WSRC-TR-2002-00209

Tank 41 Salt Dissolution Flowsheet Modeling

J. A. Pike
Westinghouse Savannah River Company
Aiken, SC 29808

This document was prepared in conjunction with work accomplished under Contract No. DE-AC09-96SR18500 with the U.S. Department of Energy.

DISCLAIMER

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

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

Available for sale to the public, in paper, from: U.S. Department of Commerce, National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161, phone: (800) 553-6847, fax: (703) 605-6900, email: orders@ntis.fedworld.gov online ordering: http://www.ntis.gov/help/index.asp

Available electronically at http://www.osti.gov/bridge

Available for a processing fee to U.S. Department of Energy and its contractors, in paper, from: U.S. Department of Energy, Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831-0062, phone: (865) 576-8401, fax: (865) 576-5728, email: reports@adonis.osti.gov

Keywords: Waste Removal, Salt Dissolution

Abstract

A computer simulation model for salt dissolution of Tank 41 was constructed and run for three cases. These case runs show that eight to nine 30,000 gal batches of discharge, 240,000 to 270,000 gal total, are needed to result in the first 100,000 gallons of saltcake transferred out of Tank 41. This equates to 140,000 to 160,000 gallons of inhibited water feed solution after initially refilling the interstitial void. The process plan should allow for 140,000 to 200,000 gallons of batch feed solution in addition to the volume to fill the interstitial void to transfer enough dissolved saltcake to reach the initial 100,000 gal of saltcake goal. The initial fill will vary greatly depending on the extent of interstitial removal and refill. The refill feed volume can range from 23,000 to 230,000 gal.

Historical data from Tank 10 salt dissolution was used to compare the results predicted by the model to a real case. The results match the bulk ratios of dissolution water used to salt dissolved and discharge volumes, but consistently under predicted total volumes. This appears to be an artifact of available batch data including flow rates in and out of the tank.

1 Introduction

SRS High Level Waste (HLW) plans to dissolve saltcake in Tank 41 to feed the low curie salt process. The low curie salt process treats high level waste saltcake for final disposal in saltstone. To prepare saltcake for final disposal, it is necessary to separate cesium, the dominant radioactive element, and soluble transuranic material from the waste stream. Since the cesium concentration is well below solubility limits in waste, all the cesium in saltcake is in the supernate and interstitial liquid. If the interstitial liquid can be removed from the saltcake, adequate seperation of cesium may be achieved for disposal in saltstone.

The process to dissolve the saltcake is simulated by an Aspen Custom Modeler™ model using Aspen Properties Plus™ to estimate chemical properties.

2 Process Overview

2.1 Process Description

Saltcake may be considered composed of four parts: salt crystals, interstitial liquid, gas, and insoluble solids. When salt crystals form and accumulate in a waste tank, interstitial space between the crystals will hold supernate and, potentially, gas. Gas can consist of trapped air, water vapor, and radiolytically produced gases, primarily, oxygen, hydrogen, and nitrous oxides. Insoluble solids may consist of entrained high-level waste sludge solids or, potentially, insoluble materials that form during concentration in the evaporator. Some materials such as aluminum hydroxide as gibbsite may take a long time to precipitate and may not be present in freshly formed saltcake.

Interstial liquid and supernate probably contains all the cesium in soluble, ionic form and none in crystalline form. A fully saturated Savannah River Site (SRS) waste supernate would contain about 10,738 Ci/gallon cesium-137. Comparison with the actual radiation values for SRS salt solutions indicates the cesium levels are only 0.2 to 0.9 percent of the saturation level. This indicates that most if not all the cesium is in solution

and resides in the interstitial liquids of the salt cake. Thus, removing the interstitial liquid will remove the cesium.

It is estimated that at least 75% of the interstitial liquid will need to be removed from saltcake to produce salt solution capable of meeting the Saltstone WAC. The plan for Tank 41 is to place the suction at the 24" level to drain the top portion of the saltcake as much as possible. The existing Fixed Length Jet (FLJ) will be used to decant the free supernate from the top of the saltcake. Once the free supernate is removed from the tank, the FLJ will be removed. A submersible pump will then be installed in the tank and mined to the 24" level thus forming a pump well in the saltcake. The pump will then be used to remove interstitial liquid and salt solution at a slow rate, less than 5 gpm on average. Inhibited water will be added to the tank to cover the saltcake. A relatively thin layer of liquid will be maintained above the saltcake by adding inhibited water while simultaneously transferring out of the tank at about 20 gpm. The dissolved saltcake will be sent to Tank 50 for final characterization before transferring to the Saltstone Facility. The dissolution will occur in relatively small batches, 10,000 to 30,000 gallons, to minimize the risk of sending a significant amount of dissolved saltcake with too much cesium[2].

2.2 Tank Farm Chemistry

The composition of the existing waste inventory accounts for radiolytic decomposition of sodium nitrate to sodium nitrite and the reactive sorption of carbon dioxide from the air to form sodium carbonate. The only other radiolytic reaction products are gaseous hydrogen and oxygen, which evolves from the waste to the tank vapor space. The vapor space vents to the atmosphere.

Free hydroxide, nitrate and nitrite concentrations are monitored in stored waste and adjusted, when necessary, by adding caustic or sodium nitrite to maintain the concentrations in a range that minimizes the corrosion rate of the carbon steel waste tanks.

3 Technical Bases for Material Balance

An Aspen Custom Modeler™ model was developed to calculate the material balance for the Tank 41 flowsheet. The following sections describe the basis of the model.

3.1 Salt Dissolution Model Assumptions

The following assumptions are used in the modeled flow sheets:

Initial Tank Conditions:

Saltcake has an average level of 350.7" or 1,230,000 gallons.[2]
Salt void fraction is varied.
Current tank level is 356.5". [2]
Specific gravity of bulk saltcake is calculated based on composition and void fraction.
Total supernate volume = 356.5" * 3510 gal/inch – 1,230,000 gal saltcake * (1- void fraction).

Pre-Dissolution Preparation:

Remove free supernate above saltcake at 55 gpm nominal rate.
Tank remains quiescent for 24 hours.

Drain:

Drain interstitial liquid at 4 gpm average rate.
Extent of drainage is varied.
Tank remains quiescent for 24 hours.

Dissolution:[2]

Refill interstitial pores with inhibited water at 30 gpm until 3" of free liquid above saltcake.
Composition of the inhibited water is 0.01 M NaOH and 0.011 M $NaNO_2$.
Tank remains quiescent for 24 hours.
Begin simultaneous transfer in and out of tank.
Establish 20 gpm discharge rate and vary inhibited water feed to maintain free liquid layer over saltcake between 2" – 3".
Pause for 24 hours after every 30,000 gallons of discharge.
Continue dissolution batches until saltcake level is less than 250".

In several instances, quiescent periods are picked to clearly delineate process steps as well as allow adequate time for the model to reach steady state conditions between steps. Though these quiescent times approximately match the technical plan, they may not completely agree or match the program schedule.

3.2 Salt Dissolution Rate

Saltcake in High Level Waste tanks consists of three phases saltcake, supernate, and gas. The gas phase is neglected in most cases since it has

insignificant contribution to total mass, but does affect bulk density. Material may transfer between phases by crystallization of salts in the supernate or by dissolution of the salt cake. The dissolution rate for each salt is approximated by:

$$\text{dissolution rate} = d[A_xB_y]/dt = K_{dis} * ([B^-]_{equil} - [B^-]_{sup}) * ([A^+]_{equil} - [A^+]_{sup})$$

where:

$[A_xB_y]$ = concentration of salt A_xB_y where A is the cation and B is the anion, kmole/m^3,
$[B^-]$ = concentration of anion B, kmole/m^3,
$[A^+]$ = concentration of cation A, kmole/m^3,
K_{dis} = dissolution rate constant, (kmole/hr) / (kmole/m^3) or m^3/hr,
$_{equil}$ = at equilibrium, and
$_{sup}$ = in supernate.

In some cases, other salts control the cation or anion concentration. For example, many salts contribute to the sodium concentration in a waste solution. As such, the sodium concentration does not change significantly with a change in concentration of a specific salt. The difference between supernate concentration and the equilibrium concentration is approximately constant for an incremental change in the salt concentration. In this case, the cation term is assumed to be incorporated in the apparent K_{dis}, thus, the rate equation becomes:

$$d[A_xB_y]/dt = K_{dis} * ([B^-]_{equil} - [B^-]_{sup}).$$

For a salt where other salts control the supernate concentration of the anion, the rate equation becomes:

$$d[A_xB_y]/dt = K_{dis} * ([A^+]_{equil} - [A^+]_{sup}).$$

An additional term is added to avoid discontinuities that cause the modeler equation solver to fail to converge on a solution. This term smoothes the transition from solid salt to no solid salt with a ratio of

$$Moles_{salt\ -i} / (Moles_{salt\ -i} + \text{factor})$$

where:

salt = salt from the saltcake,
$Moles_{salt\ -i}$ = moles of component i, and
factor = arbitrarily small value relative to $Moles_{salt\ -i}$, moles.

This ratio is approximately 1 until the number of moles of component i approaches the value of factor. As the number of moles approaches zero, the value of the ratio will asymptotically approach zero and smooth the sharp discontinuity when a dissolving component reaches complete dissolution. Thus, the final form of the dissolution rate equations is:

$$d[A_xB_y]/dt = K_{dis} * ([B^-]_{equil} - [B^-]_{sup}) * Moles_{salt\ -i} / (Moles_{salt\ -i} + \text{factor}).$$

Addition of this ratio prevents the model from simulating precipitation of salts if there is zero solids present initially. If no solids of a specific component are present in the initial conditions, then the rate equation always equals zero. Therefore the model will not initiate precipitation. If any amount of solids is present, then precipitation is initiated as normally expected. This essentially limits the model to use for dissolution only or very carefully setting initial conditions for other systems.

The model currently assumes that salt dissolves at a very high rate and the dissolution constant is set to an arbitrarily high value of 1000 kmol/hr / kmol/m^3. Laboratory tests of salt dissolution kinetics indicate that the rate constant is very high, but data with adequate time resolution is not available to calculate the value.[3] These tests clearly demonstrate that the rate limiting step is the rate of mass transfer of the concentrated anions in solution near the crystalline surface to the bulk of the solution and not a function of chemical composition in the salt cake. Diffusion from the crystalline surface can be described mathematically analogous to heat transfer. In so doing, the parameters that affect diffusion include the diffusion film thickness and surface area. The velocity of the bulk liquid above the crystalline surface controls the film thickness. The dissolution methods used will draw a significant amount of dilute solution through the salt cake, which will tend to minimize film thickness. Surface area is the same in all dissolution methods available although some methods will only affect the film thickness at the top of the salt cake. This model assumes perfect mixing in the bulk liquid, which is reasonable for dissolution methods that draw a significant amount of liquid through the salt cake, affecting the film thickness for the bulk of the salt cake. Future model improvements will incorporate limitations due to diffusion if possible.

3.3 Determination of Equilibrium Concentration of Supernate

Equilibrium concentration of soluble species in solution is based on Aspen Properties Plus™ thermodynamic solubility estimation models. Other

estimation models may be used. The total tank contents are considered when calculating the equilibrium concentration. This method presumes that the supernate has a uniform composition through out the waste tank.

The true composition function returns mole fractions. In order to determine the total moles at equilibrium, the moles of water in the apparent composition is assumed equal to the moles of water in the equilibrium composition. Thus, the total moles at equilibrium can be determined by:

$$Moles_{equil} = Moles_{water} / z_{equil\text{-}water}$$

where:

$Moles_{equil}$ = total number of moles at equilibrium,
$z_{equil\text{-}water}$ = molar fraction of water at equilibrium, and
$Moles_{water}$ = number of moles of water at equilibrium = number of moles of water before equilibrium.

Using Aspen Properties Plus; to estimate solution properties at equilibrium, concentration of each soluble component may be calculated by:

$$[i] = z_{equil\text{-}i} * Moles_{equil} / V_{equil}$$

$$V_{equil} = Moles_{equil} / r_{molar\text{-}equil}$$

$$[i] = z_{equil\text{-}i} * Moles_{equil} / (Moles_{equil} / r_{molar\text{-}equil}) = z_{equil\text{-}i} * r_{molar\text{-}equil}$$

where:

$r_{molar\text{-}equil}$ = molar density of supernate at equilibrium, kmole/m^3,
[i] = concentration of component i, kmole/m^3, and
V_{equil} = volume of supernate at equilibrium, m^3.

3.4 Determination of Concentration in Supernate

The supernate composition is calculated by estimating actual component compositions and back calculating the corresponding stoichiometric concentration of apparent components. The model listing in section 5 indicates the stoichiometric relationships under the model "Supernate", subsection labeled "SupernateChemistry".

3.5 Material Balance

Mole balance for the waste tank supernate is calculated by simultaneous solution of the following:

Component balance:

$$dMoles_{sup\text{-}i}/dt = F_{dis} * z_{dis\text{-}i} + M_{salt\text{-}i} - F_{out} * z_{out\text{-}i}$$

where:

$Moles_i$ = moles of component i,
F = total molar flow rate, moles/hr,
M = molar flow rate, moles/hr,
z = molar fraction of component i,
t = time, hr,
sup = supernate,
dis = dissolution liquid,
salt = salt from the saltcake, and
out = dissolved salt solution transferred out of tank.

Total mole balance:

$$d(Total\ Moles_{sup})/dt = F_{dis} + S\ M_{salt\text{-}i} - F_{out}$$

where:

Total Moles = Sum total of all moles

Definition of mole fraction:

$$\text{Total Moles}_i = z_i * \text{Total Moles}.$$

A specific phase may be defined similarly; for example, the supernate phase is as follows:

$$\text{Moles}_{sup\text{-}i} = z_{sup\text{-}i} * \text{Total Moles}_{sup}.$$

Mass balance for the waste tank supernate is calculated by the following:

$$m_{out} = S\ (F_{out} * z_{out\text{-}i} * MW_i)$$

where:

MW = molecular weight, kg/kmole, and
m = mass flow rate, kg/hr.

Mass fractions may be converted from mole fraction as follows:

$$w_{out\text{-}i} = z_{out\text{-}i} * MW_i / MW_{aq}$$

where:

MW_{aq} = average molecular weight of supernate, kg/kmole, S ($z_{out\text{-}i} * MW_i$), and
w = mass fraction.

