UNCLASSIFIED

AD NUMBER

AD803914

NEW LIMITATION CHANGE

TO

Approved for public release, distribution unlimited

FROM

Distribution authorized to U.S. Gov't. agencies and their contractors; Administrative/Operational Use; OCT 1966. Other requests shall be referred to Air Force Materials Lab, Research and Technology Div., Wright-Patterson AFB, OH 45433.

AUTHORITY

AFSC/IST ltr, 21 Mar 1991

THIS PAGE IS UNCLASSIFIED

803914

1

AFML-TR-65-2
Part IV, Volume III

TERNARY PHASE EQUILIBRIA IN TRANSITION METAL-BORON-CARBON-SILICON SYSTEMS

Part IV. Thermochemical Calculations

Volume III. Computational Approach to the Calculation of Ternary Phase Diagrams

Y. A. Chang
Aerojet-General Corporation

TECHNICAL REPORT NO. AFML-TR-65-2, PART IV, VOLUME III.
October 1966

Air Force Materials Laboratory
Research and Technology Division
Air Force Systems Command
Wright-Patterson Air Force Base, Ohio

TERNARY PHASE EQUILIBRIA IN TRANSITION METAL-BORON-CARBON-SILICON SYSTEMS

Part IV. Thermochemical Calculations

Volume III. Computational Approach to the Calculation of Ternary Phase Diagrams

Y. A. Chang

NOTICES

When Government drawings, specifications, or other data are used for any purpose other than in connection with a definitely related Government procurement operation, the United States Government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the Government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data, is not to be regarded by implication or otherwise as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use or sell any patented invention that may in any way be related thereto.

Copies of this report should not be returned to the Research and Technology Division unless return is required by security considerations, contractual obligations, or notice on a specific document.

FOREWORD

The research described in this report was carried out at the Materials Research Laboratory, Aerojet-General Corporation, Sacramento, California, under USAF Contract No. AF 33(615)-1249. The contract was initiated under Project No. 7350, Task No. 735001, and was administered under the direction of the Air Force Materials Laboratory, Research and Technology Division, with Captain R. A. Peterson and Lt. P.J. Marchiando acting as Project Engineers, and Dr. E. Rudy, Aerojet-General Corporation, as Principal Investigator. Professor Dr. Hans Nowotny, University of Vienna, served as consultant to the program.

The project, which includes the experimental and theoretical investigation of related binary and ternary systems in the system classes Me_1-Me_2-C, Me-B-C, Me_1-Me_2-B, Me-Si-B, and Me-Si-C, was initiated on 1 January 1964. The work on related binary systems Me-C and Me-B was initiated November 1964 as a subtask to the investigation of the ternaries.

The author wishes to thank Dr. E. Rudy for his interest, advice and encouragement during the course of this work. H also wishes to thank J. Hwang for aspects of the mathematical analysis of the problem. The computer programs were prepared by Ray Marler, Len Nole, Jerry Howard, and Bill Reuss of the Computing Science Division.

The manuscript of this report was released by the author February 1966 for publication as an RTD Technical Report.

Other reports issued under USAF-Contract AF 33(615)-1249 have included:

Part I. Related Binaries

Volume I.	Mo-C System
Volume II.	Ti-C and Zr-C Systems
Volume III.	Systems Mo-B and W-B
Volume IV.	Hf-C System
Volume V.	Ta-C System. Partial Investigations in the Systems V-C and Nb-C
Volume VI.	W-C System. Supplemental Information on the Mo-C System
Volume VII.	Ti-B System
Volume VIII.	Zr-B System
Volume IX.	Hf-B System
Volume X.	V-B, Nb-B, and Ta-B Systems

Part II. Ternary Systems

Volume I.	Ta-Hf-C System
Volume II.	Ti-Ta-C System
Volume II .	Zr-Ta-C System

FOREWORD (Cont'd)

Volume IV.	Ti-Zr-C, Ti-Hf-C, and Zr-Hf-C Systems
Volume V.	Ti-Hf-B System
Volume VI.	Zr-Hf-B System
Volume VII.	Ti-Si-C, Nb-Si-C and W-Si-C Systems
Volume VIII.	Ta-W-C System
Volume IX.	Zr-W-B System, Pseudobinary System TaB_2-HfB_2
Volume X.	Zr-Si-C, Hf-Si-C, Zr-Si-B, and Hf-Si-B Systems
Volume XI.	Hf-Mo-B and Hf-W-B Systems

Part III. Special Experimental Techniques

Volume I.	High Temperature Differential Thermal Analysis

Part IV. Thermochemical Calculations

Volume I.	Thermodynamic Properties of Group IV, V, and VI Transition Metal Carbides
Volume II.	Thermodynamic Interpretation of Ternary Phase Diagrams.

This technical report has been reviewed and is approved.

W.G. Ramke

W.G. RAMKE
Chief, Ceramics and Graphite Branch
Metals and Ceramics Division
Air Force Materials Laboratory

ABSTRACT

The general conditional equations which govern the phase equilibria in three-component systems are presented. Using the general conditional equations, a general method has been developed to precalculate the phase equilibria in three-component systems from first principle using computer technique. The method developed has been applied to several model examples and the system Ta-Hf-C. The phase equilibria in three-component systems calculated using the simplified method as originally developed by Rudy, agree well with those calculated by the present method. The only difference is in the homogeneous range with respect to the interstitial component of solid solutions which exhibit large variation with metal exchange. This is to be expected in view of the assumptions made in the simplified method.

In connection with the phase diagram calculation and other problems of the present Air Force contract, several computer programs have been developed which are included in the appendix of this report.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Cont'd)

I. INTRODUCTION AND SUMMARY

A. INTRODUCTION

Apart from theoretical interest, the capability of predicting phase equilibria in three-component systems from binary data is of practical significance since the cost of experimentally investigating phase equilibria in ternary systems is rather high. For the transition $metal_1$, $metal_2$, carbon ternary systems (i.e. the interstitial type of solid solutions), a method has been developed by Rudy[1] to predict the phase equilibria in ternary systems assuming that the intermediate phases in the $metal_1$-carbon and $metal_2$-carbon binaries are either of line-compounds or that both phases are of equal dependence of the free energy on the concentration coordinate of the interstitial component. In cases, where the homogeneous ranges of the solid solutions with respect to the interstitial component changes drastically with metal exchange, the simplified method does not predict, as to be expected from the assumptions made, the exact phase boundaries. The purpose of the present work is to develop a general method using computer technique to predict the phase equilibria in three-component systems with no assumptions made as in the simplified method. This general method will predict not only the phase equilibria but also the correct homogeneous ranges of the single phases.

B. SUMMARY

The conditional equations which govern the phase equilibria in three-component systems are presented. Using the conditional equations, a method has been developed using computer technique to precalculate the phase equilibria in three-component systems from first principle. The method developed has been applied to model examples and the system Ta-Hf-C. The phase equilibria in three-component systems calculated using the simplified method originally developed by Rudy, agree rather well with those calculated by the present method. The only difference is in the homogeneous ranges of solid solutions which exhibit large variation with metal exchange. This is to be expected in view of the assumptions made in the simplified method.

Several computer programs for the phase equilibrium calculations as well as for other problems of the present Air Force contract have been developed. The fortran statements of all the computer programs are included in the appendix at the end of the report.

II. THERMODYNAMIC DESCRIPTION OF PHASE EQUILIBRIA IN THREE COMPONENT SYSTEMS

According to the phase rule, at constant pressure the maximum number of coexisting phases in a one-component system is two, two-component system three, and three-component system, four. In addition, if one fixes temperature, the maximum number of coexisting phases in a ternary system reduces to three. Consequently, an isothermal section of a ternary phase diagram is built up of one-phase, two-phase, and three-phase equilibria. On the one hand, concentrations of the coexisting phases in a two-phase field determine the phase boundaries of single-phase fields. On the other hand, the boundaries of a three-phase equilibrium are the limiting tie-lines of the three adjacent two-phase equilibria. We shall now discuss first the thermodynamics of two-phase equilibria.

A. TWO-PHASE EQUILIBRIA IN THREE-COMPONENT SYSTEM

The Gibbs free energy of formation of a two-phase alloy, $A_xB_yC_z$ as shown in Figure 1 is a linear combination of the Gibbs free energies of formation of the two coexisting phases $A_{x'}B_{y'}C_{z'}$ and $A_{x''}B_{y''}C_{z''}$. Expressing in mathematical terms, we have

$$\Delta G = \nu_1 \Delta G_1 + \nu_2 \Delta G_2 \qquad (1)$$

where ΔG, ΔG_1, and ΔG_2 are the Gibbs free energies of formation of the two-phase alloy and of the two co-existing single phases; and ν_1 and ν_2 are the relative amounts of the two co-existing single phases. If we take one gramatom of alloy as our basis, we have the following three boundary conditions:

$$\nu_1 + \nu_2 = 1 \tag{2}$$

$$x' + y' + z' = 1 \tag{3}$$

$$x'' + y'' + z'' = 1 \tag{4}$$

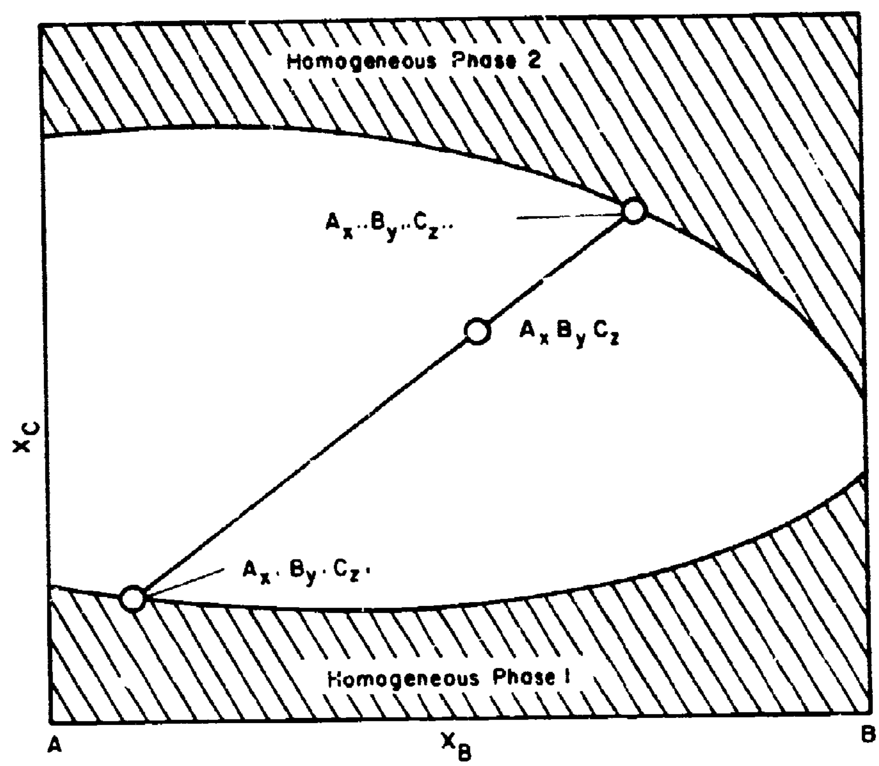

Figure 1. Two-Phase Equilibria in a Ternary System.

Two additional boundary conditions can be obtained resulting from the conservation of masses:

$$\nu_1 x' + \nu_2 x'' = x \tag{5}$$

$$\nu_1 y' + \nu_2 y'' = y \tag{6}$$

Given ΔG_1 and ΔG_2 as a function of composition at constant P and T and a set of x, y, z values, we would like to calculate the values of x', y', z' and x'', y'', and z''. Since there are eight unknowns: ν_1, ν_2, x', y', z', x'', y'', and z'' and five equations (2 to 6), we have only three independent variables. We can choose any three variables as we wish. In the present case, we shall choose x', y', and x'', as our ultimate variables. By elimination of the various variables in equations (2) through (6), we obtain the following expressions for ν_1, ν_2, z', y'' and z'' in terms of x', y' and x'':

$$\nu_1 = \frac{x - x''}{x' - x''} \tag{7}$$

$$\nu_2 = \frac{x - x'}{x'' - x'} \tag{8}$$

$$z' = 1 - x' - y' \tag{3a}$$

$$y'' = \frac{x' - x''}{x' - x} y + \frac{x - x''}{x - x'} y' \tag{9}$$

and

$$z'' = 1 - x'' - y'' \tag{4a}$$

In order to solve for x', y' and x'', we need three additional equations which must be derived from the condition that at equilibrium ΔG according to equation (1) is a minimum. We can minimize ΔG, with the five constraints expressed by equations (2) through (6), after the method of Lagrange as was originally done by Rudy[1]. We obtain,

$$\Delta G_1 - \alpha_1 \qquad\qquad -\alpha_4 x' \quad -\alpha_5 y' = 0 \tag{10}$$

$$\Delta G_2 - \alpha_1 \qquad\qquad -\alpha_4 x'' \quad -\alpha_5 y'' = 0 \tag{11}$$

$$\nu_1 \frac{\partial \Delta G_1}{\partial x'} \quad -\alpha_2 \qquad -\alpha_4 \nu_1 \qquad = 0 \tag{12}$$

$$\nu_1 \frac{\partial \Delta G_1}{\partial y'} \quad -\alpha_2 \qquad\qquad -\alpha_5 \nu_1 = 0 \tag{13}$$

$$\nu_1 \frac{\partial \Delta G_1}{\partial z'} \quad -\alpha_2 \qquad = 0 \tag{14}$$

$$\nu_2 \frac{\partial \Delta G_2}{\partial x''} \quad -\alpha_3 \quad -\alpha_4 \nu_2 \qquad = 0 \tag{15}$$

$$\nu_2 \frac{\partial \Delta G_2}{\partial y''} \quad -\alpha_3 \quad -\alpha_5 \nu_2 = 0 \tag{16}$$

$$\nu_2 \frac{\partial \Delta G_2}{\partial z''} \quad -\alpha_3 \qquad = 0 \tag{17}$$

From equations (12), (13), (15), and (16); (12), (14), (15), and (17); and (12), (14), (15), and (17), we obtain the following additional equations:

$$\left[\frac{\partial \Delta G_1}{\partial x'} - \frac{\partial \Delta G_1}{\partial y'}\right]_{T,p} = \left[\frac{\partial \Delta G_2}{\partial x''} - \frac{\partial \Delta G_2}{\partial y''}\right]_{T,p} \tag{18}$$

$$\left[\frac{\partial \Delta G_1}{\partial z'} - \frac{\partial \Delta G_1}{\partial x'}\right]_{T,p} = \left[\frac{\partial \Delta G_2}{\partial z''} - \frac{\partial \Delta G_2}{\partial x''}\right]_{T,p} \tag{19}$$

and

$$\Delta G_1 - \Delta G_2 = (x' - x'') \left[\frac{\partial \Delta G_1}{\partial x'} - \frac{\partial \Delta G_1}{\partial z'}\right]_{T,p} + (y' - y'') \left[\frac{\partial \Delta G_1}{\partial y'} - \frac{\partial \Delta G_1}{\partial z'}\right]_{T,p} \tag{20}$$

From equations (18), (19), and (20), we can in principle solve for x'', x', and y'. However, since the equations for ΔG_1 and ΔG_2 are non-linear with respect to the three independent variables x'', x', and y', the solving of the three simultaneous equations is not very simple. Instead, it is more convenient to numerically find the minimum value of ΔG according to equation (1) by varying the values of x'', x', and y' from 0.0 to 1.0. We can take any small incremental value as we wish. In general, for phase diagram calculations, we have found a value of 0.01 is sufficient.

Since $x' + y' \leq 1$, we need only to consider the values of x', y', and x'' in the prism as shown in Figure 2 for the computation of ΔG. Moreover

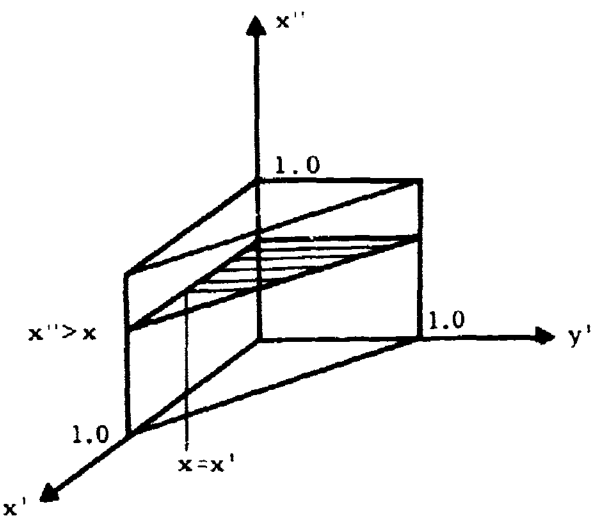

Figure 2. Schematic Representation of the Three Independent Variables x', y' and x'' for Two-Phase Equilibria in a Ternary System when x'' > x.

the values of x always lie between those of x' and x''. For each value of x'' greater than x, we compute ΔG for all values of x' smaller than x, i.e. the shaded trapezoidal region (in Figure 2) and for each value of x' smaller than x, we do the similar computations for all values of x' greater than x, i.e. the shaded trapezoidal region in Figure 3.

Since ν_1, ν_2 and y'' are all positive, rearranging equation (6), we have

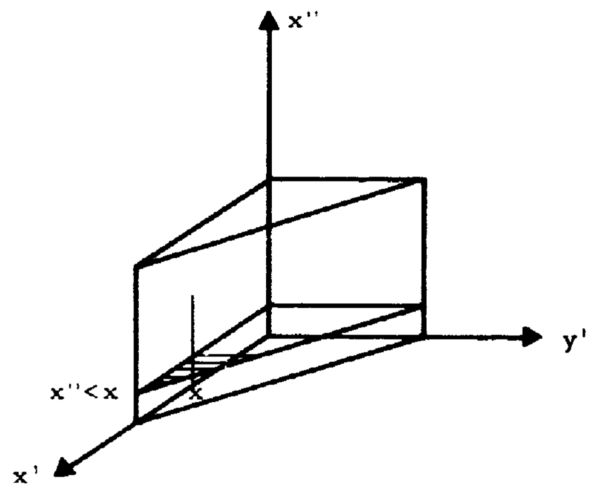

Figure 3. Schematic Representation of the Three Independent Variables x', y', and x'' for Two-Phase Equilibria in a Ternary System when x'' < x

$$y' = \frac{y}{\nu_1} - \frac{\nu_2}{\nu_1} y'' \le \frac{y}{\nu_1} = \frac{x' - x''}{x - x''} y \tag{21}$$

Equation (21) gives an upper limit of y'.

The number of computations for searching the minimum of ΔG according to equation (1) can be further reduced when we use the additional information provided by the binary phase boundaries.

Before proceeding further, we must have some mathematical expressions for ΔG_1 and ΔG_2. As shown previously,[2,3] the free energy of a

single phase of the interstitial type of solid solutions such as the ternary
carbides may be adequately represented by the following expression,
case of the interstitial type of solid solutions such as the ternary
may be adequately represented by the following expression,

$$\Delta G_1 \text{ (in cal/gatom alloy)} = x'\Delta G_{AC_u} + y'\Delta G_{BC_u} + \frac{x'y'\epsilon'}{(1-z')} + RT\left[x'\ln\frac{x'}{1-z'} + y'\ln\frac{y'}{1-z'}\right] \quad (22)$$

where x', y' and z' are atom fractions of A, B, and C in phase 1, ϵ' is the interaction parameter for the solid solution $(A,B)C_u$, R is the universal gas constant, T is the absolute temperature, and ΔG_{AC_u} and ΔG_{BC_u} are the Gibbs free energies of formation of the two respective binary phases in cal/gatom A or B. We have a similar expression for ΔG_2 which is a function of x'', y'' and z''.

Often we have the cases where the solubility of the third component, let's say C, in the solution $(A,B)C_u$ is so small that for practical purposes, we can take u as zero. For such cases, $z' = 0$, we reduce the three independent variables x', y' and x'' to only x' and x''. Therefore, the search for a minimum for ΔG according to equation (1) reduces from a three-dimensional to a two-dimensional problem.

Model Example 1:

Let us now take a hypothetical case where A and B form a series of continuous solid solutions at high temperature; A and C and B and C form two intermediate phases AC_v, AC_w and BC_v, BC_w, respectively as shown in Figures 4 and 5. Moreover, the solubility of C in the solid solution (A,B) is assumed to be small; and $(A,B)C_v$ and $(A,B)C_w$ form two series of continuous solid solutions. The homogeneous range with respect to component C is narrow in the solid solution $(A,B)C_w$ solid solution. We would like now to compare the tie-line distributions in the two-phase regions (A,B)-$(A,B)C_v$ and (A,B)-$(A,B)C_w$ calculated at 2000°K using the method described here with those calculated by the simplified method[1].

