JAEA-Data/Code
2010-034

Kiche: A Simulation Tool for Kinetics of Iodine Chemistry in the Containment of Light Water Reactors under Severe Accident Conditions

(Contract Research)

Kiyofumi MORIYAMA, Yu MARUYAMA and Hideo NAKAMURA

Reactor Safety Research Unit
Nuclear Safety Research Center

March 2011

Japan Atomic Energy Agency

日本原子力研究開発機構

本レポートは独立行政法人日本原子力研究開発機構が不定期に発行する成果報告書です。
本レポートの入手並びに著作権利用に関するお問い合わせは、下記あてにお問い合わせ下さい。
なお、本レポートの全文は日本原子力研究開発機構ホームページ（http://www.jaea.go.jp）
より発信されています。

独立行政法人日本原子力研究開発機構　研究技術情報部　研究技術情報課
〒319-1195　茨城県那珂郡東海村白方白根 2 番地 4
電話 029-282-6387, Fax 029-282-5920, E-mail:ird-support@jaea.go.jp

This report is issued irregularly by Japan Atomic Energy Agency
Inquiries about availability and/or copyright of this report should be addressed to
Intellectual Resources Section, Intellectual Resources Department,
Japan Atomic Energy Agency
2-4 Shirakata Shirane, Tokai-mura, Naka-gun, Ibaraki-ken 319-1195 Japan
Tel +81-29-282-6387, Fax +81-29-282-5920, E-mail:ird-support@jaea.go.jp

JAEA-Data/Code 2010-034

Kiche: A Simulation Tool for Kinetics of Iodine Chemistry in the Containment of Light Water Reactors under Severe Accident Conditions (Contract Research)

Kiyofumi MORIYAMA, Yu MARUYAMA and Hideo NAKAMURA

Reactor Safety Research Unit
Nuclear Safety Research Center
Japan Atomic Energy Agency
Tokai-mura, Naka-gun, Ibaraki-ken

(Received December 17, 2010)

An iodine chemistry simulation tool, Kiche, was developed for analyses of chemical kinetics relevant to iodine volatilization in the containment vessel of light water reactors (LWRs) during a severe accident. It consists of a Fortran code to solve chemical kinetics models, reaction databases written in plain text format, and peripheral tools to convert the reaction databases into Fortran codes to solve corresponding ordinary differential equation sets. Potential advantages of Kiche are the text format reaction database separated from the code that provides flexibility of the chemistry model, and, being a Fortran code which is relatively easily coupled with other Fortran codes such as severe accident analysis codes. This document describes the model, solution method, code structure, and examples of application of Kiche for simulation of experiments. The calculation results by the present model agreed well with the experimental data and it indicates the model properly includes the most important processes in the volatilization of iodine from irradiated iodide solutions with or without organic impurities. The appendixes give practical information for the usage of Kiche.

Keywords: Severe Accident, Iodine Chemistry, Containment Vessel, Source Term, Radiation Chemistry, Numerical Simulation

This work includes a part of results of a contract research funded by Japan Nuclear Energy Safety Organization (JNES) in fiscal years 2007–2010.

Kiche: 軽水炉シビアアクシデント条件下の格納容器内ヨウ素化学解析ツール (受託研究)

日本原子力研究開発機構 安全研究センター
原子炉安全研究ユニット
森山 清史・丸山 結・中村 秀夫

(2010 年 12 月 17 日 受理)

軽水炉シビアアクシデント時の格納容器内におけるガス状ヨウ素放出に関する反応速度論シミュレーションを行うため, ヨウ素化学解析ツール Kiche を開発した. これは, 反応速度論モデルを解くための Fortran コード, テキスト形式の反応データベース, 及び, 付属するツール類から成るシステムである. これには反応データベースから対応する常微分方程式系を解く Fortran コードへの変換を行うツールを含む. 本システムは, このように反応データベースをコードと分離してテキスト形式で持つため化学モデルが柔軟である, Fortran コードであるためシビアアクシデント解析コード等, 他の Fortran コードとの連成が比較的容易である, などの長所を備える. 本報には, Kiche のモデル, 解法, コードの構成, また, 実験シミュレーションへの適用例を記す. 実験のシミュレーション結果は実験データと良好な一致を示し, 本モデルが照射下のヨウ化物水溶液からのガス状ヨウ素放出について, 有機不純物が存在する場合も含め, 主要な要因を適切に含んでいることが示された. 付録には, Kiche を使用するために必要な実用上の情報ををまとめる.

本件は (独) 原子力安全基盤機構からの受託研究 (平成 19 年度 ∼ 平成 22 年度) の成果を含む.
原子力科学研究所 (駐在) : 〒 319-1195 茨城県那珂郡東海村白方白根 2-4

Contents

目 次

1 Introduction

Iodine is one of the fission products (FPs) produced in the core of light water reactors (LWRs) and is the one recognized as the most relevant to the public health consequences in case of a severe accident with environmental release of FPs, due to its large fission yield, the half life of 8 days (for ^{131}I) and complicated chemical behavior that may produce volatile species. Thus, large number of research works have been done on the iodine chemistry under the conditions relevant to the reactor cooling system (RCS) and the containment vessel (CV) of LWRs. Especially, after the core degradation accident at TMI-2 plant, US, in 1979, comprehensive studies focused on more realistic evaluation of the iodine source term have been performed internationally.[1, 2, 3]

Figure 1.1 schematically illustrates the iodine behavior in the containment vessels of boiling water reactors (BWRs) and pressurized water reactors (PWRs). In the RCS, iodine speciation is primarily determined by thermodynamics and CsI vapor or aerosol is the dominant form of iodine. Typically, Beahm et al.[4] concluded that the iodine at the release into the CV is mostly in CsI and less than 5% elemental iodine and HI. It is likely that iodine is mostly transferred into the CV as CsI gas or aerosols, and sediments on to the floor gravitationally or by the spray, then dissolves into water as iodide ion. It is known that iodide ion is converted into volatile elemental iodine or organic iodines under the effect of radiation. Due to the low temperature in the CV, the speciation does not follow the thermodynamics but is dominated by kinetics.

Influences of various factors on the radiation chemical reaction of iodine was investigated, such as pH, initial iodide concentration, dose rate, temperature, dissolved gases and impurities.[5, 6, 7, 8, 9, 10]

Also, efforts have been made in modeling mechanistically such radiation-chemical behaviors of iodine, and some computer codes or reaction databases have been developed for that purpose, *e.g.* LIRIC (AECL/Canada)[11], INSPECT (AEA/UK)[12]. Those tools include fundamental reactions of relatively large number such as water radiolysis and iodine reactions with radicals and so on, and they are called “mechanistic” models. Other kind of tools called “semi-empirical” models also exist. They have relatively simple reaction models and are tuned to reproduce experimental results[13, 14].

On the other hand, in the source term assessment by synthesized severe accident simulation codes, such as MELCOR[15], MAAP[16], THALES2[17], the chemical behavior of iodine in the containment has mostly been neglected or considered by simple assumptions. It might be partially due to the history that those codes were first developed in 1980’s when the knowledge on such iodine chemistry was limited and still involved large uncertainty. Some of the codes, at present, include iodine chemistry models[15, 18], and they seem to be at experimental or developmental stages where further verification and consideration of practical usages are underway.

We developed a chemical kinetics simulation tool “Kiche” (Kinetics of Iodine Chemistry) to simulate the iodine chemistry in the containment vessel under severe accident conditions. It is also capable as a tool for examination and development of mechanistic models for various chemical kinetics, and as a program module to be coupled with other Fortran codes such as severe accident codes. Kiche consists of a Fortran code to numerically solve chemical kinetics models and peripheral tools to handle reaction databases. At the start of the development of Kiche, we referred the methods and reaction models in IODIC3M code developed by Institute of Nuclear Safety System, Inc. (INSS)[19] and LIRIC 3.2 model by AECL/Canada[11].

This document describes the Kiche tool: the modeling and numerical solution method in Chapter 2, the system and code structure in Chapter 3, and verification/example calculations in Chapter 4. Appendixes provide practical information on the reaction database (Appendix A),

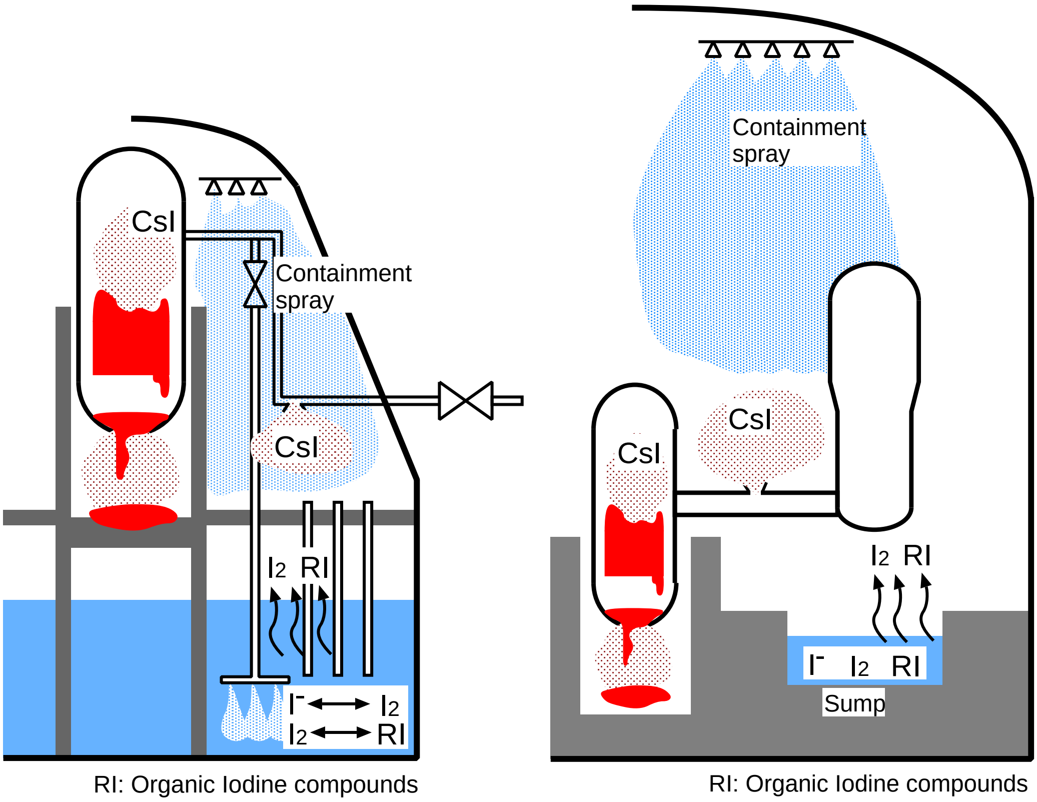

CsI
Containment
spray
CsI
I_2 RI
$I^- \longleftrightarrow I_2$
$I_2 \longleftrightarrow$ RI
RI: Organic Iodine compounds
Containment
spray
CsI
CsI
I_2 RI
I^- I_2 RI
Sump
RI: Organic Iodine compounds

Figure 1.1: Iodine behavior in the containment vessels of light water reactors. (BWRs (left), PWRs (right))

usage of the code (Appendix B) and input/output data (Appendix C).

2 Model and Solution Scheme

2.1 Reaction Model

The concept of the iodine behavior modeling by Kiche is illustrated in Fig. 2.1. Aqueous reactions, the gas-aqueous mass transfer of volatile species and wall adsorption of I_2 are involved.

At present, chemical reactions are assumed only in the aqueous phase, because it is the primary reaction field to produce volatile iodine due to the high concentration of materials and water radiolysis effect[3, 2]. As volatile iodine species, I_2 and organic iodides are considered. The organic iodides are divided into two groups, high volatility organic iodides (HVRI) and low volatiliry organic iodides (LVRI), following the method of LIRIC 3.2 model developed at AECL/Canada[11]. Alkyl iodides, typically CH_3I, are assumed for HVRI; other more hydrophilic larger molecules are assumed for LVRI.

The latest reaction set of ours as of October 2010, named “10a”, is summarized in Table 2.1–2.6.

The primary purpose of the present model is the analysis on systems with dilute aqueous solutions like the water pool in a containment vessel where molar fractions of solutes are much less than that of solvent, water. Thus, the rate constants are based on molar concentrations of the chemical species. If concentrated solutions are to be handled, more rigorous activity based reaction modeling and data should be required.

In the Kiche tool, the reaction set is given in a plain text file written in chemical reaction formula and expressions for rate constants (see Appendix A).

The reaction set is categorized in five kinds.

Table 2.1 is the water radiolysis including the generation of primary products and reactions among them[20, 21]. Radical species produced by them trigger many reactions specific to irradiated solutions.

Table 2.2 is the inorganic iodine reactions that provides the primary reaction paths to produce volatile iodine in iodide solutions under irradiation[11, 12, 9]. Oxidation/reduction of iodine species by reactions with water radiolysis products, hydrolysis of elemental iodine, and other reactions are included. The reaction of iodine atom and dissolved hydrogen (index: ih2a), and the volatility of iodine atom (i_21, i_21b) can be included optionally (activated by setting corresponding options in the input).

Table 2.3 is the organic iodine production process including radiolytic decomposition of organic impurity and reaction of organic radicals with iodine species[11, 22, 23]. The present set was developed based on the organic reaction model of LIRIC 3.2[11] and studies on the radiolytic degradation of methylethylketone (MEK) at AECL/Canada[24, 22], by putting emphasis on applicability for both air and low oxygen concentration conditions, corresponding to PWR and BWR containment atmosphere, respectively.

Table 2.4 includes miscellaneous reactions such as the catalytic consumption of $\cdot O_2^-$ by transition metal ions represented by Fe^{2+}/Fe^{3+}, acid/base equilibria, influences of chlorine. The influence of the transition metal ion reactions is important because it may significantly enhance the iodine volatility at high pHs.

Table 2.5 shows the processes other than chemical reactions, such as interface mass transfer, adsorption on walls and leaching of organic solvents from painted walls.

Table 2.6 shows the definition of some variables involved in the rate constant expressions in those tables.

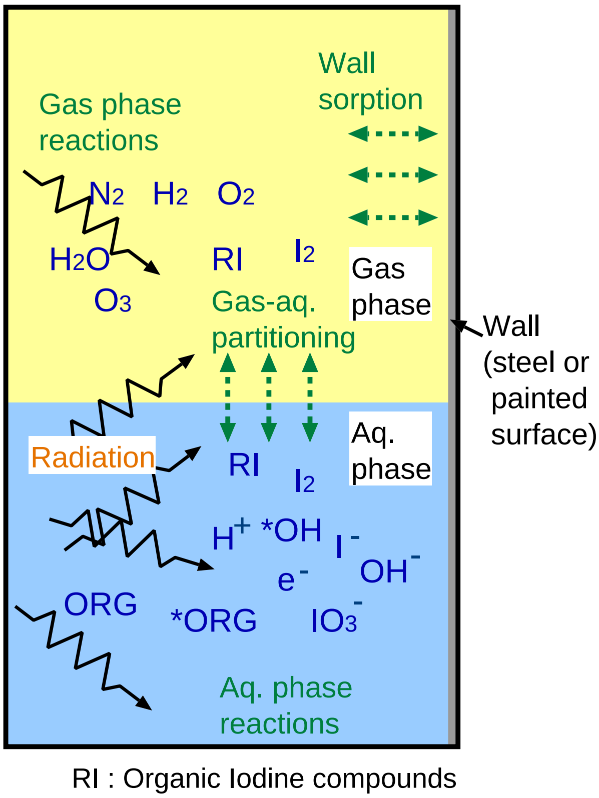


Figure 2.1: Concept of the iodine behavior model.

Table 2.1: Reaction set 10a : Reactions / Water radiolysis.

Index	Reactant	Product	Rate constant*
ge_m	$\{H_2O\}$	e^-	rad*(2.57+3.31e−3*tce)
gh	$\{H_2O\}$	$\cdot H$	rad*(0.57+1.13e−3*tce)
gh2	$\{H_2O\}$	H_2	rad*(0.43+0.75e−3*tce)
goh	$\{H_2O\}$	$\cdot OH$	rad*(2.65+7.19e−3*tce)
gh2o2	$\{H_2O\}$	H_2O_2	rad*(0.675−0.625e−3*tce)
gh_p	$\{H_2O\}$	H^+	K(ge_m)
e02	$2e^- + 2\{H_2O\}$	$2OH^- + H_2$	6.468e9*exp(2.030e4*tf)
e03	$e^- + \cdot H + \{H_2O\}$	$OH^- + H_2$	2.651e10*exp(1.400e4*tf)
e04	$e^- + \cdot OH$	OH^-	3.027e10*exp(7.920e3*tf)
e05	$e^- + H_2O_2$	$\cdot OH + OH^-$	1.413e10*exp(1.536e4*tf)
e06	$e^- + O_2$	$\cdot O_2^-$	1.791e10*exp(1.417e4*tf)
e07	$e^- + \cdot O_2^- + 2\{H_2O\}$	$2OH^- + H_2O_2$	1.284e10*exp(1.360e4*tf)
e08	$e^- + \cdot HO_2$	HO_2^-	1.284e10*exp(1.360e4*tf)
e09	$2 \cdot H$	H_2	5.447e9*exp(1.505e4*tf)
e10	$\cdot H + \cdot OH$	$\{H_2O\}$	1.528e10*exp(7.770e3*tf)
e11	$\cdot H + H_2O_2$	$\cdot OH + \{H_2O\}$	5.173e7*exp(1.594e4*tf)
e12	$\cdot H + O_2$	$\cdot HO_2$	1.324e10*exp(1.061e4*tf)
e13	$\cdot H + \cdot HO_2$	H_2O_2	1.000e10*exp(1.060e4*tf)
e14	$\cdot H + \cdot O_2^-$	HO_2^-	1.000e10*exp(1.060e4*tf)
e15	$2 \cdot OH$	H_2O_2	4.750e9*exp(7.650e3*tf)

$*$ Written by Fortran expressions; K(X)=The rate constant of reaction X; Variable definitions are given in Table 2.6; Species in { } are eliminated in the definition of rate constants.

Table 2.1: Reaction set 10a : Reactions / Water radiolysis.(continued)

Index	Reactant	Product	Rate constant*
e16	$\cdot OH + H_2$	$\cdot H + \{H_2O\}$	4.170e7*exp(1.815e4*tf)
e17	$\cdot OH + H_2O_2$	$\cdot HO_2 + \{H_2O\}$	2.879e7*exp(1.562e4*tf)
e18	$\cdot OH + \cdot HO_2$	$O_2 + \{H_2O\}$	1.077e10 *exp(5.620e3 *tf)
e19	$\cdot OH + \cdot O_2^-$	$O_2 + OH^-$	1.099e10 *exp(1.085e4 *tf)
e20	$2 \cdot HO_2$	$H_2O_2 + O_2$	6.667e5 *exp(1.910e4 *tf)
e21	$\cdot HO_2 + \cdot O_2^- + \{H_2O\}$	$O_2 + H_2O_2 + OH^-$	7.597e7 *exp(8.600e3 *tf)
e23b	$H^+ + OH^-$	$\{H_2O\}$	6.62e10+1.48e9*tce+1.28e7*tce**2 −6.03e4*tce**3+1.28e2*tce**4
e23	$\{H_2O\}$	$H^+ + OH^-$	K(e23b)*kkw
e30b	$H^+ + \cdot O_2^-$	$\cdot HO_2$	3.41e10+2.75e8*tce+1.24e7*tce**2 −6.23e4*tce**3+131.*tce**4
e30	$\cdot HO_2$	$H^+ + \cdot O_2^-$	K(e30b)*kkho2
e24b	$H^+ + HO_2^-$	H_2O_2	K(e30b)
e24	H_2O_2	$H^+ + HO_2^-$	K(e24b)*kkh2o2
e29	$\cdot OH + OH^-$	$\cdot O^- + \{H_2O\}$	7.22e9+1.62e8*tce+2.40e6*tce**2 −7.81e3*tce**3+10.6*tce**4
e29b	$\cdot O^- + \{H_2O\}$	$\cdot OH + OH^-$	K(e29)/kk29
e25	$H_2O_2 + OH^-$	$HO_2^- + \{H_2O\}$	K(e29)
e25b	$HO_2^- + \{H_2O\}$	$H_2O_2 + OH^-$	K(e25)/kk25
e26b	$e^- + H^+$	$\cdot H$	10.**(18.61−6.94e3/t+2.12e6/t**2 −2.34e8/t**3)
e26	$\cdot H$	$e^- + H^+$	K(e26b)*kkh
e27b	$\cdot H + OH^-$	$e^- + \{H_2O\}$	2.510e7 *exp(3.838e4 *tf)
e27	$e^- + \{H_2O\}$	$\cdot H + OH^-$	K(e27b)*kk27
e28b	$H^+ + \cdot O^-$	$\cdot OH$	K(e30b)
e28	$\cdot OH$	$H^+ + \cdot O^-$	K(e28b)*kkoh
e31	$\cdot HO_2 + OH^-$	$\cdot O_2^- + \{H_2O\}$	K(e29)
e31b	$\cdot O_2^- + \{H_2O\}$	$\cdot HO_2 + OH^-$	K(e31)/kk31
e32	$H_2 + \cdot O^-$	$\cdot H + OH^-$	1.211e8 *exp(1.380e4 *tf)
e33	$H_2O_2 + \cdot O^-$	$\cdot O_2^- + \{H_2O\}$	5.546e8 *exp(1.560e4 *tf)
e34	$\cdot OH + HO_2^-$	$\cdot O_2^- + \{H_2O\}$	8.319e9 *exp(1.560e4 *tf)
e35	$\cdot O^- + \cdot OH$	HO_2^-	7.610e9 *exp(7.700e3 *tf)
e36	$e^- + HO_2^-$	$\cdot O^- + OH^-$	3.507e9 *exp(1.540e4 *tf)
e37	$e^- + \cdot O^- + \{H_2O\}$	$2OH^-$	2.313e10*exp(7.900e3 *tf)
e38	$\cdot O^- + O_2$	$\cdot O_3^-$	3.709e9 *exp(1.120e4 *tf)
e38b	$\cdot O_3^-$	$\cdot O^- + O_2$	2.708e3 *exp(4.570e4 *tf)

$*$ Written by Fortran expressions; K(X)=The rate constant of reaction X; Variable definitions are given in Table 2.6; Species in { } are eliminated in the definition of rate constants.

Table 2.2: Reaction set 10a : Reactions / Iodine–Inorganic.

