DOE/EIA-M059(2002)
Part 2

EIA Model Documentation

PETROLEUM MARKET MODEL OF THE NATIONAL ENERGY MODELING SYSTEM

Part 2 – Appendices B thru I

March 2002

Oil and Gas Division
Office of Integrated Analysis and Forecasting
Energy Information Administration

APPENDIX B

Mathematical Description of Model

APPENDIX B. Mathematical Description of Model

Each refiner is trying to minimize the cost of meeting demands. Therefore, the market moves toward lower-cost refiners who have access to crude oil and markets. A key premise is that the selection of crude oils, refinery process utilization, and logistics will adjust to minimize the overall cost of supplying the market with petroleum products.

In order to generate refined product prices, the PMM contains a static linear program model of the U.S. petroleum refining and marketing system that meets demand for refined products while minimizing costs. The PMM, like the other NEMS models, is written in FORTRAN. The software includes the Optimization Modeling Library or OML, a set of FORTRAN callable subroutines. The LP portion of the PMM is a complete problem matrix prepared prior to NEMS processing. The LP remains in fast memory throughout the NEMS run, thereby avoiding many disk I/O operations.

It is necessary to view the PMM in the context of the NEMS program to understand its function. For each cycle, the main NEMS model calls the demand models to calculate energy demands. Each supply model is then called to calculate energy prices. When the prices and demands converge to within the specified tolerance, the NEMS iteration is complete and the next yearly NEMS cycle begins. If the computed prices have not converged, new demand quantities are computed, passed to the supply models, and the cycle is repeated. In the case of the PMM, a supply model, the refined product prices are obtained as the marginal prices from an optimal solution to the PMM LP. These product prices are sent to the NEMS demand models. The LP matrix is updated with the new demands for refined products and the cycle continues until convergence is reached. The demand level modifications to the PMM LP and the re-optimization of the LP matrix, which remains in core memory, are accomplished by executing FORTRAN callable subroutines.

For *AEO2002* the original generation of the PMM matrix is performed using OML[1] and FORTRAN. OML (Optimization Modeling Library) is a library of FORTRAN callable subroutines for data table manipulation, matrix generation, and solution retrieval programs for report writing. The matrix is solved with the optimizer, C-WHIZ.[2]

B.1 Mathematical Formulation

The table of column activity definitions and row constraints defined in the PMM matrix incorporate certain premises which are described in Appendix A. The general structure of the matrix is shown in Table B1.

[1]Ketron Management Science, Inc., *Optimization Modeling Library, OML User Manual*, (November 1994).

[2]Ketron Management Science, Inc., *C-WHIZ Linear Programming Optimizer, User Manual*, (July 1994).

Table B1. PMM Linear Program Structure

PMM Linear Program Overview

	Crude Trans.	Purchases Crude Oil, Other Inputs	Crude Distillation	Other Process Unit Operations	Capacity Expansion	Stream Transfers	Blending	Product Sales	Product Trans.	Row Type	RHS
Objective	-ct	-c	-o	-o	-i			+p	-pt	NC	Max
Crude Oil Balance	+1 +1 -1	+1 +1 +1	-1 -1							GE	0
Intermediate Stream Balance			+y +y	-1 -1 +y +y		-1 +1 -1 +1	-1 -1			GE	0
Utilities		+1	-u	-u +1						GE	0
Policy Constraints				+z -z				+z -z		GE LE	0
Environmental Constraints			+q	+q						GE LE	E
Unit Capacities			+1	+1	-1					LE	K
Quality Specifications							+q +q -Q			GE LE	0
Product Sales							-1	-1	-1 +1 -1	GE	0
Pipeline/Marine Capacities	+1 +1								+1	LE	C
Bounds	Up/Lo/Fix	Up/Lo/Fix						Up/Lo/Fix			

Legend: c = crude cost; y = yield; u = utility consumption; K = unit capacity; o = operating cost; p = price; z = policy ratio; q = stream quality; ct = crude transportation cost; pt = product transportation cost; Q = product specifications; C = pipeline/marine capacity; E = environmental quality limit; i = investment cost

B.2 Matrix Indices

Several indices are used in the column and row definitions presented below. The definition of these indices are shown in Table B2.

Table B2. Index Definitions

Index Symbol	Description
(a)	Alcohol
(b)	Domestic crude oil production region
(c)	Crude oil type
(d)	Marketing region (demand)
(e)	Emissions source
(g)	Dry gas
(h)	Product recipe
(i)	Intermediate stream
(l)	Utility type
(m)	Processing mode or shipping mode
(p)	Refined product
(pc)	Shipping class
(q)	Pricing level
(r)	Refinery region
(s)	Refinery stream
(u)	Processing unit type
(v)	Crude oil source (foreign(F), domestic(D), Alaska(A))
(y)	Product specification quality

B.3 Column Definitions

Table B3. Column Definitions

Column Notation	Description
$B_{a,d,q}$	Volume of alcohol (a) purchased in demand region (d) at price level (q). At present, this column exists only for ethanol.
$D_{p,d}$	Sales volume of product (p) in demand region (d); export volume of product (p) in demand region (d).
$E_{u,r}$	Stream day capacity added during this simulated period for processing unit type (u) in refinery region (r).
$G_{i,p,r}$	Volume of intermediate stream (i) blended into spec blended product (p) at refinery region (r).
$H_{p,h,r}$	Volume of product (p) manufactured in refinery region (r) using recipe (h).
$H_{p,h,d}$	Volume of product (p) made by recipe (h) in demand region (d) by splash blending.
$H_{p,i,r}$	Volume of product (p) generated from gas plant stream (i) in refinery region (r).
$H_{g,i,r}$	Volume of dry gas (g) generated from gas plant stream (i) in refinery region (r).
$HKWHMCH_r$	Electricity usage at merchant plant in refinery region (r).
$HMPRF_{i,r}$	Volume of GTL stream i transferred from Alaska (merchant plant) to refinery region (r).
$Iz_{p,d}$	Volume of distress import of product (p) into demand region (d) where d = 2,3,7,8,9.
$I_{p,r,q}$	Volume of refined product (p) imported into refinery region (r) at price level (q).
$K_{u,r}$	Base processing capacity in processing unit (u) at refinery region (r) in Mbbl/cd. This column is upper bounded.
$L_{u,r}$	Cumulative stream day capacity added for processing unit (u) at refinery region (r) during the previous simulated periods. This column is fixed.
M_r	Volume of methanol consumed by the chemical industry in refinery region (r).

Column Notation	Description
Mt	Total volume of U.S. methanol consumption by the chemical industry, an input.
$NGA_{r,q}$	Alaska natural gas supply curve (steps (q)) used to generate GTL's for transport to refinery region (r).
$NZAMHN_q$	Volume at price discount q of Alaskan Crude (AMH) exports.
$NZAMHP_q$	Volume at price increment q of Alaskan Crude (AMH) exports.
N_rNGRFN_q	Volume at price discount (q) of natural gas in refinery region (r).
N_rNGRFP_q	Volume at price increment (q) of natural gas in refinery region (r).
$O_{c,v,b}$	Export volume of crude oil (c) with source code (v) produced in domestic region (b). At present, only Alaska exports are allowed.
P_b	Volume of domestic crude oil produced at producing region (b).
$Pi_{c,r,q}$	Volume of imported crude oil type (c) imported by refinery region (r) at price level (q).
PNGL	Alaska production of NGL's.
$Q_{p,r}$	Volume of spec product (p) manufactured in refinery region (r).
$Ra_{c,v,r}$	Crude oil volume distilled in refinery region (r) from source (v) of crude oil type (c).
Rc_r	Manufacturing activity level for cogeneration in refinery region (r).
$R_{u,r,m}$	Manufacturing activity level in mode (m) operation in processing unit (u) at refinery region (r).
Tu	Total volume of imported unfinished oil over all refinery regions.
$T_{i,r}$	Volume of unfinished oil component (i) processed in refinery region (r).
TAAMHXZ	Volume of AMH crude transported from Alaska to Valdez.
$TATOT_c$	Volume of crude type (c) produced in Alaska.
TAGTL	Volume of GTL transported from Alaska to Valdez.
TANSO	Volume of North Slope oil transported from Alaska to Valdez.

Column Notation	Description
$TCBN_r$	Total tax levied on total carbon emissions resulting from refinery operations in refinery region (r).
$Tx_{s,p,r}$	Volume of stream (s) transferred into product (p) in refinery region (r).
$Tx_{p,p',r}$	Volume of product (p) transferred into product (p') storage in refinery region (r).
$Tx_{s,s',r}$	Volume of stream (s) transferred to stream (s') at refinery region (r).
$Tx_{FR,TO,i,r}$	Volume of stream (i) transferred from plant (FR) to plant (TO) in refinery region (r).
$U_{l,r}$	Quantity of utility (l) that is purchased in refinery region (r): (l) = KWH, STM, and NGF (power, steam, and natural gas fuel) with units in thousands of kWh, Mlbs., and MMcf respectively.
Vcj	Total dead weight ton capacity of Jones Act crude oil tankers. This column is constrained to some maximum.
V_{pc}	Total dead weight tons of Jones Act product tanker of shipping class (pc). This column is constrained to some maximum.
Vcts	Total crude oil transhipped from PAD District III to PAD District II. This column is constrained to some maximum. (Not used in 3-region pmm.)
$Vcp_{b,r}$	Total crude oil shipped by pipeline from domestic producing region (b) to refinery region (r). This column is constrained to some maximum.
$Vpp_{r,d,m}$	Total volume of light products (p) shipped via pipeline mode (m) from refining region (r) to demand region (d). This column is constrained to some maximum.
$Vtpl_{r,d,m}$	Total volume of LPG products (p) shipped via pipeline mode (m) from refining region (r) to demand region (d). This column is constrained to some maximum.
$W_{a,d,r,m}$,$W_{a,d,d',m}$	Volume of alcohol (a) shipped from demand region (d) to refinery region (r) <or demand region d'> via transfer mode (m).
$W_{a,r,d,m}$	Volume of alcohol (a) shipped from refinery region (r) to demand region (d) via transfer mode (m).
$W_{p,r,d,m}$	Volume of product (p) shipped from refinery region (r) to demand region (d) via transfer mode (m).

Column Notation	Description
$W_{p,d,d',m}$	Volume of product (p) transhipped from demand region (d) to demand region (d') via transfer mode (m).
$WGTL_{m,r}$	Volume of GTL shipped via transfer mode (m) to refinery region (r).
$X_{p,d}$	Volume of product (p) exported from demand region (d).
$Xz_{p,d}$	Distress volume of product (p) exported from demand region (d).
$Y_{c,v,b,r,m}$	Volume of crude oil type (c) with source code (v) shipped from domestic region (b) to refinery region (r) via transfer mode (m).
Zt	Total volume of crude oil processed over all refinery regions.
ZET_d	Total volume of ethanol (from corn and cellulose) supplied from demand region (d).
Z_rFLO_u	Sum of the base, build, and expanded capacity in processing unit (u) at refinery region (r).
ZOX_r	Total quantity of percent oxygen-barrels in reformulated gasoline produced in refinery region (r).
ZZAMHTOT	Export volume of Alaskan Crude Oil.

B.4 Objective Function

The objective function has been established based on the premise that costs associated with product imports, non-crude oil inputs, and crude oil supplies are based on a given world oil price. With this in mind, the following objective function has been defined for PMM.

Given:

$PR_{p,d}$	is the unit price of product (p) sold in demand region (d). It is the price associated with the demand volume for that product in the specified demand region, i.e. it is associated with $D_{p,d}$. Similarly, each of the other 'PR' coefficients represents the unit price of the activity it is associated with.
PRAMH	is the target price for Alaskan crude exports.
$PRAMHP_q$	is the premium added to the Alaskan crude target price.
PR_rNGRFN_q	is the discount from target natural gas fuel price in refinery region (r).
TE_d	is the sum of Federal and State tax credits for use of ethanol in gasoline.
$C_{u,r,m}$	is the variable cost per unit of column $R_{u,r,m}$, i.e. the cost of one unit of manufacturing activity in mode (m) operation in processing unit (u) at refinery region (r). Similarly, each of the other 'C' coefficients represents the unit cost of the activity it is associated with in the objective function as stated below.
CAMHXZ	is the cost of transferring Alaskan exports to Valdez.
$CAMHN_q$	is the discount from Alaskan crude target price.
C_rNGRFP_q	is the premium added to the target natural gas fuel price in refinery region (r).

