Energy

**C
O
N
S
E
R
V
A
T
I
O
N**

DOE/AL/94598--1-Add.
(DE95016651)

**REVITALIZE THE U.S. SILICON/FERROSILICON INDUSTRY
THROUGH ENERGY-EFFICIENT TECHNOLOGY**

Final Technical Report--Addendum Furnace Modeling

**By
G. Kris Schwenke
David Woolley
Julian Szekely**

July 1995

Work Performed Under Contract No. DE-FC04-93AL94598

**For
U.S. Department of Energy
Office of Industrial Technologies
Washington, D.C.**

and

**Electric Power Research Institute
Palo Alto, California**

**By
South Carolina Research Authority
N. Charleston, South Carolina**

and

**Massachusetts Institute of Technology
Cambridge, Massachusetts**

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

This report has been reproduced directly from the best available copy.

Available to DOE and DOE contractors from the Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831; prices available from (615) 576-8401.

Available to the public from the U.S. Department of Commerce, Technology Administration, National Technical Information Service, Springfield, VA 22161, (703) 487-4650.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

**REVITALIZE THE U.S. SILICON/FERROSILICON INDUSTRY
THROUGH ENERGY-EFFICIENT TECHNOLOGY**

Final Technical Report--Addendum Furnace/Modeling

July 1995

Work Performed Under Contract Number: DE-FC04-93AL94598

**Prepared for the
U.S. Department of Energy
Under DOE Albuquerque Operations Office
Sponsored by the Office of the Assistant Secretary
for Energy Efficiency and Renewable Energy
Office of Industrial Technologies
Washington, D.C.**

**By
South Carolina Research Authority
N. Charleston, SC 29418**

EXECUTIVE SUMMARY

In response to a campaign run by the South Carolina Research Authority (SCRA) to improve Silicon and Ferro-silicon production efficiency in a direct current, submerged plasma arc furnace, a project was undertaken to mathematically model and analyze the furnace system. A reaction extent model has been developed and useful results obtained. The code can be run with commercially available hardware and software. In addition, two free energy minimization models have been formulated. These may be run by modifying readily available public domain programs.

The approaches outlined here may find several practical uses in industry. First of all, this modeling work will provide insight into the results of the SCRA experimental campaigns. These models may also serve as a basis for the planning of future experiments and the development of more comprehensive analytical techniques. Future modeling work, performed in cooperation with SCRA, will aid in identifying process limitations and improving process performance. Potential benefits include increased metal recovery, better electrical efficiency, and decreased process emissions. The techniques presented here may be also quite useful to the operators of conventional arc furnaces in ferroalloy production, for trouble-shooting and process improvement studies.

The models presented here are largely based on equilibrium considerations, although simple modifications have permitted estimation of the effects of gas release during stoking, and of non-equilibrium reactions in the cooler, upper parts of the furnace. These techniques could be made especially powerful if combined with reaction rate laws, and a dynamic plasma model.

This report is an addendum to U.S. DOE Report No. DOE/AL/94598-1 which described the direct current, plasma arc smelting of silicon and ferrosilicon in a 1.5 MW test facility in Charleston, SC.

The project was conducted by the South Carolina Research Authority, coordinating the interests of the Silicon Technology Competitive Cooperative (STCC), and the Electric Power Research Institute (EPRI). EPRI is a utility-industry organization that sponsors research promoting the efficient use of electricity. EPRI supported the furnace modeling study, the subject of this report. The modeling study was conducted at the Massachusetts Institute of Technology from March 1993 through July 1995. The STCC interest was in the test program. The STCC is a sub-group of The Ferroalloys Association which is composed of member-companies that produce silicon, ferrosilicon, or both.

This is a cost-shared project conducted by South Carolina Research Authority and supported by the U.S. Department of Energy (DOE), Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Industrial Technologies, under DOE Albuquerque Field Office Cooperative Agreement DE-FC04-93AL94598.

TABLE OF CONTENTS

Thermodynamics of the Si-O-C System	1
Formulation and Selected Results of Reaction Extent Models	4
Equilibrium Model - <i>Stage-Wise Equilibrium in all Stages</i>	4
Last Gas Stage Model - <i>Equilibrium in Stages at $T > 1788\text{ K}$</i>	12
Blow-Hole Model - <i>Simulates Gas Escape from Inner Furnace</i>	12
Enthalpy-Limited Model - <i>Reaction Extent Limited by Heat Balance</i>	13
Summary of Numerical Results from Reaction Extent Models	17
Formulation of Free Energy Minimization Models	19
Theory and Limitations of Reaction Extent Models	19
Theory of Free Energy Minimization Models in the Si-O-C System	21
Theory of Optimization	22
Selected Results from SOLGASMIX-Based Programs	26
Discussion	27
Discussion of Reaction Extent Models	27
Discussion of Free Energy Minimization Models	28
Appendices: Reaction Extent Model MATLAB™ Program Code	30
References	57

THERMODYNAMICS OF THE Si-O-C SYSTEM

The pure Silicon-producing furnace is a reactive open system run in a semi-continuous manner. For modeling purposes, it can be assumed to be a continuous process at steady-state. The furnace contains significant amounts of three elements: Silicon, Carbon, and Oxygen. These elements may arrange themselves into a variety of compounds, depending upon local stoichiometry and/or temperature. Due to the strong reducing conditions present, certain species like O_2 and CO_2 are neglected, leaving the following species: SiO_2 , C, Si, SiC, SiO, and CO. Silica, Carbon, Silicon, and Silicon Carbide exist as immiscible condensed phases; Silicon Monoxide and Carbon Monoxide exist as gases.¹ Temperatures range from room temperature to in excess of 2000°C. Since the furnace is open to the air via a fume hood, the total system pressure ($P_{SiO} + P_{CO}$) remains near one atmosphere.

The Gibbs Phase Rule states that a system at equilibrium will possess a number of degrees of freedom, θ , equal to the number of components, m , plus two, minus the number of phases present, ϕ .

$$\theta = m + 2 - \phi \quad [1]$$

For the system described above, $m=3$ (Si, C, O), and one degree of freedom is used in specifying that the pressure is to remain constant at one atmosphere. These conditions result in the following law for the furnace system:

$$\theta = 4 - \phi \quad [2]$$

Four phases (three condensed and a gas) are able to coexist with zero degrees of freedom (the system picks the temperature and SiO and CO partial pressures). Two condensed phases and a gas are able to exist with one degree of freedom. The consequence of this relationship is that there is an equilibrium partial pressure of SiO in the gas above a combination of any two condensed phases at a temperature, T . Finally, a system comprised of a single condensed phase and a gas has two degrees of freedom; it may exist over a range of T and P_{SiO} .

For the four condensed phases considered in the furnace, there are six possible binary combinations, each with its own equilibrium reaction. Other authors have neglected reaction [6] because it has no impact on the system, but it is included for completeness. These combinations and equilibria are summarized in Table 1:

Phases	Reaction	Eqn #
SiO_2, C	$SiO_2 + C = SiO + CO$	[3]
SiO_2, Si	$SiO_2 + Si = 2 SiO$	[4]
SiO_2, SiC	$2 SiO_2 + SiC = 3 SiO + CO$	[5]
C, Si	$Si + CO = C + SiO$	[6]
C, SiC	$SiC + CO = 2 C + SiO$	[7]
Si, SiC	$2 Si + CO = SiC + SiO$	[8]

Table 1: Binary Phase Equilibria

¹ Müller. "Heat and Mass Transfer in the Ferrosilicon Process" in *Scandinavian Journal of Metallurgy*, Vol. 1, 1972, p.146

With three components, it is possible to write three atomic balances, relating the quantities of each of the species. With six species and three balances, only three reactions are linearly independent. Thus, any three of the reactions in Table 1 can be written as linear combinations of any other three. These reactions are chosen to represent the system, not because of any fundamental mechanistic importance, but because of convenience: they involve all possible binary pairs of the six species, and they have low integer stoichiometric coefficients. The system can be analyzed in terms of the reactions in Table 1 or in terms of a totally different set of reactions. So long as the calculation reflects the correct reaction stoichiometry, the results should be identical.

For each reaction above, it is possible to write an expression for the equilibrium constant, K . Using reaction [3] as an example,

$$K = \exp\left(\frac{-\Delta G_{Rxn}}{RT}\right) = \frac{f_{SiO} f_{CO}}{f_{SiO_2} f_C} = \frac{P_{SiO} P_{CO}}{a_{SiO_2} a_C} = P_{SiO}(1 - P_{SiO}) = -P_{SiO}^2 + P_{SiO} \quad [9]$$

where f denotes fugacity, P partial pressure, and a activity. R is the gas constant. The activities of all condensed phases are assumed to be unity. The free energy change of the reaction, ΔG_{Rxn} , can be calculated at the temperature of interest from tabulated thermodynamic data, yielding a quadratic equation for P_{SiO} with implicit temperature dependence. When similar derivations are performed for all six binary phase combinations, an equilibrium diagram, Figure 1, can be constructed. In the Figure, the six reactions in Table 1 are labeled by number. Dashed lines represent unstable equilibria (the two phases assumed present would react to give others), while solid lines represent stable equilibria.

There are two distinct points where three stable equilibria all intersect: 1788 K and 5.369×10^{-3} atm SiO, and 2084 K and 0.6760 atm SiO. These points are termed invariant points, for they

represent the temperatures and pressures picked by the system for the simultaneous existence of four phases (and zero degrees of freedom). The lower of the two corresponds to a stable mixture of SiO₂, C, and SiC, while the upper is SiO₂, Si, and SiC.

As silica and Carbon are fed into the furnace and heated as they descend into an atmosphere of SiO and CO, they follow curve [3] until 1788 K (the first invariant point), at which time Silicon Carbide begins to form. As heating continues, no temperature change is observed until either SiO₂ or C has been entirely consumed. It has been suggested that the only reaction proceeding at this point is the direct formation of SiC from SiO₂ and C, via a combination of 3 x [3]-[5]:²

Since this reaction generates CO but not SiO, it would reduce the SiO partial pressure and cause the system to leave the invariant point. To prevent this, another reaction must also proceed to preserve SiO:CO balance. Reaction [7],

can balance the CO-enriching effect of reaction [10], so the evolution of SiC proceeds by a combination of reactions [10] and [7]. Below, the extents, $e_{10}=0.6787$ moles and $e_7=0.0180$ moles, are calculated at conditions typical of this part of the furnace. Note that at this low SiO pressure, the extent of reaction [7] is small compared to that of [10], but it is important to recognize that the system undergoes two independent reactions. At the other invariant point, P_{SiO} is high enough that the extents are comparable and both reactions are important.

Once either SiO₂ or C has been entirely consumed, the mixture proceeds towards higher temperatures along another binary equilibrium curve. Depending on feed stoichiometry, either the SiO₂-SiC curve [5] or the C-SiC curve [7] will be followed. In a Silicon furnace, the C:SiO₂ ratio will be near 2:1, so C will be limiting, and the mixture will follow the SiO₂-SiC curve.

As the SiO₂-SiC mixture continues to descend into the furnace, reaction [5] dictates the equilibrium composition until the second invariant point is reached at 2084 K. Here, Silicon is formed, according to the reactions:

which is a combination of 2 x [5]-3 x [4], and

In an ideal furnace that provides complete equilibrium from room temperature to 2084 K, the extents of these two simultaneous reactions are such that their sum may be approximated (per mole Si produced) by:

This result is obtained after an iterative solution of several equilibrium stages in the vicinity of this temperature and partial pressure. It represents a simultaneous solution to Silicon, Oxygen, and Carbon atomic balances and the constraint the gas contain the equilibrium SiO partial pressure. Others have taken advantage of the proximity of the invariant point to two-thirds of an atmosphere of SiO partial pressure and written a combination of 3 x [5] - 3 x [4] + [8]:³

² Müller, p. 147.

³ Rosenqvist, Principles of Extractive Metallurgy, 2nd Ed., 1983, p. 379.

It can be seen from this examination of the system thermodynamics that the minimum temperature at which Silicon can be produced in an equilibrium furnace is 2084 K. If the SiO_2 and SiC are allowed to react completely, one phase will disappear, and the system will continue towards higher temperatures along either the SiO_2 -Si curve [4] or the Si-SiC curve [8].

FORMULATION AND SELECTED RESULTS OF REACTION EXTENT MODELS

Equilibrium Model

In the stage-wise equilibrium reaction extent model, the furnace is idealized into a series of vertically stacked stages, with counter-current flow between descending solids and liquids and rising gases, much like a distillation column. This arrangement is approached in practice when the furnace diameter is small compared to its height, or when the electrode occupies a large fraction of the furnace volume. Heat and mass transfer are assumed sufficiently rapid that thermodynamic equilibrium is attained in each stage. This, too, is an idealization, but it is approached in practice at high temperatures and with small particle sizes. No assumptions are made regarding the spatial temperature distribution. It is enough to assume that *some part* of the furnace is at a temperature, T , and that equilibrium is attained in that zone.

In reality, examination of furnaces cooled directly from operation has revealed that several distinct zones are present, and that they are not necessarily vertically oriented. Müller describes zones consisting of "loose charge mix," "sintered charge mix," a "crater crust," and a "dense metallic deposit," distributed around a central gas cavity adjacent to the electrode.⁴ The stage-wise equilibrium reaction extent model can still provide useful insight, so long as gas from the inner cavity passes through an appreciable amount of material before being released into the atmosphere. Two modifications to this model, the "Last Stage Temperature" and the "Blow-Hole" models are also developed below. They rely upon the equilibrium reaction extent model, so it is developed first.

Simply put, the equilibrium reaction extent model calculates the output flowrates from a stage based on the stage temperature and the inlet flowrates. By referring to the appropriate binary equilibrium reaction or invariant point reaction set, reaction extents are calculated to satisfy the equilibrium P_{SiO} at the stage temperature, T . Solid or liquid products are fed as input to the stage below; gaseous products enter the stage above. With an initial guess for the set of furnace flowrates, a solution is found iteratively. The actual program code is written in MATLAB™, a commercially available numerical software package.