Index	Reactant	Product	Rate constant*
i_01	$I^- + \cdot OH$	$\cdot I + OH^-$	7.7e9*exp(19e3*tf)
l2_14f	$\cdot I + e^-$	I^-	2.4e10*ea
l2_15f	$I_2 + e^-$	$\cdot I_2^-$	5.1e10*ea
l2_17f	$\cdot I_2^- + e^-$	$2I^-$	1.3e10*ea
l2_20f	$\cdot I + \cdot H$	$H^+ + I^-$	2.7e10*ea
l2_21f	$I_2 + \cdot H$	$\cdot I_2^- + H^+$	3.5e10*ea
l2_22f	$\cdot I_2^- + \cdot H$	$2I^- + H^+$	1.8e7*exp(22200.0*tf)
l2_58f	$\cdot I + I^-$	$\cdot I_2^-$	1.2e10*exp(18.8e3*tf)
l2_58b	$\cdot I_2^-$	$\cdot I + I^-$	7.4e4*exp(40.4e3*tf)
l2_62f	$2 \cdot I_2^-$	$I_3^- + I^-$	4.5e9*ea
l2_12f	$I_2 + I^-$	I_3^-	1.00E+10
l2_12b	I_3^-	$I_2 + I^-$	K(l2_12f)/kk12
l2_63f	$\cdot I_2^- + \cdot I$	I_3^-	5.0e9*ea
l2_64f	$2 \cdot I$	I_2	1.0e10*ea
l2_27f	$I_2 + \cdot O_2^-$	$\cdot I_2^- + O_2$	3.9e9*exp(6.7e3*tf)
l2_28f	$HOI + \cdot O_2^-$	$OH^- + \cdot I + O_2$	1.0e6*ea
l2_30f	$\cdot I_2^- + \cdot O_2^-$	$O_2 + 2I^-$	3.0e9*ea
l2_40f	$HOI + \cdot OH$	$\cdot IO + \{H_2O\}$	7.0e9*ea
l2_41f	$\cdot IO + \cdot OH$	HIO_2	1.0e10*ea
l2_01f	$I_2 + OH^-$	I_2OH^-	1e10*exp(12.5e3*tf)
l2_01b	I_2OH^-	$I_2 + OH^-$	K(l2_01f)/(kk1/(kkw*kk2))
l2_02f	I_2OH^-	$HOI + I^-$	1.34E+06
l2_02b	$HOI + I^-$	I_2OH^-	K(l2_02f)/kk2
l2_13f	HOI	$IO^- + H^+$	0.14*exp(1.97e4*tf)
l2_13b	$IO^- + H^+$	HOI	5e9*exp(1.47e4*tf)
l2_54f	$I_2OH^- + H_2O_2$	$HIO_2 + I^- + \{H_2O\}$	2.25e6*exp(39000.0*tf)
l2_54b	$HIO_2 + I^- + \{H_2O\}$	$I_2OH^- + H_2O_2$	1e7*exp(101000.0*tf)
l2_55f	$HIO_2 + OH^-$	$I^- + O_2 + \{H_2O\}$	2e9*exp(50e3*tf)
ih2a	$H_2 + \cdot I$	$H^+ + \cdot H + I^-$	fuseih2*2d6*exp(1.5d4*tf)
i_21	$2 \cdot Ig$	I_2g	fusexig*1e8
i_21b	I_2g	$2 \cdot Ig$	fusexig*1e8/exp(49.0)

$*$ Written by Fortran expressions; K(X)=The rate constant of reaction X; Variable definitions are given in Table 2.6; Species in { } are eliminated in the definition of rate constants.

Table 2.3: Reaction set 10a : Reactions / Iodine–Organic.

Index	Reactant	Product	Rate constant*
org01	ORG + ·OH	·ORG	7.3d7*exp(12.2d3*tf)
org01a	·OH	{none}	K(org01)*[org]*4d0
org02	·ORG + O_2	ORGIP	3.1d9
org03	ORGIP + ·OH	$3 \cdot$ R	1.0d8
org04	·R + O_2	$2CO_2$	1.0d9
org04a	O_2	{none}	K(org04)*[xr]*1d0
org05	$2 \cdot$ ORG	ORGIMP	1.0d9
org06	ORGIMP + ·OH	ORGIMP	5.0d9
org07	ORGIMP + ·OH	{decomp} + $3CO_2$	5.0d9
org07a	·OH	{none}	K(org07)*[orgimp]*20.0d0
org08	ORGIMP + e^-	OH^- + 1/2ORGIMP	5.0e9
org09	$2 \cdot$ R	ORGIMP	1.0d10
org10	·R + ·ORG	ORGIMP	1.0d10
org11	·R + ·OH	ORGIMP	1.0d10
org12	·ORG + ·OH	ORGIMP	1.0d10
org13	·R + ORGIMP	ORGIMP	1.0d6
l3_7f	·R + I_2	LVRI + ·I	6.3e9
l3_8f	·R + I_2	HVRI + ·I	3.1e8
l3_9f	·R + HOI	LVRI + ·OH	1.5e7
l3_10f	·R + HOI	HVRI + ·OH	7.0e5
l3_11f	LVRI + e^-	I^- + ·R	1.0e10
l3_12f	HVRI + e^-	I^- + ·R	1.0e10
l3_13f	LVRI	ORGIMP + I^- + H^+	1.3e−8*exp(100e3*tf)
l3_14f	HVRI	ORGIMP + I^- + H^+	1.3e−7*exp(104e3*tf)
imp1	I_2 + ORGIMP	I_2ORGIMP	1.2d6*exp(3406d0*(1d0/298d0−1d0/t))
imp1b	I_2ORGIMP	I_2 + ORGIMP	6d0

$*$ Written by Fortran expressions; K(X)=The rate constant of reaction X; Variable definitions are given in Table 2.6; Species in { } are eliminated in the definition of rate constants.
ORG: organic compounds (xylenes or MIBK from aqueous paints), ORGIP: intermediate species in the course of ORG decomposition with excess oxygen, ORGIMP: large molecules produced without oxygen by dimerization of organic radicals, I_2ORGIMP: complexes of I_2 and ORGIMP, ·R: small organic radicals producing organic iodides, ·ORG: organic radicals in the initial stage of ORG decomposition.

Table 2.4: Reaction set 10a : Reactions / Miscellaneous.

Index	Reactant	Product	Rate constant*
mc_1	$Fe^{3+} + \cdot O_2^-$	$Fe^{2+} + O_2$	1.5e8*exp(15100*tf)
mc_2	$Fe^{2+} + \cdot O_2^- + 2\{H_2O\}$	$Fe^{3+} + H_2O_2 + 2OH^-$	1e7*exp(15100*tf)
mc_3	$Fe^{2+} + \cdot HO_2 + \{H_2O\}$	$Fe^{3+} + H_2O_2 + OH^-$	1.2e6*exp(15100*tf)
l4_1	$Fe^{2+} + \cdot OH$	$Fe^{3+} + OH^-$	3e9*exp(15100.0*tf)
l4_3	$Fe^{2+} + H_2O_2$	$Fe^{3+} + \cdot OH + OH^-$	55.0*exp(85210.0*tf)
baf	$B(OH)3 + OH^-$	$B(OH)4^-$	1e10
bab	$B(OH)4^-$	$B(OH)3 + OH^-$	K(baf)*kkb_ba
l3_4ab	$H^+ + HCO_3^-$	$CO_2 + \{H_2O\}$	1e10
l3_4af	$CO_2 + \{H_2O\}$	$H^+ + HCO_3^-$	kkco2*K(l3_4ab)
l3_6b	$H^+ + CO_3^{2-}$	HCO_3^-	1.0e10
l3_6f	HCO_3^-	$H^+ + CO_3^{2-}$	kkhco3m*K(l3_6b)
cl1	$\cdot OH + Cl^-$	$\cdot HOCl^-$	4.3d9*ea
cl1b	$\cdot HOCl^-$	$\cdot OH + Cl^-$	6.1d9*ea
cl2	$\cdot HOCl^- + H^+$	$\cdot Cl + \{H_2O\}$	2.1d10*ea
cl3	$\cdot Cl + Cl^-$	$\cdot Cl_2^-$	2.1d10
cl3b	$\cdot Cl_2^-$	$\cdot Cl + Cl^-$	K(cl3)/kk_cl3
cl4	$\cdot Cl + I^-$	$\cdot I + Cl^-$	2d10*ea
cl5	$\cdot Cl_2^- + I^-$	$\cdot I + 2Cl^-$	2d10*ea
cl6b	I_2Cl^-	$I_2 + Cl^-$	1d5
cl6	$I_2 + Cl^-$	I_2Cl^-	K(cl6b)*kk_cl6
cl7	$2 \cdot Cl$	Cl_2	8.8d7*exp(−48d3/rgasgen *(1d0/t−1d0/298d0))
cl8	$Cl_2 + \{H_2O\}$	$HOCl + Cl^- + H^+$	kbol*t/hplan*exp(−8d0/rgasgen) *exp(−63d3/rgasgen/t)
cl8b	$HOCl + Cl^- + H^+$	$Cl_2 + \{H_2O\}$	kbol*t/hplan*exp(−71d0/rgasgen) *exp(−27d3/rgasgen/t)

$*$ Written by Fortran expressions; K(X)=The rate constant of reaction X; Variable definitions are given in Table 2.6; Species in { } are eliminated in the definition of rate constants.

Table 2.5: Reaction set 10a : Non-reaction processes.

Index	Dependent	Affecting	Factor*
	conc.	conc.	
ssg_1	[ISSg]	[I$_2$g]	vadissg*cut95([issg]/max(mlimissg,eps_24))
ssg_2	[ISSg]	[ISSg]	−kdeissg
ssg_3	[I$_2$g]	[I$_2$g]	−0.5*vadissg*assg/max(vg,epsvol)*cut95([issg] /max(mlimissg,eps_24))
ssg_4	[I$_2$g]	[ISSg]	0.5*kdeissg*assg/max(vg,epsvol)
spg_1	[ISPg]	[I$_2$g]	vadispg*cut95([ispg]/max(mlimispg,eps_24))
spg_2	[ISPg]	[ISPg]	−kdeispg*(one−mirvispg/max(max([ispg],mirvispg),eps_24))
spg_3	[I$_2$g]	[I$_2$g]	−0.5*vadispg*aspg/max(vg,epsvol)*cut95([ispg] /max(mlimispg,eps_24))
spg_4	[I$_2$g]	[ISPg]	0.5*kdeispg*aspg/max(vg,epsvol)*(one−mirvispg /max(max([ispg],mirvispg),eps_24))
sp_1	[ISP]	[I$_2$]	vadisp*cut95([isp]/max(mlimisp,eps_24))
sp_2	[ISP]	[ISP]	−kdeisp*(one−mirvisp/max(max([isp],mirvisp),eps_24))
sp_3	[I$_2$]	[I$_2$]	−0.5*vadisp*aspaq/max(vaq,epsvol)*cut95([isp] /max(mlimisp,eps_24))
sp_4	[I$_2$]	[ISP]	0.5*kdeisp*aspaq/max(vaq,epsvol)*(one−mirvisp /max(max([isp],mirvisp),eps_24))
spim_1	[IMSP]	[I$^-$]	vadimsp*cut95([imsp]/max(mlimimsp,eps_24))
spim_2	[IMSP]	[IMSP]	−kdeimsp
spim_3	[I$^-$]	[I$^-$]	−vadimsp*aspaq/max(vaq,epsvol)*cut95([imsp] /max(mlimimsp,eps_24))
spim_4	[I$^-$]	[IMSP]	kdeimsp*aspaq/max(vaq,epsvol)
tri2_1	[I$_2$g]	[I$_2$]	kmti2*aint/max(vg,epsvol)
tri2_2	[I$_2$g]	[I$_2$g]	−kmti2*aint/max(vg,epsvol)*hpi2
tri2_3	[I$_2$]	[I$_2$g]	kmti2*aint/max(vaq,epsvol)*hpi2
tri2_4	[I$_2$]	[I$_2$]	−kmti2*aint/max(vaq,epsvol)
tri2_1a	[I$_2$g]	[I$_2$]	qgflow*kkbi2/hpi2/max(vg,epsvol)
tri2_2a	[I$_2$g]	[I$_2$g]	−qgflow/max(vg,epsvol)
tri2_4a	[I$_2$]	[I$_2$]	−qgflow*kkbi2/hpi2/max(vaq,epsvol)
tri2con_1	[I$_2$g]	[I$_2$con]	kmti2con*acon/max(vg,epsvol)
tri2con_2	[I$_2$g]	[I$_2$g]	−kmti2con*acon/max(vg,epsvol)*hpi2
tri2con_3	[I$_2$con]	[I$_2$g]	kmti2con*acon/max(vcon,epsvol)*hpi2
tri2con_4	[I$_2$con]	[I$_2$con]	−kmti2con*acon/max(vcon,epsvol)
tri2con_5	[I$_2$]	[I$_2$con]	fcon/max(vaq,epsvol)
tri2con_6	[I$_2$con]	[I$_2$con]	−fcon/max(vcon,epsvol)
trhvri_1	[HVRIg]	[HVRI]	kmthvri*aint/max(vg,epsvol)
trhvri_2	[HVRIg]	[HVRIg]	−kmthvri*aint/max(vg,epsvol)*hphvri
trhvri_3	[HVRI]	[HVRIg]	kmthvri*aint/max(vaq,epsvol)*hphvri
trhvri_4	[HVRI]	[HVRI]	−kmthvri*aint/max(vaq,epsvol)
trhvri_1a	[HVRIg]	[HVRI]	qgflow*kkbhvri/hphvri/max(vg,epsvol)
trhvri_2a	[HVRIg]	[HVRIg]	−qgflow/max(vg,epsvol)
trhvri_4a	[HVRI]	[HVRI]	−qgflow*kkbhvri/hphvri/max(vaq,epsvol)
trlvri_1	[LVRIg]	[LVRI]	kmtlvri*aint/max(vg,epsvol)
trlvri_2	[LVRIg]	[LVRIg]	−kmtlvri*aint/max(vg,epsvol)*hplvri
trlvri_3	[LVRI]	[LVRIg]	kmtlvri*aint/max(vaq,epsvol)*hplvri
trlvri_4	[LVRI]	[LVRI]	−kmtlvri*aint/max(vaq,epsvol)
trlvri_1a	[LVRIg]	[LVRI]	qgflow*kkblvri/hplvri/max(vg,epsvol)
trlvri_2a	[LVRIg]	[LVRIg]	−qgflow/max(vg,epsvol)
trlvri_4a	[LVRI]	[LVRI]	−qgflow*kkblvri/hplvri/max(vaq,epsvol)

$*$ Written by Fortran expressions; cut95( )=Cut-off function at > 0.95

xxxg: gas phase species, xxxgout: gas phase species flowed out of the system, issg/ispg: adsorbed iodine on gas phase steel/painted walls, isp/imsp: adsorbed elemental iodine/iodide ion on aqueous phase painted walls, orgpaq: organic compounds in the aqueous phase paint.

Table 2.5: Reaction set 10a : Non-reaction processes.(continued)

Index	Dependent conc.	Affecting conc.	Factor*
trxi_1	[·Ig]	[·I]	fusexig*kmtxi*aint/max(vg,epsvol)
trxi_2	[·Ig]	[·Ig]	−fusexig*kmtxi*aint/max(vg,epsvol)*hpxi
trxi_2a	[·Ig]	[·Ig]	−fusexig*qgflow/max(vg,epsvol)
trxi_3	[·I]	[·Ig]	fusexig*kmtxi*aint/max(vaq,epsvol)*hpxi
trxi_4	[·I]	[·I]	−fusexig*kmtxi*aint/max(vaq,epsvol)
trcl2_1	[Cl$_2$g]	[Cl$_2$]	kmtcl2*aint/max(vg,epsvol)
trcl2_2	[Cl$_2$g]	[Cl$_2$g]	−kmtcl2*aint/max(vg,epsvol)*hpcl2
trcl2_3	[Cl$_2$]	[Cl$_2$]	−kmtcl2*aint/max(vaq,epsvol)
trcl2_4	[Cl$_2$]	[Cl$_2$g]	kmtcl2*aint/max(vaq,epsvol)*hpcl2
trcl2_2a	[Cl$_2$g]	[Cl$_2$g]	−qgflow/max(vg,epsvol)
fcl2	[Cl$_2$gout]	[Cl$_2$g]	qgflow
fi2	[I$_2$gout]	[I$_2$g]	qgflow
fhvri	[HVRIgout]	[HVRIg]	qgflow
flvri	[LVRIgout]	[LVRIg]	qgflow
fxi	[·Igout]	[·Ig]	fusexig*qgflow
disorg_1	[ORG]	[ORGPaq]	kdisorg*aspaq/max(vaq,epsvol)*kkdisorg
disorg_2	[ORG]	[ORG]	−kdisorg*aspaq/max(vaq,epsvol)
disorg_3	[ORGPaq]	[ORGPaq]	−kdisorg*aspaq/max(vspaq,epsvol)*kkdisorg
disorg_4	[ORGPaq]	[ORG]	kdisorg*aspaq/max(vspaq,epsvol)
trorg_1	[ORGg]	[ORG]	kmtorg*aint/max(vg,epsvol)
trorg_2	[ORGg]	[ORGg]	−kmtorg*aint/max(vg,epsvol)*hporg
trorg_2a	[ORGg]	[ORGg]	−qgflow/max(vg,epsvol) *forggout
trorg_3	[ORG]	[ORG]	−kmtorg*aint/max(vaq,epsvol)
trorg_4	[ORG]	[ORGg]	kmtorg*aint/max(vaq,epsvol)*hporg
tro2_1	[O$_2$g]	[O$_2$]	kmto2*aint/max(vg,epsvol)
tro2_2	[O$_2$g]	[O$_2$g]	−kmto2*aint/max(vg,epsvol)*hpo2
tro2_3	[O$_2$]	[O$_2$]	−kmto2*aint/max(vaq,epsvol)
tro2_4	[O$_2$]	[O$_2$g]	kmto2*aint/max(vaq,epsvol)*hpo2
trh2_1	[H$_2$g]	[H$_2$]	kmth2*aint/max(vg,epsvol)
trh2_2	[H$_2$g]	[H$_2$g]	−kmth2*aint/max(vg,epsvol)*hph2
trh2_3	[H$_2$]	[H$_2$]	−kmth2*aint/max(vaq,epsvol)
trh2_4	[H$_2$]	[H$_2$g]	kmth2*aint/max(vaq,epsvol)*hph2
trco2_1	[CO$_2$g]	[CO$_2$]	kmtco2*aint/max(vg,epsvol)
trco2_2	[CO$_2$g]	[CO$_2$g]	−kmtco2*aint/max(vg,epsvol)*hpco2
trco2_3	[CO$_2$]	[CO$_2$]	−kmtco2*aint/max(vaq,epsvol)
trco2_4	[CO$_2$]	[CO$_2$g]	kmtco2*aint/max(vaq,epsvol)*hpco2
trn2_1	[N$_2$g]	[N$_2$]	kmtn2*aint/max(vg,epsvol)
trn2_2	[N$_2$g]	[N$_2$g]	−kmtn2*aint/max(vg,epsvol)*hpn2
trn2_3	[N$_2$]	[N$_2$]	−kmtn2*aint/max(vaq,epsvol)
trn2_4	[N$_2$]	[N$_2$g]	kmtn2*aint/max(vaq,epsvol)*hpn2
trnh3_1	[NH$_3$g]	[NH$_3$]	kmtnh3*aint/max(vg,epsvol)
trnh3_2	[NH$_3$g]	[NH$_3$g]	−kmtnh3*aint/max(vg,epsvol)*hpnh3
trnh3_3	[NH$_3$]	[NH$_3$]	−kmtnh3*aint/max(vaq,epsvol)
trnh3_4	[NH$_3$]	[NH$_3$g]	kmtnh3*aint/max(vaq,epsvol)*hpnh3

∗ Written by Fortran expressions; cut95( )=Cut-off function at > 0.95
xxxg: gas phase species, xxxgout: gas phase species flowed out of the system, issg/ispg: adsorbed iodine on gas phase steel/painted walls, isp/imsp: adsorbed elemental iodine/iodide ion on aqueous phase painted walls, orgpaq: organic compounds in the aqueous phase paint.

Table 2.6: Reaction set 10a : Assignment of variables.

```
tce=t-273.15
tf=(1.0/298.15-1.0/t)/rgasgen
ea=exp(18800.0*tf)
dw=1.00017 - 2.36582e-5*tce - 4.77122e-6*tce**2 + 8.27411e-9*tce**3
rad=dose*1e3/3600.0/1.602e-19/100.0/6.022e23*dw
pkw=14.93-4.131e-2*tce+1.903e-4*tce**2-4.705e-7*tce**3+5.724e-10*tce**4
pkh=10.49-4.103e-2*tce+1.443e-4*tce**2-2.325e-7*tce**3+2.065e-10*tce**4
pkoh=12.50-3.317e-2*tce+1.964e-4*tce**2-6.198e-7*tce**3+8.244e-10*tce**4
pkho2=4.917-3.813e-3*tce+8.771e-7*tce**2+2.177e-7*tce**3-4.000e-10*tce**4
kkw=10.**(-pkw)
kkh=10.**(-pkh)
kkoh=10.**(-pkoh)
kkho2=10.**(-pkho2)
kkh2o2=kkoh
kk25=kkh2o2/kkw
kk27=kkw/kkh
kk29=kkoh/kkw
kk31=kkho2/kkw
ionst = 0d0
pkb_ba = 1573.21d0/t+28.6059d0+0.012078d0*t-13.2258d0*log10(t)+(0.3250d0-0.00033d0*t)
*ionst-0.0912d0*ionst**1.5
kkb_ba = 10d0**(-pkb_ba)
kk12=(1.0/dw)*10.0**(555.0/t+7.355-2.575*log10(t))
kk1=10.0**(13880.0/t-0.2445*t+308.4*log10(t)-749.1)
kk2=3e-3
if(index(mdlctrl,”useih2”)>0) fuseih2=1d0
if(index(mdlctrl,”usexig”) > 0) fusexig = 1d0
call acidbaseequil(”co2”,t,kkco2)
call acidbaseequil(”hco3_m”,t,kkhco3m)
kk_cl3 = 1.9d5
kk_cl6 = 1.7
if(index(mdlctrl,”orggout”) > 0) forggout = 1d0
```

Written by Fortran expressions; “if(index(mdlctrl,*string*)>0)” means “if *string* option is selected”; Subroutine “acidbasesquil” gives acid-base equilibrium constants.

2.2 Construction of the Chemical Kinetics System Equations

Kiche does the numerical analysis of a chemical kinetics system, by constructing a set of simultaneous ordinary differential equations for the evolution of concentrations from a reaction set such as presented in the previous section, and by solving it numerically. This section describes the former part.

The processes included in the kinetics model are in two categories as shown in Section 2.1, chemical reactions and other kinetic processes that are not expressed by chemical reaction equations. Here, we call the latter non-reaction processes. As a general expression, a reaction r (suffix) is written as follows,

$$r_1R_1 + r_2R_2 + \cdots + r_{m_r}R_{m_r} \xrightarrow{k_r} p_1P_1 + p_2P_2 + \cdots + p_{n_r}P_{n_r} \tag{2.1}$$

where k_r is the rate constant of the reaction r (suffix); R and P are the reactant and product species, respectively; r_x and p_x are corresponding stoichiometric coefficients, respectively. Based on this reaction, the concentration change rate of a reactant component R_i and a product component P_j include a consuming and a producing contribution terms, respectively, as follows,

$$\begin{aligned} \frac{d[R_i]}{dt} &= \cdots - r_ik_r[R_1]^{r_1}[R_2]^{r_2}\cdots[R_{m_r}]^{r_{m_r}} - \cdots \ , \qquad (2.2)\\ \frac{d[P_j]}{dt} &= \cdots + p_jk_r[R_1]^{r_1}[R_2]^{r_2}\cdots[R_{m_r}]^{r_{m_r}} + \cdots \ , \qquad (2.3) \end{aligned}$$

where suffixes i and j are one of $1, 2, \cdots, m_r$ and $1, 2, \cdots, n_r$, respectively.

On the other hand, contribution of a non-reaction process n that changes the concentration of species X_i is expressed as follows,

$$\frac{d[X_i]}{dt} = \cdots + k_n \underbrace{[X_{n,1}][X_{n,2}] + \cdots\cdots [X_{n,l_n}]}_{\text{Product of arbitrary set of concentrations}} \tag{2.4}$$

where k_n is the rate constant of the process n. This expression indicates the product of the concentrations of species $X_{n,1}$ through X_{n,l_n} drives the evolution of the concentration change of a species X_i. Note that the product of the concentrations may have duplication of species in it, or may have no species. So, higher order or zero-th order kinetics can be accommodated in the form. A process including multiple terms can be decomposed into multiple processes and casted into the above scheme.