The Gibbs free energies of the four binary intermediate phases: AC_v, BC_v, AC_w, and BC_w may be represented by the following equations (see Figures 4 and 5).

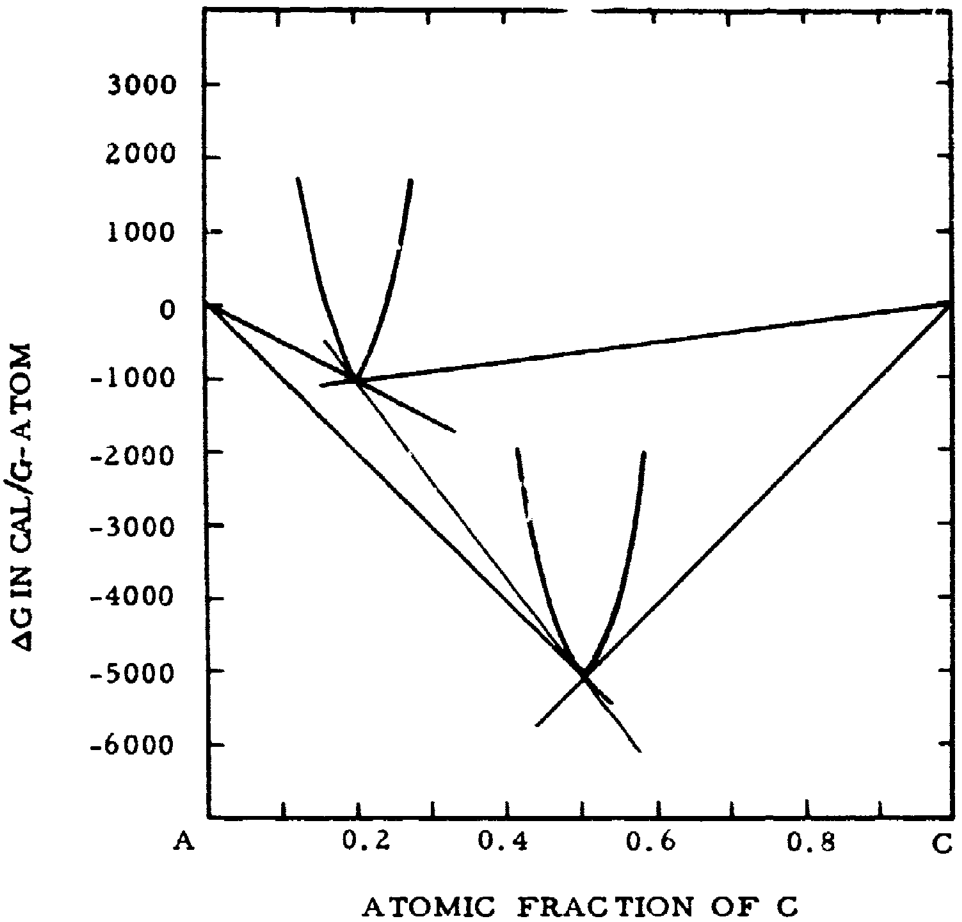

Figure 4. Gibbs Free Energies of the Two Intermediate Phases AC_v and AC_w in the binary A-C.

$$\Delta G_{AC_v} = \frac{1}{1-z}\{23,000-240,000\,z + 600,000\,z^2\} \tag{23}$$

$$\Delta G_{BC_v} = \frac{1}{1-z}\{35,710-418,882\,z + 1,102,162\,z^2\} \tag{24}$$

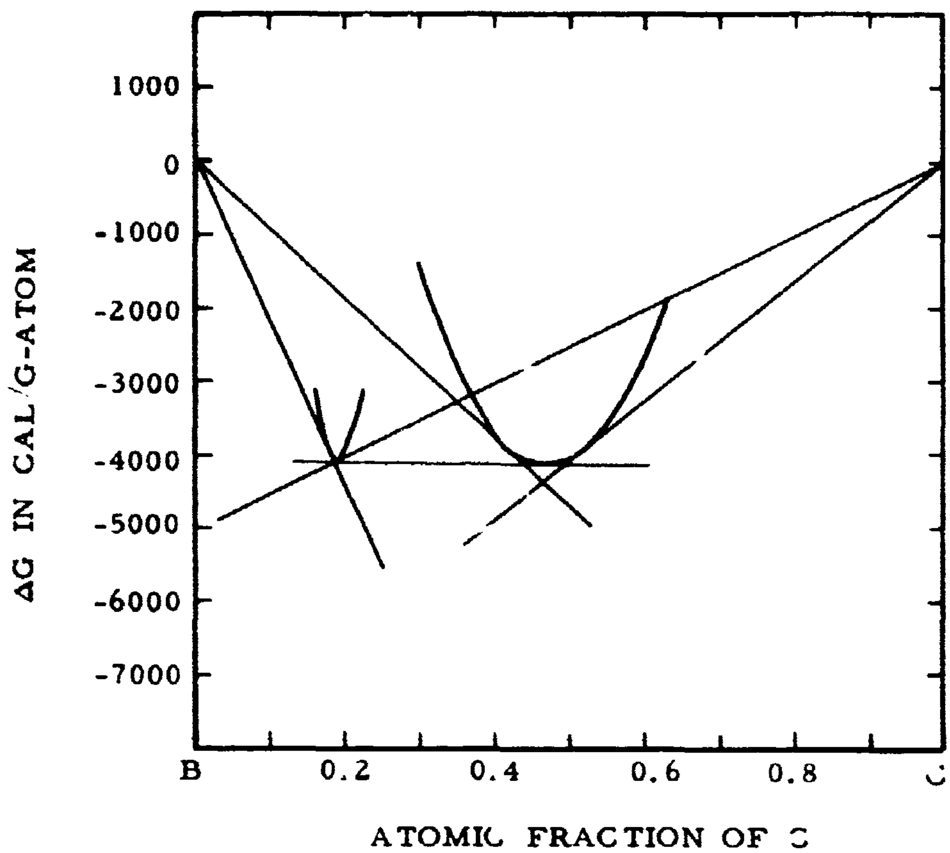

Figure 5. Gibbs Free Energies of the Two Intermediate Phases BC_v and BC_w in the Binary A-C.

$$\Delta G_{AC_w} = \frac{1}{1-z} \{121,065-504,220\,z + 504,220\,z^2\} \qquad (25)$$

$$\Delta G_{BC_w} = \frac{1}{1-z} \{15,335-83,555\,z + 89,780\,z^2\} \qquad (26)$$

The Gibbs free energies of the ternary solutions $(A,B)C_v$ and $(A,B)C_w$ may take the same form as equation (22). The interaction parameters for the three solid solutions are taken to be,

$$\epsilon_{A,B} = 5970$$

$$\epsilon_{(A,B)C_v} = 3980$$

$$\epsilon_{(A,B)C_w} = 3980$$

We shall first consider the two-phase equilibria between the solid solutions (A, B) and $(A,B)C_v$. Since we assume that the intermediate phases AC_w and BC_w are not stable in the binary systems, the phase AC_v then becomes stable. Accordingly, we have

$$\Delta G_1 = 5970\, x'y' + RT\,(x' \ln x' + y' \ln y') \tag{27}$$

$$\Delta G_2 = x''\Delta G_{AC_v} + y''\Delta G_{BC_v} + \frac{3980\, x''y''}{1 - z''} + $$

$$+ RT\left[x''\ln \frac{x''}{1-z''} + y''\ln \frac{y''}{1-z''}\right] \tag{28}$$

One can see from Figures 6 and 7 that the tie-line distribution between the solid solutions (A, B) and $(A\ B)C_v$ calculated using the method described here agrees well with that calculated using the simplified method. This is to be expected since the range of homogeneity with respect to component C in $(A,B)C_v$ is relatively narrow. The simplified method for calculating the two-phase equilibria in ternary systems has been discussed extensively by Rudy[1]. It suffices to say that the compositions of the two co-existing phases are where the concentration free energy gradients are the same, i.e.

$$\frac{d\Delta G_1}{dx'} = \frac{d\Delta G_2}{dx''} \tag{29}$$

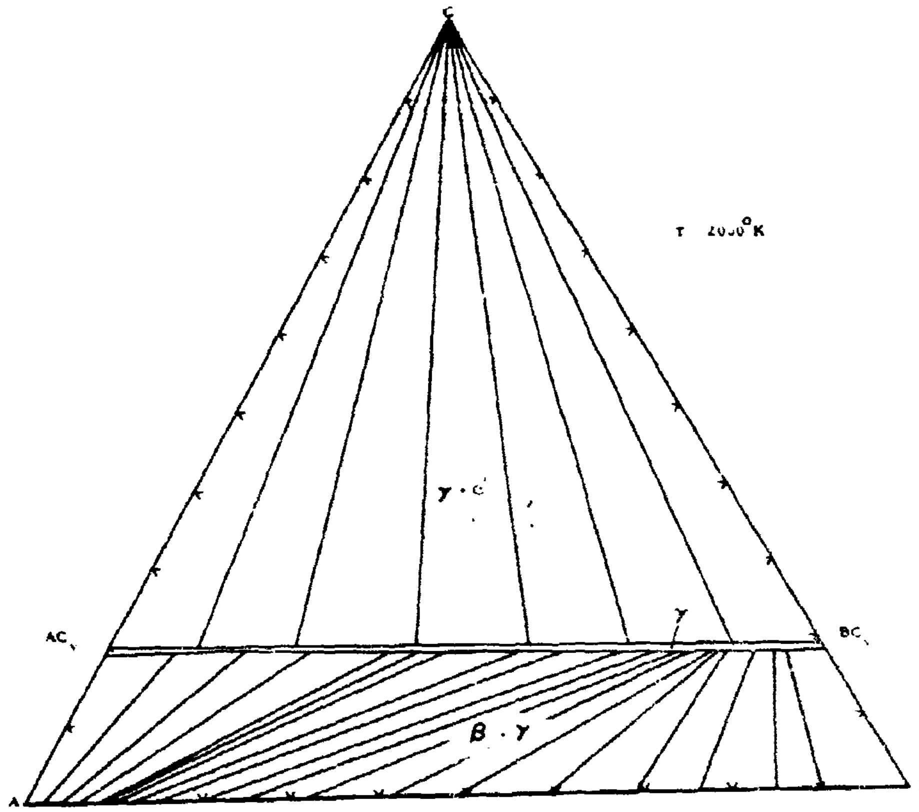

Figure 6. Phase Equilibria Between Phases β and γ and γ and C in the System A-B-C at 2000°K Calculated Using the General Method.

The concentration free energy gradient curves for the solid solutions (A, B), (A, B)C_v and (A, B)C_w are shown in Figure 8.

Next, we consider the two-phase equilibria between the solid solutions (A, B) and (A, B)C_w. In this case, we assume that the phases AC_v and BC_v do not appear in the two respective binaries. Since the homogeneous range for the phase BC_w is relatively large and the free energies of AC_w phase increase rapidly at concentrations of C-component higher and lower

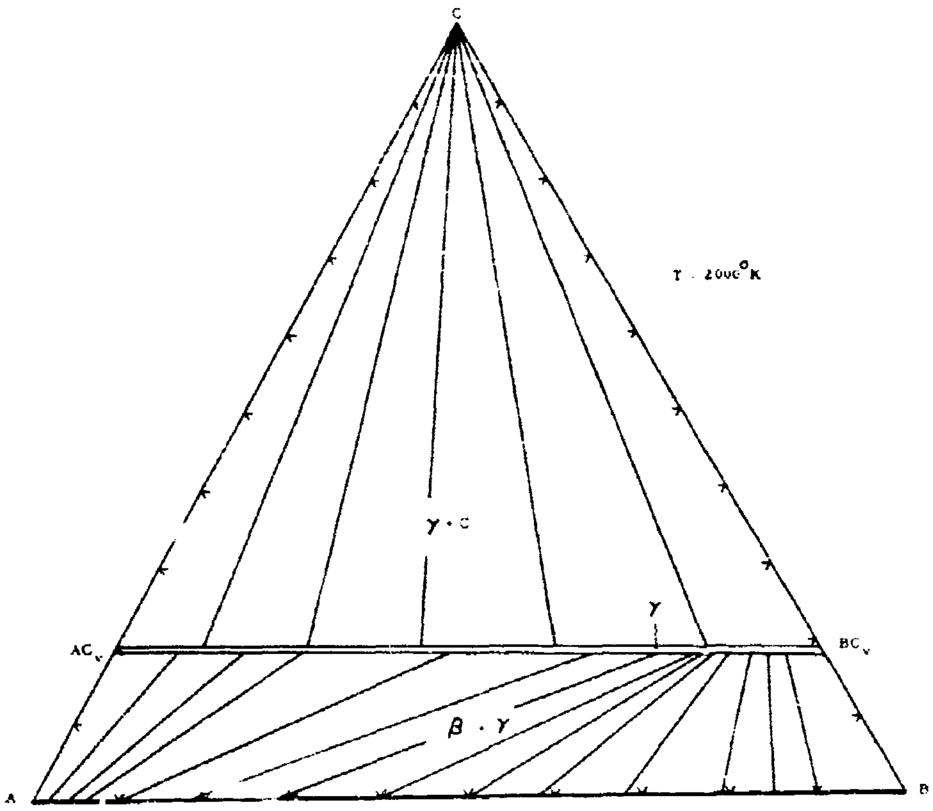

Figure 7. Phase Equilibria Between Phases β and γ and γ and C in the System A-B-C at 2000°K Calculated Using the Simplified Method.

0.5, the tie-line distribution calculated (Figure 9) used this simplified method does not agree well with that calculated (Figure 10) using the general method described here again as expected. Moreover, the A-B-rich phase boundary of the solid solution $(A,B)C_w$ calculated using the method described in this paper shows a curvature as expected while that calculated using the simplified method does not.

It should be pointed out here since the solubilities of A and B in C are assumed to be negligible, the tie lines in the two-phase regions $(A,B)C_v$ and C and $(A,B)C_w$ and C (Figures 6, 7, 9, and 10) were not calculated and must be pointed toward the pure component.

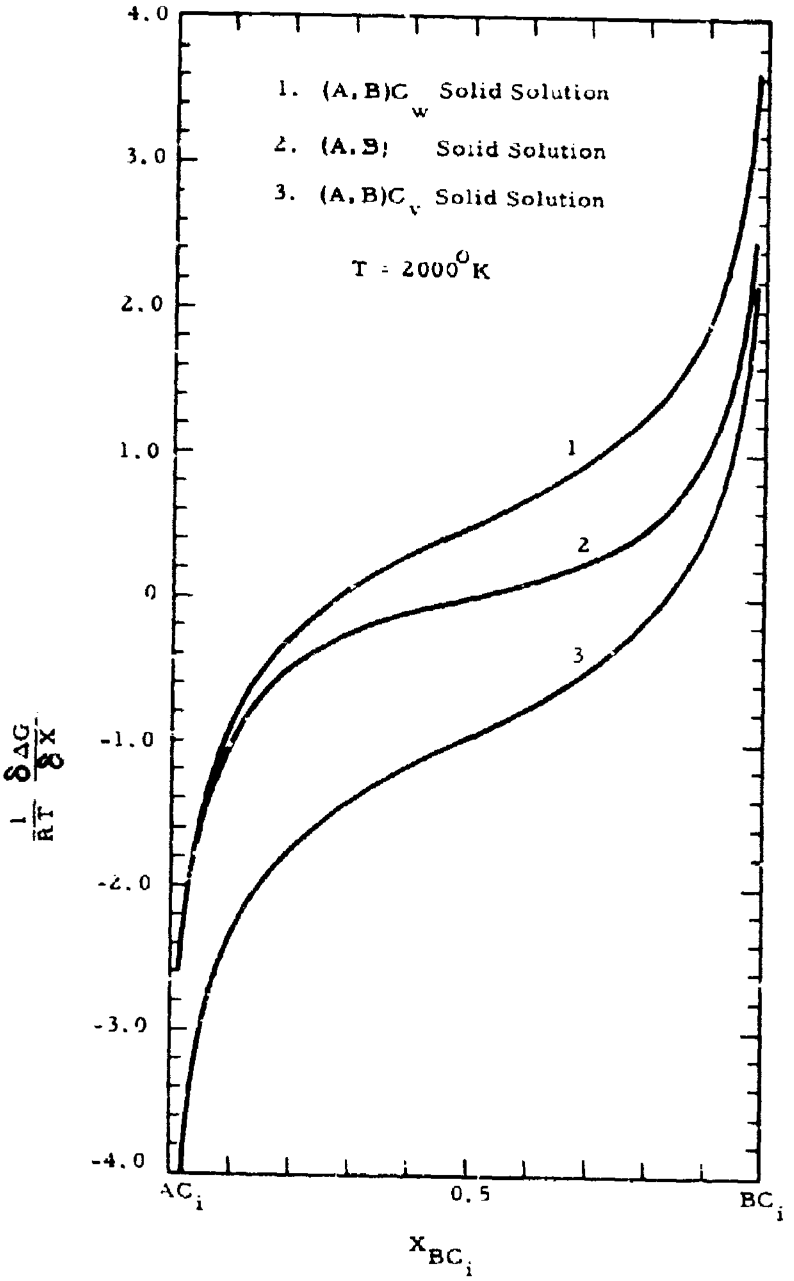

Figure 8. Concentration Free Energy Gradient Curves for Solutions (A, B), (A, B)C_v and (A, B)C_w at 2000°K.

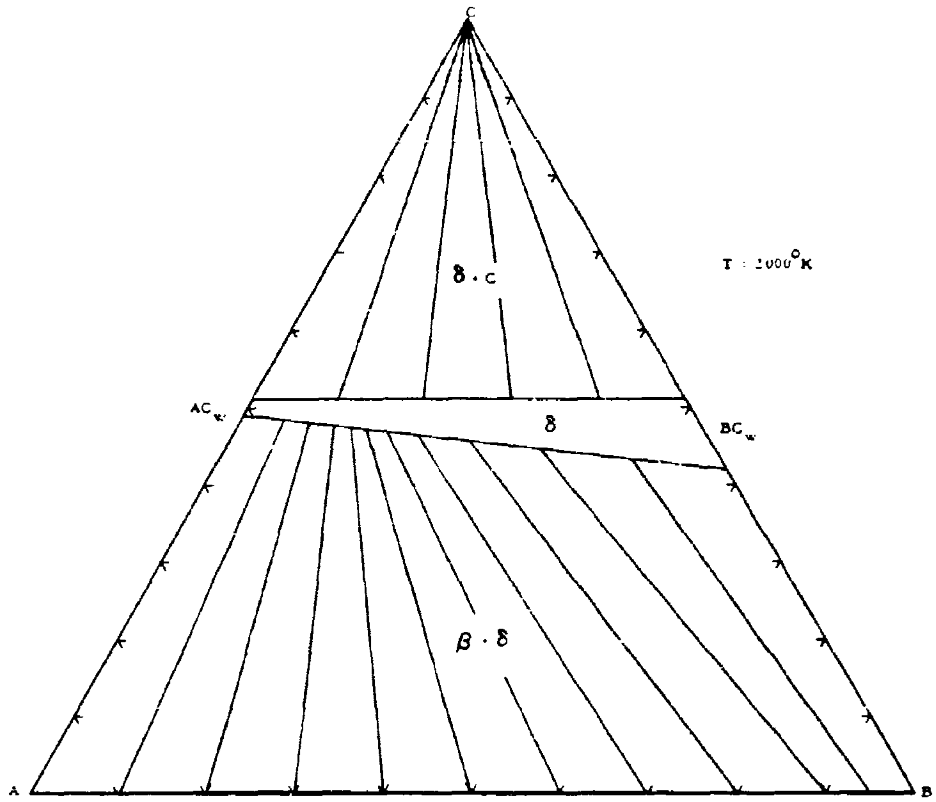

Figure 9. Phase Equilibria Between β and δ and δ and C in the System A-B-C at 2000°K Calculated Using the Simplified Method.