For example, consider again the SiO_2 -C regime. If a furnace feed of 1 mole of silica and 2 moles of Carbon enters a stage into which 2 moles of Carbon Monoxide flow, some of the silica and Carbon will react to form enough SiO to generate the equilibrium partial pressure. If the gas phase is assumed ideal, then the partial pressure of SiO is given by:

$$P_{\text{SiO}} = \frac{n_{\text{SiO}} P}{n_{\text{SiO}} + n_{\text{CO}}} \quad [14]$$

⁴ Müller, p. 146.

where n is the number of moles. For reaction [3], n_{SiO} and n_{CO} vary with reaction extent, e , according to:

$$n_{\text{SiO}} = n_{\text{SiO}}^0 + e \quad [15]$$

$$n_{\text{CO}} = n_{\text{CO}}^0 + e \quad [16]$$

where a zero superscript refers to the inlet moles. Upon substitution in [14], this yields:

$$P_{\text{SiO}} = \frac{n_{\text{SiO}}^0 + e}{n_{\text{SiO}}^0 + n_{\text{CO}}^0 + 2e} \quad [17]$$

P_{SiO} is determined by T (for equilibrium between SiO_2 and C), permitting solution for e :

$$e = \frac{n_{\text{SiO}}^0 (P_{\text{SiO}} - 1) + n_{\text{CO}}^0 P_{\text{SiO}}}{1 - 2 P_{\text{SiO}}} \quad [18]$$

The calculated extent is then added or subtracted from the inlet mole numbers, according to reaction stoichiometry, to yield the equilibrium output mole numbers:

$$n_{\text{SiO}_2} = n_{\text{SiO}_2}^0 - e \quad [19]$$

$$n_{\text{C}} = n_{\text{C}}^0 - e \quad [20]$$

$$n_{\text{SiO}} = n_{\text{SiO}}^0 + e \quad [15]$$

$$n_{\text{CO}} = n_{\text{CO}}^0 + e \quad [16]$$

If any of the calculated outlet mole numbers is negative, it is an indication that the system is not able to come to equilibrium with the inputs as specified. In that event, the numbers are recalculated, taking into account whichever reactant is limiting.

For the example above, the calculated extent is a strong function of P_{SiO} , and hence a strong function of T . Some sample values are included in Table 2:

T [K]	e [mol]	SiO_2 [mol]	C [mol]	SiO [mol]	CO [mol]	P_{SiO} [atm]
700	8.15×10^{-34}	≈ 1	≈ 2	≈ 0	≈ 0	≈ 0
1500	1.85×10^{-6}	0.999998	1.999998	1.85×10^{-6}	2.000002	9.27×10^{-7}
1700	0.0010	0.9990	1.9990	0.0010	2.0010	5.24×10^{-4}
1788	0.0109	0.9891	1.9891	0.0109	2.0109	5.37×10^{-3}

Table 2: Temperature Dependence of Reaction [3] Extent

The output from the invariant points is calculated in a similar manner, except that two extents are necessary, so more information is required. This extra information comes from that fact that one of the input condensed phases will be the limiting reactant. As an example, take the same SiO_2 - C feed producing SiC at 1788 K. The two operative reactions are [10] and [7].

They both affect the mole numbers:

$$n_{\text{SiO}_2} = n_{\text{SiO}_2}^0 - e_{10} \quad [21]$$

$$n_{\text{C}} = n_{\text{C}}^0 - 3e_{10} + 2e_7 \quad [22]$$

$$n_{\text{SiC}} = n_{\text{SiC}}^0 + e_{10} - e_7 \quad [23]$$

$$n_{\text{SiO}} = n_{\text{SiO}}^0 + e_7 \quad [24]$$

$$n_{\text{CO}} = n_{\text{CO}}^0 + 2e_{10} - e_7 \quad [25]$$

If Carbon is limiting, $n_{\text{C}} = 0$ so

$$e_7 = \frac{-n_C^0}{2} + \frac{3\dot{e}_{10}}{2} \quad [26]$$

When this is substituted into the equivalent of [17], the extent of reaction [10] can be evaluated:

$$e_{10} = \frac{n_{SiO}^0 (P_{SiO} - 1) + n_{CO}^0 P_{SiO} + \frac{n_C^0}{2}}{\frac{3}{2} - 2 P_{SiO}} \quad [27]$$

Equations [26] and [27] relate the inlet mole numbers to the outlet mole numbers since P_{SiO} is fixed at the invariant point. For the same feed as above (1 mole SiO_2 and 2 moles C), $e_{10}=0.6787$ moles, and $e_7=0.0180$ moles, giving 0.3213 moles SiO_2 , 0.6607 moles SiC, 0.0108 moles SiO, and 3.3393 moles CO out.

In neither the binary equilibrium case nor the invariant point case is it suggested that the actual reaction mechanisms are the same reactions as those as written to describe the system. The actual mechanisms make no difference when the system is viewed from a timescale sufficient for equilibrium to be achieved, because kinetic effects are ignored in this model. No reaction will be "dominant" in the sense that it proceeds more rapidly than another. Many elementary reactions may be involved in the reaction mechanism; the reactions chosen to represent the system can be expressed as linear combinations of them.

The initial guess for the furnace stage flowrates is made by examining the free energies of a variety of possible reaction products over the furnace temperature range. For a feed ratio with a C: SiO_2 molar ratio of 2, possible products include:

Figure 2 is a plot of the free energies of these products versus temperature. It reveals that the most stable product changes from SiO_2 and C at low temperatures to products involving SiC at intermediate temperatures to products involving Si at high temperatures. With this information and the insight provided by Figure 1, the initial guess can be formulated. For a C: SiO_2 feed ratio, r , the guess is given in Table 3.

For perfect stoichiometry ($r=2$), either column in Table 3 can be used. The initial guess flowrates do not satisfy a mass balance, for they include SiO rising from deep within the furnace but do not include it coming out of the top or recirculating downward. In the equilibrium furnace, a large amount of material does recirculate in this manner. The purpose of the iteration is to quantify exactly how much.

Stage T [K]	Regime on Fig. 1	Products if $0 < r < 2$	Products if $2 < r < 3$
Room Temp - 1788	SiO ₂ -C	SiO ₂ +r C (falling) +r CO (rising)	SiO ₂ +r C (falling) +2 CO (rising)
1788-2084	SiO ₂ -SiC	(1-r/3) SiO ₂ (falling) +(r/3) SiC (falling) +r CO (rising)	(1-r/3) SiO ₂ (falling) +(r/3) SiC (falling) +2 CO (rising)
2084+	SiO ₂ -Si	(1-r/2) SiO ₂ (falling) + (r/2) Si (falling) + (2r/3) SiO (rising) + (r/3) CO (rising)	x
2084+	SiO ₂ -SiC	x	(3-r) Si (falling) + (r-2) SiC (falling) + (4-4r/3) SiO (rising) + (2-2r/3) CO (rising)

Table 3: Initial Guess for Stage Output Products

With the initial guess set, all that remains is to pick enough discreet temperature values to provide a smooth approximation of the system. Ideally, an infinite number of stages, one at every temperature, would be used to reach an exact solution. Fortunately, the calculation results converge rapidly at relatively low numbers of stages. The program written to perform this calculation prompts the user for the number of stages between room temperature and the first invariant point, and then for the number of stages between the first and second invariant points. Starting with the initial guess, the input flowrates and temperatures are used to calculate new

output flowrates, which are then used as inputs themselves. The process is repeated until the results have converged to within a specified tolerance.

The invariant temperatures are handled in a somewhat special manner. Multiple stages exist at the invariant temperatures to insure that products are equilibrated before and after the invariant reactions. Special feed and tap stages are also added to provide a constant material source and sink.

For a stoichiometric feed ratio ($r=2$), with 17 total stages spanning a temperature range from 300 K to 2084 K, and a tolerance of $\pm 1 \times 10^{-5}$ moles, the equilibrium furnace output is summarized in Table 4.

T [K]	SiO ₂	C	Si	SiC	SiO	CO
300 (Feed)	1.0000	2.0000	0	0	0	2.0000
300	1.0000	2.0000	0	0	0	2.0000
1044	1.0000	2.0000	0	0	0.0000	2.0000
1416	1.0001	2.0001	0	0	0.0000	2.0000
1602	1.0009	2.0009	0	0	0.0001	2.0000
1695	1.0032	2.0032	0	0	0.0009	2.0009
1742	1.0109	2.0109	0	0	0.0032	2.0032
1788 (Invariant Top)	1.0109	2.0109	0	0	0.0109	2.0109
1788 (Invariant Bottom)	0.3390	0	0	0.6695	0.0109	2.0108
1837	0.3464	0	0	0.6732	0.0085	0.6695
1887	0.3630	0	0	0.6815	0.0196	0.6732
1936	0.4027	0	0	0.7013	0.0445	0.6815
1985	0.5206	0	0	0.7603	0.1040	0.7013
2035	2.9367	0	0	1.9684	0.2809	0.7603
2084 (Invariant Top)	3.3782	0	0	2.1891	3.9051	1.9683
2084 (Invariant Bottom)	0	0	1.0000	0.0000	4.5674	2.1891
2084 (Tap)	0	0	1.0000	0.0000	0	0

Table 4: Individual Stage Molar Flowrates - Eq. Reaction Extent Model

From Table 4, the re-circulating nature of the equilibrium furnace can be seen. Silicon Monoxide gas is formed deep within the furnace, where the temperature is high enough that a significant PSiO exists. As this gas rises, the temperature decreases, so it condenses into a mixture of solid and liquid phases (SiO_2 and SiC above 1788 K, SiO_2 and C below 1788 K). These condensed phases then descend into the furnace again to repeat the process.

Table 4 also reveals that the equilibrium furnace can be 100% efficient. One mole of Silicon metal is produced per mole of silica feed, with no losses to either hearth SiC formation or SiO escape into the fume hood. This result, at first, appears contrary to other authors' claims that equilibrium furnaces cannot be this efficient. The difference lies in two areas: others have examined different feed ratios, and they have not assumed that the mixture remains in equilibrium away from the invariant points.⁵ The variable feed ratio in our model addresses the first of these effects. Interesting modifications to this model can be made to account for the second. The "Last Gas Stage" model considers the case when the charge mix layer on top of the SiC-producing zone in the furnace is thin enough to give non-equilibrium gas escape into the fume hood. The "Blow-Hole" model accounts for the effect of observed gas escape from deep within the furnace. These two models are discussed in more detail in the next section.

A graphical representation of the Table 4 data is included as Figure 3. To examine the temperature dependence of the actual furnace SiO partial pressure, Figure 4 plots these quantities at each stage and compares the values to the continuous curve developed in Figure 1. Results indicate that the furnace is indeed at equilibrium.

Figure 3: Molar Flowrates versus Stage Temperature - Eq. Reaction Extent Model

⁵ Schei, "A Stoichiometric Model of the Ferrosilicon Process" in 39th Electric Furnace Proceedings, Vol. 39, 1982, p. 303.

As reaction proceeds, a significant amount of gas is generated within the furnace. The Silicon Carbide formation reaction at 1788 K and the Silicon formation reaction at 2084 K both release Carbon Monoxide. Silicon Monoxide is also formed to achieve the equilibrium partial pressure. As all of this gas is evolved, it must go somewhere if the pressure is to remain at one atmosphere. The volumetric flowrate, $\dot{V}$, of an ideal gas is proportional to both the molar flowrate, $\dot{n}$, and the absolute temperature, T :

$$\dot{V} = \frac{\dot{n} RT}{P} \quad [33]$$

In a right circular cylindrical furnace, the volumetric flowrate is also equal to the cross-sectional area times the vertical velocity, u .

$$\dot{V} = \pi r^2 u \quad [34]$$

When these two equations are combined, and the calculated molar flowrates and temperatures are substituted in, the velocity profile shown in Figure 5 is obtained.

So far in the treatment of the equilibrium furnace, output flow calculation has been based solely on free energy. Enthalpic effects have been secondary because it was assumed that whatever heat transfer might be required was rapid enough to maintain equilibrium. With tabulated enthalpy data, it is possible to examine the heat effects in each stage. Figure 6 plots the enthalpy change, $\Delta H = H_{\text{out}} - H_{\text{in}}$ versus stage temperature. Positive ΔH values correspond to endothermic reactions; negative values signify exothermic reactions. From the Figure, one can see that both the SiC and Si forming reactions are endothermic while the condensation reactions are exothermic.

There is no difficulty with an endothermic reaction proceeding near the furnace bottom, for the arc provides whatever heat is necessary. In the upper regions of the furnace, however, reactants are effectively shielded from the arc energy, and the only heat supply is the rising gases. Furthermore, any heat transfer between the furnace and the surroundings will remove energy from the system, because it is at a higher temperature. So away from the arc, the rising hot gas must carry enough enthalpy to compensate for this effect as well. The "Enthalpy-Limited" model, described below, addresses this case.

The basic equilibrium reaction extent model can be modified in a number of ways to account for observed or expected behavior not conforming to the basic model assumptions. The "Last Gas Stage" model addresses the case when hot gas escapes into the fume hood without proceeding through lower temperature equilibrium stages in the charge mix layer on top of the SiC-producing zone in the furnace. The "Blow-Hole" model attempts to account for the effect of blow-holes, which permit hot inner furnace gas to escape directly. The "Enthalpy-Limited" model represents

the farthest departure from the equilibrium reaction extent model, for it reveals what happens when enthalpy considerations, rather than thermodynamic equilibrium, govern furnace stage output.

Last Gas Stage Model

The last gas stage model is identical to the equilibrium reaction extent model, except that the top furnace stage temperature is specified rather than assumed to be the feed temperature. As the partial pressure of SiO is low at temperatures up to 1788 K, this value is used as the last gas equilibrium stage temperature. This simulation gives an upper bound on the deviation predicted by this model. For the same feed ratio, mid-furnace stage temperatures, and tolerance, the simulation returns the molar flowrates shown in Table 5. The deviation from the first model is small: 97.84% Silicon recovery, with small losses to both hearth SiC and fume hood SiO.

T [K]	SiO ₂	C	Si	SiC	SiO	CO
300	1.0000	2.0000	0	0	0.0108	1.9892
1788	1.0000	2.0000	0	0	0.0108	1.9892
1788	1.0000	2.0000	0	0	0.0108	1.9892
1788	0.3317	0	0	0.6659	0.0107	1.9891
1837	0.3389	0	0	0.6695	0.0084	0.6550
1887	0.3551	0	0	0.6776	0.0192	0.6587
1936	0.3940	0	0	0.6970	0.0435	0.6668
1985	0.5094	0	0	0.7547	0.1018	0.6862
2035	2.8734	0	0	1.9367	0.2749	0.7439
2084	3.3054	0	0	2.1527	3.8209	1.9259
2084	0	0	0.9784	0.0108	4.4689	2.1419
2084	0	0	0.9784	0.0108	0	0

Table 5: Individual Stage Molar Flowrates - Last Gas Stage Model

Blow-Hole Model

The blow-hole model is conceptually similar to the last gas stage model, but the gas leaves an intermediate stage rather than simply exiting the top at a higher temperature. Since the gas is richer in SiO deeper within the furnace, more significant losses can occur. Table 6 is the calculated output for a case using the same feed, temperatures, and tolerance as the equilibrium reaction extent model, but with a blow-hole diverting 50% of the gas leaving the stage at 2034 K to the fume hood.