Maximize the difference between the following sum of product revenues and costs. Thus the objective function is represented as the maximization of a quantity defined by the following revenue terms:

$$\Sigma_d\Sigma_p\, D_{d,p}*PR_{d,p} + \Sigma_p\Sigma_h\Sigma_r\, H_{p,h,r}*PR_{p,h,r} + \Sigma_g\Sigma_i\Sigma_r\, H_{g,i,r}*PR_{g,i,r} + \Sigma_p\Sigma_h\Sigma_d H_{p,h,d}*PR_{p,h,d}$$

$$+ \Sigma_p\Sigma_d\, X_{p,d}*PR_{p,d} + \Sigma_p\Sigma_d\, Xz_{p,d}*PR_{p,d} + \Sigma_d\, ZET_d * TE_d + ZZAMHTOT * PRAMH$$

$$+ \Sigma_q\, NZAMHP_q * PRAMHP_q + \Sigma_r\, \Sigma_q\, N_rNGRFN_q * PR_rNGRFN_q + \Sigma_r\, Rc_r *PR_r$$

minus the following cost terms:

$$\Sigma_a\Sigma_d\Sigma_q\ B_{a,d,q}*C_{a,d,q} + \Sigma_p\Sigma_r\Sigma_q\ I_{p,r,q}*C_{p,r,q} + \Sigma_p\Sigma_d\ Iz_{p,d}*C_{p,d} + \Sigma_b\ P_b*C_b + \Sigma_c\Sigma_r\Sigma_q\ Pi_{c,r,q}*C_{c,r,q}$$

$$+ \Sigma_i\Sigma_r\ T_{i,r}*C_{i,r} + \Sigma_l\Sigma_r\ U_{l,r}*C_{l,r} + \Sigma_u\Sigma_r\Sigma_m\ R_{u,r,m}*C_{u,r,m} + \Sigma_u\Sigma_r\ E_{u,r}*C_{u,r} + \Sigma_u\Sigma_r\ L_{u,r}*C_{u,r} +$$

$$+ \Sigma_s\ \Sigma_i\ \Sigma_r\ \Sigma_l\ W_{a,d,r,m}*C_{a,d,r,m} + \Sigma_p\Sigma_r\Sigma_d\Sigma_m\ W_{p,r,d,m}*C_{p,r,d,m} + \Sigma_p\Sigma_d\Sigma_{d'}\Sigma_m\ W_{p,d,d',m}*C_{p,d,d',m}$$

$$+ \Sigma\ _c\Sigma_v\Sigma_b\Sigma_r\Sigma_m\ Y_{c,v,b,r,m}*C_{c,v,b,r,m} + \Sigma_r\ TCBN_r + \Sigma_r\Sigma_q\ N_rNGRFP_q\ *\ C_rNGRFP_q$$

$$+ \Sigma_q\ NZAMHN_q\ *\ CAMHN_q + TAAMHXZ\ *\ CAMHXZ$$

$$+ \Sigma_r\ M_r\ *C_r + \Sigma_p\Sigma_i\Sigma_r\ H_{p,i,r}\ *\ C_{p,i,r} + \Sigma_i\Sigma_r\ Tx_{FR,TO,i,,r}\ *\ C_{FR,TO,i,r} + \Sigma_r\ HKWHMCH_r\ *\ C_r$$

$$+ \Sigma_r\ \Sigma_q\ NGA_{r,q}\ *C_{r,q} + PNGL\ *C + TAGTL\ *C + TANSO\ *C + \Sigma_m\Sigma_r\ WGTL_{m,r}\ *\ C_{m,r}$$

Note: (1) $\Sigma_u\Sigma_r\Sigma_m\ R_{u,r,m}*C_{u,r,m}$ term is represented in the matrix as T(r)OVCOBJ + T(r)GPLOVC + T(r)MCHOVC as represented by rows P9r)OVC, G(r)OVC, H(r)OVC, respectively.
(2) ZZZAMHTOT=TAAMHXZ as defined by row CZAMH.
(3) Row P(r)CBNTAX is used to set $TCBN_r$ regional totals for tax on carbon emissions.

B.5 Row Constraints

1. The implicit world oil price, WOP (the refiner's acquisition cost of imported crude oil), must be at least some fraction of premised WOP:

$$\Sigma_c\Sigma_r\Sigma_q \, Pi_{c,r,q} * C_{c,r,q} - WOP * Nwop * Zt \geq 0$$

where:

$Pi_{c,r,q}$ the volume of imported crude oil acquired by refinery region (r) of crude type (c) at price level (q). The volume of each crude type (c) is upper bounded by each step of the supply curve.

$C_{c,r,q}$ is crude oil cost applicable to $Pi_{c,r,q}$. $/bbl

WOP is the premised World Oil Price. $/bbl

Nwop is minimum fraction of the WOP by which refiners must acquire crude oils by volume weighted average; of course Nwop $\leq$ 1.

Zt is total processed foreign crude oil over all refinery regions.

I.e., the average refiner acquisition cost of crude oil will be at least some input fraction of WOP.

Note: The index (v) denoting crude oil source is always "F" for imported crude. Therefore, no summation on (v) occurs for $Pi_{c,r,q}$.

2. The implicit world oil price, WOP (the refiner's acquisition cost of imported crude oil), must not be greater than some fraction of premised WOP:

$$\Sigma_c\Sigma_r\Sigma_q \, Pi_{c,r,q} * C_{c,r,q} - WOP * Xwop * Zt \leq 0$$

where:

Xwop is the maximum fraction of the WOP by which refiners must acquire crude oils by volume weighted average, of course Xwop $\geq$ 1.

I.e., the average refinery acquisition cost of crude oil must not exceed some input fraction of WOP.

Note: This constraint in conjunction with the previous constraint confines the imported crude oil volumes so that their composite unit cost is close to the WOP. Since the costs of the five types of imported crude oil bracket the WOP, there is a continuum of import possibilities. Of course the crude oils vary in cost according to crude quality and transportation cost. However, to avoid infeasibilities due to an overly restrictive constraint, it is presently the practice to

allow a variation of 50 cents per barrel. This is also made necessary when maximum import restrictions are placed on all but a "swing" crude oil. Clearly, if the tolerance is sufficiently relaxed, a single crude type could be imported rather than a reasonably balanced mix.

3. Calculate total foreign crude oil processed by summing all crude oil volumes processed over all refinery regions:

$$\Sigma_c \Sigma_v \Sigma_r \, Ra_{c,F,r} - Zt = 0$$

where:

$Ra_{c,F,r}$ is the crude oil volume distilled in refinery region (r) from foreign source (F) of crude oil type (c).

I.e. the total U.S. refined foreign crude oil volume equals the sum of all foreign crude oils refined over all regions.

4. Place an upper bound on product import volume:

$$\Sigma_p \Sigma_r \Sigma_q \, I_{p,r,q} \leq IPmax$$

where:

$I_{p,r,q}$ is the volume of product (p) imported into refinery region (r) at cost level (q).

Ipmax is the maximum assumed volume of imported products allowed into the United States, an input value.

I.e., the sum of product volumes imported at all price levels over all refinery regions must not exceed some maximum.

5. Assure that the volumes of methanol consumed in each refinery region sum to the assumed total volume.

$$\Sigma_r \, M_r - Mt = 0$$

where:

M_r is methanol consumed by the chemical industry in refinery region (r).

Mt is total U.S. methanol consumption by the chemical industry, an input.

I.e., methanol consumed by the chemical industry in each refinery region must sum to the projected chemical industry total.

Note: The PMM models domestic methanol production aggregated to the refinery region level as though the plants were processing units integral to the refinery.

The methanol production is allocated to two demands: chemical industry demand and transportation sector demand.

6. Limit capital investment for refinery expansion in each refinery region:

$$\Sigma_u E_{r,u} * A_{r,u} \leq Emax_r \quad \forall \; r$$

where:

$E_{r,u}$ is a capacity addition for this operating year for processing unit type (u) in refinery region (r).

$A_{r,u}$ is the capital investment required per unit of capacity for processing unit type (u) in refinery region (r), million dollars per Mbbl/d.

$Emax_r$ is the upper limit on capital expenditures in refinery region (r), an input. $MM

I.e., total capital expenditures are limited for each refinery region. This limit allows the user to place limits on expansion for a given scenario. The default values are large such that they are not constraining, except in PADD I (refinery region E) where a limit of $500 million dollars is the default. This value is based on analyst judgement to reflect the low expectation of refinery expansion on the East Coast. (Region B limit is $29,000 million, region W limit is $19,000 million.)

Note: For NEMS production runs, this constraint has not been operative except for PAD District I. The assumption has been that environmental waivers and permits will preclude capacity additions in this region.

7. Limit total U.S. refinery capital investment:

$$\Sigma_r \Sigma_u E_{r,u} * A_{r,u} \leq Emax$$

where:

Emax is the upper limit on capital investment over all refinery regions, an input. $MM

I.e., total capital expenditures are constrained to some maximum. This limit allows the user to place limits on expansion for all regions in the United States. The default value is $50 billion and is not constraining.

Note: This constraint has been used to determine maximum product import needs by setting Emax to zero. It has not been operative in any production runs thus far. However, like the previous row, it is a potential policy exploration handle. For instance, one could hypothesize that in a capital-short domestic environment, total industry investment is limited.

8. Limit the volume of unfinished oil processed in U.S. refineries:

$$\Sigma_c\Sigma_v\Sigma_r \, Ra_{c,v,r} - B1 * Tu \geq B0$$

where:

$Ra_{c,v,r}$ is the crude oil volume distilled in refinery region (r) from source (v) of crude oil type (c).

Tu is total processed unfinished oil over all refinery regions.

B0,B1 are regression equation coefficients (see Appendix A). B0 = 11,674.3, B1 = 4.087.

I.e., the maximum allowable unfinished oil processed is a linear function of total crude oil processed.

9. Limit the volume of each unfinished oil component processed in each refinery region:

$$A_{i,r} * Tu - T_{i,r} \geq 0 \quad \forall \; r, i$$

where:

$T_{i,r}$ is the volume of unfinished oil component (i) processed in refinery region (r).

$A_{i,r}$ is the fraction of component (i) of total unfinished oil that is processed in refinery region (r), an input ($\Sigma_i\Sigma_r A_{i,r} = 1.0$).

I.e., the volume distribution of each type of unfinished oil processed at each refinery region is constrained to the historical pattern.

Note: No unfinished oil processing in PADD 5 (refinery region W).

10. Balance by volume, at the demand regions, each alcohol purchased by the petroleum industry and domestic methanol shipped in from the refinery regions against alcohol blended by recipe in the demand region and alcohol shipped out of the demand region:

$$\Sigma_q B_{a,d,q} + \Sigma_r\Sigma_m W_{a,r,d,m} + \Sigma_{d'}\Sigma_m W_{a,d',d,m} - \Sigma_p H_{p,a,d} * A_{p,a} - \Sigma_r\Sigma_m W_{a,d,r,m} - \Sigma_{d'}\Sigma_m W_{a,d,d',m} = 0 \quad \forall \; a, d$$

where:

$a \in h$ so that a is a subset of all recipe blends (h) and in fact, a = ETH and MET.

$B_{a,d,q}$ is the volume of alcohol (a) purchased in demand region (d) at price level (q). At present, this column exists only for ethanol.

$W_{a,r,d,m}$ is the volume of alcohol (a) received in demand region (d) from refinery region (r) via shipping mode (m). This column exists only for methanol.

$W_{a,d',d,m}$ is the volume of alcohol (a) received in demand region (d) from demand region (d') via shipping mode (m). This column exists only for ethanol.

$H_{p,a,d}$ is the volume of recipe product (p) manufactured by alcohol blend recipe (a) consumed in demand region (d).

$A_{p,a}$ is the volume fraction of recipe product (p) represented by alcohol (a), an input.

$W_{a,d,r,m}$ is the volume of alcohol (a) shipped from demand region (d) to refinery region (r) via mode (m). This column is valid only for ethanol.

$W_{a,d,d',m}$ is the volume of alcohol (a) shipped from demand region (d) to demand region (d') via mode (m). This column is valid only for ethanol.

I.e., in each demand region, all alcohol that is purchased or produced and shipped in from the refinery regions must be either splash blended into or shipped to a refining center.

Note: It is assumed that all ethanol purchases are made in the demand regions. Thus, ethanol needed by the refinery for processing or blending must be shipped from the demand regions.

11. For those products which are not blended by recipe at the demand regions, for each demand region, and for each product: imports plus what is received via domestic shipment must equal the volume blended into recipes plus regular sales volume (local and exports):

$$I_{p,d} + \Sigma_r \Sigma_m W_{p,r,d,m} + \Sigma_h H_{p,h,d} - \Sigma_{p'} \Sigma_h H_{p',h,d} * A_{p',h} - X_{p,d} - D_{p,d} = 0 \qquad \forall \; d, p$$

where:

$I_{p,d}$ is volume of product (p) imported into demand region (d).

$W_{p,r,d,m}$ is product (p) received in demand region (d) from refinery region (r) via transfer mode (m).

$X_{p,d}$ is volume of product (p) exported from demand region (d).

$H_{p,h,d}$ is volume of product (p) manufactured by recipe (h) at demand region (d).

$H_{p',h,d}$ is volume of product (p') manufactured by recipe (h) at demand region (d). This column exists only when product (p) is consumed to produce product (p') by a recipe blended at the demand region, i.e. splash blended at the terminals.

$A_{p',h}$ is the volume fraction of product (p') represented by product (p), consumed by recipe (h), an input.

$D_{p,d}$ is a sales volume (local and export demand) of product (p) in demand region (d), this activity is fixed at the last NEMS iterative demand value; i.e., it does not influence the cost decision in the solution.

I.e., in each demand region, for each product, a balance is made whereby the volume imported and the volume shipped in from the refinery regions must equal the volume splash blended at the terminal plus straight sales volume.

12. Balance, at each demand region, the volumes for each of the recipe products M85 and E85 - the products which are blended at the demand terminals - so that manufactured volume plus imports equals the recipe sales volume for these terminal splash blended recipes:

$$\Sigma_h H_{p',h,d} + I_{p',d} - D_{p',d} = 0 \quad \forall \; p', d$$

where:

$p' \in p$ so that p' is a subset of all products and in fact, p' = E85 and M85.

$H_{p',h,d}$ is volume of recipe product (p') made by recipe (h), produced at demand region (d) by splash blending.

$I_{p',d}$ is volume of product (p') imported into demand region (d).

$D_{p',d}$ is the sales volume of product (p') in demand region (d).

I.e., for each demand region, all M85 and E85 blended at the terminals plus M85 and E85 imported into the demand region must equal sales of the corresponding products.

13. For Census Divisions (demand regions) 5 and 6, balance the volume of each product that is shipped into the demand region with volume shipped out of the demand region:

$$\Sigma_r \Sigma_m W_{p,r,d',m} + \Sigma_d \Sigma_m W_{p,d,d',m} - \Sigma_d \Sigma_m W_{p,d',d,m} = 0 \quad \forall \; d', p$$

where:

$d' \in d$ and in fact, d' = Census Divisions 5 (South Atlantic) and 6 (South Central).

$W_{p,r,d',m}$ is volume of product (p) received by demand region (d') from refinery region (r) via pipeline (P/L) mode (m). These columns are generated only for the pipeline shipping mode when (d') is CD 6.

$W_{p,d,d',m}$ is volume of product (p) received in demand region (d') via transhipment from demand region (d) by mode (m), e.g. from 6 to 5 or from 5 to 2. For these columns, d' = d is not allowed.

$W_{p,d',d,m}$ is volume of product (p) transhipped from demand region (d') to demand region (d) by mode (m). However, for these columns, d' = d is allowed; in this case the activity represents P/L sales

I.e., pipeline product received in CD 5 or CD 6 must balance P/L product sales and transhipments.

