass-action rate expression for reactions involving only surface-bound reactants relates the rate to total surface site density Γ_i° and site fractions for each reacting species θ_{ki}:

$$\dot{s}_i = A_i^0 \left(\frac{T}{T^0}\right)^{n_{\mathrm{T},i}} \exp\left(-\frac{E_{\mathrm{ai}}}{RT}\right)\prod_{k=1}^{K}\left(\Gamma_i^\circ \theta_k\right)^{\nu'_{ki}} \tag{7}$$

The number of surface reactions can grow to many tens or more for catalytic mechanisms, such as CH_4 steam reforming, as illustrated in the figure Figure 4. Implementing surface reaction mechanisms for Ni-based CH_4 steam reforming [10] or for other singificant catalytic chemistries like methane dehydroaromization in gas-to-liquids fuel processing [11, 12] is challenged by the number of uncertain rate parameters such as σ_{ki}, A_i^0, $n_{\mathrm{T},i}$, and $E_{\mathrm{a},i}$ for specific elementary reaction steps. Although net species production or consumption rates in the interacting flow may have limited sensitivity to reaction rate parameters of many individual surface reactions, it is still vital that the individual reaction rates predict equilibria that are consistent with species thermodynamics. For a specific surface reaction that does not involve adsorption/desorption, thermodynamic consistency requires that rate parameters for the forward reactions (index fi) and reverse reactions (index ri) are linked to the different species molar enthalpies $\bar{h}_k$ and standard state entropies $\bar{s}_k^0$ as follows.

$$\frac{A_{fi}^0}{A_{ri}^0}\left(\frac{T}{T^0}\right)^{(n_{\mathrm{Tfi}}-n_{\mathrm{Tri}})} \exp\left(\frac{E_{\mathrm{ari}}-E_{\mathrm{afi}}}{RT}\right) = \exp\left(\frac{\Delta\bar{s}_i^0}{R}\right)\exp\left(-\frac{\Delta\bar{h}_i}{RT}\right)\Gamma_i^{\circ\sum_{k=1}^{K}\nu_{ki}}. \tag{8}$$

The variables $\Delta\bar{s}_i^0$ and $\Delta\bar{h}_i$ in Eq. (8) represent the change in standard state entropy and the change in enthalpy of the forward reaction. Eq. (8), which does not apply for adsorption/desorption equilibria, does provide a relationship for other surface reaction rate parameters with the thermodynamics for an ideal lattice model.

For surface chemistry, surface thermodynamic quantities such as $\bar{h}_k$ and $\bar{s}_k^0$ are often influenced by strong interaction potentials between surface species and between the surface and underlying bulk phases. For fixed lattice surface chemistry models, the enthalpy of interaction potentials can result in mean-field reaction expressions for $E_{\mathrm{a},ri} - E_{\mathrm{a},fi} = \Delta\bar{h}_i$ with functional dependencies on surface site fractions θ_k or bulk-phase fractions X_{kb}. However, state-of-the-art modeling tools for fixed lattice surface chemistry have neither adequate data structures for consistent thermodynamic properties nor rate expressions to capture these species-dependent interaction potentials, other than by entering them on a reaction-by-reaction basis. For example in Table 2, $E_{\mathrm{a},ri} - E_{\mathrm{a},fi}$ for reactions with CO(s) indicate an apparent interaction potential between CO(s) sites of 50.0 kJ mol^{-1}, which shows up in the effective $\Delta\bar{h}_i$ for relevant CO(s) reactions. It is highly unlikely that the CO-CO interaction potential is the only non-ideal interaction for the numerous surface species in Table 2.

A more comprehensive approach to surface species interaction potentials can enable significant gains in heterogeneous chemistry modeling. For example, interaction potentials and associated coverage dependencies of $\Delta\bar{h}_i$ can lead to metastable equilibrium points and provide a means for predicting commonly observed hysteresis in catalytic and surface growth reactions. Currently, such dependencies are handled on a reaction-by-reaction basis. However, for more complex surface thermodynamic

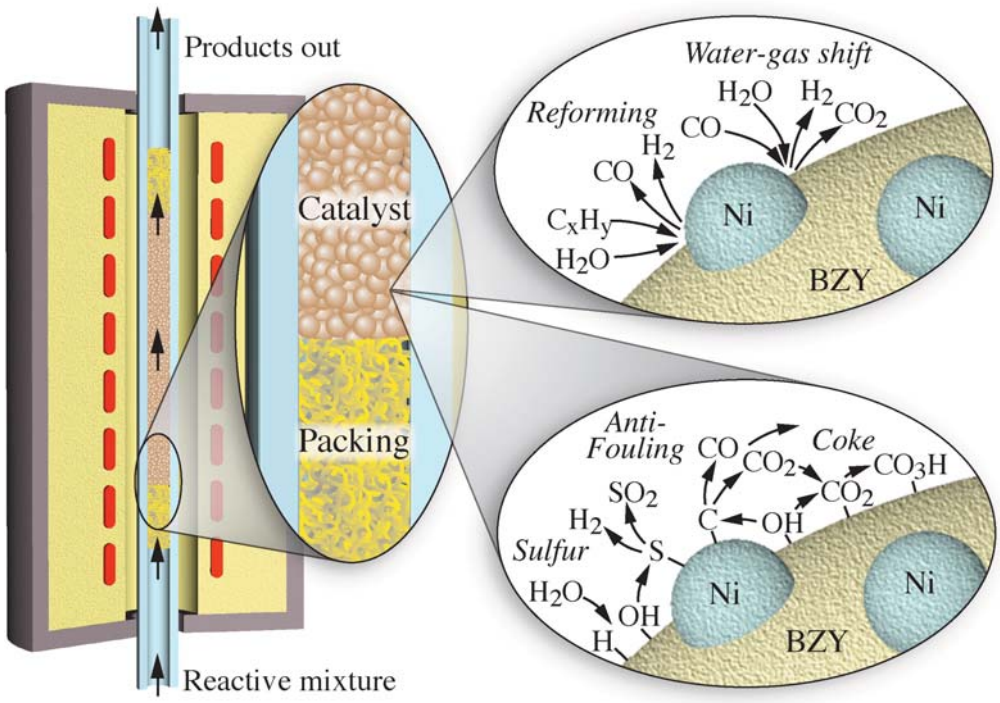

Figure 4: Schematic of a packed bed reactor configuration for testing hydrocarbon C_xH_y steam reforming over particulate Ni catalyst supported on an active yttria-doped barium zirconate (BZY) matrix. The two callouts show desired surface chemistry on the Ni catalyst and impact of the support and possible surface contamination chemistry (with S or C).

models, surface species and bulk interactions can be estimated or learned through DFT simulations to derive the interfacial equivalent of non-ideal state equations. The current proposed effort will develop data structures and tools for deriving such non-ideal surface thermodynamics and consistent surface mechanisms to support continuum-level modeling of surface-mediated chemistry.

Moving beyond ideal-lattice interface models will enable more robust surface mechanisms for chemical device/process modeling by facilitating true multi-scale simulation from atomistic modeling to continuum-level simulations. In addition, numerous other opportunities as described below present high-impact opportunities to improve surface chemistry modeling related to bifunctional interfaces, nucleation/growth, and dynamic surface structures important in catalysis, materials processing, and other reactions. Such multi-functional and dynamic surface processes must move beyond simple fixed-lattice, L-H mechanisms. The proposed effort here will advance new approaches to surface chemistry modeling and provide a general framework to make them widely accessible to reacting flow modeling communities across numerous fields:

- Develop general surface state models and data structures to facilitate thermodynamically consistent mechanism construction with species interactions that can be readily informed/trained by atomistic modeling,
- Establish numerical models and methods for simulating bifunctional and multi-phase surface reactions that incorporate surface mobility and spillover mechanisms such as oxidative coupling,
- Provide thermodynamic and transport framework for supporting bulk-surface interactions that drive critical surface phenomena in catalysis, membranes, and materials processing,
- Formulate heterogeneous chemical kinetics to facilitate dynamic surface formations that support nucleation, surface growth, evaporation, and surface restructuring.

2.2. Extending heterogeneous catalysis lattice models

The limitations of fixed-lattice surface chemistry have largely restricted detailed mechanism development to

simple surfaces, such as metals where phenomena such as bifunctionality, bulk-surface reactions, and surface site transformations are neglected. Table 2 provides an example of such a detailed mechanism for the previously illustrated CH_4 steam reforming on a Ni metal catalyst in Figure 4 [10]. This detailed mechanism has been used to predict performance for technologically significant processes such as solid-oxide fuel cells and catalytic steam reformers. However, because of the inherent limitations of the ideal lattice model, it cannot assess critical off-design or rapid-transient conditions where phenomena such as bulk Ni oxidation or carbon surface growth impact operation and durability. Firstly, both Ni oxidation and surface carbon formation depend highly on reactions between the oxide support and the Ni surface, and as such, surface chemistry models must account for interactions across multiple surfaces. Secondly, Ni oxidation and carbon formation reactions result in bulk and surface site transformations that violate fixed lattice site assumptions.

A detailed mechanism such as Table 2 can mark the approach toward conditions that lead to catalyst oxidation or surface carbon growth, but the fixed lattice site model and associated rate expressions and parameters in Eqs. (6) and (7) cannot capture dynamic evolution of the catalyst bulk and surface. Bulk and surface reactions associated with catalyst oxidation or carbon formation impact gas-surface interactions both thermodynamically (in terms of $\bar{h}_k$ and $\bar{s}^0_k$) and kinetically (in terms of σ_{ki} and $E_{\mathrm{a},i}$). This highlights the benefits of moving beyond the limitations of fixed lattice models. Key extensions to lattice models can offer significant enhancements to modeling capabilities that accelerate development of technologies driven by surface-mediated processes:

- Provide framework for handling surface-species interactions and bulk-surface interactions that ensure thermodynamically consistent non-ideal mechanisms,
- Establish computational tools that support the linking of atomistic modeling results to lattice mechanism development,
- Enhance modeling capabilities for capturing bulk-surface exchange and its effects on surface and bulk phase reactions,
- Extend modeling interfaces to support bifunctional surface and multi-surface mechanisms including surface transport.

The high density of surface species on reactive surfaces strongly suggests that non-ideal thermodynamics involving surface-species and bulk-surface interactions will play a critical role in surface chemistry. Even with the rapid rise of atomistic modeling employed to inform microkinetic surface mechanism development, such non-idealities have been largely neglected. This shortcoming may in part be explained by the lack of formal frameworks and structures in surface reaction modeling tools to support the inclusion of non-ideal interactions in a thermodynamically consistent framework. Targeted atomistic modeling can estimate surface-species interaction potentials on a fixed lattice surface to derive a functional model of nonreaction thermodynamics $\Delta\bar{h}_i$ that include non-ideal excess enthalpy functions $\bar{h}_{\mathrm{ex,s}}$. Excess enthalpy and thus $\Delta\bar{h}_i$ can depend on the vectors of surface site coverages $\left[\theta_{js}\right]$ and underlying bulk species concentrations $\left[X_{jb}\right]$, as shown here.

$$\begin{aligned} E_{\mathrm{a},ri} - E_{\mathrm{a},fi} &= \Delta\bar{h}_i \\ &= \sum_{k_g=1}^{K_g} \nu_{k_g i}\bar{h}_{k_g} + \sum_{k_s=1}^{K_s} \nu_{k_s i}\left[\bar{h}^0_{k_s} + \bar{h}_{\mathrm{ex,s}}\left([\theta_{js}],[X_{jb}]\right)\right]. \end{aligned} \tag{9}$$

$\bar{h}_{\mathrm{ex,s}}$ can be a complex function that depends upon ordered surface phases structures as demonstrated for O coverage on transition metals [13]. On the other hand, for some surfaces, $\bar{h}_{\mathrm{ex,s}}$ can be reasonably approximated with a linear dependence on $\left[\theta_{js}\right]$ and $\left[X_{jb}\right]$ as suggested here.

$$\begin{aligned} &E_{\mathrm{a},ri} - E_{\mathrm{a},fi} = \Delta\bar{h}_i = \\ &\sum_{k_g=1}^{K_g} \nu_{k_g i}\bar{h}_{k_g} + \sum_{k_s=1}^{K_s} \nu_{k_s i}\left[\bar{h}^0_{k_s} + \sum_{j_s=1}^{K_s} \epsilon_{k_s j_s}\theta_{j_s} - + \sum_{k_b=1}^{K_b} \epsilon_{k_s k_b}X_{k_b}\right] \end{aligned} \tag{10}$$

$\epsilon_{k_s j_s}$ and $\epsilon_{k_s k_b}$ are matrices that catalog the change in $\bar{h}_{k_s}$ due to the presence of surface species j_s and of bulk-phase mole fraction of k_b, respectively. Although approximate $\epsilon_{k_s j_s}$ have been documented for simple chemistry such as CO or O_2 adsorption on metal catalysts, the complexity of incorporating full matrices of $\epsilon_{k_s j_s}$ and $\epsilon_{k_s k_b}$ for a detailed mechanism like Table 2 requires a structured framework that automates the incorporation of such non-idealities into surface thermochemistry.

Incorporating non-idealities even in a simple form like Eq. (10) increases the input parameters necessary for a surface mechanism. Computational tools are needed to facilitate assignment of meaningful parameters for $\bar{h}_{\mathrm{ex,s}}$ with confidence. In this regard, it is critical that software tools like Cantera incorporate accessible data structures that can link to the wealth of thermodynamic information

Table 2: Heterogeneous reaction mechanism for CH_4 reforming on Ni-based catalysts.