T [K]	SiO ₂	C	Si	SiC	SiO	CO
300	1.0000	2.0000	0	0	0.1097	1.8904
300	1.0000	2.0000	0	0	0	1.5936
1044	1.0000	2.0000	0	0	0.0000	1.5936
1416	1.0000	2.0000	0	0	0.0000	1.5936
1602	1.0007	2.0007	0	0	0.0000	1.5936
1695	1.0026	2.0026	0	0	0.0007	1.5943
1742	1.0087	2.0087	0	0	0.0026	1.5962
1788	1.0086	2.0086	0	0	0.0087	1.6023
1788	0.3356	0	0	0.6678	0.0086	1.6022
1837	0.3384	0	0	0.6692	0.0033	0.2614
1887	0.3449	0	0	0.6725	0.0077	0.2628
1935	0.3604	0	0	0.6802	0.0174	0.2660
1985	0.4064	0	0	0.7032	0.0406	0.2738
2034	2.2928	0	0	1.6464	0.1097	0.2968
2084	2.6375	0	0	1.8187	3.0488	1.5367
2084	0	0	0.7807	0.1097	3.5659	1.7091
2084	0	0	0.7807	0.1097	0	0

Table 6: Individual Stage Molar Flowrates - Blow-Hole Model

Enthalpy-Limited Model

If a region in the furnace is effectively shielded from the arc power source, and its temperature is to remain constant, then enthalpy becomes the critical quantity. Conducting an energy balance around such a zone at steady-state reveals that the enthalpy in must equal the enthalpy out plus the heat transferred out of the system to the surroundings. The incoming enthalpy is comprised of that from the descending condensed phases at a lower temperature and that from the rising higher temperature gases. The heat transfer can be approximated by some overall heat transfer coefficient, U , times an area, A , and a temperature difference driving force, ΔT . To fully model the heat transfer, some assumption must be made regarding the spatial temperature distribution in the furnace. This is a troublesome estimate, for none of the other models provide any information toward this cause. To eliminate this difficulty, as a first approximation, the furnace insulation will be assumed effective enough so as to render the furnace adiabatic. This is clearly an idealization, but it provides results that demonstrate significant changes in furnace operation nonetheless.

Again considering the SiO₂-C regime as an example, the inlet enthalpy is the sum of the inlet flowrates times the component enthalpies at the inlet temperatures (different for condensed phases and gas):

$$H_{in} = n_{SiO_2}^o H_{SiO_2}^o + n_C^o H_C^o + n_{SiO}^o H_{SiO}^o + n_{CO}^o H_{CO}^o \quad [35]$$

The outlet enthalpy is just the sum of the outlet flowrates times the component enthalpies at the outlet temperature (the same for everything):

$$H_{out} = n_{SiO_2} H_{SiO_2} + n_C H_C + n_{SiO} H_{SiO} + n_{CO} H_{CO} \quad [36]$$

For enthalpy control, the restriction placed on the system is that the inlet and outlet enthalpies be equal. If heat transfer to the surrounding were considered, the outlet would equal the inlet minus whatever heat is lost. Note that nowhere in this formulation does the equilibrium SiO partial pressure appear. Thus, fundamentally non-equilibrium results are expected. If the reaction extent is then added to or subtracted from the mole numbers as reaction stoichiometry dictates, the system can be solved for the reaction extent:

$$e = \frac{n_{\text{SiO}_2}^{\circ} (H_{\text{SiO}_2}^{\circ} - H_{\text{SiO}_2}) + n_{\text{C}}^{\circ} (H_{\text{C}}^{\circ} - H_{\text{C}}) + n_{\text{SiO}}^{\circ} (H_{\text{SiO}}^{\circ} - H_{\text{SiO}}) + n_{\text{CO}}^{\circ} (H_{\text{CO}}^{\circ} - H_{\text{CO}})}{-H_{\text{SiO}_2} - H_{\text{C}} + H_{\text{SiO}} + H_{\text{CO}}} \quad [37]$$

Note that the denominator is simply the heat of reaction. Similar derivations exist for other binary regions.

The first invariant point (SiO₂, C, SiC) is also considered to be in a region shielded from the arc power source, so it, too, must obey the isenthalpic restriction. Unfortunately neither of the reactions chosen to represent the invariant point, [10] and [7], can proceed at 1788 K because they are both endothermic. Further examination reveals that this is the case at every temperature between 300 K and 2300 K, so no Silicon Carbide formation from silica and Carbon is expected. The ramification of this statement is that silica and Carbon continue to exist together at temperatures in excess of 1788 K, where they are thermodynamically no longer stable. The *second* invariant point is in the vicinity of the arc, so the greatly endothermic Silicon producing reaction is handled exactly as in the equilibrium case.

These assumptions give output numbers vastly different from those obtained earlier, even when all other parameters are constant. No Silicon is produced. Instead, losses are equally split between fume hood SiO and hearth SiC. Table 7 shows the calculated molar flow numbers; Figures 7-10 show the corresponding plots of molar flowrates, furnace SiO pressure, gas velocity, and heat effects.

T [K]	SiO ₂	C	Si	SiC	SiO	CO
300	1.0000	2.0000	0	0	0.5000	1.5000
300	0.9700	1.9700	0	0	0.5000	1.5000
657	0.9798	1.9798	0	0	0.4700	1.4700
1014	1.0013	2.0013	0	0	0.4798	1.4798
1370	1.0265	2.0265	0	0	0.5013	1.5013
1727	1.0515	2.0515	0	0	0.5265	1.5265
2084	0	0	0	0.5000	0.5515	1.5515
2084	0	0	0	0.5000	0	0

Table 7: Individual Stage Molar Flowrates - Enthalpy Limited Model

Figure 7: Molar Flowrates versus Stage Temperature - Enthalpy Limiting Model

Figure 9: Furnace Gas Velocity

Figure 10: Heat Effects

One point to make concerning the enthalpy-limited furnace results presented is that some SiC would actually be present before the bottom arc stage. As SiO and CO rise, several potential condensation reactions can occur. Reactions [3], [4], and [5], written backwards are all possible:

While the extents of these reactions would still be constrained to give zero net enthalpy change, their relative effects would now be determined largely by kinetic factors, since the furnace is clearly no longer at equilibrium. Still, the enthalpy-limited furnace is a useful tool, for it shows that some heat source is necessary in the furnace regions far from the arc to permit the equilibrium re-circulation pattern and its corresponding high efficiency. Real furnace operation is a hybrid of this model, the equilibrium reaction extent model, the last gas stage model, and the blow-hole model.

SUMMARY OF NUMERICAL RESULTS

In the summer of 1994, SCRA ran a series of tests in a 1.5 MW furnace, aimed at producing industrial-grade Silicon and Ferro-silicon, and investigating the potential benefits of the hollow electrode split feed technique. From tables provided in the report, pseudo-continuous feed rates were calculated, as well as operational C:SiO₂ molar feed ratios. For the pure Silicon runs: T1, T2, T3, and T6, the average silica feed rate was near 0.29 moles per second. Using the furnace feed make-up mix table and the chemical analyses provided for the raw materials used, C:SiO₂ molar feed ratios were calculated as 2.072, 2.172, and 2.232. These values were used as input in the simulations presented below.

Table 8 is a composite list of all the calculations performed, and the models and parameters used. Table 9 is a list of the predicted furnace efficiencies, with losses to each side product quantified. From the tables, it can be seen that the equilibrium model predicts high furnace efficiencies for feed ratios near 2:1. Complete recirculation of the rising Silicon Monoxide gas explains this effect. The Top Gas Model predicts some loss to fume hood SiO, but the amount is small because of the low P_{SiO} in this part of the furnace. The Blow-Hole model can account for significant losses, as the inner furnace gas is much richer in SiO. It is interesting to note that any SiO loss alters the stoichiometry in the remainder of the furnace, making it more Carbon-rich. If the beginning feed was already Carbon-rich, this effect can produce excessive amounts of SiC. Carbon-lean feeds help mitigate this effect somewhat.

The actual computer programs used for these calculations are given in the Appendix. We note that this is a working document, rather than a commercial software package. We would certainly be happy to explore future interactions with the industrial community regarding the use of both the programs and this modeling approach, provided some appropriate framework can be established for this.

Simulation #	Model	Feed Ratio	# Stages	Tolerance	Comments
1	Equilibrium	2	17	1×10^{-5}	Ideal Stoich. Presented Above
2	Equilibrium	1.5	17	1×10^{-5}	C-Lean Stoich.
3	Equilibrium	2.5	17	1×10^{-5}	C-Rich Stoich.
4	Equilibrium	2.072	17	1×10^{-5}	C-Rich Stoich. Used by SCRA
5	Equilibrium	2.172	17	1×10^{-5}	C-Rich Stoich. Used by SCRA
6	Equilibrium	2.232	17	1×10^{-5}	C-Rich Stoich. Used by SCRA
7	Last Gas Stage	2	12	1×10^{-5}	Ideal Stoich. $T_{last}=1788$ Presented Above
8	Last Gas Stage	1.5	12	1×10^{-5}	C-Lean Stoich. $T_{last}=1788$
9	Last Gas Stage	2.5	12	1×10^{-5}	C-Rich Stoich. $T_{last}=1788$
10	Blow-Hole	2	17	1×10^{-5}	Ideal Stoich. Blow-Hole at 2000 K Presented Above
11	Blow-Hole	1.5	17	1×10^{-5}	C-Lean Stoich. Blow-Hole at 2000 K

Table 8: Simulation Parameters

Simulation #	Model	Feed Ratio	# Stages	Tolerance	Comments
12	Blow-Hole	2.5	17	1x10 ⁻⁵	C-Rich Stoich. Blow-Hole at 2000 K
13	Enthalpy Limited	2	12	1x10 ⁻⁵	Ideal Stoich. Presented Above
14	Enthalpy Limited	1.5	12	1x10 ⁻⁵	C-Lean Stoich.
15	Enthalpy Limited	2.5	12	1x10 ⁻⁵	C-Rich Stoich.

Table 8 Continued: Simulation Parameters

Simulation #	% Si Recovery	% Loss - SiO ₂	% Loss - SiC	% Loss - SiO
1	100	0	0	0
2	75	25	0	0
3	50	0	50	0
4	92.8	0	7.2	0
5	82.8	0	17.2	0
6	76.8	0	23.2	0
7	97.84	0	1.08	1.08
8	74.59	24.59	0	0.82
9	47.84	0	51.08	1.08
10	78.07	0	10.97	10.97
11	70.08	20.08	0	9.84
12	39.04	0	55.48	5.48
13	0	0	50	50
14	0	0	25	75
15	0	0	75	25

Table 9: Simulation Results

FORMULATION OF FREE ENERGY MINIMIZATION MODELS

Theory and Limitations of Reaction Extent Models

Determination of the equilibrium composition of a system is straightforward when only one reaction is considered. Consider the following general reaction:

The equilibrium constant, K , for this reaction is the ratio of the activities of the products to the activities of the reactants, with each species' activity raised to a power given by the stoichiometric coefficient of the particular species:

$$K = \frac{a_C^c a_D^d}{a_A^a a_B^b} \quad [39]$$

It is generally known that the activity of a species may be related to its concentration by some function; a simple function is often used:

$$a_i = \gamma_i x_i \quad [40]$$

where the activity of species i has been written as the product of a so-called activity coefficient for that species, γ_i , and the mole fraction, x_i ; note that the activity coefficient may be a function of the mole fraction, as $\gamma_i = f(x_i)$. Also, both the activity and the activity coefficient generally vary with temperature. Commonly used approximations are Raoult's law, for nearly pure solutions:

$$\lim_{x_i \rightarrow 1} \gamma_i x_i = x_i \quad [41]$$

where it is assumed that $\gamma_i \rightarrow 1$, and Henry's law, for dilute solutions:

$$\lim_{x_i \rightarrow 0} \gamma_i x_i = \gamma_i^0 x_i \quad [42]$$

where it is assumed that $\gamma_i \rightarrow \gamma_i^0$, which is independent of x_i but may vary with temperature.

For this discussion it is assumed that the activities of condensed species may be approximated by their mole fractions:

$$a_i = x_i = \frac{n_i}{n_t} \quad [43]$$

where n_i is the number of moles of species i and n_t is the total number of moles. The activities of the gaseous species may be approximated by their mole fractions multiplied by the dimensionless total pressure, P :

$$a_i = x_i P = \frac{n_i}{n_t} P \quad [44]$$

The dimensionless total pressure is equal to the ratio of the (dimensional) total pressure to the standard pressure; the dimensionless total pressure is numerically equal to the dimensional total pressure when the standard pressure is one atmosphere. Note that the value of n_t is the total number of moles in the particular phase which harbors the species in question; n_t for a condensed species will be different than for a gaseous species, and n_t for two condensed species may be different if there are several condensed-species phases in the system. It is further assumed for simplicity in this discussion that the total number of moles in each phase is one, and that the total dimensionless pressure is one:

$$K = \frac{n_C^c n_D^d}{n_A^a n_B^b} \quad [45]$$

For species j , the mole number at equilibrium, n_j , may be written in terms of the mole number initially, $(n_j)_i$, the stoichiometric coefficient for that species, and the extent of reaction, λ . The equilibrium constant can then be written in these terms:

$$K = \frac{(n_{C_i} + c\lambda)^c (n_{D_i} + d\lambda)^d}{(n_{A_i} - a\lambda)^a (n_{B_i} - b\lambda)^b} \quad [46]$$

The reaction quotient, Q , is defined analogously to the equilibrium constant:

$$Q = \frac{n_{C_i}^c n_{D_i}^d}{n_{A_i}^a n_{B_i}^b} \quad [47]$$

The term ΔG is a measure of the driving force for reaction:

$$\Delta G = RT \ln \frac{Q}{K} \quad [48]$$

When the amount of products, species C and D in this case, is smaller than the equilibrium amount, then Q will be less than K and ΔG will be negative. The sign of the extent of reaction, λ , is given by the following inequality, which holds for spontaneous reactions:

$$\Delta G \delta \lambda < 0 \quad [49]$$

When ΔG is negative, $\delta \lambda$ is positive; that is, the reaction proceeds "forward," and more products are formed. If the free energy of formation for each species participating in the reaction is known, then the free energy for the reaction can be calculated:

$$\Delta G_{r,T}^\circ = c\Delta G_{C,T}^\circ + d\Delta G_{D,T}^\circ - a\Delta G_{A,T}^\circ - b\Delta G_{B,T}^\circ \quad [50]$$

where the subscript r denotes reaction, T denotes that the reaction is occurring at temperature T , and the superscript $^\circ$ denotes that the system is at standard pressure.