All the processes in the system in above two categories, then, construct a system of simultaneous ordinary differential equations. The equation for the concentration change of a species X_i is expressed as follows,

$$\begin{aligned} \frac{d[X_i]}{dt} &= \sum_r(-r_ik_r[R_1]^{r_1}[R_2]^{r_2}\cdots[R_{m_r}]^{r_{m_r}}) \\ &+ \sum_r(p_ik_r[R_1]^{r_1}[R_2]^{r_2}\cdots[R_{m_r}]^{r_{m_r}}) \\ &+ \sum_n(k_n[X_{n,1}][X_{n,2}]\cdots[X_{n,l_n}]) \qquad (2.5) \end{aligned}$$

where the 1st sum for r is taken for reactions involving X_i in the reactants, the 2nd sum for reactions involving X_i in the products; and the sum for n for non-reaction processes affecting X_i.

Thus, change of every concentration is expressed by a polynomial of arbitrary order including potentially all the concentrations in the system,

$$\frac{dY_k}{dt} = f_k(Y_1, Y_2, \cdots, Y_M) \tag{2.6}$$

where Y_k is the concentration of species k; M is the number of species included in the system; f_k is a polynomial of concentrations.

2.3 Numerical Solution Method

Kiche solves the simultaneous ordinary differential equations for the system, Eq.(2.6), by a simple first order finite difference method with implicit temporal discretization as follows,

$$\frac{Y_k^{n+1} - Y_k^n}{t^{n+1} - t^n} = f_k(Y_1^{n+1}, Y_2^{n+1}, \cdots, Y_M^{n+1}) \tag{2.7}$$

where n and $n+1$ denote the current and the new time steps, respectively. Reaction kinetics models often show a “stiff” nature and a practical fast solution needs the implicit scheme. The finite difference equations have nonlinear terms of concentrations in the right hand side, and are solved by Newtonian iteration[25].

With a superscript p indicating intermediate values during the iteration, and the residual errors $\delta Y_k = Y_k^{n+1} - Y_k^p$, Eq.(2.7) leads to the Newtonian iteration equation

$$\frac{Y_k^p + \delta Y_k - Y_k^n}{t^{n+1} - t^n} = f_k^p + \sum_{i=1}^{M} \left(\frac{\partial f_k}{\partial Y_i}\right)^p \delta Y_i \tag{2.8}$$

where f_k^{n+1} in the right hand side of Eq.(2.7) is approximated by the 1st order Taylor expansion around the intermediate values. This constructs a linear equation set for the residuals δY_k,

$$-\frac{1}{t^{n+1} - t^n}\delta Y_k + \sum_{i=1}^{M} \left(\frac{\partial f_k}{\partial Y_i}\right)^p \delta Y_i = \frac{1}{t^{n+1} - t^n}(Y_k^p - Y_k^n) - f_k^p. \tag{2.9}$$

The residuals obtained by this linear equation set give the corrections to move the intermediate values Y_k^p closer to the solutions Y_k^{n+1}.

Thus, the solution algorithm to obtain the concentrations at the new time step from those at the old time step is as follows.

1. Assign old time step concentrations Y_k^n to Y_k^p as initial guess.
2. Solve the simultaneous linear equations Eq.(2.9) for the corrections δY_k.
3. Assign $Y_k^p + \delta Y_k$ to Y_k^p.
4. Iterate the item 2–3 until the absolute values of the corrections δY_k become small enough.

The linear equation set Eq.(2.9) is also expressed by vectors and a matrix,

$$\boldsymbol{J} \cdot \boldsymbol{x} = \boldsymbol{c} \tag{2.10}$$

where $\boldsymbol{J}$ is the Jacobian matrix consisting of the coefficients for the corrections,

$$J(k,i) = \begin{cases} -\dfrac{1}{t^{n+1} - t^n} + \left(\dfrac{\partial f_k}{\partial Y_k}\right)^p & \text{for } i = k, \\ \left(\dfrac{\partial f_k}{\partial Y_i}\right)^p & \text{otherwise;} \end{cases} \tag{2.11}$$

vectors $\boldsymbol{x}$ and $\boldsymbol{c}$ are the corrections and the constant terms,

$$x_k = \delta Y_k, \tag{2.12}$$

$$c_k = \frac{1}{t^{n+1} - t^n}(Y_k^p - Y_k^n) - f_k^p. \tag{2.13}$$

Kiche solves the linear equation set by Gaussian elimination method[25].

Usually, every chemical species is involved in only a few processes out of more than a hundred. So, the matrix $\boldsymbol{J}$ is often sparse. There are methods to handle sparse matrices more efficiently[25] and improvement might be possible by utilizing them.

At present, we use the conventional Gauss method for simplicity and robustness. And, it is actually practical enough. For instance, chemical evolution of a system with ∼70 species, more than a hundred reactions and non-reaction processes for 300 h with maximum time step of 50 s (such as the cases in Chapter 4) is solved within a minute on a Intel 32 bit CPU of 2.3 GHz clock.

3 Code Description

3.1 Overall Concept of Kiche

Figure 3.1 shows the concept of the Kiche tool which provides numerical solution of chemical kinetics models. A Fortran 95[1] code "`kiche`" solves the chemical kinetics model such as described in Chapter 2. The part of its source code that is dependent on the reaction set—the *chemistry model*–is not hard-coded, but generated from a "reaction database" that is in the plain text format, thus readable and editable by users.

A Perl[2] script "`chem2f95`" is a tool to do the conversion of the text form database into Fortran codes. Practically, it reads the reaction and non-reaction expressions from the database, extracts species and rate constants, and write Fortran source slips for the definition of species and processes including the reactions and the non-reactions, and the rate constants of the processes. Those source slips are written as include files and referred from the main body of `kiche` source code.

Another script `chem2tex` is provided to generate a LaTeX[3] source from the reaction database for the text presentation of the reaction equations, rate constants and other variables. The script also automates generation of PDF[4] file by processing the LaTeX source by a LaTeX system and a PDF creator.

With the help of the conversion systems described above, users can easily understand what reactions are involved in the calculation, modify or newly develop the reaction set by themselves.

When the system is used for the first time or when the reaction set is updated, the following steps are needed for the calculation.

1. Generation of the part of Fortran code, the chemistry model, by `chem2f95`.

2. Compilation of `kiche` source code to make the execution module.

Note that the execution module of `kiche` is generated separately for every reaction database referred. As a convention, it is recommended that a suffix to identify the reaction database is given to the execution file name, *e.g.* `kiche.10a`, `kiche.mymodel` and so on. A "Makefile"[5] is provided to do this automatically.

Practical information on the reaction database and the input/output data for `kiche` is described in detail in Appendixes A and C.

3.2 Data Structure for Species and Processes

In the `kiche` code, the chemistry model is stored as structured data in three categories: species, reactions and non-reaction processes. As shown in the part of the source code, Fig. 3.2, they are defined as "derived types" having necessary elements, *i.e.* `spcobj`, `reacobj` and `nonreacobj`. Then, actual variables are declared as matrices of those types, *i.e.* `spcs`, `reac` and `nonreac`.

As described in Section 2.2, once those model constituents are stored as structured data, it is relatively easy to construct the system equations and the Jacobian matrix for the numerical

[1] Fortran 95 is a standard for a computer language Fortran, officially numbered as ISO/IEC-1539-1997[26].

[2] Perl (Practical Extraction and Report Language) is a script language having advantages in text processing[27].

[3] LaTeX is a high-quality typesetting system for production of technical and scientific documentation[28].

[4] PDF (Portable Document Format) is a standard for document exchange, developed by Adobe Systems Inc. and has become an open standard ISO/IEC 32000-1[29].

[5] Makefile is a default file read by "make", a utility software to automate shell processes, typically compilation of programs.

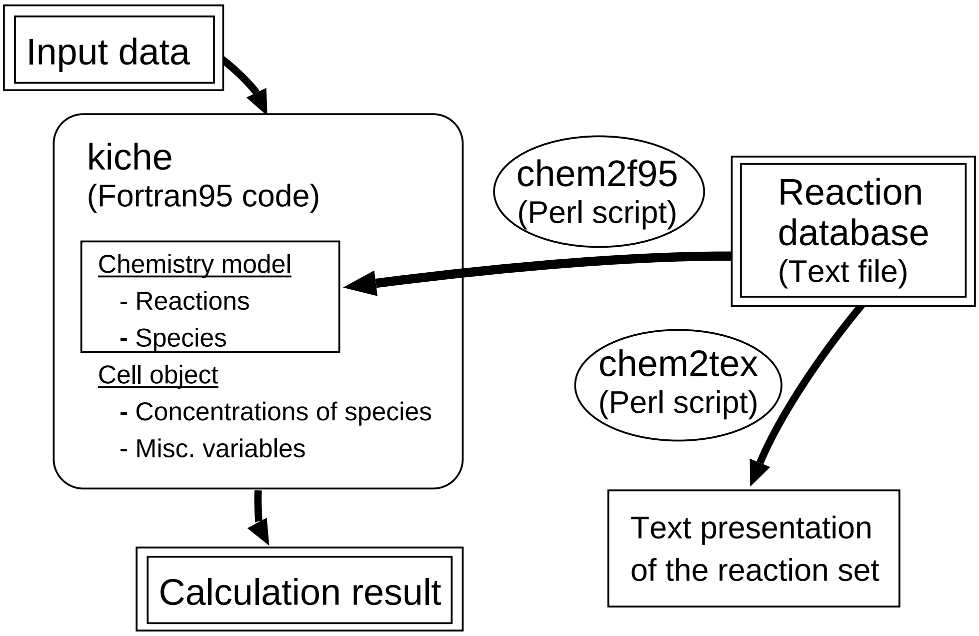


Figure 3.1: Relation of Kiche code, reaction database and peripheral tools

solution. Also, from the viewpoint of the processes necessary for the conversion of the text reaction database into Fortran code, this method makes the extraction of information easy.

Actually, the `chem2f95` script simply does the extraction of species names and writing the source to store them into the matrix `spcs`, parse of reaction and non-reaction expressions and writing the source to store the information into `reac` and `nonreac` matrices, then, copy of the rate constant expressions. The chemical species are alphabetically sorted when indexed by integer numbers. The reaction and non-reaction processes are indexed and stored by the order as it appears in the reaction database. When `kiche` is compiled by the Makefile, the Fortran sources generated by `chem2f95` are written in include files, `setchem.inc` and `rconst.inc`. Additionally, necessary size of those matrices are evaluated according to the number of species and processes involved in the model, and written in another include file `sizedef.inc`.

3.3 Data Structure for the System Cell

The system to be simulated has its attributes such as aqueous and gaseous volumes, wall surface areas, gas-liquid interface area, temperature, dose rate, and so on. Figure 3.3 shows the definition of the derived type `cellobj` for the storage of the attributes of the system volume called a "cell".

Note that, the `cellobj` type has a matrix element to store the concentrations of species. This is for the storage. The concentration element of `spcs` (`spcsobj` type) is referred during the chemical evolution calculation, and the result is stored in an element of the cell attribute variable.

At present, `kiche` handles one cell. However, by this design, separating the handling of cell attributes from the chemistry model, it should be relatively easy to extend the code for multiple cells.

3.4 Overall Structure and Flow of the Code

The overall module structure of the code is given in Table 3.1. The modules for the variables for chemistry model and system cell, "`chemvars`" and "`cellvars`", and for the chemistry tools,

```
========= vars_chem.f95 ==========
!
! variables for the chemistry model
!
module chemvars
  use modelsize
  implicit none
  save

  ! species object
  type spcsobj
    character(len=maxlensname) :: name  ! name of species
    real(kind(1d0)) :: conc, iatom    ! concentration (mol/dm3, mol/dm2, etc.)
                                     ! number of iodine atom(s) in a molecule of the species
  end type spcsobj

  ! reaction object  r1R1+r2R2+...+riRi+...+rnrRnr -> p1P1+p2P2+...+piPi+...+pnpPnp
  type reacobj
    character(len=maxlenname) :: name  ! name of reaction
    integer :: nr,             &  ! number of reactant specie(s)
               spr(maxnrnpnn),  &  ! reactants 1..nr (species index)
               np,             &  ! number of product specie(s)
               spp(maxnrnpnn)      ! products 1..np (species index)
    real(kind(1d0)) :: &
               nspr(maxnrnpnn), &  ! number of reactant molecle(s) r1..rnr (order of the reaction)
               nspp(maxnrnpnn), &  ! number of product molecle(s) p1..pnp
               k                   ! reaction rate constant
  end type reacobj

  ! non-reaction object    d[C]/dt = k*[Cdep1][Cdep2]...[Cdepnn]
  type nonreacobj
    character(len=maxlenname) :: name  ! name of non-reaction process
    integer :: sp,             &  ! species C (species index)
               nn,             &  ! number of depended species (order of the process)
               spdep(maxnrnpnn)    ! species depended, Cdep (species index)
    real(kind(1d0)) :: k            ! rate constant
  end type nonreacobj

  ! actual variable definition for chemistry model objects
  type(spcsobj) :: spcs(1:nspcs)
  type(reacobj) :: reac(1:nreac)
  type(nonreacobj) :: nonreac(1:nnonreac)
....
========= vars_chem.f95 ==========
```

Figure 3.2: Source code for the definition of the data structure to store the species, reactions and non-reaction processes.

```
========= vars_chem.f95 ==========
! cell object data type and some manipulation are defined
! (intend to have a packaged set of cell state data
!  for exchange between subroutines and with other codes,
!  e.g. thermohydraulics)
!
module cellvars
  use modelsize
  implicit none
  save

  ! gas composition
  type gascomp
     real(kind(1d0)) :: po2, pn2, ph2, pco2, pnh3, pother, & ! partial p (Pa)
          xo2, xn2, xh2, xco2, xnh3, xother                ! mol fraction
  end type gascomp

  ! cell object (all the info of a cell is contained)
  type cellobj
     real(kind(1d0)) :: &
       dose, &           ! dose rate (kGy/h)
       temp, tempg, &    ! T of liq and gas phases (K)
       xhm,         &    ! relative humidity (=pst/psat)
       ptot,        &    ! total p (Pa)
       psat,        &    ! sat. p for tempg (Pa)
       pst, pgas,   &    ! steam, gas partial pressure (Pa)
       vg, vaq,     &    ! volume of gas, aqueous phases (dm3)
       aint,        &    ! area of gas-aq. interface (dm2)
       assg, aspg,  &    ! steel, paint wall area in gas (dm2)
       aspaq,       &    ! paint wall area in aq. (dm2)
       agepaq,      &    ! age of the aqueous paint (day) (for org content)
       ufg,ufaq,    &    ! velocity in gas/aqueous phases along the wall(dm/s)
       wlsg,wlpg,wlpaq,& ! wall length, gas  steel/paint/aq paint (dm)
       tpg, tpaq,   &    ! thickness of paint in gas, aq. (dm)
       vspg, vspaq, &    ! volume of paint in gas, aq. (dm3)
       acon,        &    ! condensate area (dm2)
       hcon,        &    ! condensation wall height (dm)
       mcon,        &    ! condensation mass flux (kg/s.dm2)
       twcon,       &    ! condensation wall temperature (K)
       vcon, tcon,  &    ! condensate volume (dm3), thickness (dm)
       fcon,        &    ! condensate flow rate (dm3/s)
       ucon,        &    ! condensate average flow velocity (dm/s)
       conc(nspcs)       ! concentration of chem. species (mol/dm3),
                         ! (mol/dm2) or (mol)
                         ! indexed as same as species structure array.
       ! chemical state is only saved as "conc", a set of copy of spcs()%conc
     type(gascomp) :: gas
     integer :: typ    ! typ=0 normal, 1 vg<epsvol, 2 vaq<epsvol
  end type cellobj
......
========= vars_chem.f95 ==========
```

Figure 3.3: Source code for the definition of the data structure to store the system cell attributes.

Table 3.1: Module structure of kiche code.

Source file	Module	Contents ((sub): subroutine name)
chemtool.f95	chemtool	(sub)setupchem: Set-up of the chemistry model.
		(sub)evolchem: Numerical solution for the time step advancement by chemical kinetics model.
		(sub)rconst1, rconst2: Calculation of rate constants.
initinout.f95	initinout	(sub)initinput: Initialization and input processing.
		(sub)output: Output processing.
		Other related subroutines are contained.
misccorr.f95	misccorr	Miscellaneous correlations and models (see Table 3.2).
vars.f95	ctrlvars	Definition for controlling variables. Subroutines handling time table data are contained.
	tctrlvars	Definition for controlling variables related to time.
vars_chem.f95	modelsize	Definition of matrix sizes for the chemistry model.
	cellvars	Definition of the system cell data structure, related variables and subroutines.
	chemvars	Definition of the chemistry model system data structures, and related variables.
General purpose modules		
consts_mod.f95	consts	General physical and mathematical constants.
iochartools_mod.f95	iochartools	I/O and character handling tools.
math_mod.f95	math	Mathematics tools including matrix solvers.
misc_mod.f95	misc	Miscellaneous tools for data handling.
wrsttab.f95	wrsttab	Fast running steam table (water physical properties) for wide range of thermal conditions.

“chemtool”, provide the most important functions of kiche code.

The overall flow of the code is illustrated in Fig. 3.4. The main program controls the overall processes, and the subroutine “evolchem” handles the numerical solution of the chemical kinetics model.

The time step is controlled in evolchem. If it fails in the matrix inversion, Δt is reduced by a factor of 0.75, concentrations are reset to the old step values, and another trial is given. If the fail continues till Δt becomes smaller than the lower limit, dtmin= 10^{-9} s by default, the run is quit. Contrary, when the solution is obtained by a small number of iterations, Δt is increased by a factor of 1.05.

Note that the system “cell” is defined as a local variable in the main program. Thus, the system to be simulated can be flexibly modified by replacing the main program with other programs or an interfacing program to others.

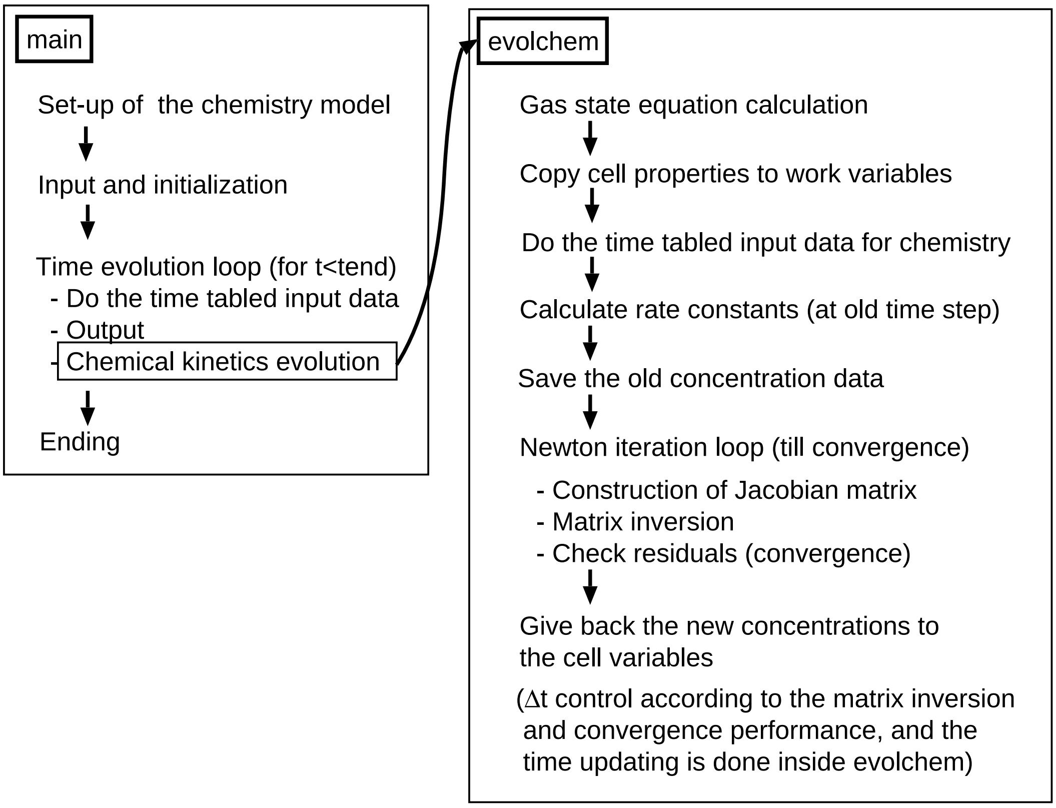

main
Set-up of the chemistry model
Input and initialization
Time evolution loop (for t<tend)
- Do the time tabled input data
- Output
- Chemical kinetics evolution
Ending
evolchem
Gas state equation calculation
Copy cell properties to work variables
Do the time tabled input data for chemistry
Calculate rate constants (at old time step)
Save the old concentration data
Newton iteration loop (till convergence)
- Construction of Jacobian matrix
- Matrix inversion
- Check residuals (convergence)
Give back the new concentrations to the cell variables
(Δt control according to the matrix inversion and convergence performance, and the time updating is done inside evolchem)

Figure 3.4: Flow of `kiche` code.

3.5 Built-in Physical and Chemical Property Data and Its Extension

A module "`misccorr`" provides miscellaneous subroutines for the physical and chemical properties related to gas-aqueous partition of volatile species, dissolution of organic compounds from wall paints, adsorption and desorption of iodine species on walls, acid-base equilibria, wall condensate film behaviors, and so on.

At present, we have the correlations as listed in Table 3.2. Users can add new subroutines there so that they can use them in their reaction databases.

The "`wrsttab`" module, originally developed for severe accident simulation[30] (see Table 3.1), provides water physical properties in a wide range of temperatures (274–1000 K) and pressures (10^3–10^8 Pa). It is at present refferred only in subroutines in `misccorr` module.

Table 3.2: Physical/Chemical data correlations and models in `misccorr` module.

Subroutine	Functions of the correlations and models	Reference
hpargas	Gas-aqueous partition coefficients and interface mass transfer velocities of gases (O_2, N_2, CO_2, NH_3, NO, NO_2, O_3), organic solvent dissolved from aqueous paint (ORG), and iodine species (I_2, ·I, HVRI, LVRI).	[31][32][33][34] [35][36]
kkorgdissol	The equilibrium constant and the rate of leaching of organic solvents from epoxy paints in aqueous phase. (ORGPaq $\leftrightarrow$ ORG)	[37][38]
c0orgdissol	The initial concentration of organic solvents remaining in the paint matrix (depending on the age and the paint thickness).	[37][38][33]
sorp_sg	Adsorption/desorption of I_2 on gaseous steel surfaces.	[39][40]
sorp_paint	Adsorption/desorption of I_2 on gaseous or aqueous painted surfaces, and that of I^- on aqueous painted surfaces.	[39]
gasrule	Gas phase pressure and concentration calculations.	
acidbaseequil	Acid-base equilibrium constants (HSO_4^-, HCO_3^-, CO_2, NH_4^+, $H_2PO_4^-$, H_3PO_4, HCOOH, RCOOH).	[41][42][43]
wallcondfilm	Calculation of the flow rate and average thickness of falling condensate water film on the wall.	

The source code of the correlations and models are found in `misccorr.f95`.

4 Verification and Example Calculations

4.1 Small Scale Experiment at JAEA

We performed experiments in a small scale system, Fig. 4.1, in which aqueous CsI solution with prescribed additives was irradiated in a Gamma-cell and the cover gas was swept. The evolution of gaseous release fractions of elemental iodine and organic iodines were separately monitored at the filters in the downstream[10]. Influences of added organic compound, methylisobutylketone (MIBK), and oxygen concentration in the cover gas was examined. Simulations of this series of experiments are presented here.