	Reaction	A_i or σ_{ki}	$n_{T,i}$	$E_{a,i}$
f1.	H_2 + Ni(s) + Ni(s) → H(s) + H(s)	$1.000 \times 10^{-02\,\dagger}$	0.0	0.00
r1.	H(s) + H(s) → Ni(s) + Ni(s) + H_2	$5.593 \times 10^{+19}$	0.0	88.12
f2.	O_2 + Ni(s) + Ni(s) → O(s) + O(s)	$1.000 \times 10^{-02\,\dagger}$	0.0	0.00
r2.	O(s) + O(s) → Ni(s) + Ni(s) + O_2	$2.508 \times 10^{+23}$	0.0	470.39
f3.	CH_4 + Ni(s) → CH_4(s)	$8.000 \times 10^{-03\,\dagger}$	0.0	0.00
r3.	CH_4(s) → Ni(s) + CH_4	$5.302 \times 10^{+15}$	0.0	33.15
f4.	H_2O + Ni(s) → H_2O(s)	$1.000 \times 10^{-01\,\dagger}$	0.0	0.00
r4.	H_2O(s) → Ni(s) + H_2O	$4.579 \times 10^{+12}$	0.0	62.68
f5.	CO_2 + Ni(s) → CO_2(s)	$1.000 \times 10^{-05\,\dagger}$	0.0	0.00
r5.	CO_2(s) → Ni(s) + CO_2	$9.334 \times 10^{+07}$	0.0	28.80
f6.	CO + Ni(s) → CO(s)	$5.000 \times 10^{-01\,\dagger}$	0.0	0.00
r6.	CO(s) → Ni(s) + CO	$4.041 \times 10^{+11}$	0.0	112.85-50.0$\theta_{CO(s)}$
f7.	O(s) + H(s) → OH(s) + Ni(s)	$5.000 \times 10^{+22}$	0.0	97.90
r7.	OH(s) + Ni(s) → O(s) + H(s)	$2.005 \times 10^{+21}$	0.0	37.19
f8.	OH(s) + H(s) → H_2O(s) + Ni(s)	$3.000 \times 10^{+20}$	0.0	42.70
r8.	H_2O(s) + Ni(s) → OH(s) + H(s)	$2.175 \times 10^{+21}$	0.0	91.36
f9.	OH(s) + OH(s) → O(s) + H_2O(s)	$3.000 \times 10^{+21}$	0.0	100.00
r9.	O(s) + H_2O(s) → OH(s) + OH(s)	$5.423 \times 10^{+23}$	0.0	209.37
f10.	O(s) + C(s) → CO(s) + Ni(s)	$5.200 \times 10^{+23}$	0.0	148.10
r10.	CO(s) + Ni(s) → O(s) + C(s)	$1.418 \times 10^{+22}$	-3.0	115.97-50.0$\theta_{CO(s)}$
f11.	O(s) + CO(s) → CO_2(s) + Ni(s)	$2.000 \times 10^{+19}$	0.0	123.60-50.0$\theta_{CO(s)}$
r11.	CO_2(s) + Ni(s) → O(s) + CO(s)	$3.214 \times 10^{+23}$	−1.0	86.50
f12.	HCO(s) + Ni(s) → CO(s) + H(s)	$3.700 \times 10^{+21}$	0.0	-50.0$\theta_{CO(s)}$
r12.	CO(s) + H(s) → HCO(s) + Ni(s)	$2.338 \times 10^{+20}$	−1.0	127.98
f13.	HCO(s) + Ni(s) → O(s) + CH(s)	$3.700 \times 10^{+24}$	−3.0	95.80
r13.	O(s) + CH(s) → HCO(s) + Ni(s)	$7.914 \times 10^{+20}$	0.0	114.22
f14.	CH_4(s) + Ni(s) → CH_3(s) + H(s)	$3.700 \times 10^{+21}$	0.0	57.70
r14.	CH_3(s) + H(s) → CH_4(s) + Ni(s)	$4.438 \times 10^{+21}$	0.0	58.83
f15.	CH_3(s) + Ni(s) → CH_2(s) + H(s)	$3.700 \times 10^{+24}$	0.0	100.00
r15.	CH_2(s) + H(s) → CH_3(s) + Ni(s)	$9.513 \times 10^{+22}$	0.0	52.58
f16.	CH_2(s) + Ni(s) → CH(s) + H(s)	$3.700 \times 10^{+24}$	0.0	97.10
r16.	CH(s) + H(s) → CH_2(s) + Ni(s)	$3.008 \times 10^{+24}$	0.0	76.43
f17.	CH(s) + Ni(s) → C(s) + H(s)	$3.700 \times 10^{+21}$	0.0	18.80
r17.	C(s) + H(s) → CH(s) + Ni(s)	$4.400 \times 10^{+22}$	0.0	160.49
f18.	O(s) + CH_4(s) → CH_3(s) + OH(s)	$1.700 \times 10^{+24}$	0.0	88.30
r18.	CH_3(s) + OH(s) → O(s) + CH_4(s)	$8.178 \times 10^{+22}$	0.0	28.72
f19.	O(s) + CH_3(s) → CH_2(s) + OH(s)	$3.700 \times 10^{+24}$	0.0	130.10
r19.	CH_2(s) + OH(s) → O(s) + CH_3(s)	$3.815 \times 10^{+21}$	0.0	21.97
f20.	O(s) + CH_2(s) → CH(s) + OH(s)	$3.700 \times 10^{+24}$	0.0	126.80
r20.	CH(s) + OH(s) → O(s) + CH_2(s)	$1.206 \times 10^{+23}$	0.0	45.42
f21.	O(s) + CH(s) → C(s) + OH(s)	$3.700 \times 10^{+21}$	0.0	48.10
r21.	C(s) + OH(s) → O(s) + CH(s)	$1.764 \times 10^{+21}$	0.0	129.08

* Arrhenius parameters for the rate constants written in the form: $k_{fi} = AT^n \exp(-E/RT)$. The units of A are given in terms of moles, centimeters, and seconds. E is in kJ/mol.
† Sticking coefficient pre-exponential σ^0
Total available surface site density is $\Gamma = 2.60 \times 10^{-9}$ mol/cm^2.

derived from atomistic DFT models of surface species phases and interactions and associated changes in electronic structure of the bulk phase.

Energetic and entropic properties for fixed lattice models can be numerically derived from surface partition functions that combine numerical sampling of atomistic modeling and classical analytical expressions of internal energy storage modes derived from known molecular structure of adsorbed species [14]. Current data structures for microkinetic modeling in CANTERA and other software tools do not facilitate the transfer of molecular structural data such as surface-adsorbate bond energetics and vibrational frequencies into the calculation of thermodynamic properties. The proposed effort will develop those data structures and functions and demonstrate key examples to support the development of lattice-site surface chemistry mechanisms that maintain thermodynamic consistency for non-ideal surface models. This will reduce the degree of empiricism involved in formulating complex non-ideal surface thermochemistry. Algorithms that can learn from atomistic simulations will improve numerical estimations of configurational contributions to surface partition functions to develop appropriate functions for $\bar{h}_{ex,s}$ for improved fidelity surface chemistry.

Much of the advances in non-ideal surface thermo-

chemistry models have focused on metallic surfaces. However, many reactive surfaces for catalysis involve carbides, oxides, or other ceramic materials. Like metals, ceramic surfaces can be modeled to first approximation with fixed lattice sites, although in many cases such as zeolites with multiple types of surface sites. Ceramic materials readily exhibit strong bulk-surface interactions in reactive environments. For example, oxide surfaces, like the yttria-doped barium zirconate (BZY) surface in Figure 4, can react with the underlying bulk phase due to mobile oxide vacancies, polarons, and other defects. Depending on the effective oxygen partial pressure p_{O_2}, sub-surface oxide vacancies $V_O^{\bullet\bullet}$ can pull surface oxygen O(s) into the bulk BZY, as shown in reaction 11, to create bulk phase polarons $O_O^{\bullet}$ and a surface vacancy V(s),

$$O(s) + V_O^{\bullet\bullet} + O_O^{\times} \rightleftharpoons 2O_O^{\bullet} + V(s) \qquad (11)$$

This reaction can change the BZY sub-surface electronic structure and the resulting surface thermochemistry. The incorporation of thermodynamically consistent bulk-surface reactions for oxides, other ceramics, and bulk-active metals will require additional reaction rate functionality as well as thermodynamic functions akin to eq. 9. Ideally, these models will be informed and guided by links with external atomistic modeling of these sub-surface interactions to develop robust and accurate models for the bulk composition $\left[X_{jb}\right]$ dependency in $\bar{h}_{\mathrm{ex,s}}$. Such dependencies can still incorporate fixed lattice chemistry with the realization that major bulk compositional changes will likely lead to surface restructuring and the need for other approaches to surface chemistry modeling.

Figure 4 illustrates a common configuration where multiple surfaces interact in one system. The primary catalyst (metallic Ni particles) resides on the BZY oxide support that also participates in reactions. Although both surfaces can be approximated with independent fixed-lattice models, reactions can occur at the three-phase boundaries (tpb) of the Ni-BZY-gas. These tpb reactions, such as oxidation of Ni-adsorbed fuel species at the tpb or spillover of H from the Ni to the oxide surface, require a modeling framework that permits cross-surface reactions with consistent thermodynamics. CANTERA has developed the framework for such reactions but improved implementation of consistent thermodynamics for these reaction models is required. Further, 'tpb' reactions create surface chemical potential gradients that drive surface transport on length scales of nanometers to microns. Surface transport involves activated diffusion over barriers between lattice sites, which can be extracted from atomistic modeling. Modeling multi-surface tpb reactions and surface transport will require calculations of surface mobility and the effects of surface chemical potential gradients on tpb surface site densities. The development effort here will establish the models for surface transport with fixed lattice chemistry involving multi-surface tpb reactions. The incorporation of surface-transport limited reactions in conjunction with non-ideal thermochemistry will extend modeling capabilities to allow for complex reactions like CH_4 reforming on Ni-BZY composites to be simulated in a more robust manner with higher fidelity for continuum-level reactor design.

2.3. Surface chemistry for bulk deposition and removal

The previous section focused on fixed lattice surface chemistry where the dynamic nature of the surface was principally limited to changes in adsorbate species and perhaps minimal changes in the bulk phase due to species incorporation. However, in those cases, the surface lattice sites did not significantly change in nature or structure. Many surface chemistry processes involve significant changes in the surface structure due to bulk-phase deposition and/or removal. For example, Figure 5 shows how bulk oxidation of a transition metal (Pd) and the shift in bulk-surface thermodynamics leads to substantial hysteresis and metastable states in heating and cooling cycles [13, 15]. Such transitions may also occur in dynamic material processes such as sputtering or aerosol formation where the reactive surface and underlying bulk structures have dynamic thermodynamic properties as well as number of sites. Similarly surface reactions involving bulk removal such as solid fuel combustion/gasification and surface ablation also require consideration of evolving surface structures and thermodynamics for reliable simulation.

Dynamic surface structures invalidates the fixed-lattice modeling discussed in the previous section and requires additional surface chemistry modeling capabilities. The chemistry of bulk material deposition or ablation requires some degree of understanding how phase structures are formed or removed during reactions. The numerous advances in solid-phase modeling of structural materials such as metals and ceramics have improved the ability to quantify the thermodynamics and kinetics of bulk-material transformations that underpin surface. Tools such as Thermo-Calc and FactSage have developed sophisticated numerical tools including the CALPHAD methodology to derive experimentally calibrated bulk-phase diagrams and associated thermodynamics for a host of material sets. These tools can provide a taxonomy and framework on which to build dynamic bulk

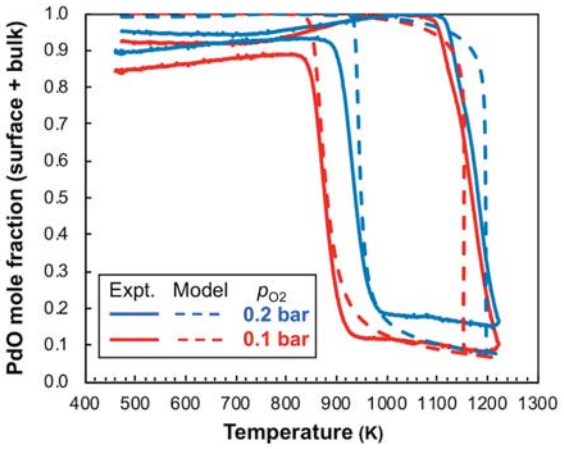

Figure 5: Hysteresis in bulk and surface oxidation and reduction of Al_2O_3-supported polycrystalline Pd/PdO catalysts as observed in experiments and captured in modeling derived from non-ideal surface and bulk thermodynamics as reported in [15]

chemistry models and associated free-surface chemistry models that support bulk deposition and ablation reactions. The current effort in heterogeneous chemistry modeling for a next-generation Cantera will develop links between those tools and other bulk-phase modeling advances to achieve the following objectives toward advancing dynamic surface chemistry models:

- Provide framework for linking state-of-the-art bulk-phase thermochemistry models with reactive surface chemistry models that involve deposition and/or ablation of bulk material,
- Formulate heterogeneous chemical kinetic functions that account for dynamic evolution of surface sites and thermodynamics due to deposition or removal
- Implement bulk-surface chemistry modeling tools that permit the formation and destruction of aerosols, precipitates, and particulate materials.

Many important surface-mediated reactions, such as the aforementioned Pd oxidation in Figure 5 as well as carbon deposition and associated carbide formation [16], have significant bulk transformation and surface growth and/or restructuring. These examples have additive bulk reactions and in both cases the number and type of surface sites available at the interface with the surrounding fluid are dynamic. The approach to modeling such reactions requires at a minimum that the surface site density Γ_i, the number of surface sites, or the interface area be treated dynamically. In addition the transition of material from the surface to the bulk or vice versa must be accounted for in the bulk material thermodynamics models. Dynamic bulk models in cases such as oxide or carbide formation material models that can handle dynamic composition evolution not just in stoichiometry but also in terms of phase. To this end, adaptation of models like those developed in state-of-the-art materials modeling like the CALPHAD models [5–7] in Thermo-Calc and FactSage provide an ideal approach for simulating realistic bulk materials evolution during additive surface reactions. The implementation of such material thermochemical models into Cantera will require the development of new phases with dynamic properties that vary within compositional space.