B. THREE-PHASE EQUILIBRIA IN THREE-COMPONENT SYSTEM

In a similar manner as for the two-phase equilibria, the Gibbs free energy of a three-phase alloy $A_xB_yC_z$ as shown in Figure 11 is a linear combination of the Gibbs free energies of formation of the three co-existing phases $A_{x'}B_{y'}C_{z'}$, $A_{x''}B_{y''}C_{z''}$ and $A_{x'''}B_{y'''}C_{z'''}$:

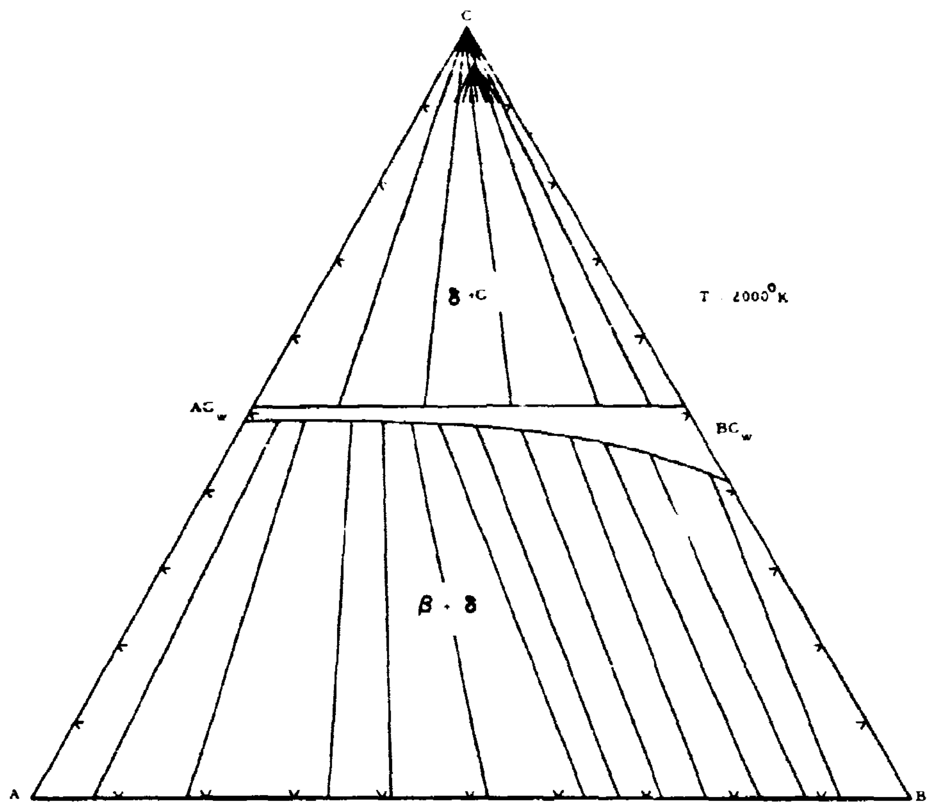

Figure 10. Phase Equilibria Between β and δ and δ and C in the System A-B-C at 2000°K Calculated Using the General Method.

$$\Delta G = \nu_1 \Delta G_1 + \nu_2 \Delta G_2 + \nu_3 \Delta G_3 \tag{30}$$

with the following constraints:

$$\nu_1 + \nu_2 + \nu_3 = 1 \tag{31}$$

$$x' + y' + z' = 1 \tag{32}$$

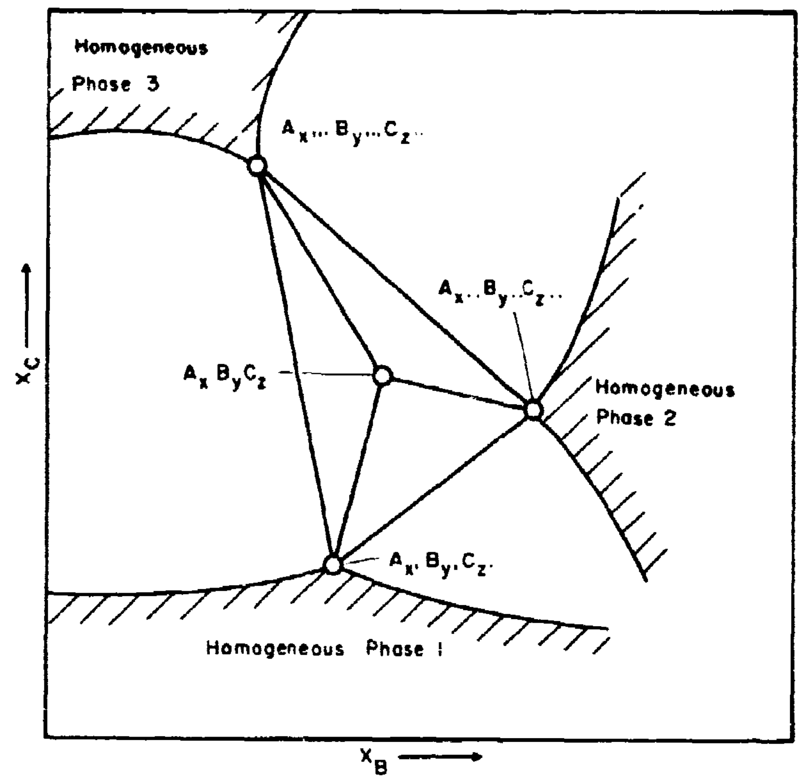

Figure 11. Three-Phase Equilibrium in a Ternary System

$$x'' + y'' + z'' = 1 \tag{33}$$

$$x''' + y''' + z''' = 1 \tag{34}$$

$$\nu_1 x' + \nu_2 x'' + \nu_3 x''' = x \tag{35}$$

$$\nu_1 y' + \nu_2 y'' + \nu_3 y''' = y \tag{36}$$

We have twelve unknowns and six equations (31 through 36). Therefore, we have six independent variables. It is no longer practical for us to use the similar method as for the two-phase equilibria to find the combination of the compositions of the three co-existing phases where ΔG according to equation (30) is a minimum.

Fortunately, we can reduce the six independent variables to two by taking advantage of the fact that the boundaries of a three-phase equilibrium are nothing but the limiting tie-lines of the three adjacent two-phase equilibria. We shall first discuss the method of seeking the compositions (x''', y''', z''') of the third co-existing phase when the compositions (x', y', z' and x'', y'', z'') of the other two co-existing phases are known. We will then discuss how to find the compositions x', y', z' and x'', y'', z''. Under such conditions, instead of having the six constraints as expressed by equations (31) through (36), we have only four boundary conditions: Equations (31), (34), (35), and (36) with six unknowns, i.e. ν_1, ν_2, ν_3, x''', y''' and z'''. Therefore, we have ultimately two independent variables. We can choose any two variables as we wish, but from a practical point of view, it is convenient to choose the two variables z''' and y'''. Expressing ν_1, ν_2, ν_3 and x''' in terms of y''' and z'''; we have

$$\nu_1 = \frac{(y'-y)(z'''-z) - (z''-z)(y'''-y)}{D} \tag{37}$$

$$\nu_2 = \frac{(y-y')(z'''-z') - (z-z')(y'''-y')}{D} \tag{38}$$

$$\nu_3 = 1 - \nu_1 - \nu_2 \tag{31a}$$

$$x''' = 1 - y''' - z''' \tag{34a}$$

where

$$D = (y''-y')(z'''-z') - (z'-z')(y'''-y') \tag{39}$$

To find the values of y''' and z''' we again have to use the condition that at equilibrium ΔG according to equation (30) is a minimum. The method for seeking the values of y''' and z''' where ΔG is a minimum is similar to the method used for solving the two-phase equilibria discussed earlier.

For the three-phase equilibria resulting from a miscibility gap in one of the solid solutions, the compositions of the miscibility gap of this solid solution are well-defined. In this case, we can solve for the composition of third co-existing phase using the method described here.

Model Example 2.

We shall use the same data as in Model Example 1 to illustrate the application of the method described here. When the temperature is lower from 2000°K to 1250°K, a miscibility gap results from the solid solution (A, B). The compositions of the miscibility gap at 1250°K are $x' = 0.83$, $y' = 0.17$ and $x'' = 0.17$ and $y'' = 0.83$. Again, as shown in Figures 12 and 13, the phase equilibria calculated using the simplified method agree well with those calculated using the general method with the exception that homogeneous range of the solid solution $(A,B)C_v$ with respect to component C is narrower. The free energy-concentration-gradient curves used to determine the phase equilibria at 1250°K as shown in Figure 12, are displaced in Figure 14.

Referring to Figure 15, the method for seeking the composition of the three co-existing phases β-γ-δ will be discussed using the following conditions:

at x_1, y_1, z_1:	ΔG (2φ between β-δ)	<	ΔG(2φ between β-γ)
		<	ΔG(3φ between β-γ-δ)
at x_2, y_2, z_2:	ΔG (3φ between β-δ-γ)	<	ΔG(2φ between β-δ)
		<	ΔG(2φ between β-γ)
at x_3, y_3, z_3	ΔG (2φ between β-γ)	<	ΔG(2φ between β-δ)
		<	ΔG(3φ between β-γ-δ)

In reality, we don't have any idea about the compositions of the three co-existing phases except we know there is a three-phase equilibrium β-γ-δ since the corresponding phase in the A-C binary is not stable. Thus, we can start, let's say, with a composition x_1, y_1, z_1 and ask ourselves what possible three-phase field it is in. We know one of the phase boundaries of this three-phase equilibrium β-γ-δ is the tie-line between β-δ and furthermore, this tie-line must lie at the left of the composition x_1, y_1, z_1. Thus, we can start with various compositions at the left of the composition x_1, y_1, z_1 and first find the corresponding compositions of the two co-existing phases β and δ. Knowing the various values of x', y', z' (β-phase) and x'', y'', z'' (δ-phase), one can calculate the corresponding values of x''', y''', z''' (γ-phase) and the corresponding values of ΔG (3φ between β-δ-γ). One then compares the lowest value among all the ΔG values calculated with those of ΔG (2φ between β-δ) and ΔG(2φ between β-γ)

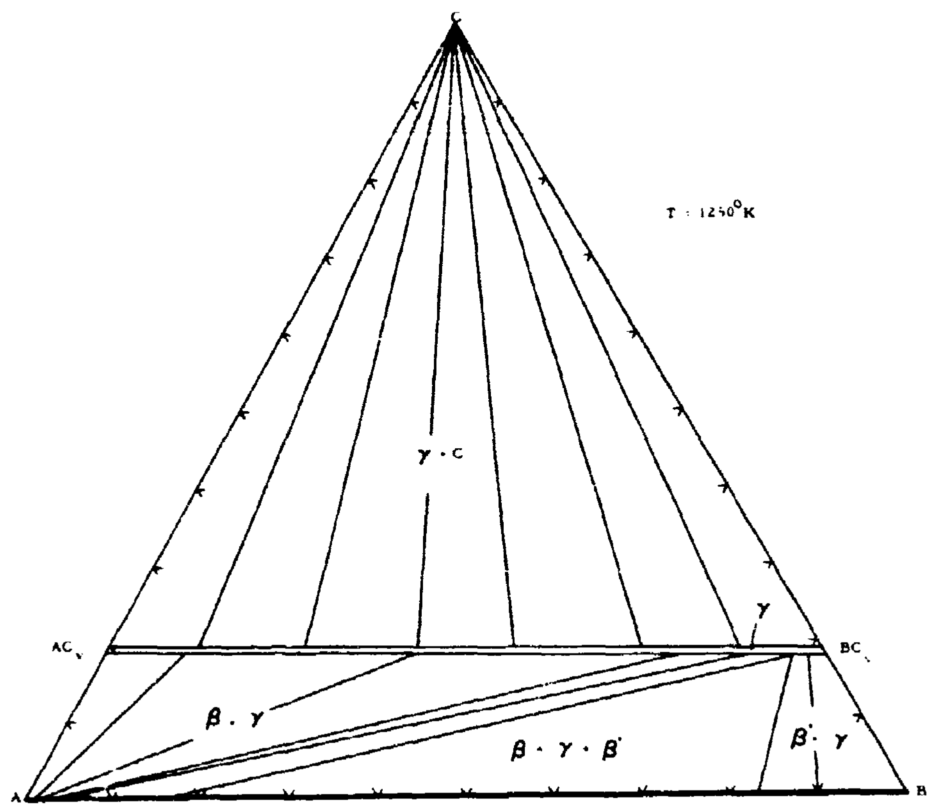

Figure 12. The Phase Equilibria Calculated Using the Simplified Method in the System A-B-C at 1251°K When the Two Binary Phases AC_w and BC_w are Unstable.

using the method described earlier. The stable equilibrium is defined by the lowest value of the Gibbs free energy. If ΔG (2ϕ between β-δ) is the lowest, the three-phase equilibrium must lie at the right of the chosen values of x_1, y_1, z_1. If on the other hand, the value of ΔG (2ϕ between β-γ) is the lowest, then the three-phase equilibrium must lie to the left of the chosen values of x_1, y_1, z_1. If the value of ΔG (3ϕ between β-γ-δ) is the lowest, then we have found the three-phase equilibrium. A simple and rapid iteration process can be prepared to solve the compositions of the three co-existing phases.

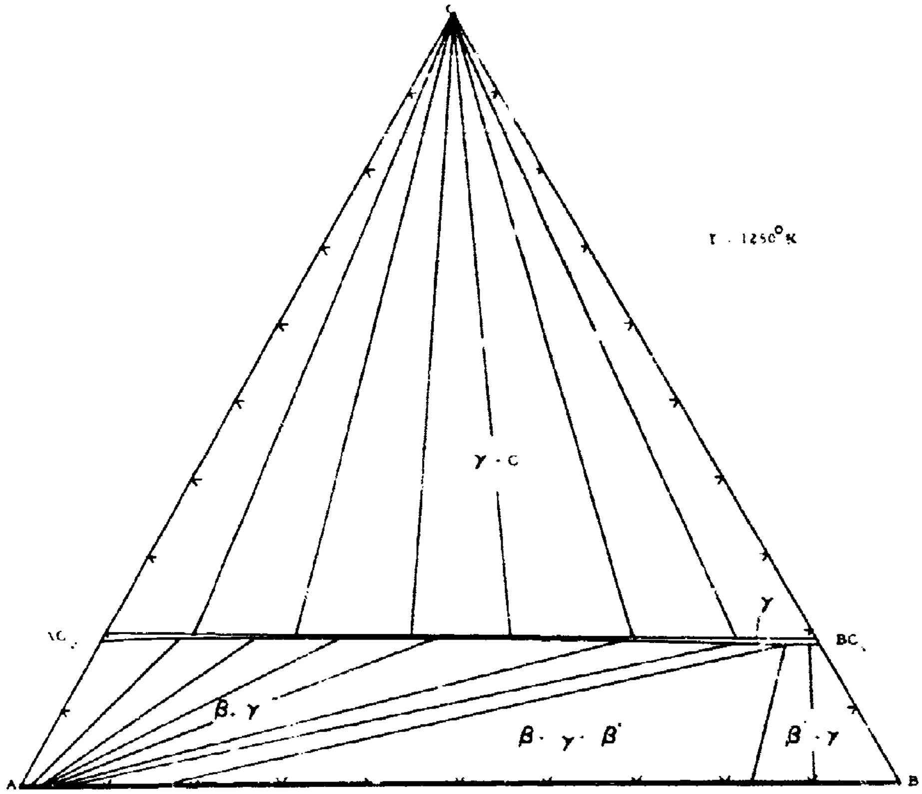

Figure 13. The Phase Equilibria Calculated Using the General Method in the System A-B-C at 1250°K When the Two Intermediate Phases AC_w and BC_w are Unstable.

Model Example 3:

Using the same data as in Model Example 1 and assuming all the three phase BC_v, AC_w and BC_w appear in the corresponding two binaries A-C and B-C, we shall use the method described here to find the three-phase field β-γ-δ and the tie-lines in the three two-phase fields: β-δ, β-γ and γ-δ. The phase diagrams calculated at 2000°K using the general method as shown in Figure 16 agrees well with that calculated using the simplified method (Figure 17). The obvicus difference between the two calculated

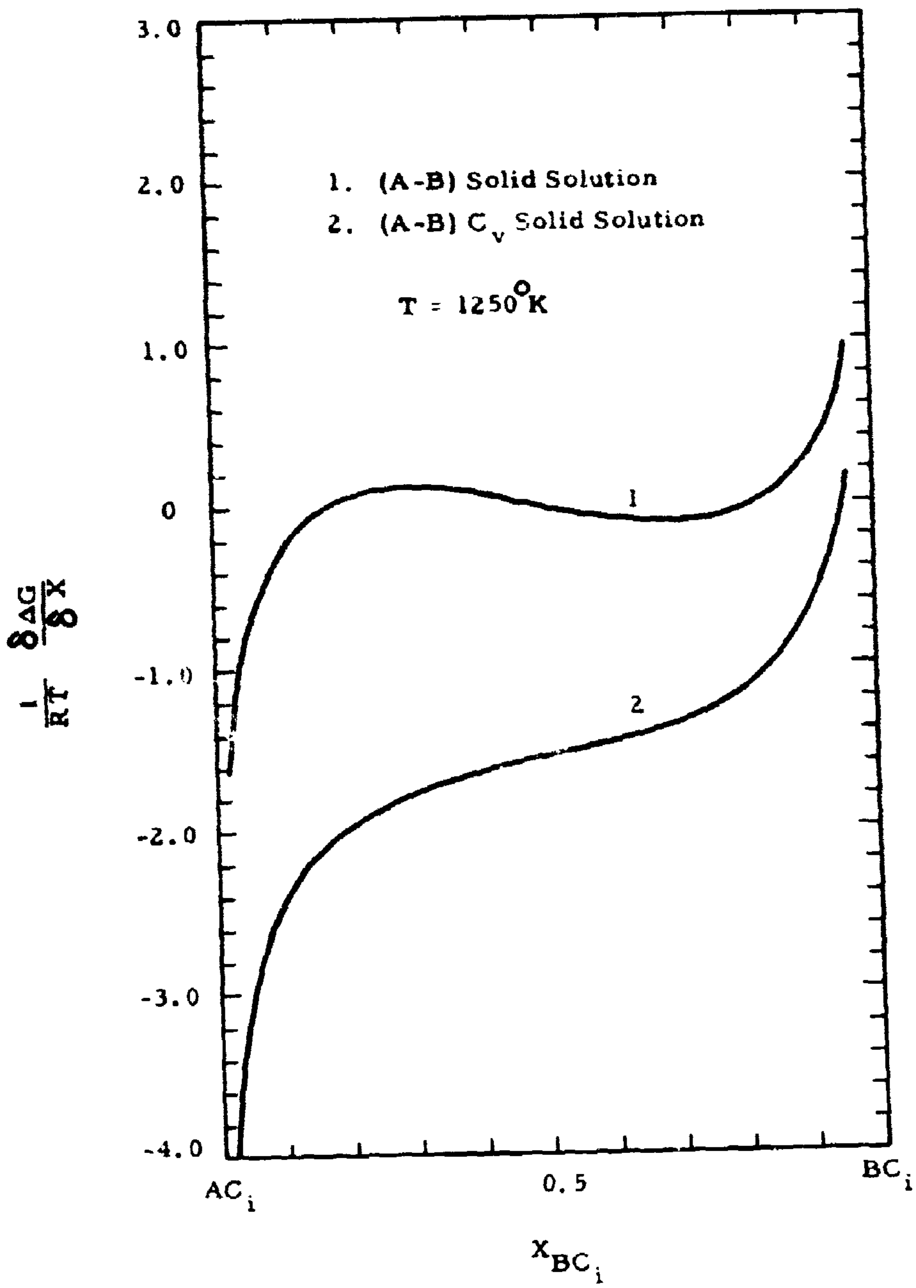

Figure 14. The Concentration-Free-Energy-Gradient Curves for Solutions (A,B) and (A,B)C_v at 1250°K.

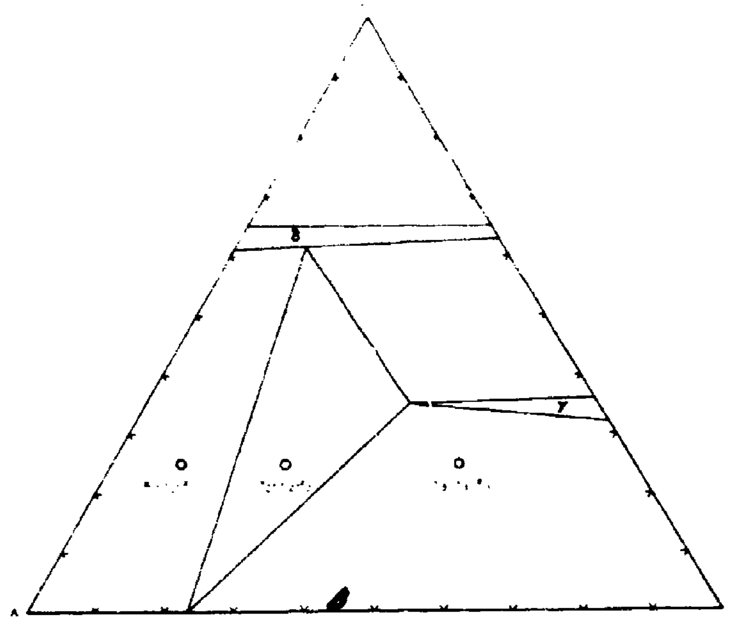

Figure 15. A Typical Ternary Phase Diagram with a Three-Phase Field Resulting from the Absence of a Corresponding Intermediate Phase in the Binary A-C.