The experimental conditions are given in Table 4.1. In the calculations, the initial concentrations of iodine (as I^-), MIBK and boric acid/NaOH buffer at specified pH ($\sim$7), temperature, dose rate, gas composition and flow rate were given according to the experimental condition. The pH was set at the experimental condition at the start, and the following change was calculated. The interface mass transfer velocities were given by a method we call “O_2 basis relative evaluation” (see the item “kmto2base” in “Control keywords” paragraph of Appendix C.1.2). For the basis mass transfer velocity for O_2 at 25 °C, 4×10^{-4} dm/s as measured in the system was used. The adsorption on the vessel wall was negligible, and not considered in the calculation.

Figures 4.2–4.4 are the comparison of the experimental data and calculation results on the history of gaseous iodine release fractions. The release as I_2 and organic iodines are separately compared. Also, the present model “10a” was compared with the one that is the same as “10a” except the organic reaction model is the one in LIRIC 3.2[11]. It uses simpler reactions,

$$\begin{array}{ll} \mathrm{ORG} + \cdot\mathrm{OH} \longrightarrow 6\cdot\mathrm{R} & (\mathrm{1d9}), \\ \cdot\mathrm{OH} \longrightarrow \{\mathrm{none}\} & (\mathrm{1d9*5d0*[org]}), \\ \cdot\mathrm{R} + \mathrm{O_2} \longrightarrow \mathrm{CO_2} & (\mathrm{1d9}), \end{array}$$

instead of reactions org01–org13 in Table 2.3. It was developed based on the study on the radiolytic degradation of methylethylketone (MEK) at AECL/Canada, and verified with experiments in various conditions with mostly air environment[24, 22, 23]. The present organic reaction model was developed based on their work to improve the performance to cover low oxygen concentration conditions relevant to the containment environment of BWRs.

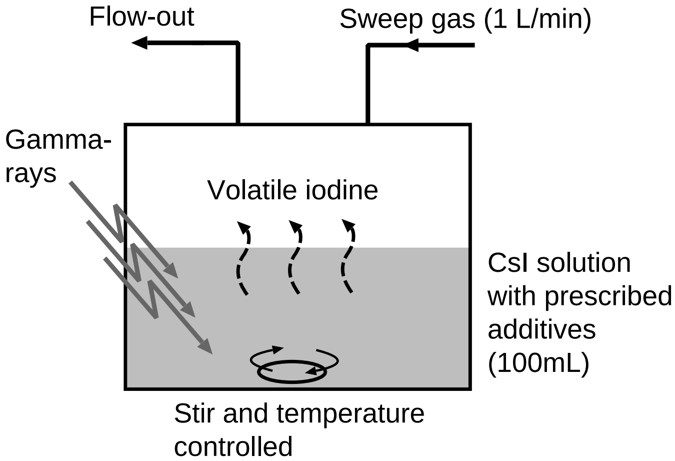

Flow-out
Sweep gas (1 L/min)
Gamma-rays
Volatile iodine
CsI solution with prescribed additives (100mL)
Stir and temperature controlled

Figure 4.1: Small scale experiment at JAEA

Table 4.1: Conditions of JAEA small scale experiment.

Case	Sweep gas	MIBK conc.	Dose rate	pH
		(mol/L)	(kGy/h)	(initial)
Base	Air	0	6.6	6.97
OrgL		1×10^{-4}	7.4	6.99
OrgH		1×10^{-3}	6.9	6.93
LO	N_2-2%O_2	0	6.6	7.03
OrgL-LO		1×10^{-4}	6.5	7.12
OrgH-LO		1×10^{-3}	6.7	7.04
N2	N_2	0	7.5	6.72
OrgL-N2		1×10^{-4}	7.4	6.99
OrgH-N2		1×10^{-3}	6.9	7.16

Common condition: Initial I^- conc. 1×10^{-4}mol/L; Buffer 0.1mol/L $B(OH)_3$-NaOH; Solution volume 100mL, Gas volume in the vessel 286mL, Gas-aq. interface area 0.567dm^2 (ϕ85mm); Sweep gas flow rate 1L/min; Stir $\phi6 \times 25$(mm) glass stir bar, 350rpm; Temperature 25±2°C during irradiation; Irradiation period 2h.

Figures 4.2–4.4 show that, the present model “10a” simulates the history of both I_2 and organic iodines (“RI” in the figures) relatively well for most of the conditions, and significant improvement from the LIRIC 3.2 model is seen in low/no oxygen conditions.

Figure 4.5 shows the comparison of the fractions of iodine release as I_2 and organic iodines, by models “10a” and “10a with LIRIC 3.2 organic reactions” for all the cases. The LIRIC 3.2 organic model tends to overestimate the concentration of the organic radical and the release as organic iodines in low or no oxygen conditions, because it assumes the decay path of the organic radical only through oxygen. This point was improved in the present model by considering another path that takes effect in low/no oxygen concentration conditions (see Table 2.3)[10].

Note that small fractions of organic iodines were released even with no MIBK addition in the experiments. This is probably because the existence of organic impurity in the water and reagents. In the calculations, 1.4×10^{-6} M of MIBK corresponding to 100 ppb TOC (total organic carbon) was assumed. The calculations by the present model show less organic iodine release than LIRIC 3.2 organic model with this very low organic imurity (Base, LO and N2 cases), though it is considered not so technically significant.

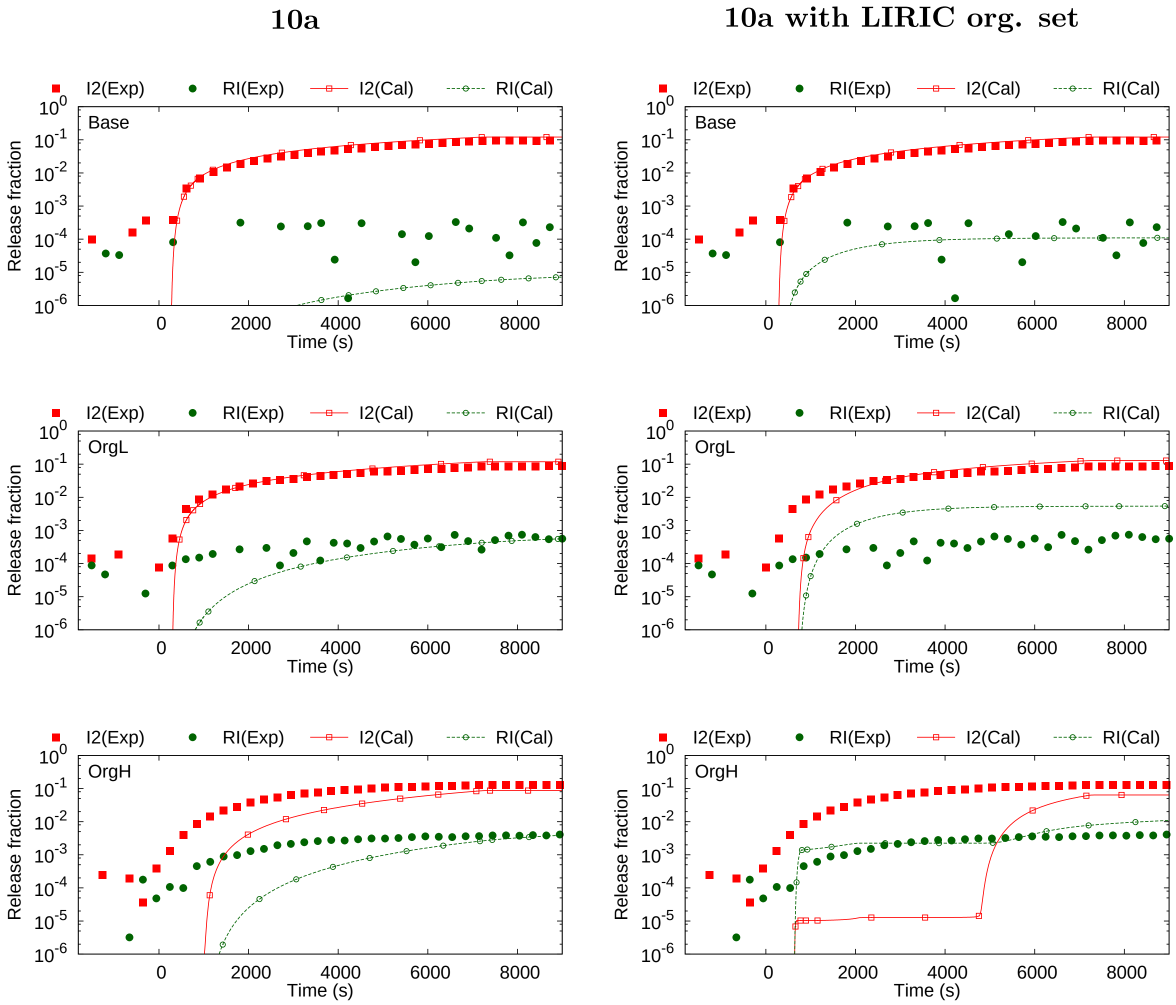


Figure 4.2: Gaseous iodine release history: comparison of experimental data and calculation results, 10a (left) and 10a with LIRIC organic reaction set (right); I2: elemental iodine, RI: organic iodines; Origin of time is at the start of irradiation; Sweep gas: Air.

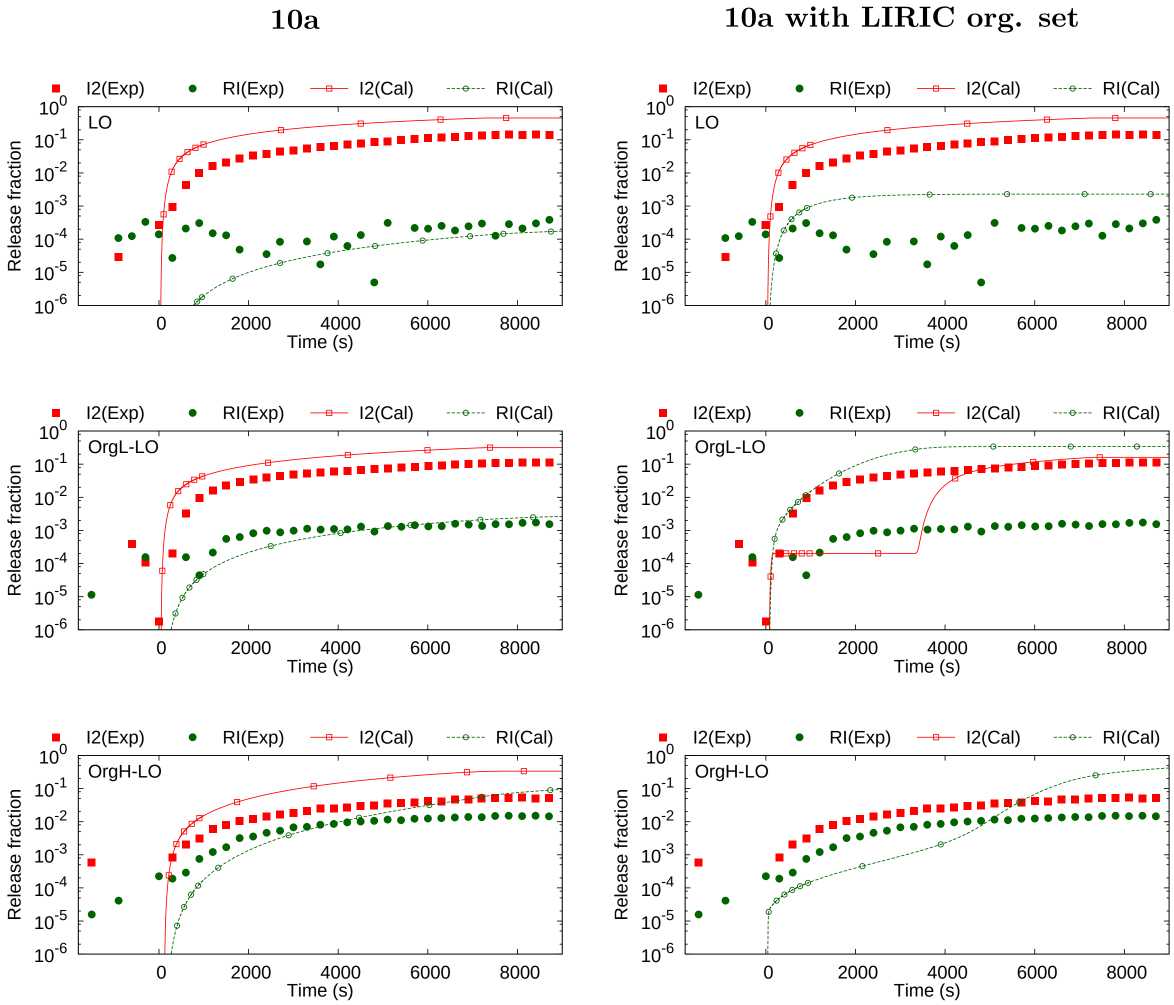


Figure 4.3: Gaseous iodine release history: comparison of experimental data and calculation results, 10a (left) and 10a with LIRIC organic reaction set (right); I2: elemental iodine, RI: organic iodines; Origin of time is at the start of irradiation; Sweep gas: N_2-2%O_2.

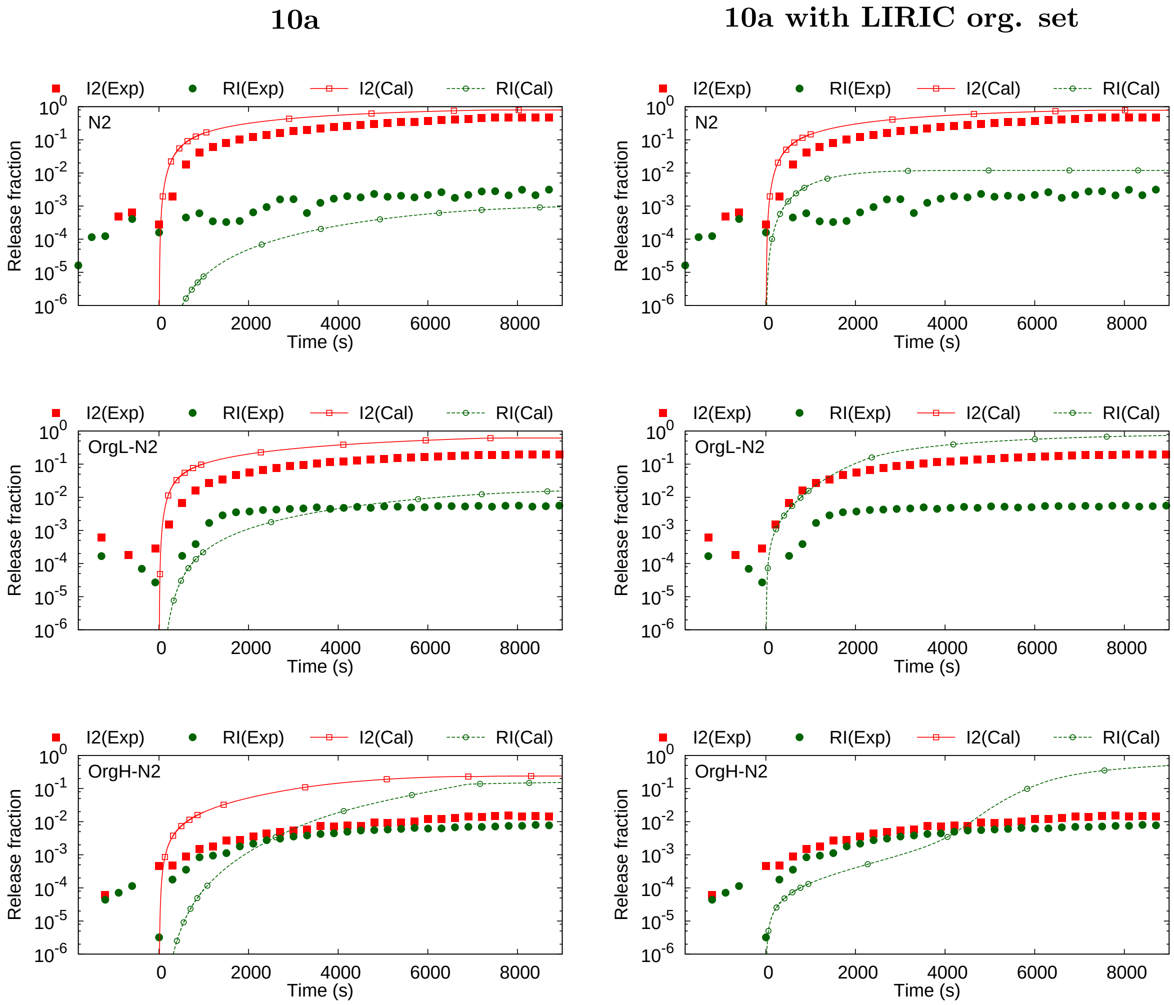
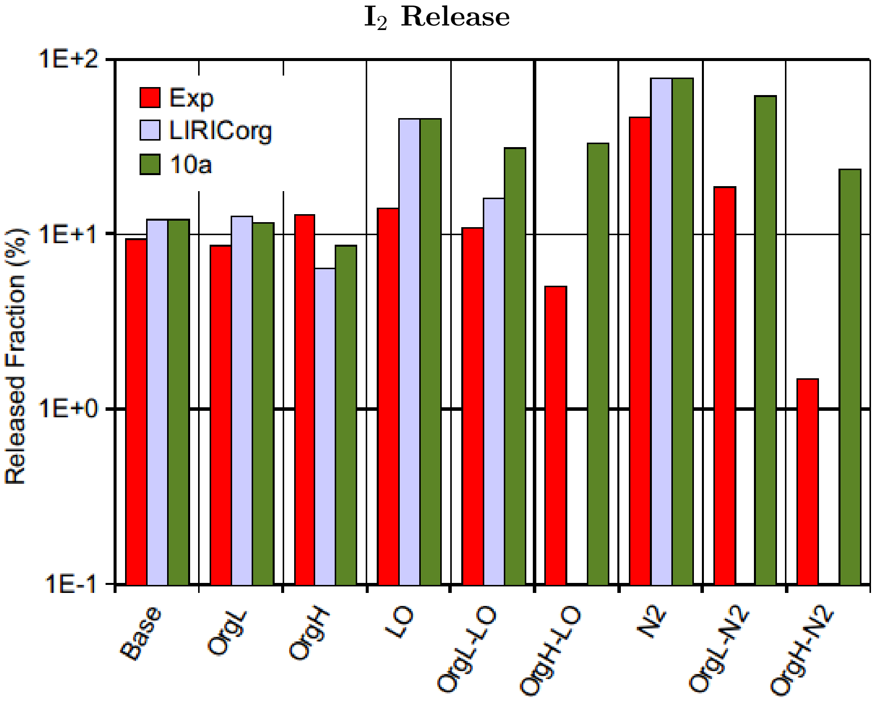


Figure 4.4: Gaseous iodine release history: comparison of experimental data and calculation results, 10a (left) and 10a with LIRIC organic reaction set (right); I2: elemental iodine, RI: organic iodines; Origin of time is at the start of irradiation; Sweep gas: N_2.



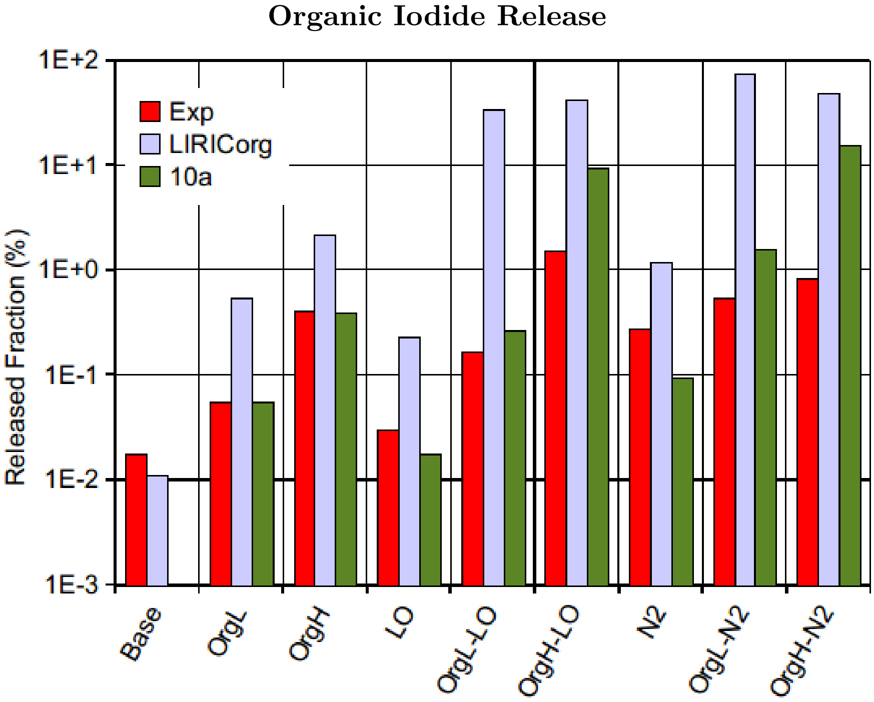


Figure 4.5: Gaseous iodine release fractions: comparison of experimental data and calculation results at the end of irradiation. The cases and conditions are described in Table 4.1. (Exp: experiments, LIRICorg: 10a with LIRIC 3.2 organic model, 10a: 10a model.)

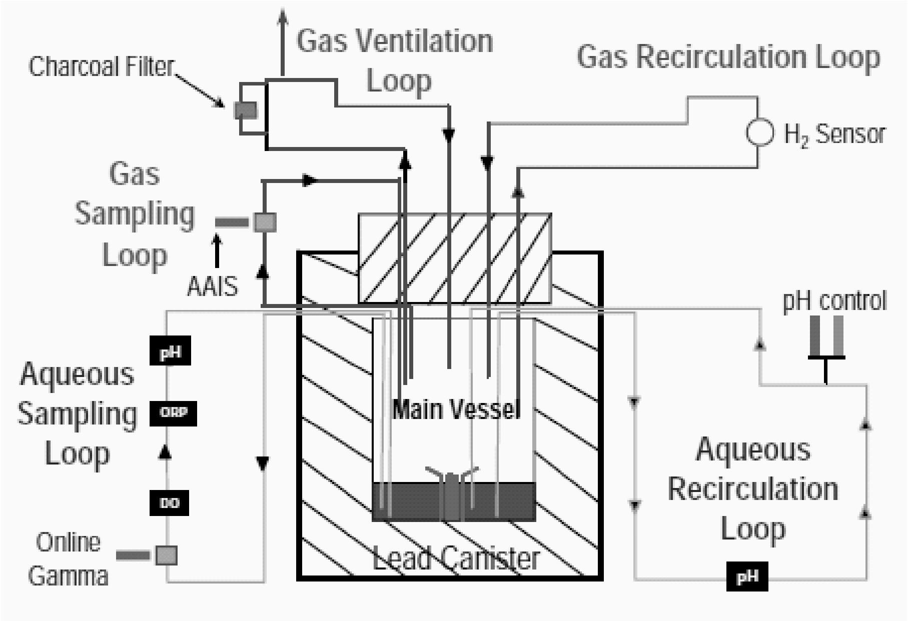


Figure 4.6: RTF test apparatus.[44, 46]

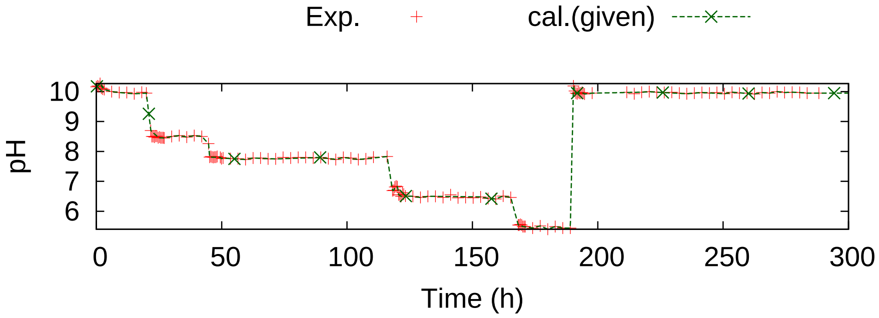


Figure 4.7: pH control in RTF Phase-0 Test-2 Stage-2 experiment.