In addition to additive bulk processes, subtractive processes like coal combustion or material ablation presents some of the same challenges. In some cases, reactions that involve loss of surface material can cause very dynamic surface structures with substantial increases in porosity as in coal combustion. As with additive processes, the need for dynamic models of surfaces as well as of any gas and bulk-phase transport to the surfaces. The implementation of such dynamic surface models will enable advances in critical processes such as coal combustion or oxide ablation in hypersonic flight. The combination of subtractive and additive processes in redox cycles for processes like chemical looping will also provide a key test for incorporating dynamic bulk material models based on state-of-the-art material thermochemical frameworks.

Additional bulk-surface modeling capabilities must be implemented for simulating aerosol growth and destruction for simulating combustion of solid or liquid fuels, the formation of soot, and the condensation of emissions species into atmospheric particulate matter. Aerosol particles are subject to a homogeneous and heterogeneous nucleation, coagulation and coalescence, and surface growth and ablation through surface reactions and fragmentation. The rates of these processes both depend on and determine how a population and structure of aerosol particles evolves, which often requires, coupled solving of both the gas phase chemistry and aerosol surface chemistry.

While many aerosol models have been described in the literature [17–19], software implementations of the necessary heterogeneous chemistry models are generally not available, particularly in generalized forms where they can be modified, extended, or coupled to other codes. Implementing individual tightly-coupled aerosol model in Cantera would not be beneficial,.but development of a general aerosol chemistry framework through which

a range of aerosol models, including both sectional and moment-based methods, can have great impact for the relevant scientific and engineering . Functionality for coupling models built on this framework will be introduced for both CANTERA's reactor network model and its one-dimensional reacting flow (flame) model. To demonstrate the application of this framework, a high-level implementation of a soot model will be developed, incorporating soot nucleation from gas-phase species, growth through a range of chemical mechanisms, coagulation, and oxidation.

2.4. Heterogeneous Chemistry Software Needs

Heterogeneous chemistry at solid-fluid interfaces remains an area where the impact of improved fidelity and flexibility of models in generalize software frameworks has not been realized. The links between rapidly advancing atomistic modeling and mean-field microkinetic models for continuum-level device modeling have not been fully exploited for developing non-ideal thermodynamics and kinetics for heterogeneous chemistry mechanisms. In addition, the advances in thermochemical modeling of bulk materials and their phase transitions in compositional space has not been well established except in a few isolated efforts. CANTERA, as the proposed software environment, provides an ideal platform for developing the computational functions, data structures, and solvers that will enable more timely and rigorous interfaces between continuum-level heterogeneous chemistry models and atomistic modeling and state-of-the-art bulk materials thermodynamic models.

New heterogeneous chemistry modeling for surface and bulk phases in CANTERA as detailed above presents a broad range of needs to be addressed in this proposed effort. New data structures, new dynamic non-ideal bulk and surface models are needed that have accessible and flexible links to atomistic thermochemical calculations and to advanced material phase models. Furthermore, the ability to enhance heterogeneous chemistry by allowing heretofore static properties such as surface sites and bulk thermodynamics to evolve is critical for advancing the software capabilities for heterogeneous chemical process simulation. Specific capabilities to be implemented for heterogeneous chemistry include the following:

- Develop new data structures to facilitate complex, non-ideal thermodynamics that evolve with composition for both surfaces and bulk phase.
- Implement tools that allow for surface and bulk-phase energetic and entropic properties to be adopted from atomistic modeling and statistical sampling of relevant configurations derived from atomistic modeling
- Establish numerical models for thermodynamically consistent three-phase boundary reactions with the possibility for incorporating relevant surface transport effects,
- Provide material phase models that link to bulk materials modeling frameworks in Thermo-Calc and/or FactSage to support the modeling of bulk additive and subtractive processes that involve notable compositional changes in materials.
- Incorporate new solvers for handling dynamic structure morphologies as well as aerosol formation and destruction.

3. Electrochemistry

Team: **DeCaluwe**, Kee, Jackson, and Goldsmith.

3.1. Overview and objectives

Electrochemistry plays a central role for technologies in energy conversion, energy storage, and material/chemical processing. Corrosion processes and their mitigation are also grounded in electrochemistry. Electrochemical systems such as batteries and fuel cells can provide primary or auxiliary power with high efficiency and low environmental impact for a range of vehicles or stationary applications, including military operations. However, improved energy density and durability are needed to enable broader deployment. Such technological advances must build on fundamental understanding of electrochemical processes over many length scales. Flexible and robust computational tools can play a key supporting role in accelerating that understanding and technology breakthroughs. Advances in atomistic modeling have helped identify materials with finely tuned properties at the nanoscale [20–23], but the lack of concurrent development of modeling frameworks at the device scale impedes the translation of these materials into working electrochemical systems.

With few exceptions, the electrochemistry modeling literature is based on one- or two-step global charge-transfer reactions. In actual applications, the chemistry and electrochemistry are often far more complex. For example, in Li-ion batteries, the formation of solid-electrolyte-interface (SEI) films involves poorly understood, multi-step electrochemistry. Solid-oxide fuel cells

operating on hydrocarbon fuels are typically modeled with two global reforming reactions and one charge-transfer reaction, but actual chemistry involves competing reactions for hydrogen production, fuel oxidation, electrode oxidation, and surface carbon formation. Numerous other examples can illustrate where more complete models of electrochemistry can benefit technology development.

Despite this opportunity, the lack of available software to express electrochemical complexity limits the inclusion of such complexity into experimental interpretation and system analysis. Although current modeling efforts can be effective and useful, there are significant benefits to developing new electrochemistry modeling capabilities as proposed here. Such new modeling tools should be closely coupled with functional relationships and material properties that can be readily extracted from atomistic modeling or basic analytical measurements, rather than from empirical parameter fits that are difficult to generalize.

Fundamental underpinnings for electrochemistry are well documented in textbooks and a vast archival literature and cannot be explained in a few pages. Rather, the following sections attempt to highlight areas where new modeling and software capabilities can advance understanding. Drawing from specific applications in batteries and fuel cells, the following discussion shows how development and deployment of new generalized software capabilities can enable modeling of complex electrochemical processes such as multi-ion species transport and multi-step charge-transfer chemistry, Such advances in electrochemistry software tools can significantly assist the design of new high-performance electrochemical systems to accelerate the pace of technology development.

3.2. Electrochemical device processes

All electrochemical cells depend on interactions between electrodes and electrolytes. Electrodes (anode and cathode) are incorporated with an electronically conductive phase, and electrolytes conduct the ions from one electrode to the other. Mixed conducting electrolytes, such as those used in ionic membrane separators, can conduct multiple ions, often including small polarons. Charge-transfer reactions, which are at the core of electrochemistry, represent the transfer of charge across the electrode/electrolyte phase interfaces. Using illustrations from Li-ion batteries, the following sections elucidate key software needs related to ion transport in the electrolyte and charge transfer at electrolyte/electrode interfaces.

Figure 6 uses a Li-ion battery to illustrate the processes in a typical electrochemical cell. During discharge, electrochemical charge-transfer reactions remove Li from the anode phase, delivering Li^+ ions into the liquid electrolyte and electrons to the anode. The Li^+ ions are transported through the electrolyte in the separator by diffusion and migration to the cathode. Charge transfer reactions between the Li^+ and electrons from the external circuit deliver Li for incorporation into the cathode phase as shown in the inset in Figure 6 and as expressed in its simplest form in Reaction 12.

$$Li^+(el) + e^-(C) \rightleftharpoons Li(ed). \tag{12}$$

$Li^+(el)$ and Li(ed) represent a Li ion within an electrolyte and a charge-neutral Li incorporated in a battery electrode, respectively, and $e^-(C)$ represents an electron provided from the carbon current collector. As written, the forward direction is cathodic, meaning that the reaction consumes electrons, which is the case for a battery cathode during discharge or for a battery anode during charging. The reaction is driven by the Gibbs free energy of reaction (i.e., change in chemical potentials) and by the electric potential difference at the electrode-electrolyte interface. The electric potential differences at the two electrolyte/electrode interfaces and the electric potential gradients needed to drive ion fluxes establish the cell voltage, from which usable electrical energy is derived during discharge.

With simple one-step charge transfer expressions like Reaction 12 and simple single-ion transport as described here, electrochemical modeling can appear relatively straightforward. However, actual Li-ion battery chemistry and ion-transport can be significantly more complex, with secondary charge transfer reactions involving the electrolyte and multi-species ion transport due to mobile electrolyte species. To develop higher performance and intrinsically safe Li-ion batteries with new materials and micro-structures, robust modeling tools must capture these important complexities to accelerate material adoption and facilitate device design.

3.3. Electrolyte-phase transport

Within an electrolyte or ion-conducting phase, molar fluxes of species k can be represented as

$$\mathbf{J}_k = -u_k[X_k]\left(\nabla\mu_k + z_k F\nabla\Phi\right), \tag{13}$$

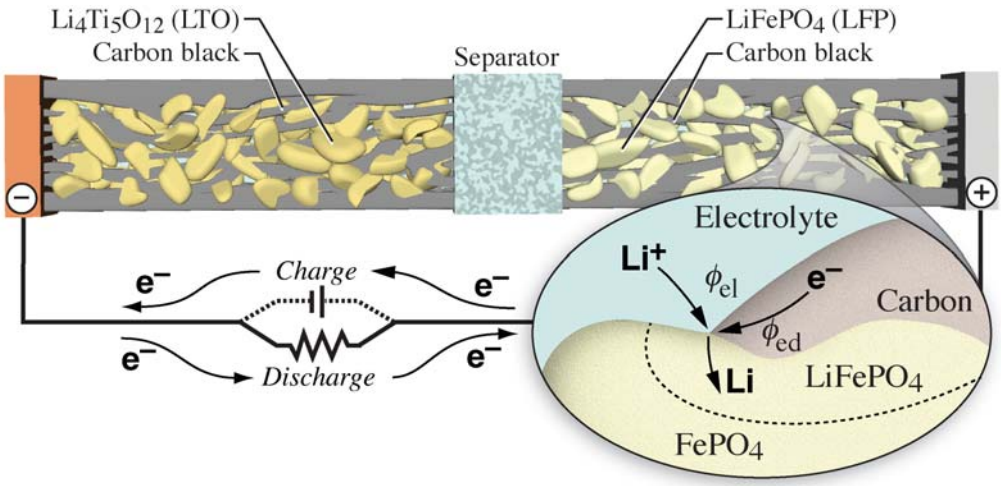

Figure 6: Illustration of processes in a Li-ion battery with a $Li_4Ti_5O_{12}$ anode, liquid electrolyte, and $LiFePO_4$ cathode.

where u_k, $[X_k]$, μ_k, and z_k represent species-specific mobilities, molar concentrations, chemical potentials, and charge numbers respectively. F and Φ represent the Faraday constant and electrostatic potential. Assuming an ideal solution leads to the Nernst–Planck representation for ion fluxes:

$$\mathbf{J}_k = -u_k \left(RT\,\nabla[X_k] + [X_k] z_k F\,\nabla\Phi\right) \tag{14}$$

where R is the universal gas constant. The right-hand-side terms in Eq. (14) represent diffusion and migration respectively.

The commonly used Nernst–Planck formulation assumes dilute solution theory and thus neglects the effects of ion/species interactions on both mobilities u_k and chemical potentials μ_k. A more accurate model for fluxes with one or two mobile ions uses a form of concentrated solution theory (CST) [24], which employs a more accurate representation of μ_k based on non-ideal thermodynamics. Fluxes with CST may be written as

$$\mathbf{J}_k = -u_k[X_k]\left[2RT\left(1+\frac{\partial(\ln f_\pm)}{\partial(\ln[X_k])}\right)(1-t_\pm^\circ)\,\nabla \ln C_{e,k} + z_k F \nabla\Phi\right], \tag{15}$$

where $t_\pm^\circ$ is the ionic transference number. The "thermodynamic factor" $(1+\partial(\ln f_\pm)/\partial(\ln[X_k]))$ is generally fit to experimental measurements and only handles one or two mobile ions in the phase. Furthermore, the impact of non-ideal species interactions on mobilities u_k [25, 26] is not captured in Eq. (15). Thus, although CST transport is widely used in Li-ion battery models [27–38], Eq. (15) has limited value in establishing a general, thermodynamically consistent framework or data structure for modeling diverse ion-conducting phases. Establishing a generalized thermodynamic framework to capture non-ideal species interactions and their impact on transport and thermodynamic properties will be a critical part of the proposed software development.

Electrolyte phases may involve liquids, solids, or multi-phase composites. Current day Li-ion batteries use organic-solvent electrolytes (e.g., ethylene carbonate), but battery safety concerns are driving advances in solid electrolytes [39–41]. High-temperature solid-oxide electrochemical cells use solid ceramic, mixed ionic-electronic conductor electrolytes for oxide-ion and/or proton conduction, placing significant challenges on the electrolyte transport and thermodynamic models extending beyond Eq. (15). Numerous other electrolyte chemistries and architectures, including nano-porous polymers, ionic liquids, and molten salts, present unique and complex transport challenges that will benefit from improved transport modeling frameworks.

A generalized approach to electrolyte transport modeling must tie chemical potentials to structural properties derived from atomistic modeling and analytical measurements to enable model implementation for a range of electrolyte chemistries. Flexible but well-defined input syntax must allow users to choose a suitable level of complexit, similar to how gas-phase transport enables users to choose between mixture-averaged or full multicomponent diffusion, with additional optional physics such as the Soret effect for increased accuracy. In this

proposed effort, a generalized approach for calculating chemical potentials necessary for transport and reaction rates (as described in the next section) will be implemented in Cantera and based on readily defined input parameters that also enable the representation of non-ideal effects on species mobilities. This will provide a framework for evaluating Eq. (13) for a range of electrolyte chemistries. A flexible user-interface syntax will be developed such that the species mobility can be derived from a range of accessible user inputs with software in place to generalize and convert inputs into the required transport parameters.