The equilibrium constant and then the extent of reaction can then be calculated:

$$K = \exp\left(-\frac{\Delta G_{r,T}^\circ}{RT}\right) = \frac{(n_{C_i} + c\lambda)^c (n_{D_i} + d\lambda)^d}{(n_{A_i} - a\lambda)^a (n_{B_i} - b\lambda)^b} \quad [51]$$

For simple cases, an analytical expression for the extent of reaction may be obtained; in other cases, numerical methods will allow solution for λ . This discussion has shown that for a single reaction, the equilibrium composition may be found by the law of mass action and solution for a single variable, namely the extent of reaction, λ .

For simultaneous equilibria among a set of reactions at so-called invariant points, the analysis becomes increasingly difficult. The problem is simplified by replacing several reactions with a single "invariant reaction." The invariant reaction is a linear combination of this set of reactions. The extent of the invariant reaction can be found by the method outlined above.

Theory of Free Energy Minimization Models in the Si-O-C System

Several researchers have proposed to calculate the equilibrium composition in a system with many components by a free-energy-minimization approach. The theoretical basis for this method is due to Gibbs⁶, and the technique was outlined and developed by Brinkley⁷, White⁸, Eriksson⁹, and others. The most popular implementation of the technique is the program SOLGASMIX (originally SOLGAS), written by Eriksson. Several versions of this program are available from

⁶ Gibbs, J., *Collected Works of J. Willard Gibbs*, Longmans, Green and Company, New York, 1928, Vol. 1, pp. 63-82

⁷ Brinkley, S., *J. Chem. Phys.* **14** (1946) 563-564 and Brinkley, S., *J. Chem. Phys.* **15** (1947) 107-110

⁸ White, W., Johnson, S. and Dantzig, G., *J. Chem. Phys.* **28** (1959) 251-255

⁹ Eriksson, G., *Acta Chem. Scand.* **25** (1971) 2651-2658 and Eriksson, G., *Chem. Scr.* **8** (1975) 100-103

Professor Morris at the University of Missouri at Rolla (UMR). He has used this software to analyze several chemical systems, and his results are in the literature.

We review here the theory behind the free-energy-minimization approach, using the Si-O-C system. We are motivated to use this system by our current work with the South Carolina Research Authority (SCRA). Our collaborators at SCRA have built and operated a direct-current plasma arc furnace for the carbothermic reduction of silica to Silicon metal and Ferro-silicon alloys.

Thermodynamic equilibrium at an absolute temperature T in the Si-O-C system is defined by the global minimum of the function G_{sys} :

$$G_{sys} = n_C G_C^o + n_{SiO_2} G_{SiO_2}^o + n_{Si} G_{Si}^o + n_{SiC} G_{SiC}^o + n_{SiO} G_{SiO}^o + n_{CO} G_{CO}^o + n_{SiO} RT \ln \frac{n_{SiO}}{n_{SiO} + n_{CO}} + n_{CO} RT \ln \frac{n_{CO}}{n_{SiO} + n_{CO}} + n_{SiO} RT \ln P + n_{CO} RT \ln P \quad [52]$$

where G_{sys} is the Gibbs free energy of the system, n_i is the number of moles of species i in the (closed) system, G_i^o is the molar Gibbs free energy of species i at temperature T and standard pressure, R is the universal gas constant, T is the absolute temperature, and P is the dimensionless total pressure. It is assumed that the activity of the condensed species is equal to one, and that the activities of the gaseous species are equal to their respective dimensionless partial pressures. The solution is subject to mass balance constraints on the three components in the system, namely, Silicon, Oxygen, and Carbon:

$$\begin{aligned} n_{SiO_2} + n_{Si} + n_{SiC} + n_{SiO} &= n_{SiO_2}^i + n_{Si}^i + n_{SiC}^i + n_{SiO}^i \\ 2n_{SiO_2} + n_{SiO} + n_{CO} &= 2n_{SiO_2}^i + n_{SiO}^i + n_{CO}^i \\ n_C + n_{SiC} + n_{CO} &= n_C^i + n_{SiC}^i + n_{CO}^i \end{aligned} \quad [53]$$

The solution is also subject to non-negativity constraints:

$$\begin{aligned} n_C &\geq 0 & n_{SiO_2} &\geq 0 & n_{Si} &\geq 0 \\ n_{SiC} &\geq 0 & n_{SiO} &\geq 0 & n_{CO} &\geq 0 \end{aligned} \quad [54]$$

Theory of Optimization

For a function z that depends only on one variable x , the necessary condition for an optimum (either a local maximum or local minimum) is that the derivative of z with respect to x is equal to zero:

$$\frac{dz(x)}{dx} = 0 \quad [55]$$

Taking the derivative gives one equation, and solving this equation gives the value of x at the local optimum; the nature of this optimum must be investigated by examining the higher-order derivatives in the neighborhood of this value of x .

For a function z that depends on n variables x_i ($i=1, 2, \dots, n$), the necessary conditions for a local optimum are analogous:

$$\frac{\partial z(\vec{x})}{\partial x_i} = 0 \quad [56]$$

Taking these partial derivatives gives a system of n equations in n unknowns.

The situation is more complicated when there are constraints. If the constraints are all equalities, then the constrained optimization problem may be replaced by an unconstrained optimization problem by several methods.

Suppose that there are m equality constraints, so that the problem may be stated:

Optimize $z(\vec{x})$ subject to $g_j(\vec{x}) = b_j$

$$\vec{x} = (x_1, x_2, \dots, x_n)^T \quad [57]$$

$$j = 1, 2, \dots, m$$

The function z is usually called the "objective function" in constrained optimization problems. For each constraint equation, if the equation can be solved for one of the variables x_i , then this variable can be eliminated from the function z . If the constraint equations can each be solved for a different variable, then the number of variables in the objective function is reduced to $n-m$, and the "calculus" method outlined above can be used to form $n-m$ equations for the $n-m$ variables. Usually this cannot be done, so this method is not very practical.

Another method is to "eliminate increments" from the function z . The objective function is written as a Taylor series about an optimum point; this process in itself is quite complicated when the number of variables n is more than two or three. In the Taylor series, the independent "variables" are the increments $\Delta x_i = x_i - x_i^*$ for $i=1, 2, \dots, n$, where the asterisk superscript denotes the value of the variable x_i at the optimum. These increments are not independent, however. If the m constraint equations are similarly written as Taylor series, these form a system of m equations in n unknowns (namely the n increments Δx_i); m increments can be expressed in terms of $n-m$ remaining unknown increments. At this point, the m increments just solved for are substituted into the objective function. By this method, the series expansion of the objective function can be written in terms of $n-m$ increments. The necessary conditions for this case are that the derivative of the series with respect to each independent increment be zero, so the coefficient of each independent increment must be zero; this gives $n-m$ equations, and each equation is a function of the original n variables. If we include the m constraint equations, then we have a system of n equations in n unknowns.

The most popular method to replace a constrained optimization problem by an unconstrained optimization problem is to use Lagrange multipliers. The solution method is to write the objective function and each constraint equation as a Taylor series. Each constraint equation is multiplied by a "Lagrange multiplier", and the result is added to the series for the objective

function. Again, the necessary conditions for an optimum are that the coefficient for each independent increment be zero; this gives n equations, but there are now $n+m$ variables since one multiplier was introduced for each constraint equation. We get m more equations by using the constraint equations, so we have a system of $n+m$ equations in $n+m$ unknowns. Another way to obtain the necessary conditions using the Lagrange multiplier method is to form a new objective function, called the Lagrangian:

$$L(\vec{x}) = z(\vec{x}) + \sum_{j=1}^m \Lambda_j [g_j(\vec{x}) - b_j] \quad [58]$$

The necessary conditions are formed by partial differentiation of the Lagrangian with respect to the n variables x_i and the m variables Λ_j , which gives a system of $n+m$ equations in $n+m$ unknowns.

When some of the constraints are "inequality" constraints, then "slack" variables are introduced:

$$g(\vec{x}) \leq b_j \text{ is replaced by } g(\vec{x}) + s_j^2 = b_j$$

and

$$g(\vec{x}) \geq b_j \text{ is replaced by } g(\vec{x}) - s_j^2 = b_j$$

[59]

By this method we can replace each inequality with an equality constraint. The problem of optimization with inequality constraints is thus transformed into an optimization subject to equality constraints. This can be further transformed into an unconstrained optimization by the Lagrange-multiplier method outlined above. In the general case, we want to optimize a function of n variables subject to m equality constraints and p inequality constraints. We introduce one slack variable for each inequality constraint and then one Lagrange multiplier for each constraint (including those constraints that were previously inequality constraints) to obtain a Lagrangian which is a function of $n+m+2p$ variables. We obtain $n+m+2p$ equations by partial differentiation with respect to the original n variables of the objective function, the p slack variables, and the $m+p$ Lagrange multipliers.

Another method that is possible for solution of the original problem is direct-search. This method is computationally and conceptually simple, and is receiving increased attention as the computing power available to engineers increases. The idea is to try many possible combinations of values for the variables and see which combination gives the optimal value of the objective function. For an integer problem, that is, one in which the variables can take only integer values, the direct-search method gives exact results. For a continuous problem, the direct-search method gives an approximate result. The solution can be improved by sampling values at closer intervals. This method is useful even if other methods are used to solve the problem; this method can be used with a coarse sampling interval and the solution can thereby be verified. As we have stated, the accuracy of the method depends on a fine discretization of the solution space. The speed, on the other hand, is reduced as the discretization is made finer. Therefore, a compromise is necessary. One method is to locate the solution approximately with a coarse search, and then use a narrower, finer search to obtain an improved solution.

For the solution of the Si-O-C system, we would seek an optimum of the function G_{mix} and the six mole number variables would be allowed to vary from zero to the maximum allowed by the mass balance:

$$\begin{aligned}
 0 \leq n_C &\leq (n_C^i + n_{\text{SiC}}^i + n_{\text{CO}}^i) \\
 0 \leq n_{\text{SiO}_2} &\leq \min \left\{ \left(n_{\text{SiO}_2}^i + n_{\text{Si}}^i + n_{\text{SiC}}^i + n_{\text{SiO}}^i \right), \left(n_{\text{SiO}_2}^i + \frac{1}{2} n_{\text{SiO}}^i + \frac{1}{2} n_{\text{CO}}^i \right) \right\} \\
 0 \leq n_{\text{Si}} &\leq \left(n_{\text{SiO}_2}^i + n_{\text{Si}}^i + n_{\text{SiC}}^i + n_{\text{SiO}}^i \right) \\
 0 \leq n_{\text{SiC}} &\leq \min \left\{ \left(n_{\text{SiO}_2}^i + n_{\text{Si}}^i + n_{\text{SiC}}^i + n_{\text{SiO}}^i \right), \left(n_C^i + n_{\text{SiC}}^i + n_{\text{CO}}^i \right) \right\} \\
 0 \leq n_{\text{SiO}} &\leq \min \left\{ \left(n_{\text{SiO}_2}^i + n_{\text{Si}}^i + n_{\text{SiC}}^i + n_{\text{SiO}}^i \right), \left(2n_{\text{SiO}_2}^i + n_{\text{SiO}}^i + n_{\text{CO}}^i \right) \right\} \\
 0 \leq n_{\text{CO}} &\leq \min \left\{ \left(2n_{\text{SiO}_2}^i + n_{\text{SiO}}^i + n_{\text{CO}}^i \right), \left(n_C^i + n_{\text{SiC}}^i + n_{\text{CO}}^i \right) \right\}
 \end{aligned} \tag{60}$$

The idea then is that these six ranges make up a "solution space" and some combination of values of these variables, that is, some point in the solution space, will give the optimal value to the objective function—all we have to do is try all the possibilities and see which one is best. Of course, the confidence we have in our solution increases as the number of points we try increases, but then the computing time increases also. This method is easy to understand and to program, and with increasing computing speed available, it may become the method of choice for this type of problem. As an example of the dependence on computing speed, we solved a problem:¹⁰

$$\text{Minimize } 2x^3 + 3y^2 \tag{61}$$

where $0 \leq x \leq 10$ and $0 \leq y \leq 10$

The solution was also subject to another inequality constraint:

$$x + \frac{2}{y} \leq 50 \tag{62}$$

The direct-search method in this case is to try values of x and y , and then, if these values satisfy the inequality constraint, evaluate the objective function and compare the value against the current minimum. If the new value is smaller, then this value becomes the optimal one, and the associated x and y values identify the optimal point in the solution space. This problem was solved three times, with increasing density of sampling. First, the increment between points on both x and y was 1.0. The second and third times it was 0.1 and 0.01, respectively. On a Macintosh personal computer, the first run took 3 seconds and 10 feasible points were considered (that is, the objective function was evaluated at 10 points). The second run took 85 seconds and 23 points were evaluated. The third run took 2 hours and 15 minutes and 391 points were evaluated. These three runs were repeated on an Sun workstation, and the results are as follows:

¹⁰ Hanna, O. and Sandall, O., *Computational Methods in Chemical Engineering*, Prentice Hall PTR, Upper Saddle River, New Jersey, 1995, p. 209

Sampling Increment	# Points Evaluated	Time [s]
1.0	10	<1
0.1	85	<1
0.01	871	4

Table 10: Direct Search Computation Times on a Sun Workstation

The same solution was obtained (on both machines at the finest sampling interval), but the time is significantly different. Note that the number of points evaluated is different; this may be due to differences in the number digits used in double-precision calculations on the two machines.