4.2 RTF Experiment with a Steel Vessel

RTF (Radioiodine Test Facility) was an experimental apparatus at AECL/Canada where middle scale (∼300 L gamma-irradiated tank) iodine chemistry experiments were performed. Some cases of the experiments were referred for International Standard Problems (ISPs) of OECD/NEA[44, 45]. Figure 4.6 shows the schematic of the apparatus. About 25 L of ^{131}I-labeled iodide solution was contained in a vessel, in which ^{60}Co gamma source was located at the center. The evolution of iodine concentration in the cover gas and the solution was measured by sampling loops equipped with gamma counters.

The case referred in ISP-41[44], Phase-0 Test-2, was gamma-irradiation of CsI solution in a stainless steel vessel at room temperature. Aqueous pH was controlled as a given condition. Practically no organic impurities were involved, and it was considered as a simple basic condition. Table 4.2 summarizes the experimental condition and some calculation parameters. The interface mass transfer and adsorption parameters for iodine were based on the values recommended in ISP-41. As indicated in the bottom row of the table, three cases of calculations were done. Figure 4.7 shows the controlled pH history that was given as input.

Figure 4.8 shows the evolution of gaseous iodine concentration by the calculation, compared with the experimental data. The base case calcuation agreed with the experiment very well.

Table 4.2: Experimental condition and calculation parameters for RTF Phase-0 Test-2 Stage-2 (ISP-41).

Experimental	
Temperature/Pressure	25 °C / Atmospheric
Cover gas	Air
Initial iodine conc.	9×10^{-6} M as CsI
Dose rate	1.36 kGy/h
pH control/range	Controlled, 10–5.5 (see Fig. 4.7)
Vessel / Geometry	316 stainless, Gas volume 315 L, Aq. volume 25 L, Gas-aq. interface 37 dm^2, Gas wall surface 220 dm^2, Aq. wall surface 52 dm^2
Calculation	
Adsorption parameters	Ads. velocity 1.8×10^{-2} dm/s, desorption rate 1.8×10^{-6} s^{-1}
Interface mass transfer	5×10^{-4} dm/s for I_2
Initial concentrations	$[I^-] = 9 \times 10^{-6}$ M, $[Fe^{2+}]=[Fe^{3+}] = 1 \times 10^{-5}$ M
Cases	"Base": as indicated above "NoFe": not assuming Fe^{2+}/Fe^{3+} existence "Ig": considering volatility of iodine atom (without Fe ions)

However, other cases, "NoFe" and "Ig", showed significant underestimation for the stages of high pHs. In the model, the iron ions work as catalysts that oxidize $\cdot O_2^-$ radical, a reducing agent for I_2 (Table 2.4), and enhance the oxidization of I^- to I_2. In our experiences, we needed to assume such catalysis effect by transition metals to obtain good simulation of RTF experiments at high pHs. Note that we did not use that in the simulation of JAEA experiment with a glass vessel.

The volatility of iodine atom examined in the "Ig" case was proposed by Dickinson and Sims[12]. They considered the similarity of $\cdot I$ with Xe and the importance of its volatility in the case I_2 concentration was very low. However, our calculation on RTF experiments showed that its effect is much weeker than the metal catalysis effect.

Thus, we suppose that consideration of the effect of metal catalysis should be included in the cases having metals in the system, *i.e.* experiments with steel vessels or simulation of real plants.

Figure 4.14 shows the evolution of aqueous iodine concentration. The base case showed a good agreement with the experiment. Figure 4.10 shows iodine distribution in the system. It is depicted that, in the experiment, most of iodine went to the gas phase wall and very little amount was in the gas phase at the end state. It is because I_2 is reactive and easily adsorbs on walls. Note that the adsorption of I_2 on the wall is handled empirically by given rate constants based on experimental observation, despite such quantitative importance, at present.

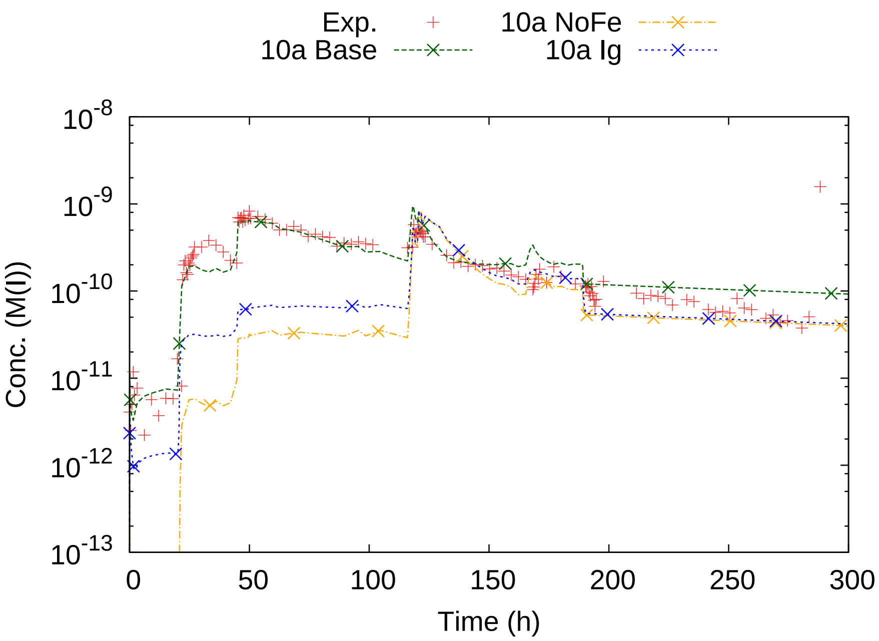


Figure 4.8: Gaseous iodine concentration: comparison of experimental data and calculations.

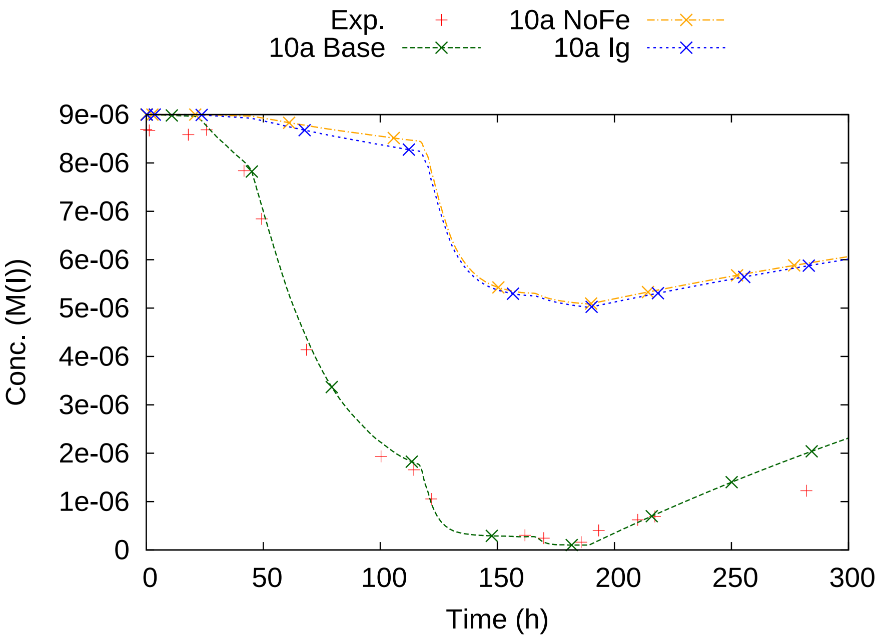


Figure 4.9: Aqueous iodine concentration: comparison of experimental data and calculations.

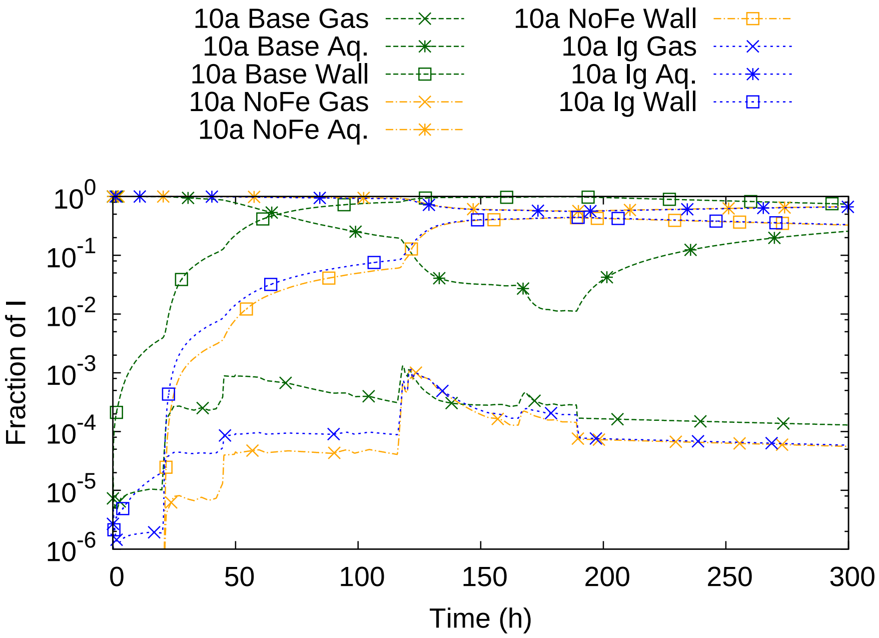


Figure 4.10: Iodine distribution in the system: comparison among the calculation results.

Table 4.3: Experimental condition and calculation parameters for RTF Phase-10 Test-1 (ISP-41 follow-up phase 2).

Experimental	
Temperature/Pressure	60 °C / Atmospheric
Cover gas	Air
Initial iodine conc.	8.6×10^{-6} M as CsI
Dose rate	0.67 kGy/h
pH control/range	Controlled at 10 before 72h and after 285h, otherwise not controlled
Vessel / Geometry	316 stainless, Gas volume 315 L, Aq. volume 25 L, Gas-aq. interface 37 dm^2, Gas wall surface 220 dm^2, Aq. wall surface 52 dm^2; Amerlock 400 epoxy paint for all the walls (xylene based, about 0.3mm thick, aged 3 months)
Calculation	
Adsorption parameters	For I_2: Ads. velocities $2 \times 10^{-3} \exp(5.1 \times 10^3(1/298 - 1/T))$ dm/s (gas, painted wall), 4×10^{-3} dm/s (aq., painted wall); Des. rates 0 (gas, painted wall), 1×10^{-6} s^{-1} (aq., painted wall) For I^-: Ads. velocity 4×10^{-7} dm/s (aq., painter wall); Des. rate 1×10^{-6} s^{-1} (aq., painted wall)
Interface mass transfer	$1/(1/k_{maq} + H_p/k_{mg})$ for I_2, HVRI and LVRI where $k_{maq} = 7 \times 10^{-4}(T/298)^{1.5}$ dm/s, $k_{mg} = 0.01(T/298)$ dm/s, H_p is the partition coefficient (C_{aq}/C_g)
Paint solvent leaching	JAEA model[37]
Initial concentrations	$[I^-]= 8.6 \times 10^{-6}$ M, $[Fe^{2+}]=[Fe^{3+}] = 1 \times 10^{-5}$ M, [orgpaq]= 0.25 M
Cases	"Base": as indicated above "NoFe": not assuming Fe^{2+}/Fe^{3+} existence "Cp0def": using default value for the initial organic conc. in paint = 0.08 M

4.3 RTF Experiment with a Painted Vessel

Another case of RTF experiment, Phase-10 Test-1 which was also referred in the follow-up studies of ISP-41[45], was simulated. Table 4.3 shows the experimental condition and some calculation parameter settings. The vessel wall was coated with an epoxy paint, and this case involved the influence of the organic impurity coming from the paint. The pH was initially set at 10 and the control was stopped at 72 h. After that, pH changed naturally by the reaction in the system.

The mechanism of the influence of organic impurity on the iodine chemistry is understood as follows[2, 3]. The paint matrix involves significant amount of remaining organic solvent that leaches when the paint is submerged under water. Under irradiation, such an organic impurity undergoes radiolytic degradation and provides organic radicals and acids. The acids lowers the pH of the solution and enhance the volatility of I_2. The organic radicals react with iodine and produce organic iodines.

The "10a" model of Kiche includes those processes, and this experiment was picked up for the test of those models.

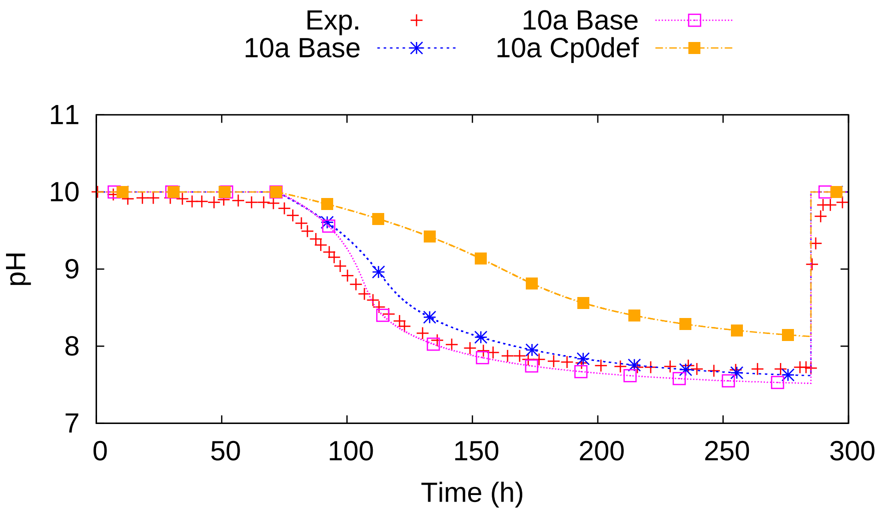


Figure 4.11: pH evolution: comparison of experimental data and calculation results for RTF Phase-10 Test-1 experiment.

In the calculation, parameters for the adsorption and interface mass transfer of iodine species were given according to the recommendation in the ISP document[45]. The initial concentration of iron ions was assumed as same as the previous section.

The leaching of the organic compound from the paint was handled by our own model which gives the equilibrium between the solution and the paint matrix, K, and the leaching rate, k_d (dm/s), by the following equations[37].

$$K = 3 \times 10^{-4} \exp\left\{-\frac{\Delta E_a}{R}\left(\frac{1}{T} - \frac{1}{298}\right)\right\},\ \Delta E_a = 55\ (\mathrm{kJ/mol})\ , \tag{4.1}$$

$$k_d = 4.17 \times 10^{-7} \exp\left\{-\frac{\Delta E_a}{R}\left(\frac{1}{T} - \frac{1}{298}\right)\right\},\ \Delta E_a = 50\ (\mathrm{kJ/mol}). \tag{4.2}$$

The initial concentration of the organic solvent in the paint matrix obtained from our experiment was around 0.08 M. However, it depends on the paint application process, age of the paint and so on, and has relatively large uncertainty. So, we tried several values for it and found that assuming 0.25 M gives a good agreement with the experimental data on the gaseous organic iodine concentration.

Figure 4.11 shows the calculated pH change compared with the experimental result. The selected base case parameters gave a good agreement on this pH drop behavior. The “NoFe” case showed more pH drop probably because more oxygen was available for more CO_2 production. The “Cp0def” case showed less pH drop due to less organic compound available for CO_2 production.

Figure 4.12 compares gaseous I_2 and organic iodines (HVRI+LVRI) concentrations by the base case calculation and the experimental data. The calculation showed a good agreement with the experimental data. Comparison among the calculations on the gaseous iodines is shown in Fig. 4.13. The “Cp0def” case showed less volatility than the base case though the trend was similar. The “NoFe” case showed much less volatility. Figure 4.14 shows the aqueous total iodine concentration. The base case calculation underestimated the aqueous concentration in the latter half. Figure 4.15 shows the concentration of organic solvent in the paint obtained in the base case calculation. It shows that the solvent remaining in the paint mostly leached during the experiment.

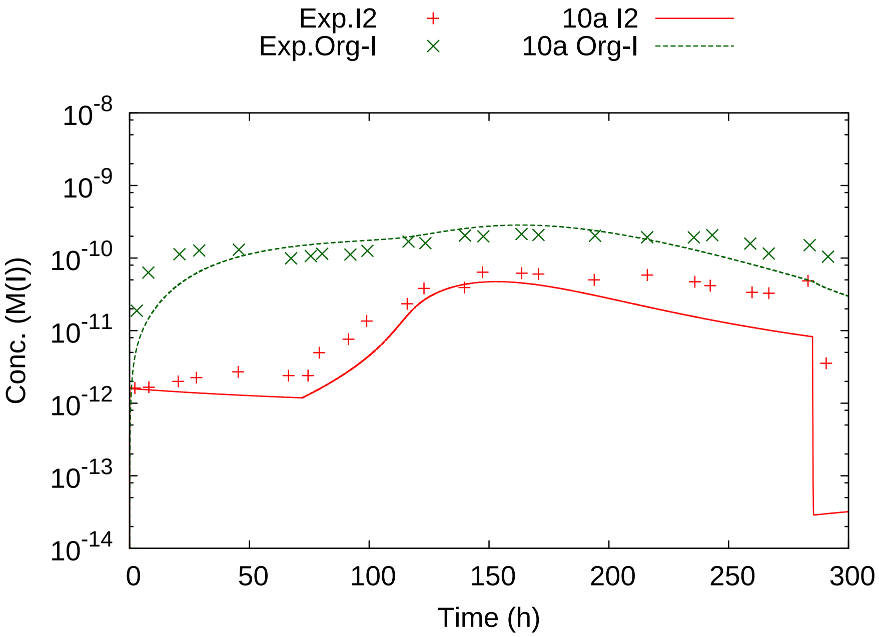


Figure 4.12: Gaseous iodine concentration: comparison of experimental data and calculation (base case).

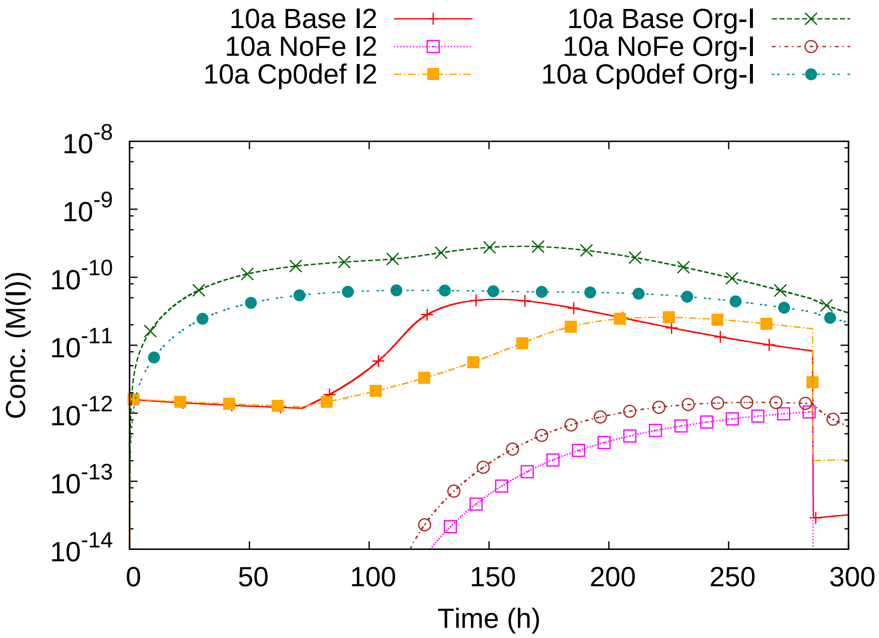


Figure 4.13: Gaseous iodine concentration: comparison of calculation results on the influence of parameters.

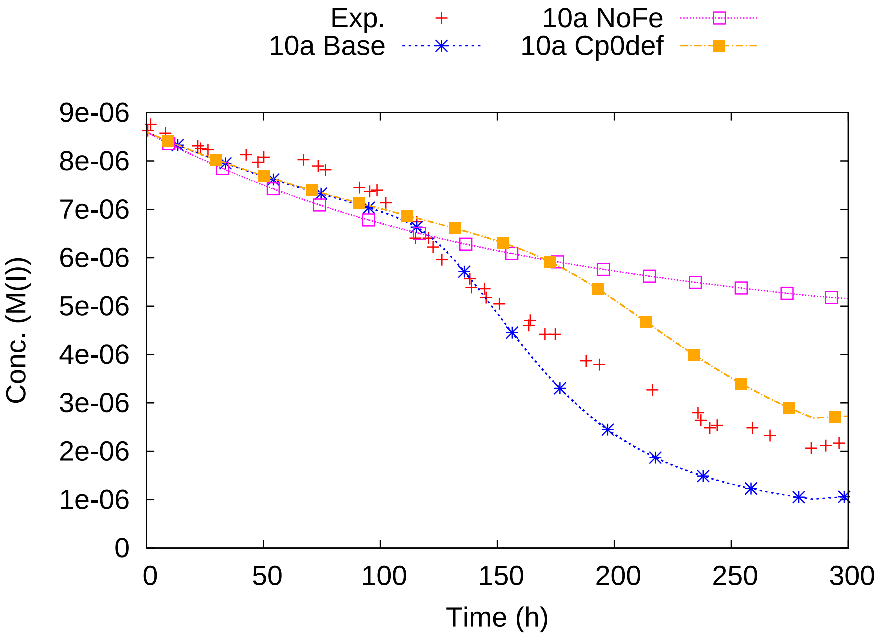


Figure 4.14: Aqueous iodine concentration: comparison of experimental data and calculations.

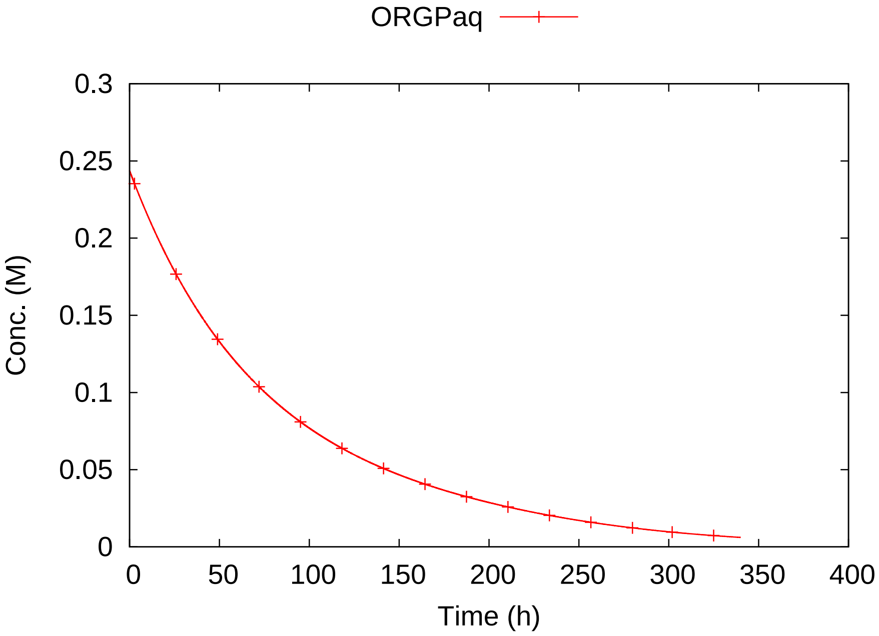


Figure 4.15: Calculation result on the decrease of organic solvent (ORGPaq) in the aqueous paint matrix by leaching.