Note: CD 5 and CD 2 (Mid Atlantic) receive much of their product volumes via pipeline (P/L) originating in PAD District III. Thus PAD District II production which is transported via P/L is split into one activity representing P/L product which is sold in CD 6 and another activity which is product transhipped through CD 6 into CD 5. Similarly, this latter component is split into two column activities, one which represents P/L product sales in CD 5, and another which represents product transhipped through CD 5 to CD 2.

14. Balance the domestic production of each crude type at each producing region (b) against shipments to domestic refineries (r) and exports:

Lower 48 regions (all crude types) and Alaska crude type ALL :

$$P_b * A_{c,v,b} - \Sigma_r \Sigma_m Y_{c,v,b,r,m} - O_{c,v,b} = 0 \quad \forall \; b, c, v \qquad \text{(currently, } O_{c,v,b} = 0\text{)}$$

Alaska crude type AMH only (takes into account gain due to mixing of GTL's during transport from N. Slope, and exports to both Japan and Canada) :

$$P_b * A_{c,v,b} + GTLLOS * TAGTLTOT - \Sigma_r \Sigma_m Y_{c,v,b,r,m} - O_{c,v,b} - TAAMHXZ = 0 \quad \forall \; b, c, v$$

where:

P_b is total volume of domestic crude oil produced at producing region (b). NOTE: for Alaska, $P_b * A_{c,v,b} = TATOT_c$

$A_{c,v,b}$ is the fraction by volume of P_b that is crude type (c) with source code (v), an input.

$Y_{c,v,b,r,m}$ is volume of crude oil type (c) with source code (v) produced in domestic region (b) that is shipped to refinery region (r) via mode (m).

$O_{c,v,b}$ is the export volume of crude oil (c) with source code (v) produced in domestic region (b). At present, only Alaska crude exports are allowed (but set to 0 in code).

TAGTLTOT is total volume of GTL transported in Alaska to Valdez.

GTLLOS percent of GTL lost due to mixing with AK oil during transport in Alaska to Valdez.

TAAMHXZ is total volume of AMH crude transferred from Alaska to Valdez. Only used with Alaska production region (A) row constraint.
(Note: TAAMHXZ = ZZAMHTOT as defined by row constraint CZAMH.)

NOTE: for Alaska, $P_b * A_{c,v,b} = TATOT_c$

I.e, for each production region and crude oil type, domestic production must be balanced against exports and shipments to refinery regions.

15. Limit shipments of crude oil on Jones Act marine tankers:

$\Sigma_c \Sigma_v \Sigma_b \Sigma_r \Sigma_{m'} Y_{c,v,b,r,m'} * A_c - Vcj = 0$

where:

$m' \in m$ m' is the set of shipping modes that correspond to Jones Act crude oil tankers.

$Y_{c,v,b,r,m'}$ is volume of crude oil type (c) with source code (v) produced in domestic region (b) that is shipped to refinery region (r) via mode (m').

A_c is dead weight tons per barrel of crude oil of type (c), about 0.1344.

Vcj is total dead weight tons of Jones Act crude oil. This column is constrained to some maximum.

I.e., total Jones act crude oil shipments are limited by the existing fleet.

Note: This constraint is stated in the form 'X - MAX = 0' with bounds on MAX rather than in the form '$X \leq MAX$' (which is more straight forward from a mathematical standpoint) because this allows the analyst to add a minimum constraint or change over to a fixed constraint within the fortran code without a regeneration of the MPS file. Thus it is an artifice of convenience.

16. Limit shipments of refined product (and GTL) on Jones Act marine tankers by product class (clean, dirty...):

$\Sigma_{p'} \Sigma_r \Sigma_d \Sigma_{m'} W_{p',r,d,m'} * A_{p'} + \Sigma_{p'} \Sigma_r \Sigma_d \Sigma_{m'} W_{GTL,A,r,m'} * A_{GTL} - V_{pc} = 0 \quad \forall$ product class (pc)

where:

$m' \in m$ m' is the set of shipping modes that correspond to Jones Act product tankers carrying product class (pc).

$p' \in p$ p' is the set of products which correspond to the product class (pc) of the particular constraint row.

$W_{p',r,d,m'}$ is volume of product (p') shipped from refinery region (r) to demand region (d) via mode (m').

$A_{p'}$ is dead weight tons per barrel of product (p'), an input.

$W_{GTL,A,r,m'}$ is volume of product GTL shipped from Alaska to refinery region (r) via mode (m').

A_{GTL} is dead weight tons per barrel of GTL, an input.

V_{pc} is total dead weight tons of Jones Act product of a shipping class (pc). This column is constrained to some maximum.

I.e., Jones Act product shipment volume is limited by tanker availability.

17. Allow and limit transhipments of crude oil from the Gulf Coast to the PAD District II refining region

(Note: **Not used** in latest 3-region version of PMM.):

$$\Sigma_v \Sigma_c \, Y_{c,v,G,C,m} - Vcts = 0$$

where:

$Y_{c,v,G,C,m}$ is volume of crude oil type (c) with source code (v) that is transhipped from the PAD District III (code=G) refinery region to the PAD District II (code=C) refinery region.

Vcts is total crude oil transhipped from PAD District III to PAD District II. This column is constrained to some maximum

I.e., the volume of crude oil shipped from the Gulf Coast to PAD District II refineries is limited.

18. For each domestic crude oil producing region and refinery region (r) allowable combination, allow and limit pipeline shipments of crude oil:

$$\Sigma_c \Sigma_v \Sigma_m \, Y_{c,v,b,r,m} - Vcp_{b,r} = 0 \quad \forall \; b, r$$

where:

$Y_{c,v,b,r,m}$ is volume of crude oil type (c) that is shipped from domestic producing region (b) to refinery region (r) via pipeline mode (m).

$Vcp_{b,r}$ is total crude oil shipped by pipeline from domestic producing region (b) to refinery region (r). This column is constrained to some maximum.

I.e., limit the crude oil volume shipped by pipeline from each applicable producing region//refinery region pair.

19. For each refinery/demand region applicable combination, limit pipeline shipments of light products to available capacity:

$$\Sigma_{p'}\Sigma_r\Sigma_d\Sigma_m\ W_{p',r,d,m} - Vpp_{r,d,m} = 0 \quad \forall \ r, d, m$$

where:

$p' \in p$ p' is the set of light products which can be shipped from refinery region (r) to demand region (d) via pipeline mode (m).

$W_{p',r,d,m}$ is volume of product (p') shipped from refinery region (r) to demand region (d) via pipeline mode (m).

$Vpp_{r,d,m}$ is total volume of light products (p') shipped via pipeline mode (m) from refining region (r) to demand region (d). This column is constrained to some maximum.

I.e., the volume of light products that can be shipped by pipeline from each refinery region to each demand region is limited by the available pipeline capacity.

Note: Special cases exist for transfer from CD6 to CD5, from CD5 to CD2, from CD6 to CD6 (P/L sales), and from CD5 to CD5 (P/L sales).

20. For each refinery/demand region applicable combination, limit pipeline shipments of liquid petroleum gas and PCF volumes to available LPG pipeline capacity:

$$\Sigma_{p'}\Sigma_r\Sigma_d\Sigma_m\ W_{p',r,d,m} - Vtpl_{r,d,m} = 0 \quad \forall \ r, d, m$$

where:

$p' \in p$ p' is the set of LPG and PCF products which can be shipped from refinery region (r) to demand region (d) via pipeline mode (m).

$W_{p',r,d,m}$ is volume of LPG and PCF product (p') shipped from refinery region (r) to demand region (d) via pipeline mode (m).

$Vtpl_{r,d,m}$ is total volume of LPG and PCF products (p') shipped via pipeline mode (m) from refining region (r) to demand region (d). This column is constrained to some maximum.

I.e., the volume of LPG products that can be shipped by pipeline from each refinery region to each demand region is limited by the available pipeline capacity.

Note: Special case exists for transfer from CD6 to CD5.

21. Refinery 'policy' table entries are adhered to:

$$\Sigma_u \Sigma_m R_{u,r,m} * A_{e,u,r,m} - A_{e,r} * Z_r FLO_u \quad \leq, \geq, = \; 0 \qquad \forall \quad e, r$$

where:

$R_{u,r,m}$ is the manufacturing activity level in mode (m) operation in processing unit (u) at refinery region (r).

$A_{e,u,r,m}$ is the coefficient entered into the processing unit (u) table for refinery region (r) in the operating mode (m) column at policy row (e), an input.

$A_{e,r}$ is the fractional amount of total capacity value entered in the policy table in row (e) for refinery region (r), an input, i.e., constraints may be introduced by the analyst via the "policy" rows.

Note: The type of row ($\leq, \geq, =$) is determined by the entry in column heading TYPE of the policy table (r)POL where $A_{e,r}$, appears. It may also be a non-constraining row, in which case the row is free. The total processing unit throughput is the base for the policy limits in each refinery region:

$$Z_r FLO_u - K_{u,r} - A_{u,r} * (L_{u,r} + E_{u,r}) \quad = \quad 0 \qquad \forall \quad r, u$$

where:

$Z_r FLO_u$ is the sum of the base, build, and expanded capacity in processing unit (u) at refinery region (r).

$K_{u,r}$ is the base processing capacity in processing unit (u) at refinery region (r) in Mbbl/cd. This column is upper bounded rather than fixed. See note below.

$A_{u,r}$ is the stream factor for processing unit (u) at refinery region (r) defined as the ratio of calendar day capacity to stream day capacity.

$L_{u,r}$ is the cumulative stream day capacity added for processing unit (u) at refinery region (r) during the previous simulated periods. This column is, of course, fixed.

$E_{u,r}$ is the stream day capacity added during this simulated period for processing unit (u) at refinery region (r). This column is generally upper bounded.

22. For each applicable combination of domestic crude oil and refinery region, balance shipments received directly from the producing region plus transhipments received from other refinery regions against crude oil consumed at the refinery and crude that is transhipped to other refinery regions:

$$\Sigma_b \Sigma_m Y_{c,v,b,r,m} + \Sigma_{r'} \Sigma_m Y_{c,v,r',r,m} - \Sigma_{r'} \Sigma_m Y_{c,v,r,r',m} - Ra_{c,v,r} \quad = \quad 0 \qquad \forall \quad c, v, r$$

where:

$Y_{c,v,b,r,m}$ is volume of crude oil type (c) with source code (v) produced in domestic region (b) that is shipped to refinery region (r) via mode (m).

$Y_{c,v,r',r,m}$ is volume of domestic crude oil type (c) with source code (v) that is received at refinery region (r) by transhipment through refinery region (r') via mode (m). (Not applicable in 3-region model.)

$Y_{c,v,r,r',m}$ is volume of domestic crude oil type (c) with source code (v) that is transhipped through refinery region (r) to refinery region (r') via mode (m). (Not applicable in 3-region model.)

$Ra_{c,v,r}$ is volume of domestic crude oil type (c) with source code (v) that is processed through the atmospheric tower at refinery region (r).

I.e., for each domestic crude oil at each refinery region, the volume consumed at the refinery plus what is shipped out of the refinery region must equal what is shipped into the refinery region.

23. For each applicable combination of imported crude oil and refinery region, balance imports received directly plus imports transhipments received from other refinery regions against crude oil consumed at the refinery and crude that is transhipped to other refinery regions:

$$\Sigma_q Pi_{c,r,q} + \Sigma_{r'}\Sigma_m Y_{c,F,r',r,m} - \Sigma_{r'}\Sigma_m Y_{c,F,r,r',m} - Ra_{c,F,r} - O_{SPR} = 0 \quad \forall \; c, r$$

where:

$Pi_{c,r,q}$ is the quantity of imported crude oil acquired by refinery region (r) of crude type (c) at price level (q).

$Y_{c,F,r',r,m}$ is volume of imported crude oil type (c) that is received at refinery region (r) by transhipment from refinery region (r') via mode (m). (Not applicable in 3-region model.)

$Y_{c,F,r,r',m}$ is volume of imported crude oil type (c) that is transhipped from refinery region (r) to refinery region (r') via mode (m). (Not applicable in 3-region model.)

$Ra_{c,F,r}$ is volume of imported crude oil type (c) that is processed through the atmospheric tower at refinery region (r).

O_{SPR} volume of imported oil moved to the Strategic Petroleum Reserve in PADD III

I.e., each imported crude oil must be balanced at each refinery by matching imports and what is received through transhipment against refinery consumption and what is transhipped to other refineries or to the SPR.

24. Balance each product at each refinery region:

$$Q_{p,r} + \Sigma_i H_{p,i,r} + \Sigma_q I_{p,r,q} + \Sigma_{p'} Tx_{s,p,r} - \Sigma_u\Sigma_m R_{u,r,m} * A_{p,u,r,m}$$

$$- \Sigma_m \Sigma_d W_{p,r,d,m} = 0 \quad \forall \; p, r$$

where:

$Q_{p,r}$ is volume of spec product (p) manufactured in refinery region (r). When product (p) is a spec product, column $H_{p,h,r}$ does not exist.

$H_{p,i,r}$ is volume of product (p) generated from gas plant stream (i) in refinery region (r). (Also includes $H_{p,h,r}$.)When this column is active for product (p), column $Q_{p,r}$ does not exist.

$I_{p,r,q}$ is volume of refined product (p) imported into refinery region (r) at price level (q).

$Tx_{s,p,r}$ is the volume of stream (s) transferred into product (p) in refinery region (r).

$R_{u,r,m}$ is the manufacturing activity level in mode (m) operation in processing unit (u) at refinery region (r). (Applicable to product N6I and N6B only.)

$A_{p,u.r,m}$ is the volume fraction of manufacturing activity level in mode (m) operation in processing unit (u) which defines the volume of product (p) produced (or consumed if the sign is negative) per unit of manufacturing activity level in refinery region (r).

$W_{p,r,d,m}$ is the volume of product (p) shipped from refinery region (r) to demand region (d) via mode (m).

I.e., for each product at each refinery, the volume manufactured plus volume imported plus volume transferred from another higher quality product must equal to the volume transferred to other lower quality products plus the amount consumed by recipe plus the volume shipped to market.

25. Balance each utility at each refinery region:

$$U_{l,r} + \Sigma_u \Sigma_m R_{u,r,m} * A_{l,u,r,m} - \Sigma_p Q_{p,r} * A_{l,p,r} - \Sigma_p \Sigma_h H_{p,h,r} * A_{l,h,r} = 0 \quad \forall \; r, l$$

where:

$U_{l,r}$ is the quantity of utility (l) that is purchased in refinery region (r). Of course (l) = KWH, STM, and NGF (power, steam, and natural gas fuel) with units in thousands of kWh, lbs., and Mcf respectively.