3.4. Charge-transfer chemistry

Electrochemical reactions occur at phase interfaces where charge is transferred between phases at different electric potentials (Φ_m where m represents a specific phase). The rate of a charge-transfer reaction at a multi-phase interface depends on the difference in Φ_m between the participating electrode and electrolyte phases and the Gibbs free energy of the reaction ΔG_{rxn}. As such, accurate electrochemical reaction modeling must incorporate adequate thermodynamic phase models to support thermodynamically reversible kinetics which are self-contained and generalizable.

To show the importance of formulating a consistent thermodynamic framework to support detailed electrochemical rate calculations, we consider the problem of solid electrolyte interface (SEI) chemistry in the Li-ion battery anodes. The SEI is a passivation layer at the anode-electrolyte interface which prevents electrolyte degradation, but which reduces battery efficiency and power and contributes to capacity fade and safety concerns. Although details of anode-SEI chemistry are not well understood, active studies to understand the charge transfer and related reactions at the interface have elucidated key SEI processes [27, 42, 43]. Table 3 shows the reaction mechanisms proposed by Colclasure et al. [42] for SEI chemistry. As just one example of the need for chemical complexity in electrochemical modeling, software must support sophisticated models beyond that shown in Table 3 to enable the scientifically-guided design of durable, conductive SEI layers for Li-ion batteries. A more complete understanding of SEI formation and evolution requires software which can readily support parallel charge transfer pathways at multiple interfaces, multiple charge carriers in the SEI, and as described in Sec. 2.3, generalized software routines to describe surface chemical reactions with dynamic surface restructuring (in this case, growth and evolution of the SEI phase).

Table 3: Proposed SEI film growth reaction mechanism for a graphite particle [42]. Terms in parentheses describe the phase of each species – Electrolyte (E), SEI surface (S^s), SEI bulk (S^b), and carbon anode (C_6).

Graphite-SEI interface
$C_3H_4O_3(E) + (S^s) \rightleftharpoons C_3H_4O_3(S^s)$
$Li^+(S^s) \rightleftharpoons Li^+(E) + (S^s)$
$C_2H_4(E) + 2(S^s) \rightleftharpoons C_2H_4(S^s)$
$C_3H_4O_3^-(S^s) \rightleftharpoons C_3H_4O_3(S^s) + e^-(S^b)$
$C_2H_4(S^s) + CO_3^{2-}(S^s) \rightleftharpoons C_3H_4O_3^-(S^s) + e^-(S^b) + 2(S^s)$
$CO_3^{2-}(S^s) + 2Li^+(S^s) + (S^b) \rightleftharpoons Li_2CO_3(S^b) + 3(S^s)$
$Li(S^b) + (S^s) \rightleftharpoons V^-(S^b) + Li^+(S^s)$
$Li^+(S^b) + (S^s) \rightleftharpoons Li^+(S^s)$
SEI-electrolyte interface
$e^-(S^b) \rightleftharpoons e^-(C_6)$
$Li(C_6) + V^-(S^b) \rightleftharpoons Li(S^b) + e^-(C_6) + (C_6)$
$Li(C_6) \rightleftharpoons Li^+(S^b) + e^-(C_6) + (C_6)$

Two approaches to charge-transfers kinetics—one based on fundamental Marcus theory and the other based on the commonly used Butler-Volmer formalism—will be implemented and linked in the proposed software development. While properly implemented Marcus theory inherently captures thermodynamic reversibility, it requires precise thermodynamic expressions for all species in a mechanism. The Butler-Volmer formalism, meanwhile, can use easily measured cell voltages to define a sufficient thermodynamic framework with less rigor than is required for Marcus theory. However, the Butler-Volmer form must be carefully formulated to ensure reversibility and accurate cell potential predictions. This becomes particularly challenging when multiple charge transfer reactions occur simultaneously at an electrode/electrolyte interface. The proposed modeling framework will incorporate appropriate tools to enable researchers and developers to assess detailed electrochemical reaction formulations for appropriate cell potential and current evaluations.

3.4.1. Marcus Theory

A reaction's rate of progress can be written in terms of the difference between forward (anodic) and backward (cathodic) reaction rates of progress q_i as

$$q_i = q_{\mathrm{f}i} - q_{\mathrm{b}i} = k_{\mathrm{f}i} \prod_{k=1}^{K} \left(\gamma_k \left[X_k\right]\right)^{\nu'_{ki}} - k_{\mathrm{b}i} \prod_{k=1}^{K} \left(\gamma_k \left[X_k\right]\right)^{\nu''_{ki}}, \tag{16}$$

where $[X_k]$ and γ_k are the molar concentrations and activity coefficients of the participating species, respectively, as discussed in Section 1. For an ideal surface, the activity coefficients of surface-adsorbed species are equal to

unity, and the molar concentrations are the surface coverages $\Gamma_m\theta_{k,m}$, where $\theta_{k,m}$ is the site fraction for species k on the surface of phase m and Γ_m is the total available surface site density on phase m. ν'_{ki} and ν''_{ki} are the reactant and product stoichiometric coefficients, respectively.

When there is a transfer of charge between phases at different electric potential, the charge transfer current density j is evaluated by multiplying the rate of progress by the Faradaty constant F and the total charge transferred, n:

$$j_i = nF\dot{q}_i. \tag{17}$$

The rate is influenced by the electric-potential difference between the two participating phases. The forward and backward rate expressions for each reaction i are written as

$$k_{\mathrm{f}i} = k^{\mathrm{t}}_{\mathrm{f}i} \exp\left[-\beta_{\mathrm{f}i} \sum_{k=1}^{K} \frac{\nu_{ki} z_k F \Phi_{m,k}}{RT}\right], \tag{18}$$

$$k_{\mathrm{b}i} = k^{\mathrm{t}}_{\mathrm{b}i} \exp\left[+\beta_{\mathrm{b}i} \sum_{k=1}^{K} \nu_{ki} z_k F \Phi_{m,k} RT\right], \tag{19}$$

where $\nu_{ki} = \nu''_{ki} - \nu'_{ki}$. $\beta_{\mathrm{f}i}$ and $\beta_{\mathrm{b}i}$ are the forward and backward symmetry factors, with $\beta_{\mathrm{f}i} + \beta_{\mathrm{b}i} = 1$ for elementary (i.e., single-electron transfer) reactions. The forward and backward thermal rate coefficients (i.e., at zero electric-potential difference) are represented as $k^{\mathrm{t}}_{\mathrm{f}i}$ and $k^{\mathrm{t}}_{\mathrm{b}i}$. Generally, each species electric potential $\Phi_{m,k}$ assumes the electric potential of phase in which it exists, Φ_m. Gas phases are usually considered to be electrically neutral.

The thermal reaction rate coefficients k^{t}_i ($k^{\mathrm{t}}_{\mathrm{f}i}$ or $k^{\mathrm{t}}_{\mathrm{b}i}$) are usually represented by a modified Arrhenius expression, as in gas-phase thermochemistry. To satisfy microscopic reversibility and maintain thermodynamic consistency, the thermal component of the backward rate $k^{\mathrm{t}}_{\mathrm{b}i}$ is related to the forward rate $k^{\mathrm{t}}_{\mathrm{f}i}$ via the reaction equilibrium constant K_i as

$$K_i = \frac{k^{\mathrm{t}}_{\mathrm{f}i}}{k^{\mathrm{t}}_{\mathrm{b}i}} = \exp\left(-\frac{\Delta G^\circ_i}{RT}\right) \prod_{k=1}^{K} \left(\left[X^\circ_k\right]\right)^{\nu_k}, \tag{20}$$

where ΔG°_i is the change of the standard-state Gibbs free energy for the reaction, and $\left[X^\circ_k\right]$ are the species concentrations at the standard state at which ΔG°_i is evaluated. Evaluating ΔG°_i, and hence the equilibrium constant requires quantitative thermochemical properties for all species in the reaction, but guarantees that the mechanism is thermodynamically consistent.

3.4.2. Butler-Volmer

The Marcus theory is therefore flexible and inherently thermodynamically reversible, so long the user avoids specifying the reverse reaction rate coefficient. However, nearly all battery and fuel-cell models are based on the Butler–Volmer formulation. In the so-called Butler–Volmer form, the charge-transfer rate can be expressed as

$$j_i = j_{0,i}\left[\exp\left(\frac{\alpha_a F \eta_{\mathrm{act}}}{RT}\right) - \exp\left(\frac{-\alpha_c F \eta_{\mathrm{act}}}{RT}\right)\right], \tag{21}$$

where i is the current at the electrode/electrolyte interface, the rate coefficient i_0 is known as the exchange current density, F is Faraday's constant, η_{act} is the activation overpotential, and α_a and α_c are the anodic and cathodic transfer coefficients, respectively. The activation overpotentials η_{act} are defined as

$$\eta_{\mathrm{act}} = (\Phi_{\mathrm{ed}} - \Phi_{\mathrm{el}}) - (\Phi_{\mathrm{ed}} - \Phi_{\mathrm{el}})^{\mathrm{eq}}, \tag{22}$$

where Φ_{ed} and Φ_{el} are the electrostatic potentials in the electrode and electrolyte phases, respectively. The term $(\Phi_{\mathrm{ed}} - \Phi_{\mathrm{el}})^{\mathrm{eq}}$ represents the equilibrium electrostatic-potential difference, which is typically measurable at open-circuit conditions.

As expressed by Eq. (22), the Butler–Volmer formulation references electrostatic potentials to equilibrium reversible potential differences. At first, this may seem indirect and cumbersome. However, the advantage is that open-circuit voltages can be easily measured and, in many cases, evaluated theoretically. However, as discussed subsequently, as the chemistry becomes more complex, there are also some limitations and complications associated with the Butler–Volmer formulation.

Although the Butler–Volmer formulation is very widely used and has some significant advantages, it also has some very significant limitations. The Butler–Volmer formulation depends on there being a single rate-limiting charge transfer reaction. If the chemistry is represented simply as a single reaction (e.g., Eq. (12)), then the limitations are not demanding. In fact, for such simple chemistry, there is relatively little need for generalized software. However, if the charge-transfer chemistry is a multi-step process, then the Butler–Volmer formulation is much less appropriate and difficult to implement in a generalized software framework, as discussed in the next section.

3.4.3. Thermodynamic consistency

Maintaining thermodynamic consistency and microscopic reversibility are important considerations in most, if not all, chemical kinetics modeling. With gas-phase mixtures, the equilibrium composition is easily evaluated because the thermodynamics for most gas-phase species are readily available. In electrochemistry, the situation is quite different. Consider an elementary electrochemical reaction mechanism, as in Table 3, with the rates being expressed in the form of Marcus theory. Such mechanisms typically involve ionic species and surface adsorbates, the thermodynamic properties for which are largely unknown. These thermodynamics are critical in establishing the equilibrium state, which in turn impacts the predicted equilibrium electrostatic potential differences (i.e., the open-circuit potential). For applications such as batteries or fuel cells, predicting correct open-circuit potentials is a critically important aspect of a model. Thus, it is critically important that a model using the Marcus formalism have complete and accurate thermodynamic properties for all species.

In the Butler–Volmer setting, the reversible half-cell potentials can be specified independently. Then, the activation overpotentials are measured relative to the reversible potentials (Eq. (22)). Thus, a model based on the Butler–Volmer formulation is suited to respecting known reversible potentials and open-circuit potentials with easily accessible user inputs. However, the Butler–Volmer formulation depends on simplifying assumptions, some of which may not be consistent with the intent of the multi-step reaction mechanism. As reaction mechanisms become more complex, the Butler–Volmer formulation becomes increasingly cumbersome. Moreover, depending on details of the multi-step reaction mechanism, thermodynamic properties for individual species may be required to derive the Butler–Volmer exchange current densities. Thus, the potential value of the Butler–Volmer framework is diminished, and maybe inappropriate, with increasing electrochemical complexity.

The Marcus framework is certainly the most appropriate as the electrochemistry complexity increases. Because the Marcus framework demands thermodynamic properties, there is a need to establish and validate the needed properties. Direct measurements, if possible, are difficult. Thus, there are opportunities for developing new atomistic-scale and ab-initio simulation capabilities that are specifically targeted toward ionic species and surface adsorbates.

3.5. Electrochemistry Software Needs

As a practical matter, a general software implementation of the preceding topics is needed. The user should be able to describe the chemistry in terms of understandable reactions, with the software providing the needed functionality to pose and solve complex electrochemistry problems. The challenge for developing such software is much more complex than is the case for homogeneous gas-phase chemistry. At this time, no such general purpose electrochemistry software is available to assist the research and development of critical electrochemistry applications and technologies.

The proposed approach builds on the Cantera framework, which has already implemented significant electrochemistry capabilities. The following development directions are likely required to adequately describe needed levels of electrochemical generality and complexity:

- Implement data structures and a computational framework for self-consistent thermodynamic properties required for participating phases and species including condenses solids and liquids, ions, and surface adsorbates.
- Develop general electrochemical kinetic modeling tools within the Marcus theory.
- Develop and implement computational functions and user input syntaxes to accommodate kinetics as expressed in Butler–Volmer form. The inherent limitations and restrictions associated with Butler–Volmer formulations notwithstanding, the vast majority of published literature uses Butler–Volmer charge-transfer kinetics.
- Build out thermodynamically consistent transport software functions which incorporate thermodynamics based on fundamental species parameters and species interactions.
- The software must be designed to accept material properties (e.g., thermodynamic and transport parameters) from other independent models or experiments. For example, needed thermodynamic properties may be developed from atomistic-scale and ab-initio simulations.

4. Numerics

Team: **Niemeyer**, Speth

4.1. Overview and objectives

Cantera's development to date has focused more on functionality than computational performance, due to a lack of dedicated support; as a result, for some problems its performance lags behind what users may expect, particularly compared with that of commercial software. There is thus an opportunity to greatly improve the software's usability and adoption by leveraging advanced numerical routines for increased computational efficiency.