Individual components dissolve or precipitate until saturation levels are reached or the solids are depleted. The model assumes saturation can be achieved. Future revisions may allow user input to limit extent of saturation.

3.6 Model Limitations

The number of components in the salt cake and supernate are limited to major components at this time. Other components can be added by expanding the chemistry model to include these components. Stoichiometry and dissolution rate constant for each additional component will need to be added to the model.

The components currently modeled are shown in Table 1.

Table 1: Waste Components Modeled

Ionic	Aqueous Phase	Solid
Cs^+	$CsNO_3$	$CsNO_3(s)$
AlO_2^-	$NaAlO_2$	NaAlO(s)
Cl^-	NaCl	NaCl(s)
CO_3^{-2}	Na_2CO_3	$NaCO_3(s)$
F^-	NaF	NaF(s)
NO_2^-	$NaNO_2$	$NaNO_2(s)$
NO_3^-	$NaNO_3$	$NaNO_3(s)$
OH^-	NaOH	NaOH(s)
PO_4^{-3}	Na_3PO_4	$NaPO_4(s)$
SO_4^{-2}	Na_2SO_4	$NaSO_4(s)$
Na^+	Na_2SiO_3	
	H_2O	

Solids of a specific component must be present in the initial conditions for precipitation to be initiated. Initial saltcake must include a small amount of all components to assure equilibration is achieved at the start of each run.

The model assumes an isothermal process, so changes in temperature during the process, exothermic reactions, or endothermic reactions are not affected. Considering that the waste tank is a large heat sink, and the dissolution process is relatively slow, the waste tank temperature is unlikely to change much during dissolution, effectively approximating an isothermal process.

3.7 Salt Waste Composition

Table 2 through Table 4 show the salt composition for Tank 41 as extracted from the Waste Characterization System (WCS) and used in the low curie salt process feed basis. The WCS describes the chemical composition of the salt based on an average saltcake compositions with an interstitial void space filled with supernate. The supernate is assumed to have the same composition as the supernate on top of the salt, which is sampled routinely. Saltcake is not sampled routinely.

The model design anticipates that the known starting compositions are not at equilibrium or steady state. When initiating a model run, the model starts simulating the dissolution or precipitation of salt until a steady state condition is achieved. Using the WCS composition for saltcake and supernate, the bulk saltcake level is reduced over 50" before simulating the actual process. The resulting supernate does not approximate the supernate sample data. Considering that the WCS composition for saltcake is based on an overall average and not necessarily samples from the specific tank, this result is not unexpected. Therefore, other or additional data is needed for input.

Salt sample data from Tank 41 is shown in Table 5. The results are shown for composition of the total sample. Part of the sample is supernate and part is solids. Using the measurement of water content, the amount of interstitial supernate can be approximated as shown in Table 5. The composition is estimated by assuming the interstitial supernate is the same composition as the bulk supernate in the tank. The composition of saltcake solids is calculated by subtracting the quantity of each supernate component from the total composite composition. The supernate composition is determined by averaging several supernate samples before and after the saltcake sample date. No transfers into the tank that would change the supernate composition occurred during this period.

Table 2: WCS Saltcake Composition of Tank 41

Area	H
Type	IIIA
DryBulk Sp.G.	1.45
Interstitial Fraction (vol)	0.22
Salts (wt%)	
$CaSO_4$	9.41E-06
KNO_3	1.00E-03
$Na_2C_2O_4$	4.20E-03
Na_2CO_3	3.13E-02
Na_2CrO_4	1.21E-03
Na_2MoO_4	1.94E-04
Na_2SiO_3	1.59E-03
Na_2SO_4	5.80E-02
Na_3PO_4	5.69E-03
$NaAg(OH)_2$	2.91E-06
$NaAlO_2 \bullet 2H_2O$	1.04E-01
NaCl	5.00E-03
NaF	3.37E-03
NaHgO(OH)	4.45E-05
$NaNO_2$	1.13E-01
$NaNO_3$	5.00E-01
NaOH	1.63E-01
NH_4NO_3	2.50E-04
Other Salts	5.47E-03
Insol. Solids	2.58E-03
Radionuclides (Ci/gal)	
Sr-90 (Ci/gal)	3.56E-01
Y-90	3.56E-01

Cs-137	1.59E-02
Ba-137m	1.50E-02
Pu-238	4.54E-03
Pu-239	1.51E-04

Table 3: WCS Saltcake Composition as Equivalent Ionic Species of Major Components

Component	Concentration (M)
Na^+	2.06E+01
$Al(OH)_4^-$	1.28E+00
CO_3^{2-}	4.28E-01
$C_2O_4^-$	4.54E-02
NO_2^-	2.37E+00
NO_3^-	8.55E+00
OH^-	5.91E+00
PO_4^{3-}	5.03E-02
SO_4^{2-}	5.92E-01
Ca^{2+}	1.00E+00
Cl^-	1.24E-01
F^-	1.16E-01
K^+	1.43E-02
NH_4^-	4.53E-03

Within error of measurement, the model input composition shown in Table 5 is probably at or near equilibrium. Since the model will dissolve or precipitate salts until an equilibrium condition is established, the initial calculation will bring the composition of solids and supernate to equilibrium. The amount of salt dissolved or precipitated at the start of the calculation will indicate how close the initial composition is to equilibrium estimated in the chemistry model. As seen in Figure 1 and Figure 2, the initial correction of saltcake is about 5", ~1.4%. This is within the error to measure average saltcake level as well as chemical compositions. For the high void case, the initial composition produces a saltcake level change of ~15". In order to minimize this error, the resulting chemistry after allowing equilibration time is used as initial conditions instead of the composition shown in Table 5 directly. Figure 3 shows nearly no correction after this adjustment.

Table 4: WCS Supernate Composition of Tanks 41

Assumed Compound	Supernate (kmol/m3)
$NaAl(OH)_4$	2.00E-01
Na_2CO_3	2.60E-01
$Na_2C_2O_4$	5.70E-03
$NaNO_2$	9.89E-01
$NaNO_3$	3.30E+00
NaOH	3.95E+00
Na_3PO_4	2.63E-02
Na_2SO_4	4.86E-02
$AgNO_3$	1.69E-06
$NaAs(OH)_4$	2.18E-03
$Ba(NO_3)_2$	0.00E+00
$Ca(NO_3)_2$	9.97E-06

$Cd(NO_3)_2$	5.32E-05
NaCl	1.16E-02
$Co(NO_3)_3$	1.32E-05
Na_2CrO_4	1.10E-02
$Cu(NO_3)_2$	3.16E-03
$Fe(NO_3)_3$	1.45E-04
NaF	5.20E-03
NaHgOOH	6.03E-04
KNO_3	3.70E-02
$Mg(NO_3)_2$	4.11E-06
$Mn(NO_3)_4$	8.32E-06
$Nd(NO_3)_3$	7.95E-05
$Ni(NO_3)_2$	2.94E-05
$Pb(NO_3)_2$	0.00E+00
$Ru(NO_3)_4$	3.69E-04
Na_2SeO_4	8.70E-03
Na_2SiO_3	0.00E+00
$Sr(NO_3)_2$	3.98E-05
$NaZn(OH)_3$	4.41E-03
$Zr(O)(NO_3)_2$	1.05E-04

Table 5: Tanks 41 Salt and Supernate Data

Salt Sample Data

		wt % Concentration				Salt Sample Average			Moles Per mole of sample		Model input for initial composition: Molar Fractions		WCS Data for comparison: Molar Fractions	
Anion	MW (g/mole)	1-1	2-1	2-3	Sample Average	Assumed molecule	Equivelant Moles per Kg Basis	Molar Fractions	Interstitial Supernate	Salt	Interstitial Supernate	Salt	Interstitial Supernate	Salt
Free OH-	17.0074	2.0	1.0	1.5	1.5	NaOH	0.882	0.05420	0.04457	0.00964	0.09207	0.01868	0.06691	0.30543
Al(OH)4-	95.0095	1.0	0.22	0.70	0.64	NaAlO2	0.0674	0.00414	0.00303	0.00111	0.00627	0.00215	0.00339	0.06615
CO3 2-	60.0092	11.3	6.5	4.7	7.5	Na2CO3	1.250	0.07681	0.00187	0.07494	0.00386	0.14524	0.00440	0.02212
NO3-	62.0052	40.6	44.1	46.3	43.7	NaNO3	7.04	0.43279	0.01399	0.41880	0.02891	0.81167	0.05590	0.44187
NO2-	46.0058	0.98	0.50	0.80	0.76	NaNO2	0.165	0.01015	0.00781	0.00234	0.01613	0.00454	0.01675	0.12248
PO4 3-	94.9716	0.71	0.98	0.54	0.74	Na3PO4	0.0783	0.00481	0.00035	0.00446	0.00073	0.00864	0.00045	0.00260
SO4 2-	96.0616	1.2	0.038	0.65	0.63	Na2SO4	0.0655	0.00403	0.00009	0.00393	0.00019	0.00762	0.00082	0.03059
C2O4 2-	88.0196	0.17	0.076	0.077	0.11	Na2C2O4	0.0122	0.00075	0.00005	0.00070	0.00010	0.00136	0.00010	0.00235
Cl-	35.453	0.024	0.020	0.020	0.021	NaCl	0.00602	0.00037	0.00031	0.00005	0.00065	0.00011	0.00020	0.00641
wt% solids		86.3	89.0	89.2	88.2									
wt% H2O	18.0153	13.7	11.0	10.8	11.8	H2O	6.70	0.41195	0.41195	0.00000	0.85109	0.00000	0.85109	0.00000
						Total Moles	16.272	1	0.48402	0.51598	1.00000	1.00000	1.00000	1.00000

Based on supernate sample data.

Note: Al(OH)4- + Na+ = 2 H2O + NaAlO2

Supernate Sample Data Around Time of Salt Samples

	MW (g/mole)	Concentration (M)									Supernate Sample Average				
Date		07/06/95	02/26/95	08/19/94	05/27/93	10/27/92	02/17/92	08/24/91	01/26/91	Average	g/L	Equivalent molecule	Moles per L Basis	Molar Fractions	molar ratio to water
Specific Gravity		1.4338	1.3835	1.3561	-	1.3588	1.4189	1.4142	1.3951	1.3943					
Free OH-	17.0074	6.7800	6.3900	7.0000	5.4400	5.3300	6.9400	6.3400	6.7400	6.3700	108	NaOH	6.3700	0.09207	0.10818
Al(OH)4-	95.0095	0.4400	0.5500	0.2700	0.4740	-	-	-	-	0.4335	41	NaAlO2	0.4335	0.00627	0.00736
CO3 2-	60.0092	0.2000	0.1900	0.4900	0.1880	-	-	-	-	0.2670	16	Na2CO3	0.2670	0.00386	0.00453
NO3-	62.0052	2.2900	2.1100	2.0700	1.6000	1.5600	2.2900	1.9800	2.1000	2.0000	124	NaNO3	2.0000	0.02891	0.03397
NO2-	46.0058	1.3300	1.2000	1.2100	0.9300	0.9000	1.0900	1.0400	1.2300	1.1163	51	NaNO2	1.1163	0.01613	0.01896
PO4 3-	94.9716	0.0220	0.0190	0.0210	0.1400	-	-	-	-	0.0505	4.8	Na3PO4	0.0505	0.00073	0.00086
SO4 2-	96.0616	0.0140	0.0130	0.0140	0.0120	-	-	-	-	0.0133	1.3	Na2SO4	0.0133	0.00019	0.00023
C2O4 2-	88.0196	<0.0068	<0.0068	<0.0066	<0.0070	-	-	-	-	0.0068	0.6	Na2C2O4	0.0068	0.00010	0.00012
Cl-	35.453	<0.0070	<0.0760	0.0770	<0.0200	-	-	-	-	0.0450	1.6	NaCl	0.0450	0.00065	0.00076
F-	19.00	<0.0130	<0.0530	-	-	-	-	-	-	0.0330	0.6				
									Total mass based on SpG:		1394				
H2O	18.0153									Calculated:	1045	H2O	58.8825	0.85109	1
												Total Moles	69.1848		

Note: Concentration results that show < values are assumed to be at the upper value.

Salt sample data from reference and supernate data from reference .

3.8 Results

Table 6 shows the variations made for different model runs. The base case represents idealized conditions and the other cases are variations from the ideal conditions. The parameters selected for the base case represent relative maximum drainage and void fraction. The remaining two cases vary the extent of drainage and void fraction to obtain results for the expected range of operation. Using the data in Table 5, the high void fractions for each of the three samples are calculated and are shown in Table 7. The sample void fraction ranges from 0.3 to 0.5. Void faction of 0.42 is used to match the estimate made in the hydrogen evaluation for Tank 41 salt dissolution.

Table 6: Variations to Model Runs

Case Label	% Salt Void	Extent of Interstitial Supernate Drainage	Average Drainage Rate (gpm)	Discharge Rate (gpm)	Batch Size
min drain case	22	72 hours of drainage	4.0	20	30,000 gal. discharge
base case	22	Maximum saturation zone 120"	4.0	20	30,000 gal. discharge
high void	42	Maximum saturation zone 120"	4.0	20	30,000 gal. discharge

Note: Dissolution water composition in all cases is 0.01 M NaOH and 0.011 M $NaNO_2$.

Table 7: Tank 41 Sample Void Fraction

	1-1	2-1	2-3
Supernate Density	1.39	1.39	1.39
Dry Salt Density	2.2	2.2	2.2
Bulk Denstiy	1.9	1.4	1.3
Water in Sample (wt. fraction)	0.14	0.11	0.11
Water in Supernate (wt. fraction)	0.75	0.75	0.75
Supernate in Sample (wt. fraction)	0.18	0.15	0.14
Salt in Sample (vol. fraction)	0.70	0.54	0.51
Air in Sample (vol. fraction)	0.04	0.31	0.36
Supernate in Sample (vol. fraction)	0.25	0.15	0.14
Void Fraction	0.30	0.46	0.49

Table 8 through Table 10 show the volumetric results of each case. These case runs show that eight to nine 30,000 gal batches of discharge, 240,000 to 270,000 gal total, are needed to result in the first 100,000 gallons of saltcake transferred out of Tank 41. This equates to 140,000 to 160,000 gallons of inhibited water feed solution after the after initially refilling the interstitial void. The process plan should allow for 140,000 to 200,000 gallons of batch feed solution in addition to the volume to fill the interstitial void to transfer enough dissolve saltcake to reach the initial 100,000 gal of saltcake goal. The initial fill will vary greatly depending on the extent of interstitial removal and refill. The refill feed volume can range from 23,000 to 230,000 gal.