$R_{u,r,m}$ is the manufacturing activity level in mode (m) operation in processing unit (u) at refinery region (r).

$A_{l,u,r,m}$ is the quantity of utility (l) consumed (-) or manufactured (+) per unit of operation of processing unit (u) in mode (m) in refinery region (r). The (u) index includes the utility manufacturing units.

$Q_{p,r}$ is the volume of spec product (p) manufactured at refinery region (r). This column exists only when product (p) is a spec blend.

$A_{l,p,r}$ is the quantity of utility (l) consumed per unit of spec product (p) manufactured at refinery region (r).

$H_{p,h,r}$ is the volume of product (p) manufactured by recipe blend (h) at refinery region (r). This column exists only when product (p) is a spec blend.

$A_{l,h,r}$ is the quantity of utility (l) consumed per unit of recipe product (h) manufactured at refinery region (r).

I.e., for each utility at each refinery region, the quantity purchased plus the amount manufactured must equal the consumption.

26. Constrain each processing unit throughput to maximum capacity at each refinery region:

$$\Sigma_m R_{u,r,m} - K_{u,r} - A_{u,r}*(L_{u,r} + E_{u,r}) = 0 \quad \forall \; r, u$$

where:

$R_{u,r,m}$ is the manufacturing activity level in mode (m) operation in processing unit (u) at refinery region (r).

$K_{u,r}$ is the base processing capacity in processing unit (u) at refinery region (r) in Mbbl/cd. This column is upper bounded rather than fixed. See note below.

$A_{u,r}$ is the stream factor for processing unit (u) at refinery region (r) defined as the ratio of calendar day capacity to stream day capacity.

$L_{u,r}$ is the cumulative stream day capacity added for processing unit (u) at refinery region (r) during the previous simulated periods. This column is, of course, fixed.

$E_{u,r}$ is the stream day capacity added during this simulated period for processing unit (u) at refinery region (r). This column is generally upper bounded.

I.e., the activity of a particular processing unit must be limited to the maximum operating capacity.

Note: By making this row fixed with the base capacity upper bounded, the processing throughput is calculated as $K_{u,r} + A_{u,r}*(L_{u,r} + E_{u,r})$. Of course, in a model lacking capacity expansion capability, the capacity constraint row is commonly constructed as throughput and is equal to or less than a right-hand-side capacity value so that the throughput is merely the row activity.

27. Balance each intermediate refinery stream at each refinery region:

$$\Sigma_u \Sigma_m R_{u,r,m} * A_{i,u,r,m} + \Sigma_s (Tx_{s,i,r} - Tx_{i,s,r}) - \Sigma_p G_{i,p,r} - \Sigma_p \Sigma_h H_{p,i,r} * A_{i,h} = 0 \quad \forall \; i, r$$

where:

$i \in s$ i.e., the intermediate streams are a subset of all refinery streams.

$R_{u,r,m}$ is the manufacturing activity level in mode (m) operation in processing unit (u) at refinery region (r).

$A_{i,u,r,m}$ is the volume fraction of intermediate stream (i) created (or consumed if the sign is negative) per unit of manufacturing activity level in mode (m) operation for processing unit (u) at refinery region (r).

$Tx_{s,i,r}$ is the volume of stream (s) transferred to intermediate stream (i) at refinery region (r).

$Tx_{i,s,r}$ is the volume of intermediate stream (i) transferred to stream (s) at refinery region (r).

$G_{i,p,r}$ is the volume of intermediate stream (i) blended into spec blended product (p) at refinery region (r).

$H_{p,i,r}$ is volume of product (p) generated from gas plant stream (i) in refinery region (r). (Also includes $H_{p,h,r}$.)

$A_{i,h}$ is the volume fraction of product (p) for which intermediate stream (i) is consumed (per unit volume of product manufactured) according to recipe (h) at refinery region (r).

I.e., at each refinery region, each intermediate stream must be volume balanced so that the amount manufactured plus the amount transferred from higher quality streams must equal the volumes consumed by manufacturing processes, the amount that may be transferred to other lower quality streams plus the volumes blended into spec and recipe products.

28. Constrain qualities of spec blended products:

$$\Sigma_i G_{i,p,r} * A_{y,i,p} - Q_{p,r} * A_{y,p,r} \quad \leq, \geq, = \quad 0 \quad \forall \; y, p, r$$

where:

$G_{i,p,r}$ is the volume of intermediate stream (i) blended into spec-blended product (p) at refinery region (r).

$A_{y,i,p}$ is the blend value of spec blend property (y) for spec product (p) of stream (i).

$Q_{p,r}$ is the total volume of spec-blended product (p) manufactured at refinery region (r).

$A_{y,p,r}$ is the constraining value of property (y) that spec product (p) must adhere to; e.g., an octane number, at refinery region (r).

Note: The row type varies depending upon whether the blend specification quality is a maximum, minimum, or fixed value.

i.e., for each spec for each product at each refinery region, the aggregate spec value of the product determined by volume weighting the spec properties of the consumed blending stocks must not violate the specification limit.

29. Balance blending rows with specific blended products:

$$\Sigma_i G_{i,p,r} - \Sigma_{p'} Q_{p',r} = 0 \quad \forall \quad p, r$$

where:

p is only TRG, RFG, N6I, N6B, DSL, DSU, N2H, and JTA.

$p' \in p$ for p=TRG, p'=TRG, TRH, SST, SSE;
for p=RFG, p'=RFG, RFH, SSR;
for remaining products (p), p'=p only (i.e., for JTA, p'=JTA only).

$G_{i,p,r}$ is the volume of intermediate stream (i) blended into spec-blended product (p) at refinery region (r).

$Q_{p',r}$ is the total volume of spec-blended product (p) manufactured at refinery region (r).

30. Sum oxygen percentage contribution by oxygenates blended to reformulated gasoline

$$\Sigma_{OX} G_{OX,RFG,r} * PO_{OX} + \Sigma_{d \in r} H_{ETH,RFG,d} * PO_{ETH} * F_{ETHRFG} + \Sigma_{d \in r} H_{ETH,RFH,d} * PO_{ETH} * F_{ETHRFH} - ZOX_r = 0 \quad \forall \quad r$$

where:

$G_{OX,RFG,r}$ is the volume of oxygenate stream (OX) blended to RFG in region (r).

PO_{OX} is the percentage oxygen in oxygenate stream (OX). This group includes MTBE, TAM, and THM from methanol and ETBE, TAE, and THE from ethanol. Also included is the ethanol splash blended.

$H_{ETH,RFG,d}$ is the volume of RFG splash blended with ethanol in demand region d (within region r).

PO_{ETH} is the percentage oxygen in ethanol

F_{ETHRFG} is the fraction of ethanol in RFG for 2.0 percent min oxygen (= 5.8 percent).

$H_{ETH,RFH,d}$ is the volume of RFH splash blended with ethanol in demand region d (within region r).

F_{ETHRFH} is the fraction of ethanol in RFH for 2.7 percent min oxygen (= 7.8 percent).

ZOX_r is the total volume in units of percent oxygen-barrels of RFG.

31. Sum oxygen percentage contribution by *renewable oxygenates* blended to RFG.

$$\Sigma_{OE} G_{OE,RFG,r} * PO_{OE} + \Sigma_{d \in r} H_{ETH,RFG,d} * PO_{ETH} * F_{ETHRFG} + \Sigma_{d \in r} H_{ETH,RFH,d} * PO_{ETH} * F_{ETHRFH} - L * ZOX_r \geq 0 \quad \forall \quad r$$

where, in addition to the terms defined in equation 29:

$G_{OE,RFG,r}$ is the volume of renewable oxygenate stream (OE) blended to RFG in region (r).

PO_{OE} is the percentage oxygen in oxygenate stream (OE). This group includes ETB, TAE, and THE from ethanol. Also included is the ethanol splash blended.

L is the volume fraction of the oxygen that must come from *renewable oxygenates*, i.e., ethanol, ethylbenzene (ETB) and ethyl ethers (TAE and THE).

32. Calculate refinery consumption of natural gas supply by summing volumes processed for each refinery region:

$$\Sigma_q N_r NGRFN_q + \Sigma_q N_r NGRFP_q - U_r NGF = 0$$

where:

$N_r NGRFN/P_q$ is the volume of natural gas consumed in refinery region (r) at supply price delta (q).

$U_r NGF$ is the total volume of natural gas consumed in refinery region r.

33. Place an upper bound on each natural gas supply step volume:

$$N_r NGRFN/P_q \leq NG_q \text{ max}$$

where:

$N_r NGRFN/P_q$ is the volume of natural gas allowed on step q in region (r) at cost a cost delta.

NG_q max is the maximum volume of natural gas supply allowed, based on an input value. Note that step N1 has a minimum lower bound volume.

34. Calculate Alaskan crude export volumes:

$$\Sigma_q \, NZAMHP_q + \Sigma_q \, NZAMHN_q - ZZAMHTOT = 0$$

where:

$\Sigma_q \, NZAMHP_q$ Volume at price increment q of Alaskan Crude (AMH) exports.

$\Sigma_q \, NZAMHN_q$ Volume at price discount q of Alaskan Crude (AMH) exports.

35. Place an upper bound on each Alaskan crude supply step volume:

$$NZAMHx_q \leq NZAMHx_q \, max$$

where:

$NZAMHx_q$ Volume at price increment/discount q of Alaskan Crude (AMH) exports.

$NZAMHx_q max$ is the maximum volume of crude export allowed, based on an input value.

36. Calculate taxable carbon emissions:

$$TCBN_r - \Sigma_m \, R_{FUM,\, r,m} = 0 \qquad \forall \quad r$$

where:

$TCBN_r$ Total carbon emissions resulting from refinery operations in refinery region r for which tax is levied.

$R_{FUM,\, r,m}$ is the manufacturing activity level in mode (m) operation in processing unit FUM at refinery region (r).

37. Calculate ethanol available for tax subsidy:

$$\Sigma_q \, B_{ETH,d,q} - ZET_d = 0 \qquad \forall \quad d$$

where:

ZET_d Total ethanol production in demand region d for which tax subsidy is allowed.

$B_{ETH,d,q}$ Volume of ethanol produced in demand region d at price q.

38. Calculate volume of GTL movement from Alaska to refining regions:

$$\Sigma_m \, W_{GTL,A,r,m} - \Sigma_i \, HMPRF_{i,r} = 0 \qquad \forall \quad r$$

where:

$W_{GTL,A,r,m}$ Total GTL transferred from Alaska to region r via transportation mode m

39. Limit volume percent of undesireable oxygenate in RFG (and TRG)- related mogas manufactured at refinery r for sale or splash blending:

$$\Sigma_{ox} G_{ox,RFG,r} - \Sigma_{mg \in RFG} Q_{mg,r} * PO_{mg} \leq 0 \qquad \forall \quad r$$

and

$$\Sigma_{ox} G_{ox,TRG,r} - \Sigma_{mg \in TRG} Q_{mg,r} * PO_{mg} \leq 0 \qquad \forall \quad r$$

where:

$G_{ox,TRG,r}$ Volume of oxygenate ox (MTBE, ETBE, TAE, TAM for TRG component) blended or splash blended in the manufacture of TRG mogas components.

$G_{ox,RFG,r}$ Volume of oxygenate ox (MTBE, ETBE, TAE, TAM, THE, THM for RFG component) blended or splash blended in the manufacture of RFG mogas components.

$Q_{mg,r}$ Volume of spec product mg (SST, SSE, TRG, TRH for TRG component, and SSR, RFG, RFH for RFG component) manufactured in refinery region r.

PO_{mg} Percentage oxygen required in mogas component mg.

40. Maximum and minimum flow requirements along TAPS in Alaska:

$$TANSOTOT + TAGTLTOT > NSOmin$$

and

$$TANSOTOT + TAGTLTOT < NSOmax$$

where:

TANSOTOT Total oil transported from Alaska North Slope to Valdez along TAPS.

TAGTLTOT Total GTL transported fron Alaska North Slope to Valdez along TAPS.

NSO Min and Maximum (capacity) flow allowed on TAPS in Alaska.

41. Maximum natural gas available for processing to GTL in Alaska:

$$\Sigma_r \Sigma_q NGA_{r,q} \leq NGmax$$

$NGA_{r,q}$ Natural gas used at price q to generate GTL for transport to refinery region r

NSO Maximum quantity of natural gas allowed to be consumed for GTL production in Alaska.

42. In addition to the above, several non-constraining rows exist merely as a convenience to sum over certain columns via the row activity parameter, including rows CAALLTOT, CAAMHTOT, CAGTLTOT, CANSOTOT.

B.6 Row and Column Cross References

The PMM LP matrix is generated from a program written in the FORTRAN language using callable subroutines from OML. The correspondence between the rows and the column symbols in the preceding matrix description and the generated matrix names of PMM are shown in Table B4.