Selected Results from SOLGASMIX-Based Programs

As discussed earlier, SOLGASMIX is the most popular program for solution of chemical-equilibrium compositions by free-energy minimization. It is the basis of a number of commercial and freeware software packages, including STEPSOL and STGSOL from University of Missouri at Rolla (UMR) and EQUILIB from Ecole Polytechnique de Montreal. The STEPSOL program is designed to solve for the equilibrium composition of a batch reactor. The input to the reactor, reactor temperature, reactor pressure or volume, and thermodynamic data must be provided. The program calculates the equilibrium composition by minimizing the free energy of the reactor; all possible species that may exist in the reactor must be specified.

For the Si-O-C system, several calculations were made using the STEPSOL program. The reactor pressure was maintained at one atmosphere, and the input amounts were 2 moles of Carbon, one mole of silica, and 0.1 mole of CO gas; a small amount of gas must be added or SOLGASMIX cannot converge to a solution. The same six species are considered in calculating the equilibrium composition of the reactor for various temperatures. Tables 11 and 12 show the outlet mole numbers and the gas partial pressures, respectively.

Temp [K]	SiO ₂ (s,l) [mol]	C (s) [mol]	Si (s,l) [mol]	Si (g) [mol]	SiC (s) [mol]	SiO (g) [mol]	CO (g) [mol]
1500	1.0	2.0	0.0	0	0.0	0.0	0.1
1800	0.33	0.0	0.0	0	0.66	0.0096	1.44
2100	0.0	0.0	0.0	0	0.50	0.50	1.6
2400	0.0	0.0	0.0	0	0.50	0.50	1.6
2600	0.0	0.0	0.0	0.02	0.49	0.49	1.61
2700	0.0	0.0	0.17	0.05	0.39	0.39	1.71
2800	0.0	0.0	0.29	0.07	0.32	0.32	1.78
2900	0.0	0.0	0.35	0.15	0.25	0.25	1.85
2950	0.0	0.0	0.35	0.19	0.23	0.23	1.87
3000	0.0	0.0	0.33	0.25	0.21	0.21	1.89
3050	0.0	0.0	0.29	0.33	0.19	0.19	1.91
3100	0.0	0.0	0.21	0.45	0.17	0.17	1.93

Table 11: STEPSOL Outlet Molar Flowrates

Temp [K]	Si [atm]	SiO [atm]	CO [atm]
1500	0.0	0.00	1.0
1800	0.0	0.01	0.99
2100	0.0	0.24	0.76
2400	0.0	0.24	0.76
2600	0.0098	0.23	0.76
2700	0.019	0.18	0.80
2800	0.036	0.14	0.82
2900	0.063	0.11	0.82
2950	0.083	0.10	0.82
3000	0.088	0.11	0.80
3050	0.14	0.077	0.78
3100	0.18	0.066	0.76

Table 12: STEPSOL Outlet Partial Pressures

These results from the STEPSOL program are typical of SOLGASMIX-based software. This software typically models idealized situations, and the user must adapt the programs to model their actual process. The calculations are much more rapid than hand calculations since a database of thermodynamic data is accessed by the program. Unfortunately, one typically wants to model something more complicated. In the SCRA furnace, for example, we might want to model the reactor as a series of equilibrium batch reactors, with material and energy passing between reactors. Unfortunately, the software from UMR is not easily modified by one unfamiliar with the code; in fact, the source code is difficult to work with since it is already a modified version of the original SOLGASMIX program. Also, there is no heat balance incorporated in the model.

DISCUSSION

We have presented two modeling approaches: reaction extent and free energy minimization. Both have the goal of predicting the equilibrium composition of the furnace output. Slight modifications to the reaction extent model permit examination of some non-equilibrium behavior. Reaction extent models are convenient because the reaction framework can be specified, so that inlet and outlet are connected by a clear path rather than a complex mathematical "black box." They are also computationally simple since they involve solving only linear algebraic equations. The primary drawback is that they are not readily expandable; they require some degree of re-derivation when additional species are considered. Free energy minimization models have the advantage that they do not require extensive modification when new species are taken into account. They tend to be somewhat more mathematically and computationally intensive and less transparent than reaction extent models, however. Both types of models can adequately describe the reduction of silica in a direct current, submerged plasma arc furnace. The best model for any system will be determined by a combination of factors such as system complexity, computational time constraints, and parametric sensitivity.

Discussion of Reaction Extent Models

Reaction extent models are attractive because the calculations are largely transparent. Reactions are specified, and extents are calculated, so that it is possible to see exactly what is happening. Some reaction extent model calculations have been presented and discussed in the section above. For the Si-O-C system described, this model is a valuable tool. Reaction extent models become more difficult as the number of components and/or species increases, however. With extra elemental components, the system will possess more degrees of freedom, so more phases will exist together. As the number of species increases, more reactions between them become possible, requiring examination of more equilibria and producing a more complex equilibrium diagram.

If the system were modified to include Iron as well, application of the Gibbs Phase Rule, equation [1], would reveal that three condensed phases and a gas will co-exist along a line, and that four condensed phases and a gas will exist at an invariant point. With tertiary and quaternary equilibria, instead of binary equilibria, multiple reactions would need to be specified and multiple extents calculated. Even if only one Iron Oxide, one Iron Carbide, and one Iron-Silicon compound were considered as additional species, thirty-five possible combinations of three phases would exist, requiring the construction of thirty-five $P_{\text{SiO}}-T$ lines on the equilibrium diagram!

Additional gaseous components and/or species, such as Nitrogen or Oxygen, would further complicate matters, as the partial pressures of Silicon Monoxide and Carbon Monoxide would no longer sum to one. This would violate the assumption made in deriving equation [9], and require additional information such as the amounts or partial pressures of the extra gaseous species.

Discussion of Free Energy Minimization Models

The free energy minimization approach is best when the number of components is large. No reaction scheme must be specified to find the solution by this approach. One must specify all possible species, and the set which minimizes the system free energy is chosen. Several numerical solution methods are available once the objective function and constraints have been specified. The direct-search method simply tries every combination of species that satisfies the constraints and finds which one gives the optimal value of the objective function; one must specify how densely the trial values are spaced. Other solution methods start by replacing the constrained optimization problem by an unconstrained optimization problem. Several methods are available to solve the problem at this point. We discuss the methods used in SOLGAS and SOLGASMIX below, and then recommend an approach based on the method of conjugate gradients.

Eriksson's SOLGAS program¹¹ uses the Lagrange multiplier method. He ignores the non-negativity constraints, and seeks solutions with only positive mole numbers by removing a species from the system if it has negative mole numbers in the early iterations. This leaves him with an optimization problem with equality constraints, namely, the mass balance equations. This is replaced by an unconstrained optimization problem by introducing Lagrange multipliers. The necessary conditions provide a system of equations to solve for the original variables and the Lagrange multipliers. The system of equations is nonlinear, however, since it contains logarithmic

¹¹ Eriksson, G., *Acta Chem. Scand.* **25** (1971) 2651-2658

terms for the gaseous species, and Eriksson removes the nonlinearities by expanding the nonlinear equations in Taylor series and neglecting all higher-than-first-order terms. The result is a system of linear equations, which is solved by Gaussian elimination. In Eriksson's SOLGASMIX program¹² the unconstrained optimization problem is solved using the method of steepest descent. Still non-negativity constraints are neglected, and again he opts to remove from the calculation species that have negative mole numbers in early iterations.

A preferable method for optimization of the objective function with equality and inequality constraints is to introduce slack variables for each inequality constraint, then Lagrange multipliers for all constraints. The resulting Lagrangian is solved using the method of conjugate directions (also called the method of conjugate gradients), originally developed by Fletcher and Reeves¹³. It is preferable to incorporate the non-negativity constraints into the original problem; this was probably avoided by Eriksson to avoid the additional variables and increased computing time that would be required. The method of conjugate directions has been improved in recent years and is computationally superior to the method of steepest descent used in SOLGASMIX¹⁴.

If computing power is available the direct-search method may be advantageous. It is easy to program, and the solution can be improved by choosing an increasingly fine mesh. The approach would be to minimize the system free energy (equation [52]) and the solution space is given in equation [60]. It is clear that an engineering workstation would be necessary to obtain a solution in a reasonable time.

If one would prefer to use commercial software, then Aspen Plus Release 9 would be useful for modeling this type of system. The RGIBBS model in Aspen can be used to make a reactors-in-series model. The furnace can be modeled using a flow-sheet interface, and heat flows can be incorporated. We can model the reactor as several stages in series. The temperature and pressure of each stage is given by the user, and energy and mass can flow into or out of each stage. The RGIBBS model allows one to calculate the equilibrium composition by free energy minimization, or by the extent of reaction method.

¹² Eriksson, G., *Chem. Scr.* 8 (1975) 100-103

¹³ Fletcher, R. and Reeves, C., *Computer J.* 7 (1964) 149-154

¹⁴ Hanna, O. and Sandall, O., *Computational Methods in Chemical Engineering*, Prentice Hall PTR, Upper Saddle River, New Jersey, 1995, p. 209; Edgar, T. and Himmelblau, D., *Optimization of Chemical Processes*, McGraw-Hill Book Company, New York, 1988, p. 206; and Reklaitis, G., Ravindran, A. and Ragsdell, K., *Engineering Optimization: Methods and Applications*, John Wiley and Sons, New York, 1983, p.107

APPENDIX

All reaction extent models are programs written in Matlab™, version 4.2c1 for Windows™. Matlab is a numerical package with its own data structures and programming language, but the program content and structure can be adapted for other platforms. The equilibrium reaction extent model, blow-hole model, and enthalpy-limited model use Matlab m-files (programs), called "Stages.m," "Blowhole.m," and "Stagesh.m," respectively. The Last Gas Stage Model uses the same code as the equilibrium reaction extent model. These three files call other functions (subroutines) to do such things as interpolate thermodynamic data or calculate reaction extents. "Gcalc.m" and "Hcalc.m" are the thermodynamic data functions. "SiO2C.m," "SiO2Si.m," "SiO2SiC.m," "CSi.m," "CSiC.m," and "SiSiC.m" are the functions that calculate binary equilibrium reaction extents, given temperature and inlet moles. "Invar124.m" and "Invar134.m" are the functions that calculate two simultaneous reaction extents at the invariant points. "SiO2Ch.m," "SiO2SiCh.m," and "Invarh.m" are the corresponding functions for the enthalpy-limited case. Finally, "Phase.m" is included as the program that generates the equilibrium diagram.

Please note that this is a working document rather than a commercial software package. The program code is provided to illustrate the modeling approach employed for this one system. For those experienced in doing thermodynamic equilibrium calculations of this type, it may be quite readily adopted for their purposes. For those less experienced in doing calculations of this type, a major investment of time may be required. We would certainly be willing to explore possible ways in which we could help in the full utilization of this approach.

Stages.m

```

% Equilibrium Furnace Composition Calculation
% Assumes 1 mole SiO2 descending into furnace
r=input('Enter C to SiO2 molar ratio in charge ');
% Top and Bottom "Border Stages" and User-Specified Number of Real Stages,
% at Temperature Intervals bunched towards hotter end of zone
% Code vector for each stage:
% 1=Top Border, 2=SiO2+C 3=SiO2+C+SiC (invariant)
% 4=SiO2+SiC 5=SiO2+Si+SiC (invariant) 6=Bottom Border
% Old is a matrix of output flowrates used to test convergence
Tlast=input('Enter Temperature of Last Gas Equilibrium Stage [K] ');
n1=input('How Many Stages Between Last Gas Eq. Stg. and First Invariant
Point? ');
n2=input('How Many Stages Between First and Second Invariant Points? ');
for i=1:n1+n2+7;
    old(i,1:6)=[0 0 0 0 0 0];
    if i==1
        StgTemp(1)=300;
        Code(1)=1;
    elseif i==2
        StgTemp(2)=Tlast;
        Code(2)=2;
    elseif i<n1+3
        StgTemp(i)=(StgTemp(i-1)+1788)/2;
        Code(i)=2;
    elseif i==n1+3
        StgTemp(i)=1788;
        Code(i)=2;
    elseif i==n1+4
        StgTemp(i)=1788;
        Code(i)=3;
    elseif i<n1+n2+5
        StgTemp(i)=1788+(2084-1788)*(i-n1-4)/(n2+1);
        Code(i)=4;
    elseif i==n1+n2+5;
        StgTemp(i)=2084;
        Code(i)=4;
    elseif i==n1+n2+6;
        StgTemp(i)=2084;
        Code(i)=5;
    else
        StgTemp(i)=2084;
        Code(i)=6;
    end
end
% Initial Guess for Stage Outputs - Depends on C to SiO2 ratio
% Columns are SiO2, C, Si, SiC, SiO, CO, respectively
if r<2
    for i=1:length(StgTemp);
        if (Code(i)==1)|(Code(i)==2)
            n(i,1:6)=[1 r 0 0 0 2*r/3];
        elseif Code(i)==3

```

APPENDIX

```

        n(i,1:6)=[1-r/3 0 0 r/3 0 2*r/3];
    elseif (Code(i)==4)|(Code(i)==5)
        n(i,1:6)=[1-r/3 0 0 r/3 2*r/3 r/3];
    elseif Code(i)==6
        n(i,1:6)=[1-r/2 0 r/2 0 0 0];
    end
end
elseif r<3
    for i=1:length(StgTemp);
        if (Code(i)==1)|(Code(i)==2)
            n(i,1:6)=[1 r 0 0 0 2*r/3];
        elseif Code(i)==3
            n(i,1:6)=[1-r/3 0 0 r/3 0 2*r/3];
        elseif (Code(i)==4)|(Code(i)==5)
            n(i,1:6)=[0 0 3-r r-2 4-4*r/3 2-2*r/3];
        elseif Code(i)==6
            n(i,1:6)=[0 0 3-r r-2 0 0];
        end
    end
    end
    % If r is over 3, C and SiC will be stable phases leaving invariant
    point 1, so no Si will form
end
R=0.008314;
tol=input('Enter Convergence Tolerance ');
flag=0; % Overall Convergence Flag
j=1; % Iteration Index
while flag==0
    % Reference Vector for plotting iteration progress
    J(j)=j;
    % Go through each stage in turn, calling appropriate function
    for i=1:length(StgTemp);
        if Code(i)==1
            n(i,1:6)=[1 r 0 0 n(i+1,5) n(i+1,6)];
            eq(i)=1;
        elseif Code(i)==2
            out=SiO2C(StgTemp(i),[n(i-1,1) n(i-1,2) n(i-1,3)
                n(i-1,4) n(i+1,5) n(i+1,6)]);
            n(i,1:6)=out(1:6);
            eq(i)=out(7);
        elseif Code(i)==3
            out=invar124([n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4)
                n(i+1,5) n(i+1,6)]);
            n(i,1:6)=out(1:6);
            eq(i)=out(7);
        elseif Code(i)==4
            out=SiO2SiC(StgTemp(i),[n(i-1,1) n(i-1,2) n(i-1,3)
                n(i-1,4) n(i+1,5) n(i+1,6)]);
            n(i,1:6)=out(1:6);
            eq(i)=out(7);
        elseif Code(i)==5
            out=invar134([n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4)
                n(i+1,5) n(i+1,6)]);
            n(i,1:6)=out(1:6);
            eq(i)=out(7);
        end
    end
end