Currently, Cantera can take advantage of multicore parallelism (on a single workstation or computing node) by using some parallelized versions of core linear algebra routines. This provides a performance benefit when using larger kinetic models, by speeding up linear system factorization and solution steps. However, performance can be improved significantly by taking advantage of system sparsity—or reformulating problems to achieve sparsity—and by specifically parallelizing computationally expensive operations. In addition, Cantera has yet to leverage recent advances in hardware acceleration using graphics processing units (GPUs) and similar devices such as Intel Xeon Phis; dedicated development targeting these platforms could enable order-of-magnitude improvement in performance for some applications.

This project thrust will focus on improving computational performance of Cantera's existing and proposed capabilities, by both improving the efficiency of solution procedures and modifying physical representations to enable better performance. This will be achieved through automatic differentiation, integrator improvements, parallelization, and hardware acceleration.

4.2. Automatic differentiation

Most of the computationally-expensive routines in Cantera involve linear algebra; both the zero-dimensional unsteady reactor network and one-dimensional flame applications require linear system factorization and solution steps of large Jacobian matrices, which grow in size with the kinetic model. In theory, many or most of these systems are sparse—for example, few species in large models interact directly—but currently Cantera estimates Jacobians numerically using finite differences. As a result, the Jacobian structure is not known a priori, and dense system routines are used. For example, the CVODE integrator used from the Sundials suite of solvers [44, 45] relies on first-order forward finite differences to evaluate the Jacobian matrix of the kinetic system.

Jacobian matrix sparsity can be obtained by evaluating the matrices analytically through automatic differentiation, rather than the current numerical evaluation. This will also increase performance, since the cost of evaluating Jacobians analytically scales linearly with the number of species while numerical evaluation scales quadratically with species [46]. Automatic differentiation may be achieved by using existing tools such as Adept, a C++ library [47] based on expression templates, or through pyJac, a Cantera-based Python software package for generating analytical Jacobian matrices for chemical kinetics [48, 49]. (The latter focuses specifically on the chemical kinetics ODE system, while the former may be used more generally.)

However, as Schwer et al. [50] pointed out, the typical temperature and mass fraction-based kinetic system for an adiabatic, constant-pressure reactor leads to a fully dense Jacobian matrix. Thus, to expose sparsity in the species evolution equations, we will develop alternate system formulations based on, e.g., species concentration or mass (as opposed to species mass fraction or mole fraction). [1]

4.3. Integrator improvements

Cantera currently relies on the CVODE integrator from the Sundials suite of solvers [45] to advance reactor networks. While CVODE is a robust solver capable of handling the sometimes-extreme stiffness of chemical kinetics ODEs, its computational complexity and cost may not be necessary for all problems at all times. As Niemeyer and Sung showed, depending on the level of stiffness in the chemical kinetics system, even explicit methods can outperform typical implicit integrators [51].

We propose to incorporate additional ODE integrators with Cantera to improve performance. The selection of solvers to be explored will be based on those included in the acceleRInt library [52] and provided by the Julia language's differential equations solver [53];

[1] Cantera previously exposed an alternate set of governing equations to improve computational performance in the ideal gas constant pressure/volume reactors.

particularly interesting methods include Tsitouras's explicit 5/4 Runge–Kutta method [54, 55], stabilized explicit Runge–Kutta–Chebyshev methods [51], the RODAS4 4th-order Rosenbrock method [56, 53], and the RADAU IIA implicit Runge–Kutta method [57]. The now-available analytical Jacobian matrix enables or supports some of these methods, in particular the Rosenbrock method which can outperform the backwards differentiation formula method that CVODE uses [56]. We will provide these additional integrators as options for a user to select, and also implement dynamic algorithm-switching based on stiffness estimates. The latter feature will determine when an explicit algorithm is unstable or too expensive (due to extremely small required timestep sizes) and switch to a more-efficient implicit algorithm; estimates of stiffness will be based on the JULIA `DifferentialEquations.jl` package [53] and ongoing research.

4.4. Improved parallelization

Currently, CANTERA can take advantage of multicore parallelism for certain operations when built with a parallel BLAS/LAPACK implementation, such as Intel's MKL library or OPENBLAS. We will extend this to OpenMP-based parallelization for evaluating reaction and species rates, thermochemical properties, and transport properties (particularly the multicomponent diffusion model). For individual reactor problem (e.g., autoignition), this will particularly benefit larger kinetic models (paired with the existing parallelized linear algebra routines). For one-dimensional flame problems, we will explore parallelizing either (a) over grid points, with one CPU thread/core handling rate/property evaluations for an individual grid point; or (b) over all evaluations combined with grid points, to further saturate the processors with tasks, particularly at lower grid resolutions. It is not clear at this point which strategy will lead to higher performance, so we will investigate both directions.

4.5. Leverage advances in hardware acceleration

Currently, CANTERA is not able to benefit from the advances in computational performance offered by modern accelerators such as GPUs and Intel Xeon Phi processors. We will extend CANTERA to exploit these platforms, inspired by the work of Shi et al. [58], who first used GPU-based linear algebra routines and kinetic rate evaluations to accelerate CHEMKIN-based autoignition problems. They showed an *order of magnitude* improvement in performance with larger kinetic models, and we expect similar improvement in performance—possibly more for one-dimensional flame problems—with CANTERA.

We will integrate the open-source MAGMA library [59–61] with CANTERA to enable straightforward hardware acceleration; MAGMA provides linear algebra routines for heterogeneous and hybrid architectures, i.e., systems with combinations of CPUs, GPUs, Intel Xeon Phis, and similar processors. Currently, the MAGMA project does not provide routines for banded matrices, which CANTERA relies on for one-dimensional flame problems; we will develop the appropriate routines for CANTERA, and contribute them back to MAGMA.

4.6. Coupling with CFD

Finally, while most CANTERA users will work within its ecosystem, many users—particularly in industry or working on more applied problems—perform multidimensional computational fluid dynamics (CFD) simulations of reacting flows, including combustion, electrochemistry, and catalysis. These users need a straightforward way to integrate CANTERA's capabilities into complex modeling software (beyond CANTERA's scope), including the existing gas-phase chemistry and proposed heterogeneous chemistry models.

Currently, CANTERA *can* be coupled CFD software, but two obstacles prevent adoption: (1) there is very limited documentation and no examples showing how to do this, and (2) the complete CANTERA C++ framework must be built and linked with the software. We will work to solve both issues, first by exposing a lightweight C++/FORTRAN interface that avoids building unnecessary and complex components such as, the one-dimensional flame solvers. Then, we will document how to integrate CANTERA with CFD software, culminating in an example using the popular, open-source CFD platform OPENFOAM [62, 63].

Users of CANTERA coupled with multidimensional CFD applications will also benefit from most the performance improvements discussed previously in this section, since these applications will treat rate evaluations and source term integrations (e.g.) as "black boxes" within CANTERA. Furthermore, we will integrate additional techniques to reduce cost such as dynamic/adaptive chemistry [64], which automatically reduce the size of kinetic models based on the local thermochemical state.

Software Management and Integration

5. Extensibility and Interoperability

Team: **Speth**, Weber, Niemeyer, Goldsmith, West, Jackson

5.1. Overview and objectives

Cantera has so-far been successful in applications where it forms the top of software stack, i.e., where users interact directly with its models and algorithms to solve problems and generate results, as in the case of the homogeneous reactor networks and one-dimensional flame models. However, Cantera is currently less widely-used in cases where it serves as a middle layer, e.g., as a chemical source term evaluator embedded in a CFD code, and in cases where the built-in physics models need to be extended by the user, e.g., particular formulations of non-ideal equations of state. The improvements described in this section aim to make Cantera more suitable for these purposes, by introducing an innovative, flexible interface for users to implement new models, and a new data file format for input and output that would improve interoperability with other software.

5.2. High-level interface for model implementation

Currently, all of the models present in Cantera, representing everything from species thermodynamic properties to reaction rates to reactor governing equations, are implemented in C++. While this is a computationally-efficient choice, and while these capabilities are available to be used from other Cantera interfaces (Python, Matlab, and Fortran), new models or extensions to the existing ones can currently only be implemented in C++. Implementing models in C++ can represent a significant barrier to scientific users who are not necessarily expert software engineers, as well as being time-consuming even for those who are. This barrier discourages exploratory model implementations, which are valuable in many cases where those new models need to be evaluated and refined within the context of other Cantera capabilities, e.g., a new thermodynamic model which must be evaluated within the context of kinetic and transport calculations. Here, a method which would allow new models to be implemented in a high-level language is described, reducing this barrier.

The proposed software development effort is the addition of a new interface layer to the Cantera Python module, which would allow users to implement new thermodynamic, reaction rate, transport property, chemical reactor, and other models as Python classes. The interface layer would then allow these user-defined classes to be called by existing Cantera classes, e.g., a new rate expression implemented in Python could be used in a 1D flame simulation using the current C++ solver for these problems. The interface layer would be implemented using the method already used in Cantera to implement "callback" functions in the 1D flame module. For each interface class, the underlying C++ class used would track which methods had custom implementations provided by the user, and use the implementations from the base C++ class for any other methods. This approach would allow the user to provide only the minimal set of methods needed to implement a particular model and avoid incurring the overhead of calling the slower Python methods except where necessary.

Compared to C++, Python is a more accessible, less error-prone language for modelers to use, and is well-suited for rapid prototyping. Since Python code does not need to be explicitly compiled as part of a library, users can make changes to a Python-implemented class and test them without recompiling Cantera. In addition, users could share their implementations of a model easily as standalone modules, in contrast to modifications to the C++ code which require users to distribute and compile modified versions of Cantera. The value of this flexibility increases further when considering the normally daunting task of merging changes from multiple modified versions of the source code.

The approach described here makes an explicit trade-off between speed and accessibility. Python-implemented capabilities which prove to be valuable in the longer term, but introduce an undesirable performance penalty, can be later converted to C++. Computationally-efficient C++ implementations of a model frequently involve transformations which are not trivial to relate back to a straightforward mathematical description of the model. For example, the Arrhenius rate expression $AT^b e^{-E/RT}$ is actually implemented in Cantera as `A * exp(b * logT - EoverR * recipT)`. More complex formulas can differ even more from their mathematical representations. To maintain a clear link between the original model and the software, Python implementations which follow to the more natural formulation can be retained as "reference implementations" when it is eventually desired to implement a higher-performance version in C++. Such reference implementations would be valuable tools in diagnos-

ing errors which can be introduced when attempting to implement high-performance versions of a model. An intermediate level of performance and accessibility could be achieved with user-developed models implemented in Cython [65], an optimizing compiler for Python which allows users to generate faster code by selectively applying type declarations to performance-sensitive code.

The new interface layer described here will support the development of the enhanced models which are described in other sections of this proposal, e.g., generalizations of heterogeneous reaction rates, representation of aerosols in homogeneous reactors, etc. Furthermore, developing these capabilities simultaneously with the implementations of these models will ensure that this interface layer will provide comprehensive access to the underlying C++ features that need to be accessed in order to implement capabilities from Python.

5.3. Enhanced data exchange capabilities

Cantera currently supports two separate input file formats: XML and CTI. XML files are machine-readable, but are verbose and difficult for humans to read or write. CTI files, on the other hand, are human-readable and -writeable, but are essentially Python files which must be converted to XML format for efficient machine parsing. Moreover, metadata in CTI files are currently entered as unstructured Python comments (i.e., preceded by a hash #), which means that this information is not accessible from within Cantera, and cannot be utilized by relevant extension software. Interoperability between Cantera and other software packages would therefore greatly benefit from a more flexible data exchange file format in Cantera.

Part of the proposed effort will develop and implement a new, extensible file format to replace CTI and XML input and output files, based on the YAML format. Unlike the XML and CTI formats, YAML documents are both machine- and human-readable. YAML documents are constructed using a few general-purpose data structures, e.g., arrays and key–value mappings, that provide a consistent approach for structuring data and make it easy to preserve metadata that should go along with an input file. Cantera data structures will be updated to preserve input data which is not directly used by Cantera, but may be used by user applications which utilize Cantera, e.g., molecular structures, and to make this data accessible across different Cantera interfaces. These capabilities will make it easier to build tools which extend Cantera's capabilities, since these tools will be able to simply extend Cantera's input and output data, without the complexity of introducing data structures of their own which parallel those in Cantera. In particular, this approach to handling input data will make such data available to user-defined model classes, as described in the previous section. Additionally, using the standard YAML format will simplify the implementation of tools which generate input files from other formats, such as the existing tool for converting Chemkin-format mechanism files.

YAML is a standard file format, which can be read in many programming languages (C++, Python, Java, Javascript, and more), which makes it easier for other tools to work with Cantera. A generic interface for serializing objects using the YAML format will be added to Cantera, applicable to both built-in Cantera object types (e.g., species and mixture definitions) as well as user-defined types which extend Cantera's capabilities. Cantera-generated YAML files will be restricted to a subset of YAML which can easily be transformed into JSON, for which libraries are available for an even wider range of languages, including Matlab and Fortran. Among other capabilities, this serialization interface will enable the generation of new input files from within Cantera, and allow mechanism generation and reduction codes to be implemented on top of Cantera's built-in functionality.

6. Software Quality: Usability and Maintainability

Team: **Weber**, Speth, Niemeyer, DeCaluwe

6.1. Overview and objectives

Software quality is a broad topic, encompassing elements of *software correctness*, *user satisfaction*, and *suitability for purpose*. The International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) have developed the ISO/IEC 25010:2011 standard to define software quality as understood by the industry [66]. This standard defines software quality in terms of 8 characteristics. Several of these characteristics have been implicitly discussed in previous sections, especially relating to the *functional suitability* and the *performance efficiency* of Cantera. In this section, we will focus on the usability and the maintainability of Cantera.