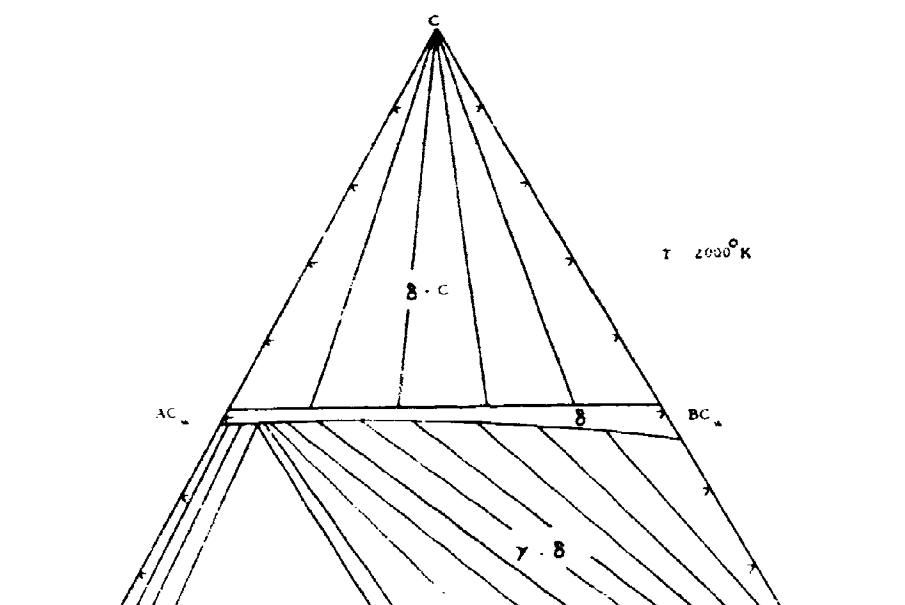

Figure 16. Phase Equilibria Calculated Using the General Method in the System A-B-C at 2000°C when the three Phases AC_w, BC_v and BC_w are Stable in the Binaries

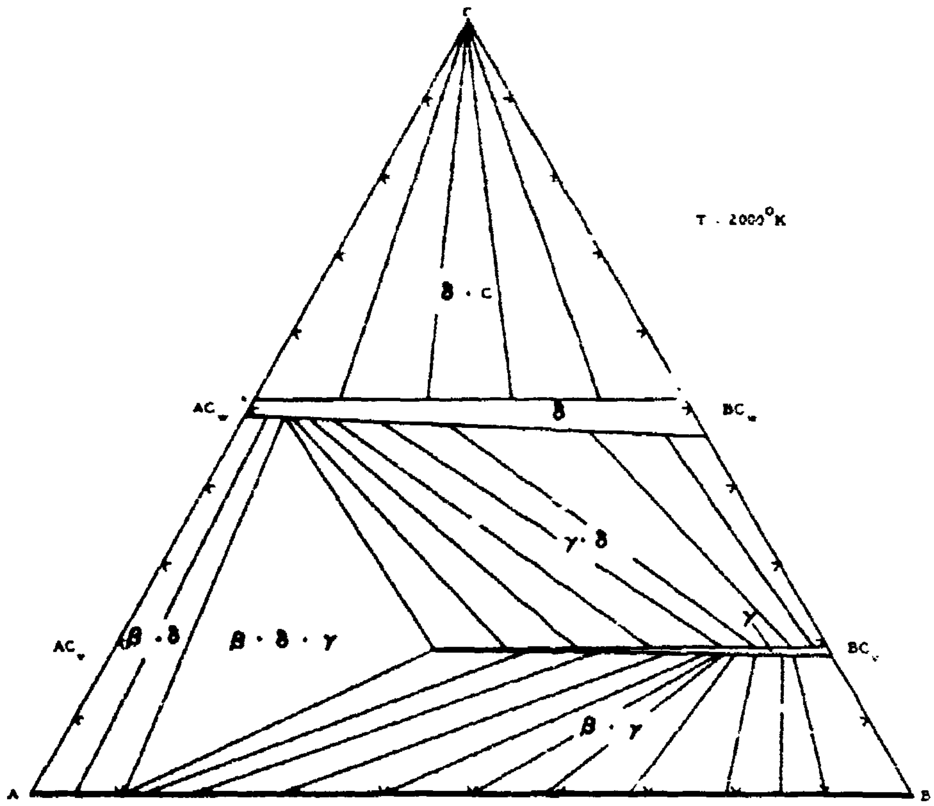

Figure 17. Phase Equilibria Calculated Using the Simplified Method in the System A-B-C at 2000°K When the Three Phases AC_w, BC_v and BC_w are Stable in the Binaries.

phase diagrams is the (A-B) rich homogeneous range of the solid solution $(A\text{-}B)C_w$ with respect to C-component. The homogeneous range calculated using the general method shows the curvature while that calculated using the simplified method does not.

Although the technique of fixing the compositions of the coexisting phases for a three-phase equilibrium using the simplified method has been discussed extensively previously, it is worthwhile to point out that in addition to the condition at equilibrium, the free-energy-concentration gradients, i.e.

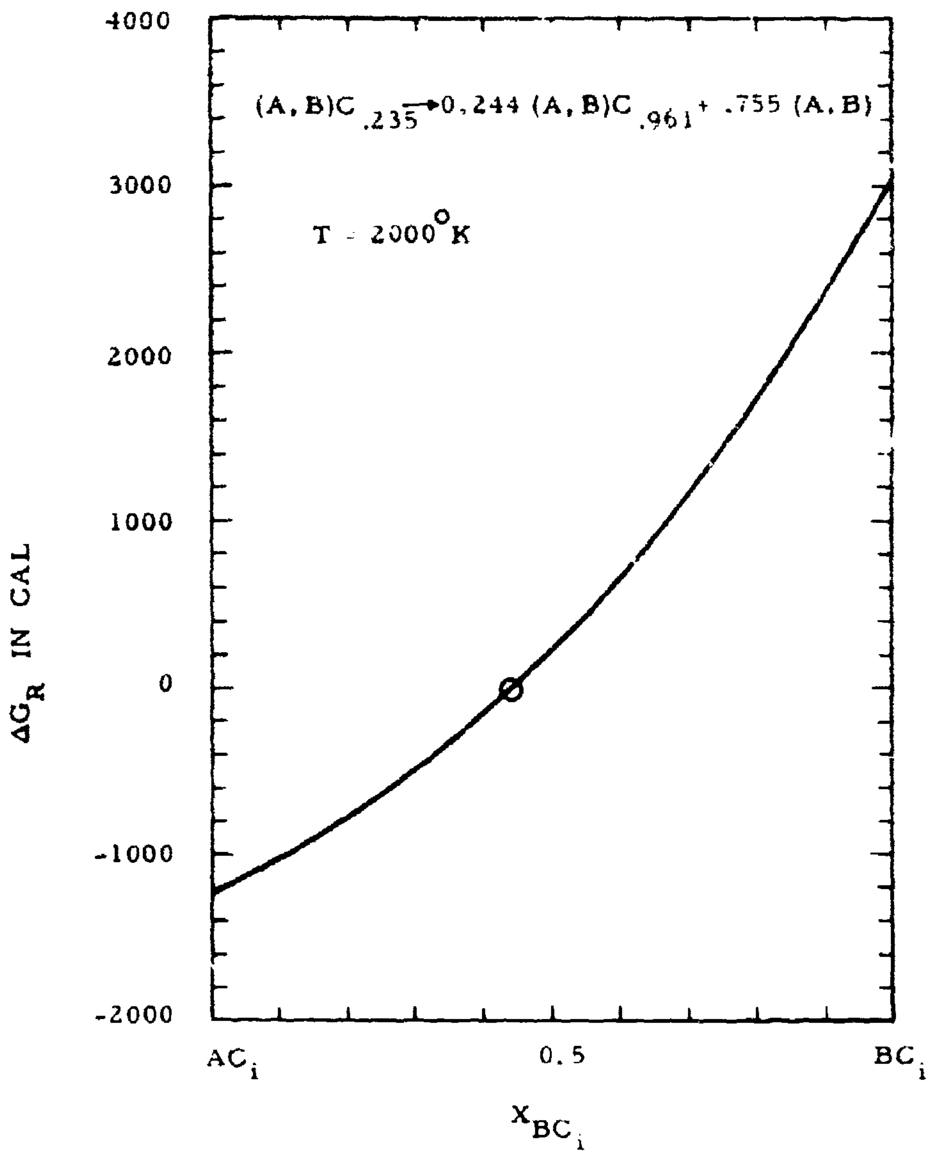

Figure 18. Integral Gibbs Free Energies of Decomposition of $(A,B)C_{235}$ to $(A,B)C_{0.961}$ and (A,B).

$$\frac{\partial \Delta G_{(A,B)C_i}}{\partial x_{BC_i}}$$

must all be necessarily equal, we need the additional stability condition in order to fix the compositions of the three phases. Expressing in thermodynamic terms, the stability condition simply says that at equilibrium the integral Gibbs free energy of decomposing an alloy $(A,B)C_v$ into two other alloys $(A,B)C_u$ and $(A,B)C_w$ must be zero. For the model example described here, the total Gibbs free energy of decomposition of the alloy $(A,B)C_{0.235}$ to the alloys $(A,B)C_{0.961}$ and (A,B) according to the following reaction is shown in Figure 18.

$$(A,B)C_{0.235} <ss> \rightarrow 0.244\,(A,B)C_{0.961} <ss> + 0.755(A,B)_{<ss>}$$

III. APPLICATION OF THE METHOD DEVELOPED TO THE SYSTEM Ta-Hf-C

As already discussed earlier, in order to precalculate the phase equilibria in a ternary system, it is necessary to know the free energies of the binary phases as a function of composition and the solution behaviors of the ternary phases. In the binary system Hf-C, one intermediate phase, hafnium monocarbide HfC (B1-type), forms with a wide range of homogeneity[4]. The α-Hf terminal phase is stabilized by the addition of carbon atoms to higher temperatures. However, since little is known about the free energy of this phase as a function of composition, we shall at the present ignore the stabilization of the α-phase to high temperatures and assume that metal and carbon components at the metal-rich boundaries of the monocarbide phase are in equilibrium with those in the β-Hf terminal phase at high temperatures.

In addition to the monocarbide phase (B1-type), a second intermediate phase, Ta_2C forms in the system tantalum-carbon. Tantalum subcarbide, Ta_2C, is polymorphic. The low-temperature form, α-Ta_2C, has a hexagonal close-packed arrangement of the metal atoms. The distribution of the carbon atoms among the interstitial sites is not known but undoubtedly dependent on temperature.

The high-temperature form β-Ta_2C, probably has the same arrangement of metal atoms[5] but differs in the degree of disorder in the carbon sublattice from the α-modification. Again, for the lack of pertinent thermodynamic data with respect to the two modifications of Ta_2C, we shall assume α-Ta_2C and β-Ta_2C to be of one phase for the calculation at the present.

The Gibbs free energies of the monocarbide and subcarbide phases will be represented by a sum of two contributions: the thermal and the configurational contributions to the total free energies of these phases, i.e.

$$\Delta G = \Delta G^{th} + \Delta G^{conf} \tag{39}$$

Since the free energies of formation[6] for both of the two monocarbide phases at the stoichiometric composition have been determined, we shall represent the free energies of formation due to thermal vibration for compositions less than $x_C = 0.5$ by,

$$\Delta G^{th} \text{ (B1 phase)} = 2(1-z)\,\Delta G_f + (1-2z)(a + bz + Cz^2) \tag{40}$$

where ΔG_f is the Gibbs free energy of formation of the stoichiometric monocarbide phase in cal/gatom alloy, z is the atom fraction of the carbon component, and a, b and c are parameters needed to be determined. When b and c are set equal to zero, we have the Wagner-Schottky[6, 7, 8, 9] vacancy model with 'a' being the Gibbs free energy of forming a carbon vacancy on the carbon sublattice. For the free energy contribution due to configurational mixing of carbon atoms among the various lattice sites, we assume ideal entropy of mixing which is reasonable at high temperatures. Accordingly, we have

$$\Delta G^{conf} = RT\left[z\ln\frac{z}{1-2z} + (1-z)\ln\frac{1-2z}{1-z}\right] \tag{41}$$

For the subcarbide phase, the thermal contribution to the total free energy will be represented by a 3rd degree polynominal, i.e.

$$\Delta G^{th} = a' + b'z + c'z^2 + d'z^3 \tag{40b}$$

For the configurational free energy, we shall use the model developed by Rudy[10]. In this model, one assumes that the two interstitial sites occupied by the carbon atoms are energetically different by an amount of ΔE. Expressing in terms of two gram atoms of metal (Me_2C_w), the configurational free energy as derived earlier[10] is

$$\Delta G^{conf}_{Me_2C_w} = z_B \Delta E + \Big[z_A \ln z_A + (1-z_A) \ln (1-z_A) + z_B \ln z_B + (1-z_B) \ln (1-z_B)\Big] RT \qquad (42)$$

where ΔE is the energy difference between the two interstitial sites in the cal/2-gram atom metal, z_A is the mole fraction of carbon atoms on the A-sites, $(1-z_A)$ is the mole fraction of vacant A-sites, z_B is the mole fraction of the carbon atoms on the energetically unfavorable B-sites and $(1-z_B)$ is the mole fraction of the vacant B-sites. It is understood that

$$z_A + z_B = w \qquad (43)$$

However, since it is more convenient to discuss the free energies of solid solutions in terms of one gram atom alloy, we shall divide equation (42) by $(2 + w)$ and change the variable from w to z (the atom fraction of carbon atoms in the solution). Accordingly, we have

$$\Delta G^{conf.}_{Me_{1-z}C_z} = \frac{1-z}{2} \Big[z_B \Delta E + \Big[z_A \ln z_A + (1-z_A) \ln (1-z_A) + z_B \ln z_B + (1-z_B) \ln (1-z_B)\Big] RT\Big] \qquad (44)$$

Using the well-known thermodynamic relationships relating the partial molar and integral quantities, we have,

$$\Delta\overline{G}^{\,conf}_{Me} = \Delta G - z\ \frac{d\Delta G}{dz}$$

$$= \frac{z_B \Delta E}{2} + \frac{RT}{2}\Big[z_A \ln z_A + (1-z_A)\ln(1-z_A) +$$

$$+ z_B \ln z_B + (1-z_B)\ln(1-z_B)\Big]$$

$$- \frac{z}{1-z}\left[\Delta E\, s + RT\, r \ln \frac{z_A}{1-z_A} + RT\, s \ln \frac{z_B}{1-z_B}\right]$$

$$\Delta\overline{G}^{\,conf}_{C} = \Delta G + (1-z)\ \frac{d\Delta G}{dz} \tag{45}$$

$$= \Delta E\, s + r\, RT \ln \frac{z_A}{1-z_A} + s\, RT \ln \frac{z_B}{1-z_B} \tag{46}$$

where

$$r = \frac{(1-z_A)^2(1+u)^2}{(1-z_A)^2(1+u)^2 + e^{-\Delta E/RT}} \tag{47}$$

$$s = \frac{(1-z_B)^2(1+v)^2}{(1-z_A)^2(1+u)^2 + e^{\Delta E/RT}} \tag{48}$$

$$u = \frac{z_A}{1-z_A}\ e^{-\frac{\Delta E}{RT}} \tag{49}$$

$$v = \frac{z_B}{1-z_B}\ e^{\frac{\Delta E}{RT}} \tag{50}$$

$$z_B = \frac{-\left[\frac{1-3z}{1-z} + e^{-\frac{\Delta E}{RT}}\left(\frac{1+z}{1-z}\right)\right] \pm \left[\left(\frac{1-3z}{1-z} + e^{-\frac{\Delta E}{RT}}\frac{1+z}{1-z}\right)^2 + \left(\frac{8z}{1-z}\right)e^{-\frac{\Delta E}{RT}}\left(1-e^{-\frac{\Delta E}{RT}}\right)\right]^{1/2}}{2\,(1-e^{\Delta E/RT})} \tag{51}$$

$$z_A = \frac{2z}{1-z} - z_B \qquad (52)$$

If we set $\Delta E = 0$, $z_A = z_B$ since there is no difference between the carbon atoms on the A-sites and B-sites. From equation (52) one obtains,

$$z_A = z_B = \frac{z}{1-z} \qquad (53)$$

Substituting for z_A and z_B according to equation (53), equation (44) reduced to equation (41).

Before we can proceed any further, we must know the values of a, b, c, (equation 40a) and of a', b', c' (equation 40b) for the monocarbide and subcarbide phases in addition to the values of ΔE for the two subcarbide phases. These parameters were determined by using the following conditions.

a. Values of ΔG_f for TaC and HfC are those selected by Chang[6].

b. The free energy decomposition of $TaC_{0.5}$ to 0.633 $TaC_{0.79}$ + 0.367 Ta is taken to be 2300 cal at 2273°K[6].

c. The free energy difference between $HfC_{0.5}$ and $TaC_{0.5}$ is taken to be 2250 cal at 2273°K[6].

d. The phase boundaries of the subcarbide and monocarbide phases are those determined by Rudy[4] and Rudy and Harmon[5].

e. The partial molar quantities in the two-phase regions: metal-subcarbide, subcarbide-monocarbide and monocarbide-graphite must be all the same.

With these five conditions, we have obtained the following results for our computation:

HfC-phase (B1-type)

$$2\Delta G_{f,0.5} = -52,350 + 2.08\ T \qquad 298.15°-2073°K$$
$$= -55,630 + 3.66\ T \qquad 2073°-2491°K$$

$a = 250,154;\quad b = -968,150;\quad c = 1,113,170$

Hf_2C-phase

$a' = 72,445;\quad b' = -466,000;\quad c' = 650,000;\quad d' = 0;\ \Delta E = 0$

TaC-phase (B1-type)

$$\Delta G_{f,0.5} = -35,335 - 1.7949 \log T + 6.4757\ T$$

$a = 363,950;\quad b = 1,527,083;\quad c = 1,770,833$

Ta_2C-phase

$a' = 129,848;\quad b' = 1,336,117;\quad c' = -4,7\text{[illegible]}6,667;\quad d' = 5,373,333$

$\Delta E = 4000$ cal.

The phase equilibria in the system Ta-Hf-C at 2273°K and 1273°K are calculated and presented in Figures 19 and 20. The phase diagrams calculated using the simplified method are shown in Figures 21 and 22 for comparison. As might be expected, the marked difference between the calculated phase diagrams by the two different methods is the homogeneous range of the monocarbide phase. The concentration-free-energy-gradient curves for the metal, subcarbide and monocarbide phases as well as the integral free energy of decomposition of the subcarbide phase into the metal and monocarbide phases at 2273°K used to establish the phase equilibria are presented in Figures 23 and 24. Similar curves used to determine the phase equilibria at 1773°K are not included in this report.