5 Summary

A computer code Kiche was developed for analyses of chemical kinetics relevant to iodine volatilization in the containment vessel of LWRs during a severe accident. The model and code was described in this document. Application of Kiche for simulation of experiments, including JAEA experiments for the influence of organic impurity and oxygen concentration, RTF experiments with steel walls and with painted walls, demonstrated good agreement of the calculation results with those experimental data. It indicates that the present model covers the primary processes of the volatilization of iodine from irradiated iodide solutions with or without organic impurities.

Potential advantages of Kiche are as follows.

- The reaction database is in the plain text format and appears as reaction and rate constant expressions, so that users easily read or modify the reaction model.
- Being a Fortran code, it is relatively easily coupled with other Fortran codes such as severe accident analysis code.
- It solves "stiff" reaction kinetics models practically fast enough by the implicit scheme.

The present limitations are as follows.

- There are factors still not included in the present reaction model, "10a": contribution of nitrogen to nitrate or ammonia production, gas phase reactions by ozone or electron, gas phase organic iodine production on walls, etc.
- The reaction model is based on concentrations, and limited for dilute solutions. Concentrated solutions need more rigorous formulation with activities.
- The present code handles only one cell, though extension for multiple cells should be relatively easy.
- More sophisticated methods might be applicable for the discretization and matrix inversion, and might improve the accuracy and speed.

Acknowledgments

The development of Kiche code was done in a contract research program with Japan Nuclear Energy Safety Organization (JNES) in fiscal years 2007–2010. Authors appreciate a valuable communications with Mr. Glenn Glowa on the LIRIC reaction model.

Appendix A Syntax of the Reaction Database

A.1 Structure and Control Keywords

Figure A.1 gives an example of how the reaction database appears. The database should be written according to the syntax described in the followings so as to be processed by `chem2f95` script properly.

The database consists of 4 sections,

- Variable definition section,
- Reaction section,
- Non-reaction process section, and
- Iodine atom number section.

Those sections are started with control keywords, and ended by a blank line or the EOF (end of file) code. There are keywords other than the section declaration, namely for comments and variable definitions. Table A.1 summarizes the control keywords.

A.2 Syntax of Each Section

A.2.1 The "Variable Definition" Section

This section consists of space separated list of Fortran variable names. Those variables are defined as double precision real variables in the Fortran source to be generated. Basically, variables appearing in the ♯VAR expression must be listed so that they are defined in the source. However, some are already defined in the code (temperature, dose, adsorption and mass transfer related variables).

Calling of subprograms need their definition elsewhere. Typically, `misccorr.f95` is the place for definition of miscellaneous subprograms.

The variable definition section can not contain ♯VAR lines.

A.2.2 The "Reaction" Section

Assume a reaction named "reac1" is expressed as follows,

$$r_1R_1 + r_2R_2 + \cdots + r_{m_r}R_{m_r} \xrightarrow{k_r} p_1P_1 + p_2P_2 + \cdots + p_{n_r}P_{n_r} \tag{A.1}$$

where k_r is the rate constant of the reaction r (suffix); R and P are the reactant and product species, respectively; r_x and p_x are the corresponding stoichiometric coefficients, respectively. A line of the reaction database for this reaction is written like the following.

```
! reac_name      reactant              product              rate_const
 reac1        r1[R1]+...+rmr[Rmr]  p1[P1]+...+pnr[Pnr]   expression_for_kr
```

The syntax for this section is described as below.

- A reaction is defined by a line having 4 fields, "reaction name", "reactant", "product" and "rate constant".

```
========= 10a.db ==========
!
! kiche reaction database
....
#DEFVAR
tce tf ea dw rad kkw kk12 kk1 kk2 kkco2 kkhco3m
pkw pkh pkoh pkho2 kkh kkoh kkho2 kkh2o2 kk25 kk27 kk29 kk31
.......

#REACTION
! temprature vars
#VAR  tce=t-273.15 ! temperature t[k], tce[C]
.....
! generation of primary products by water radiolysis
! fit of Katsumura's review fig (CHARGEDPART04, chap.23, fig.1)
ge_m    {h2o}   [e_m]     rad*(2.57+3.31e-3*tce)
gh      {h2o}   [xh]      rad*(0.57+1.13e-3*tce)
.....
!
! Radiolytic oxidation-reduction
#VAR  kk12=(1.0/dw)*10.0**(555.0/t+7.355-2.575*log10(t)) ! Palmer et al.(1984) good for 277--483K
i_01    [i_m]+[xoh]          [xi]+[oh_m]          7.7e9*exp(19e3*tf)  !   DIC00_1
l2_14f  [xi]+[e_m]           [i_m]                2.4e10*ea
......

#NONREACT
!
.......
! i2
tri2_1  [i2g] [i2]     kmti2*aint/max(vg,epsvol)
tri2_2  [i2g] [i2g]   -kmti2*aint/max(vg,epsvol)*hpi2
tri2_3  [i2]  [i2g]    kmti2*aint/max(vaq,epsvol)*hpi2
tri2_4  [i2]  [i2]    -kmti2*aint/max(vaq,epsvol)
.......

#IATOMS
[orgimp]    0
[i2orgimp]  2
[orgip]     0
========= 10a.db ==========
```

Figure A.1: Appearance of the reaction database. (10a)

Table A.1: Control keywords for the reaction database.

Keywords for declaration of sections	
♯DEFVAR	Start of the "Variable definition" section.
♯REACTION	Start of the "Reaction" section.
♯NONREACT	Start of the "Non-reaction" section.
♯IATOMS	Start of the "Iodine atom number" section.
Keywords other than section declaration.	
!	Comments or comment lines. (Following characters to the end of the line are ignored.)
♯VAR	Assignment of variables. Fortran sentences to assign values to variables should follow. It can appear inside of "Reaction" or "Non-reaction" sections. It must appear before the reference of the variable.

One or more blank line(s) are recognized as end of sections. Comment lines has no effect. In the Fortran expression(s) following ♯VAR keyword, "K(*process name*)" is replaced with the rate constant for the reaction or non-reaction process of the specified name. Also "[*species name*]" is replaced with the concentration of the species. Any Fortran sentence can be inserted (*e.g.* if sentence, calling subprograms, etc.).

- One or more space(s) delimit the fields until the rate constant field is reached. Spaces are not significant within the rate constant field. Heading space(s) in this line are ignored.
- the reaction name is a string of a length less than 20 characters, and used for identification of the reaction.
- The reactant and product are strings to describe the left hand side and the right hand side of the reaction, which must be written without spaces within. "+" is used as delimiter between terms.
- "`[...]`" means species names. The solvent, or any species that should be ignored in the kinetic formula can be written by "`{...}`", *e.g.* `{h2o}`.
- The stoichiometric coefficients can be integers, real numbers (involving a decimal point) or fractions (1/2, 2/3 etc.).
- The rate constant expression should be valid as a Fortran mathematical expression. Variables can be used. Those variables should be defined in the ♯DEFVAR section, or known to be defined elsewhere. "K(*name*)" is replaced with the rate constant for reaction or non-reaction named *name*. "[*name*]" is replaced with the concentration of species *name*. ♯VAR directives should be used to give values of variables before they are referred.

A.2.3 The "Non-Reaction" Section

Various processes other than chemical reactions that make concentration changes of species, for example, gas-liquid interface mass transfer, adsorption/desorption on walls, mass transfer with flows and so on, are listed in this section. They are expressed as contributions to concentration changes of species. As an example, assume a process named "proc1" expressed as follows,

$$\frac{d[X]}{dt} = \cdots + k_1([X_1] - K[X]) + k_2[X_{2a}][X_{2b}] + k_3 + \cdots . \tag{A.2}$$

It is written in the database term by term as follows.

```
! non-reac_name     specs       affecting_specs    factor
 proc1a             [X]         [X1]              k1
 proc1b             [X]         [X]              -k1*K
 proc1c             [X]         [X2a][X2b]        k2
 proc1d             [X]         none              k3
```

The process "proc1" is decomposed into 4 terms and every term makes a line in the "Non-reaction" section. The syntax for this section is as follows.

- A non-reaction process line accommodates one of the decomposed terms that, as a whole, express contribution of a process to concentration change of a species. It consists of 3 fields, "non-reaction process name", "dependent species", "affecting species" and "factor".
- One or more space(s) delimit the fields until the rate constant field is reached. Spaces are not significant within the rate constant field. Heading space(s) in this line are ignored.
- The non-reaction process name is a string of a length less than 20 characters, and used for identification of the process (and the term).
- The dependent and affecting species names are expressed by "[*name*]".
- "none" as the affecting species means a 0th-order process, the rate constant of which does not depend on any concentrations.
- The factor is given by a Fortran mathematical expression. Variables, "K(*name*)" and "[*name*]" syntax are the same as in the "Reaction" section.

A.2.4 The "Iodine Atom Number" Section

The number of iodine atoms in a species (molecule) is, by default, recognized by reading a number following a letter "i" in the species name (e.g. "i2", "i3_m"). If no number is found after "i", the number is set 1. This section exists to give the number that is not found by this default method. For example, "x(hio2)3" has 3 iodine atoms and the above method is not valid in this case. A line in this section as follows specifies explicitly the number of iodine atoms in such a case.

```
! species_name       number_of_i_atoms
[x(hio2)3]             3
```

A.2.5 Line Length

The 132-character limitation of Fortran free form source is imposed on the reaction database. So, the length of the significant part of lines in the database should not exceed about 120 characters to make the generated Fortran source fit in the source format limitation. Comments are erased at conversion and not subject to this requirement. A very long variable expression can be folded as a Fortran multiple-line sentence.

```
#VAR xabc = very * long * 3.14159265358979 * 1./234.56789 * cos(33.123/54.345) &
#VAR     & + sin(45.65789*34.1234/1234.4567)/log(213.87163)/3.1234567 &
#VAR     & - expression * is / folded / like / this
```

When the conversion is done automatically by the Makefile, the line length of the generated source is checked, and an error message is shown if too long lines are found.

Table A.2: Species and components names hard-coded in kiche.

Name	Species or components
i2g	Gaseous I_2.
xig	Gaseous $\cdot$I.
hvrig	Gaseous HVRI (high volatility organic iodide, namely CH_3I and others).
lvrig	Gaseous LVRI (low volatility organic iodide).
i2gout, xigout, hvrigout, lvrigout	Gaseous I_2, $\cdot$I, HVRI and LVRI flowed out of the system.
issg	Iodine adsorbed on gaseous steel wall.
ispg	Iodine adsorbed on gaseous painted wall.
isp	Iodine adsorbed on aqueous painted wall.
imsp	Iodide ion adsorbed on aqueous painted wall.
i2con	Iodine absorbed in gaseous wall condensate water film.
h_p	H^+
o2g	Gaseous O_2.
n2g	Gaseous N_2.
h2g	Gaseous H_2.
co2g	Gaseous CO_2.
nh3g	Gaseous NH_3.
orgpaq	Organic solvent remaining in aqueous paint matrix.

A.2.6 Convention for Naming Chemical Species or Components

Though any naming style is possible as long as each name is identifiable, the following convention is recommended for the naming of chemical species for it is simple and it makes the database easy to be read.

- Use the molecular forumula with lower case alphabets, numbers, parenthesis "(..)" and under bar "_".
- For plus and minus ionic charges, put "_p" and "_m", respectively, at the tail, *e.g.* "h_p", "co3_2m", "i_3m".
- Put a heading "x" for radical species, *e.g.* "xoh", "xo2_m".
- Put a tailing "g" for gaseous species, *e.g.* "i2g", "n2g".

Some examples are: Fe^{3+} ⇒ fe_3p , e^- ⇒ e_m , $\cdot IO_3^{2-}$ ⇒ xio3_2m , NH_4^+ ⇒ nh4_p , $B(OH)_3$ ⇒ b(oh)3 .

There are some species for which names are presumed and hard-coded in kiche. They are iodine species, gas components and organic compounds in the paint matrix as listed in Table A.2. Those names should be used in the database consistently.

Appendix B Practice of Usage

B.1 System Requirements

Kiche has been developed on Linux workstations with the environment as follows.

- Hardware: Intel CPU later than Pentium 4, Reasonable RAM and HDD capacity to run the OS with X-Window
- Linux distribution: Vine Linux 4.2, Ubuntu 9.04 and 10.04LTS
- Perl interpreter: Perl v.5
- Shell: Bash 4.1.5
- Fortran 95 compiler: Absoft Fortran 9.0 or g95
- Plotter: Gnuplot 4.2

The command line appearances shown below are in this environment. Among the list above, a Fortran 90/95 compiler and a Perl interpreter suffices the least requirement, and any system having them are probably able to run the Kiche tool.

B.2 Source Directory Tree

When the archive file for distribution is extracted, the following directory tree appears. It is recommended to extract an archive in a directory such as `$HOME/kiche`.

```
. --+-- README.txt Copyright notice etc
    |   NOTE.txt   Note on the development
    |   VERSION    Version
    |   mktar      Archiving script
    |
    +-- src        Source files
    |
    +-- bin        Execution files and scripts (Add this to your user PATH)
    |              chem2f95 and chem2tex tools are here.
    |
    +-- models     Reaction databases
    |
    +-- doc        Documents
```

B.3 Compiling

Assume you are in the directory where the archive is extracted. The following command does the compilation with a reaction database file “xxxx.db”.

```
$ cd src
$ make MODEL=xxxx
```

The execution module is in the “bin” directory and named “kiche.xxxx”. The compilation process runs by the make command, by default reading “Makefile”. If the option specification “MODEL=...” is omitted, a default model specified in the Makefile is used. The following does only the generation of Fortran include files from the reaction database.

```
$ cd src
$ make model MODEL=xxxx
```

To clean the object files and module files, or the execution files are done by the following.

```
$ make clean      (to erase object and module files)
$ make binclean   (to erase execution files)
```

The reaction database files to be used should be in “models” directory. As of October 24, 2010, following reaction database files are maintained.

- 10a : The latest database compiled at JAEA, based on water radiolysis set by Elliot (1994)[20], iodine reaction set based on LIRIC 3.2[11] and others[12, 9] with improvement for low oxygen concentration conditions at JAEA.
- liric : Nearly the copy of LIRIC 3.2 set[11].
- 10a-lorg: Same as 10a except that the organic reaction set is the same as LIRIC 3.2.

B.4 Running the Code

A brief instruction on the execution of `kiche` is presented by calling it without command line options or with “-help” option. The following is the output by that.

```
$ kiche.org -help
kiche: chemistry of iodine in LWR containment
 version=1.2 model=10a
 usage: kiche.<model> [options]
 options:
 -i  <file> : input file (default 'in')
 -h  <file> : output for iodine inventory history (default 'hi')
 -c  <file> : output for species concencrations history (default 'hc')
 -hr <file> : output for rate constants history (default none)
 -lr <file> : output for change rate and contributions (list) (default none)
 -bd <file> : binary dump data (for restart) (default none)
 -rs <file> <step> : restart calc. (get cell and time data from a binary
     dump file and start the calculation from there.) (default none)
     kiche.<model> -rs <file> shows a list of time and step in the file.
 -db        : output debug information into file 'debug'
 -help      : print this message (to stderr) and stop
 -index     : print indexes of chemistry model (to stdout) and stop
 -ispc <name> : print index for a species and stop
 -oisp <name> : print output column index for a species (index+2) and stop
(if no option is given and no default input file is found, this message
 is printed and program stops.)
```

As in the above, The chemistry model information, *i.e.* species, reactions and non-reaction processes included, and their indexes, are displayed by the option “-index”.

```
$ kiche.xxx -index | less
```

This displays the index list by a pager “less”. With the “-ispc” option, the index of the specified species is displayed. The “-oisp” option displays practically “species index+2” that is the column number in the concentration history output file. These functions are useful for finding the species indexes for preparation of the input data file, for plotting the history data, and so on. The followings are examples of execution by those options.

```
$ kiche.10a -ispc h_p
16
$ kiche.10a -oisp h_p
18
```

It can be used in the Unix shell script (Bourn shell) or gnuplot script as follows.

```
(In the shell)
$ echo ‘kiche.10a -oisp h_p‘
18
(In gnuplot)
gnuplot> plot "hc" us 1:(-log10(column(‘kiche.10a -oisp h_p‘))) ti "pH" w l
... plot is displayed ...
```

B.5 Preprocessor to Ease Indexes in the Input

Concentrations of some species should be given as initial conditions by input data file for `kiche`. If modification of some rate constants are intended, modification factors for those reactions or non-reaction processes should be given in the input. Appendix C describes about such matters in detail, and just one point is given here.

The specification of species, reaction or non-reaction processes in the input file is given by their integer indexes, not the names defined in the reaction database. This is due to the syntax allowed for the “namelist” function of Fortran. However, this causes an inconvenience in preparing input files. That is, we need to prepare input files for different reaction models with different index numbers if the indexes are different, even though the difference is very little. To make this situation easier, a preprocessor script was prepared. It finds the names of species and processes in the input file, converts them into integer index numbers and writes an input file directly readable by `kiche` code, and runs `kiche` code with given reaction model (kiche.*model*). The name of the script is `kiche` (without suffix for a specific reaction database). The usage of the `kiche` preprocessor is shown by calling it without options or with -h.

```
$ kiche -h
usage: kiche <model> [command line options]
 this is a wrapper to run kiche.xxx with the chosen reaction model.
preprocessing:
 replace of species, reaction/non-reaction names with corresponding
 numerical indexes is available in
   - initial conc. specification (e.g. c%conc(i_m) : i_m => ##)
   - rate const. modification factors (e.g. fkr(i_21) : i_21 => ##)
   - material addition time table (e.g. ttadd(1)%isp = org : org => ##)
 * lines headed by ! (preceding blanks are ignored) are not pre-processed.
 ---- help message of kiche ----
....
```

Appendix C Input/Output Data Files

C.1 Input Data Format

C.1.1 The Style

The input data is fed by using the “namelist” function of Fortran. A namelist `inputvars` is defined in a subroutine `initinput`, and an input file appears like below.

```
!
! comments
!
&inputvars

....
/
```

The data for the geometry and conditions of the system cell is contained in a derived type variable “`c`”. The cell data variable is defined as a local variable “`cell`” in `main.f95`, then passed to `initinput` as an argument. The elements of the cell data are accessed by “c%xxx” in the input data. Other variables are defined in `vars.f95` or `vars_chem.f95`.

Samples of input data are shown at the end of this section in Figs. C.1 and C.2, for JAEA experiment (“Base” case, see Section 4.1) and for RTF Phase-0 Test-2 Stage-2 (“Base” case, see Section 4.2), respectively.

C.1.2 Input Variables

The units for the data are as follows. Those units are used also inside the code.

- Length, area, volume: dm, dm^2, dm^3
- Time: s
- Mass: kg
- Force: N
- Energy: J
- Molarities and concentrations: mol, mol/dm^3(=M), mol/dm^2, mol/dm
- Dose rate: kGy/h

The followings are the list of input variables. Default values are indicated by (default ♯) or *.

Control variables Control variables are as follows.

tstart	Start time (default 0s)
tend	End time (default 0s)
maxstep	Maximum step number (default 5000000)

dtmax	Upper limit time step (default 60s)
dtmin	Lower limit time step (default 1e-9s) (The program terminates with error if matrix inversion fails with it)
outdt	Output time step (default 600s)
outstep	Output step (It overrides if it is reached before the output time step is reached.)(default 1000)
iphgiven	Switch for pH handling: 0 pH is calculated (H^+ conc.), 1 pH is given as input data by ttbl_ph (If time dependent ON/OFF of pH control is desired, ttbl_phctrl should be set.)
igasrule	Switch for cover gas composition handling (except volatile iodine that is dilute): 0 Closed system (Constant mass of gas components), 1 Open system (Constant partial pressure of gas components; (Volatile iodines are closed in the system if not instructed differently (see qgflow)) (Initial conditions are given by partial pressures in either of the settings)
ihumid	Switch for humidity (partial pressure of water) handling: 0 Relative humidity is given as constant ($x_{hm} = p_s/p_{sat}$), 1 Saturation vapor pressure is assumed ($p_s = p_{sat}, x_{hm} = 1$)
iwallcond	Switch for the wall condensate water film: 0 Condensation mass flux is given (kg/dm^{2}s), 1 Wall temperature is given (K)
ioutstep	Switch for output time step control: 0 Output according only to outdt or ttbl_outdt, 1 outstep is also referred (to prevent silence when time step reduces to very short, taking long time to reach next output time)
mdlctrl	String variable for model control keywords (Multiple keywords can be set with spaces as a delimiter; Keywords are listed later in “Control keywords”.)
epsnewton	Newton conversion criteria, relative error tolerance (default 1e-4)
epsneg1	Newton conversion criteria, neglected small concentration level (default 1e-20M)
epsneg2	Newton conversion criteria, a relative criterion to neglect negative concentrations (default 1e-4)
iitnewton	Switch for Newton iteration: 0 No iteration (Simplified method, fast but less accurate), 1 With iteration (default)
ngoodnewton	Number of Newton iteration for convergence that is taken as fast enough and increase of time step is tried (default 10)
nmaxnewton	Upper limit number of Newton iteration with which convergence is quit and retry with reduced time step is decided (default 50)

Control keywords The model behavior is controlled by setting keywords in a string input variable `mdlctrl`. Multiple keywords can be given by delimiting them by speces. Care should be taken in conflicts or inconsistency in multiple keywords.

givekmt : kmtxxx(gas-aqueous interface mass transfer velocities) are given from input as constants and built-in correlation equations are not used. (for kmto2/n2/h2/co2/nh3/org/i2/xi/hvri/lvri/i2con) (This setting overrides others, *e.g.* “isp41...”, “nogaspart” and so on.)

kmto2base : Relative evaluation of gas-aqueous interface mass transfer velocities based on the value for O_2 at 25°C is used. The basis value should be given by kmt0o2. The values for other volatile species are evaluated by assuming a dependence law for the case of liquid phase resistance dominant, $k_{mt} \sim (T/\mu)/V_A^{0.6}$; T: temperature (K), μ: water viscosity, V_A: molar volume of the volatile species at normal boiling point.

nogaqpart : Gas-aqueous partition is prohibited.

givesorption : Wall adsorption/desorption parameters are given from input as constant and built-in correlation equations are not used. (for vadissg,kdeissg,mlimissg; vadispg,kdeispg,mlimispg,mirvispg; vadisp,kdeisp,mlimisp,mirvisp; vadimsp,kdeimsp,mlimimsp) (This setting overrides others, *e.g.* “isp41...”, “nosorption” and so on.)

nosorption : Wall adsorption/desorption is prohibited.

nopaintdissol : Organic compound leaching from the aqueous wall paint is prohibited. (default: leaching model enabled if aspaq> 0 and tpaq> 0)

givepaintcp0 : Initial organic concentration in the aqueous wall paint is given from input (by c%conc(orgpaq)). (for organic leaching model) (default: evaluated by a built-in empirical correlation equation)

givepaintdissol : Rate constant for organic leaching from the aqueous wall paint, kdisorg (dm/s), and equilibrium constant kkdisorg(=$([\mathrm{org}]/[\mathrm{orgpaq}])_{eq}$), are given from input as constants, and the built-in correlation equations are not used. (default: evaluated by built-in empirical correlation equations)

givewallcond : Wall condensate film thickness and flow rate are given from input as constants and built-in correlation equations are not used. (for c%tcon, c%fcon) (default: evaluated by built-in empirical correlation equations)

usexig : Atomic I volatility is considered. (Based on Dickinson[12] model) (default: not considered)

useih2 : H_2+I reaction is considered. (Reduction of iodine volatility under the presence of hydrogen gas (upto 5% at atmospheric pressure and room temperature) can be simulated. Not confirmed at other conditions.) (default: not considered)

orggout : Flow-out of gaseous organic compound, ORGg, is considered if qgflow> 0. (default: flow-out is considered only for predifined gases and volatile iodines)

isp41rtfset : ISP41 (RTF Phase-0 Test-2) recommended parameter values (gas-aqueous interface mass transfer, wall adsorption/desorption) are used.

isp41fu1set : ISP41 follow-up phase-1 common parameter values for a plant like condition (gas-aqueous interface mass transfer, wall adsorption/desorption, wall condensate film) are used.

isp41fu2phebusrtf1set

: ISP41 follow-up phase-2 Phebus/RTF1 common parameter values (gas-aqueous interface mass transfer, wall adsorption/desorption) are used.

isp41fu2rtfp10t1set

: ISP41 follow-up phase-2 RTF Phase-10 Test-1 common parameter values (gas-aqueous interface mass transfer, wall adsorption/desorption) are used.