6.1.1. Usability

While software usability perhaps lacks the gravitas of some of the aforementioned scientific capability improvements, ensuring software quality is essential to support high-impact scientific research. Without software that is easy to install, use, and extend, the likelihood that the improvements in the preceding sections are leveraged for scientific breakthroughs drops considerably. Therefore, Cantera's scientific capabilities must proceed in parallel with improvements to the software's usability.

It is well recognized that open-source software (OSS) has a reputation for being more difficult to use than commercial, closed-source software. This tendency has been studied by several authors [67–70], with the general conclusion that since many OSS projects start with a small group of user-developers satisfying a particular unmet need, making the software "useable" for people other than the original developers has traditionally taken a back seat. Nonetheless, Cantera is (and will be) used by a wide range of users, from inexperienced undergraduate students to world-leading domain experts; thus, the usability of Cantera in a wide range of conditions is paramount to its success. Moreover, we note that software with poor usability (frustrating installation, unclear documentation, and poor backward compatibility) will have little impact in scientific research, no matter how many advanced capabilities it enables, because users will give up trying before they get to use the advanced capabilities.

The ISO/IEC 25010:2011 standard formally breaks software usability into six sub-characteristics [66]: 1. Appropriateness recognizability; 2. Learnability; 3. Operability; 4. User error protection; 5. User interface aesthetics; and 6. Accessibility. The Cantera development team has put considerable effort to improve the usability of the software. Primarily, we have focused on operability by developing several interfaces to to the core library in languages that are recognized as being "easier to use" than the C++ used for the core (namely, interfaces are available for the Python and Matlab environments). We have also partially addressed the learnability aspect by having extensive documentation with detailed examples available for users. More recently, we have begun hosting workshops at the US meetings of the Combustion Institute, which address both the "appropriateness recognizability" and "learnability" aspects. Attendance at these workshops has been between 15–60 attendees of the conference, demonstrating the impact these kinds of events can have on users' ability to learn and operate Cantera.

However, as we discuss in the following sections, we have several ideas to further improve several of these aspects as they relate to Cantera and its users. Many of these ideas are driven by user feedback, both implicit (judged by repeated, similar, questions posted to our mailing list or asked at the workshops), and explicit (judged by specific feature or change requests from users).

6.1.2. Maintainability

Another important part of ensuring that Cantera can meet the needs of its community is the maintainability of the software. The ISO/IEC 25010:2011 standard defines software maintainability as the "degree of effectiveness and efficiency with which a product or system can be modified by the intended maintainers." The previous sections have discussed many new areas where we envision Cantera to be extended; actually implementing these ideas requires Cantera to be effectively able to be modified.

Similar to usability, the ISO/IEC 25010:2011 standard breaks software maintainability into five components [66]: 1. Modularity; 2. Reusability; 3. Analysability; 4. Modifiability; and 5. Testability. One of the current strengths of Cantera is that it already incorporates several of these characteristics, particularly the modularity, reusability, and testability aspects. Cantera was designed from the beginning to be modular and allow for significant reusability of the components. Moreover, Cantera has an extensive test suite that ensures new changes to the code don't break older functionality and that results from the code are correct.

Nonetheless, there are several areas where the maintainability of Cantera should be improved. These primarily focus around ensuring that users do not have to be expert programmers to add functionality to Cantera (c.f., Sec. 5), replacing areas of the code that are difficult to modify and maintain, and automating as much as possible.

6.2. Usability and Maintainability Software Needs

6.2.1. Documentation Improvements

This topic is related to all six of the subcharacteristics of the "usability" definition in Section 6.1.1. Cantera's current documentation contains a thorough description

of our API and a good set of examples for users to peruse. However, there are several issues with how this information is presented, which limit its usability. The landing page (`http://cantera.org`) is essentially a list of links to pages deeper in the documentation. There is no distinction between links to API documentation, links to examples, or links to other high-level material (physical modeling descriptions, etc.). This format makes it difficult for both new and experienced users to find required information. For new users looking for a few helpful hints to get started, the format is simply intimidating. Moreover, because information on the desired class or method may be buried behind two or three links, it is difficult for even experienced users to find the particular method or class they want. Finally, the overall theme is outdated and in need of a refresh.

In addition, the method of generating the documentation inhibits our ability to keep the documentation up-to-date. At present, Cantera's user documentation (examples, physical model descriptions, compilation/installation instructions, etc.) and API documentation (detailed descriptions of the specific functions in Cantera) are collected in the same repository. This arrangement has several flaws, the most glaring of which is that changes to the user documentation must be committed to the main Cantera code repository, coupling the generation of the user documentation and the API documentation. This limits our ability to release updates to the user documentation because it requires copying all user documentation changes to the release maintenance branch, resulting in duplicated commits and builds.

This portion of the proposal will benefit the community by providing easier access to the information that users need, covering a range of activities from installation to basic use to advanced applications. It will make it easier for new users to find tutorial material or examples, and it will make it easier for advanced users and developers to find detailed documentation. This will make using Cantera easier for current users, and also help encourage adoption by new users, broadening Cantera's reach. In addition, the proposed changes will make it easier for developers to maintain the documentation and provide timely updates or add new material. Finally, improving the "look-and-feel" of the website will improve the professional appearance of the project and make it more appealing to new users.

6.2.2. User Interface Improvements

Cantera's core functionality can be accessed either by coding directly in C++ or via a number of interface applications, including Python, Matlab, and Fortran. While the diverse array of access points ultimately benefits the user community, it is challenging from a software maintenance standpoint. Each interface language essentially acts as a new software dependency, and the need to keep the Cantera code up-to-date with new releases of each interface places additional burden on the development team. Below, we outline the development priorities for four software interfaces: each of the current Cantera interfaces, plus Cantera's graphical user interface 'MixMaster.'

Python. At present, the Python interface is relatively well-developed and stable. Recently, we have added new capabilities to attempt automatic solution of simple 1-D flame cases. Aside from the examples discussed in Sec. 5, there are other additions we would like to make to the Python interface to improve usability. For instance, by default, Cantera accepts and returns values in SI units (Pa, K, kg, s, etc.) with the exception of using kmol instead of mol. However, many users work on problems in some other set of customary units, whether the cgs system, the English Engineering system, or others. For these users, specifying inputs to or working with outputs from Cantera results in significantly more arithmetic, increasing the chances the user introduces a mistake in their process. Thus, we envision adding an option to the Python interface that uses a standard unit conversion library to allow the user to easily accomplish these conversions. Ideally, this option would allow the user to specify whatever units they would like as an input to a function (provided the quantities are dimensionally consistent) and the relevant output quantity would be returned in a form that would allow automatic conversion to the user's desired unit system.

C/Fortran. We envision the C and Fortran interfaces as primarily an intermediate layer between the underlying Cantera functionality and other higher-level programs, such as CFD solvers. At present, these interfaces are capable of this task; however, they are severely underdocumented and very few examples exist for how to integrate these interfaces with other programs (see also Sec. 4.6). We would like to ensure that users who want to use Cantera in downstream codes have the resources available so that they can do that in a straightforward manner.

MATLAB. MATLAB's ubiquity in undergraduate STEM programs means that it is many students' first software language. MATLAB is easy to learn and has significant user support in the form of help files, online documentation, and undergraduate instruction. Maintaining a MATLAB software interface to CANTERA functionality therefore makes the software accessible to a wide array of users, ranging from undergraduates to professors.

That said, the current CANTERA–MATLAB interface is restrictive and cumbersome, driven primarily by the fact that all function calls from MATLAB to CANTERA's C++ core are linked through MATLAB's 'Mex' interface, which requires all calls to the CANTERA library to be routed through a single function. The resulting CANTERA–MATLAB interface is therefore difficult for the average user to understand and debug, and writing the interface code for new features implemented in CANTERA's C++ core is tedious and error-prone, leading the MATLAB toolbox to fall behind in functionality.

However, MATLAB currently supports several alternative methods for calling external code, which would make it easier to connect MATLAB to CANTERA's underlying functionality. We therefore propose to revamp CANTERA's MATLAB toolbox using one of three possible pathways. One pathway involves linking calls in MATLAB directly to the CANTERA C interface. This method would be computationally efficient, but may limit the degree of integration possible. The second pathway utilizes MATLAB's ability to interact with PYTHON modules, and would involve the MATLAB toolbox calling CANTERA's PYTHON module, which in turn interfaces directly with the underlying C++ functionality. This is advantageous in that examples of similar MATLAB–PYTHON interfacing are quite common and can be used as a template for this work. Furthermore, such an approach reduces the maintenance workload for the CANTERA developers, by reducing duplicate coding tasks: functionality implemented in the PYTHON module is automatically accessible via MATLAB. However, this approach requires any user interested solely in the MATLAB toolbox to also install the PYTHON module. Furthermore, it is not clear whether such an indirect method of calling CANTERA's C++ code carries a computational expense (i.e., codes run slower). Finally, a third approach is to continue using the Mex interface, but replace the manually-written wrapper functions in C++ and MATLAB with a code generation approach where these functions are automatically generated as part of the build process, eliminating the repetitive coding tasks that lead to the MATLAB interface falling behind in functionality. As part of the proposed work, we will first evaluate the strengths and drawbacks of each approach on a limited set of test functions, before deciding on a final approach.

*GUI interface/*MIXMASTER. Graphical user interfaces (GUIs) provide an intuitive, visually accessible interface to the underlying software capabilities. While advanced users will no doubt need to access CANTERA functionality through coded software commands, access to basic CANTERA functions through a GUI can reduce barriers to entry for a broad class of users who are intimidated by command-line or compiled software or otherwise prefer a visual approach. CANTERA originally included a GUI, called MIXMASTER, but it has not been updated in several years and is essentially non-functional. As part of the upgrade to CANTERA's user interface, then, we propose to restore MIXMASTER's functionality and update the module's documentation to include examples of its capabilities, to include in outreach materials for new CANTERA users. An easy-to-understand, highly-functional MIXMASTER GUI will a key component to extending CANTERA's reach to a include the broader range of researchers who would benefit from chemical kinetic software capabilities, but who are intimidated by traditional software interfaces.

Aside from access to basic CANTERA features, another important use case for GUIs is visualizing chemical kinetic information or simulation output. For instance, Figure 7a shows how the temperature output from a 1-D simulation can be visualized interactively, while Figure 7b shows a reaction path analysis, following the elemental flux of H through reactions involving OH. Aside from these functions, users may be interested in visualizing rates of reactions or thermodynamic properties. This is particularly valuable in light of recent studies showing that many kinetic mechanisms contain reaction rates that are unphysical [71]. In the work of Chen et al. [71], they propose "that computational tools should be made available for authors to conduct the same rate coefficient screening," a role that MIXMASTER would be well equipped to handle.

Finally, as discussed in Section 5.3, the CANTERA input file format for chemistry is a format specific to CANTERA that is not directly compatible with legacy CHEMKIN input files. CANTERA offers a conversion utility from CHEMKIN to CTI format, but unfortunately, many CHEMKIN files contain formatting errors, duplicate thermodynamic/transport/species entries, thermodynamic/transport entries for species not in the species list, and other errors where the correct conversion behavior is unclear. In many cases, users are not sure where the error

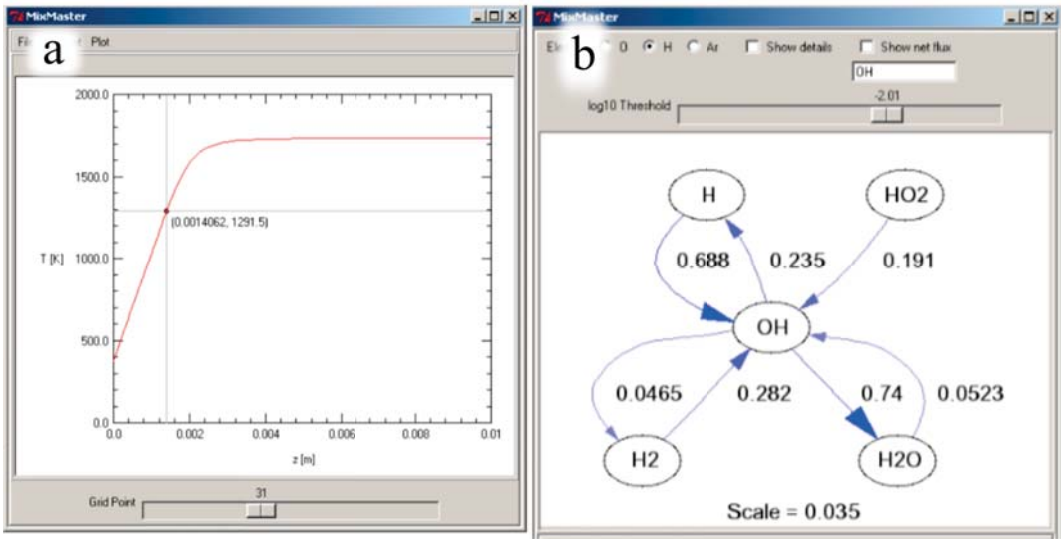

Figure 7: Demonstration of MixMaster functionality, including: a) Post-processing of simulation results. Here, a slider is used to interrogate the 1-D flame simulation temperature as a function of axial length z; b) Reaction path analysis. Details such as contributions from specific reactions and reactants are displayed by clicking on the figure. Part of the proposed work will restore and upgrade MixMaster's functionality.

in the file is located and are unable to fix the problem on their own. This is another case where a GUI could help users identify and correct issues with input files independently, a role that MixMaster is well suited for.

6.2.3. Installers

Another important aspect of usability is installation of the software. For many platforms, we have binary installer packages available that allow the users to install Cantera with little effort; however, some other common platforms are missing or somewhat more difficult to use. For instance, we do not offer a complete installer package for RHEL/CentOS, which are commonly used operating systems on HPC clusters. To ensure the maintainability of this aspect, and streamline the process of releasing new versions of Cantera, we envision a long-term, low-maintenance automated solution to build binary packages for all of the platforms of interest to the project. We already build several artifacts from a new release on free-for-open-source Continuous Integration platforms such as TravisCI [72] and Appveyor [73]. These platforms run automated build scripts when a new version of Cantera is released, and can significantly ease the burden on the maintainers. However, even automated build systems require continuous maintenance efforts in order to keep up with updated versions of operating systems, compilers, build tools, and libraries on which the process depends.