These runs show that merely measuring the change in saltcake level will not accurately indicate the quantity of saltcake transferred out of the tank. These runs show that 10,000 to 90,000 gallons of saltcake will dissolve on initial water addition before sending any to Tank 50. Further, if there is a high void fraction that is significantly larger than 0.22, then the volume of saltcake dissolved on initial filling could double.

In addition, if the drained interstitial void is not completely filled before transferring dissolved salt solution out of the tank, some amount of dissolved salt solution will continue to seep into the saltcake. In this case, the change in salt level would indicate the total of the salt transferred out of the tank and the salt that remains by filling the interstitial void.

When water is added to the top of the saltcake, the dissolution liquid begins refilling the interstitial pores. The salt around the pores dissolves to create progressively larger flow paths. As the liquid advances into the saltcake it will eventually become saturated and no longer dissolve saltcake. From this point, the liquid will still flow deeper into the saltcake, but the flow rate will be limited by the percolation rate, which is expected to be around 4 gpm.[1,2] At some point while adding dissolution water, the liquid will start to accumulate over the saltcake surface before fully saturating the interstitial void of the saltcake. When the flow stops, the liquid on the surface may appear to sorb into the saltcake, when, in fact, it is continuing to flow into the interstitial void. Planned dissolution may proceed without completely saturating the void space. During dissolution some of the dissolved saltcake is transferred into the interstitial void at the same time liquid is transferred out. A fair approximation of saltcake removed from the tank may be obtained by applying an average ratio of discharge liquid to dissolved saltcake as follows:

$$V_{saltcake} = V_{discharge} * R_{saltcake\ to\ discharge}$$

where

$V_{saltcake}$ = saltcake transferred from Tank 41,
$V_{discharge}$ = measured transfer volume from Tank 41, and
$R_{saltcake\ to\ discharge}$ = estimated ratio of gallons of saltcake dissolved per gallon of discharge.

This relationship assumes perfect saturation. Adding a percent relative saturation term corrects this deficiency:

$$V_{saltcake} = V_{discharge} * R_{saltcake\ to\ discharge} * \%_{sat}$$

where

$\%_{sat}$ = percent relative saturation defined as sodium concentration of the solution divided by the sodium concentration at saturation.

The ideal term $R_{saltcake\ to\ discharge}$ can be calculated from model results as shown in Table 8 through Table 10. A model run may be run to fit the more accurately determined parameters for void fraction and/or composition resulting from the actual dissolution of salt. In order to determine percent relative saturation, composition data from at least two supernate samples from the pump suction point are needed, one at the end of the dissolution and one around the midpoint. Initial composition data already exist as shown in Table 5.

An alternate method to determine R uses the data in Table 5 some measure of total solids in the dissolved supernate as follows:

$$R_{saltcake\ to\ discharge} = [\text{total solids}]_{saltcake} / [\text{total solids}]_{sup}$$

where:

[total solids] = concentration of total solids.

Total solids in the supernate may be estimated by using a correlation of the specific gravity to total solids. This model results provides one such correlation and more specific correlation can be generated for each batch if needed.[7]

The accuracy of this estimation method can be determined by propagating the error of the parts. The equation for $V_{saltcake}$ consists of three parts: measurement of transferred liquid, estimation of R, and $\%_{sat}$. Relative error of $V_{saltcake}$ may be estimated by:[8]

$$(e_{Vsaltcake} / V_{saltcake})^2 = (e_{Vdischarge} / V_{discharge})^2 + (e_R / R_{saltcake\ to\ discharge})^2 + (e_{\%sat} / \%_{sat})^2$$

where

e = estimated potential error.

$V_{discharge}$ is determined as follows:

$$V_{discharge} = (L_f - L_i) * C$$

where

L_f = final level in receiving tank, inches,
L_i = initial level in receiving tank, inches, and
C = waste tank calibration factor, gal./inch.

The error of $V_{discharge}$ is estimated as follows:

$$(e_{Vdischarge} / V_{discharge})^2 = (((e_{Lf})^2 + (e_{Lf})^2)^{½} / (L_f - L_i))^2 + (e_C / C)^2$$

Therefore,

$$(e_{Vsaltcake} / V_{saltcake})^2 = (((e_{Lf})^2 + (e_{Lf})^2)^{½} / (L_f - L_i))^2 + (e_C / C)^2 + (e_R / R_{saltcake\ to\ discharge})^2 + (e_{\%sat} / \%_{sat})^2$$

Liquid level in the receiving tanks is measured with reel tapes with a precision of 0.05 inches. For a 250,000 gallon discharge volume, $(L_f - L_i)$ is expect to be 71.2 inches. C for the expected receiving tank is 3510 gal/inch with e_C about ± 35. The values for $R_{saltcake\ to\ discharge}$ can range from 1.9 to 3.5. Estimating e_R at ± 0.3 for R of 2.7, and $e_{\%sat}$ at ± 5% for 80% relative saturation determination, then

$$(e_{Vsaltcake} / V_{saltcake})^2 = (((0.05)^2 + (0.05)^2)^{½} / 71.2)^2 + (35 / 3510)^2 + (0.3 / 2.7)^2 + (5\% / 80\%)^2$$

$$= 0.0164$$

$$e_{Vsaltcake} / V_{saltcake} = 0.128 \text{ or about 13\% error.}$$

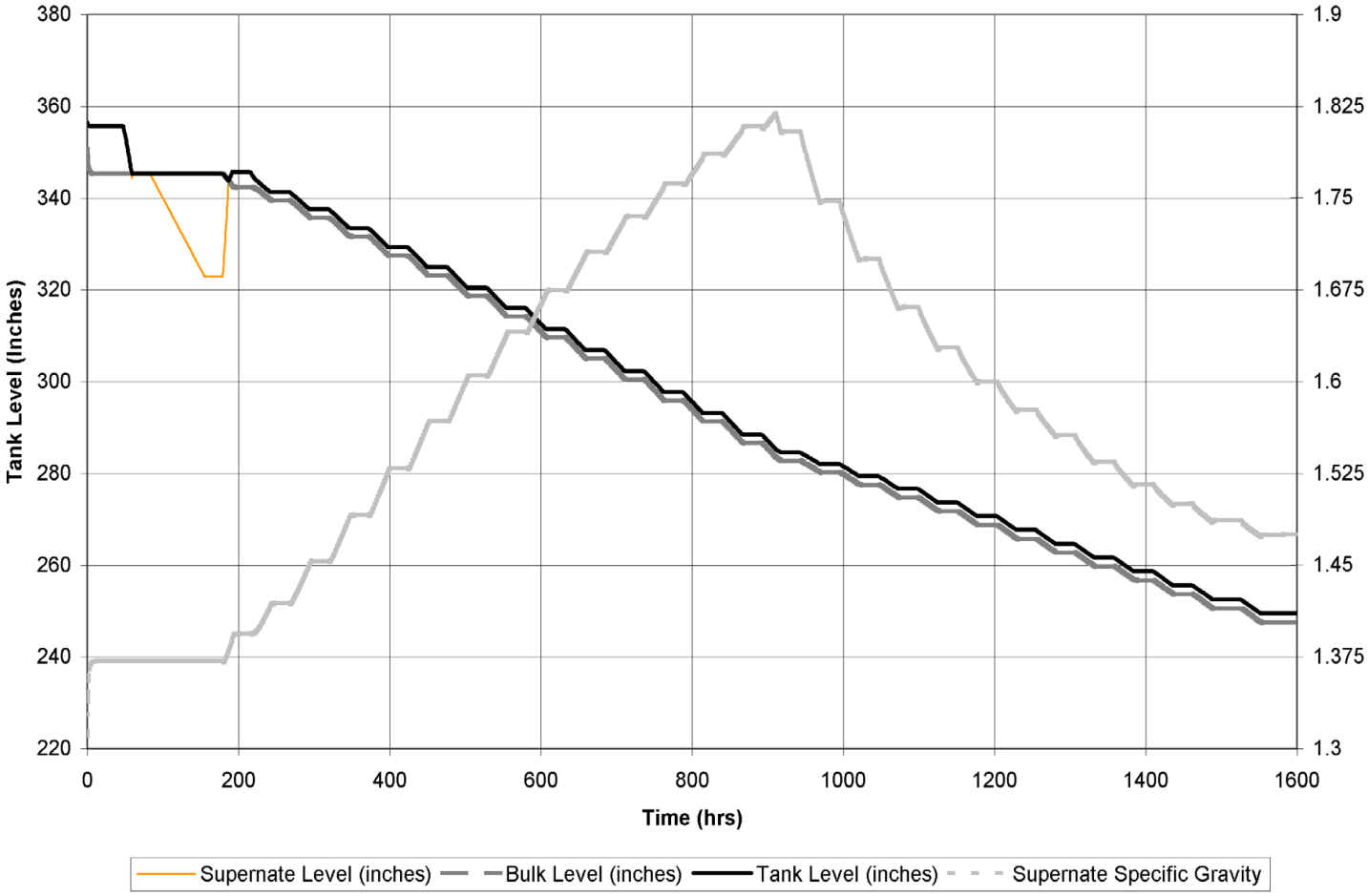

Figure 1: Results for Min Drain Case

Table 8: Batch Summary for Min Drain Case

Batch	Feed (gal)	Salt Dissolved (gal)	Ratio Feed to Salt Dissolved	Salt Solution Discharged (gal)	Ratio Discharge to Feed	Ratio Discharge to Salt Dissolved
Initial Drain	0	0		53,580		
Initial Fill	23,400	10,137	2.31	0		
Batch 1	20,688	10,215	2.03	31,080	1.50	3.04
Batch 2	24,538	13,344	1.84	31,200	1.27	2.34
Batch 3	24,245	14,182	1.71	31,200	1.29	2.20
Batch 4	24,055	14,791	1.63	31,200	1.30	2.11
Batch 5	23,928	15,231	1.57	31,200	1.30	2.05
Batch 6	23,844	15,545	1.53	31,200	1.31	2.01
Batch 7	23,788	15,766	1.51	31,200	1.31	1.98
Batch 8	23,753	15,921	1.49	31,200	1.31	1.96
Batch 9	23,730	16,028	1.48	31,200	1.31	1.95
Batch 10*	47,425	32,254	1.47	62,400	1.32	1.93
Batch 11	23,704	16,188	1.46	31,200	1.32	1.93
Batch 12	23,702	16,212	1.46	31,200	1.32	1.92
Batch 13	23,295	13,712	1.70	30,653	1.32	2.24
Batch 14	23,353	8,926	2.62	31,200	1.34	3.50
Batch 15	22,794	9,465	2.41	31,200	1.37	3.30

Batch 16	22,336	9,906	2.25	31,200	1.40	3.15
Batch 17	21,974	10,255	2.14	31,200	1.42	3.04
Batch 18	21,693	10,526	2.06	31,200	1.44	2.96
Batch 19	21,480	10,733	2.00	31,200	1.45	2.91
Batch 20	21,001	10,504	2.00	30,632	1.46	2.92
Batch 21	21,566	10,507	2.05	31,200	1.45	2.97
Batch 22	21,448	10,624	2.02	31,200	1.45	2.94
Batch 23	21,361	10,702	2.00	31,200	1.46	2.92
Batch 24	21,301	10,753	1.98	31,200	1.46	2.90
Batch 25	21,262	10,791	1.97	31,200	1.47	2.89

* Batch 10 actual sums two batches. This combination is due to the model failing to output data between these two batches.

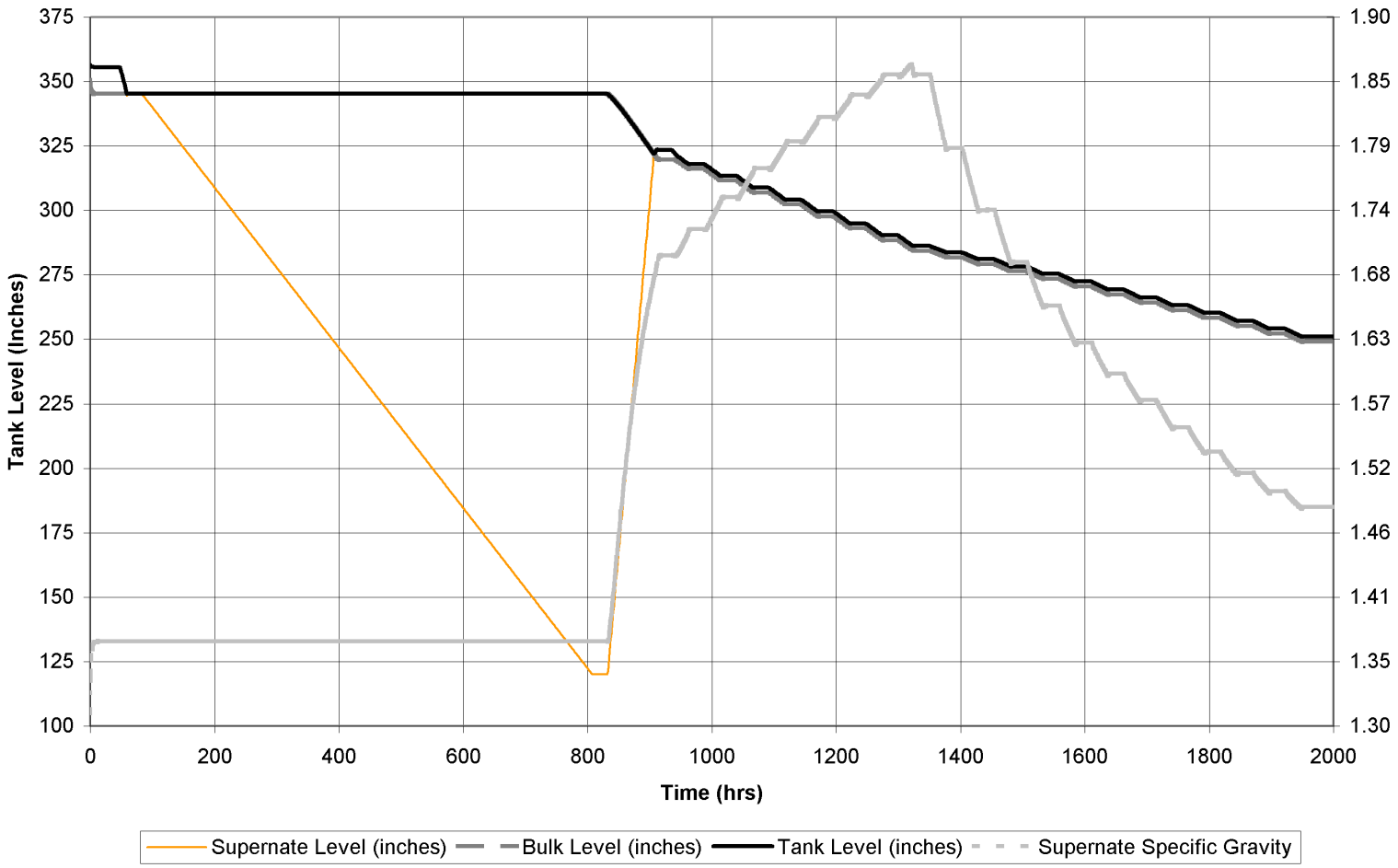

Figure 2: Results for Baseline Case

Table 9: Batch Summary for Baseline Case

Batch	Feed (gal)	Salt Dissolved (gal)	Ratio Feed to Salt Dissolved	Salt Solution Discharged (gal)	Ratio Discharge to Feed	Ratio Discharge to Salt Dissolved
Initial Drain	0	2		210,300		