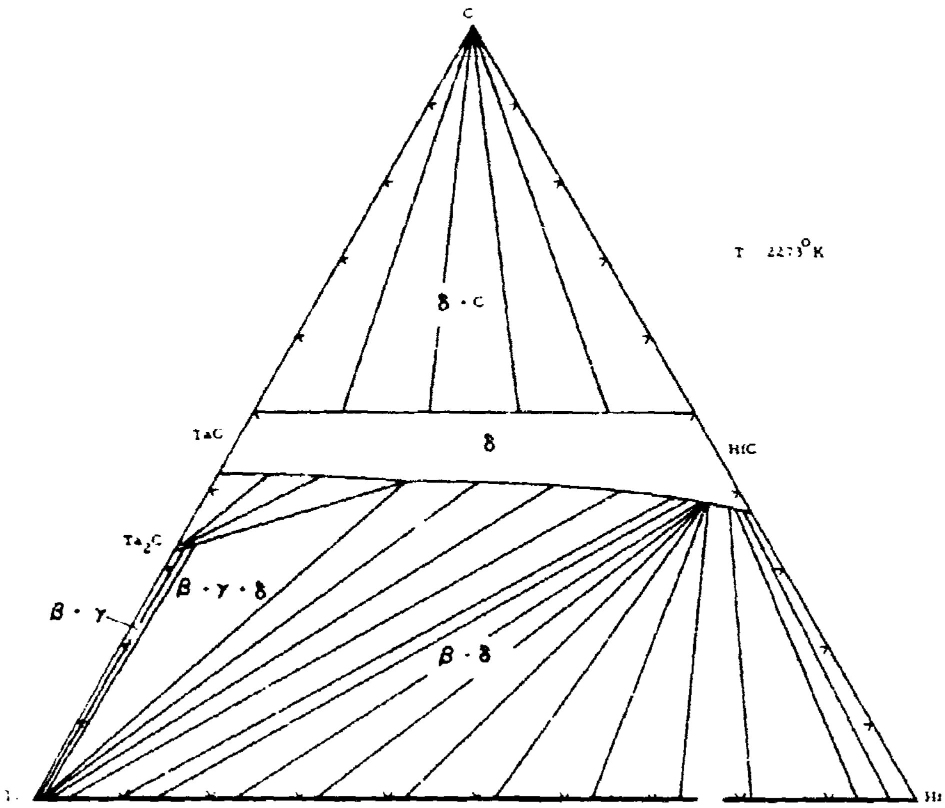

Figure 19. Calculated Phase Equilibria in the System Ta-Hf-C Using the General Method at 2273°K.

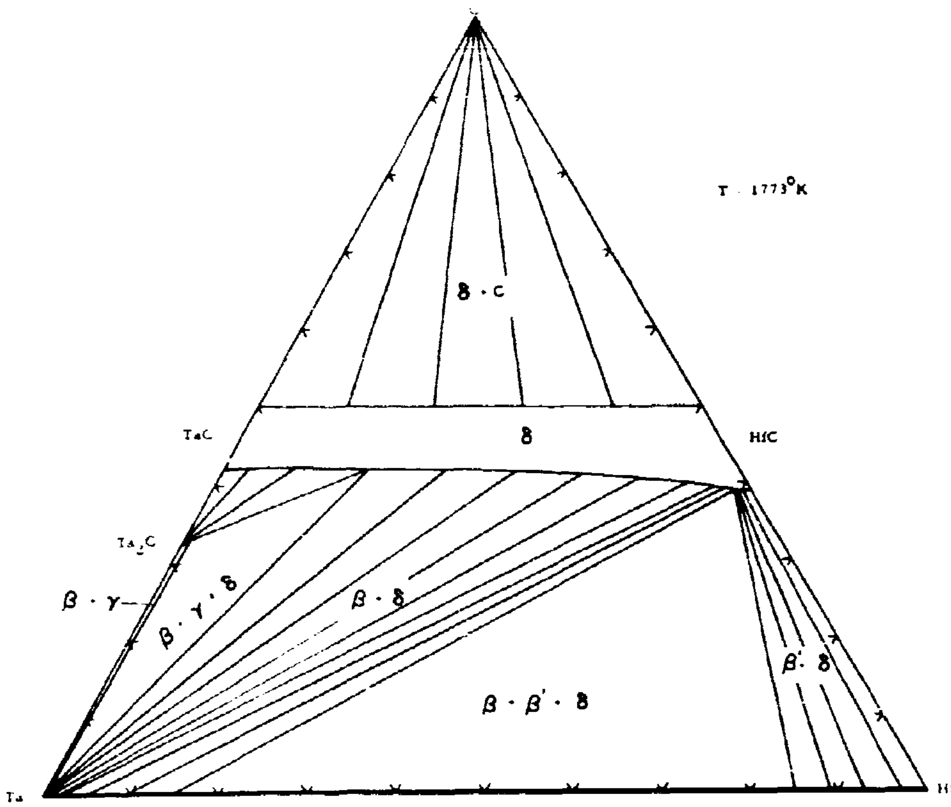

Figure 20. Calculated Phase Equilibria in the System Ta-Hf-C Using the General Method at 1773°K.

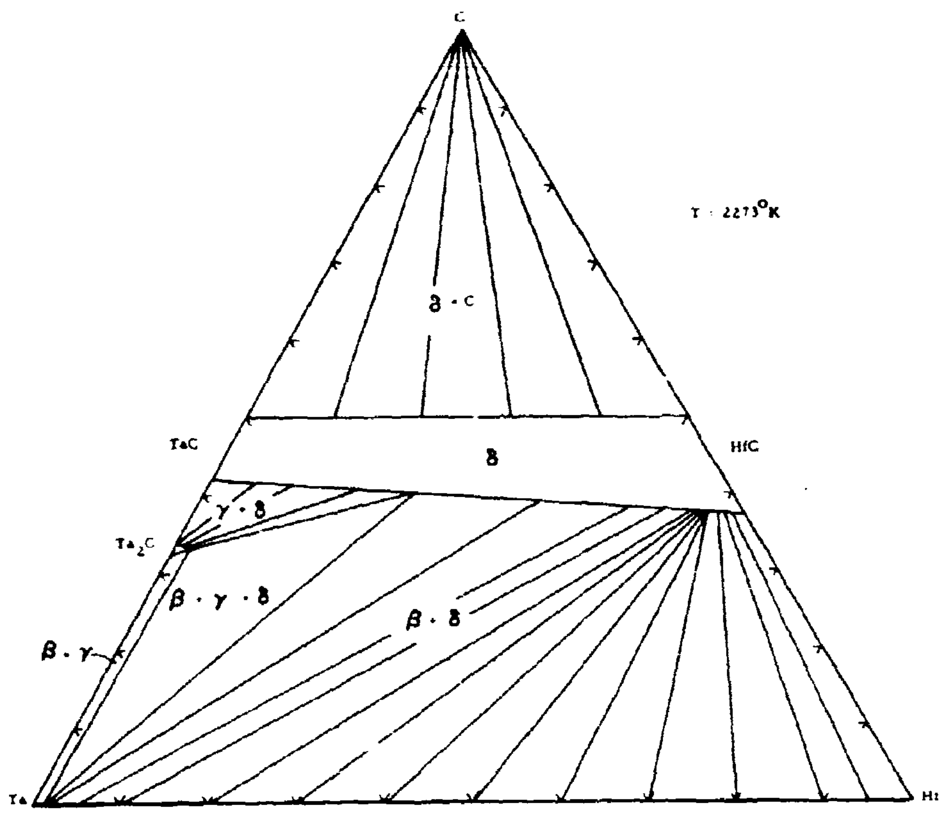

Figure 21. Calculated Phase Equilibria in the System Ta-Hf-C Using the Simplified Method at 2273°K.

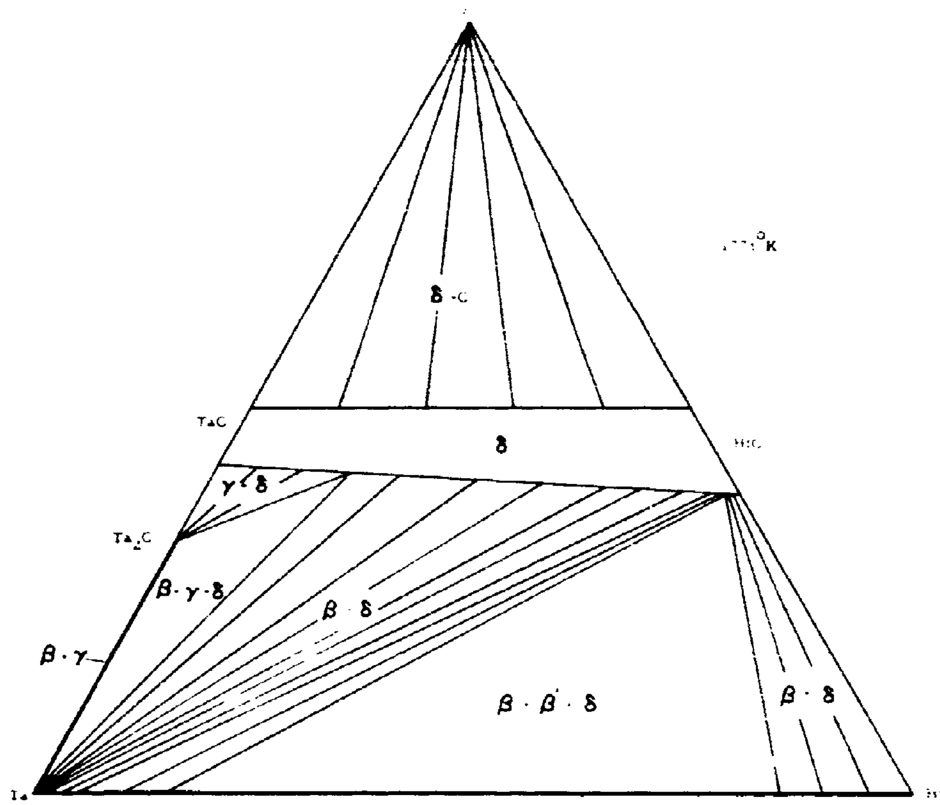

Figure 22. Calculated Phase Equilibria in the System Ta-Hf-C Using the Simplified Method at 1773°K.

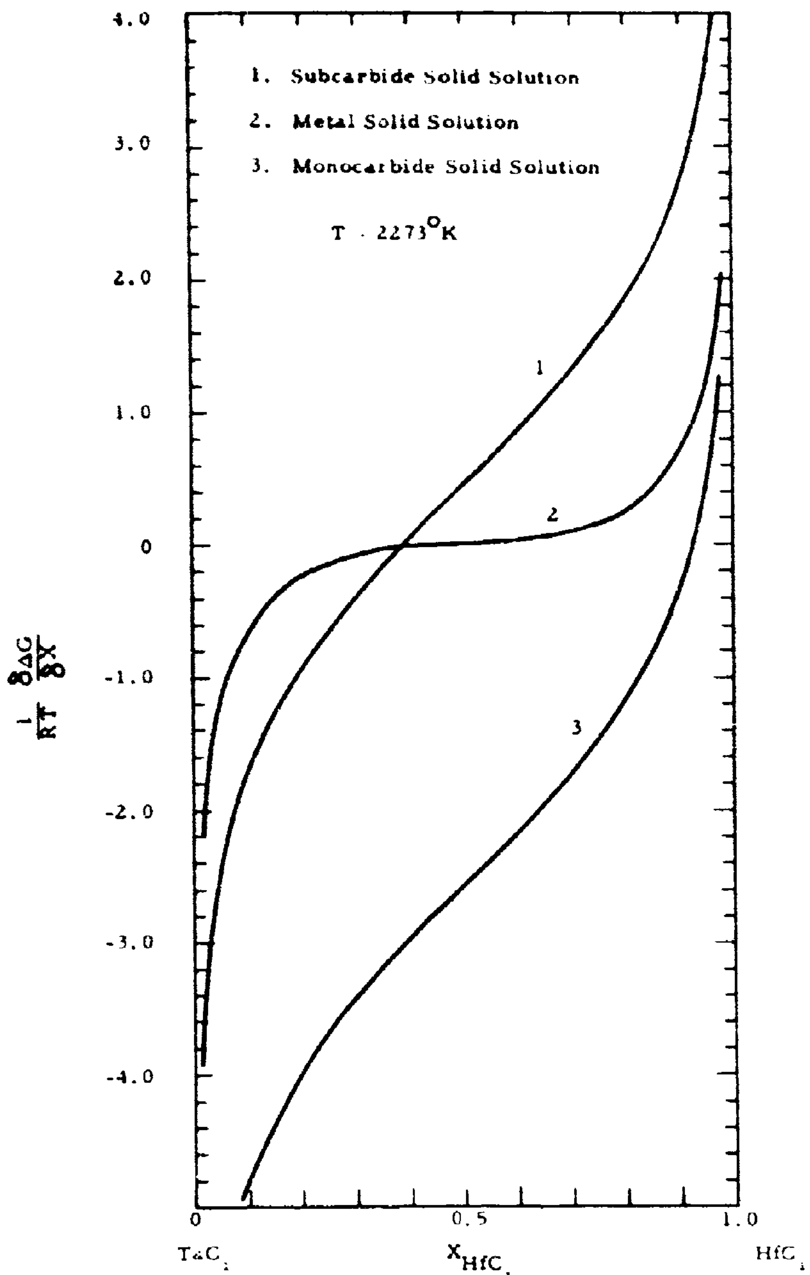

Figure 23. Concentration Free-Energy-Gradient Curves for the Metal, Subcarbide and Monocarbide Solid Solutions.

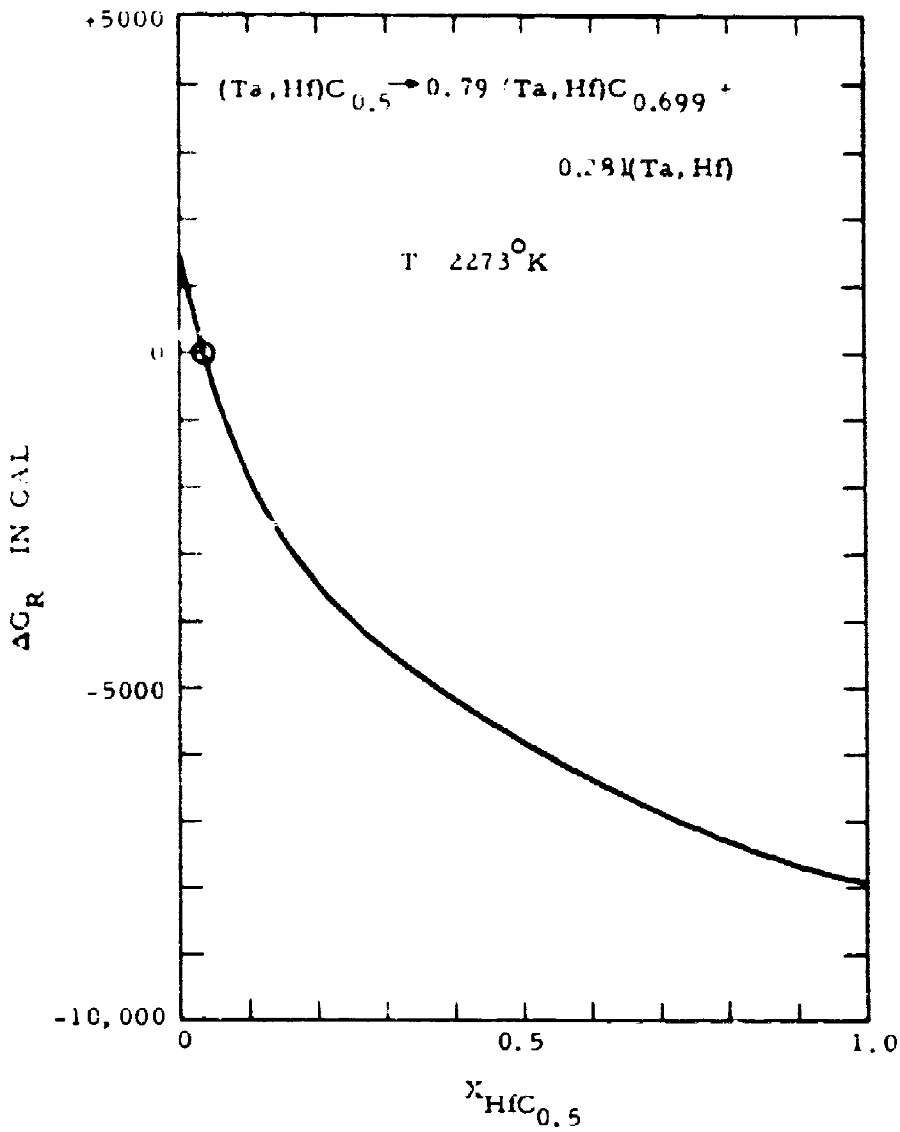

Figure 24. Integral Free Energy of Decomposition of the Subcarbide Phase into the Metal and Monocarbide Phases at 2273°K.

IV. DISCUSSION

The method described in this report has demonstrated that the phase equilibria in three-component systems can be adequately predicted from binary data based on the first principle provided that the Gibbs free energies of formation of all the binary phases are accurately known as a function of composition and temperature. Moreover, one must also know the solution behaviors of the ternary alloys. It has been demonstrated previously[2, 3], that for the interstitial type of solid solution such as the transition metal carbides, the regular solution theory with proper values of the interaction parameters, is adequate in most cases.

The general method also demonstrates that the simplified method originally developed by Rudy for predicting phase equilibria in three-component systems is sufficiently accurate. In cases where the homogeneous ranges of the ternary solid solutions with respect to the interstitial component such as carbon are small, the simplified method would predict the identical results as the general method. Only, when the homogeneous ranges of a ternary solid solution with respect to the interstitial element change drastically with the metal exchange, will there be a difference between the phase diagrams calculated using the two different methods. In practice, the phase equilibria in a ternary system predicted by the simplified method are sufficiently accurate in most cases.

REFERENCES

1. E. Rudy: Z. Metallk. 54 ,112 (1963).
2. E. Rudy: Z. Metallk. 54 213 (1963).
3. E. Rudy and Y.A. Chang: Plansee Proceedings 1964, p786, Metallk Plansee, AG., Reutte, Tirol (1965).
4. E. Rudy: Hf-C Phase Diagram, AFML-TR-65-2, Part I, Vol. IV. (1965)
5. E. Rudy and D.P. Harmon: Ta-C Phase Diagram, AFML-TR-65-2, Part I, Vol. V (1965).
6. Y.A. Chang: AFML-TR-65-2, Part IV, Vol. I (1965).
7. C. Wagner: Thermodynamics of Alloys, Addison-Wesley, Reeding, Mass. (1952).
8. C. Wagner and W. Schottky: Z. Physik. Chem. 11 1963 (1931).
9. L. Kaufman and A. Sarney: Compounds of Interest in Nuclear Reactor Technology, American Institute of Mining, Metallurgical and Petroleum Engineers, Inc., 267 (1964).
10. E. Rudy, St. Windisch, and Y.A. Chang: AFML-TR-65-2, Part I, Vol. I (1965).