Table B4. Column Cross References

Column Notation	Matrix Name
$B_{a,d,q}$	C(d)(a)R(q)
$D_{p,d}$	D(d)(p)S1 & D(d)(p)SX
$E_{u,r}$	E(r)(u)INV
$G_{i,p,r}$	B(r)(p)(i) & F(r)(p)(i)
$H_{p,h,r}$	X(r)(h)(p)
$H_{p,h,d}$	X(d)(h)(p)
$H_{p,i,r}$	G(r)(i)(p)
$H_{g,i,r}$	G(r)(i)(g)
$HKWHMCH_r$	H(r)KWHMCH
$HMPRF_{i,r}$	H(r)MPRF(i)
$I_{p,r,q}$	I(r)(p)R(q)
$Iz_{p,d}$	I(d)(p)Z9
$K_{u,r}$	K(r)(u)CAP
$L_{u,r}$	L(r)(u)BLD
M_r	G(r)METDEM
Mt	D@METS1
$NGA_{r,q}$	N(r)NGKN(q)
N_rNGRFN_q	N(r)NGRFN(q)
Column Notation	**Matrix Name**

N_rNGRFP_q	N(r)NGRFP(q)
$NZAMHN_q$	NZAMHN(q)
$NZAMHP_q$	NZAMHP(q)
$O_{c,v,b}$	O@CRDEXP
O_{SPR}	O@CRDSPR
P_b	P(b)DCRQ1
$Pi_{c,r,q}$	P(r)F(c)Q(q)
$Q_{p,r}$	Q(r)(p)
$Ra_{c,v,r}$	R(r)ACU(v)(c)
Rc_r	R(r)CGNCGN, R(r)CGXCGN
$R_{u,r,m}$	R(r)(u)(m) & H(r)(u)(m)
Tu	T@UNFTOT
$T_{i,r}$	T(r)UNF(i)
TAAMHXZ	TAAMHXZ
$TATOT_c$	TAA(c)TOT
TAGTL	TAGTLTOT
TANSO	TANSOTOT
$TCBN_r$	T(r)CBNTAX
$Tx_{s,p,r}$	T(r)(s)(p)
$Tx_{p,p',r}$	T(r)(p)(p')
$Tx_{s,s',r}$	T(r)(s)(s')
$Tx_{FR,TO,i,r}$	H(r)(FR)(TO)(i)
$U_{l,r}$	U(r)(l)
U_rNGF	U(r)NGF
Vcj	VTVC(m)CP

Column Notation	Matrix Name
V_{pc}	VTVP(m)CP
Vcts	VTPCGAC (not used)
$Vcp_{b,r}$	VTPC(b)(m)(r)
$Vpp_{r,d,m}$	VTPP(r)(m)(d)
$Vtpl_{r,d,m}$	VTPL(r)(m)(d)
$WGTL_{m,r}$	WAGTL(m)(r)
$W_{a,r,d,m}$	W(r)(a)(m)(d)
$W_{a,d,r,m}$, $W_{a,d,d',m}$	W(d)(a)(m)(r), W(d)(a)(m)(d')
$W_{p,r,d,m}$	W(r)(p)(m)(d)
$W_{p,d,d',m}$	W(d)(p)(m)(d')
$X_{p,d}$	D(d)(p)SX
$Xz_{p,d}$	D(d)(p)Z9
$Y_{c,v,b,r,m}$	Y(b)(v)(c)(m)(r)
Zt	Z@TOTCRD
ZET_d	Z(d)ETHTAX Z(d)ETCTAX
Z_rFLO_u	Z(r)FLO(u)
ZOX_r	Z(r)RFGOXY
ZZAMHTOT	ZZAMHTOT

Table B5. Row Cross References

Row Number	Matrix Name
1	Z@IRACN
2	Z@IRACX
3	Z@CRDTOT
4	A@PRDIMP
5	D@MET
6	A(r) INVST
7	A@INVST
8	F@TOTCRD
9	F(r)UNF(i)
10	D(d)(a)
11	D(d)(p)
12	D(d)(p')
13	M(d)(p)
14	C(b)(v)(c), (CZAMH)
15	TVC(m)CP
16	TVP(m)CP
17 (not used in 3-reg model)	TPCGAC
18	TPC(b)(m)(r)
19	TPP(r)(m)(d)
20	TPL(r)(m)(d)
21	P(r)(e), Z(r)CAP(u), H(r)(e), G(r)(e)
22	C(r)(v)(c)
23	C(r)F(c)
24	M(r)(p)
25	U(r)(l)
26	L(r)(u)CAP, H(r)FUMCAP
27	B(r)(i), G(r)(i), H(r)(i)
28	Q(r)(p)(y)
29	S(r)(p)E
30	S(r)RFGOXY
31	Q(r)RFGREN
32	Z(r)NGFSUM
33*	N(r)NGRFN/P(q)
34	ZZAMHSUM
35*	NZAMHN/P(q)
36	P(r)CBNTAX
37	A(d)ETHTAX
38	C(r)GTL
39	M(r)MTBRFG, M(r)MTBTRG
40	TAOILGTN, TAOILGTX
41	TANGKGTX
42	(misc)

* Bound on column variable.

APPENDIX C

Bibliography

APPENDIX C Bibliography

1. Abt Associates Inc. 1991. *A Methodology for Evaluating the Costs and Global Warming Implications of Ethanol*. Report prepared for Office of Policy Analysis, U.S. Environmental Protection Agency, April 2, 1991.

2. F. Gerard Adams and James M. Griffin, "An Economic-Linear Programming Model of the U.S. Petroleum Refining Industry," *Journal of the American Statistical Association* (September 1972), 67:339, pp. 542-551.

3. ASM Consulting Services, *Petroleum Market Module Review*, prepared for the Office of Statistical Standards, EIA/DOE, (July 1992).

4. American Petroleum Institute, *How Much We Pay For Gasoline: 1996 Annual Review*, (Washington, DC, May 1997).

5. ASM Consulting Services, *Petroleum Market Module Review*, prepared for the Office of Statistical Standards, EIA/DOE, (September 1992).

6. C.E. Bodington, "A History of Mathematical Programming in the Petroleum Industry," *Interfaces* (July-August 1990), 20:4, pp. 117-127.

7. Bureau of the Census, *1989 Annual Survey of Manufacturers, Statistics for Industry Groups and Industries* (M89AS1).

8. Chowdhury, A., E.O. Heady, and S. Bhide. 1981. *Optimum Crop Production and Resource Use Under Alternative Energy Prices and Agricultural Exports: A Separable and Change-Constrained Programming Analysis*. Report 103. The Center for Agricultural and Rural Development, Ames, IA.

9. Christensen, D.A., R.J. Schatzer, E.O. Heady, and B.C. English. 1981. *The Effects of Increased Energy Prices on U.S. Agriculture: An Econometric Approach*. Report 104. The Center for Agricultural and Rural Development, Ames, IA.

10. Decision Analysis Corporation of Virginia, *The Oil Trade Model (OTM) Draft Documentation, User's Manual [and] Data Base*, (Vienna, VA, October 31, 1989).

11. Decision Analysis Corporation of Virginia, *Transportation and Refining of International Petroleum (TRIP) Model Documentation, Volume I* (Vienna, VA, December 1988).

12. Decision Analysis Corporation of Virginia, *Transportation and Refining of International Petroleum (TRIP) Model Documentation, Volume II* (Vienna, VA, December 1988).

13. DiPardo, Joseph. April 2000. *Outlook for Biomass Ethanol Production and Demand*. Energy Information Administration, U.S. Department of Energy.

14. Dvoskin, D., E.O. Heady, and B.C. English. 1978. *Energy Use in U.S. Agriculture: An Evaluation of National and Regional Impacts from Alternative Energy Policies*. The Center for Agricultural and Rural Development. Report 78. Ames, IA.

15. Energy Information Administration, *Annual Energy Review*, DOE/EIA-0384(99)(Washington, DC, July 2000).

16. Energy Information Administration, *Component Design Report: The Integrating Module*, (Washington, DC, May 1992).

17. Energy Information Administration, *Component Design Report: The Macroeconomic Activity Module of the National Energy Modeling System*, (Washington, DC, March 1992).

18. Energy Information Administration, *Fuel Oil and Kerosene Sales 1999*, DOE/EIA-0535(99)(Washington, DC, September 2000).

19. Energy Information Administration, *International Energy Annual 1998*, DOE/EIA-0219(98)(Washington, DC, January 2000).

20. Energy Information Administration, *Modeling Environmental Costs*, memorandum prepared by S.R. MacIntyre (Washington, DC, February 1993).

21. Energy Information Administration, "Motor Fuels Tax Trands and Assumptions", *Issues in Midterm Forecasting 1998*, DOE/EIA-0607(98), (Washingtion DC, July 1998).

22. Energy Information Administration, *Natural Gas Annual 1999*, DOE/EIA-0131(99), (Washington, DC, October 2000).

23. Energy Information Administration, *NEMS Crude Oil Categories*, memorandum prepared by Xavier Szebrat (Washington, DC, July 1992).

24. Energy Information Administration, *Outstanding Issues of Refinery Plant Expansion*, memorandum prepared by G.R. Harp (Washington, DC, May 1993).

25. Energy Information Administration, *Petroleum Marketing Annual 1998*, DOE/EIA-0487(98) (Washington, DC, January 2000).

26. Energy Information Administration, *Petroleum Supply Annual 1999*, DOE/EIA-0340(99/1)(Washington, DC, June 2000).

27. Energy Information Administration, *PMM Procedure to Calculate Investment Recovery and Fixed Cost Charges on Existing U.S. Refinery Plant*, memorandum prepared by G.R. Harp (Washington, DC, August 1993).

28. Energy Information Administration, *Preliminary Design for the PMM and Detailed Refinery Model*, memorandum prepared by S.R. MacIntyre (Washington, DC, February 1992).

29. Energy Information Administration, *Proceedings of the National Energy Modeling System Conference*, (Washington, DC, May 1993).

30. Energy Information Administration, *Requirements Analysis for a National Energy Modeling System* (Washington, DC, 18 May 1992).

31. Energy Information Administration, *Test Plan for the Petroleum Market Module (PMM)*, memorandum prepared by G.R. Harp (Washington, DC, May 1993).

32. Energy Information Administration, *The Treatment of Foresight and Expectation*, Decision Memorandum, Mary J. Hutzler and Mark E. Rodekohr, (Washington, DC, March 12, 1992).

33. Energy Information Administration, *NEMS Component Design Report for Biofuels (Ethanol) Supply Submodule - Renewable Fuels Model - National Energy Modeling System*. Draft 3/12/93.

34. English, Smith, Atwood, Johnson, and Oamek. 1989. *Resource Conservation Act Analysis; An Overview of the CARD Agricultural Resource Interregional Modeling System*. The Center for Agricultural and Rural Development, Iowa State University, Ames, IA, Technical Report 89-TR11.

35. Ensys Energy & Systems, Inc., *WORLD Reference Manual*, prepared for the Department of Energy, Contract No. DE-AC01-87FE-61299, September 1992.

36. Farmer, Richard D., "Problems and Lessons in Estimating Supply Curves for Refined Petroleum Products," *The Journal of Energy and Development*, Autumn, 1986, Vol. XII, No. 1, pp. 27-42.

37. J.H. Gary and G.E. Handwerk, *Petroleum Refining: Technology and Economics* (New York: Marcel Dekker, 1975), Chapters 13 and 14.

38. Giovando, CarolAnn June 1998. “1998 Powerplant Awards Sweeny Cogeneration Facility”, *Power*.

39. Haverly Systems, Inc., *OMNI Linear Programming System, User Reference Manual,* (Denville, NJ, January 1987).

40. ICF Resources, *North American Crude Oil Demand Model*, July 1992.

41. Kane, S.M., and J.M. Reilly. 1989. *Economics of Ethanol Production in the United States*. Agricultural Economic Report 607. USDA/ERS, Washington, DC.

42. Ketron Management Science, Inc., *MPS III Mathematical Programming System, User Manual*, (Arlington, VA, August 1988).

43. Ketron Management Science, Inc., *Optimization and Modeling Subroutine Library, User Manual*, Draft, (Arlington, VA, 1992).

44. LeBlanc, M. and J. Reilly. 1988. *Ethanol: Economic and Policy Tradeoffs*. Agricultural Economic Report No. 585. Resources and Technology Division, Economic Research Service, U.S. Department of Agriculture, Washington, DC.

45. Lee, Hyunok. June 1993. "Ethanol's Evolving Role in the U.S. Automobile Fuel Market." *Industrial Uses of Agricultural Materials*, Situation and Outlook Report. United States Department of Agriculture, Washington, DC.

46. Lee, R., S.M. Cohn, and R.D. Perlack. 1991. *Prototype of an Integrated Model for Projecting Biofuels Consumption*. Draft report prepared for Energy Information Administration, U.S. Department of Energy. Oak Ridge National Laboratory, Oak Ridge, TN.

47. Linear Programming, Inc., *PAL/REMS Integration Methodology Study*, final report prepared for the Office of Statistical Standards, EIA/DOE, Contract No. DE-AC01-84EI-19633 (April 1987).

48. Macro International, Inc., *EIA-888 and EIA-878 Data Comparisons and Performance Measures*, Third Quarter 1997 (Washington, DC, December 15, 1997).

49. Marland, G. and A.F. Turhollow. 1991. “CO_2 Emissions From the Production and Combustion of Fuel Ethanol from Corn”. *Energy*, 16(11/12):1307-1316.

50. National Petroleum Council, *Petroleum Storage and Transportation*, (Washington, DC, April 1989).

51. National Petroleum Council, *U.S. Petroleum Refining* (Washington, DC, June 2000).

52. National Renewable Energy Laboratory 1999. *Bioethanol Multi-Year Technical Plan*.

53. National Research Council, *The National Energy Modeling System* (Washington, DC, January 1992).

54. Oak Ridge National Laboratory, *Review of the Component Design Report for the Petroleum Market Module*, letter prepared by Jerry Hadder (Oak Ridge, TN, July 1992).

55. The Pace Consultants, Inc., *Pace Petrochemical Service: Annual Issue*, (Houston, TX, October 1990).

56. Solar Energy Research Institute. March 1990. *The Potential of Renewable Energy: An Interlaboratory White Paper,* SERI/TP-260-3674, Golden, CO, prepared for the Office of Policy, Planning, and Analysis, U.S. Department of Energy, in support of the National Energy Strategy.

57. Trumble, David A. 1994. *Estimation of Supply Curve for Ethanol with Corn as the Feedstock.* Draft report prepared for Energy Information Administration, U.S. Department of Energy. Oak Ridge National Laboratory, Oak Ridge, TN.

58. Turhollow, A.F. 1991. *Economic Consideration for the Production of Wood for Energy*. Presentation at Trees for Energy, First National Fuelwood Conference, November 11-13, 1991, Lincoln, NE.

59. Turhollow, A.F., D.A. Christensen, and E.O. Heady. 1984. *The Potential Impacts of Large-Scale Fuel Alcohol Production from Corn, Grain Sorghum, and Crop Residues Under Varying Technologies and Crop Expert Levels*. Report 126. The Center for Agricultural and Rural Development, Ames, IA.

60. Turhollow, A.F., C. Short, and E.O. Heady. 1983. *Potential Impacts of Future Energy Price Increases on U.S. Agricultural Production*. Report 116. The Center for Agricultural and Rural Development, Ames, IA.