Initial Fill	144,000	89,500	1.61	0		
Batch 1	18,700	12,700	1.48	31,100	1.66	2.46
Batch 2	23,700	16,100	1.47	31,200	1.32	1.93
Batch 3	23,700	16,200	1.47	31,200	1.32	1.93
Batch 4	23,700	16,200	1.46	31,200	1.32	1.93
Batch 5	23,700	16,200	1.46	31,200	1.32	1.92
Batch 6	23,700	16,200	1.46	31,200	1.32	1.92
Batch 7	23,700	16,200	1.46	31,200	1.32	1.92
Batch 8	23,000	14,400	1.59	30,300	1.32	2.10
Batch 9	23,400	8,900	2.63	31,200	1.33	3.51
Batch 10	22,800	9,400	2.42	31,200	1.37	3.31
Batch 11	22,400	9,900	2.27	31,200	1.39	3.16
Batch 12	22,000	10,200	2.15	31,200	1.42	3.05
Batch 13	21,700	10,500	2.07	31,200	1.44	2.97
Batch 14	21,500	10,700	2.01	31,200	1.45	2.92
Batch 15	21,300	10,900	1.96	31,200	1.46	2.87
Batch 16	21,500	10,600	2.03	31,200	1.45	2.95
Batch 17	21,500	10,600	2.02	31,200	1.45	2.94
Batch 18	21,400	10,700	2.00	31,200	1.46	2.92
Batch 19	21,300	10,800	1.98	31,200	1.46	2.90
Batch 20	21,300	10,800	1.97	31,200	1.47	2.89

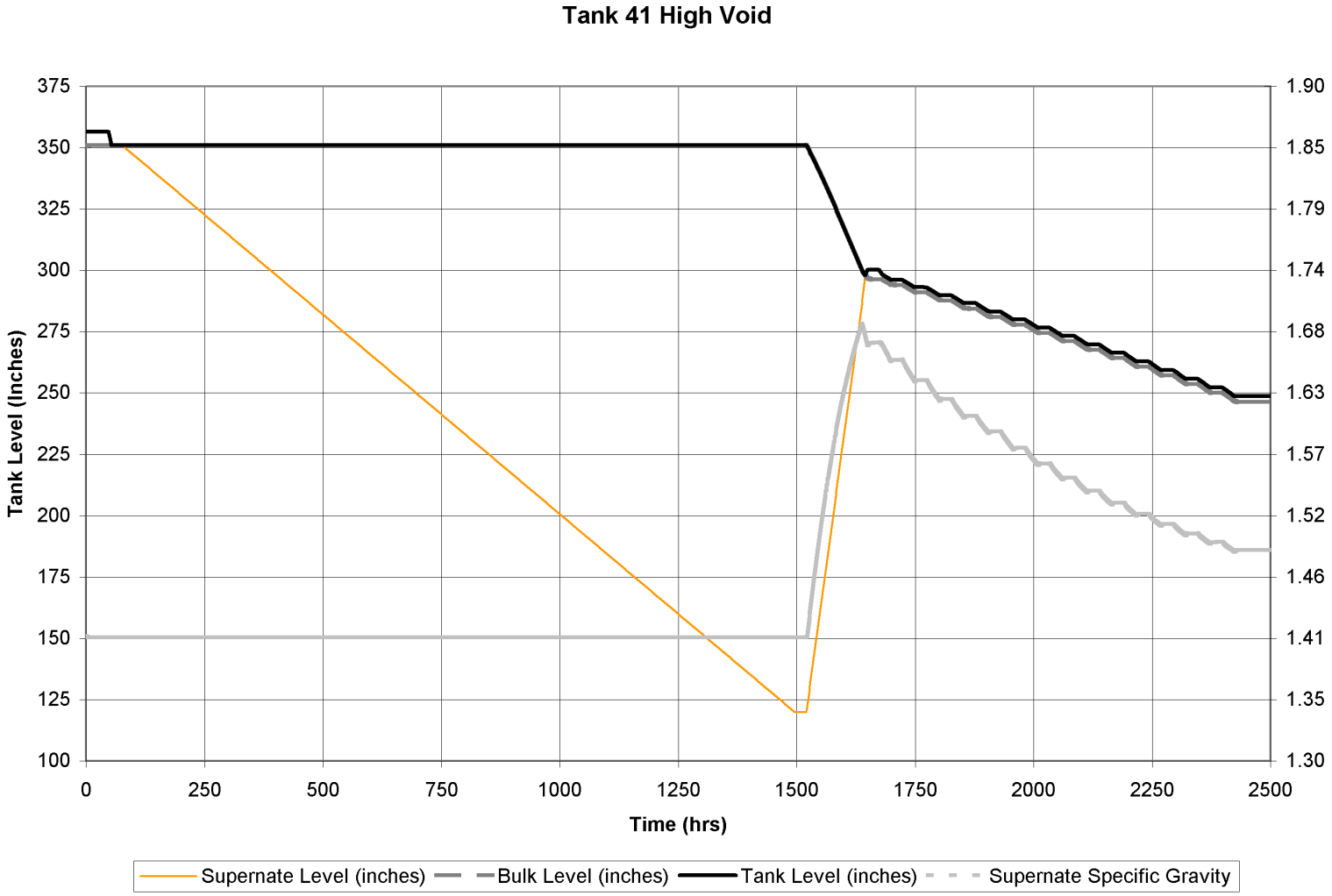

Figure 3: Results for High Void Case

Table 10: Batch Summary for High Void Case

Batch	Feed (gal)	Salt Dissolved (gal)	Ratio Feed to Salt Dissolved	Salt Solution Discharged (gal)	Ratio Discharge to Feed	Ratio Discharge to Salt Dissolved
Initial Drain	0	1		360,000		
Initial Fill	234,000	191,000	1.22	0		
Batch 1	16,400	8,200	2.00	30,100	1.83	3.67
Batch 2	21,000	10,900	1.92	30,900	1.47	2.83
Batch 3	20,800	11,300	1.84	31,200	1.50	2.76
Batch 4	20,600	11,600	1.77	31,200	1.52	2.69
Batch 5	20,300	11,800	1.71	31,200	1.54	2.64
Batch 6	20,400	11,500	1.78	31,200	1.53	2.72
Batch 7	20,400	11,600	1.75	31,200	1.53	2.68
Batch 8	20,200	11,800	1.71	31,200	1.55	2.64
Batch 9	20,000	12,000	1.67	31,200	1.56	2.60
Batch 10	19,800	12,200	1.63	31,200	1.57	2.57
Batch 11	19,700	12,300	1.60	31,200	1.58	2.54
Batch 12	19,600	12,400	1.58	31,200	1.59	2.51
Batch 13	19,500	12,500	1.56	31,200	1.60	2.50
Batch 14	19,400	12,600	1.54	31,200	1.61	2.48
Batch 15	19,300	12,700	1.53	31,200	1.62	2.47

Figure 4 shows the flow of apparent component from the solid phase, saltcake solids, to the liquid phase, supernate, for the max drain case. A negative value indicates precipitation from the supernate. The initial few hours in the graph indicates that some components are dissolving and others precipitating to adjust the initial input to an equilibrium condition. Essentially all the $NaAlO_2$, NaCl, NaOH, and $NaNO_2$ in the initial composition of the solid phase dissolves. A small amount of Na_2SO_4 and Na_2CO_3 dissolves and a small amount of $NaNO_3$ precipitates. The final composition is still substantially similar to that shown in Table 5, e.g. primarily nitrate and carbonate salts with small amounts of sulfate, phosphate, and oxalate salts as shown in Table 11. The relatively small amount of the more soluble species identified in the salt sample are likely due to imperfect estimation of supernate composition as identified in Table 5.

Table 11: Composition after Equilibrating in Mole Fraction

Component	Initial Salt Composition	Equilibrated Composition
$CsNO_3$	-	5.47E-36
Na_2CO_3	0.108548	0.1067
Na_2SiO_3	-	0
Na_2SO_4	0.007957	0.007706
Na_3PO_4	0.009012	0.009739
$NaAlO_2$	0.002204	4.38E-07
NaCl	0.000108	3.49E-07
NaF	-	3.08E-24
$NaNO_2$	0.004648	5.20E-07
$NaNO_3$	0.847128	0.8759
NaOH	0.018972	1.03E-06

Figure 4 shows additional details about which components dissolve first. The initial components that dissolve are primarily carbonate and nitrate salts with a small amount of sulfate salts. Nitrate and sulfate salts dissolve more rapidly once the carbonate salts are completely dissolved. This

behavior is similar to what is observed in numerous laboratory tests.[9,10,11,12] In one laboratory study of SRS saltcake dissolution, the carbonate did not dissolve as readily as predicted and substantial quantities remained in the saltcake until most of the nitrate salts dissolved.[13] Earlier studies[14] and salt dissolution in waste tanks did not measure chemical compositions adequately to determine chemical changes occurring at intermediate steps in the dissolution.

These earlier tests and waste dissolution tracked specific gravity as an indicator of saturation. If the composition is constant, then specific gravity would indicate extent of saturation. If the composition fluctuates, the composition needs to be known to determine relative saturation. To effectively interpret the specific gravity values obtained during dissolution, the measured value can be contrasted against the predicted values at saturation for a given composition. Predicted compositions produce a relative high specific gravity peak as dissolution

progresses. This peak corresponds to the depletion of carbonate salts from the saltcake. Tank 10 dissolution history shown in Table 12 indicate an initial high specific gravity followed by a substantially lower value during last part of the cycle. Though past interpretation of this data suggests that this is due to lower levels of saturation, that may not be the case. The change in composition may be a significant contributor to this effect.

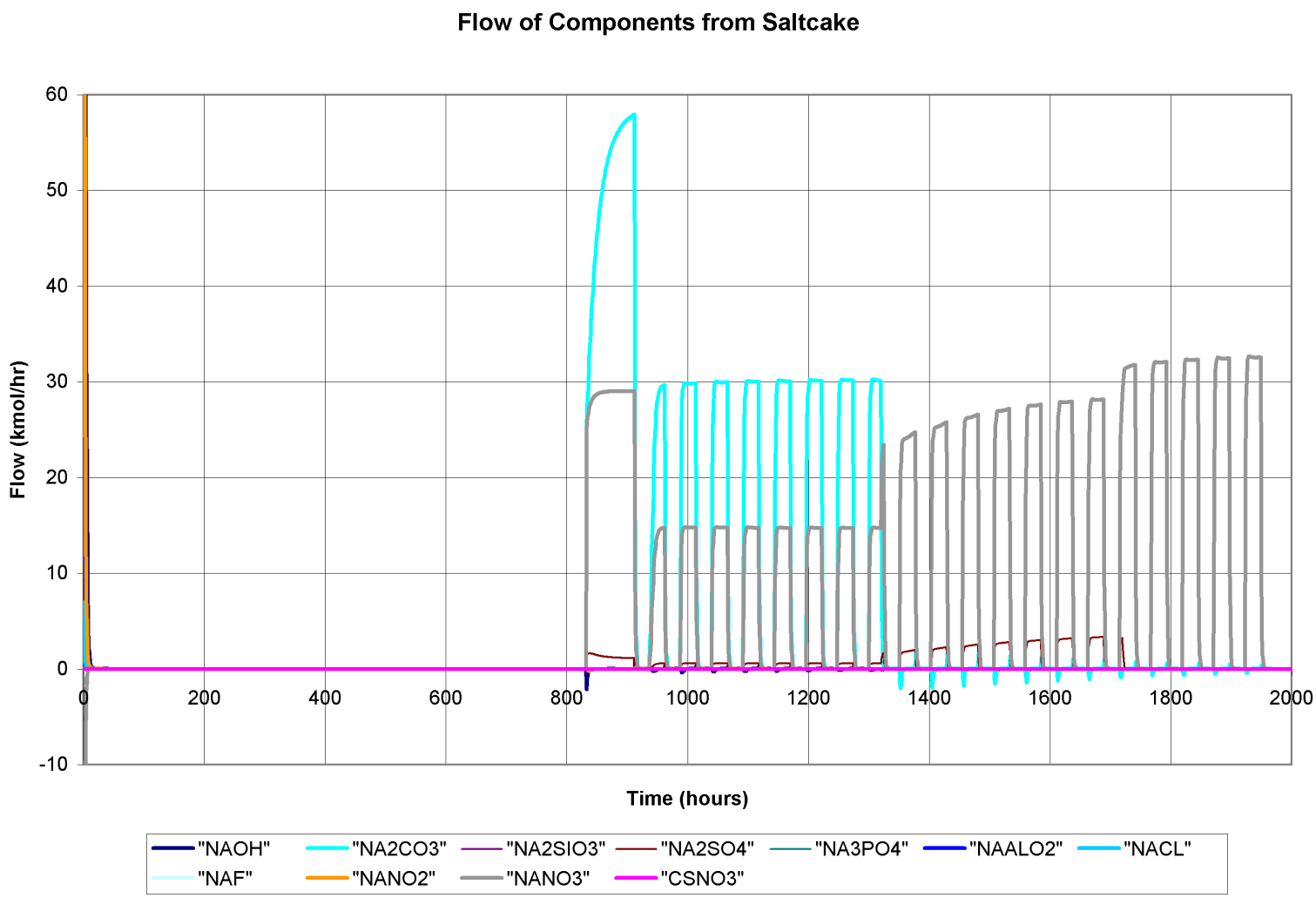

Figure 4: Flow of Components from Saltcake to Supernate for Max Drain Case

4 Model Check With Tank 10 History

This version of the model is compared to historical dissolution process as was done for the earlier version created for Tank 37 dissolution.[15] Records from Tank 10 dissolution are complete enough to reconstruct the process history and compare the actual dissolution results with the model predictions.