APPENDIX I

COMPUTER PROGRAMS FOR TERNARY PHASE DIAGRAM CALCULATIONS (Ray Marler)

```
$IBFTC K-1     FULIST
      DIMENSION X(50),Y(50),A3(3),A4(3),TITLE(13)
      NAMELIST/INPUT/NOXYS,IPRINT,CASE,DUMP,XLIM,R,T,EPSP,EPSPP,A3,A4,
     1X,Y ,K1,K2
      NAMELIST/DTADMP/LINE,K,J,XP,YP,XPP,YPP,ZPP,DG,DG3,DG4,GNU1,GNU2,
     1FXP,FYP,DG1,C1,C2,FC1,FC2,DG2
    1 FORMAT(1H139X53HA E R O J E T - G E N E R A L   C O R P O R A T I
     10 N/1H039X53HC O M P U T I N G   S C I E N C E S   D I V I S I O N
     2/1H050X31HS A C R A M E N T O   P L A N T//1H057X17HP R O G R A M
     3 K/1H027X77HT W O - P H A S E   E Q U I L I B R I U M   I N   T E
     4 R N A R Y   S Y S T E M/16HOC U S T O M E R 97X19HP R O G R A M M
     5 E R/12HOY. A. CHANG109X11HR.H. MARLER//)
      READ(5,17)TITLE
    2 READ (5,INPUT)
      WRITE(6,1)
      WRITE(6,18)TITLF
      WRITE(6,INPUT )
      LIM = XLIM + .005
      NDUMP1= DUMP
      NDUMP2 = INT(DUMP*XLIM+.005) - NDUMP1*LIM
      RT = R*T
      WRITE(6,1)
      WRITE(6,18)TITLF
      DO 14 I = 1, NOXYS
      WRITE(6,16)X(I),Y(I)
      PASS = 1.
      L = 1
      LINF = 1
      LIMX = XLIM * X(I) + .005
      BLIMX = LIMX
      JMAX = LIM - LIMX
      DO 12 K = K1,K2
      CAY = K
      XPP = CAY/XLIM
      DO 12 J = 1, JMAX
      XJ = J
      XP = (BLIMX + XJ)/XLIM
      ZPP = (XP - XPP)*(1.-X(I)-Y(I))/(XP - X(I))
      IF(ZPP.GE.1..OR.ZPP.LE.0..OR.XP.FQ.XPP.OR.XP.EQ.X(I))GO TO 12
      DG3 = ((A3(3)*ZPP+A3(2))*ZPP+A3(1))/(1.-ZPP)
      DG4 = ((A4(3)*ZPP+A4(2))*ZPP+A4(1))/(1.-ZPP)
      GNU1 = (X(I)-XPP)/(XP-XPP)
      GNU2 = (XP-X(I))/(XP-XPP)
      IF(GNU1.LT.0..OR.GNU1.GT.1..OR.GNU2.LT.0..OR.GNU2.GT.1.)GO TO 12
      YP = 1.-XP
      YPP = 1. -XPP - ZPP
      CALL XLNX(XP,FXP,$12)
      CALL XLNX(YP,FYP,$12)
      DG1 = (FXP+FYP)*RT+XP*YP*EPSP
      C1 = XPP/(1.-ZPP)
      C2 = YPP/(1.-ZPP)
      CALL XLNX(C1,FC1,$12)
      CALL XLNX(C2,FC2,$12)
      DG2=((FC1+FC2)*RT+C1*C2*EPSPP+C1*DG3+C2*DG4)*(1.-ZPP)
      DG    =GNU1*DG1+GNU2*DG2
      IF(DG.GE.0.)GO TO 12
      IF(PASS.NE.1.)GO TO 5
      PASS = 0.
      GO TO 6
```

```
 5 IF (DG -  DGOLD)6,8,8
 6 DGOLD = DG
   XPS = XP
   YPS = YP
   XPPS= XPP
   YPPS= YPP
   ZPPS= ZPP
   LSAVE = LINE
 8 IF(NDUMP1.NE.K.OR.NDUMP2.NE.J)GO TO 9
   WRITE (6,DTADMP)
   GO TO 11
 9 IF(IPRINT.EQ.0)WRITE(6,10)LINE,XP,YP,XPP,YPP,ZPP,DG
10 FORMAT(1H 15X14,6E16.8)
11 LINE = LINE + 1
12 CONTINUE
   WRITE(6,10)LSAVE,XPS,YPS,XPPS,YPPS,ZPPS,DGOLD
14 CONTINUE
   CASE = CASE - 1.
   IF(CASE.GT.0.)GO TO 2
15 STOP
16 FORMAT(1H019X 4HX = E15.8,10X,4HY = E15.8/1H 15X4HLINE7X2HXP14X
  12HYP13X3HXPP13X3HYPP13X3HZPP12X5HDEL G)
17 FORMAT(13A6)
18 FORMAT(1H027X13A6)
   END
```

```
$IBFTC K-1-A    FULIST
      DIMENSION X(50),Y(50),TITLE(13),VA(5),VB(5),RA(5),RB(5)
      NAMELIST/INPUT/NOXYS,IPRINT,CASE,DUMP,XLIM,R,T,EPSP,EPSPP,VA,VB,
     1     X,Y,K1,K2,D3,D4,RA,RB,DEACV,DEBCV,DGACU,DGBCU
      NAMELIST/DTADMP/K,J,XP,YP,XPP,YPP,ZPP,DG,DGACV,DGBCV,GNU1,GNU2,
     1FXP,FYP,DG1,C1,C2,FC1,FC2,DG2
    1 FORMAT(1H139X53HA E R O J E T - G E N E R A L   C O R P O R A T I
     1O N/1H039X53HC O M P U T I N G   S C I E N C E S   D I V I S I O N
     2/1H050X31HS A C R A M E N T O   P L A N T//1H057X17HP R O G R A M
     3  K/1H027X77HT W O - P H A S E   E Q U I L I B R I U M   I N   T E
     4 R N A R Y   S Y S T E M/16HOC U S T O M E R 97X19HP R O G R A M M
     5 E R/12HOY. A. CHANG109X11HR.H. MARLER//)
      READ(5,17)TITLE
    2 READ (5,INPUT)
      WRITE(6,1)
      WRITE(6,18)TITLE
      WRITE(6,INPUT )
      LIM = XLIM + .005
      NDUMP1= DUMP
      NDUMP2 = INT(DUMP*XLIM+.005) - NDUMP1*LIM
      FLAG = NDUMP1
      RT = R*T
      WRITE(6,1)
      WRITE(6,18)TITLE
      DO 14 I = 1, NOXYS
      WRITE(6,16)X(I),Y(I)
      PASS = 1.
      LIMX = XLIM * X(I) + .005
      BLIMX = LIMX
      JMAX = LIM - LIMX
      DO 12 K = K1,K2
      CAY = K
      XPP = CAY/XLIM
      DO 12 J = 1, JMAX
      XJ = J
      XP = (BLIMX + XJ)/XLIM
      IF(XP.EQ.XPP.OR.XP.EQ.X(I))GO TO 12
      ZPP = (XP - XPP)*(1.-X(I)-Y(I))/(XP - X(I))
      IF(ZPP.EQ..5)ZPP = .499
      IF(ZPP.GT.D4.OR.ZPP.LT.D3)GO TO 12
      CALL DLGC(DGACV,R,T,ZPP,VA,RA,DEACV,FLAG,$12)
      CALL DLGC(DGBCV,R,T,ZPP,VB,RB,DEBCV,FLAG,$12)
      GNU1 = (X(I)-XPP)/(XP-XPP)
      GNU2 = (XP-X(I))/(XP-XPP)
      IF(GNU1.LT.0..OR.GNU1.GT.1..OR.GNU2.LT.0..OR.GNU2.GT.1.)GO TO 12
      YP = 1.-XP
      YPP = 1. -XPP - ZPP
      CALL XLNX(XP,FXP,$12)
      CALL XLNX(YP,FYP,$12)
      DG1 = (FXP+FYP)*RT+XP*YP*EPSP+DGACU*XP+DGBCU*YP
      C1 = XPP/(1.-ZPP)
      C2 = YPP/(1.-ZPP)
      CALL XLNX(C1,FC1,$12)
      CALL XLNX(C2,FC2,$12)
      DG2=((FC1+FC2)*RT + C1*C2*EPSPP + C2*DGBCV + C1*DGACV)*(1.-ZPP)
      DG    =GNU1*DG1+GNU2*DG2
      IF(DG.GE.0.)GO TO 12
      IF(PASS.NE.1.)GO TO 6
      PASS = 0.
```

```
      GO TO 6
    5 IF (DG - DGOLD)6,8,8
    6 DGOLD = DG
      XPS = XP
      YPS = YP
      XPPS= XPP
      YPPS= YPP
      ZPPS= ZPP
    8 IF(NDUMP1.NE.K.OR.NDUMP2.NE.J)GO TO 9
      WRITE (6,DTADMP)
      FLAG = 0
      GO TO 12
    9 IF(IPRINT.EQ.0)WRITE(6,10)      XP,YP,XPP,YPP,ZPP,DG
   10 FORMAT(1H 19X5F16.8,F16.4)
   12 CONTINUE
      WRITE(6,10)       XPS,YPS,XPPS,YPPS,ZPPS,DGOLD
   14 CONTINUE
      CASE = CASE - 1.
      IF(CASE.GT.0.)GO TO 2
   15 STOP
   16 FORMAT(1H023X4HX = F10.8,5X,4HY = F10.8/1H 26X2HXP14X
     12HYP13X3HXPP13X3HYPP13X3HZPP12X5HDEL G)
   17 FORMAT(13A6)
   18 FORMAT(1H027X13A6)
      END
```

```
$IBFTC K-2      FULIST
      DIMENSION X(50),Y(50),Z(50),TITLE(13),CDG3(3),CDG4(3),CDG5(3),
     1CDG6(3)
      NAMELIST/INPUT/        A,B,C,D,R,T,DELTA,EPSP,EPSPP,CDG3,CDG4,
     1CDG5,CDG6,X,Y,Z
    1 FORMAT(1H139X53HA E R O J E T - G E N E R A L   C O R P O R A T I
     1O N/1H039X53HC O M P U T I N G   S C I E N C E S   D I V I S I O N
     2/1H050X31HS A C R A M E N T O   P L A N T//1H057X17HP R O G R A M
     3  K/1H019X93HG E N E R A L   T W O - P H A S E   E Q U I L I B R I
     4U M   I N   T E R N A R Y   S Y S T E M/16H0C U S T O M E R 97X
     519HP R O G R A M M E R/12H0Y  A. CHANG109X11HR.H. MARLER//)
      READ(5,3)TITLE,ID
    3 FORMAT(13A6,I2)
    2 READ (5,INPUT)
      WRITE (6,1)
      WRITE(6,4)TITLE,ID
    4 FORMAT(1H026X13A6,25XI2)
      WRITE (6,INPUT)
      WRITE(6,1)
      WRITE(6,16)TITLE
   16 FORMAT(1H026X13A6// 7X1HX12X1HY12X1HZ11X2HXP11X2HYP11X2HZP11X3HXPP
     111X3HYPP11X3HZPP 9X7HDELTA G/)
      RT = R*T
      DO 14 I = 1,50
      IF(X(I).EQ.0..AND.Y(I).EQ.0..AND.Z(I).EQ.0.)GO TO 15
      IPASS = 1
   17 IF(A.EQ.X(I))A = X(I)+DELTA/10.
      XPP = A
    5 IF(XPP - X(I)) 6,13,7
    6 XP = X(I) + DELTA
      XPMAX  = 1.
      GO TO 8
    7 XP = 0.
      XPMAX = X(I)
    8 ZP = 0
   12 YP = 1.-XP-ZP
      IF(XP.EQ.XPP)GO TO 11
      GNU1 =(X(I)-XPP)/(XP-XPP)
      IF(GNU1.GE.1..OR.GNU1.LT.0.)GO TO 11
      GNU2 = 1.-GNU1
      YPP =(Y(I)-GNU1*YP)/GNU2
      ZPP =1.-XPP-YPP
      IF(ZPP.GE.1..OR.ZPP.LT.0.)GO TO 11
      CALL CUBIC(CDG3,ZPP,DG3,$11)
      CALL CUBIC(CDG4,ZPP,DG4,$11)
      U1 = XPP/(1.-ZPP)
      U2 = YPP/(1.-ZPP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG2=((FU1+FU2)*RT+U1*U2*EPSPP +U2*DG4 + U1*DG3)*(1.-ZPP)
      CALL CUBIC(CDG5,ZP,DG5,$11)
      CALL CUBIC(CDG6,ZP,DG6,$11)
      U1= XP/(1.-ZP)
      U2= YP/(1.-ZP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG1=((FU1+FU2)*RT + U1*U2*EPSP +U2*DG6 + U1*DG5)*(1.-ZP)
      DG = GNU1*DG1 + GNU2 * DG2
      IF(IPASS.EQ.0)GO TO 19
```

```
      IPASS = 0
      DGS = DG
   19 IF(DG.GE.DGS)GO TO 9
      DGS = DG
      XPS = XP
      YPS = YP
      XPPS =XPP
      YPPS=YPP
      ZPS = ZP
      ZPPS= ZPP
    9 IF(ID.NE.0)WRITE(6,10)X(I),Y(I),Z(I),XP,YP,ZP,XPP,YPP,ZPP,DG
   10 FORMAT(1H 9F13.8,F13.4)
   11 ZP = ZP + DELTA
      IF(ZP.LE.D)GO TO 12
      XP = XP + DELTA
      IF(XP.LT.XPMAX)GO TO 8
   13 XPP = XPP + DELTA
      IF(XPP.LE.B)GO TO 5
      WRITE(6,10)X(I),Y(I),Z(I),XPS,YPS,ZPS,XPPS,YPPS,ZPPS,DGS
   14 CONTINUE
   15 GO TO 2
      END
```

```
$IBFTC K-2-A    FULIST
      DIMENSION X(50),Y(50),Z(50),TITLE(13),VACV(5),VBCV(5),VACU(5),
     1       VBCU(5),RACV(5),RACU(5),RBCV(5),RBCU(5)
      NAMELIST/INPUT/   A,B,C,D,E,F,R,T,DELTA,EPSP,EPSPP,DEACV,DEBCV,
     1DEACU,DEBCU,VACV,VBCV,VACU,VBCU,RACV,RBCV,RACU,RBCU,X,Y,Z
    1 FORMAT(1H139X53HA E R O J E T - G E N E R A L   C O R P O R A T I
     10 N/1H039X53HC O M P U T I N G   S C I E N C E S   D I V I S I O N
     2/1H050X31HS A C R A M E N T O   P L A N T//1H057X17HP R O G R A M
     3 K/1H019X93HG E N E R A L   T W O - P H A S E   E Q U I L I B R I
     4U M   I N   T E R N A R Y   S Y S T E M/16H0C U S T O M E R 97X
     519HP R O G R A M M E R/12H0Y. A. CHANG109X11HR.H. MARLER//)
      READ(5,3)TITLE,ID
    3 FORMAT(13A6,I2)
    2 READ (5,INPUT)
      WRITE (6,1)
      WRITE(6,4)TITLE,ID
    4 FORMAT(1H026X13A6,25XI2)
      WRITE (6,INPUT)
      WRITE(6,1)
      WRITE(6,16)TITLE
   16 FORMAT(1H026X13A6// 7X1HX12X1HY12X1HZ11X2HXP11X2HYP11X2HZP11X3HXPP
     111X3HYPP11X3HZPP 9X7HDELTA G/)
      FLAG =ID/10
      RT = R*T
      DO 14 I = 1,50
      IF(X(I).EQ.0..AND.Y(I).EQ.0..AND.Z(I).EQ.0.)GO TO 15
      IPASS = 1
      XPP = A
      DO 17 IXPP=1,1000
      IF(XPP-X(I))6,13,7
    6 XP = X(I) + DELTA
      XPMAX  = F
      GO TO 8
    7 XP = E
      XPMAX = X(I)
    8 DO 16 IXP=1,1000
      ZP=C
      DO 22 IZP=1,1000
      YP=1.-XP-ZP
      IF(XP.EQ.XPP)GO TO 11
      GNU1 =(X(I)-XPP)/(XP-XPP)
      IF(GNU1.GE.1..OR.GNU1.LT.0.)GO TO 11
      GNU2 = 1.-GNU1
      YPP =(Y(I)-GNU1*YP)/GNU2
      ZPP =1.-XPP-YPP
      IF(ZPP.GE.1..OR.ZPP.LT.0.)GO TO 11
      CALL DLGC(DGACV,R,T,ZPP,VACV,RACV,DEACV,FLAG,$11)
      CALL DLGC(DGBCV,R,T,ZPP,VACV,RBCV,DEBCV,FLAG,$11)
      U1 = XPP/(1.-ZPP)
      U2 = YPP/(1.-ZPP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG2=((FU1+FU2)*RT+U1*U2*EPSPP+U2*DGBCV+U1*DGACV)*(1.-ZPP)
      CALL DLGC(DGACU,R,T,ZP,VACU,RACU,DEACU,FLAG,$11)
      CALL DLGC(DGBCU,R,T,ZP,VBCU,RBCU,DEBCU,FLAG,$11)
      U1= XP/(1.-ZP)
      U2= YP/(1.-ZP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
```

```
      DG1 =((FU1+FU2)*RT+U1*U2*EPSP+U2*DG3CU+U1*DGACU)*(1.-ZP)
      DG = GNU1*DG1 + GNU2 * DG2
      IF(IPASS.EQ.0)GO TO 19
      IPASS = 0
      DGS = DG
   19 IF(DG.GE.DGS GO TO 9
      DGS = DG
      XPS = XP
      YPS = YP
      XPPS =XPP
      YPPS=YPP
      ZPS = ZP
      ZPPS= ZPP
    9 IF(ID.NE.0)WRITE(6,10)X(I),Y(I),Z(I),XP,YP,ZP,XPP,YPP,ZPP,DG
   10 FORMAT(1H 9F13.8,F13.-)
   11 ZP = ZP + DELTA
      IF(ZP.GT.0)GO TO 12
   22 CONTINUE
      WRITE(6,23)
   23 FORMAT( 25H0THE ZP LOOP WAS EXCEEDED)
      GO TO 5
   12 XP =XP+DELTA
      IF(XP.GE.XPMAX)GO TO 13
   18 CONTINUE
      WRITE(6,20)
   20 FORMAT(25H0THE XP LOOP WAS EXCEEDED)
      GO TO 5
   13 XPP = XPP + DELTA
      IF(XPP.GT.0)GO TO 5
   17 CONTINUE
      WRITE(6,21)
   21 FORMAT(26H0THE XPP LOOP WAS EXCEEDED)
    5 WRITE(6,10)X(I),Y(I),Z(I),XPS,YPS,ZPS,XPPS,YPPS,ZPPS,DGS
   14 CONTINUE
   15 GO TO 2
      END
```

```
$IBFTC N-3      FULIST
      DIMENSION TITLE(13)
      COMMON     CDGACV(3),CDGBCV(3),CDGACW(3),CDGBCW(3)
      NAMELIST/INPUT/B1,B2,DELTA,X,Y,Z,XP,YP,ZP,XPP,YPP,ZPP,R,T,M,
     1CDGACV,CDGBCV,CDGACW,CDGBCW,EPSP,EPSPP,EPSPPP
    1 FORMAT(1H139X53HA E R O J E T - G E N E R A L   C O R P O R A T I
     10 N/1H039X53HC O M P U T I N G   S C I E N C E S   D I V I S I O N
     2/1H050X31HS A C R A M E N T O   P L A N T//1H057X17HP R O G R A M
     3  K/1H019X93HG E N E R A L   T W O - P H A S E   E Q U I L I B R I
     4 U M   I N   T E R N A R Y   S Y S T E M/16HOC U S T O M E R 97X
     519HP R O G R A M M E R/12HOY. A. CHANG109X11HR.H. MARLER//)
      READ(5,3)TITLE,ID
    3 FORMAT(13A6,I2)
    2 READ (5,INPUT)
      WRITE (6,1)
      WRITE(6,4)TITLE,ID
    4 FORMAT(1H026X13A6,25XI2)
      WRITE (6,INPUT)
      WRITE(6,1)
      WRITE(6,5)TITLE
      RT=R*T
      CALL DGACVS(DGACV,ZPP ,$11)
      CALL DGBCVS(DGBCV,ZPP ,$11)
      U1=XP/(1.-ZP)
      U2=YP/(1.-ZP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG1 =    (FU1+FU2)*RT + EPSP*U1*U2
      U1=XPP/(1.-ZPP)
      U2=YPP/(1.-ZPP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG2 =   ((FU1+FU2)*RT +EPSPP*U1*U2 + DGBCV*U2 + DGACV*U1)*(1.-ZPP)
   13 IPASS=1
      DYP = YP-Y
      DYPP=YPP-Y
      DZP = ZP-Z
      DZPP=ZPP-Z
      AREA2 =DYPP*DZP - DYP*DZPP
      ZPPP= B1
    7 YPPP=0.
      YPPPMX=1.-ZPPP
      DZPPP=ZPPP-Z
    8 DYPPP=YPPP-Y
      AREA1 =DYP*DZPPP-DYPPP*DZP
      AREA3 =DYPPP*DZPP - DYPP*DZPPP
      IF(.NOT.(AREA1.LT.0..AND.AREA2.LT.0..AND.AREA3.LT.0..OR.AREA1.GT.
     10..AND.AREA2.GT.0..AND.AREA3.GT.0.))GO TO 11
      XPPP=1.-YPPP-ZPPP
      CALL DGACWS(DGACW,ZPPP,$11)
      CALL DGBCWS(DGBCW,ZPPP,$11)
      U1=XPPP/(1.-ZPPP)
      U2=YPPP/(1.-ZPPP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG3 = ((FU1+FU2)*RT+EPSPPP*U1*U2 + DGBCW*U2 + DGACW*U1)*(1.-ZPPP)
      D = (YPP  - YP)*(ZPPP- ZP) - (ZPP  - ZP) * (YPPP- YP)
      IF(D.EQ.0.)GO TO 11
      GMU1 = ((YPP-Y)*(ZPPP-Z) - (ZPP-Z)*(YPPP-Y))/D
```

```
      IF(GNU1.LE.0..OR.GNU1.GT.1.)GO TO 11
      GNU2 = ((Y-YP)*(ZPPP-ZP) - (Z-ZP)*(YPPP-YP))/D
      IF(GNU2.LE.0..OR.GNU2.GT.1.)GO TO 11
      GNU3 =1.-GNU1-GNU2
      DG = GNU1* DG1 + GNU2 *DG2 + GNU3*DG3
      IF(IPASS.EQ.0)GO TO 9
      IPASS = 0
      DGS = DG
    9 IF(DG.GE.DGS)GO TO 10