Time table data Time tables are derived type variables with following elements to specify data at given points of time.

n : Number of data (frequency of specification) (≤max_ttbl= 500)(default 0)

t() : Array for the times (n elements)

x() : Array for the values (n elements)

ihr : Switch for time units (0: second, 1: hour)(default 0)

typ : Switch for interpolation method (0: linear interpolation between the given points, 1: step function changing to the values at given points) (default 0 for (ttbl_ph, ttbl_temp,ttbl_tempg,ttbl_press), 1 for (ttbl_phctrl, ttbl_dtmax, ttbl_outdt, ttbl_dose, ttbl_qgflow))

These elements are accessed in the input data by ttbl_xxx%n, ttbl_xxx%t(..), ttbl_xxx%x(..) and so on. The following is the time tables provided.

ttbl_ph	pH time table (required when iphgiven= 1)
ttbl_phctrl	ON/OFF of pH control (referred when iphgiven= 1, x= 0: OFF, x= 1: ON) (Not giving this means controlling pH always.)
ttbl_dose	dose rate (kGy/h) time table, referred if n> 0 (c%dose is referred if n= 0)
ttbl_temp	Aqueous temperature (K) time table, referred if n> 0 (c%temp is referred if n= 0)
ttbl_tempg	Gas temperature (K) time table, referred if n> 0 (c%tempg is referred if n= 0)
ttbl_press	Pressure (Pa) time table, referred if n> 0 (c%ptot is referred if n= 0) (At present not implemented)
ttbl_dtmax	dtmax time table, referred if n> 0 (dtmax is referred if n= 0)
ttbl_outdt	outdt time table, referred if n> 0 (outdt is referred if n= 0)
ttbl_qgflow	qgflow time table, referred if n> 0 (qgflow is referred if n= 0)

Material addition time table data A time table input variable is specially provided for addition of materials into the system with time. It is a derived type variable with following elements.

isp : Index of the species added

n : Number of data (frequency of specification) ($\leq$max_ttbl= 500)(default 0)

t() : Array for the times (n elements)

x() : Array for the values of addition amount or rate depending on typ (n elements)

ihr : Switch for time units (0: second, 1: hour) (default 0) (applied also for the addition rate, e.g. M/h for ihr= 1)

typ : 0: Amount (concentration of the species added) increment at specified time, 1: Rate (concentration increase rate) given at specified time and kept constant till the next point

Multiple number of material addition time tables can be used.

nttadd	Number of material addition time tables used ($\leq$ maxttadd= 10)
ttadd(1:nttadd)	Time table data for addition of materials at specified points of time; the elements of each table should be specified.

Cell attributes A cell data variable is defined for the input of cell attributes.

c	Cell attributes (default 0 for all elements)

It is a derived type variable with the following elements.

dose : Dose rate (kGy/h)

temp, tempg : Aqueous and gas temperatures (K)

xhm : Relative humidity(-) ($= p_{st}/p_{sat}$)

ptot : Total pressure (Pa)

psat : Saturation vapor pressure of water at temperature tempg (Pa)

pst : Partial pressure of water vapor (Pa)

pgas : Partial pressure of non-condensible gases (sum of partial pressures of gas elements) (Pa)

vg, vaq : Gas and aqueous phase volumes (dm^3)

aint : Gas-aqueous interface area (dm^2)

assg, aspg : Gaseous steel and paint wall areas (dm^2) (for adsorption)

aspaq : Aqueous paint wall area (dm^2) (for adsorption)

agepaq : Age of aqueous paint age (day) (for evaluation of remaining organic solvents in the paint)

ufg,ufaq : Flow velocities along gaseous and aqueous walls (dm/s) (for evaluation of adsorption)

wlsg,wlpg : Lengths of gaseous steel and painted walls (dm) (for evaluation of adsorption) wlpaq : Length of aqueous painted wall (dm) (for evaluation of adsorption) (The effect of uf*, wl* is described in vadfluid_g/aq in misccorr.f95. A larger uf* or smaller wl* makes larger adsorption velocities. For the real scale, uf*= 1, wl*= 20 likely to be conservative (modest adsorption). Unless a control keyword “givesorption” is specified, wl*, uf* should be specified for walls of non-zero area.)

tpg : Gaseous paint thickness (dm) (for evaluation of adsorption)

tpaq : Aqueous paint thickness (dm) (for evaluation of adsorption and organic compound leaching)

vspg, vspaq : Gaseous, aqueous paint volume (dm^3) [input not taken]

acon : Surface area of the wall condensate water film (dm^2)

hcon : Hight of the wall condensate water film (vertical length) (dm)

tcon : Average thickness of the wall condensate water film (dm)

fcon : Flow rate of the wall condensate water film (flow into the pool) (dm^3/s)

vcon : Volume of the wall condensate water film (dm^3) [input not taken]

ucon : Average down-flow velocity of the wall condensate water film (dm/s) [input not taken]

mcon : Condensation mass flux onto the wall condensate water film ($kg/s\cdot dm^2$)

twcon : Wall temperature under the wall condensate water film (K)

conc() : Concentration of species (mol/dm^3, mol/dm^2 or mol)

gas : A variable for the cover gas composition (see below)

typ : Cell type:
0: Normal, 1: No gas phase (vg<epsvol), 2 :No aqueous phase (vaq<epsvol)

The element for the gas composition, “gas”, is a derived type variable and has the following elements.

po2, pn2, ph2, pco2, pnh3, pother : Partial pressures (Pa) of O_2, N_2, H_2, CO_2, NH_3 and others

xo2, xn2, xh2, xco2, xnh3, xother : Molar fractions of O_2, N_2, H_2, CO_2, NH_3 and others

These gas elements are accessed by c%gas%po2, c%gas%xhn3 and so on.

Chemistry model variables The variables to control the chemistry model or used for it are as follows.

fkr()	Modification factors for reaction rate constants (default 1.0)
fknr()	Modification factors for rate constants of non-reaction processes (default 1.0)

Array indexes for them are to be given by integer indexes that can be checked by “`kiche.xxx -index`”.

The following variables are by default calculated by built-in correlation equations. They can be given from the input as constants by setting relevant keywords in the control string variable `mdlctrl`.

vadissg	I_2 adsorption velocity on to the gaseous steel wall (dm/s)
kdeissg	I_2 desorption rate from the gaseous steel wall (1/s)
mlimissg	Upper limit adsorption density of I_2 on the gaseous steel wall (mol/dm^2)
vadispg	I_2 adsorption velocity on to the gaseous painted wall (dm/s)
kdeispg	I_2 desorption rate from the gaseous painted wall (1/s)
mlimispg	Upper limit adsorption density of I_2 on the gaseous painted wall (mol/dm^2)
mirvispg	Irreversible adsorption capacity of I_2 on the gaseous painted wall (mol/dm^2)
vadisp	I_2 adsorption velocity on to the aqueous painted wall (dm/s)
kdeisp	I_2 desorption rate from the aqueous painted wall (1/s)
mlimisp	Upper limit adsorption density of I_2 on the aqueous painted wall (mol/dm^2)
mirvisp	Irreversible adsorption capacity of I_2 on the aqueous painted wall (mol/dm^2)
vadimsp	I^- adsorption velocity on to the aqueous painted wall (dm/s)
kdeimsp	I^- desorption rate from the aqueous painted wall (1/s)
mlimimsp	Upper limit adsorption density of I^- on the aqueous painter wall (mol/dm^2)
kmti2	I_2 gas-aqueous interface mass transfer velocity (dm/s)
kmthvri	HVRI gas-aqueous interface mass transfer velocity (dm/s)
kmtlvri	LVRI gas-aqueous interface mass transfer velocity (dm/s)
kmti2con	gas-aqueous interface mass transfer velocity of I_2 on the wall condensate film surface (dm/s)
kmto2	O_2 gas-aqueous interface mass transfer velocity (dm/s)
kmth2	H_2 gas-aqueous interface mass transfer velocity (dm/s)
kmtn2	N_2 gas-aqueous interface mass transfer velocity (dm/s)
kmtco2	CO_2 gas-aqueous interface mass transfer velocity (dm/s)
kmtnh3	NH_3 gas-aqueous interface mass transfer velocity (dm/s)
kmtcl2	Cl_2 gas-aqueous interface mass transfer velocity (dm/s)

kmtorg	Gas-aqueous interface mass transfer velocity of the organic compound, ORG (dm/s)
kdisorg	Rate constant for organics leaching from paints (dm/s)
kkdisorg	Equilibrium constant for organics leaching from paints (=([org]/[orgpaq])$_{eq}$)
hporgspc	Selection switch of the gas-aqueous partition coefficient for the organic compound (string): "mibk" (default) or "xylene"

* The concentrations of iodine adsorbed on walls (chemical form is unknown) are expressed by I atom molarity (mol-I/dm^2). Thus, care should be taken for the adsorption/desorption rate constants about the definition of the concentrations of both sides, on the wall and in the adjacent bulk phase.

Parameters for O_2 basis relative evaluation method for gas-aqueous mass transfer velocities, referred when the keyword "kmto2base" is set in "`mdlctrl`"

kmt0o2	Gas-aqueous interface mass transfer velocity for O_2 at 298 K (25°C) (dm/s) as a basis
fkmtiod	Modification factor for the gas-aqueous mass transfer velocities for volatile iodine species (taking effect only on the iodine species)

Gas flow parameters for open systems (sparge/sweep)

qgflow	Gas flow rate (dm^3/s) (The flow-in equals the flow-out.) (default 0)
kkbi2	Ratio of I_2 concentration to the equilibrium value (0–1) inside bubbles (default 0)
kkbhvri	Ratio of HVRI concentration to the equilibrium value (0–1) inside bubbles (default 0)
kkblvri	Ratio of LVRI concentration to the equilibrium value (0–1) inside bubbles (default 0)

Sparge (bubbling) systems need qgflow and kkbxxx> 0. Sweep (no-bubbling) systems need qgflow.

C.2 Tips and Examples for Input Preparation

Inputs for derived type variables (structures) and array variables

- Elements of derived type variables: "*variable*%*element*= ..."
- Elements of array variables: "*variable*(*range*) = `###  ###  ###`"

```
! Time table data
   ttbl_ph%n = 3
   ttbl_ph%t(1:3) = 0 100 200 ! Range given explicitly
   ttbl_ph%x = 10 8.5 6   ! Range can be omitted.
   ttbl_ph%ihr = 0
```

```
! Material addition time table data
    nttadd = 1        ! Number of ttadd() elements (number of tables)
    ttadd(1)%isp = 7 ! Integer index of species are given.
    ttadd(1)%n = 3
    ttadd(1)%t = 10 20 30
    ttadd(1)%x = 1e-4 1e-4 1e-4 ! Amount for 1e-4M increase is added in every 10s.
    ttadd(1)%ihr = 0
```

Giving initial concentrations and modifying rate constants `kiche` code reads as namelist inputs the following format. So, an input file directly fed to the code should be written by this style.

```
c%conc(33) = 1e-4 ! Concentration of a species
fkr(52) = 0.1      ! Factor 0.1 for a reaction
```

The integer indexes for species and reaction/non-reaction processes can be checked by the following. (for the detail, see Appendix B)

```
$ kiche.xxx -index
```

However, a preprocessor to convert string names of species and processes into integer indexes is available. The input file written by string names are converted as follows.

```
$ cat in                # displaying the content of a file, "in"
...
c%conc(o2) = 2.67e-5 ! O2 saturation at 25C
c%conc(i_m) = 1e-4   ! I-
c%conc(org) = 1e-3   ! org
...
...

$ cat in.pp             # "in.pp" is the file after conversion
...
c%conc(46) = 2.67e-5 ! O2 saturation at 25C
c%conc(33) = 1e-4   ! I-
c%conc(49) = 1e-3   ! org
...
```

The detail about the preprocessor is given in Appendix B.

Wall adsorption Models are provided to the followings.

- I_2 adsorption on to gaseous steel walls
- I_2 adsorption on to gaseous painted walls
- I_2 adsorption on to aqueous painted walls
- I^- adsorption on to aqueous painted walls

They take effects when corresponding wall area, and the paint thickness in the case of adsorption on to the paint (c%assg, c%aspg, c%aspaq, c%tpg, c%tpaq) are given non-zero values. The adsorption/desorption velocities, upper limit of the adsorption densities, irreversible adsorption capacities can be evaluated by built-in correlation equations based on the data by Rosenberg(1969)[39]. The adsorption/desorption parameters can be given as constants by setting a keyword “givesorption” in the string `mdlctrl`.

Organic solvent leaching from aqueous paints A built-in model was developed based on a leaching experiment with an epoxy paint[37] and it also showed consistency with other data by Ball et al.[38]. The model needs input for aqueous painted wall area, paint thickness and paint age. The initial organic solvent concentration in the paint matrix that is by default evaluated by an empirical correlation for remaining xylene depending on the age (not generally reliable), can be given as a constant from input by setting a keyword "givepaintcp0" in `mdlctrl`. When "givepaintdissol" is set in `mdlctrl`, the solvent leaching rate constant and equilibrium constant, kdisorg and kkdisorg, are to given from input. When "nopaintdissol" is set in `mdlctrl`, the organic solvent leaching is not considered.

Time step control

- Fixed steps: Give dtmax and outdt.
- Steps variable with time: Give time tables for ttbl_dtmax, ttbl_outdt.

These time tables, by default having typ= 1 attribute, keep the value given at a point of time till the next point. The variable time step is necessary to solve for a long period, starting from rapidly changing initial phase and slow long term, or a process including rapid changes within long slow process.

Cover gas composition

- Closed system (mass of components are conserved) : Set igasrule= 0 and give initial partial pressures of components. The molarities of gas components are calculated at the start and conserved (with changes by chemical reactions and mass transfer between phases).
- Open system with ventilation (constant cover gas composition) : Set igasrule= 1 and give initial partial pressures of components. The molarities of gas components are calculated from the partial pressures at every time step. Mass is not conserved. (Gaseous iodine species are not treated by this.)

Simulation of experiments with swept system

- Sparge (bubbling) : Give qgflow, kkbi2, kkbhvri, kkblvri: qgflow (volume flow rate of the gas) from the experimental condition, kkbxxx measured at the specific system, or assume 1.
- Sweep (no-bubbling) : Give qgflow and leave kkb... = 0 (as default). The gas-aqueous mass transfer velocities for gas components should be adequately given according to the system.

Condensate water film flowing down along the wall The condensate water film is considered when the gaseous wall area wet by the condensate, c%acon, and the height of it, c%hcon, are given.

- By default, the water film thickness and flow rate are calculated by built-in correlation equations with either of the following condition: iwallcond= 0 (given condensation mass flux, c%mcon (kg/dm^{2}s)) or iwallcond= 1 (given wall temperature, c%twcon (K)) (c%tcon, c%fcon, c%vcon, c%ucon are calculated.)
- When a keyword "givewallcond" is set in `mdlctrl`, the built-in model is not used, c%tcon and c%fcon are read from the input as constants.

C.3 Output Data Format

History of iodine inventories (default: hi)

```
# kiche-1.1 with model=10a
# history of iodine inventory (time=s, inventory=mol-I atom)
#  1:time    2:step |   3:mitot      4:mig        5:miaq       6:misg ...
   0.000000         0   2.2500E-04   0.0000E+00   2.2500E-04   0.0000E+00 ...
   1.032590       363   2.2500E-04   9.9740E-13   2.2500E-04   2.1561E-15 ...
   2.044457       377   2.2500E-04   3.8961E-12   2.2500E-04   1.6730E-14 ...
   3.044457       387   2.2500E-04   8.7115E-12   2.2500E-04   5.6863E-14 ...
...
```

The variables output in this file is as follows. The unit of the iodine mass is mol-I.

mitot	Total iodine mass (in the system+flowed out)
mig	Gaseous iodine mass
miaq	Aqueous iodine mass
misg	Gaseous wall (including what is dissolved in the wall condensate)
misaq	Aqueous wall
mi2g	Gaseous I_2
mxig	Gaseous atomic iodine (considered by Dickinson et al.(2000)[12])
mrig	Gaseous organic iodines
migout	Total gaseous iodine flowed out
mi2gout	Gaseous I_2 flowed out
mrigout	Gaseous organic iodines flowed out

History of concentrations (default: hc)

```
# kiche-1.1 with model=10a
# history of species concentration & gas pressures(time=s, conc.=mol/dm3, ...
#  1:time    2:step |     3:b(oh)3   4:b(oh)4_m        5:cl2       6:cl2g  ...
   0.000000         0   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00 ...
   1.032590       363   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00 ...
   2.044457       377   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00 ...
   3.044457       387   0.0000E+00   0.0000E+00   0.0000E+00   0.0000E+00 ...
....
```

The chemical species names are sorted alphabetically. The species included depend on the reaction set, and so do the column numbers.

History of some cell variables and rate constants (default: none) When the output by “`-hr <file>`” is specified, some cell attributes (dose rate, temperatures, gas-aqueous mass transfer, adsorption and so on) and rate constants of the processes are written in the specified file.

```
# kiche-1.2 with model=10a
# history of selected cell vars. & rate constantsas  (variable names ...
# 1:time 2:step 3:dose 4:temp 5:tempg 6:vcon 7:tcon 8:fcon 9:hpi2 ...
   0.000000          0   0.0000E+00   0.0000E+00   0.0000E+00   ...
   1.032590        364   1.3600E+00   2.9800E+02   2.9800E+02   ...
   2.044457        378   1.3600E+00   2.9800E+02   2.9800E+02   ...
....
```

List of contributing processes to concentration changes of species (default: none) When the output by "`-lr <file>`" is specified, a list of concentrations and their change rates, the processes contributing to the changes, and their contributions are written as follows.

```
# kiche-1.2 with model=10a
# list of conc. change rate and contributions
# (not shown if conc and dc/dt are both 0)
# step=         0 time=   0.000000
fe_2p   : conc. dc/dt=  5.0000E-06   0.0000E+00  <= If conc. is not changing,
fe_3p   : conc. dc/dt=  5.0000E-06   0.0000E+00     no processes of contribution
i_m     : conc. dc/dt=  9.0000E-06   0.0000E+00     are displayed.
....

# step=      1508 time=     854.11
cl2     : conc. dc/dt= -1.1454E-49   1.7380E-58
  trcl2_4  :   1.7380E-58
cl2g    : conc. dc/dt=  2.1336E-55  -1.3794E-59
  trcl2_2  :  -1.3794E-59
cl_m    : conc. dc/dt=  4.8880E-27   0.0000E+00
e_m     : conc. dc/dt=  4.2330E-14   7.5788E-12
  ge_m     :   1.0352E-07                        <= Processes contributing to the
  e02      :  -2.3080E-17                           conc. change of e_m, and
  e03      :  -1.6153E-17                           their contributions
  e04      :  -1.6378E-15
  e05      :  -8.6861E-09
  e06      :  -9.4789E-08
  e07      :  -2.4490E-13
  e08      :  -6.2910E-18
....
```

In this example, the record headed by "e_m" and following lines in the step=1508 says the concentration change rate by each process, ge_m, e02, e03, ... are those numbers, and they are summed up to give the change rate $dc/dt = 7.5788 \times 10^{-12}$ (mol/Ls). This list is output for every species whose concentration or concentration change rate is not zero.

Restart data (binary dump, default: none) When the output by "`-bd <file>`" is specified, step, time, dt, c are written as binary data at every output time step.

Debug output (default: none) When the output by "`-db`" is specified, information for debugging is written in a file, "debug". At present, an echo of the input namelist is written.

```
!
! kiche.10a input
!
!simulation of h20 experiment (sweep sys)
!base=20-08: pH=7 buffer, air
! pH 6.97 -> 6.91
! T  24--26 C
! dose 6.6 kGy/h
!
&inputvars

tstart=-1800
tend=9000
ttbl_dtmax%n = 6
ttbl_dtmax%t = -1800 -1790 -800  0    10  1000
ttbl_dtmax%x = 1e-2  0.5   5.    1e-2 0.5  5.
ttbl_outdt%n = 6
ttbl_outdt%t = -1800 -1790 -800  0    10  1000
ttbl_outdt%x = 1e-1  1     10    1e-1 1    10

ttbl_dose%n= 3
ttbl_dose%t=-1800  0   7200
ttbl_dose%x= 0     6.6  0

iphgiven = 1
ttbl_phctrl%n = 2
ttbl_phctrl%t = -1800 0
ttbl_phctrl%x =  1    0
ttbl_ph%n = 1
ttbl_ph%t = -1800
ttbl_ph%x = 6.97
!ttbl_ph%n = 2
!ttbl_ph%t = -1800 7200
!ttbl_ph%x = 6.97  6.91

c%temp = 298.
c%tempg = 298.
c%vaq = 0.10
c%vg = 0.286
c%aint = 0.567
qgflow = 0.0167  ! gas flowrate (1L/min=0.0167L/s)

c%gas%po2 = 0.021e6  ! o2  21%
c%gas%pn2 = 0.079e6  ! n2  79%
c%gas%pco2= 32       ! co2 3.2e-2 %
c%xhm = 0.80

c%conc(b(oh)3) = 0.1      ! B(OH)3 0.1M buffer
c%conc(o2) = 2.67e-4 ! O2 saturation at 25C
c%conc(i_m) = 1e-4    ! I-
c%conc(org) = 1.4e-6    ! orgktn 100ppb TOC

mdlctrl="kmto2base orggout"
kmt0o2=4e-4 ! measured O2 transfer coeff at 25C
/
```

Figure C.1: Input for simulation of JAEA experiment, “Base” case. (see Section 4.1)

```
!
! input for isp41-rtf st2
!
&inputvars

tstart = 0
tend = 1080000 ! 300h
dtmin = 1e-9
outstep = 1000
ttbl_dtmax%ihr = 0
ttbl_dtmax%n = 3
ttbl_dtmax%t = 0   100 1000
ttbl_dtmax%x = 0.1  10   50
ttbl_outdt%ihr = 0
ttbl_outdt%n = 4
ttbl_outdt%t = 0 100 1000 10000
ttbl_outdt%x = 1  10  100   600

c%typ = 0
c%temp = 298
c%tempg = 298
c%dose = 1.36

c%vaq = 25
c%vg = 315
c%aint = 37
c%assg = 220
c%wlsg = 0.5
c%ufg = 0.1

c%gas%po2 = 0.021e6
c%gas%pn2 = 0.079e6

igasrule = 0
iphgiven = 1

ihumid = 0
c%xhm = 0.5

!ISP41-RTF St2 t(h)-pH data,
ttbl_ph%ihr = 1
ttbl_ph%n = 78
ttbl_ph%t =
0.00 1.50 3.18 6.14 9.14 15.14 19.95 21.73 24.19 24.69 27.11 30.07 33.07
36.07 39.07 42.07 44.75 45.21 49.48 49.94 50.55 53.52 59.52 62.52 71.52
89.52 95.52 98.52 104.52 115.99 118.14 118.61 119.11 120.11 120.61 121.61
122.11 122.61 123.31 126.28 129.28 132.28 153.28 156.28 159.28 162.28
165.28 168.28 170.13 174.04 177.04 180.04 183.04 186.04 189.04 190.29
190.75 191.25 194.25 194.76 197.73 217.53 220.53 223.53 235.53 238.53
241.53 250.53 253.53 262.53 265.53 268.53 271.53 274.53 277.53 280.53
283.53 288.21
ttbl_ph%x =
10.16 10.27 10.07 10 9.97 9.93 9.95 8.7 8.51 8.46 8.45 8.5 8.53 8.48
8.53 8.5 8.26 7.81 7.8 7.76 7.78 7.76 7.73 7.78 7.75 7.8 7.73 7.8 7.73 7.83
6.69 6.7 6.81 6.82 6.55 6.5 6.62 6.57 6.51 6.5 6.46 6.5 6.48 6.43 6.41
6.51 6.46 5.54 5.49 5.44 5.51 5.4 5.49 5.44 5.44 10.19 10.02 9.95 9.95 9.92
9.95 9.98 10 9.98 9.93 9.95 9.97 9.93 9.98 9.92 9.97 9.95 10 9.97 9.98
9.97 9.95 9.95

c%conc(i_m) = 9e-6 ! i_m
c%conc(fe_2p) = 1e-5 ! fe_2p  1e-5 for both seems to give
c%conc(fe_3p) = 1e-5 ! fe_3p  a good agreement.

mdlctrl="isp41rtfset kmto2base"
kmt0o2=5e-4

/
```

Figure C.2: Input for RTF Phase-0 Test-2 Stage-2 simulation, “Base” case. (see Section 4.2)

References

[1] M. Naritomi. Physical and chemical behaviors of radioiodine in atmospheres from degraded core to environment following nuclear reactor accident. *Hoken-Butsuri*, 22, pp.189–207, 1987. [In Japanese].