4.1 Tank 10 History and Model Input

Tank 10 was dissolved by flooding the saltcake with liquid from Tank 23, RBOF/RRF waste. The waste is a relatively dilute salt solution and a good substitute for inhibited water. The additional salt in the waste makes only a minor difference to the amount of salt dissolved per gallon of dissolution liquid. The original Tank 23 composition is not available and inhibited water is substituted for the dissolution water in the model, 0.01 M NaOH and 0.011 M $NaNO_2$ solution.

Once flooded, the level was maintained while simultaneously feeding dissolution water and removing saturated solution. The removal jet was buried deep in the salt bed, about 3' from the bottom of the tank. The flow rates in and out were maintained relatively low, about 10 gpm in and 15 gpm out. At the end of a batch, the feed was stopped, but the discharge was continued until a certain level was reached to allow inspections.

The history of salt dissolution from Tank 10 came from several documents.[16,17,18,19,20] The compiled timeline, including flow rate, average discharge specific gravity, and tank levels, is shown in Table 12.

Historic saltcake composition is not available, so the saltcake composition used for Tank 41 dissolution modeling is assumed. This is different from what was used in reference , which used WCS composition for saltcake. This was not adequate since a significant change in saltcake level and supernate composition would occur during initial equilibration. In addition, the saltcake void fraction is assumed to be 22%.

Table 12: Tank 10 Salt Dissolutuion Historical Timeline

	Flow In		Flow Out							
Date	Start Tank Level	Stop Tank Level	Start Tank Level	Stop Tank Level	Notes	Flow in (gpm)	Flow out (gpm)	SpG	Hours in	Hours Out
3/27/79			207	202	11000 gallons to Tank 37, Tank 10 reel tape touching salt, so no level recorded					
4/25/79	205.5				Start Batch A					
4/26/79		209	209		Maintain for 8 hrs then reduced level					
4/27/79				203.3	Stopped Batch A	8	11.8	1.43	36.2	23.5
4/28/79	203.3		203.7		Start Batch B	10.8	17.1	1.41	147.1	137.2
4/29/79										
4/30/79										
5/1/79										
5/2/79			194		Dropping 3"/day					
5/3/79		190.8		189	Stopped Batch B					
5/4/79	190.1									
5/5/79				191.214	Flush water spray added 6000 gal flush water - Batch C					
5/6/79					Start Batch D ?					
5/7/79	202				Increased level					
5/8/79					Stopped until 5/9					
5/9/79			202		Started out flow	11.2	18	1.41	100.9	40.0
5/10/79										
5/11/79		?		198.7	Stopped Batch D					
5/12/79										
5/13/79										
5/14/79										
5/15/79	197.3		?		Start Batch E	11.5	17.8	1.41	37.4	32.0
5/16/79		?		195.3	Stopped Batch E					
5/17/79										
5/18/79										
5/19/79										
5/20/79										
5/21/79	194.8		194.8		Start Batch F	11.2	17.3	1.34	72.9	91.2
5/22/79										
5/23/79										
5/24/79										
5/25/79		189.3		181.9	Stopped Batch F					
5/26/79										
5/27/79										
5/28/79										
5/29/79										
5/30/79										
5/31/79	182				In flow only to final level					
6/1/79		185								
6/2/79										

4.2 Model Output

The Tank 10 dissolution is simulated in two ways. One method simulates the batch flow rates and tank levels as input to the model. The other

method simulates the batch flow rates and flow times as input. To simulate tank levels, the input selected is based on the batch data for flow rates and ending tank levels. The batch size and time is calculated. If the tank level is not available, the batch time calculated from batch volume and flow rate is substituted. To simulate time, the model input is based on the batch time calculated from batch volume and flow rate. The levels are calculated. In both cases, the salt dissolved and batch chemistry is calculated.

Table 14 and Table 15 show the simulation results of dissolution. The simulation indicates that the historical batch flow rates and times are not consistent with the tank level data available. The exact timing of the level data may be inaccurate.

Simulating flow time produces accurate simulation of batch sizes as shown in Table 14. The model predicts volume ratios well, thus, predicts dissolved salt volume well. Simulation based on tank levels consistently under predicts the feed and discharge volumes as shown in Table 15.

This result is inherent in the available data. The input is based on the batch data available, which indicates an average flow rate, tank levels, and time. The model was designed for predicting the level and dissolution volumes based on the batch size. Since the flow rates are approximated from level changes, the historical data probably does not accurately reflect actual flow rates. If the input is restructured for batch sizes, the flow rates would need to be adjusted in order to match tank level stopping points. Given that the model predicts the important ratios well, the model would predict the salt dissolution volumes reasonably well too. The over all ratio matches well.

Predicted specific gravity is higher than measured during actual dissolution. The specific gravity drops dramatically during the last part of the dissolution, which seems to correspond to the trend observed during Tank 10 dissolution. For a given composition, the properties estimation methods used in this model over predicts the density by 4.0% to 5.6%.[21] This amount of error accounts for a significant portion of the difference. The remaining difference probably indicates the relative saturation of the dissolution liquid.

Table 13: Tank 10 Salt Dissolution Historical Results

Batch	Average Sp.G.	Average Feed (gpm)	Average Flow Out (gpm)	Feed Batch (gal)	Feed Time (hrs)	Out Flow Batch (gal)	Out Flow Time (hrs)	Salt Dissolved (gal)	Ratio Feed to Salt Dissolved	Ratio Discharge to Feed
A	1.43	8	11.8	17,400	36.3	16,600	23.4	8,000	2.18	0.95
B	1.41	10.8	17.1	95,300	147.1	140,800	137.2	62,500	1.52	1.48
C	Flush water addition			6,000						
D	1.41	11.2	18	67,800	100.9	43,200	40.0	19,800	3.42	0.64
E	1.41	11.5	17.8	25,800	37.4	34,200	32.0	13,800	1.87	1.33
F	1.34	11.2	17.3	49,000	72.9	94,600	91.1	36,000	1.36	1.93
Sum				261,300	394.5	329,400	323.8	140,100	1.87	1.26

Table 14: Tank 10 Salt Dissolution Model Results Simulating Flow Time

	Time (hrs)	Salt Bulk Level (inches)	Feed Batch (gal)	Out Flow Batch (gal)	Salt Dissolved (gal)	Ratio Feed to Salt Dissolved	Ratio Discharge to Feed
A	68 130	173.761 168.67	21,992	18,408	13,797	1.59	0.84
B	142 291	168.669 148.378	95,904	157,922	54,989	1.74	1.65
C	314 317	148.351 147.45	6,000	0	2,442	2.46	0.00
D	325 477	147.449 134.958	74,368	44,928	33,851	2.20	0.60
E	549 586	134.91 130.57	25,116	35,543	11,761	2.14	1.42
F	657 732	130.486 121.64	49,459	79,885	23,973	N/A	1.62
Sum			272,839	336,685	140,812	1.94	1.23

Table 15: Tank 10 Salt Dissolution Model Results Simulating Tank Level

Batch	Time (hrs)	Salt Bulk Level (inches)	Feed Batch (gal)	Out Flow Batch (gal)	Salt Dissolved (gal)	Ratio Feed to Salt Dissolved	Ratio Discharge to Feed
A	68 135	173.432 168.669	21,992	19,881	12,908	1.70	0.90
B	144 213	168.669 158.927	40,176	67,223	26,401	1.52	1.67
C	234 238	158.926 157.434	6,000	0	4,043	1.48	0.00
D	245 346	157.433 145.944	66,701	43,805	31,135	2.14	0.66
E	417 455	145.943 141.881	25,806	35,543	11,008	2.34	1.38
F	526 569	141.877 139.462	14,482	45,340	6,545	N/A	3.13
		Sum	175,157	211,792	92,040	1.90	1.21