61. Tyson, K.S. 1990. *Biomass Resource Potential of the United States*. Report prepared for Office of Transportation Technologies, U.S. Department of Energy, October 1990. Solar Energy Research Institute, Golden, CO.

62. The University of Oklahoma, *Review of the Component Design Report for the Petroleum Market Module*, letter prepared by D.J. O'Neil (Norman, OK, July 1992).

63. U.S. Department of Agriculture, *USDA Agricultural Baseline Projections to 2009*, Staff Report WAOB-00-1 (Washington, DC, February 2000).

64. U.S. Department of Energy, *Integrated Analysis Supporting the National Energy Strategy: Methodology, Assumptions, and Results*, Technical Annex 2, First Addition, 1991/1992, DOE/S-0086P, p.148.

65. U.S. Department of Energy, *National Energy Strategy* (Washington, DC, February 1991).

66. U.S. Department of Energy, *Review of the Component Design Report for the Petroleum Market Module*, memorandum prepared by A.E. Haspel (Washington, DC, June 1992).

67. U.S. Department of Energy, *Review of the Component Design Report for the Petroleum Market Module*, memorandum prepared by J.D. Pearson (Washington, DC, June 1992).

68. U.S. Department of Energy, *Review of the Component Design Report for the Petroleum Market Module*, memorandum prepared by J.L. Petersen (Washington, DC, June 1992).

69. U.S. Department of Energy, *Review of the Component Design Report for the Petroleum Market Module*, memorandum prepared by C.P. Shirkey (Washington, DC, June 1992).

70. Urbanchuk, J.M. 1998. *Review of Alternative Ethanol Supply Curves Used in the Energy Information Administration's National Energy Modeling System.*

71. Walsh, M. et al June 1997. *Evolution of the Fuel Ethanol Industry: Feedstock Availability and Price*. Oak Ridge National Laboratory.

72. Wang, M. et al 1997. *Fuel-Cycle Fossil Energy Use and Greenhouse Gas Emissions of Fuel Ethanol Produced from U.S. Midwest Corn.*

73. Wharton Econometric Forecasting Associates, *WEFA Energy Analysis Quarterly*, Winter 1992.

74. Also see citations in Appendix A.

APPENDIX D

Model Abstract

APPENDIX D Model Abstract

D.1 Model Name:

Petroleum Market Model

D.2 Model Acronym:

PMM

D.3 Description:

The Petroleum Market Model is a simulation of the U.S. petroleum industry. It includes 12 domestic crude oil production regions, 3 refining centers with full processing representations and capacity expansion capability and gas plant liquid production, and 9 marketing regions. The heart of the model is a linear program optimization which ensures a rational economic simulation of decisions of petroleum sourcing, resource allocations, and the calculation of marginal price basis for the products. Nineteen refined products are manufactured, imported, and marketed. Eight of these products are specification blended, while the remaining 11 are recipe blended. Capacitated transportation systems are included to represent existing intra-U.S. crude oil and product shipments (LPG, clean, dirty) via pipeline, marine tanker, barge, and truck/rail tankers. The export and import of crude oil and refined products is also simulated. All imports are purchased in accordance with import supply curves. Domestic manufacture of methanol is represented as though the processing plants were a part of the refinery complexes whereas ethanol sources are treated as merchant. Transportation is allowed for ethanol shipments to the demand region terminals for splash blending.

The program is written in FORTRAN which includes callable subroutines allowing full communication with the LP portion of the model which is in the form of an MPS resident file.

D.4 Purpose of the Model:

The PMM models domestic petroleum refining activities, the marketing of petroleum products to consumption regions, the production of natural gas liquids in gas processing plants, and domestic methanol production. The purpose of the PMM is to project petroleum product prices, refining activities, and movements of petroleum into the United States and among domestic regions. In addition, the model contains adequate structure and is sufficiently flexible to examine the impact of a wide variety of

petroleum-related issues and policy options, in order to foster understanding of the petroleum refining and marketing industry as well as determine the effects of certain policies and regulations.

The PMM projects sources of supply for meeting petroleum product demand. The sources of supply include crude oil, both domestic and imported; other inputs including alcohols and ethers; natural gas plant liquids production; petroleum product imports; and refinery processing gain. In addition, the PMM estimates domestic refinery capacity expansion and fuel consumption. Product prices are estimated at the Census division level and much of the refining activity information is at the Petroleum Administration for Defense (PAD) District level.

D.5 Most Recent Model Update:

October 2001

D.6 Part of Another Model?

National Energy Modeling System (NEMS)

D.7 Model Interfaces:

Receives information from the International, Natural Gas Transmission and Distribution, Oil and Gas Supply, Renewable Fuels, Electricity Market, Residential, Commercial, Industrial, and Transportation Models. Delivers information to each of the models listed above plus the Macroeconomic Model.

D.8 Official Model Representative:

Han-Lin Lee
Office of Integrated Analysis and Forecasting
Oil and Gas Division
(202) 586-4247

D.9 Documentation:

EIA Model Documentation: Petroleum Market Model of the National Energy Modeling System (NEMS), January 2002. (DOE/EIA-M059).

D.10 Archive Media and Installation Manual

Archived as part of the NEMS *AEO2002* production runs.

D.11 Energy System Described:

Petroleum refining industry and refined products market.

D.12 Coverage:

Geographic: Twelve domestic crude oil production regions (East Coast, Gulf Coast, Mid-Continent, Permian Basin, Rocky Mountain, West Coast, Atlantic Offshore, Gulf Offshore, Pacific Offshore, Alaska South, Alaska North, Alaska Offshore); three refining regions (PAD District I, an aggregate of PAD Districts II-IV, and PAD District V); nine market regions, the Census divisions (New England, Mid Atlantic, East North Central, West North Central, South Atlantic, East South Central, West South Central, Mountain, Pacific)

Time Unit/Frequency: Annual, 1990 through 2020

Products: LPG, conventional motor gasoline, conventional high oxygen motor gasoline, reformulated motor gasoline, reformulated high oxygen motor gasoline, M85, E85, jet fuel, distillate fuel oil, low-sulfur highway diesel, ultra-low-sulfur highway diesel, low-sulfur residual fuel oil, high-sulfur residual fuel oil, petrochemical feedstocks, asphalt/road oil, marketable coke, still gas, other.

Refinery Processes: crude distillation, vacuum distillation, delayed coker, fluid coker, visbreaker, fluid catalytic cracker, thermal cracker, hydrocracker-dist, hydrocracker-resid, solvent deasphalter, resid desulfurizer, FCC feed hydrofiner, distillate HDS, naphtha hydrotreater, catalytic reformer-450 psi, catalytic reformer-200 psi, alkylation plant, catalytic polymerization, pen/hex isomerization, butane isomerization, etherification, butanes splitter, dimersol, butylene isomerization, total recycle isomerization, naphtha splitter, C2-C5 dehydrogenator, cyclar unit, hydrogen plant, sulfur plant, aromatics recovery plant, lube + wax plants, FCC gasoline splitter, gas/H2 splitter, stream transfers, fuel system, steam production, power generation.

Crude Oil: Alaska low-sulfur light, Alaska mid-sulfur heavy, domestic low-sulfur light, domestic mid-sulfur heavy, domestic high-sulfur light, domestic high-sulfur heavy, domestic high-sulfur very heavy, imported low-sulfur light, imported mid-sulfur heavy, imported high-sulfur light, imported high-sulfur heavy, imported high-sulfur very heavy.

Transportation Modes: Jones Act dirty marine tanker, Jones clean marine tanker, LPG marine tanker, import tankers, clean barge, dirty barge, LPG pipeline, clean pipelines, dirty pipelines, rail/truck tankers. These cover all significant U.S. links.

D.13 Modeling Features:

Model Structure: FORTRAN callable subroutines which update the linear programming matrix, re-optimize, extract and post-process the solution results, update system variables, and produce reports.

Model Technique: Optimization of linear programming representation of refinery processing and transportation which relates the various economic parameters and structural capabilities with resource constraints to produce the required product at minimum cost, thereby producing the marginal product prices in a manner that accounts for the major factors applicable in a market economy.

Special Features: Choice of imports or domestic production of products is modeled, capacity expansion is determined endogenously, product prices include fixed and environmental costs, oxygenated and reformulated gasolines and low-sulfur diesel fuel are explicitly modeled.

D.14 Non-DOE Input Sources:

Information Resources Inc. (IRI), WORLD model data, National Petroleum Council, ICF Resources, Oil and Gas Journal.

D.15 DOE Input Sources:

Forms:

EIA-14	Refiners' Monthly Cost Report
EIA-182	Domestic Crude Oil First Purchase
EIA-782A	Monthly Petroleum Product Sales
EIA-782B	Reseller/Retailer's Monthly Petroleum Product Sales
EIA-782C	Monthly Petroleum Products Sold into States for Consumption
EIA-759	Monthly Power Plant Report
EIA-810	Monthly Refinery Report
EIA-811	Monthly Bulk Terminal Report
EIA-812	Monthly Product Pipeline Report
EIA-813	Monthly Crude Oil Report
EIA-814	Monthly Imports
EIA-817	Monthly Tanker and Barge Movement
EIA-820	Annual Refinery Report
EIA-826	Monthly Electric Utility Sales
EIA-856	Monthly Foreign Crude Oil Acquisition
EIA-860B	Annual Nonutility Power Producer Report
FERC-423	Monthly Report of Cost and Quality of Fuels for Electric Plants

In addition to the above, information is obtained from several Energy Information Administration formal publications: *Petroleum Supply Annual*, *Petroleum Supply Monthly*, *Petroleum Marketing Annual*, *Petroleum Marketing Monthly*, *Fuel Oil and Kerosene Sales*, *Natural Gas Annual*, *Natural Gas Monthly*, *Annual Energy Review*, *Monthly Energy Review*, *State Energy Data Report*, *State Energy Price and Expenditure Report*.

D.16 Independent Expert Reviews Conducted:

Independent reviews of the Component Design Report of the PMM were conducted by:

A.S. Manne, ASM Consulting Services, July 1992
A.S. Manne, ASM Consulting Services, September 1992
N. Yamaguchi, Trans-Energy Research Associates, Inc., November 1997.
J. Urbanchuk, AUS Consultants, May 1998.

D.17 Status of Evaluation Efforts by Sponsor:

None.

APPENDIX E

Data Quality

APPENDIX E. Data Quality

E.1 Quality of Distribution Cost Data

Costs related to distributing petroleum products to end-users are incorporated by adding fixed transportation markups to the wholesale prices which include the variable and fixed refinery costs. Transportation markups for petroleum products are estimated as the average annual difference between retail and wholesale prices over the years 1990 through 2000[1]. The differences are based on **wholesale prices** in the producing Census division and **end-use prices** (which do not include taxes) in the consuming Census division. See Appendix F for a discussion of programs and input files used in estimating these markups.

Annual **wholesale prices** for all petroleum products are aggregated from state-level prices from the EIA-782A. The estimation and reliability of the EIA-782A data is discussed in the *Petroleum Marketing Annual 2000 (*http://www.eia.doe.gov/pub/oil_gas/petroleum/data_publications/petroleum_marketing_annual/current/pdf/enote.pdf) , Explanatory Notes, August 30, 2001). See Explanatory Notes for inputs and sources.

With the exception of gasoline, non-utility distillate fuel, and jet fuel, **sectoral end-user prices** through 1997 are aggregated from prices from the State Energy Price and Expenditures Report 1997 (SEPER). The methodology behind these state-level sectoral prices are discussed in the *State Energy Price and Expenditures Report 1997 (*ftp://ftp.eia.doe.gov/pub/pdf/multi.fuel/037697.pdf, December 8, 2000*)* . One area of concern is the SEPER methodology for residual fuel transportation prices where the vessel bunkering prices are tied to State electric utility prices. This methodology is questionable because most utility residual fuel use is on the East Coast whereas vessel bunkering is concentrated on the Gulf and West Coasts. The entire kerosene price series was re-estimated for SEPERS97.

Gasoline, jet fuel, and non-utility distillate prices are estimated as weighted averages using end-user prices from EIA-782A and sectoral consumption from the *State Energy Data Report* (SEDS) (http://www.eia.doe.gov/emeu/sedr/contents.html , May 2001).

Due to a lag in the publication of the SEPER data, end-use price estimates for 1998, 1999, and 2000 are calculated using the same data series and methodology described in SEPER. The SEPER methodology uses prices from EIA-782A, FERC-423, EIA-759, and weights them with most recent consumption volumes from SEDS. Refer to SEDS for a discussion of the reliability of consumption data (http://eia.doe.gov/pub/state.data/pdf/petrol.pdf, May 2001).

Due to electric restructuring the number of electric utility survey respondents has been declining and the quality of the data is likely to be reduced. See the Electric Power Annual 1999 (http://www.eia.doe.gov/cneaf/electricity/epav1/epav1_sum.html, December 8, 2000) for a discussion of the reliability of electric utilities data from FERC-423 and EIA-759. Table E1 shows the data series used in the estimation of end-use prices.

[1]Transportation markups for kerosene are based on the difference between end-user kerosene prices and wholesale distillate prices.

Table E1. Sources of Markup Inputs

Products	Sectors	Data Series Inputs
Distillate	CM, IN, RS	EIA-782A, SEDS
Jet Fuel	TR	EIA-782A, SEDS
Low Sulfur Diesel Fuel	TR	EIA-782A, SEDS
Motor Gasoline	CM, IN, TR	EIA-782A, SEDS
Asphalt and Road Oil	IN	SEPER, EIA-782A, SEDS
Kerosene	CM, IN, RS	SEPER, EIA-782A, SEDS
Liquified Petroleum Gases	CM, IN, RS, TR	SEPER, EIA-782A, SEDS
Low Sulfur Residual Fuel	CM, IN	SEPER, EIA-782A, SEDS
High Sulfur Residual Fuel	TR	SEPER, EIA-782A, SEDS
Distillate	EU	SEPER, EIA-759, FERC-423
Low Sulfur Residual Fuel	EU	SEPER, EIA-759, FERC-423
High Sulfur Residual Fuel	EU	SEPER, EIA-759, FERC-423

E.2 Quality of Tax Data

In the PMM, State and Federal taxes are added to the prices of gasoline, distillate fuel, liquefied petroleum gas (LPG), jet fuel, ethanol and methanol in the transportation sector. State taxes are assumed to keep pace with inflation (held constant in real terms) while Federal taxes are held at current nominal levels (deflated in each forecast year).[2] The Federal tax assumption supports the overall forecast assumption of current laws and legislation. The assumption that State taxes will increase at the rate of inflation supports an implied need for additional highway revenues as driving increases. An additional 2 cents per gallon is added to the State gasoline taxes to approximate local taxes. The average local tax estimate was taken from *How Much We Pay for Gasoline: 1996 Annual Review* published by the American Petroleum Institute in May 1997.