[2] J.C. Wren, J.M. Ball, and G.A. Glowa. The chemistry of iodine in containment. *Nuclear Technology*, 129, pp.297–325, 2000.

[3] Working Group on Analysis and Management of Accidents (WGAMA). State of the art report on iodine chemistry. NEA/CSNI/R(2007)/1, OECD/NEA/CSNI, 2007.

[4] E.C. Beahm, C.F. Weber, and T.S. Kress. Iodine chemical forms in LWR severe accidents. NUREG/CR-5732, ORNL/TM-11861, Oak Ridge National Laboratory, 1991.

[5] C.C. Lin. Chemical effects of gamma radiation on iodine in aquous solutions. *Journal of Inorganic & Nuclear Chemistry*, 42, pp.1101–1107, 1980.

[6] C.B. Ashmore, J.R. Gwyther, and H.E. Sims. Some effects of pH on inorganic iodine volatility in containment. *Nulcear Engineering and Design*, 166, pp.347–355, 1996.

[7] C.B. Ashmore, D. Brown, S. Dickinson, and H.E. Sims. Measurement of the radiolytic oxidation of aqueous CsI using a sparging apparatus. *Nuclear Technology*, 129, pp.387–397, 2000.

[8] F. Taghipour and G.J. Evans. Radioiodine volatilization in the presence of organic compounds. *Nuclear Technology*, 134, pp.208–220, 2001.

[9] F. Taghipour and G.J. Evans. Iodine behavior under conditions relating to nuclear reactor accidents. *Nuclear Technology*, 137, pp.181–193, 2002.

[10] K. Moriyama, S. Tashiro, N. Chiba, F. Hirayama, Y. Maruyama, H. Nakamura, and A. Watanabe. Experiments on the release of gaseous iodine from gamma-irradiated aqueous CsI solution and influence of oxygen and methyl isobutyl ketone (MIBK). *Journal of Nuclear Science and Technology*, 47(3), pp.229–237, 2010.

[11] J. C. Wren and J. M. Ball. LIRIC 3.2 an updated model for iodine behaviour in the presence of organic impurities. *Radiation Physics and Chemistry*, 60, pp.577–596, 2001.

[12] S. Dickinson and H. E. Sims. Development of the INSPECT model for the prediction of iodine volatility from irradiated solutions. *Nuclear Technology*, 129, pp.374–386, 2000.

[13] Salih Güntay and Robin Cripps. IMPAIR/3 a computer program to analyze the iodine behaviour in multi-compartments of a LWR containment. PSI-Bericht No.128, Paul Scherrer Institut, 1992.

[14] C. Poletiko, D. Jacquemain, and C. Hueber. The volatility of iodine from irradiated iodide solutions in the presence or absence of painted surfaces: modeling work on the IODE code at IPSN. *Nuclear Technology*, 126, pp.215–227, 1999.

[15] R.O. Gauntt, J.E. Cash, R.K. Cole, C.M. Erickson L.L. Humphries, S.B. Rodriguez, and M.F. Young. MELCOR computer code manuals: vol.1 primer and use's guide, version 1.8.6. NUREG/CR-6119 vol.1, rev.3, SAND2005-5713, Sandia National Laboratories, 2005.

[16] C.Y. Paik, R.E. Henry, and M.A. McCartney. MAAP4.0 benchmarking with the TMI-2 experience. *Proc. International Conference on Probabilistic Safery Assessment Methodology and Applications, Seoul, Korea*, 1995.

[17] J. Ishikawa, K. Muramatsu, and T. Sakamoto. Systematic source term analysis for level 3 PSA of a BWR with Mark-II containment with THALES-2 code. *10th International Conference on Nuclear Engineering (ICONE-10), Hyatt Regency Crystal City, USA (CD-ROM, no. ICONE10-22080)*, 2002.

[18] L. Ammirabile, A. Bielauskas, A. Bujan, B. Toth, G. Gyenes, J. Gienstbier, L. Herranz, J. Fontanet, N. Reinke, A. Rizoiu, and J. Jancovic. Progress of ASTEC validation on fission product release and transport in circuits and containment. *Proc. the 3rd European Review Meeting on Severe Accident Research (ERMSAR-2008), Nesseber, Bulgaria*, 2008.

[19] H. Iizuka. Development of an analysis code for the iodine behavior in the containment vessel during severe accidents and application for isp-41. *INSS Journal (Institute of Nuclear Safety System, Inc.)*, 11, pp.256–266, 2004. [in Japanese].

[20] A.J. Elliot. Rate constants and G-values for the simulation of the radiolysis of light water over the range 0–300^{o}C. AECL-11073, COG-94-167, Atomic Energy of Canada, Ltd., 1994.

[21] Y. Katsumura, G. Sunaryo, D. Hiroishi, and K. Ishigure. Fast neutron radiolysis of water at elevated temperatures relevant to water chemistry. *Progress in Nuclear Energy*, 32, pp.113–121, 1998.

[22] G. Glowa, P. Driver, and J. C. Wren. Irradiation of MEK–II: A detailed kinetic model for the degradation of 2-butanone in aerated aqueous solutions under steady-state γ-radiolysis conditions. *Radiation Physics and Chemistry*, 58, pp.49–68, 2000.

[23] J. C. Wren and G. A. Glowa. A simplified kinetic model for the degradation of 2-butanone in aerated aqueous solutions under steady-state gamma-radiolysis. *Radiation Physics and Chemistry*, 58, pp.341–356, 2000.

[24] P. Driver, G. Glowa, and J. C. Wren. Steady-state γ-radiolysis of aqueous methyl ethyl ketone (2-butanone) under postulated nuclear reactor accident conditions. *Radiation Physics and Chemistry*, 57, pp.37–51, 2000.

[25] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical recipes in FORTRAN, 2nd ed.* Cambridge Univ. Press, 1992.

[26] J3-Fortran. http://www.j3-fortran.org, 2010.

[27] The Perl programming language. http://www.perl.org, 2010.

[28] LaTeX–a document preparation system. http://www.latex-project.org, 2010.

[29] Adobe and industry standards. http://www.adobe.com/enterprise/standards, 2009.

[30] K. Moriyama, Y. Maruyama, and H. Nakamura. Steam explosion simulation code JASMINE v.3 user's guide. JAEA-Data/Code 2008-014, Japan Atomic Energy Agency, 2008.

[31] The International Association for the Properties of Water and Steam. *Guideline on the Henry's constant and vapor-liquid distribution constant for gases in H_2O and D_2O at high temperatures*, 2004.

[32] Kagaku-Kogaku Kyokai, editor. *Kagaku Kogaku Binran 5th ed.* Maruzen, 1988. [in Japanese].

[33] J.C. Wren, D.J. Jobe, G.G. Sanipelli, and J.M. Ball. Dissolution of organic solvents from painted surfaces into water. *Canadian Journal of Chemistry*, 78, pp.464–473, 2000.

[34] J. Dewulf, D. Drijvers, and H. van Langenhove. Measurements of Henry's law constant as function of temperature and salinity for the low temperature range. *Atmospheric Environment*, 29, pp.323–331, 1995.

[35] L.F. Parsly. Design consideration of reactor containment spray systems–Part IV calculation of iodine-water partition coefficients. ORNL-TM-2412, Oak Ridge National Laboratory, 1970.

[36] G.A. Glowa and J.C. Wren. Aqueous-gas partitioning and hydrolysis of organic iodides. *Canadian Journal of Chemistry*, 81, pp.230–243, 2003.

[37] K. Moriyama, N. Chiba, S. Tashiro, Y. Maruyama, H. Nakamura, and A. Watanabe. Experiment and modeling for solvent leaching from paint matrix considering equilibrium. *Journal of Nuclear Science and Technology*, 48(6), 2011. [to be published].

[38] J.M. Ball, J.C. Wren, and J.R. Mitchell. The dissolution of organic solvents from Amerlock 400 epoxy paint. *Canadian Journal of Chemistry*, 81, pp.850–860, 2003.

[39] H.S. Rosenberg, J.M. Genco, and D.L.Morrison. Fission-product deposition and its enhancement under reactor accident conditions: deposition on containment-system surfaces. BMI-1865, Battelle Memorial Institute, 1969.

[40] J. C. Wren, G. A. Glowa, and J. Merritt. Corrosion of stainless steel by gaseous I_2. *Journal of Nuclear Materials*, 265, pp.161–177, 1999.

[41] Atomic Energy Society of Japan, editor. *Handbook of Water Chemistry of Nuclear Reactor System.* Corona Publishing Co.,Ltd., 2000.

[42] M.H. Kim, C.S. Kim, H.W. Lee, and K. Kim. Temperature dependence of dissociation constants for formic acid and 2,6-dinitrophenol in aqueous solutions up tp 175°C. *Journal of Chemical Society, Faraday Transactions*, 92, pp.4951–4956, 1996.

[43] A.J. Read. Ionization constants of benzoic acid from 25 to 250 °C and to 2000 bar. *Journal of Solution Chemistry*, 10, pp.437–450, 1981.

[44] J. Ball et al. ISP 41 containment iodine computer code exercise based on a Radioiodine Test Facility (RTF) experiment. NEA/CSNI/R(2000)6/vol.1–2, OECD/NEA, 2000.

[45] J. Ball et al. ISP-41–follow-up exercise (phase 2): iodine code comparison exercise against CAIMAN and RTF experiments. NEA/CSNI/R(2004)16, OECD/NEA, 2004.

[46] B. Clément, C. Marchand, H.J. Allelein, and G. Weber. Lessons learnt concerning iodine chemistry ISP-41, ISP-46. *EUROSAFE 2003, Seminar 2 (http://www.eurosafe-forum.org/)*, 2003.

国際単位系（SI）

表１．SI 基本単位

基本量	SI 基本単位	
	名称	記号
長さ	メートル	m
質量	キログラム	kg
時間	秒	s
電流	アンペア	A
熱力学温度	ケルビン	K
物質量	モル	mol
光度	カンデラ	cd

表２．基本単位を用いて表されるSI組立単位の例

組立量	SI 基本単位	
	名称	記号
面積	平方メートル	m^2
体積	立法メートル	m^3
速さ，速度	メートル毎秒	m/s
加速度	メートル毎秒毎秒	m/s^2
波数	毎メートル	m^{-1}
密度，質量密度	キログラム毎立方メートル	kg/m^3
面積密度	キログラム毎平方メートル	kg/m^2
比体積	立方メートル毎キログラム	m^3/kg
電流密度	アンペア毎平方メートル	A/m^2
磁界の強さ	アンペア毎メートル	A/m
量濃度(a)，濃度	モル毎立方メートル	mol/m^3
質量濃度	キログラム毎立法メートル	kg/m^3
輝度	カンデラ毎平方メートル	cd/m^2
屈折率(b)	（数字の） 1	1
比透磁率(b)	（数字の） 1	1

(a) 量濃度（amount concentration）は臨床化学の分野では物質濃度（substance concentration）ともよばれる。
(b) これらは無次元量あるいは次元１をもつ量であるが、そのことを表す単位記号である数字の１は通常は表記しない。

表３．固有の名称と記号で表されるSI組立単位

組立量	SI 組立単位			
	名称	記号	他のSI単位による表し方	SI基本単位による表し方
平面角	ラジアン(b)	rad	1 (b)	m/m
立体角	ステラジアン(b)	sr(c)	1 (b)	$m^{2/}m^2$
周波数	ヘルツ(d)	Hz		s^{-1}
力	ニュートン	N		$m\ kg\ s^{-2}$
圧力，応力	パスカル	Pa	N/m^2	$m^{-1}\ kg\ s^{-2}$
エネルギー，仕事，熱量	ジュール	J	N m	$m^2\ kg\ s^{-2}$
仕事率，工率，放射束	ワット	W	J/s	$m^2\ kg\ s^{-3}$
電荷，電気量	クーロン	C		s A
電位差（電圧），起電力	ボルト	V	W/A	$m^2\ kg\ s^{-3}\ A^{-1}$
静電容量	ファラド	F	C/V	$m^{-2}\ kg^{-1}\ s^4\ A^2$
電気抵抗	オーム	Ω	V/A	$m^2\ kg\ s^{-3}\ A^{-2}$
コンダクタンス	ジーメンス	S	A/V	$m^{-2}\ kg^{-1}\ s^3\ A^2$
磁束	ウエーバ	Wb	Vs	$m^2\ kg\ s^{-2}\ A^{-1}$
磁束密度	テスラ	T	Wb/m^2	$kg\ s^{-2}\ A^{-1}$
インダクタンス	ヘンリー	H	Wb/A	$m^2\ kg\ s^{-2}\ A^{-2}$
セルシウス温度	セルシウス度(e)	℃		K
光束	ルーメン	lm	cd sr(c)	cd
照度	ルクス	lx	lm/m^2	$m^{-2}\ cd$
放射性核種の放射能(f)	ベクレル(d)	Bq		s^{-1}
吸収線量, 比エネルギー分与, カーマ	グレイ	Gy	J/kg	$m^2\ s^{-2}$
線量当量, 周辺線量当量, 方向性線量当量, 個人線量当量	シーベルト(g)	Sv	J/kg	$m^2\ s^{-2}$
酸素活性	カタール	kat		$s^{-1}\ mol$

(a)SI接頭語は固有の名称と記号を持つ組立単位と組み合わせても使用できる。しかし接頭語を付した単位はもはやコヒーレントではない。
(b)ラジアンとステラジアンは数字の１に対する単位の特別な名称で、量についての情報をつたえるために使われる。実際には、使用する時には記号rad及びsrが用いられるが、習慣として組立単位としての記号である数字の１は明示されない。
(c)測光学ではステラジアンという名称と記号srを単位の表し方の中に、そのまま維持している。
(d)ヘルツは周期現象についてのみ、ベクレルは放射性核種の統計的過程についてのみ使用される。
(e)セルシウス度はケルビンの特別な名称で、セルシウス温度を表すために使用される。セルシウス度とケルビンの単位の大きさは同一である。したがって、温度差や温度間隔を表す数値はどちらの単位で表しても同じである。
(f)放射性核種の放射能（activity referred to a radionuclide）は、しばしば誤った用語で”radioactivity”と記される。
(g)単位シーベルト（PV,2002,70,205）についてはCIPM勧告2（CI-2002）を参照。

表４．単位の中に固有の名称と記号を含むSI組立単位の例

組立量	SI 組立単位		
	名称	記号	SI 基本単位による表し方
粘度	パスカル秒	Pa s	$m^{-1}\ kg\ s^{-1}$
力のモーメント	ニュートンメートル	N m	$m^2\ kg\ s^{-2}$
表面張力	ニュートン毎メートル	N/m	$kg\ s^{-2}$
角速度	ラジアン毎秒	rad/s	$m\ m^{-1}\ s^{-1}=s^{-1}$
角加速度	ラジアン毎秒毎秒	rad/s^2	$m\ m^{-1}\ s^{-2}=s^{-2}$
熱流密度，放射照度	ワット毎平方メートル	W/m^2	$kg\ s^{-3}$
熱容量，エントロピー	ジュール毎ケルビン	J/K	$m^2\ kg\ s^{-2}\ K^{-1}$
比熱容量，比エントロピー	ジュール毎キログラム毎ケルビン	J/(kg K)	$m^2\ s^{-2}\ K^{-1}$
比エネルギー	ジュール毎キログラム	J/kg	$m^2\ s^{-2}$
熱伝導率	ワット毎メートル毎ケルビン	W/(m K)	$m\ kg\ s^{-3}\ K^{-1}$
体積エネルギー	ジュール毎立方メートル	J/m^3	$m^{-1}\ kg\ s^{-2}$
電界の強さ	ボルト毎メートル	V/m	$m\ kg\ s^{-3}\ A^{-1}$
電荷密度	クーロン毎立方メートル	C/m^3	$m^{-3}\ sA$
表面電荷	クーロン毎平方メートル	C/m^2	$m^{-2}\ sA$
電束密度，電気変位	クーロン毎平方メートル	C/m^2	$m^{-2}\ sA$
誘電率	ファラド毎メートル	F/m	$m^{-3}\ kg^{-1}\ s^4\ A^2$
透磁率	ヘンリー毎メートル	H/m	$m\ kg\ s^{-2}\ A^{-2}$
モルエネルギー	ジュール毎モル	J/mol	$m^2\ kg\ s^{-2}\ mol^{-1}$
モルエントロピー, モル熱容量	ジュール毎モル毎ケルビン	J/(mol K)	$m^2\ kg\ s^{-2}\ K^{-1}\ mol^{-1}$
照射線量（Ｘ線及びγ線）	クーロン毎キログラム	C/kg	$kg^{-1}\ sA$
吸収線量率	グレイ毎秒	Gy/s	$m^2\ s^{-3}$
放射強度	ワット毎ステラジアン	W/sr	$m^4\ m^{-2}\ kg\ s^{-3}=m^2\ kg\ s^{-3}$
放射輝度	ワット毎平方メートル毎ステラジアン	$W/(m^2\ sr)$	$m^2\ m^{-2}\ kg\ s^{-3}=kg\ s^{-3}$
酵素活性濃度	カタール毎立方メートル	kat/m^3	$m^{-3}\ s^{-1}\ mol$

表５．SI 接頭語

乗数	接頭語	記号	乗数	接頭語	記号
10^{24}	ヨタ	Y	10^{-1}	デシ	d
10^{21}	ゼタ	Z	10^{-2}	センチ	c
10^{18}	エクサ	E	10^{-3}	ミリ	m
10^{15}	ペタ	P	10^{-6}	マイクロ	µ
10^{12}	テラ	T	10^{-9}	ナノ	n
10^9	ギガ	G	10^{-12}	ピコ	p
10^6	メガ	M	10^{-15}	フェムト	f
10^3	キロ	k	10^{-18}	アト	a
10^2	ヘクト	h	10^{-21}	ゼプト	z
10^1	デカ	da	10^{-24}	ヨクト	y

表６．SIに属さないが、SIと併用される単位

名称	記号	SI 単位による値
分	min	1 min=60s
時	h	1h =60 min=3600 s
日	d	1 d=24 h=86 400 s
度	°	1°=(π/180) rad
分	’	1’=(1/60)°=(π/10800) rad
秒	”	1”=(1/60)’=(π/648000) rad
ヘクタール	ha	$1ha=1hm^2=10^4m^2$
リットル	L，l	$1L=1l=1dm^3=10^3cm^3=10^{-3}m^3$
トン	t	$1t=10^3$ kg

表７．SIに属さないが、SIと併用される単位で、SI単位で表される数値が実験的に得られるもの

名称	記号	SI 単位で表される数値
電子ボルト	eV	$1eV=1.602\ 176\ 53(14)\times10^{-19}J$
ダルトン	Da	$1Da=1.660\ 538\ 86(28)\times10^{-27}kg$
統一原子質量単位	u	1u=1 Da
天文単位	ua	$1ua=1.495\ 978\ 706\ 91(6)\times10^{11}m$

表８．SIに属さないが、SIと併用されるその他の単位

名称	記号	SI 単位で表される数値
バール	bar	$1\ bar=0.1MPa=100kPa=10^5Pa$
水銀柱ミリメートル	mmHg	1mmHg=133.322Pa
オングストローム	Å	$1\ Å=0.1nm=100pm=10^{-10}m$
海里	M	1 M=1852m
バーン	b	$1\ b=100fm^2=(10^{-12}cm)2=10^{-28}m^2$
ノット	kn	1 kn=(1852/3600)m/s
ネーパ	Np	SI単位との数値的な関係は、対数量の定義に依存。
ベル	B	
デジベル	dB	

表９．固有の名称をもつCGS組立単位

名称	記号	SI 単位で表される数値
エルグ	erg	$1\ erg=10^{-7}\ J$
ダイン	dyn	$1\ dyn=10^{-5}N$
ポアズ	P	$1\ P=1\ dyn\ s\ cm^{-2}=0.1Pa\ s$
ストークス	St	$1\ St\ =1cm^2\ s^{-1}=10^{-4}m^2\ s^{-1}$
スチルブ	sb	$1\ sb\ =1cd\ cm^{-2}=10^4cd\ m^{-2}$
フォト	ph	$1\ ph=1cd\ sr\ cm^{-2}\ 10^4lx$
ガル	Gal	$1\ Gal\ =1cm\ s^{-2}=10^{-2}ms^{-2}$
マクスウェル	Mx	$1\ Mx = 1G\ cm^2=10^{-8}Wb$
ガウス	G	$1\ G\ =1Mx\ cm^{-2}\ =10^{-4}T$
エルステッド(c)	Oe	$1\ Oe \triangleq (10^3/4\pi)A\ m^{-1}$

(c) ３元系のCGS単位系とSIでは直接比較できないため、等号「 ≙ 」は対応関係を示すものである。

表10．SIに属さないその他の単位の例

名称	記号	SI 単位で表される数値
キュリー	Ci	$1\ Ci=3.7\times10^{10}Bq$
レントゲン	R	$1\ R = 2.58\times10^{-4}C/kg$
ラド	rad	$1\ rad=1cGy=10^{-2}Gy$
レム	rem	$1\ rem=1\ cSv=10^{-2}Sv$
ガンマ	γ	1 γ =1 nT=10-9T
フェルミ		1フェルミ=1 fm=10-15m
メートル系カラット		1メートル系カラット = 200 mg = 2×10-4kg
トル	Torr	1 Torr = (101 325/760) Pa
標準大気圧	atm	1 atm = 101 325 Pa
カロリー	cal	1cal=4.1858J（「15℃」カロリー），4.1868J（「IT」カロリー）4.184J（「熱化学」カロリー）
ミクロン	µ	$1\ \mu\ =1\mu m=10^{-6}m$

（第8版，2006年改訂）

この印刷物は再生紙を使用しています