7. Broader Impacts: Sustainable Software Development

While the proposed efforts will provide unprecedented support for the development of open-source chemical kinetics software, it is also incumbent we develop a sustainable long-term model for software development in Cantera. Some of the items described above will contribute to long-term sustainability, particularly those items in Sections 5 and 6 that streamline processes related to software maintenance and documentation, and those that lower the entry barrier for developers wishing to contribute to Cantera. However, there must also be a concurrent emphasis on developing the software's leadership structure and revenue streams to extend and support Cantera well beyond the timeline of the current proposed work.

This proposal comes at an exciting time in Cantera's development. The past 1.5 years have seen Cantera formalize its leadership structure and establish a Fiscal Sponsorship Agreement with NumFOCUS, a non-profit devoted to the support of open-source software. Both of these developments have enabled a more strategic approach to Cantera development and outreach, and present new opportunities for funding improvements to Cantera, as described below. In the following section, we will describe ways in which the present funding proposal will be leveraged to develop sustainable software development practices in the Cantera organization. This

includes establishing a broader leadership team, the definition of operational norms to govern and streamline future Cantera developments, and the development of new sources of income to put Cantera on a sustainable financial footing that will pay dividends to the community, down the road. This section demonstrates how we intend to leverage the current proposed funding to establish relationships and build the infrastructure to make Cantera sustainable, long-term.

7.1. Leadership Structure

For most of its history, Cantera development has been almost entirely ad hoc and has relied on the spontaneous contributions of individual volunteers. While Cantera's success can be attributed largely to the incredible dedication of these volunteers, there is an opportunity to extend Cantera's reach and formalize institutional knowledge by implementing a more formal leadership structure, identifying strategic development priorities, and defining operational norms for how the software is managed. Over the past 1.5 years, Cantera has formalized its leadership structure by identifying a Steering Committee [74], composed of six members from the current team (in alphabetical order: DeCaluwe, Goldsmith, Niemeyer, Speth, Weber, and West). The Steering Committee has established both formal and informal channels for internal communication including an annual web-conference and real-time chat system.

The Steering Committee is responsible for managing day-to-day Cantera development—user support, documentation, developing and managing improvements to the codebase, etc. Guaranteeing that all code merged into Cantera is of high quality typically requires significant oversight and multiple rounds of edits on an author's initial submission, and direclty supporting this work will have significant impact on the quality and sustainability of Cantera's future development. The Steering Committee is also responsible for identifying and pursuing near-term strategic opportunities for code development, software support, and outreach. Outreach has been a particular focus in the last year, primarily by hosting three Cantera workshops in conjunction with Technical Meetings of the Combustion Institute (2017 US National Combustion Meeting, 2018 Eastern States Section of the Combustion Institute Technical Meeting, and 2018 Western States Section of the Combustion Institute Technical Meeting; see Figure 8).

Another important role for the Steering Committee is to define operational norms around how developers

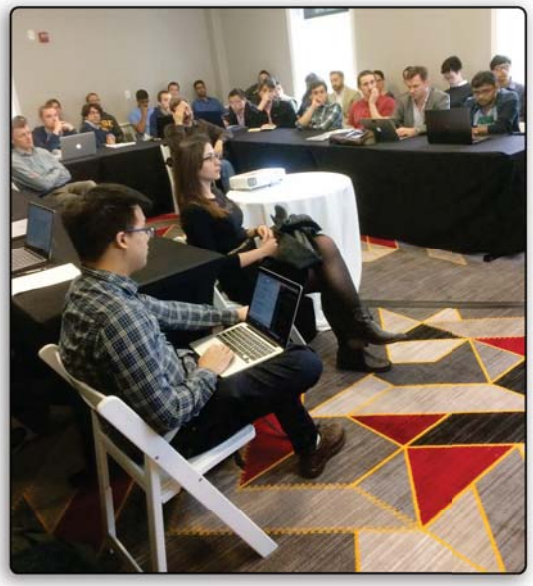

Figure 8: Cantera Workshop held in conjunction with the National Combustion meeting in Washington, DC in April 2017. The workshop was organized and facilitated by Drs. Raymond Speth, Steven DeCaluwe, and Bryan Weber.

interact with the software. This necessarily includes developing requirements for code to be included in Cantera, in terms of relevance to the user community and quality of the code itself. For instance, the Steering Committee has created suggestions for code style that developers should follow for their code to be included in Cantera [75]. Moreover, the Steering Committee should define a standard for when a feature should be included in Cantera. In this sense, the Steering Committee must balance the freedom of the community to add new features to Cantera with the requirement for that feature to be useful and for it to be supported over the life of the software. Further defining such operational norms will be a near-term objective of the current proposed work.

In addition to the Steering Committee, the proposed project will establish a Cantera Advisory Board to supplement the leadership provided by the Steering Committee. The Advisory Board's job will be to work with the Steering Committee to:

- Ensure that software developments are relevant and compatible with ongoing adjacent efforts in academia, industry, and national labs.
- Provide feedback on the overall software development and identify strategic opportunities for future developments.

- Identify and develop revenue streams to support successful software development.

As suggested by the above responsibilities, Advisory Board membership will consist of senior leaders in a variety of fields closely tied to chemical kinetics simulation, and should include members from academia, national labs, and industries including CFD companies and others that rely on chemical kinetics software (e.g., engine manufacturers, aerospace companies, etc.). The Advisory Board will be established within the first year of the grant, and standard procedures will be developed to govern how they operate and interact with the Steering Committee.

7.2. NumFOCUS

In February, 2018, Cantera entered into a Fiscal Sponsorship Agreement with NumFOCUS [76, 77] a non-profit organization that provides a stable, independent, and professional home for projects in the open source scientific data stack. NumFOCUS is the home to several prominent open-source data science tools, such as NumPy, Matplotlib, and Pandas, and provides several direct benefits to Cantera:

- Financial account service to support fundraising, an account for centralized Cantera development funds, and mechanisms for accepting donations.
- Opportunities for small, competitive development grants.
- Advising, community-building resources, and outreach support.
- Legal assistance to protect the code's open source status.

Cantera has already leveraged the relationship with NumFOCUS to support development. As part of the the NumFOCUS organization, Cantera will mentor a 2018 Google Summer of Code (GSoC) [78] student, who will spend three months developing a plug flow reactor model with surface chemistry in Cantera. Additionally, NumFOCUS has awarded a grant to Steering Committee members Weber and DeCaluwe to begin upgrading and modernizing Cantera's online documentation during Summer 2018.

By providing a dedicated fundraising channel for Cantera, NumFOCUS also provides a way to combine funding from multiple streams to directly support Cantera development. With this fundraising channel, the support does not have to come entirely from one grant or funding source. Rather, funds can be appropriated toward Cantera development as they become available. Furthermore, routing such funds through NumFOCUS instead of through individual developer's academic institutions helps reinforce the open-source nature of the project, and avoids the perception that any one institution 'owns' Cantera.

7.3. Fiscal Sustainability

To date, most funding for Cantera development has come in the form of grants to individual PIs interested in model development to support a particular project. While this is appropriate and grounds Cantera development in relevant scientific capabilities, there are some necessary aspects of code development that are not well supported by such a model. In particular, given the pressure to meet funding milestones and deadlines, such projects typically do not have sufficient time to fully document their code or develop test suite modules which help assure that developments elsewhere in the code do not 'break' the implemented functionality. As a result, modules added to Cantera sometimes fall into disuse and inoperability if the original developer does not remain directly involved maintaining the code and documentation. Moreover, user support and user interface development do not typically fall under the purview of traditional funding agencies, and as such rarely receive any direct funding.

To support Cantera's long-term fiscal sustainability, we will pursue a hybrid financial model for Cantera. Development of new scientific capabilities will continue to be funded primarily by grants to individual PIs. However, these funds will be supplemented by dedicated Cantera support, routed through NumFOCUS. These funds will be used to support user interface development, integration of new modeling capabilities, code maintenance and documentation, release management, and outreach to the broader Cantera community. Identifying and developing revenue streams for these purposes will be a joint responsibility of the Steering Committee and Advisory Board. Potential sources include, but are not limited to:

- Strategic partnerships with individual CFD and other commercial software companies (e.g., Mathworks, who develop Matlab) to develop tailored interfaces or even Cantera modules for integrating advanced Cantera functionality into CFD packages. This would have the ancillary benefit of simultaneously increasing Cantera's recognition and use in the CFD community.
- Establish a structure for Cantera users to write a subscription fee into grant budgets for enhanced

user support. While the CANTERA software will always be freely-available and open-source, this would provide a way for the broader user community to support the software, with some further tangible benefit to the user.

- Given the popularity of the first three CANTERA workshops offered (attendance has ranged between 15–60 users per workshop), a registration fee would help support outreach activities.
- Finally, a sustainable long-term model would be to enlist corporate sponsors to form a CANTERA foundation, similar to that for PYTHON [79]. This would take a number of years and significant outreach to develop, but if successful, would represent the most secure approach to long-term fiscal sustainability.

Concurrent with the proposed software developments, the executive council formed by the current proposal team will develop the leadership structure and build relationship in each of the aforementioned areas to build a sustainable foundation for CANTERA's long-term development. Through these efforts, we will establish a long-term model to guarantee that the investment made by the current proposal will be leveraged to build next-generation chemical kinetic tools that have broad-ranging scientific impacts for many years to come.

8. References

[1] D.-Y. Peng and D.B. Robinson. A new two-constant equation of state. *Ind. & Eng. Chem. Fund.*, 15(1):59–64, 1976.

[2] G. Kogekar, C. Karakaya, G.J. Liskovich, M.A. Oehlschlaeger, S.C. DeCaluwe, and R.J. Kee. Impact of non-ideal behavior on ignition delay and chemical kinetics in high-pressure shock tube reactors. *Comb. Flame*, 189:1–11, 2018.

[3] M. Mehl, W.J. Pitz, M. Sjöberg, and J.E. Dec. Detailed kinetic modeling of low-temperature heat release for PRF fuels in an HCCI engine. In *SAE 2009 International Powertrains, Fuels and Lubricants Meeting*, Florence, Italy, 2009.

[4] Mehl M., W.J. Pitz, C.K. Westbrook, and H.J. Curran. Kinetic modeling of gasoline surrogate components and mixtures under engine conditions. *Proc. Comb. Inst.*, 33:193–200, 2011.

[5] T. Tanaka, K. Hack, T. Iida, and S. Hara. Application of thermodynamic databases to the evaluation of surface tensions of molten alloys, salt mixtures and oxide mixtures. *ZEITSCHRIFT FUR METALLKUNDE*, 87(5):380–389, MAY 1996.

[6] S.M. Schnurre, J. Grobner, and R. Schmid-Fetzer. Thermodynamics and phase stability in the Si-O system. *J. Non-cryst. Solids*, 336(1):1–25, APR 1 2004.

[7] L. Kaufman and J. Agren. CALPHAD, first and second generation - Birth of the materials genome. *Scripta Mater.*, 70:3–6, JAN 1 2014.

[8] I.H. Bell, J. Wronski, S. Quoilin, and V. Lemort. Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp. *Ind. Eng. Chem. Res.*, 53(6):2498–2508, FEB 12 2014.

[9] Jens K. Norskov, Frank Abild-Pedersen, Felix Studt, and Thomas Bligaard. Density functional theory in surface chemistry and catalysis. *Proceedings of the National Academy OF Sciences of the United States of America*, 108(3):937–943, 2011.

[10] H. Y. Zhu, A. Kromp, A. Leonide, E. Ivers-Tiffee, O. Deutschmann, and R. J. Kee. A model-based interpretation of the influence of anode surface chemistry on solid oxide fuel cell electrochemical impedance spectra. *Journal Of The Electrochemical Society*, 159(7):F255–F266, 2012. Times Cited: 0 Zhu, Huayang Kromp, Alexander Leonide, Andre Ivers-Tiffee, Ellen Deutschmann, Olaf Kee, Robert J.

[11] C. Karakaya, H. Zhu, and R.J. Kee. Kinetic modeling of methane dehydroaromatization chemistry on Mo/Zeolite catalysts in packed-bed reactors. *Chem. Engr. Sci.*, 123:474–486, 2015.

[12] B. Kee, C. Karakaya, H. Zhu, S.C. DeCaluwe, and R.J. Kee. The Influence of Hydrogen-Permeable Membranes and Pressure on Methane Dehydroaromatization in Packed-Bed Catalytic Reactors. *Ind. Eng. Chem. Res.*, 56(13):3551–3559, 2017.

[13] Ya-Huei (Cathy) Chin, Corneliu Buda, Matthew Neurock, and Enrique Iglesia. Reactivity of chemisorbed oxygen atoms and their catalytic consequences during ch4-o-2 catalysis on supported pt clusters. *Journal of the American Chemical Society*, 133(40):15958–15978, 2011.

[14] Lynza H. Sprowl, Charles T. Campbell, and Líney Árnadóttir. Hindered translator and hindered rotor models for adsorbates: Partition functions and entropies. *Journal of Physical Chemistry C*, 120(18):9719–9731, 2016.

[15] M. M. Wolf, H. Y. Zhu, W. H. Green, and G. S. Jackson. Kinetic model for polycrystalline pd/pdox in oxidation/reduction cycles. *Applied Catalysis A-General*, 244(2):323–340, 2003.

[16] Qingfan Zhang, Bo Han, Xiaowei Tang, Kevin Heier, Jimmy X. Li, John Hoffman, Minfa Lin, Stephanie L. Britton, Agnes Derecskei-Kovacs, and Hansong Cheng. On the mechanisms of carbon formation reaction on ni(111) surface. *Journal of Physical Chemistry C*, 116(31):16522–16531, 2012.