The State taxes are added as Census division weighted averages which are based on the most recently-available State taxes. State taxes for jet fuel are derived from unpublished data collected by the Petroleum Marketing Division of EIA, while state taxes for ethanol and methanol are taken from The Clean Fuels and Electric Vehicles Report, February 2000, published by Energy Futures Inc. State and Federal taxes for gasoline, transportation distillate, and LPGs are based on data from the Federal Highway Administration, but are modified to include other known changes to State taxes. The quality of the State level tax data is unknown but deemed reliable. The local tax estimate of 2 cents-per-gallon seems reasonable given that a comparison of two EIA data series, one including local taxes and one not, revealed a gasoline price difference of 1.6 cents-per-gallon.[3] Federal taxes, which were increased by 4.3 per gallon in 1993, are widely published, and deemed highly reliable.

See Appendix F for a description of programs and input files used in the calculation of historical taxes and the estimation of taxes used in the price projections.

E.3 PMM Critical Variables

The PMM contains numerous variables and parameters. Some variables have greater impact on model results than others. The following is a list of variables that we believe has a high degree of influence on PMM results. It is provided to help users understand the critical factors affecting the PMM.

- World oil price
- Product demands
- Imported crude supply curves
- Imported product supply curves
- Domestic crude production
- Prices and available supplies of methanol, ethanol, MTBE, and other ethers

[2]Refer to Energy Information Administration, "Motor Fuels Tax Trends and Assumptions", Issues in Midterm Analysis and Forecasting 1998, DOE/EIA-0607(98), (Washington, D.C., July 1998).

[3]Macro International, Inc., *EIA-888 and EIA-878 Data Comparisons and Performance Measures*, Third Quarter 1997 (Washington, D.C., December 15, 1997).

- Investment cost for capacity expansion
- Market shares for gasoline and distillate types
- NGL supply volumes

Most of these variables are provided by other models in the NEMS system. Ethanol supply and prices are provided by the Ethanol Supply Model, a submodule of the PMM, documented in Appendix I. The investment cost and market share data are developed offline and read in to the PMM.

Tests on some of these variables are discussed in a separate appendix to this documentation, titled *Documentation of the Petroleum Market Model, Appendix: Model Developer's Report*. In an earlier effort, a detailed calibration was made, using the same refinery model technology database as in PMM, compared against the EIA Petroleum Supply Annual (PSA) for 1989.[4] This comparison indicated that the validation was quite good with total crude volumes from the model only 0.48 percent above actual PSA values and total production just 0.28 percent higher. A more complete comparison, including assumptions, can be found in the cited reference.

[4]"U.S. Detailed Refinery Model," Letter of October 20, 1993, from Martin Tallett of EnSys to G. H. Harp of EIA.

APPENDIX F

Estimation Methodologies

APPENDIX F. Estimation Methodologies

F.1 Refinery Investment Recovery Thresholds

The threshold for expansion investment decisions is composed of the refinery capital recovery threshold ($CRT_{i,j}$) and the fixed costs (FC_i) associated with a processing unit. The negative of the sum of these two components is entered into the objective function row corresponding to the unit expansion vectors [E(r)(un)INV and B(r)(un)BLD] of the PMM LP to provide an investment decision criteria; i.e., the investment will occur to the extent that it is economic. The methodologies used to calculate the capital recovery threshold and the fixed costs are presented below.

Refinery Capital Investment Recovery Threshold

The calculation methodology for the capital investment recovery threshold values have been taken from a standard refinery industry reference.[1] The inside battery limits (ISBL) investment cost and labor costs for most of the processing unit types were obtained from a study by Bonner and Moore Associates.[2] The data for typical unit sizes and stream factors as well as supplementary investment and labor was obtained from the World Oil Refining, Logistics, and Distribution (WORLD) model.[3] A basic premise used in the PMM application is that the investment recovery value at the end of project life (PL) will equal site decommissioning cost, an expectation that seems to be widely shared in the petroleum industry.

The inside battery limits cost per barrel investment cost at the Gulf Coast (P) for each refinery process modeled was obtained from the cited data sources. The total investment cost (INV) was then calculated using the referenced methodology. Finally, the capital recovery threshold for each process was calculated.

Given that a Gulf Coast plant and equipment battery limits cost for an expansion of processing plant type I is $\boldsymbol{P_i}$ dollars per barrel per day of stream day capacity (assumed to include required storage facilities but not necessarily all mandated environmental systems), then the nominal total plant and equipment cost at some arbitrary location would be:

[1]J.H. Gary and G.E. Handwerk, *Petroleum Refining: Technology and Economics* (New York: Marcel Dekker, 1975), Chapters 13 and 14; and (New York: Marcel Dekker Inc., 1994), Chapters 17 and 18.

[2]Bonner & Moore Associates, Inc., *A Capital Expansion Methodology Review of the Department of Energy's Petroleum Market Model*, prepared for the United States Department of Energy, Contract No. EI-94-25066 (Houston, TX, July 1994).

[3]Ensys Energy & Systems, Inc., *WORLD Reference Manual*, a reference for use by the analyst and management prepared for the United States Department of Energy, Contract No. DE-AC-01-87FE-61299 (Washington, D.C., September 1992).

$$NTPE_i = P_i * (1 + U) * (1 + O) \quad \textbf{(1)}$$

where U = Utilities cost multiplier
O = Offsites multiplier

With construction occurring over a 3-year period at a constant rate, the present value of the nominal total plant and equipment is:

$$PVNTPE_i = (NTPE_i/3) * \Sigma^{3}_{j=1} * (1+DEFi)^{j} \quad \textbf{(2)}$$

where DEF = inflation rate during construction

The final land, plant, and equipment cost is:

$$FLPE_i = PVNTPE_i * (1 + S + C) \quad \textbf{(3)}$$

where S = Special costs multiplier (includes land)
C = Contingency cost multiplier

The total investment is:

$$INV_i - (1 + W) * FLPE_i \quad \textbf{(4)}$$

where W = Working capital multiplier

A multiplier called the plant depreciation factor or PDF, is used in the investment recovery calculations. It is merely the ratio of depreciable investment to total investment. More specifically, the numerator of the PDF is total investment less land, working capital, and supplies.

The capital recovery threshold is calculated from:

$$CRT_{i,j} = CAPREC_{i,j} * L_j * E_j \quad \textbf{(5)}$$

where,

$$CAPREC_{i,j} = (INV_i * (U_{R,PL} - PDF_i * (TR_j/(PL*365))) / (1 - TR_j) \quad \textbf{(6)}$$

$CRT_{i,j}$ = The daily investment recovery required for processing unit type I at location j, $/bbl

$CAPREC_{i,j}$ = Base level daily investment recovery required for processing unit type I at location j, $/bbl

INV_i = The investment required for processing unit type I on the Gulf Coast, $/bbl

L_j = Location factor for PAD District j

E_j = Investment factor for environmental capital expenditure for PAD District j (= 1)

PDF_i = Plant depreciation factor (fraction of investment that may be depreciated) for processing unit type I

TR_j = Effective combined income tax rate in PAD District j

R = Investment return rate, fraction

PL = Plant life, years

when R > 0, then

$$U_{R,PL} = (R / (365 * (1- (1+R/365)^{-PL*365}))) \quad \textbf{(7)}$$

otherwise,

$$U_{R,PL} = (1 /(PL *365) \quad \textbf{(8)}$$

Also,

$$TR_j = TRF + TRS_j - TRFxTRS_j$$

where

TRF = Federal income tax rate

TRS_j = Average state income tax rate for PAD District j, weighted by crude oil charge

Note that an investment tax credit can be added to this investment model by decreasing the INV by the applicable fraction. Also, the investment return rate (R) is an input variable which may be different for build coefficient calculations ($RQBLDRAT_t$) versus investment coefficient calculations ($RQOPRRAT_t$); however, currently both are set equal to 10 percent.

The location factors (Lj) were based on analyst judgment and are presented in Table F2. In order to do this, several premises have been set in accordance with Table F1. The multipliers in Table F1 were obtained from the Gary and Handwerk reference. The variability of these factors is unknown. Examples of total Gulf Coast investment cost and depreciable investment fraction (PDF) as well as the fixed costs are presented in Table F4.

Table F1. Investment Multiplier Values

Multiplier	Value
U	0.075
O	0.15
DEF	0.03
C	0.15
S	0.04
W	0.10

Table F2. Location Index (J) and Location Factors

PAD District	Location Factor
I	1.5
II - IV	1.07
V	1.2

Refinery Unit Fixed Cost Factor

The fixed cost, in conjunction with the capital recovery threshold, forms the threshold for expansion investment decisions for processing unit type I. Fixed cost is calculated from:

$$FXOC_i = (1/365) * (B_i + PVNTPE_i \Sigma^{6}_{j=1} F_j) \quad \textbf{(9)}$$

$$FC_i = FXOC_i * L_j * E_j \quad \textbf{(10)}$$

where

B = Operations labor, $/year/BBL/day
the 6 "F" multipliers are shown in Table F3.

Table F3. Fixed Cost Multiplier Values

Multiplier	Value
1, Insurance	0.005
2, Local tax rate	0.01
3, Maintenance	0.04
4, Supplies	0.002
5, Overhead	0.015
6, Environment	0.02

The labor charge, B, and the overhead multiplier is taken from the WORLD model. The first 4 multipliers was obtained from the Gary and Handwerk reference. The environment base multiplier results from analyst judgment. The 1993 NPC study estimates the new environmental refinery operation and maintenance cost as about one third of base operations and maintenance over the 1991-1995 time period. Gary and Handwerk estimate the maintenance factor to vary between 3 percent and 8 percent, 4 percent was used because the trend is for refineries to increase operating efficiency over time. Variability of the other factors is unknown.

Table F4. Refinery Investment Recovery Estimates ($1993)

Processing Unit	Unit Index	INV (Dollars per bbl/d)	PDF	Fixed Cost ($/bbl)
Crude oil tower	ACU	$565	0.775	$0.12
Vacuum unit	VCU	$895	0.775	$0.18
Solvent deasphalting	SDA	$1,909	0.775	$0.40
Delayed coker	KRD	$4,308	0.775	$0.85
Fluid/flexicoker	KRF	$5,502	0.775	$1.10
Visbreaker/TCC	VBR	$1,390	0.775	$0.29
Naphtha hydrotreater	NDS	$1,101	0.775	$0.24
Distillate desulfurizer	DDS	$1,622	0.775	$0.35
FCC feed hydrofiner	FDS	$1,971	0.775	$0.41
Residuum desulfurizer	RDS	$4,525	0.775	$0.95
Gas oil hydrocracker	HCR	$5,463	0.775	$1.14
Residuum hydrocracker	HCV	$7,300	0.775	$1.49
Naphtha hydrotreater	HCN	$3,347	0.775	$0.69
Lube and wax units	LUB	$13,514	0.775	$2.75
Gas oil dewaxer	DEW	$1,353	0.775	$0.30
Prefrac hi density ATF	JFP	$2,168	0.775	$0.76
Hi density ATF HD unit	HDN	$14,949	0.775	$3.11
HP Semi regen REFORMER	RFH	$1,810	0.775	$0.42
LP Cyclic reformer	RFL	$2,232	0.775	$0.50
LP Continuous reformer	RFC	$2,655	0.775	$0.59
Naphtha splitter	SPL	$724	0.775	$0.15
C3/C4 dehydrogenation	OLE	$11,101	0.775	$2.34
Fluid cat cracker	FCC	$3,656	0.775	$0.75
FCC fractionation	FGS	$579	0.775	$0.11
Alkylation unit	ALK	$5,999	0.775	$1.32
Polymerization unit	CPL	$1,882	0.775	$0.42
Dimersol unit	DIM	$2,606	0.775	$0.56
Aromatics recovery	ARP	$3,041	0.775	$0.67
Pen/Hex Isomerization	PHI	$3,620	0.775	$0.98
Butane isomerization	C4I	$5,213	0.775	$1.46

Processing Unit	Unit Index	INV (Dollars per bbl/d)	PDF	Fixed Cost ($/bbl)
Total recycle isom	TRI	$5,642	0.775	$1.17
Cyclar unit	CYC	$10,736	0.775	$2.13
H2 Steam ref bfoe/d	H2P	$27,499	0.775	$5.62
H2 Partial Ox bfoe/d	H2X	$27,378	0.775	$5.40
Sulfur, S tons/day	SUL	$168	0.775	$0.04
Steam gener, lbs/hr	STG	$60	0.775	$0.01
Power generation, MKW	KWG	$7	0.775	$0.00
M Dist Furfural trting	FFX	$1,722	0.775	$0.35
Synsat Distillate Hydrotreating	SYD	$2,392	0.775	$0.48
Methanol unit	MOH	$25,681	0.775	$3.41
Merchant (?) unit	MBR	$5,289	0.775	$1.12
TAME unit	TAM	$9,787	0.775	$1.99
ETBE unit	ETB	$5,744	0.775	$1.20
ETAE unit	TAE	$10,241	0.775	$2.08
Merchant MTBE	MBM	$37,332	0.775	$7.39
C4E based oxy unit	ETH	$5,744	0.775	$1.20
C5E+ based oxy unit	ETM	$10,241	0.775	$2.08
Cryogenic C2 fractionator	ETS	$1,485	0.775	$0.33
C2E - C4E Dimerization	C24	$10,682	0.775	$2.51
Hydrogenate NC5E/NC6E	C56	$2,020	0.775	$1.30
Atmospheric Residuum Desulfurization	ARD	$29,016	0.775	$5.68
CD Tech FCC naphtha desulfurization	CDT	$860	0.775	$0.17
Low Conversion Hydrocracking	HCL	$5,026	0.775	$1.03
Hydrodesulfurizer 1	HD1	$4,811	0.775	$0.96
Hydrodesulfurizer 2	HD2	$2,448	0.775	$0.52
Alt. Hydrodesulfurizer 1	HL1	$4,811	0.775	$0.96
Alt. Hydrodesulfurizer 2	HS2	$2,448	0.775	$0.52
Mobil MAK hydrocracker for FCC feed	MAK	$5,026	0.775	$1.03
Catalytic Fluidized Bed	MOD	$2,596	0.775	$0.50
Olefin Saturation Process	MDH	$1,477	0.775	$0.29
Octgain naphtha desulfurization	OCT	$1,604	0.775	$0.32
Caustic Sox Scrubber	SOX	$2,954	0.775	$0.57

F.2 Gas Plant Models

The gas plant models for each PADD is maintained on the spreadsheet 'NGL.WK1' maintained within EIA by the Oil and Gas Division. These models require gas plant wet gas volumes as input. In order to accommodate the information available, dry gas production volume, and permit gas plant activity to be driven by dry natural gas demand, factors are applied to the dry gas demand volumes to calculate imputed volumes of processed wet gas. In PADD V, the PMM models only the California gas plants. Although Alaska produces and processes a considerable volume of natural gas, it is nearly all used for reinjection with some NGL dumped into the crude pipeline with the exception of modest volumes of southern Alaska production. The southern Alaska production does have a local NGL market with much of the dry gas shipped out as LNG. In any case, the PADD V refinery industry is virtually unaffected by Alaska NGL production. Thus, the PMM aggregate gas plant includes California only. Gas residue volumes are not available to the PMM but total dry gas volumes by PADD are. This information is in the NEMS common block region. Specifically, the total dry gas volumes are available in:

Common block: NGTDMOUT

Variable matrix: PRNG_PADD(PADD,YEAR), BCF

Description: Total dry gas produced including plant fuel and lease fuel (P&L) for PADD 'PADD' in year 'YEAR' excluding Alaska.