      DGS = DG
      XPPPS = XPPP
      YPPPS = YPPP
      ZPPPS = ZPPP
   10 IF(ID.NE.0)WRITE(6,14)X,Y,Z,XP,YP,ZP,XPP,YPP,ZPP,XPPP,YPPP,ZPPP,DG
   11 YPPP = YPPP + DELTA
      IF(YPPP.LE.YPPPMX)GO TO 8
      ZPPP = ZPPP + DELTA
      IF(ZPPP.LE.B2) GO TO 7
      WRITE(6,14)X,Y,Z,XP ,YP ,ZP ,XPP ,YPP ,ZPP ,
     1XPPPS,YPPPS,ZPPPS,DGS
      M = M -1
      IF(M.LT.0)GO TO 2
      IF(DGS.GE.0.)GO TO 12
      B1 =AMAX1(B1,DGS-DELTA)
      B2 =AMIN1(B2,DGS+DELTA)
   12 DELTA = DELTA/10.
      GO TO 13
   14 FORMAT(1H 12F10.7,F10.3)
    5 FORMAT(1H026X
     1          13A6//130H      X           Y           Z          XP          Y
     1P          ZP          XPP         YPP         ZPP         XPPP        YPPP
     2    ZPPP      DELTA G/)
      END
```

```
$IBFTC K-3-A    FULIST
      DIMENSION TITLE(13),RACV(5),RBCV(5),RACW(5),RBCW(5),VACV(5),
     1     VBCV(5),VACW(5),VBCW(5)
      NAMELIST/INPUT/B1,B2,DELTA,X,Y,Z,XP,YP,ZP,XPP,YPP,ZPP,R,T,M,EPSP,
     1     EPSPP,EPSPPP,DEACV,DEBCV,DEACW,DEBCW,VACV,VBCV,VACW,VBCW,
     2     RACV,RBCV,RACW,RBCW,DGACU,DGBCU
    1 FORMAT(1H139X53HA E R O J E T - G E N E R A L   C O R P O R A T I
     1O N/1H039X53HC O M P U T I N G   S C I E N C E S   D I V I S I O N
     2/1H050X31HS A C R A M E N T O   P L A N T//1H057X17HP R O G R A M
     3 K/1H019X93HG E N E R A L   T W O - P H A S E   E Q U I L I B R I
     4 U M   I N   T E R N A R Y   S Y S T E M/16HOC U S T O M E R 97X
     519HP R O G R A M M E R/12HOY. A. CHANG109X11HR.H. MARLER//)
      READ(5,3)TITLE,ID
    3 FORMAT(13A6,I2)
    2 READ (5,INPUT)
      WRITE (6,1)
      WRITE(6,4)TITLE,ID
    4 FORMAT(1H026X13A6,25XI2)
      WRITE (6,INPUT)
      WRITE(6,1)
      WRITE(6,5)TITLE
      FLAG = ID/10
      RT=R*T
      CALL DLGC(DGACV,R,T,ZPP,VACV,RACV,DEACV,FLAG,$11)
      CALL DLGC(DGBCV,R,T,ZPP,VBCV,RBCV,DEACV,FLAG,$11)
      U1=XP/(1.-ZP)
      U2=YP/(1.-ZP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG1 =    (FU1+FU2)*RT + EPSP*U1*U2+DGACU*XP+DGBCU*YP
      U1=XPP/(1.-ZPP)
      U2=YPP/(1.-ZPP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG2 =   ((FU1+FU2)*RT +EPSPP*U1*U2 + DGBCV*U2 + DGACV*U1)*(1.-ZPP)
   13 IPASS=1
      DYP = YP-Y
      DYPP=YPP-Y
      DZP = ZP-Z
      DZPP=ZPP-Z
      AREA2 =DYPP*DZP - DYP*DZPP
      ZPPP= B1
    7 YPPP = 0.
      YPPPMX=1.-ZPPP
      DZPPP=ZPPP-Z
    8 DYPPP=YPPP-Y
      AREA1 =DYP*DZPPP-DYPPP*DZP
      AREA3 =DYPPP*DZPP - DYPP*DZPPP
      IF(.NOT.(AREA1.LT.0..AND.AREA2.LT.0..AND.AREA3.LT.0..OR.AREA1.GT.
     10..AND.AREA2.GT.0..AND.AREA3.GT.0.))GO TO 11
      XPPP=1.-YPPP-ZPPP
      CALL DLGC(DGACW,R,T,ZPPP,VACW,RACW,DEACW,FLAG,$11)
      CALL DLGC(DGBCW,R,T,ZPPP,VBCW,RBCW,DEBCW,FLAG,$11)
      U1=XPPP/(1.-ZPPP)
      U2=YPPP/(1.-ZPPP)
      CALL XLNX(U1,FU1,$11)
      CALL XLNX(U2,FU2,$11)
      DG3 = ((FU1+FU2 *RT+EPSPPP*U1*U2 + DGBCW*U2 + DGACW*U1)*(1.-ZPPP)
      D = (YPP  - YP)*(ZPPP- ZP) - (ZPP  - ZP) * (YPPP- YP)
```

```
      IF(D.EQ.0.)GO TO 11
      GNU1 = ((YPP-Y)*(ZPPP-Z) - (ZPP-Z)*(YPPP-Y))/D
      IF(GNU1.LE.0..OR.GNU1.GT.1.)GO TO 11
      GNU2 = ((Y-YP)*(ZPPP-ZP) - (Z-ZP)*(YPPP-YP))/D
      IF(GNU2.LE.0..OR.GNU2.GT.1.)GO TO 11
      GNU3 =1.-GNU1-GNU2
      DG = GNU1* DG1 + GNU2 *DG2 + GNU3*DG3
      IF(IPASS.EQ.0)GO TO 9
      IPASS = 0
      DGS = DG
    9 IF(DG.GT.DGS)GO TO 10
      DGS = DG
      XPPPS = XPPP
      YPPPS = YPPP
      ZPPPS = ZPPP
   10 IF(ID.NE.0)WRITE(6,14)X,Y,Z,XP,YP,ZP,XPP,YPP,ZPP,XPPP,YPPP,ZPPP,DG
   11 YPPP = YPPP + DELTA
      IF(YPPP.LE.YPPPMX)GO TO 8
      ZPPP = ZPPP + DELTA
      IF(ZPPP.LE.B2) GO TO 7
      WRITE(6,14)X,Y,Z,XP ,YP ,ZP ,XPP ,YPP ,ZPP ,
     1XPPPS,YPPPS,ZPPPS,DGS
      M = M -1
      IF(M.LT.0)GO TO 2
      IF(DGS.GE.0.)GO TO 12
      B1 =AMAX1(B1,DGS-DELTA)
      B2 =AMIN1(B2,DGS+DELTA)
   12 DELTA = DELTA/10.
      GO TO 13
   14 FORMAT(1H 12F10.7,F10.3)
    5 FORMAT(1H025X
     1          13A6//130H     X          Y          Z          XP         Y
     1P         ZP         XPP        YPP        ZPP        XPPP       YPPP
     2    ZPPP      DELTA G/)
      END
```

```
$IBLDR XLNXF                                                      12/17/65    XLNX0000
$TEXT  XLNXF                                                      12/17/65    XLNX0001
*N ((X$7V*7'*7-5 00 80 9  9        94  8 93 77 7749G405 Z4  8 93 'G '5 'XLNX0002
*N9(('*XV*7(*7-5  5 '   74+03' W=   '  7  9  5 Z   *)PO ')4'G47'4'04'74'XLNX0003
*N8V*'*7V*X(*7-74 04'7477- ' '5      7 'G 074074'74'449' '' 7' '' 7447' 'XLNX0004
*59(2'*/         ' '5 '  '                                                XLNX0005
$CDICT XLNXF                                                      12/17/65    XLNX0006
*/ ()( 0' 1'*)P 0     0 9*)P  '*)P-6 P*)-                                 XLNX0007
$DKEND XLNXF                                                      12/17/65    XLNX0008
$IBFTC CUBICF  NOLIST
      SUBROUTINE CUBIC(COEF,X,Y,*)
      DIMENSION COEF(3)
      IF(X.EQ.1.)RETURN 1
      Y = ((X*COEF(3)+COEF(2))*X+COEF(1))/(1.-X)
      RETURN
      END
```

```
$IBFTC DGACVF   FULIST
      SUBROUTINE DGACVS(DGACV,ZPP ,*)
      COMMON    CDGACV(3),CDGBCV(3),CDGACW(3),CDGBCW(3)
      CALL CUBIC(CDGACV,ZPP,DGACV,$10)
      RETURN
   10 RETURN 1
      END
$IBFTC DGBCVF   FULIST
      SUBROUTINE DGBCVS(DGBCV,ZPP ,*)
      COMMON    CDGACV(3),CDGBCV(3),CDGACW(3),CDGBCW(3)
      CALL CUBIC(CDGBCV,ZPP,DGBCV,$10)
      RETURN
   10 RETURN 1
      END
$IBFTC DGACWF   FULIST
      SUBROUTINE DGACWS(DGACW,ZPPP,*)
      COMMON    CDGACV(3),CDGBCV(3),CDGACW(3),CDGBCW(3)
      CALL CUBIC(CDGACW,ZPPP,DGACW,$10)
      RETURN
   10 RETURN 1
      END
$IBFTC DGBCWF   FULIST
      SUBROUTINE DGBCWS(DGBCW,ZPPP,*)
      COMMON    CDGACV(3),CDGBCV(3),CDGACW(3),CDGBCW(3)
      CALL CUBIC(CDGBCW,ZPPP,DGBCW,$10)
      RETURN
   10 RETURN 1
      END
```

```
$IBFTC DELGC    FULIST
      SUBROUTINE  DLGC (DGC,R,T,ARG,V,ARE,DELE,FLAG,*)
      DIMENSION V(5),ARE(5)
      GC = ARE(5)*T + ARE(4)
      DGFC =(ALOG10(T)*ARE(2)+ARE(3))*T + ARE(1)
      CALL     DGXS(ARG,DELE,R,T,DELGXS,FLAG,$2)
      DGC=((((ARG*V(5)+V(4))*ARG+V(3))*ARG+V(2))*ARG+V(1)+(1.-2.*ARG)*
     1GC+(1.-ARG)*DGFC+DELGXS)/(1.-ARG)
      IF(FLAG.GT.0.)WRITE(6,1)GC,DGFC,DGC
    1 FORMAT( 6H GC = E15.8,5X 7HDGFC = E15.8,5X6HDGC = E15.8)
      RETURN
    2 RETURN1
      END
$IBFTC DLGXS    FULIST
      SUBROUTINE DGXS(X,DELE,R,T,DELGXS,FLAG,*)
      RT=R*T
      IF(DELE)10,5,1
    1 W=DELE/RT
      EXPW = EXP(W)
      EXPNW =1./EXPW
      CALL XSUBAB(X,EXPNW,XA,XB,FLAG,$10)
      CALL PRELIM(XA,XB,EXPNW,EXPW,SMALLR,S,FLAG)
      CALL DELGCX(DELE,RT,XA,XB,SMALLR,S,DLGCXS,FLAG)
      CALL DELGMX(X,XA,XB,RT,DELE,DLGCXS,DGMEXS,FLAG)
      CONJXA=1.-XA
      CONJXB=1.-XB
      CALL XLNX(XA,FXA,$10)
      CALL XLNX(CONJXA,FCNJXA,$10)
      CALL XLNX(XB,FXB,$10)
      CALL XLNX(CONJXB,FCNJXB,$10)
      DELGXS=((FXA+FXB+FCNJXA+FCNJXB)*RT +XB*DELE)*(1.-X)/2.
      GO TO 9
    5 CONJX=1.-X
      CALL XLNX (X,FX,$10)
      CALL XLNX (CONJX,FCONJX,$10)
      UX=1.-2.*X
      CALL XLNX(UX,FUX,$10)
      DELGXS =(FUX-FCONJX+ FX)*RT
    9 IF(FLAG.GT.0.)WRITE(6,11)DELGXS
      RETURN
   10 RETURN 1
   11 FORMAT(10H DELGXS = E15.8)
      END
```

```
$EXECUTE        IBJOB
$IBJOB A48052  MAP
$IBFTC TAB      FULIST
      DIMENSION DELTAE(10),TEE(10)
    1 READ(5,100)IDE,IT,(DELTAE(I),I=1,IDE),(TEE(I),I=1,IT),XF,XL,DX,R,
     1FLAG
      WRITE(6,500)(DELTAE(I),I=1,IDE),(TEE(I),I=1,IT),XF,XL,DX,R,FLAG
      DO 50 I=1,IT
      T=TEE(I)
      RT=T*R
      DO 49 J=1,IDE
      DELE=DELTAE(J)
      WRITE (6,200)T,DELE
      W=DELE/RT
      EXPW = EXP(W)
      EXPNW = EXP(-W)
      X=XF
      IF(FLAG.GT.0.) WRITE(6,300)RT,W,EXPW,EXPNW,X
   10 CALL XSUBAB(X,EXPNW,XA,XB,FLAG,$11)
      CALL PRELIM(XA,XB,EXPNW,EXPW,SMALLR,S,FLAG)
      CALL DELGCX(DELE,RT,XA,XB,SMALLR,S,DLGCXS,FLAG)
      CALL DELGMX(X,XA,XB,RT,DELE,DLGCXS,DGMEXS,FLAG)
      DELGXS=((XA*ALOG(XA)+XB*ALOG(XB)+(1.-XA)*ALOG(1.-XA)+(1.-XB)*ALOG(
     11.-XB))*RT+XB*DELE)*(1.-X)/2.
      WRITE(6,400)X,DLGCXS,DGMEXS,DELGXS
      IF(FLAG.EQ.1.)FLAG=0.
   11 X=X+DX
      IF(X.LE.XL)GO TO 10
   49 CONTINUE
   50 CONTINUE
  100 FORMAT(2I2/(5E13.7 ))
  200 FORMAT(1H139X4HT = E15.8,10X10HDELTA E = E15.8/1H038X1HX14X12HDELT
     1A G C XS 7X13HDELTA G ME XS 9X10HDELTA G XS/)
  300 FORMAT(1H026X5E16.8)
  400 FORMAT(27X4E20.8)
  500 FORMAT(1H1/(26X5E16.8 ))
      GO TO 1
      END
```

```
$IBFTC XSBAB    FULIST                                                  A
      SUBROUTINE XSUBAB(X,EXPNW,XA,XB,FLAG,*)                           B
      ASTAR=1.-EXPNW                                                    C
      CONJX=1.-X                                                        D
      BSTAR=((1.+X)*EXPNW-3.*X+1.)/CONJX
      CSTAR=-2.*X/CONJX*EXPNW
      XB1=-BSTAR/2./ASTAR                                               G
      XB2=SQRT(BSTAR*BSTAR-4.*ASTAR*CSTAR)/2./ASTAR                     H
      XB=XB1+XB2                                                        I
      IF((XB.GT.0.).AND.(XB.LT.1.))GO TO 10                             J
      XB=XB1-XB2                                                        K
      IF((XB.GT.0.).AND.(XB.LT.1.))GO TO 10                             L
      WRITE (6,100)XB,ASTAR,BSTAR,CSTAR                                 M
  100 FORMAT(6HOXB = E15.8,38H , IS NOT IN THE RANGE (O LT XB LT 1)./   N
     1 9HOASTAR = E15.8,10X 8HBSTAR = E15.8,10X8HCSTAR = E15.8/         O
     216HOGO TO NEXT CASE)                                              P
    9 RETURN 1                                                          Q
   10 XA=2./CONJX*X-XB                                                  R
      IF((XA.GT.0.).AND.(XA.LT.1.))GO TO 12
      WRITE (6,200)XA,ASTAR,BSTAR,XB                                    T
  200 FORMAT(6HOXA = E15.8,38H , IS NOT IN THE RANGE (O LT XA LT 1)./   U
     1 9HOASTAR = E15.8,10X 8HBSTAR = E15.8,10X 8HCSTAR = E15.8,10X     A
     25HXB = E15.8/16HOGO TO NEXT CASE)                                 B
      GO TO 9                                                           C
   12 IF(FLAG.LE.0.)GO TO 14                                            D
      WRITE (6,300)XA,XB,ASTAR,BSTAR,CSTAR                              E
  300 FORMAT( 21HOD U M P  X S U B A B/6HOXA = E15.8,5X5HXB = E15.8,5X  F
     18HASTAR = E15.8,5X8HBSTAR = E15.8,5X,8HCSTAR = E15.8/)            G
   14 RETURN                                                            H
      END                                                               I
$IBFTC PRLIM    FULIST                                                  A
      SUBROUTINE PRELIM(  XA,XB,EXPNW,EXPW,SMALLR,S,FLAG)               B
      CONJXA=1.-XA                                                      C
      CONJXB=1.-XB                                                      D
      U=XA/CONJXA*EXPNW                                                 E
      V=XB/CONJXB*EXPW                                                  F
      RNUM=((1.+U)*CONJXA)**2                                           G
      SMALLR=RNUM/(RNUM+EXPNW)                                          H
      SNUM=((1.+V)*CONJXB)**2                                           I
      S=SNUM/(SNUM+EXPW)                                                J
      IF(FLAG.LE.0.)GO TO 1                                             K
      WRITE (6,100)U,V,RNUM,SMALLR,SNUM,S                               L
  100 FORMAT(21HOD U M P  P R E L I M/5HOU = E15.7,5H V = E15.7,8H RNUM M
     1= E15.7, 9HSMALLR = E15.7,8H SNUM = E15.7,5H S = E15.7/)          N
    1 RETURN                                                            O
      FND                                                               P
```

```
$IBFTC DGCXS    FULIST
      SUBROUTINE DELGCX (DELE,RT,XA,XB,SMALLR,S,DLGCXS,FLAG)
      DLGCXS=(ALOG(XA/(1.-XA))*SMALLR + ALOG(XB/(1.-XB))*S)*RT + S*DELE
      IF(FLAG.GT.0.)WRITE(6,100)DLGCXS
      RETURN
  100 FORMAT(21H0D U M P  D L G C X S/13H0 DEL GCXS = E15.7/)
      END
$IBFTC DGMX     FULIST
      SUBROUTINE DELGMX (X,XA,XB,RT,DELE,DLGCXS,DGMEXS,FLAG)
      CONJXA=1.-XA
      CONJXB=1.-XB
      ZAMX=ALOG(CONJXA)*CONJXA + ALOG(XA)*XA
      ZBMX=ALOG(CONJXB)*CONJXB + ALOG(XB)*XB
      DGMEXS=((ZAMX+ZBMX)*RT+XB*DELE)/2.-X/(1.-X)*DLGCXS
      IF(FLAG.GT.0.)WRITE (6,100)ZAMX,ZBMX,DGMEXS
      RETURN
  100 FORMAT(21H0D U M P  D E L G M X/8H0ZAMX = E15.7,10X7HZBMX = E15.7,
     110X9HDGMEXS = E15.7/)
      END
$IBFTC DGMS     FULIST
      SUBROUTINE DGMES(X,RT,ALPHA,GME,DGMESG,GC,DELGF,FLAG)
      IF(X-.5)10,20,30
   10 DGMESG =ALOG((1.-2.*X)/(1.-X))*RT + 2.*DELGF + GC
      GO TO 40
   20 DGMESG = ALOG((1.-2.*ALPHA)/ALPHA)*RT - GME
      GO TO 40
   30 DGMESG = ALOG((1.-X)/(2.*X-1.))*RT - GME
   40 IF(FLAG.GT.0.)WRITE(6,100) DGMESG
  100 FORMAT(19H0D U M P  D G M E S/10H0DGMESG = E15.8/)
      RETURN
      END
$IBFTC DGCS     FULIST
      SUBROUTINE DGCSIG(X,GC,GME,ALPHA,DELGF,RT,DLGCSG,FLAG)
      IF(X-.5)10,20,30
   10 DLGCSG = ALOG(X/(1.-2.*X))*RT -GC
      GO TO 40
   20 DLGCSG = ALOG((1.-2.*ALPHA)/ALPHA)*RT-GC
      GO TO 40
   30 DLGCSG = ALOG((2.*X-1.)/X)*RT + 2.*DELGF + GME
   40 IF(FLAG.GT.0.)WRITE(6,100)DLGCSG
  100 FORMAT(21H0D U M P  D G C S I G/10H0DLGCSG = E15.7/)
      RETURN
      END
```