```

APPENDIX

```
elseif Code(i)==6;
    n(i,1:6)=[n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4) 0 0];
    eq(i)=1;
end
end
% Check to see if calculations have converged
if max(max(abs(n-old)))<tol
    flag=1;
end
old=n;
j=j+1;
end
```

APPENDIX

Blowhole.m

```

% Equilibrium Furnace Composition Calculation
% Assumes 1 mole SiO2 descending into furnace
r=input('Enter C to SiO2 molar ratio in charge ');
% Top and Bottom "Border Stages" and User-Specified Number of Real Stages,
% at Temperature Intervals bunched towards hotter end of zone
% Code vector for each stage:
% 1=Top Border, 2=SiO2+C 3=SiO2+C+SiC (invariant)
% 4=SiO2+SiC 5=SiO2+Si+SiC (invariant) 6=Bottom Border 7=Blow-Hole
% Old is a matrix of output flowrates used to test convergence
Tlast=input('Enter Temperature of Last Gas Equilibrium Stage [K] ');
n1=input('How Many Stages Between Last Gas Eq. Stg. and First Invariant
Point? ');
n2=input('How Many Stages Between First and Second Invariant Points? ');
n3=input('How Many Stages Down from First Invariant Point is Blow-Hole? ');
for i=1:n1+n2+7;
    old(i,1:6)=[0 0 0 0 0 0];
    if i==1
        StgTemp(1)=300;
        Code(1)=1;
    elseif i==2
        StgTemp(2)=Tlast;
        Code(2)=2;
    elseif i<n1+3
        StgTemp(i)=(StgTemp(i-1)+1788)/2;
        Code(i)=2;
    elseif i==n1+3
        StgTemp(i)=1788;
        Code(i)=2;
    elseif i==n1+4
        StgTemp(i)=1788;
        Code(i)=3;
    elseif i<n1+n2+5
        StgTemp(i)=1788+(2084-1788)*(i-n1-4)/(n2+1);
        if i==n1+n3+4
            Code(i)=7;
        else
            Code(i)=4;
        end
    elseif i==n1+n2+5;
        StgTemp(i)=2084;
        Code(i)=4;
    elseif i==n1+n2+6;
        StgTemp(i)=2084;
        Code(i)=5;
    else
        StgTemp(i)=2084;
        Code(i)=6;
    end
end
% Initial Guess for Stage Outputs - Depends on C to SiO2 ratio
% Columns are SiO2, C, Si, SiC, SiO, CO, respectively

```

```

if r<2
    for i=1:length(StgTemp);
        if (Code(i)==1)|(Code(i)==2)
            n(i,1:6)=[1 r 0 0 0 2*r/3];
        elseif Code(i)==3
            n(i,1:6)=[1-r/3 0 0 r/3 0 2*r/3];
        elseif (Code(i)==4)|(Code(i)==5)
            n(i,1:6)=[1-r/3 0 0 r/3 2*r/3 r/3];
        elseif Code(i)==6
            n(i,1:6)=[1-r/2 0 r/2 0 0 0];
        end
    end
elseif r<3
    for i=1:length(StgTemp);
        if (Code(i)==1)|(Code(i)==2)
            n(i,1:6)=[1 r 0 0 0 2*r/3];
        elseif Code(i)==3
            n(i,1:6)=[1-r/3 0 0 r/3 0 2*r/3];
        elseif (Code(i)==4)|(Code(i)==5)
            n(i,1:6)=[0 0 3-r r-2 4-4*r/3 2-2*r/3];
        elseif Code(i)==6
            n(i,1:6)=[0 0 3-r r-2 0 0];
        end
    end
    end
    % If r is over 3, C and SiC will be stable phases leaving invariant
    % point 1, so no Si will form
end
R=0.008314;
tol=input('Enter Convergence Tolerance ');
flag=0; % Overall Convergence Flag
j=1; % Iteration Index
while flag==0
    % Reference Vector for plotting iteration progress
    J(j)=j;
    % Go through each stage in turn, calling appropriate function
    for i=1:length(StgTemp);
        if Code(i)==1
            n(i,1:6)=[1 r 0 0 n(i+1,5)+n(n1+n3+4,5)
                    n(i+1,6)+n(n1+n3+4,6)];
            eq(i)=1;
        elseif Code(i)==2
            out=SiO2C(StgTemp(i),[n(i-1,1) n(i-1,2) n(i-1,3)
                                   n(i-1,4) n(i+1,5) n(i+1,6)]);
            n(i,1:6)=out(1:6);
            eq(i)=out(7);
        elseif Code(i)==3
            out=invar124([n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4)
                          n(i+1,5) n(i+1,6)]);
            n(i,1:6)=out(1:6);
            eq(i)=out(7);
        elseif Code(i)==4
            out=SiO2SiC(StgTemp(i),[n(i-1,1) n(i-1,2) n(i-1,3)
                                      n(i-1,4) n(i+1,5) n(i+1,6)]);
            n(i,1:6)=out(1:6);

```

APPENDIX

```

        eq(i)=out(7);
    elseif Code(i)==5
        out=invar134([n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4)
            n(i+1,5) n(i+1,6)]);
        n(i,1:6)=out(1:6);
        eq(i)=out(7);
    elseif Code(i)==6
        n(i,1:6)=[n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4) 0 0];
        eq(i)=1;
    elseif Code(i)==7
        out=SiO2SiC(StgTemp(i),[n(i-1,1) n(i-1,2) n(i-1,3)
            n(i-1,4) n(i+1,5) n(i+1,6)]);
        n(i,1:6)=[out(1) out(2) out(3) out(4) 0.5*out(5)
            0.5*out(6)];
        eq(i)=0;
    end
end
% Check to see if calculations have converged
if max(max(abs(n-old)))<tol
    flag=1;
end
old=n
j=j+1
end

```


APPENDIX

Stagesh.m

```

% Enthalpy-Controlled Non-Equilibrium Furnace Composition Calculation
% Assumes 1 mole SiO2 descending into furnace
r=input('Enter C to SiO2 molar ratio in charge ');
% Top and Bottom "Border Stages" and User-Specified Number of Real Stages,
% at Temperature Intervals bunched towards hotter end of zone
% Code vector for each stage:
% No SiC will be formed except in arc stage, so no SiC-involving equilibria
% or true invariant points exist. In arc stage SiO2,C-->Two of (SiO2,C,Si,SiC)
% 1=Top Border, 2=SiO2+C
% 5=Bottom "Invariant" 6=Bottom Border
% Old is a matrix of output flowrates used to test convergence
nstg=input('How Many Stages Between 300K and Arc Stage (2084K) ? ');
for i=1:nstg+3;
    old(i,1:6)=[0 0 0 0 0 0];
    if i==1
        StgTemp(1)=300;
        Code(1)=1;
    elseif i<nstg+2;
        StgTemp(i)=300+(2084-300)*(i-2)/nstg;
        Code(i)=2;
    elseif i==nstg+2
        StgTemp(i)=2084;
        Code(i)=5;
    else
        StgTemp(i)=2084;
        Code(i)=6;
    end
end
% Initial Guess for Stage Outputs - Depends on C to SiO2 ratio
% Columns are SiO2, C, Si, SiC, SiO, CO, respectively
for i=1:length(StgTemp);
    if (Code(i)==1)|(Code(i)==2)
        n(i,1:6)=[1 r 0 0 0 2*r/3];
    elseif Code(i)==5
        n(i,1:6)=[1-r/3 0 0 r/3 2*r/3 r/3];
    elseif Code(i)==6
        n(i,1:6)=[1-r/2 0 r/2 0 0 0];
    end
end
R=0.008314;
tol=input('Enter Convergence Tolerance ');
flag=0;          % Overall Convergence Flag
j=1;             % Iteration Index
while flag==0
    % Reference Vector for plotting iteration progress
    J(j)=j;
    % Go through each stage in turn, calling appropriate function
    for i=1:length(StgTemp);
        if Code(i)==1
            n(i,1:6)=[1 r 0 0 n(i+1,5) n(i+1,6)];
            eq(i)=0;
        end
    end
    flag=1;
    j=j+1;
end

```

APPENDIX

```

elseif Code(i)==2
    out=SiO2Ch(StgTemp(i-1),StgTemp(i),StgTemp(i+1),
        [n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4) n(i+1,5)
        n(i+1,6)]);
    n(i,1:6)=out(1:6);
    eq(i)=0;
elseif Code(i)==5
    % Non-Equilibrium SiO2 and C mixture reacts to
    % give Equilibrium Product (SiC) now that heat supply
    % is available
    % Operative Reaction
    %  $\text{SiO}_2 + 3 \text{C} = \text{SiC} + 2 \text{CO}$ 
    if n(i-1,2)/n(i-1,1)<3
        % C limiting so SiO2 and SiC produced
        e=n(i-1,2)/3;
        n(i,1:6)=[n(i-1,1)-e 0 0 e 0 2*e];
        % Then adjust for SiO2 - SiC Eq PSiO
        out=SiO2SiC(StgTemp(i),n(i,1:6));
        n(i,1:6)=out(1:6);
        eq(i)=out(7);
    elseif n(i-1,2)/n(i-1,1)>3
        % SiO2 limiting so C and SiC produced
        e=n(i-1,1);
        n(i,1:6)=[0 n(i-1,2)-3*e 0 e 0 2*e];
        % Then adjust for Si - SiC Eq PSiO
        out=SiSiC(StgTemp(i),n(i,1:6));
        n(i,1:6)=out(1:6);
        eq(i)=out(7);
    else
        SiO2Csto=1
    end
elseif Code(i)==6;
    n(i,1:6)=[n(i-1,1) n(i-1,2) n(i-1,3) n(i-1,4) 0 0];
    eq(i)=1;
end
end
% Check to see if calculations have converged
if max(max(abs(n-old)))<tol
    flag=1;
end
old=n;
j=j+1;
end

```

APPENDIX

Gcalc.m

```

function G=G(T);
% Gibbs Free Energy Calculating Function for use with Si-O-C System
% T in K
% Gf in kJ/mol
Temp=1e2.*[3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26
27 28];
Si=[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
SiC=[-70.835 -70.018 -69.187 -68.368 -67.566 -66.780 -66.007 -65.245 -64.491
-63.741 -62.993 -62.246 -61.497 -60.746 -59.544 -55.813 -52.089
-48.375 -44.669 -40.973 -37.287 -33.611 -29.946 -26.292 -22.649
-19.017];
SiO=[-127.469 -136.401 -145.181 -153.831 -162.368 -170.804 -179.147 -187.404
-195.580 -203.680 -211.707 -219.663 -227.551 -235.373 -242.684
-247.406 -252.081 -256.709 -261.294 -265.837 -270.339 -274.802
-279.228 -283.617 -287.972 -292.292];
SiO2=[-856.106 -837.813 -819.540 -801.367 -783.331 -765.451 -747.785 -730.256
-712.844 -695.654 -678.536 -661.483 -644.487 -627.545 -610.202
-590.378 -570.606 -550.902 -531.735 -512.666 -493.690 -474.802
-455.997 -437.271 -418.620 -400.040];
C=[0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0];
CO=[-137.345 -146.354 -155.426 -164.494 -173.522 -182.494 -191.408 -200.261
-209.056 -217.796 -226.482 -235.118 -243.707 -252.250 -260.751
-269.210 -277.631 -286.015 -294.362 -302.675 -310.955 -319.203
-327.419 -335.605 -343.762 -351.889];
TindexL=floor((T-200)/100);
TindexH=ceil((T-200)/100);
frac=(T-Temp(TindexL))/100;
G(1)=SiO2(TindexL)+frac*(SiO2(TindexH)-SiO2(TindexL));
G(2)=C(TindexL)+frac*(C(TindexH)-C(TindexL));
G(3)=Si(TindexL)+frac*(Si(TindexH)-Si(TindexL));
G(4)=SiC(TindexL)+frac*(SiC(TindexH)-SiC(TindexL));
G(5)=SiO(TindexL)+frac*(SiO(TindexH)-SiO(TindexL));
G(6)=CO(TindexL)+frac*(CO(TindexH)-CO(TindexL));

```

APPENDIX

Hcalc.m

```

function H=H(T);
% Enthalpy Calculating Function for use with Si-O-C System
% T in K
% H in kJ/mol
Temp=1e2.*[3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23];
Si=[0.037 2.159 4.438 6.815 9.264 11.772 14.333 16.943 19.598 22.297 25.039
    27.823 30.648 33.513 36.597 39.316 42.036 44.756 47.475 50.195
    52.914];
SiC=[-73.170 -70.129 -66.499 -62.452 -58.115 -53.573 -48.877 -44.063 -39.155
    -34.171 -29.121 -24.016 -18.862 -13.664 -8.426 -3.151 2.158 7.499
    12.870 18.270 23.697];
SiO=[-100.361 -97.319 -94.143 -90.848 -87.463 -84.008 -80.5 -76.949 -73.364
    -69.750 -66.113 -62.456 -58.783 -55.095 -51.395 -47.686 -43.967
    -40.241 -36.508 -32.770 -29.026];
SiO2=[-910.774 -905.840 -900.170 -893.959 -887.301 -880.186 -872.343 -865.498
    -856.539 -849.427 -842.254 -835.03 -827.762 -820.458 -813.12 -805.752
    -798.357 -781.326 -772.749 -764.172 -755.594];
C=[0.016 1.053 2.384 3.964 5.742 7.666 9.699 11.818 14.007 16.253 18.545
    20.874 23.233 25.617 28.021 30.441 32.876 35.323 37.783 40.252
    42.731];
CO=[-110.487 -107.564 -104.609 -101.599 -98.518 -95.364 -92.140 -88.851
    -85.506 -82.111 -78.674 -75.199 -71.691 -68.156 -64.596 -61.015
    -57.415 -53.797 -50.165 -46.52 -42.862];
TindexL=floor((T-200)/100);
TindexH=ceil((T-200)/100);
frac=(T-Temp(TindexL))/100;
H(1)=SiO2(TindexL)+frac*(SiO2(TindexH)-SiO2(TindexL));
H(2)=C(TindexL)+frac*(C(TindexH)-C(TindexL));
H(3)=Si(TindexL)+frac*(Si(TindexH)-Si(TindexL));
H(4)=SiC(TindexL)+frac*(SiC(TindexH)-SiC(TindexL));
H(5)=SiO(TindexL)+frac*(SiO(TindexH)-SiO(TindexL));
H(6)=CO(TindexL)+frac*(CO(TindexH)-CO(TindexL));