[17] F. Gelbard and J.H. Seinfeld. Simulation of multicomponent aerosol dynamics. *Journal of Colloid and Interface Science*, 78(2):485–501, December 1980.

[18] M.D. Smooke, C.S. McEnally, L.D. Pfefferle, R.J. Hall, and M.B. Colket. Computational and experimental study of soot formation in a coflow, laminar diffusion flame. *Combustion and Flame*, 117(1):117–139, April 1999.

[19] M.E. Mueller, G. Blanquart, and H. Pitsch. Hybrid Method of Moments for modeling soot formation and growth. *Combustion and Flame*, 156(6):1143–1155, June 2009.

[20] K.S. Kang, Y.S. Meng, J. Breger, C.P. Grey, and G. Ceder. Electrodes with high power and high capacity for rechargeable lithium batteries. *Science*, 311(5763):977–980, 2006.

[21] B. Kang and G. Ceder. Battery materials for ultrafast charging and discharging. *Nature*, 458(7235):190–193, 2009.

[22] S.-W. Kim, D.-H. Seo, X. Ma, G. Ceder, and K. Kang. Electrode Materials for Rechargeable Sodium-Ion Batteries: Potential Alternatives to Current Lithium-Ion Batteries. *Adv. Energy Mats.*, 2(7, SI):710–721, 2012.

[23] A. Jain, S.P. Ong, G. Hautier, W. Chen, S. Richards, W.D. Dacek, S. Cholia, D. Gunter, D. Skinner, G. Ceder, and K.A. Persson. Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *Apl. Mats.*, 1(1), 2013.

[24] Y.P. Ma, M. Doyle, T.F. Fuller, M.M. Doeff, L.C. Dejonghe, and J. Newman. The Measurement of a Complete Set of Transport-properties for a Concentrated Solid Polymer Electrolyte Solution. *J. Electrochem. Soc.*, 142(6):1859–1868, 1995.

[25] K. Angenendt and P. Johansson. Ionic Liquid Based Lithium

Battery Electrolytes: Charge Carriers and Interactions Derived by Density Functional Theory Calculations. *Journal of Physical Chemistry B*, 115(24):7808–7813, JUN 23 2011.

[26] Y. Saito, W. Morimura, R. Kuratani, and S. Nishikawa. Ion Transport in Separator Membranes of Lithium Secondary Batteries. *Journal of Physical Chemistry C*, 119(9):4702–4708, MAR 5 2015.

[27] C. Kupper and W.G. Bessler. Multi-Scale Thermo-Electrochemical Modeling of Performance and Aging of a LiFePO4/Graphite Lithium-Ion Cell. *J. Electrochem. Soc.*, 164(2):A304–A320, 2017.

[28] K.E. Thomas, S.E. Sloop, J.B. Kerr, and J. Newman. Comparison of lithium-polymer cell performance with unity and nonunity transference numbers. *J. Power Sources*, 89(2):132–138, 2000. Solid State Lithium Battery Workshop, TOWSON, MARYLAND, JUL, 1999.

[29] M.M. Doeff, L. Edman, S.E. Sloop, J. Kerr, and L.C. De Jonghe. Transport properties of binary salt polymer electrolytes. *J. Power Sources*, 89(2):227–231, 2000. Solid State Lithium Battery Workshop, TOWSON, MD, JUL, 1999.

[30] J.B. Kerr, S.E. Sloop, G. Liu, Y.B. Han, J. Hou, and S. Wang. From molecular models to system analysis for lithium battery electrolytes. *J. Power Sources*, 110(2):389–400, 2002. Workshop on Development of Advanced Battery Engineering Models, ALEXANDRIA, VIRGINIA, AUG 14-16, 2001.

[31] J. Newman, K.E. Thomas, H. Hafezi, and D.R. Wheeler. Modeling of lithium-ion batteries. *J. Power Sources*, 119(SI):838–843, 2003. 11th International Meeting on Lithium Batteries, MONTEREY, CALIFORNIA, JUN 24-28, 2002.

[32] P. Georen and G. Lindbergh. Characterisation and modelling of the transport properties in lithium battery gel electrolytes - Part I. The binary electrolyte PC/LiClO4. *Eleectrochim. Acta*, 49(21):3497–3505, 2004.

[33] J.H. Shin, W.A. Henderson, and S. Passerini. PEO-based polymer electrolytes with ionic liquids and their use in lithium metal-polymer electrolyte batteries. *J. Electrochem. Soc.*, 152(5):A978–A983, 2005.

[34] L.O. Valoen and J.N. Reimers. Transport properties of LiPF6-based Li-ion battery electrolytes. *J. Electrochem. Soc.*, 152(5):A882–A891, 2005.

[35] A.M. Colclasure and R.J. Kee. Thermodynamically consistent modeling of elementary electrochemistry in lithium-ion batteries. *Electrochim. Acta*, 55(28):8960–8973, dec 2010.

[36] J. P. Neidhardt, D. N. Fronczek, T. Jahnke, T. Danner, B. Horstmann, and W.G Bessler. A Flexible Framework for Modeling Multiple Solid, Liquid and Gaseous Phases in Batteries and Fuel Cells. *J. Electrochem. Soc.*, 159(9):A1528–A1542, 2012.

[37] F. Bella, F. Colo, J. R. Nair, and C. Gerbaldi. Photopolymer Electrolytes for Sustainable, Upscalable, Safe, and Ambient-Temperature Sodium-Ion Secondary Batteries. *Chemsuschem*, 8(21):3668–3676, 2015.

[38] M. Chintapalli, T.N.P. Le, N.R. Venkatesan, N.G. Mackay, A.A. Rojas, J.L. Thelen, X.C. Chen, D. Devaux, and N.P. Balsara. Structure and Ionic Conductivity of Polystyrene-block-poly(ethylene oxide) Electrolytes in the High Salt Concentration Limit. *Macromol.*, 49(5):1770–1780, 2016.

[39] V. Thangadurai, S. Narayanan, and D. Pinzaru. Garnet-type solid-state fast li ion conductors for li batteries: critical review. *Chemical Society Reviews*, 43(13):4714–4727, 2014.

[40] K. Fu, Y. Gong, B. Liu, Y. Zhu, S. Xu, Y. Yao, W. Luo, C. Wang, S.D. Lacey, J. Dai, Y. Chen, Y. Mo, E. Wachsman, and L. Hu. Toward garnet electrolyte-based li metal batteries: An ultrathin, highly effective, artificial solid-state electrolyte/metallic li interface. *Science Advances*, 3(4), APR 2017.

[41] A. Manthiram, X. Yu, and S. Wang. Lithium battery chemistries enabled by solid-state electrolytes. *Nature Reviews Materials*, 2(4), APR 2017.

[42] A.M. Colclasure, K.A. Smith, and R.J. Kee. Modeling detailed chemistry and transport for solid-electrolyte-interface (SEI) films in Li-ion batteries. *Electrochim. Acta*, 58:33–43, 2011.

[43] D. Aurbach, B. Markovsky, I. Weissman, E. Levi, and Y. Ein-Eli. On the correlation between surface chemistry and performance of graphite negative electrodes for Li ion batteries. *Electrochim. Acta*, 45(1-2):67–86, 1999.

[44] S.D. Cohen, A.C. Hindmarsh, and P.F. Dubois. CVODE, a stiff/nonstiff ODE solver in c. *Computers in Physics*, 10(2):138, 1996.

[45] A.C. Hindmarsh, P.N. Brown, K.E. Grant, S.L. Lee, R. Serban, D.E. Shumaker, and C.S. Woodward. SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers. *ACM Transactions on Mathematical Software*, 31(3):363–396, 2005.

[46] Tianfeng Lu and Chung K. Law. Toward accommodating realistic fuel chemistry in large-scale computations. *Progress in Energy and Combustion Science*, 35(2):192–215, April 2009.

[47] R.J. Hogan. Fast reverse-mode automatic differentiation using expression templates in c++. *ACM Transactions on Mathematical Software*, 40(4):1–16, 2014.

[48] Kyle E. Niemeyer, Nicholas J. Curtis, and Chih-Jen Sung. pyJac: Analytical jacobian generator for chemical kinetics. *Computer Physics Communications*, 215:188–203, 2017.

[49] Nicholas J Curtis and Kyle E Niemeyer. pyJac v1.0.6, 2018.

[50] D.A. Schwer, J.E. Tolsma, W.H. Green, and P.I. Barton. On upgrading the numerics in combustion chemistry codes. *Combustion and Flame*, 128(3):270–291, 2002.

[51] K.E. Niemeyer and C.-J. Sung. Accelerating moderately stiff chemical kinetics in reactive-flow simulations using GPUs. *Journal of Computational Physics*, 256:854–871, 2014.

[52] Nicholas J Curtis and Kyle E Niemeyer. accelerint: v1.1-beta, 2017. `https://doi.org/10.5281/zenodo.842845`.

[53] Christopher Rackauckas and Qing Nie. DifferentialEquations.jl – a performant and feature-rich ecosystem for solving differential equations in julia. *Journal of Open Research Software*, 5, May 2017.

[54] Ch. Tsitouras. Explicit runge–kutta pairs appropriate for engineering applications. *Applied Mathematical Modelling*, 26(1):77–88, 2002.

[55] Ch. Tsitouras. Runge–kutta pairs of order 5(4) satisfying only the first column simplifying assumption. *Computers & Mathematics with Applications*, 62(2):770–775, July 2011.

[56] Ernst Hairer and Gerhard Wanner. *Solving Ordinary Differential Equations II*. Springer Berlin Heidelberg, 1996.

[57] Ernst Hairer and Gerhard Wanner. Stiff differential equations solved by radau methods. *Journal of Computational and Applied Mathematics*, 111(1-2):93–111, November 1999.

[58] Y. Shi, W.H. Green, H.-W. Wong, and O.O. Oluwole. Redesigning combustion modeling algorithms for the graphics processing unit (GPU): Chemical kinetic rate evaluation and ordinary differential equation integration. *Combustion and Flame*, 158(5):836–847, 2011.

[59] S. Tomov, J. Dongarra, and M. Baboulin. Towards dense linear algebra for hybrid GPU accelerated manycore systems. *Parallel Computing*, 36(5-6):232–240, June 2010.

[60] S. Tomov, R. Nath, H. Ltaief, and J. Dongarra. Dense linear algebra solvers for multicore with GPU accelerators. In *Proc. of the IEEE IPDPS'10*, pages 1–8, Atlanta, GA, April 19-23 2010. IEEE Computer Society. DOI: 10.1109/IPDPSW.2010.5470941.

[61] J. Dongarra, M. Gates, A. Haidar, J. Kurzak, P. Luszczek, S. Tomov, and I. Yamazaki. Accelerating numerical dense linear algebra calculations with gpus. *Numerical Computations with*

GPUs, pages 1–26, 2014.

[62] H. G. Weller, G. Tabor, H. Jasak, and C. Fureby. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Computers in Physics*, 12(6):620, 1998.

[63] The OpenFOAM Foundation. OpenFOAM v5.0, 2018. `https://openfoam.org`.

[64] Nicholas J. Curtis, Kyle E. Niemeyer, and Chih-Jen Sung. An automated target species selection method for dynamic adaptive chemistry simulations. *Combustion and Flame*, 162(4):1358–1374, April 2015.

[65] S. Behnel, R. Bradshaw, C. Citro, L. Dalcin, D.S. Seljebotn, and K. Smith. Cython: The best of both worlds. *Computing in Science Engineering*, 13(2):31 –39, March 2011.

[66] Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models. ISO Standard ISO/IEC 25010:2011, International Organization for Standardization/International Electrotechnical Commission, Geneva, March 2011.

[67] A. Raza, L.F. Capretz, and F. Ahmed. Improvement of Open Source Software Usability: An Empirical Evaluation from Developers' Perspective. *Advances in Software Engineering*, 2010:1–12, 2010.

[68] D. Nichols and M. Twidale. The Usability of Open Source Software. *First Monday*, 8(1), January 2003.

[69] H. Hedberg, N. Iivari, M. Rajanen, and L. Harjumaa. Assuring quality and usability in open source software development. In *Emerging Trends in FLOSS Research and Development, 2007. FLOSS '07. First International Workshop on*, pages 2–2. IEEE, May 2007.

[70] N. Viorres, P. Xenofon, M. Stavrakis, E. Vlachogiannis, P. Koutsabasis, and J. Darzentas. Major HCI Challenges for Open Source Software Adoption and Development. In *Online Communities and Social Computing*, Lecture Notes in Computer Science, pages 455–464. Springer, Berlin, Heidelberg, July 2007.

[71] D. Chen, K. Wang, and H. Wang. Violation of collision limit in recently published reaction models. *Combustion and Flame*, 186:208–210, December 2017.

[72] Travis-CI. `travis-ci/travis-api`. `https://github.com/travis-ci/travis-api`. Accessed: 13 April 2018.

[73] Appveyor. `appveyor/ci`. `https://github.com/appveyor/ci`. Accessed: 13 April 2018.

[74] Cantera steering committee. `http://cantera.org/docs/sphinx/html/about.html#steering-committee`. Accessed: 16 April 2018.

[75] Contributing to Cantera. `https://github.com/Cantera/cantera/blob/master/CONTRIBUTING.md`. Accessed: 16 April 2018.

[76] NumFOCUS: Open Code = Better Science. `https://www.numfocus.org/`. Accessed: 16 April 2018.

[77] Cantera Joins NumFOCUS Sponsored Projects. `https://www.numfocus.org/blog/cantera-joins-numfocus-sponsored-projects`. Accessed: 16 April 2018.

[78] Google Summer of Code. `https://summerofcode.withgoogle.com/`. Accessed: 16 April 2018.

[79] Python Software Foundation. `https://www.python.org/psf/`. Accessed: 16 April 2018.

Appendix A: Software Planning Roadmap