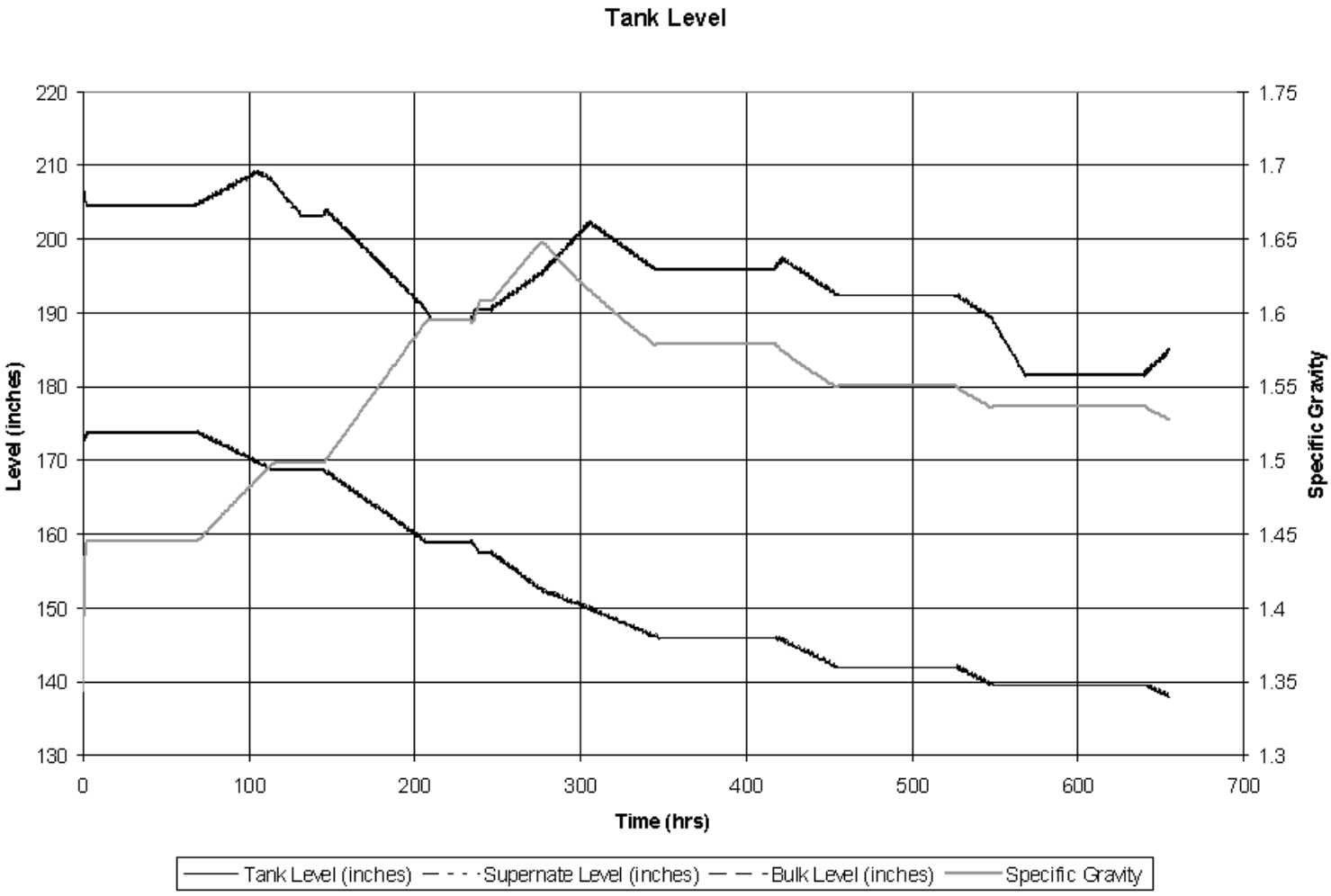

Figure 5: Tank 10 Model Results

5 Model Listing

The following model listing shows example input data for the base case:

```
Version "10.2-0";
Libraries "Modeler.acml", "SystemLibrary.acml";
Variable SMass

// kg
      value:10;    lower:0;     upper:1E20;   PhysicalQuantity : "Mass";

End
Variable SVolume

/* m3 */
      value:10;    lower:0;     upper:1E10;   PhysicalQuantity : "Volume";
End
Variable SConc_mole

/* kmol/m3 */
      value:0.5;   lower:0;     upper:1000;   PhysicalQuantity : "MolarDensity";
End
Variable SFlow_Mol

/* kmol/hr */
      value:10;    lower: -1E8;     upper:1E8;   PhysicalQuantity : "MoleFlow";

End
Densitywater AS  dens_mass;

kdis AS  RealVariable;

JetOn AS  RealVariable;

WaterOn AS  RealVariable;

SitOn AS  RealVariable;

factor AS  RealVariable;

kdis1  AS  RealVariable;
Port SaltSolution
// <variable name> as <variable type> (<default value>, description:"<description>");

 Ions                          as ComponentListName("Ions");
 Salts                         as ComponentListName("Salts");
 All                           as ComponentListName("All");
 Apparent                      as ComponentListName("Apparent");

 Fmol                          as flow_mol    (Description:"Molar flow rate");
 z(Apparent)                   as molefraction (Description:"Mole fractions", 1.0/size(Apparent));

 Fmass                 as flow_mass    (Description:"Molar flow rate");
 w(Apparent)                   as massfraction (Description:"Mass fractions", 1.0/size(Apparent));

 T                             as temperature  (Description:"Temperature");
 P                             as pressure     (Description:"Pressure");


End
Port SolidSalt
// <variable name> as <variable type> (<default value>, description:"<description>");

 Ions                          as ComponentListName("Ions");
 Salts                         as ComponentListName("Salts");
 All                           as ComponentListName("All");
 Apparent                      as ComponentListName("Apparent");

 V                             as volume (Description:"Bulk Salt Volume Remaining");
 Void                          as volume (Description:"Bulk Salt Void Volume");
 TotalMoles(apparent)          as pos_large            (description:"kmol");

 Fmol(Apparent)        as flow_mol_    (Description:"Molar flow rate");

 spg                           as RealVariable (Description:"Specific Gravity");

 T                             as temperature  (Description:"Temperature");
 P                             as pressure     (Description:"Pressure");
 H                             as enth_mol             (Description:"enthalpy - GJ/kmol");
```

```
End
Port chemistryport
// <variable name> as <variable type> (<default value>, description:"<description>");

  all                          as ComponentListName("All");

  Tmol                 as RealVariable                 (free, description:"kmol");
  z(all)               as molefraction                 (free, description:"mole fraction");
  T                    as temperature          (free, Description:"Temperature - C");
  P                    as pressure             (free, Description:"Pressure - bar");


End
Port chemistrysaltout
// <variable name> as <variable type> (<default value>, description:"<description>");

  salts                        as ComponentListName("Salts");

  Tmol                 as RealVariable                 (free, description:"kmol");
  z(salts)             as molefraction                 (free, description:"mole fraction");
  T                    as temperature          (free, Description:"Temperature - C");
  P                    as pressure             (free, Description:"Pressure - bar");


End
Port chemistryliquidout
// <variable name> as <variable type> (<default value>, description:"<description>");

  ions                         as ComponentListName("Ions");

  Tmol                 as RealVariable                 (free, description:"kmol");
  z(ions)              as molefraction                 (free, description:"mole fraction");
  T                    as temperature          (free, Description:"Temperature - C");
  P                    as pressure             (free, Description:"Pressure - bar");
End
Port SupernateConc
// <variable name> as <variable type> (<default value>, description:"<description>");

  Ions                         as ComponentListName("Ions");
  Salts                        as ComponentListName("Salts");
  All                          as ComponentListName("All");
  Apparent                     as ComponentListName("Apparent");

  equilconc(all)       as pos_large                    (free, description:"kmol/m3 = M");

//  equilconc(all)     as conc_mole                    (free, description:"kmol/m3 = M");
  actualconc(all)      as conc_mole                    (free, description:"kmol/m3 = M");

End
Model Supernate

/*




*/

  WaterOn              as output realvariable          (0, fixed, description:"On/Off Switch");
  DisState             as realvariable                 (0, fixed, description:"Disolution Step");
  InitialLevel                 as realvariable                 (0, fixed, description:"Level for Dis Batch");

/*      Dissolution States:
        0       Initial conditions
        1       After Tank Prep
        2       After Drain/remove
        3       After Add/sit
        4
        5
        6
*/
```

```
// Component lists available
  Ions                          as ComponentListName("Ions");
  Salts                         as ComponentListName("Salts");
  All                           as ComponentListName("All");
  Apparent                      as ComponentListName("Apparent");

// Ports
  Diswaterin                    as input SaltSolution  ;
  Saltin              as input SolidSalt      ;
  Superout            as output SaltSolution;
  Superproperties     as output SupernateConc          ;          // Information used only by the salt model

// Variables

// General Properties
  Densitywater                  as global dens_mass   ;
  factor              as global RealVariable            ;
  MW(all)             as molweight                      (free, description:"Molecular weights of components");
  T                   as temperature          (fixed, Description:"Temperature of waste assumed uniform within tank");
  P                   as pressure             (fixed, Description:"Pressure");
  H                             as enth_mol                       (free, Description:"enthalpy - GJ/kmol");

// Properties of just supernate
  MWaq                          as molweight                      (free, description:"Molecular weight of supernate");
  SpG_Supernate                 as RealVariable                   (free);
  Density_Supernate             as dens_mass                      (free, description:"Supernate Density");

  SupernateMoles      as pos_large                      (free, description:"kmol - Total moles of supernate");
  SuperMoles(apparent)          as pos_large                      (free, description:"kmol - moles of each apparent component in
supernate");
  Supernatez(apparent)          as molefraction         (initial, description:"mole fraction of each apparent component in supernate");
  Supernatezext(all)  as molefraction         (free, description:"mole fraction of each apparent component in supernate - remaining
components set to 0");

  liqratiosuper       as ratio_               (free, description:"True liquid to apparent liquid mole ratio");
  solratioSuper                 as ratio_               (free, description:"True solid to true liquid mole ratio");
  Supernatez_actual(all) as molefraction      (free, description:"Calculated mole fraction of actual components in supernate");
  Supernatez_truecomp(all) as molefraction  (free, description:"Mole fraction of actual components in supernate returned by ptruecomp");
  Actualconc(all)     as conc_mole                      (free, description:"Calculated actual supernate concentrations");

  SupernateConc(apparent)       as conc_mole                      (free, description:"Apparent concentration");
// Properties of whole tank - True composition
  Total_Moles                   as pos_large                      (free, description:"kmol - Total moles in tank");
  TotalMoles(apparent)          as pos_large                      (free, description:"kmol - Total moles of each component in tank");
  Totalz(apparent)    as molefraction         (free, description:"mole fraction of apparent components in tank");
  Totalzext(all)      as molefraction         (free, description:"mole fraction of apparent components in tank - other components set to 0");
  liqratioTotal                 as ratio_               (free, description:"True liquid to apparent liquid mole ratio");
  solratioTotal                 as ratio_               (free, description:"True solid to true liquid mole ratio");


// Properties of supernate at theoretical equilibrium
  MWaq_equil                    as molweight                      (free, description:" Molecular weight of supernate at equilibrium");
  Supernatez_equil(all) as molefraction       (free, description:"mole fraction of true components in supernate at equilibrium");
  supersumz                     as pos_large                      (free, description:"sum check on mole fractions should always be 1");
  equilz(ions)                  as molefraction         (free, description:"mole fraction of true supernate components normalized with ions
component list only at equilibrium");
  totalmolesequil     as pos_large                      (free, description:"kmol - Total moles in tank at equilibrium");
  equildens           as dens_mass                      (free, description:"mass density of supernate at equilibrium");
  equil_mol_dens      as dens_mol                       (free, description:"molar density of supernate at equilibrium");
  equilconc(all)      as pos_large                      (free, description:"kmol/m3 = M - molar concentration of true supernate
components at equilibrium ");

// Waste Volumes
  SupernateVolume     as volume               (initial, description:"volume of supernate");
  SupernateVolumeEquil          as volume               (free, description:"volume of supernate at theoretical equlibrium");
  Vdischargerate      as output RealVariable            (fixed, description:"m3/hr - volumetric flow rate out");
  TotalDischargeVol   as RealVariable                   (initial, description:"m3 - total volumetric flow out");

  GoalSpG             as RealVariable                   (Fixed, description:"specific gravity goal of salt dissolution");

  TotalVolume                   as volume               (free, description:"Total volume of waste in tank");
  TankLevel                     as length               (free, description:"tank level equivalent to measured");
  TankLevelInches     as RealVariable                   (free, description:"inches - tank level as to measured");
  SupernateLevel      as output RealVariable            (free, description:"inches - level of supernate in tank as if always measurable");
  Tankcalibration     as RealVariable                   (fixed, description: "gallons/inch");

  Procedures:

        call(MW) = pMolWeights() all;
        call(MWaq) = pMolWeight(Supernatez) apparent;
        call(Density_Supernate) = pDens_Mass_Liq(T,P, Supernatez) apparent;
        call(H) = pEnth_Mol_Liq(T,P, Supernatez) apparent;
```

SupernateChemistry:

```
// apparent concentrations
SupernateConc(apparent) = SuperMoles(apparent)/SupernateVolume;  // kmoles/m3 = moles/L

// Calculate true components in supernate based on supernate apparent
call(Supernatez_truecomp(all), solratioSuper, liqratiosuper) = pTrueComp(T, P, Supernatezext(all)) all;
Supernatezext(apparent) = Supernatez(apparent);
Supernatezext(all-apparent) = 0;

// Calculate actual conc of ions based on apparent supernate
// assume perfect dissolution of salts for stoichiometry calculations

If SupernateVolume > 0 then

        // NaNO3
        actualconc("NO3-") = (SuperMoles("NaNO3") + SuperMoles("CsNO3")) / SupernateVolume;
        // CsNO3
        actualconc("Cs+") = SuperMoles("CsNO3") / SupernateVolume;
        // NaNO2
        actualconc("NO2-") = SuperMoles("NaNO2") / SupernateVolume;
        // NaOH
        actualconc("OH-") = SuperMoles("NaOH") / SupernateVolume;
        // Na2SO4
        actualconc("SO4--") = SuperMoles("Na2SO4") / SupernateVolume;
        // Na2CO3
        actualconc("CO3--") = SuperMoles("Na2CO3") / SupernateVolume;
        // Na3PO4
        actualconc("PO4-") = SuperMoles("Na3PO4") / SupernateVolume;
        // NaCl
        actualconc("Cl-") = SuperMoles("NaCl") / SupernateVolume;
        // NaF
        actualconc("F-") = SuperMoles("NaF") / SupernateVolume;
        // NaAlO2
        actualconc("AlO2-") = SuperMoles("NaAlO2") / SupernateVolume;
        // Na2SiO3
        actualconc("Na2SiO3") = SuperMoles("Na2SiO3") / SupernateVolume;
        // H2O
        actualconc("H2O") = SuperMoles("H2O") / SupernateVolume;
        //Na - this one assumes sodium concentration from equilibrium calc
        actualconc("Na+") = Supernatez_truecomp("Na+") * SupernateMoles * liqratiosuper / SupernateVolume;

      //zero out others
      actualconc(all - "Na2SiO3" - "AlO2-" - "F-" - "Cl-" - "PO4-" - "CO3--" - "SO4--" - "OH-" - "NO2-" - "Cs+" - "NO3-
" - "H2O" - "Na+") = 0;

else

      actualconc(all) = 0;

endif

// Calculate true component concentrations
Supernatez_actual(ions) * sigma(actualconc(ions) * SupernateVolume) = actualconc(ions) * SupernateVolume;
Supernatez_actual(all - ions) = 0;
Superproperties.actualconc(all) = actualconc(all);
```