```

SiO2C.m

```

% Function returns outlet mole numbers for equilibrium between
% SiO2 and C, given temperature and inlet mole numbers
% Operative Reaction is SiO2 + C = SiO + CO
function out=SiO2C(T,in)
G=gcalc(T);
DG=G(5)+G(6)-G(1)-G(2);
K=exp(-DG/(0.008314*T));
P=min(roots([-1 1 -K]));
e=(in(5)*(P-1)+in(6)*P)/(1-2*P);
out=[in(1)-e in(2)-e 0 0 in(5)+e in(6)+e 1];
% If negative, reaction cannot proceed to equilibrium because one reactant
% is limiting
if (min(out(1:6))<0)&(e>0)
    if in(1)>in(2)
        % C limiting
        e=in(2);
        out=[in(1)-e 0 0 0 in(5)+e in(6)+e 0];
    elseif in(1)<in(2)
        % SiO2 limiting
        e=in(1);
        out=[0 in(2)-e 0 0 in(5)+e in(6)+e 0];
    end
elseif (min(out(1:6))<0)&(e<0)
    if in(5)>in(6)
        % CO limiting
        e=-in(6);
        out=[in(1)-e in(2)-e 0 0 in(5)+e 0 0];
    elseif in(5)<in(6)
        % SiO limiting
        e=-in(5);
        out=[in(1)-e in(2)-e 0 0 0 in(6)+e 0];
    end
end
end

```

APPENDIX

SiO2Si.m

```
% Function returns outlet mole numbers for equilibrium between
% SiO2 and Si, given temperature and inlet mole numbers
% Operative Reaction is  $\text{SiO}_2 + \text{Si} = 2 \text{SiO}$ 
function out=SiO2Si(T,in)
G=gcalc(T);
DG=2*G(5)-G(1)-G(3);
K=exp(-DG/(0.008314*T));
P=sqrt(K);
e=(in(5)*(P-1)+in(6)*P)/(2-2*P);
out=[in(1)-e 0 in(3)-e 0 in(5)+2*e in(6) 1];
% If negative, reaction cannot proceed to equilibrium because one reactant
% is limiting
if (min(out(1:6))<0)&(e>0)
    if in(1)>in(3)
        % Si limiting
        e=in(3);
        out=[in(1)-e 0 0 0 in(5)+2*e in(6) 0];
    elseif in(1)<in(3)
        % SiO2 limiting
        e=in(1);
        out=[0 0 in(3)-e 0 in(5)+2*e in(6) 0];
    end
elseif (min(out(1:6))<0)&(e<0)
    % SiO limiting
    e=-in(5)/2;
    out=[in(1)-e 0 in(3)-e 0 0 in(6) 0];
end
end
```

SiO2SiC.m

```

% Function returns outlet mole numbers for equilibrium between
% SiO2 and SiC, given temperature and inlet mole numbers
% Operative Reaction is 2 SiO2 + SiC = 3 SiO + CO
function out=SiO2SiC(T,in)
G=gcalc(T);
DG=3*G(5)+G(6)-2*G(1)-G(4);
K=exp(-DG/(0.008314*T));
junk=roots([-1 1 0 0 -K]);
count=1;
for k=1:4;
    if junk(k)==real(junk(k))
        realP(count)=junk(k);
        count=count+1;
    end
end
P=min(realP);
e=(in(5)*(P-1)+in(6)*P)/(3-4*P);
out=[in(1)-2*e 0 0 in(4)-e in(5)+3*e in(6)+e 1];
% If negative, reaction cannot proceed to equilibrium because one reactant
% is limiting
if (min(out(1:6))<0)&(e>0)
    if in(1)>2*in(4)
        % SiC limiting
        e=in(4);
        out=[in(1)-2*e 0 0 0 in(5)+3*e in(6)+e 0];
    elseif in(1)<2*in(4)
        % SiO2 limiting
        e=in(1)/2;
        out=[0 0 0 in(4)-e in(5)+3*e in(6)+e 0];
    end
elseif (min(out(1:6))<0)&(e<0)
    if in(5)>3*in(6)
        % CO limiting
        e=-in(6);
        out=[in(1)-2*e 0 0 in(4)-e in(5)+3*e 0 0];
    elseif in(5)<3*in(6)
        % SiO limiting
        e=-in(5)/3;
        out=[in(1)-e 0 0 in(4)-e 0 in(6)+e 0];
    end
end
end

```

CSi.m

```

% Function returns outlet mole numbers for equilibrium between
% C and Si, given temperature and inlet mole numbers
% Operative Reaction is Si + CO = SiO + C
function out=CSi(T,in)
G=gcalc(T);
DG=G(2)+G(5)-G(3)-G(6);
K=exp(-DG/0.008314*T);
P=K/(1+K);
e=in(5)*(P-1)+in(6)*P;
out=[0 in(2)+e in(3)-e 0 in(5)+e in(6)-e 1];
% If negative, reaction cannot proceed to equilibrium because one reactant
% is limiting
if (min(out(1:6))<0)&(e>0)
    if in(3)>in(6)
        % CO limiting
        e=in(6);
        out=[in(1) in(2)+e in(3)-e 0 in(5)+e 0 0];
    elseif in(3)<in(6)
        % Si limiting
        e=in(3);
        out=[in(1) in(2)+e 0 0 in(5)+e in(6)-e 0];
    end
elseif (min(out(1:6))<0)&(e<0)
    if in(5)>in(2)
        % C limiting
        e=-in(2);
        out=[in(1) 0 in(3)-e 0 in(5)+e in(6)-e 0];
    elseif in(5)<in(2)
        % SiO limiting
        e=-in(5);
        out=[in(1) in(2)+e in(3)-e 0 0 in(6)-e 0];
    end
end
end

```


APPENDIX

CSiC.m

```
% Function returns outlet mole numbers for equilibrium between
% C and SiC, given temperature and inlet mole numbers
% Operative Reaction is SiC + CO = SiO + 2 C
function out=CSiC(T,in)
G=gcalc(T);
DG=2*G(2)+G(5)-G(4)-G(6);
K=exp(-DG/(0.008314*T));
P=K/(1+K);
e=in(5)*(P-1)+in(6)*P;
out=[0 in(2)+2*e 0 in(4)-e in(5)+e in(6)-e 1];
% If negative, reaction cannot proceed to equilibrium because one reactant
% is limiting
if (min(out(1:6))<0)&(e>0)
    if in(4)>in(6)
        % CO limiting
        e=in(6);
        out=[0 in(2)+2*e 0 in(4)-e in(5)+e 0 0];
    elseif in(4)<in(6)
        % SiC limiting
        e=in(4);
        out=[0 in(2)+2*e 0 0 in(5)+e in(6)-e 0];
    end
elseif (min(out(1:6))<0)&(e<0)
    if in(5)>2*in(2)
        % C limiting
        e=-in(2)/2;
        out=[0 0 0 in(4)-e in(5)+e in(6)-e 0];
    elseif in(5)<2*in(2)
        % SiO limiting
        e=-in(5);
        out=[0 in(2)+2*e 0 in(4)-e 0 in(6)-e 0];
    end
end
end
```

APPENDIX

SiSiC.m

```
% Function returns outlet mole numbers for equilibrium between
% Si and SiC, given temperature and inlet mole numbers
% Operative Reaction is 2 Si + CO = SiC + SiO
function out=SiSiC(T,in)
G=gcalc(T);
DG=G(4)+G(5)-2*G(3)-G(6);
K=exp(-DG/(0.008314*T));
P=K/(1+K);
e=in(5)*(P-1)+in(6)*P;
out=[0 0 in(3)-2*e in(4)+e in(5)+e in(6)-e 1];
% If negative, reaction cannot proceed to equilibrium because one reactant
% is limiting
if (min(out(1:6))<0)&(e>0)
    if in(3)>2*in(6)
        % CO limiting
        e=in(6);
        out=[0 0 in(3)-2*e in(4)+e in(5)+e 0 0];
    elseif in(3)<2*in(2)
        % Si limiting
        e=in(3)/2;
        out=[0 0 0 in(4)+e in(5)+e in(6)-e 0];
    end
elseif (min(out(1:6))<0)&(e<0)
    if in(4)>in(5)
        % SiO limiting
        e=-in(5);
        out=[0 0 in(3)-2*e in(4)+e 0 in(6)-e 0];
    elseif in(4)<in(5)
        % SiC limiting
        e=-in(4);
        out=[0 0 in(3)-2*e 0 in(5)+e in(6)-e 0];
    end
end
end
```

APPENDIX

Invar124.m

```
% Function returns outlet mole numbers for equilibrium between
% SiO2, C, and SiC at invariant point (1788K, 5.369e-3 atm SiO),
% given inlet mole numbers.
function out=invar124(in)
P=5.369e-3;
% Operative Reactions depend upon which reactants are present in the feed
if (in(1)~=0)&(in(2)~=0)&(in(3)==0)&(in(4)==0)
    % SiO2 and C react to give SiC and which ever reactant is in excess
    % Operative Reaction are:
    %  $\text{SiO}_2 + 3 \text{C} = \text{SiC} + 2 \text{CO}$  and
    %  $\text{SiC} + \text{CO} = \text{SiO} + 2 \text{C}$ 
    % Reaction Extents depend upon which reactant is limiting
    % Assume C is limiting and calculate output
    e1=(in(2)/2+in(5)*(P-1)+in(6)*P)/(1.5-2*P);
    e2=-in(2)/2+1.5*e1;
    out=[in(1)-e1 0 0 in(4)+e1-e2 in(5)+e2 in(6)+2*e1-e2 1];
    % If negative mole numbers, SiO2 is limiting so recalculate
    if min(out(1:6))<0
        e1=in(1);
        e2=in(1)*2*P+in(5)*(P-1)+in(6)*P;
        out=[0 in(2)-3*e1+2*e2 0 in(4)+e1-e2 in(5)+e2 in(6)+2*e1-e2
            1];
    end
elseif (in(1)~=0)&(in(2)==0)&(in(3)==0)&(in(4)~=0)
    % SiO2 and SiC react to give C and which ever reactant is in excess
    % Operative Reaction are:
    %  $\text{SiO}_2 + \text{SiC} = \text{C} + 2 \text{SiO}$  and
    %  $\text{SiC} + \text{CO} = \text{SiO} + 2 \text{C}$ 
    % Reaction Extents depend upon which reactant is limiting
    % Assume SiO2 is limiting and calculate output
    e1=in(1);
    e2=in(1)*(2*P-2)+in(5)*(P-1)+in(6)*P;
    out=[0 in(2)+e1+2*e2 0 in(4)-e1-e2 in(5)+2*e1+e2 in(6)-e2 1];
    % If negative mole numbers, SiC is limiting so recalculate
    if min(out(1:6))<0
        e1=(-in(4)+in(5)*(P-1)+in(6)*P)/(1-2*P);
        e2=in(4)-e1;
        out=[in(1)-e1 in(2)+e1+2*e2 0 0 in(5)+2*e1+e2 in(6)-e2 1];
    end
elseif (in(1)==0)&(in(2)~=0)&(in(3)==0)&(in(4)~=0)
    % C and SiC react to give SiO2 and which ever reactant is in excess
    % Operative Reaction are:
    %  $\text{SiO}_2 + \text{C} = \text{SiO} + \text{CO}$  and
    %  $\text{SiC} + \text{CO} = \text{SiO} + 2 \text{C}$ 
    % Reaction Extents depend upon which reactant is limiting
    % Assume C is limiting and calculate output
    e2=(in(2)*(2*P-1)+in(5)*(P-1)+in(6)*P)/(3-4*P);
    e1=in(2)+2*e2;
    out=[in(1)-e1 0 0 in(4)-e2 in(5)+e1+e2 in(6)+e1-e2 1];
    % If negative mole numbers, SiC is limiting so recalculate
    if min(out(1:6))<0
```

APPENDIX

```
e1=(-in(4)+in(5)*(P-1)+in(6)*P)/(1-2*P);
e2=in(4);
out=[in(1)-e1 in(2)-e1+2*e2 0 0 in(5)+e1+e2 in(6)+e1-e2 1];
end
else
    % System is not in equilibrium
    out=[NaN NaN NaN NaN NaN NaN 0];
end
% Check for negative mole numbers
if min(out(1:6))<0
    out=[NaN NaN NaN NaN NaN NaN 0];
end
end
```