APPENDIX II.

Computer program for calculating the excess free energies of the Me_2C phases in terms of 2 + w gramatom of alloy. For theoretical background of this problem, refer to AFML-TR-65-2, Part I, Volume I (1965).........(Bill Reuss).

```
$EXECUTE        IBJOB
$IBJOB 42051    MAP
$IBFTC 42051    FULIST,DECK,REF

C  PROGRAM C (42051)    ME2CY PHASE CALCULATION   -  EXCESS QUANTITIES
      DIMENSION  Y(2500), XCA(2500) , XCB(2500),  ZA(2500), ZB(2500),
     1    Z(2500), HXS(2500), GXS(2500)

   10 READ (5,20) MODEL, YMIN, YMAX, DY, WMIN, WMAX, DW
   20 FORMAT  (I1,8X,6E10.0)
      WRITE (6,30) MODEL,YMIN,YMAX,DY,WMIN,WMAX,DW
   30 FORMAT(1H1/////////////////61X,9HPROGRAM C//62X,7H(42051)///44X,
     1 44HME2CY PHASE CALCULATIONS - EXCESS QUANTITIES///62X,6HMODEL ,
     2 I1////3X,6HYMIN = F11.5  ,4X, 6HYMAX = F11.5 ,4X, 6H  DY = F11.5,
     3  4X, 6HWMIN = F11.5,4X, 6HWMAX = F11.5 ,4X, 6H  DW = F11.5 )

      W = WMIN

   35 LINCT = 50
   37 DO 39  I=1,2500
      Y(I) = 0.
      XCA(I) = 0.
      XCB(I) = 0.
      ZA(I) = 0.
      ZB(I) = 0.
      Z(I) = 0.
      HXS(I) = 0.
   39 GXS(I) = 0.

   40 I = 1
      Y(I) = YMIN
      EMW  =  1./ EXP(W)

   50 GO TO  (100,200),MODEL

  100 AA = 2.* (1.-EMW)
      AAAA = 2. * AA
  105 B =  1. - Y(I)  + EMW*(1.+Y(I))
      C =  Y(I) * EMW
      QUAD = B**2 + AAAA*C
      IF (QUAD) 107,110,110
  107 ZA(I) = 1.
      ZB(I) = QUAD
      GO TO 280

  110 QUAD =  SQRT(QUAD)
      XCB(I) = (-B+QUAD)/AA
      IF (XCB(I)-1.)120,130,130
  120 IF (XCB(I))130,130,160
  130 ZB(I) = XCB(I)
      XCB(I) = (-B-QUAD)/AA
      IF (XCB(I)) 150,155,140
  140 IF (XCB(I)-1.)160,150,150
  150 Z(I) = XCB(I)
      XCB(I) = 0.
  155 ZA(I) = 3.
      GO TO 280

  160 XCA (I) = Y(I) - XCB(I)
      ZA(I)   = XCA(I)* ALOG(XCA(I)) + (1.-XCA(I))*ALOG(1.-XCA(I))
```

```
      ZB(I)= XCB(I)*ALOG(XCB(I)) +  (1.-XCB(I))* ALOG(1.-XCB(I))
      HXS(I) = XCB(I) * W
      GO TO 270

  200 CON = -2. +4.* EMW
  205 YEMW =  Y(I) * EMW
      A =  CON + Y(I) - YEMW
      B =   2. - YEMW
      QUAD  =  B**2 + 4.*A*Y(I)
      IF (QUAD) 107,210,210

  210 QUAD = SQRT(QUAD)
      AA = 2.*A
      XCA(I) = (-B + QUAD)/AA
      IF (XCA(I)-1.)220,230,230
  220 IF (XCA(I)) 230,230,260
  230 ZB(I) = XCA(I)
      XCA(I) = (-B-QUAD)/AA
      IF (XCA(I)) 250,255,240
  240 IF (XCA(I) - 1.) 260,250,250
  250 Z(I) = XCA(I)
  255 ZA(I) = 2.
      GO TO 280

  260 XCA1 = 1. + XCA(I)
      XCB(I) = Y(I)/2.*XCA1/XCA(I) - 1.
      TO1PX = 2./XCA1
      TXO1PX  =  TO1PX * XCA(I)
      ZA(I)= TO1PX * ( XCA(I)*ALOG(XCA(I))+(1.-XCA(I))* ALOG(1.-XCA(I)))
      ZB(I)=TXO1PX * ( XCB(I)*ALOG(XCB(I))+(1.-XCB(I))* ALOG(1.-XCB(I)))
      HXS(I) = TXO1PX * XCB(I) * W

  270 Z(I)    = ZA(I) + ZB(I)
      GXS(I) =  HXS(I) + Z(I)
      IF (I - 2499)280, 300 ,300

  280 Y(I+1) = Y(I) + DY
      IF (Y(I+1) - YMAX) 290,290,298
  290 I = I+1
      GO TO (105,205),MODEL

  298 Y(I+1) = 0.
  300 DO 380 J = 1,I
  303 IF (LINCT - 42) 305,350,350
  350 LINCT = 0
      WRITE (6,360)MODEL ,W
  360 FORMAT (1H1,61X, 9HPROGRAM C //62X, 7H(42051) ///44X,44HME2CY PHAS
     1E CALCULATIONS - EXCESS QUANTITIES/// 62X , 6HMODEL ,I1///
     2   57X, 3HW = E15.8 ////)
  305 IF (XCB(J)) 330,310,330
  310 IF (ZA(J)- 1.) 320,315,320
  315 WRITE (6,316) Y(J) ,ZB(J)
  316 FORMAT(F11.4,21X,31HNEGATIVE VALUE UNDER RADICAL =  E16.8,6X ,
     1 15HIMAGINARY ROOTS   )
      GO TO 375
  320 IF (ZA(J)-2.) 325,327,325
  327 WRITE (6,328) Y(J) , ZB(J), Z(J)
  328 FORMAT( F11.4,13X, 11HXCB ROOTS (   ,E16.8 , 3H ,   ,E16.8 , 40H ) A
     1RE NOT IN THE RANGE BETWEEN 0 AND 1     )
```

```
      GO TO 375
  325 IF (ZA(J) - 2.) 330,323,330
  323 WRITE (6,324) Y(J) , ZB(J) , Z(J)
  324 FORMAT (F11.4,13X, 11HXCA ROOTS (   ,E16.8 , 3H ,   ,E16.8 , 40H ) A
     1RE NOT IN THE RANGE BETWEEN 0 AND 1      )
      GO TO 375
  330 CALL INT4D (Y,HXS,Y(J) ,TRP, DHXS )
      CALL INT4D(Y,Z ,Y(J),TRP,DZ)
      CALL INT4D(Y,GXS,Y(J),TRP,DGXS)
  370 WRITE (6,371)  Y(J) ,XCA(J) , XCB(J) ,ZA(J),ZB(J),Z(J) ,HXS(J),
     1 GXS(J), DHXS , DZ , DGXS
  371 FORMAT (1X,4HY = E15.8,8H  XCA = E15.8,8H  XCB = E15.8,7H  ZA =
     1 E15.8,7H  ZB = E15.8,6H  Z = E15.8/5X,6HHXS = E15.8,8H  GXS =
     2 E15.8,12H  DHXS/DY = E15.8,10H  DZ/DY = E15.8,12H  DGXS/DY =
     3 E15.8)
  375 LINCT = LINCT + 2
  380 CONTINUE

      W = W+DW
      IF (W-WMAX) 35,35,10

      END
$ENTRY          42051
$DATA
1        .96        1.04       .02        .68315768 .70315768 .01
2        .96        1.04       .02        .68315768 .70315768 .01
2        .05        1.95       .05        .5        9.        .5
1        .05        1.95       .05        .5        9.        .5
```

APPENDIX III.

Computer program for evaluating the high-temperature thermodynamic properties. For theoretical background, refer to AFML-TR-65-2, Part IV, Volume I (1965)..... (Len Nole).

```
$IBJOB          GO,MAP
$IBFTC 8064     LIST,REF,DECK
C      EVALUATION OF HIGH TEMPERATURE THERMODYNAMIC PROPERTIES
C      VIA NUMERICAL INTEGRATION
       DIMENSION DATA(500),HEAD(12),TABLT(100),KDIS(20),TABLY(100),
      1          TABT(1),TABY(1)
       EQUIVALENCE (TABT,DATA(1)),(TABY,DATA(201)),(SST,DATA(401))
C
C
     1 DO 2 I =1,500
     2 DATA(I)= 0.
       CALL SLITE(0)
       CALL AS138 (DATA,HEAD,NE)
       IF (NE - 2)6,3,4
     3 STOP
     4 WRITE (6,5)
     5 FORMAT (27H (((I N P U T  E R R O R))))
       GO TO 1
     6 N = DATA(402)
       DO 10 I =1,200
       IF (TABT(I))4,7,10
     7 IF (TABT(I+1))4,11,10
    10 CONTINUE
    11 NOFT = I-1
       WRITE (6,12) HEAD
    12 FORMAT (1H1,42X,27HAEROJET GENERAL CORPORATION/ 36X,41HLIQUID ROCK
      1ET PLANT,SACRAMENTO CALIFORNIA / 31X,55HEVALUATION OF HIGH TEMPERA
      2TURE THERMODYNAMIC PROPERTIES/11H DEPT. 4600,80X,8HJOB-8064/32X,
      312A6)
       WRITE (6,13) SST, N, (TABT(I),TABY(I),TABT(I+50),TABY(I+50),
      1TABT(I+100),TABY(I+100),TABT(I+150),TABY(I+150), I = 1,50)
    13 FORMAT (//,50X,20H I N P U T  D A T A  //7X,3HSST,3X,7HSUB INT/,
      1F10.2,8X,I2,//,23X,2HT ,8X,2HY ,10X,10H      T(50),10H      Y(50),10
      2X,10H     T(100),10H     Y(100),10X,10H     T(150),10H     Y(150),//,
      3(15X,F10.2,F10.3,10X,F10.2,F10.3,10X,F10.2,F10.3,10X,F10.2,F10.3))
       WRITE (6,130)
   130 FORMAT (1H1,48X,21H O U T P U T  D A T A  //,10H  T DEG K ,5X,
      15H  Y  ,5X,5H  CP ,4X,6HHT-HST,4X,6HST-SST,5X,5HFOGFE,5X,10H INTER
      2VALS)
       KDIS(1)= 1
       I = 2
       K = 2
       TINTP = 0.
       TABLT(1) = TABT(1)
       TABLY(1) = TABY(1)
       J = 2
    14 TABLT(I) = TABT(J)
       TABLY(I) = TABY(J)
       IF (TABT(J+1))20,16,20
    16 IF (TABT(J+2))4,22,160
   160 J = J+2
       I = I+1
       KDIS(K) = J
       K = K + 1
       TABLT(I) = TABT(J)
       TABLY(I) = TABY(J)
    20 I = I + 1
       J=J+4
       GO TO 14
```

```
   22 KDIS(K) = NOFT
      K = 0
      LMN = 1
   24 K = K+1
      NML = KDIS(K)
      LN = KDIS(K+1)
      IF (LMN -1)231,231,25
  231 DELHT = 0.
      DELST = 0.
      CP = TABLY(1)
      FOGFE = DATA(401)
      GO TO 27
   25 LMN = LMN+1
      DELHT = TABLY(LMN)*(TABLT(LMN) - 298.15)
      CALL INT4D (TABT(NML),TABY(NML),TABLT(LMN),YO,DY)
      CP = DY*(TABLT(LMN) - 298.15) + TABLY(LMN)
      TEMP = TABLY(LMN)*(1. - 298.15/TABLT(LMN))
      CALL SLITET(1,KL)
      IF (KL-1)26,26,251
  251 CALL INTGR (TABLT(LMN-1),TABLT(LMN),TX,N)
      CALL INT4 (TABT(NML),TABY(NML),TX,YOFT)
      GOFTX = YOFT * (TX - 298.15)/TX/TX
      CALL INTGS (GOFTX,TINT,.00001,M)
      TINT = TINT + TINTP
      TINTP = TINT
   26 DELST = TEMP + TINT
      FOGFE = -DELHT/TABLT(LMN)+ DELST + SST
   27 WRITE (6,30) TABLT(LMN), TABLY(LMN), CP, DELHT, DELST, FOGFE, M
   30 FORMAT    (F10.2,F10.3,4F10.2,12X,I3)
   31 IF (TABLT(LMN))4,1,32
   32 IF (TABLT(LMN) - TABT(LN))25,34,34
   34 IF (TABT(LN) - TABT(NOFT))35,1,1
   35 CALL SLITE(1)
      GO TO 24
      END
$ENTRY          8064
$DATA
```

APPENDIX IV.

Computer program for calculating the free energies of the monocarbide phases using the Wagner-Schottky vacancy model. For theoretical background, refer to AFML-TR-65-2, Part IV, Volume I (1965)....................(Jerry Howard)

```
$EXECUTE        IBJOB
$IBJOB 8072     MAP
$IBFTC 8072     LIST,DECK
C
C                      MONOCARBIDES BY USING THE VACANCY MODEL.
C8072- PROGRAM H. THERMODYNAMIC CALCULATIONS FOR GROUP IV METAL
      DIMENSION DATA(200),HEAD(12),TEMP1(20),GAO(20),GBO(20),
     1          ALGPAO(20),ALGPBO(20),GA(20),GB(20),RTLN(20),
     2          ALFA(20),TDGF(20)
      DIMENSION A(20),B(20),BET(20),GAM(20)
      EQUIVALENCE (DATA(1),TA),(DATA(2),TAB),(DATA(3),TEL),(DATA(4),
     1          BETA),(DATA(5),GAMMA),(DATA(6),BETAP),(DATA(7),GAMMAP),
     2          (DATA(8),BETAPP),(DATA(9),GAMMPP),(DATA(11
     3         ),XMIN),(DATA(12),XMAX),(DATA(13),DELX),
     4          (DATA(15),TEMP1),(DATA(55),GAO),
     5          (DATA(75),GBO),(DATA(95),ALGPAO),(DATA(115),ALGPBO),
     6          (DATA(155),T1CONT),(DATA(35),AA),(DATA(36),BB),(DATA(37)
     7         ,AAP),(DATA(38),BBP),(DATA(39),AAPP),(DATA(40),BBPP)
C
    1 CALL AS138(DATA,HEAD,NE)
      WRITE(6,88)
   88 FORMAT(1H1)
      GO TO (5,4,2),NE
    2 WRITE(6,3)
    3 FORMAT(20X,24H I N P U T   E R R O R  )
      GO TO 1
    4 CALL EXIT
    5 PT5=.005
      ALN2=.69315
      R=1.98726
      I=1
      ICNT1=T1CONT+ .01
C
    6 T = TEMP1(I)
      IF(T.LT.TAB) GO TO 7
      IF(T.LT.TEL) GO TO 8
      BET(I)=BETAPP
      GAM(I)=GAMMPP
      A(I) = AAPP
      B(I) = BBPP
      GO TO 9
    7 BET(I)=BETA
      GAM(I)=GAMMA
      A(I) = AA
      B(I) = BB
      GO TO 9
```

```
    8 BET(I)=BETAP
      GAM(I)=GAMMAP
      A(I) = AAP
      B(I) = BBP
    9 RT=R*T
      TDGF(I)=BET(I)+GAM(I)*T
      GB(I)=A(I)+B(I)*T
      RTLN(I)=-GB(I)-RT*ALN2
      ALFA(I)=.5*EXP(-GB(I)/RT)
      GA(I)=-RTLN(I)-RT*ALN2-TDGF(I)
C
      I=I+1
      IF(I.LE.ICNT1) GO TO 6
C
   20 J=1
C
   21 T=TEMP1(J)
      CGA = GA(J)
      CGB = GB(J)
      CRLNA = RTLN(J)
      CALF = ALFA(J)
      RT = R*T
      WRITE(6,22) HEAD
   22 FORMAT(1H1,42X,27HAEROJET GENERAL CORPORATION/36X,41HLIQUID ROCKET
     1 PLANT,SACRAMENTO CALIFORNIA//38X,12A6//19H I N P U T   D A T A//)
      WRITE(6,100) BET(J),GAM(J), A(J), B(J)
  100 FORMAT(9X,16HLINEAR EQUATIONS/12X,18H2 DELTA GF     A = ,E12.5,
     16H, B = ,E12.5/12X,18HGB+              A = ,E12.5,6H, B = ,E12.5)
      WRITE(6,101)GAO(J),GBO(J),ALGPAO(J),ALGPBO(J)
  101 FORMAT(9X,6HGAO = ,F7.0,8H, GBO = ,F7.0,12H, LOG PAO = ,F7.3,
     112H, LOG PBO = ,F7.3)
   28 WRITE(6,29) T
   29 FORMAT(9X,15HTEMPERATURE IS ,F6.1,15H DEGREES KELVIN,/)
      WRITE(6,102) CGA,CGB,CRLNA,CALF
  102 FORMAT(21H O U T P U T   D A T A//9X,5HGA+= ,E12.5,6H,GB+= ,E12.5,
     113H,RT LN ALFA= ,E12.5,7H,ALFA= ,E12.5//8X,1HX,11X,4HDGAB,10X,
     24HDGBB,11X,2HDG,12X,3HGAB,11X,3HGBB,12X,1HG,10X,6HLOG PA,8X,
     36HLOG PB)
      C1=4.576*T
      C2=C1*ALGPAO(J)
      C3=C1*ALGPBO(J)
      X=XMIN
C
   30 ARG = ABS(X/.5-1.0)
      IF(ARG.LE.1.E-4) GO TO 32
      IF(X.LT.0.5) GO TO 34
```

```
      IF(X.GT.0.5) GO TO 35
   32 X = .500
      DGA = -CGA-RT*ALN2-CRLNA
      DGB = -CGB+CGA+DGA
      GO TO 36
   34 DGA = TDGF(J)+CGB+RT*ALOG((1.-2.*X)/(1.-X))
      DGB = -CGB+RT*ALOG(X/(1.-2.*X))
      GO TO 36
   35 DGA = -CGA+RT*ALOG((1.-X)/(2.*X-1.))
      DGB = TDGF(J)+CGA+RT*ALOG((2.*X-1.)/X)
   36 VR1 = 1. -X
      DG=VR1*DGA+DGB*X
      GAB=DGA+GAO(J)
      GBB=DGB+GBO(J)
      G=VR1*GAB+X*GBB
      ALGPA=(DGA+C2)/C1
      ALGPB=(DGB+C3)/C1
      WRITE(6,33) X,DGA,DGB,DG,GAB,GBB,G,ALGPA,ALGPB
   33 FORMAT(9(2X,F12.5))
      X=X+DELX
      IF(X.LE.XMAX) GO TO 30
      J=J+1
      IF(J.LE.ICNT1) GO TO 21
      GO TO 1
      END
```

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author): Aerojet-General Corporation, Materials Research Laboratory, Sacramento, California	2a. REPORT SECURITY CLASSIFICATION: Unclassified
	2b. GROUP: N.A.

3. REPORT TITLE

(6) Ternary Phase Equilibria in Transition Metal-Boron-Carbon-Silicon Systems. Part IV. Thermochemical Calculations, Volume III, Computational Approach to to the Calculation of Ternary Phase Diagrams.

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

(9) Technical ( )

5. AUTHOR(S) (Last name, first name, initial)

(10) Y. A. Chang [redacted]

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
(11) February 1966 (15)	(12) 69 p.	10

8a. CONTRACT OR GRANT NO. AF 33(615)-1249	9a. ORIGINATOR'S REPORT NUMBER(S) (19) (18) AFML-TR-65-2 - Pt-4- Vol-3 ~~Part IV, Volume III~~
b. ~~PROJECT NO.~~ (16) AF-7350	
c. ~~Task No.~~ (17) 735001	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	N.A.

10. AVAILABILITY/LIMITATION NOTICES

This document is subject to special export controls, and each transmittal to foreign governments for foreign nationals may be made only with prior approval of Metals & Ceramics Div., AF Materials Laboratory, Wright-Patterson AFB, Ohio.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	AFML (MAMC) Wright-Patterson AFB, Ohio

13. ABSTRACT

The general conditional equations which govern the phase equilibria in three-component systems are presented. Using the general conditional equations, a general method has been developed to precalculate the phase equilibria in three-component systems from first principle using computer technique. The method developed has been applied to several model examples and the system Ta-Hf-C. The phase equilibria in three-component systems calculated using the simplified method as originally developed by Rudy, agree well with those calculated by the present method. The only difference is in the homogeneous range with respect to the interstitial component of solid solutions which exhibit large variation with metal exchange. This is to be expected in view of the assumptions made in the simplified method.

In connection with the phase diagram calculation and other problems of the present Air Force contract, several computer programs have been developed which are included in the appendix of this report.

DD FORM 1 JAN 64 1473

Unclassified

Security Classification

14. KEY WORDS	LINK A ROLE	LINK A WT	LINK B ROLE	LINK B WT	LINK C ROLE	LINK C WT
Thermodynamics Ternary Phase Diagram						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (*corporate author*) issuing the report.

2a. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2b. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7a. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7b. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8a. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8b, 8c, & 8d. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9a. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9b. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers (*either by the originator or by the sponsor*), also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
____."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
____."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
____."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring (*paying for*) the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as *(TS), (S), (C),* or *(U)*.

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, rules, and weights is optional.