EquilibriumChemistry:

```
// calculate equilibrium composition of whole tank
call(Supernatez_equil(all), solratioTotal, liqratioTotal) = pTrueComp(T, P, Totalzext(all)) all;
TotalMoles(apparent) = SaltinTotalMoles(apparent) + SuperMoles(apparent);
Total_Moles = sigma(TotalMoles(apparent));
Totalz(apparent) * Total_Moles = TotalMoles(apparent);
Totalzext(apparent) = Totalz(apparent);
Totalzext(all-apparent) = 0;

supersumz = sigma(Supernatez_equil(all));  // check for sum = 1

// Calculate equilibrium moles and volume - use water as a reference i.e. moles water remain constant
totalmolesequil * Supernatez_equil("H2O") = TotalMoles("H2O");
SupernateVolumeEquil * equil_mol_dens = totalmolesequil;
call(equil_mol_dens) = pDens_mol_Liq(T, P, equilz) ions;
equilz(ions) * sigma(Supernatez_equil(ions)) = Supernatez_equil(ions);

call(equildens) = pDens_Mass_Liq(T, P, equilz) ions;
call(MWaq_equil) = pMolWeight(equilz) ions;

// Calculate Equilibrium Concentrations
equilconc(ions) * SupernateVolumeEquil = equilz(ions) * totalmolesequil;
equilconc(all - ions) = 0;
Superproperties.equilconc(all) = equilconc(all);
```

```
 Supernateout:

        Superout.z(apparent) = Supernatez(apparent);

        Superout.T = T;
        Superout.P = P;

 SupernateMoleBalance:

        SuperMoles(apparent) = Supernatez(apparent) * SupernateMoles;

        $SupernateMoles = Diswaterin.Fmol + sigma(Saltin.Fmol(Apparent)) - Superout.Fmol;   // moles in + moles in - moles out

        $SuperMoles(Apparent) = Diswaterin.Fmol * Diswaterin.z(Apparent) - Superout.Fmol * Superout.z(Apparent)
                + Saltin.Fmol(Apparent);

 MassBalance:

        Superout.Fmass = Sigma(Superout.z(apparent) * Superout.Fmol * MW(apparent));

        Superout.w(componentlist) = Superout.z(apparent) * MW(apparent) / MWaq;

 VolumeEquation:

        SpG_Supernate* Densitywater = Density_Supernate;    /* Identified salt removal goal to meet outlet SpG */

        Vdischargerate * Density_Supernate = sigma(Superout.Fmol * Superout.z(apparent) * MW(apparent));

        $TotalDischargeVol = Vdischargerate;  //discharge totalizer

        SupernateVolume * Density_Supernate = Sigma(SuperMoles(apparent) * MW(apparent));

        if  SupernateVolume - Saltin.void > 0 then
                TotalVolume = Saltin.V + SupernateVolume - Saltin.void;
        else
                TotalVolume = Saltin.V;
        endif
 LevelEquations:

        TankLevel = TotalVolume / (Tankcalibration * 0.003785412 * 39.36996); /* convert gal/inch to cubic meters/meter */

        TankLevelInches = TankLevel * 39.36996; // convert to inches

        if SupernateVolume - Saltin.void > 0 then
                SupernateLevel = TankLevelInches;
        else
                SupernateLevel = SupernateVolume / (Saltin.void / TankLevelInches); // calculate void in cubic meters/inch
        endif


//SYSTEM SECTION - WARNING: DO NOT EDIT
  Current_Icon : "System";
DefaultForm : "AllVariables";

//SYSTEM SECTION END
End
Model Feeder

/*

This model outputs a solution in the format of SaltSolution.  The total output volume is tracked by FeedVol.
Volumetric flow rate must be specified.

This model assumes composition of solution as mole fractions and calculates mass fractions.  The opposite could be accomodated.

Temperature and pressure are assumed constant.

*/

// variable to hold current flow rate for control task
  CurrentVrate                  as RealVariable                 (fixed, description:"m3/hr");

// Component list names available
  Ions                          as ComponentListName("Ions");
  Salts                         as ComponentListName("Salts");
  All                           as ComponentListName("All");
  Apparent                      as ComponentListName("Apparent");
```

```
// Ports
  p_ou                    as output SaltSolution;
  SupernateLevel     as input control_signal        (free);
  Supernateflow           as input control_signal        (free);
  SaltLevel          as input control_signal        (free);
  Dissolutionstate   as input control_signal        (free);

// Molar composition
  Fmol                    as flow_mol                    (free, description:"kmol/hr");
  Lz(Apparent)            as molefraction        (fixed, Description:"Mole fractions");

// Mass Composition
  Fmass                   as flow_mass                   (free, description:"kg/hr");
  Lw(Apparent)            as massfraction        (free, Description:"Mass fractions", 1.0/size(componentlist));

// Properties
  LSpG                    as RealVariable                (free);
  Ldens                   as dens_mass                   (free, description:"kg/m3");
  Densitywater       as global dens_mass     ;

  MW(Apparent)            as molweight                   (free, description:"kg/kmole");
  MWsol                   as molweight                   (free, description:"kg/kmole");

  T                  as temperature         (fixed, Description:"Temperature - C");
  P                  as pressure            (fixed, Description:"Pressure - bar");
  H                       as enth_mol                    (free, Description:"enthalpy - GJ/kmol");

// Volumeteric flow
  Vrate              as RealVariable                (free, description:"m3/hr");
  Fixedrate               as RealVariable                (fixed, description:"m3/hr");
  FeedVol            as RealVariable                (initial, description:"m3");

  Layer              as RealVariable                (fixed, description:"m3/hr");
// Flow control  added for Tank 41 modeling only

            // as supernate layer increases, feed rate decreases and vice versa
            // "0.6" is approximately expected steady state feed rate relative to supernate discharge
  if Dissolutionstate == 1.0 then
            Vrate = Supernateflow * 0.6 * (2*layer - (SupernateLevel-SaltLevel))^2 / layer^2 ;
  else
            Vrate = Fixedrate;
  endif

// Specify output variables
  OutputComp:

            p_ou.Fmol = Fmol;
            p_ou.z(Apparent) = Lz(Apparent);

            p_ou.Fmass = Fmass;
            p_ou.w(Apparent) = Lw(Apparent);

            p_ou.T = T;
            p_ou.P = P;

// Relate volumetric flow to molar flow and composition
  VolFlowrates:

            $FeedVol = Vrate;  //Feed totalizer

            Vrate * Ldens = sigma(Fmol * Lz(Apparent) * MW(Apparent));

// Relate volumetric flow to mass flow and composition
  MassFlowrates:

            Vrate * Ldens = Fmass;

  MassFractions:

            Lw(Apparent) * Fmass = Fmol * Lz(Apparent) * MW(Apparent);

// Calculate properties
  CalcProperties:
```

```
            LSpG * Densitywater = Ldens;

            call(MW) = pMolWeights();
            call(MWsol) = pMolWeight(Lz);
            call(Ldens) = pDens_Mass_Liq(T, P, Lz);
            call(H) = pEnth_Mol_Liq(T, P, Lz);


//SYSTEM SECTION - WARNING: DO NOT EDIT
  Current_Icon : "System";
DefaultForm : "AllVariables";

//SYSTEM SECTION END
End
Model SaltCake

/*

This model describes the saltcake layer in a waste tank.

Outputs include salt dissolution flow rate for individual apparent components based on inputs of supernate actual and equilibrium
concentrations.  Outputs also include saltcake composition and volume data.  Dissolution rates depend on a dissolution rate constant and
difference between actual and equilibrium concentrations.

To assure the dissolution rate does not overwhelm the model engine and overshoot zero mass, a ratio of moles to moles + a factor was
added to the dissolution rate equation.  This assures asymptotic approach to zero material remaining.  As long as the factor is small, the
added ratio is mathematically insignificant until the remaining moles become small.

            Assumption:  Salt solids = salt apparent composition.

*/

// Available component lists
  Ions                          as ComponentListName("Ions");
  Salts                         as ComponentListName("Salts");
  All                           as ComponentListName("All");
  Apparent                      as ComponentListName("Apparent");

// Ports
  Saltout               as output SolidSalt     ;
  Superin               as input SupernateConc          ;

// Variables:

// Global Variables
  Densitywater                  as global dens_mass   ;
  factor                as global RealVariable          ;  // dissolution factor to assure dissolution converges mathmatically

// Dissolution rate constants
  kdis(apparent)        as RealVariable                 (1000, fixed, description:"kmol/hr / kmol/m3 - Dissolution rate constant");

// General properties
  MW(Apparent)                  as molweight                    (free, description:"kg/kmole");
  MWsol                 as molweight                    (free, description:"kg/kmole");
  T                             as temperature          (fixed, Description:"Temperature of saltcake");
  P                             as pressure             (fixed, Description:"Pressure - bar");
  H                             as enth_mol                     (free, Description:"enthalpy - GJ/kmol");
  BulkSaltSpG                   as RealVariable                 (free);
  SaltVoid              as RealVariable                 (fixed, description:"saltcake void fraction");
  Salt_dens             as dens_mass                    (free, description:"salt crystalline mass density");
  Salt_mol_dens                 as dens_mol                     (free, description:"saltcake crystalline molar density");

//  Saltcake volume
  BulkVolume                    as volume               (initial, description:"Saltcake volume");
  BulkLevel                     as output RealVariable          (free, description:"inches - measured level");
  Tankcalibration       as RealVariable                 (3510, fixed, description: "gallons/inch");

// Salt composition
  TotalSaltMoles        as pos_large                    (free, description:"Total moles of apparent components in the saltcake");
  SaltMoles(Apparent)           as pos_large                    (free, description:"Moles of each apparent component in the saltcake ");
  Saltz(Apparent)       as molefraction                 (initial, description:"Mole fraction of apparent components in saltcake ");


  Procedures:

            // Property calculations
            call(MW) = pMolWeights() apparent;
            call(MWsol) = pMolWeight(Saltz) apparent;
            call(H) = pEnth_Mol_sol(T, P, Saltz) apparent;
            call(Salt_dens) = pDens_Mass_sol(T, P, Saltz) apparent;
            call(Salt_mol_dens) = pDens_mol_sol(T, P, Saltz) apparent;
```

```
SaltMoleBalance:

// salt flows through only one port in either direction - assume flow out for saltcake system is positive for consistency of signs

        $TotalSaltMoles = - sigma(Saltout.Fmol(Apparent));

        SaltMoles(Apparent) = TotalSaltMoles * Saltz(Apparent);

        $SaltMoles(Apparent) = - Saltout.Fmol(Apparent);

        // incase more components are added to the component list, the model will still function with:
        $SaltMoles(Apparent - "NaNO3"-"CsNO3"-"NaNO2"-"NaOH"-"Na2SO4"-"Na2CO3"-"Na3PO4"-"NaCl"-"NaF"-"NaAlO2"-
"Na2SiO3") = 0;

SaltDissolutionRates:

// Each apparent component must be matched with ionic species to model apparent affect on dissolution rates for each component.

        // NaNO3
        $SaltMoles("NaNO3") = - kdis("NaNO3") * (Superin.equilconc("NO3-") - Superin.actualconc("NO3-"))
                * SaltMoles("NaNO3") / (SaltMoles("NaNO3") + factor);          // disrate

        // CsNO3
        $SaltMoles("CsNO3") = - kdis("CsNO3") * (Superin.equilconc("Cs+") - Superin.actualconc("Cs+"))
                * SaltMoles("CsNO3") / (SaltMoles("CsNO3") + factor);          // disrate

        // NaNO2
        $SaltMoles("NaNO2") = - kdis("NaNO2") * (Superin.equilconc("NO2-") - Superin.actualconc("NO2-"))
                * SaltMoles("NaNO2") / (SaltMoles("NaNO2") + factor);          // disrate

        // NaOH
        $SaltMoles("NaOH") = - kdis("NaOH") * (Superin.equilconc("OH-") - Superin.actualconc("OH-"))
                * SaltMoles("NaOH") / (SaltMoles("NaOH") + factor);                    // disrate

        // Na2SO4
        $SaltMoles("Na2SO4") = - kdis("Na2SO4") * (Superin.equilconc("SO4--") - Superin.actualconc("SO4--"))
                * SaltMoles("Na2SO4") / (SaltMoles("Na2SO4") + factor);        // disrate

        // Na2CO3
        $SaltMoles("Na2CO3") = - kdis("Na2CO3") * (Superin.equilconc("CO3--") - Superin.actualconc("CO3--"))
                * SaltMoles("Na2CO3") / (SaltMoles("Na2CO3") + factor);        // disrate

        // Na3PO4
        $SaltMoles("Na3PO4") = - kdis("Na3PO4") * (Superin.equilconc("PO4-") - Superin.actualconc("PO4-"))
                * SaltMoles("Na3PO4") / (SaltMoles("Na3PO4") + factor);        // disrate

        // NaCl
        $SaltMoles("NaCl") = - kdis("NaCl") * (Superin.equilconc("Cl-") - Superin.actualconc("Cl-"))
                * SaltMoles("NaCl") / (SaltMoles("NaCl") + factor);            // disrate

        // NaF
        $SaltMoles("NaF") = - kdis("NaF") * (Superin.equilconc("F-") - Superin.actualconc("F-"))
                * SaltMoles("NaF") / (SaltMoles("NaF") + factor);              // disrate

        // NaAlO2
        $SaltMoles("NaAlO2") = - kdis("NaAlO2") * (Superin.equilconc("AlO2-") - Superin.actualconc("AlO2-"))
                * SaltMoles("NaAlO2") / (SaltMoles("NaAlO2") + factor);        // disrate

        // Na2SiO3
        $SaltMoles("Na2SiO3") = 0;  //- kdis("Na2SiO3")                                   // disrate

        // H2O


VolumeEquation:

        BulkSaltSpG * Densitywater = Salt_dens * (1 - SaltVoid);

        BulkVolume * (BulkSaltSpG * Densitywater) = TotalSaltMoles * MWsol;

        BulkLevel = BulkVolume/(Tankcalibration * 0.003785412); // need to convert gal/inch to cubic meters/inch

DataOutFlows:

        Saltout.V = BulkVolume;
        Saltout.void = BulkVolume * SaltVoid;

        Saltout.TotalMoles(apparent) = SaltMoles(Apparent);
        Saltout.SpG = BulkSaltSpG;

        Saltout.H = H;
        Saltout.T = T;
        Saltout.P = P;
```

```
//SYSTEM SECTION - WARNING: DO NOT EDIT
  Current_Icon : "System";
DefaultForm : "AllVariables";

//SYSTEM SECTION END
End
Model Jet

/*

This model describes the dilution from a jet.

For a given input, the output is diluted by a fixed volume percent assumed for jet operation.

*/

// Available component lists
  Ions                    as ComponentListName("Ions");
  Salts                   as ComponentListName("Salts");
  All                     as ComponentListName("All");
  Apparent                as ComponentListName("Apparent");

// Ports
  Superin          as input SaltSolution          ;
  Superout         as output SaltSolution         ;

// Variables:

// General Properties
  Densitywater                as global dens_mass          ;
  Supernatedensin     as dens_mass                 (free, description:"mass density of in flow");
  Supernatedensout    as dens_mass                 (free, description:"mass density of out flow");
  MW(all)             as molweight                 (free, description:"Molecular weight");
  MWsol                       as molweight                 (free, description:"Molecular weight of solution in jet");

// Assume that the jet has a fixed volume in order to create a consistent material balance
  JetVol              as volume                (0.1, fixed, description:"volume of jet");
  Total_Moles                 as RealVariable                  (0, free, description:"kmol - moles in jet");
  TotalMoles(apparent)        as RealVariable                  (free, description:"kmol - moles in jet");
  Totalz(apparent)    as molefraction          (0, initial, description:"composition in jet");

  VolIn               as flow_vol                      (free, description:"calculated volumetric flow in");
  Volout              as flow_vol                      (free, description:" calculated volumetric flow out");
  SteamIn             as flow_mol                      (0, initial, Description:"Steam flow in");
  JetDilution                 as vol_percent                   (1.04, fixed, description:"Dilution Factor");


  Procedures:

          call(MW) = pMolWeights() all;
          call(MWsol) = pMolWeight(Superout.z(Apparent)) Apparent;
          call(Supernatedensin) = pDens_Mass_Liq(Superin.T, Superin.P, Superin.z(Apparent));
          call(Supernatedensout) = pDens_Mass_Liq(Superout.T, Superout.P, Superout.z(Apparent));

  VolumetricAdjustmentCalc:

          VolIn * Supernatedensin = Sigma(Superin.z(Apparent) * Superin.Fmol * MW(Apparent));

          VolOut = VolIn * JetDilution;

          VolOut * Supernatedensout = Sigma(Superout.z(Apparent) * Superout.Fmol * MW(Apparent));

          JetVol * Supernatedensout = Total_Moles * MWsol;

  MoleBalance:

          Totalz(apparent) * Total_Moles = TotalMoles(apparent);

          $Total_Moles = Superin.Fmol + SteamIn - Superout.Fmol;

          // Balance water:
          $TotalMoles("H2O") = Superin.z("H2O") * Superin.Fmol + SteamIn - Superout.Fmol * Superout.z("H2O");

          // Balance all but water:
          For I in [Apparent - "H2O"] do
                  $TotalMoles(I) = Superin.z(I) * Superin.Fmol - Superout.Fmol * Superout.z(I);
          Endfor

  Massout:

          Superout.Fmass = Sigma(Superout.z(Apparent) * Superout.Fmol * MW(Apparent));

          Superout.w(Apparent) = Superout.z(Apparent) * MW(Apparent) / MWsol;
```

```
  OutputStreams:

            Superout.z(apparent) = Totalz(apparent);

            Superout.T = Superin.T;
            Superout.P = Superin.P;


//SYSTEM SECTION - WARNING: DO NOT EDIT
  Current_Icon : "System";
DefaultForm : "AllVariables";

//SYSTEM SECTION END
End

//SYSTEM SECTION - WARNING: DO NOT EDIT
Globals as Table;
Globals : Text { Version : 1
SizeX : 7848
SizeY : 4704
ShowAllVariables : True
ShowAllAttributes : False
ExpandAll : True
ShowRegistryAttributes : True
VariablesPaths : [ ]
AttributesPaths : [ Value Spec Description ]
ColumnWidths : [ 2436 1224 1224 1080 ]
}

EndText;

//SYSTEM SECTION END

FLOWSHEET
 Feed as Feeder;
 TankSalt as SaltCake;
 TankSuper as Supernate;
 TkTransferJet as Jet;
 SupernateProperties as Connection;
 DissolvingSalt as Connection;
 DissolvedSalt as Connection;
 JetDilutedSaltSol as Connection;
 DissolutionFeed as Connection;
 S1 as ControlSignal;
 S2 as ControlSignal;
 S3 as ControlSignal;
 S5 as ControlSignal;
 Connect TankSuper.Superproperties and TankSalt.Superin with SupernateProperties;
 Connect TankSalt.Saltout and TankSuper.Saltin with DissolvingSalt;
 Connect TankSuper.Superout and TkTransferJet.Superin with DissolvedSalt;
 Connect TkTransferJet.Superout with JetDilutedSaltSol;
 Connect Feed.p_ou and TankSuper.Diswaterin with DissolutionFeed;
 Connect TankSalt.BulkLevel via TankSalt.OutputSignal and Feed.SaltLevel via Feed.InputSignal with S1;
 Connect TankSuper.SupernateLevel via TankSuper.OutputSignal and Feed.SupernateLevel via Feed.InputSignal with S2;
 Connect TankSuper.Vdischargerate via TankSuper.OutputSignal and Feed.Supernateflow via Feed.InputSignal with S3;
 Connect TankSuper.WaterOn via TankSuper.OutputSignal and Feed.Dissolutionstate via Feed.InputSignal with S5;

CONSTRAINTS

Densitywater as global dens_mass          (1000, fixed, description:"kg/m3");
factor as global RealVariable             (1, fixed, description:"rate factor based on moles");

componentlist: "All";

Within Feed
```

```
// feed composition
lz("CsNO3"): 0, FIXED;
lz("H2O"): 0.999621705, FIXED;
lz("Na2CO3"): 0, FIXED;
lz("Na2SiO3"): 0, FIXED;
lz("Na2SO4"): 0, FIXED;
lz("Na3PO4"): 0, FIXED;
lz("NaAlO2"): 0, FIXED;
lz("NaCl"): 0, FIXED;
lz("NaF"): 0, FIXED;
lz("NaNO2"): 0.000168155, FIXED;
lz("NaNO3"): 0, FIXED;
lz("NaOH"): 0.000180141, FIXED;
//lz("PuO2"): 0.0, FIXED;

// properties
LSpG: 1.00, free;
Fixedrate: 0.0, fixed;
FeedVol: 0.0, initial;
T: 25, Fixed;
P: 1, Fixed;

// thickness in inches of liquid layer over saltcake
layer: 2.0, fixed;

EndWithin


Within TankSalt

// constraint to avoid failure to converge
BulkVolume >= 0.000001;
// initial saltcake composition
Saltz("CsNO3"): 0, initial;
Saltz("H2O"): 0, initial;
Saltz("Na2CO3"): 0.108548, initial;
Saltz("Na2SiO3"): 0.0, initial;
Saltz("Na2SO4"): 0.007957, initial;
Saltz("Na3PO4"): 0.009012, initial;
Saltz("NaAlO2"): 0.002204, initial;
Saltz("NaCl"): 0.000108, initial;
Saltz("NaF"): 0.0, initial;
Saltz("NaNO2"): 0.004648, initial;
Saltz("NaNO3"): 0.847128, initial;
Saltz("NaOH"): 0.018972, initial;
//Saltz("PuO2"): 0.0, initial;


SaltVoid: 0.22, FIXED;  // void fraction

Saltout.Fmol: 0, free; // kmole/hr

BulkVolume: 4659.679, initial; //m3 = 1,230,957 gallons = 350.7"


EndWithin
```

```
Within TankSuper

SupernateVolume >= 0.000001;

Supernatez("CsNO3"): 2.6e-5, initial;
Supernatez("H2O"): 0.850353, initial;
Supernatez("Na2CO3"): 0.003878, initial;
Supernatez("Na2SiO3"): 0.0, initial;
Supernatez("Na2SO4"): 0.000192, initial;
Supernatez("Na3PO4"): 0.000734, initial;
Supernatez("NaAlO2"): 0.006297, initial;
Supernatez("NaCl"): 0.000654, initial;
Supernatez("NaF"): 0.0, initial;
Supernatez("NaNO2"): 0.016214, initial;
Supernatez("NaNO3"): 0.029051, initial;
Supernatez("NaOH"): 0.092528, initial;
//Supernatez("PuO2"): 0.0, initial;


GoalSpG:  1.4, Fixed;


// SupernateVolume:  1102.193, initial; //m3 == current tank level 356.5" – 1,251,315-1,230,957*.78=291,168.54 gal

SupernateVolume:  1102.193, initial;

Tankcalibration:  3510, Fixed;
Vdischargerate:  0, fixed;

EndWithin


END

Task BatchStart RUNS once WHEN time == 0.

// For event driven tasks, <Trigger> can be either:
//   Runs When <condition> e.g. Runs When Time==2.5, or
//   Runs Once When <condition>
// Ramp (<variable>, <final value>, <duration>);
// SRamp(<variable>, <final value>, <duration>);
// Note that all variable values must be in base units (usually Metric)

/*  WaterOn as realvariable                    (0, fixed, description:"On/Off Switch");
  JetOn as realvariable                        (0, fixed, description:"On/Off Switch");
  SitOn as realvariable                        (0, fixed, description:"On/Off Switch");
*/

TankSuper.DisState: 0; //Initialization

// waiting for equilibrium
WAIT 48;  //hrs
            invoke compout;

TankSuper.DisState: 1; //completed tank prep
            invoke compout;

// Remove free liquid
TankSuper.Vdischargerate:  12.49186; //55 gpm = m3/hr

//Wait until supernate is less than salt level
WAIT for TankSuper.Supernatelevel < TankSalt.BulkLevel;

TankSuper.Vdischargerate: 0; //0 gpm = m3/hr
WAIT 24;  //hrs

// Remove interstial liquid - DRAIN step
TankSuper.Vdischargerate: 0.9084988; //4 gpm = m3/hr

//Wait until supernate is at 120" level
//WAIT for TankSuper.Supernatelevel <= 120;

// Wait 72 hrs per plan
wait 72;

TankSuper.Vdischargerate: 0; //0 gpm = m3/hr
WAIT 24;  //hrs

TankSuper.DisState: 2; //completed Drain
            invoke compout;
```

```
// Maintain 2 level above salt while transfering out at constant 20 gpm.  Adjust inflow to maintain level.

// Add initial fill

        Feed.Fixedrate: 6.813741; //30 gpm = m3/hr
        WAIT for TankSuper.tanklevelinches > TankSalt.BulkLevel + 3;  //Fill Batch to >2" above salt above 3" to initiate flow control
properly
        Feed.Fixedrate: 0; //0 gpm = m3/hr

// pause

        WAIT 24;  //hrs
        TankSuper.DisState: 3; //completed add
        invoke compout;



TankSuper.WaterOn: 1.0;

End
Task DissolutionCycle runs when TankSuper.WaterOn == 1.0// <Trigger>

// Start Salt Dissolution Batches

// Remove in batches of 30,000 gallons

        TankSuper.InitialLevel: TankSuper.TotalDischargeVol;  // Save initial discharge volume at start of addition

        TankSuper.Vdischargerate: 4.542494; //20 gpm = m3/hr

        WAIT for TankSuper.TotalDischargeVol - TankSuper.InitialLevel >= 113.5624;  // 113.5624 m3 = 30,000 gal batch

        TankSuper.Vdischargerate: 0; //0 gpm = m3/hr

// Delay between batches

        wait 24; // hrs
        TankSuper.DisState: 2; //completed remove
        invoke compout;
// If goal acheived, stop

            Restart when TankSalt.BulkLevel > 250;

            pause;

End

TYPE CompOut ROLE "SCRIPT" TEXT 'script:  OuputToXLS

dim i, j, num, comp, state
num = 1   'The sheet number

Set xlApp = CreateObject("Excel.Application")
Set xlBook1 = xlApp.Workbooks.Open("d:\user\my simulations\tank 37 salt dissolution\CompResult.xls")
Set xlSheet = xlBook1.Worksheets(num)

' Determine state

If TankSuper.DisState.value = 0 then
        state = "initialization"
elseif TankSuper.DisState.value = 1 then
        state = "Tank Prep Complete"
elseif TankSuper.DisState.value = 2 then
        state = "Removal/Drain Complete"
elseif TankSuper.DisState.value = 3 then
        state = "add/Sit Complete"
else
        state = "unknown"
end if

i = xlSheet.Cells(1,1).value
```

```
If i = 2 then

        j = 1

        xlSheet.Cells(i,j).value = "State"
        j = j + 1
        xlSheet.Cells(i,j).value = "Supernate Specific Gravity"
        j = j + 1
        xlSheet.Cells(i,j).value = "Cumulative Feed Volume (m3)"
        j = j + 1
        xlSheet.Cells(i,j).value = "Supernate Volume (m3)"
        j = j + 1
        xlSheet.Cells(i,j).value = "Total Supernate Discharge Volume (m3)"
        j = j + 1
        xlSheet.Cells(i,j).value = "Salt Volume (m3)"
        j = j + 1

        For Each Comp In TankSuper.apparent.components
                xlSheet.Cells(i,j).value = comp
                j = j + 1
        Next

        i = 3

end if

' start row of data
j = 1

' initial data

xlSheet.Cells(i,j).value = state
j = j + 1
xlSheet.Cells(i,j).value = TankSuper.SpG_Supernate.value
j = j + 1
xlSheet.Cells(i,j).value = Feed.FeedVol.value
j = j + 1
xlSheet.Cells(i,j).value = TankSuper.SupernateVolume.value
j = j + 1
xlSheet.Cells(i,j).value = TankSuper.TotalDischargeVol.value
j = j + 1
xlSheet.Cells(i,j).value = TankSalt.BulkVolume.value
j = j + 1

' Composition

For Each Comp In TankSuper.apparent.components
                xlSheet.Cells(i,j).value = TankSuper.SupernateCond(Comp).value
                j = j + 1
Next

' end row

i = i + 1
xlSheet.Cells(1,1).value = i

xlBook1.Save

xlApp.quit()

ENDTEXT
```

```
Properties
WorkingDirectory : "D:\User\My Simulations\Tank 37 Salt Dissolution\";
RunID : "evap";

Default as ComponentList;
Within Default
  Components
:["CSNO3","H2O","NA2CO3","NA2SIO3","NA2SO4","NA3PO4","NAALO2","NACL","NAF","NANO2","NANO3","NAOH"];
  Option("TRUE-COMP") : "NO";
EndWithin

All as ComponentList;
Within All
  Components :["ALO2-","CL-","CO3--","CS+","CSNO3","CSNO3(S)","F-
","H2O","NA+","NA2CO3","NA2SIO3","NA2SO4","NA3PO4","NAALO(S)","NAALO2","NACL","NACL(S)","NACO3(S)","NAF","NA
F(S)","NANO2","NANO2(S)","NANO3","NANO3(S)","NAOH","NAOH(S)","NAPO4(S)","NASO4(S)","NO2-","NO3-","OH-","PO4-
","SO4--"];
EndWithin

Salts as ComponentList;
Within Salts
  Components
:["CSNO3(S)","NAALO(S)","NACL(S)","NACO3(S)","NAF(S)","NANO2(S)","NANO3(S)","NAOH(S)","NAPO4(S)","NASO4(S)"];
EndWithin

Ions as ComponentList;
Within Ions
  Components :["ALO2-","CL-","CO3--","CS+","CSNO3","F-
","H2O","NA+","NA2CO3","NA2SIO3","NA2SO4","NA3PO4","NAALO2","NACL","NAF","NANO2","NANO3","NAOH","NO2-
","NO3-","OH-","PO4-","SO4--"];
EndWithin

Apparent as ComponentList;
Within Apparent
  Components
:["CSNO3","H2O","NA2CO3","NA2SIO3","NA2SO4","NA3PO4","NAALO2","NACL","NAF","NANO2","NANO3","NAOH"];
EndWithin
End
```