Invar134.m

```

% Function returns outlet mole numbers for equilibrium between
% SiO2, Si, and SiC at invariant point (2084K, 0.6760 atm SiO),
% given inlet mole numbers.
function out=invar134(in)
P=0.6760;
% Operative Reactions depend upon which reactants are present in the feed
if (in(1)~=0)&(in(2)==0)&(in(3)~=0)&(in(4)==0)
    % SiO2 and Si react to give SiC and which ever reactant is in excess
    % Operative Reaction are:
    %  $\text{SiO}_2 + \text{Si} = 2 \text{SiO}$  and
    %  $2 \text{SiO}_2 + \text{SiC} = 3 \text{SiO} + \text{CO}$ 
    % Reaction Extents depend upon which reactant is limiting
    % Assume SiO2 is limiting and calculate output
    e2=(in(1)*(2*P-2)+in(5)*(P-1)+in(6)*P)/(3-4*P);
    e1=in(1)-2*e2;
    out=[0 0 in(3)-e1 in(4)-e2 in(5)+2*e1+3*e2 in(6)+e2 1];
    % If negative mole numbers, Si is limiting so recalculate
    if min(out(1:6))<0
        e1=in(3);
        e2=(in(3)*(2*P-2)+in(5)*(P-1)+in(6)*P)/(3-4*P);
        out=[in(1)-e1-2*e2 0 0 in(4)-e2 in(5)+2*e1+3*e2 in(6)+e2 1];
    end
elseif (in(1)~=0)&(in(2)==0)&(in(3)==0)&(in(4)~=0)
    % SiO2 and SiC react to give Si and which ever reactant is in excess
    % Operative Reaction are:
    %  $\text{SiO}_2 + 2 \text{SiC} = 3 \text{Si} + 2 \text{CO}$  and
    %  $2 \text{Si} + \text{CO} = \text{SiC} + \text{SiO}$ 
    % Reaction Extents depend upon which reactant is limiting
    % Assume SiO2 is limiting and calculate output
    e1=in(1);
    e2=in(1)*2*P+in(5)*(P-1)+in(6)*P;
    out=[0 0 in(3)+3*e1-2*e2 in(4)-2*e1+e2 in(5)+e2 in(6)+2*e1-e2 1];
    % If negative mole numbers, SiC is limiting so recalculate
    if min(out(1:6))<0
        e1=(in(4)+in(5)*(P-1)+in(6)*P)/(2-2*P);
        e2=2*e1-in(4);
        out=[in(1)-e1 0 in(3)+3*e1-2*e2 0 in(5)+e2 in(6)+2*e1-e2 1];
    end
elseif (in(1)==0)&(in(2)==0)&(in(3)~=0)&(in(4)~=0)
    % Si and SiC react to give SiO2 and which ever reactant is in excess
    % Operative Reaction are:
    %  $\text{SiO}_2 + \text{Si} = 2 \text{SiO}$  and
    %  $2 \text{Si} + \text{CO} = \text{SiC} + \text{SiO}$ 
    % Reaction Extents depend upon which reactant is limiting
    % Assume Si is limiting and calculate output
    e2=(in(3)*(2*P-2)+in(5)*(P-1)+in(6)*P)/(4*P-3);
    e1=in(3)-2*e2;
    out=[in(1)-e1 0 0 in(4)+e2 in(5)+2*e1+e2 in(6)-e2 1];
    % If negative mole numbers, SiC is limiting so recalculate
    if min(out(1:6))<0
        e1=(in(4)+in(5)*(P-1)+in(6)*P)/(2-2*P);

```

APPENDIX

```
        e2=-in(4);
        out=[in(1)-e1 0 in(3)-e1-2*e2 0 in(5)+2*e1+e2 in(6)-e2 1];
    end
else
    % System is not in equilibrium
    out=[NaN NaN NaN NaN NaN NaN 0];
end
% Check for negative mole numbers
if min(out(1:6))<0
    out=[NaN NaN NaN NaN NaN NaN 0];
end
```

APPENDIX

SiO2Ch.m

```
% Function returns outlet mole numbers for enthalpy-controlled,  
% non-equilibrium reaction between SiO2 and C,  
% given temperature and inlet mole numbers  
% Operative Reaction is  $\text{SiO}_2 + \text{C} = \text{SiO} + \text{CO}$   
function out=SiO2CH(Ta,T,Tb,in)  
Ha=hcalc(Ta);  
H=hcalc(T);  
Hb=hcalc(Tb);  
e=(in(1)*(Ha(1)-H(1))+in(2)*(Ha(2)-H(2))+in(5)*(Hb(5)-H(5))+in(6)*  
    (Hb(6)-H(6)))/(-H(1)-H(2)+H(5)+H(6));  
out=[in(1)-e in(2)-e 0 0 in(5)+e in(6)+e e];
```

APPENDIX

SiO2SiCh.m

```
% Function returns outlet mole numbers for enthalpy-controlled,
% non-equilibrium between reaction between SiO2 and SiC,
% given temperature and inlet mole numbers
% Operative Reaction is 2 SiO2 + SiC = 3 SiO + CO
function out=SiO2SiCH(Ta,T,Tb,in)
Ha=hcalc(Ta);
H=hcalc(T);
Hb=hcalc(Tb);
e=(in(1)*(Ha(1)-H(1))+in(2)*(Ha(2)-H(2))+in(5)*(Hb(5)-H(5))+in(6)*
(Hb(6)-H(6)))/(-2*H(1)-H(4)+3*H(5)+H(6));
out=[in(1)-2*e 0 0 in(4)-e in(5)+3*e in(6)+e e];
```


APPENDIX

Invarh.m

```
% Function returns outlet mole numbers for enthalpy-controlled,
% non-equilibrium reaction between SiO2, C, and SiC
% at invariant point (1788K, 5.369e-3 atm SiO), given inlet mole numbers.
function out=invarh(in)
H=hcalc(1788);
% Assume SiO2 and C react to give SiC and which ever reactant is in excess
% Operative Reaction are:
%  $\text{SiO}_2 + 3 \text{C} = \text{SiC} + 2 \text{CO}$  and
%  $\text{SiC} + \text{CO} = \text{SiO} + 2 \text{C}$ 
% Reaction Extents depend upon which reactant is limiting
% Assume C is limiting and calculate output
e1=in(2)*(H(2)-0.5*H(4)+0.5*H(5)-0.5*H(6))/(-H(1)-0.5*H(4)+1.5*H(5)+0.5*H(6));
e2=-in(2)/2+1.5*e1;
out=[in(1)-e1 0 0 in(4)+e1-e2 in(5)+e2 in(6)+2*e1-e2 1];
% If negative mole numbers, SiO is limiting so recalculate
if min(out(1:6))<0
    e2=in(5);
    e1=in(5)*(-2*H(2)+H(4)-H(5)+H(6))/(-H(1)-3*H(2)+H(4)+2*H(6));
    out=[in(1)-e1 in(2)-3*e1+2*e2 0 in(4)+e1-e2 0 in(6)+2*e1-e2 1];
end
% Check for negative mole numbers
% SiC cannot be formed without absorbing heat, so no reaction will occur
if min(out(1:6))<0
    out=[NaN NaN NaN NaN NaN NaN 0];
end
```

APPENDIX

Phase.m

```
% Equilibrium Phase Diagram for Si-O-C System vs T and PSiO
% Total System Pressure (PSiO+PCO) is 1 atm
% Composed of Plots of PSiO vs T lines for gas above any of six possible
% mixtures of two condensed phases
% SiO2-C Equilibrium  $\text{SiO}_2 + \text{C} = \text{SiO} + \text{CO}$  Free Energy Change DG1
% SiO2-Si Equilibrium  $\text{SiO}_2 + \text{Si} = 2 \text{SiO}$  Free Energy Change DG2
% SiO2-SiC Equilibrium  $2 \text{SiO}_2 + \text{SiC} = 3 \text{SiO} + \text{CO}$  Free Energy Change DG3
% C-Si Equilibrium  $\text{C} + \text{SiO} = \text{Si} + \text{CO}$  Free Energy Change DG4
% C-SiC Equilibrium  $2 \text{C} + \text{SiO} = \text{SiC} + \text{CO}$  Free Energy Change DG5
% Si-SiC Equilibrium  $2 \text{Si} + \text{CO} = \text{SiC} + \text{SiO}$  Free Energy Change DG6
R=8.314e-3; % Gas Constant in [kJ/mol K]
for i=1:171;
    % Temperature Steps [K] - finer near invariant points
    if i<90
        T(i)=900+10*(i-1);
    elseif i<101
        T(i)=1783+(i-90);
    elseif i<114
        T(i)=1800+10*(i-101);
    elseif i<125
        T(i)=1922+(i-114);
    elseif i<139
        T(i)=1940+10*(i-125);
    elseif i<150
        T(i)=2079+(i-139);
    else
        T(i)=2090+10*(i-150);
    end
    G=Gcalc(T(i)); % Thermo Properties
    DG1(i)=G(5)+G(6)-G(1)-G(2); % Free
    DG2(i)=2*G(5)-G(1)-G(3); % Energy
    DG3(i)=3*G(5)+G(6)-2*G(1)-G(4); % Changes
    DG4(i)=G(3)+G(6)-G(2)-G(5); % Of
    DG5(i)=G(4)+G(6)-2*G(2)-G(5); % Reactions
    DG6(i)=G(4)+G(5)-2*G(3)-G(6); %
    K1(i)=exp(-DG1(i)/(R*T(i))); % Equilibrium
    K2(i)=exp(-DG2(i)/(R*T(i))); % Constants
    K3(i)=exp(-DG3(i)/(R*T(i))); % Of
    K4(i)=exp(-DG4(i)/(R*T(i))); % Reactions
    K5(i)=exp(-DG5(i)/(R*T(i))); %
    K6(i)=exp(-DG6(i)/(R*T(i))); %
    if K1(i)<=0.25 % Multiple PSiO possible
        SiO1(i,1:2)=roots([-1 1 -K1(i)]); % for Rxn 1
    else % Only those within
        SiO1(i,1:2)=[NaN NaN]; % bounds considered
    end
    SiO2(i)=sqrt(K2(i)); % Equilib PSiO for Rxn 2
    if K3(i)<=0.10546875 % Multiple PSiO possible
        junk=roots([-1 1 0 0 -K3(i)]); % for Rxn 3
        n=1; % Only those within
        for j=1:4; % bounds considered
```

APPENDIX

```

        if junk(j)==real(junk(j))
            SiO3(i,n)=junk(j);
            n=n+1;
        end
    end
else
    SiO3(i,1:4)=[NaN NaN NaN NaN];
end
SiO4(i)=1/(1+K4(i));           % Equilib PSiO for Rxn 4
SiO5(i)=1/(1+K5(i));           % Equilib PSiO for Rxn 5
SiO6(i)=K6(i)/(1+K6(i));       % Equilib PSiO for Rxn 6
end
% Break curves 1, 2, and 3 into stable and unstable Equilibria
for i=1:length(T);
    for j=1:2;
        if (SiO1(i,j)<=SiO5(i))&(SiO1(i,j)<=SiO6)
            T1s(i)=T(i);
            SiO1s(i,j)=SiO1(i,j);
        else
            T1u(i)=T(i);
            SiO1u(i,j)=SiO1(i,j);
        end
    end
    if T(i)>=2084
        T2s(i)=T(i);
        SiO2s(i)=SiO2(i);
    else
        T2u(i)=T(i);
        SiO2u(i)=SiO2(i);
    end
    for j=1:2;
        if (SiO3(i,j)>=SiO5(i))&(SiO3(i,j)<=SiO6(i))
            T3s(i)=T(i);
            SiO3s(i,j)=SiO3(i,j);
        else
            T3u(i)=T(i);
            SiO3u(i,j)=SiO3(i,j);
        end
    end
end
end
% Eq Curve 4 wholly unstable
T4s=0;
SiO4s=0;
T4u=T;
SiO4u=SiO4;
% Eq Curve 5 wholly stable
T5s=T;
SiO5s=SiO5;
T5u=0;
SiO5u=0;
% Eq Curve 6 wholly stable
T6s=T;
SiO6s=SiO6;
T6u=0;

```

APPENDIX

```
SiO6u=0;
semilogy(T1s, SiO1s, 'w', T2s, SiO2s, 'w', T3s, SiO3s, 'w', T4s, SiO4s, 'w',
          T5s, SiO5s, 'w', T6s, SiO6s, 'w', T1u, SiO1u, 'w:', T2u, SiO2u, 'w:',
          T3u, SiO3u, 'w:', T4u, SiO4u, 'w:', T5u, SiO5u, 'w:', T6u, SiO6u, 'w:')
axis([900 2300 1e-4 1])
xlabel('T [K]')
ylabel('PSiO [atm]')
title('Figure 1: Si-O-C Phase Equilibrium at P=1 atm')
```

REFERENCES

- Barin, I. Thermochemical Data of Pure Substances, Parts I and II. Weinheim, Germany: VCH, 1993.
- Brinkley, S., *J. Chem. Phys.* **14** (1946) 563-564.
- Brinkley, S., *J. Chem. Phys.* **15** (1947) 107-110.
- Edgar, T. and Himmelblau, D. Optimization of Chemical Processes. New York: McGraw-Hill, 1988.
- Eriksson, G., *Acta Chem. Scand.* **25** (1971) 2651-2658.
- Eriksson, G., *Chem. Scr.* **8** (1975) 100-103.
- Fletcher, R. and Reeves, C., *Computer J.* **7** (1964) 149-154.
- Gibbs, J. Collected Works of J. Willard Gibbs, Vol. 1. New York: Longmans, Green, and Company, 1928.
- Hanna, O. and Sandall, O. Computational Methods in Chemical Engineering. Upper Saddle River, New Jersey: Prentice Hall PTR, 1995.
- Larson, H. R. et al. "STCC Technical Report: D. C. Smelting of Silicon Metal and Ferro-Silicon Alloys in a 1.5 MW Furnace," 1995.
- Müller, M. B. et al. "Heat and Mass Transfer in the Ferrosilicon Process" in *Scandinavian Journal of Metallurgy*, Vol. 7, pgs. 145-155, 1972.
- Reklaitis, G., Ravindran, A. and Ragsdell, K. Engineering Optimization: Methods and Applications. New York: John Wiley and Sons, 1983.
- Rosenqvist. Principles of Extractive Metallurgy, 2nd Ed. New York: McGraw Hill, 1983.
- Schei, A. "A Stoichiometric Model of the Ferrosilicon Process" in 39th Electric Furnace Conference Proceedings, pgs. 301-309, Chelsea, Michigan: Bookcrafters, Inc., 1982.
- Selmer-Olsen, S. "The Silicon Metal Process, A Thermodynamical Study" in 39th Electric Furnace Conference Proceedings, pgs. 310-318, Chelsea, Michigan: Bookcrafters, Inc., 1982.
- Smith and Van Ness, "Introduction to Chemical Engineering Thermodynamics" 4th Ed. New York: McGraw Hill, 1987.
- White, W., Johnson, S. and Dantzig, G., *J. Chem. Phys.* **28** (1959) 251-255.

REFERENCES

Woolley, D. and Szekely, J. "Thermodynamic Study of Hollow Graphite Cathode Direct Current Plasma Arc Furnace For Carbothermic Reduction of Silica." 1994.

Woolley, D. and Szekely, J. "Hollow Graphite Cathode Direct Current Plasma Arc Furnace For Carbothermic Reduction of Silica: A Literature Review and Preliminary Thermodynamic Study," 1993.