6 References

1. J. N. Brooke, K. Stahlie, and J. F. Peters, "Hydrological Methods can Separate Cesium from Nuclear Waste Saltcake", WSRC-TR-99-00358, July 1999.
2. L. B. Romanowski, "Low Curie Salt Processing Technical Plan", HLW-SDT-2002-00031, Rev. 0, March 18, 2002
3. D. L. Herting to N. W. Kirch, "Double-Shell Slurry Dissolution Kinetics", NHC Numatec Hanford Corporation, 82100-98-018, May 10, 1999.
4. D. T. Hobbs and C. J. Coleman, "Final Report: Analysis of Tank 41H Saltcake Sample #2 and Comparison to Sample #1 (U)", WSRC-TR-94-057, January 26, 1994.
5. High Level Waste Tank sample data base maintained by High Level Waste Engineering.
6. J. R. Hester, "High Level Waste Characterization System (WCS)", WSRC-TR-96-0264, Rev. 0, December 1996.
7. J. A. Pike, "Target Dilution Levels To Prevent Plugging During Tank 41 Saltcake Draining", HLW-PRE-2001-0014, Rev. 0, April 25, 2002.
8. "Statistical Methods of Uncertainty Analysis", National Institute of Standards and Technology, Information Technology Laboratory, http://www.itl.nist.gov/div898/carroll/u.htm.
9. D. G. Karraker, "Uranium Solubility Studies During Waste Evaporation (U)", WSRC-TR-93-433, Rev. 0, August 16, 1993.
10. D. L. Herting, "Saltcake Dissolution FY 2001 Status Report", HNF-8849, Rev 0., September 2001.
11. D. L. Herting, "Saltcake Dissolution FY 2000 Status Report", HNF-7031, Rev 0., September 2000.
12. D. L. Herting, et. el., "Saltcake Dissolution FY 1999 Status Report", HNF-5193, Rev 0., September 1999.
13. B. J. Wiersma, "An Investigation of Density Driven Salt Dissolution Techniques (U)", WSRC-TR-96-0160, August 1996.
14. D. L. Kiser, "Removal of Salt from High-Level Waste Tanks by Density-Driven Circulation or Mechanical Agitation", DP-1587, January 1981.
15. J. A. Pike, "Tank 37 Salt Dissolution Flowsheet Basis", WSRC-TR-2001-00599, Rev. 0, December 12, 2001.
16. J. C. Baily, Jr., to G. L. Albert, "Tank 10 Salt Removal Demonstration Detail", DPSP-79-17-11, April 11, 1979.
17. Several informal "Teletype" messages available in the HLWE File Room for Waste Removal about Tank 10 dissolution progress though most of this information is supported by the data in references 18, 19, and 20.
18. Waste Management Programs Report for April 1979, 79-21-4.
19. Waste Management Programs Report for May 1979, 79-21-5.
20. Waste Management Programs Report for June 1979, 79-21-6.
21. S. E. Aleman, "Calculation of the Mixture Density of Binary Aqueous and Simulated Waste Salt Solutions using Aspen Plus® ElecNRTL and Aspen OLI™ Property Methods", SRT-EMS-2002-00002, January 18, 2002.