Each of the five values must be multiplied by a factor to obtain an estimate of the corresponding wet gas that is processed by the gas plants; i.e. the total dry gas volumes are multiplied by the ratio of processed wet gas to total dry gas production. These ratios are derived from 1993 data[4] so that data variability is unknown. The five multipliers are shown in Table F5.

Table F5. Total Dry Gas Multiplier

PADD	Multiplier
I	0.4536
II	0.7659
III	0.7572
IV	1.0136
CA	0.7745

The multiplier for PADD I is low because relatively little of the modest PADD I gas production is processed beyond field decontamination. The multiplier is high for PADD IV because a great deal of Utah gas production is reinjected for field pressure maintenance. This reinjected gas is not counted in

[4]Energy Information Administration, *Natural Gas Annual 1993*, DOE/EIA-0131(93), (Washington, DC, October 1994).

total dry gas production. The three region PMM uses the quantity weighted average multipliers of PADDs II, III, and IV for refining region two.

The NGL extraction is allowed to occur at a minimum, maximum, or average level. Complete ethane rejection is allowed with a processing credit for each barrel rejected. Propane rejection occurs in U.S. gas plants but to an unknown extent. The model allows propane rejection up to one fourth of the propane volume. The basic model structure is devised from the Pace Consultants annual petrochemical report[5]. The liquids extraction data have been calculated by averaging actual liquid extraction volumes from the 4-year period 1990 - 1993[6] as well as obtaining minimum and maximum values. The coefficients were then normalized. The minimum and maximum values seem to suggest the variability of the data but the reader is cautioned that statistically based models, as opposed to engineering models, may understate processing flexibility. Furthermore, the gas business is not statistical. It is driven by technology and economics, i.e. history is suspect as a basis for such values. The model for PADD III is shown in the following table. Models for the other PADDs are similar except the extraction coefficients differ.

Table F6. Gas Plant Model for PADD III

	MIN	MAX	AVG	ETH	PRO	Row Type	R HS
Wet Nat. Gas (MMcf)	-1.000	-1.000	-1.000			=	0
Dry Gas (MMcf)	0.945	0.938	0.941	0.002	0.002	=	+
Ethane (bbl)	14.070	14.332	14.562	-1.000			
Propane (bbl)	11.165	11.628	11.782		-1.000		
I-Butane (bbl)	4.375	4.776	4.693				
N-Butane (bbl)	2.506	4.409	3.349				
Nat. Gasoline (bbl)	7.025	7.310	7.286				
Volume Loss (MMcf)	-39.058	-42.394	-41.613	0.998	0.998		
Fuel (MMcfl)	-0.037	-0.040	-0.040				
Oper. Cost ($)	-115.07	-124.82	-122.51				
Oper. Cost ($/bbl)				5.88	2.94		
Capacity limit (bbl/d)					1.000	≤	+

Table F6 differs from the implemented model in three respects. Propane rejection is controlled via column limit in the LP rather than as a row limit and the operating costs are merely entered into the objective function row rather than occupying two rows as the above depiction suggests. Of course these

[5]Pace Petrochemical Service, *Annual Issue*, (Houston, TX, September 1989).

[6]Energy Information Administration, *Natural Gas Annual 1991*, DOE/EIA-0131(91), (Washington, DC, October 1992) and similarly, the Natural Gas Annuals for 1986-1990.

things are shown in the above form for the sake of clarity. The major difference is that the wet gas balance row is an equal zero row in the LP representation. There is a column representing pseudo purchases of wet gas with a plus one in the wet gas balance row and minus the forecasted gas price in the objective function row. Of course, this column is fixed at the determined level of processed wet gas.

F.3 Chemical Industry Demand for Methanol

Since the PMM includes methanol plant models in each PADD representing all U.S. methanol capacity, U.S. chemical industry demand (demand other than for MTBE/TAME feedstock and neat fuel) is a required input. The Pace Consultants make long range forecasts of the chemical industry demand[7]. Their forecast is:

Table F7. Chemical Industry Demand for Methanol

Year	Demand (Mbbl/d)	Year	Demand (Mbbl/d)
1990	66.7	2006	102.1
1991	73.7	2007	106.6
1992	72.0	2008	111.4
1993	73.4	2009	116.4
1994	74.8	2010	121.6
1995	76.6	2011	127.1
1996	78.0	2012	132.8
1997	79.5	2013	138.8
1998	81.0	2014	145.0
1999	82.6	2015	151.6
2000	84.2	2016	151.6
2001	85.8	2017	151.6
2002	87.6	2018	151.6
2003	89.4	2019	151.6
2004	93.4	2020	151.6
2005	97.6		

The methanol plant model in each PADD is represented by a single column activity that consumes natural gas and produces methanol. Two additional transportation activities allow the methanol produced to be transported to the refining region for production of MTBE/TAME and/or splash blending in gasoline or sent to meet the national demand for methanol by chemical plants. The model also allows capacity expansion of the methanol plant.

[7]Pace Petrochemical Service, *Annual Issue*, (Houston, TX, October 1992).

F.4 Estimation of Refinery Fixed Costs

The marginal prices computed in the PMM by the use of the Simplex algorithm cannot be used directly to determine the wholesale (refinery gate) price because they do not include the refinery source environmental costs. These are expenditures incurred to satisfy regulations related to air and water pollutants, solid waste management, and health and safety controls at refineries. Related costs are based on estimates of capital investment, one-time, and operation and maintenance expenses provided in the 1993 National Petroleum Council Study.[8]

The NPC Study provides PADD level estimates of capital, one-time, and operating and maintenance expenses over three time periods: 1991-1995, 1996-2000, and 2000-2010. Operation and maintenance expenses existing prior to 1996 are not used as they are already reflected in the fixed operating cost estimates. The NPC estimates were converted to an annual average cost per barrel of distillation capacity for inclusion as a fixed cost (Table F8) The same fixed costs are assumed for forecast years 2011 through 2020.

The underlying NPC analysis reflects the actions described below:

Air related costs include attainment of ambient air quality standards (Title I of the Clean Air Act Amendments), hazardous air pollutants (Title III, MACT standards), and permits (Title V)[9].

Water related costs include technology required to satisfy toxicity requirements of the National Pollution Discharge Elimination System (NPDES) authorized by the Clean Water Act. Costs for additional measures to protect groundwater are also assumed[10].

Solid waste related costs include remediation of contaminated soil at refineries sites, recovery and monitoring wells, the listing of additional refinery wastes as hazardous, the closing of unlined impoundments, and an assumed phased replacement of half the tanks over 40 years of age[11].

Safety and health related costs includes an assumed phase-out of hydrofluoric alkylation (hf) plants due to the classification of hf as a highly hazardous material. Small costs associated with implementing process hazards analysis are also contained in the NPC estimates[12].

[8]National Petroleum Council, *U.S. Petroleum Refining - Meeting Requirements for Cleaner Fuels and Refineries*, Volume I (Washington, DC, August 1993).

[9]Ibid, Section V-9-12.

[10]Ibid, Section V-13.

[11]Ibid, Section V-16.

[12]Ibid, Section V-19.

For comparative purposes NPC's estimated capital and OTE costs aggregated over 1991-2010 results in total environmental investments of $43.5 billion. The NPC sum can be compared to estimated investments published by the API,[13] by adding the API estimates for individual legislation that are consistent with NPC. The NPC total falls in the middle of the range of API investments of $19.4 to $60.0 billion.

Table F8. Refinery Source Environmental Costs, by PADD

(1987 $/bbl)

	PADD I	PADD II	PADD III	PADD IV	PADD V
Annual Environmental Cost	0.48	0.47	0.38	0.70	0.48

A methodology was developed to allocate refinery source environmental costs to the marginal prices of light products (LPG, gasoline, kerosene, jet fuel, No. 2 heating oil, and low sulfur highway diesel) thus including fixed costs in the prices reported at the refinery gate. The heavy products (residual oil, petrochemical feedstocks, asphalt, and other) are priced at their marginal cost due to competition from other fuels. Fixed refinery costs in each PADD are estimated as the sum of fixed operating costs, return on assets, and environmental costs associated with controlling pollution at refineries (Table F8). The PADD costs were allocated to their prospective PMM regions.

The average annual cost is applied to all years and is recovered in the prices of light products only. A variable cost methodology was investigated but was deemed to be an unnecessary complication. The costs for each 5-year time period were estimated to vary by less than 1-cent per gallon but would require significant changes to the model.

The methodology for allocating costs among light products is based on the concept of marginal cost and was developed from a similar approach used by the Gas Research Institute (GRI).[14] The steps used are as follows:

(1) Determine the environmental investment and operating costs.
(2) Determine the marginal revenue from the LP solution that will recover the marginal cost (the product of the marginal price of each product times the production volume).
(3) Calculate ratios that will apportion the environmental costs to the light products marginal prices.
(4) Use the ratios from (3) to scale the marginal prices to refinery gate wholesale prices.

[13]The sum of the initial cost estimates in *Costs to the Petroleum Industry of Major New and Future Federal Government Environmental Requirements*, American Petroleum Institute, Discussion Paper #070, (Washington, D.C., October 1991), Tables ES-1 and ES-2. Aggregate estimates exclude product specification and off-site costs, and utilized minimum RCRA reauthorization.

[14]Gas Research Institute, "U.S. Refining Model Methodology", (May 1991).

Equations: First, marginal revenue is determined for each refining region R:

$REVLP(LP)_R = VALUE(LP)_R * VOLUME(LP)_R$
$REVHP(HP)_R = VALUE(HP)_R * VOLUME(HP)_R$

where:

VALUE(LP)	is the marginal price for each light product (LP) that will bear the fixed cost
VALUE(HP)	is the marginal price for each heavy product (HP) that will not bear the fixed cost
VOLUME(LP)	is the production volume for each light product
VOLUME(HP)	is the production volume for each heavy product
$TOTREV_R$	$= REVLP(LP)_R + REVHP(HP)_R$

Second, the total revenue required, including the fixed costs, FIXCOST is determined:

$NTOTREV_R = REVLP(LP)_R + REVHP(HP)_R + (FIXCOST_R * VOLUME)$
$NLTREV_R = NTOTREV_R - REVHP(HP)_R$

The last step is to calculate the refinery gate price for each light product:

$GATEPR(LP)_R = (NLTREV_R / REVLP(LP)_R) * VALUE(LP)_R$

F.5 Estimation of Distribution Costs

Costs related to distributing petroleum products to end-users are incorporated by adding fixed transportation markups to the wholesale prices which include the variable and fixed refinery costs. Transportation markups for petroleum products except gasoline are estimated as the average annual difference between retail and wholesale prices. Due to an observable market shift before and after 1990, the transportation markups for all products except gasoline are estimated based on data between 1990 and 2000. Due to the requirement for oxygenated gasoline beginning in October of 1992, gasoline markups are estimated with data beginning in 1993. The differences are based on wholesale prices in the producing Census division and end-use prices (which do not include taxes) in the consuming Census division. Wholesale prices are aggregated from State-level prices from the EIA-782A. Sectoral end-user prices are aggregated from State-level prices from the State Energy Price and Expenditures Report (SEPER) 1997. End-use prices after 1997 are estimated according to the SEPER's methodology. Computer programs and data files used to estimate transportation markups are discussed below.

Data-Reading Programs

The following programs access EIA survey data and should be updated each year. Member names of SAS data sets are given in parenthesis.

Program: **PRJ6007.NEMS.MARKUP.WHOLSALE**

Files Read:
PRJ6007.A.SASDB.STATE.AB
(IMPSTATA)
PRJ6007.A.SASDB.STATE.AB9495
(IMP94A)
(IMP95A)
PRJ6007.A.SASDB.STATE.AB9697
(IMP96A)
(IMP97A)
PRJ6007.A.SASDB.STATE.AB9899
(IMP98A)
(IMP99A)
PRJ6007.A.SASDB.STATE.AB0001
(IMP00A)
PRJ6007.NEMS.MARKUP.SASDB
(REGIONS)
(WHOLSALE)

This program reads databases containing State-level refiner wholesale prices from the EIA-782A. The program &PRJ6007.A.IMPYY.ADF must be run prior to this step to update the database (see Appendix H-3). A separate file is read for data prior to 1994 because the 782 data was kept in a different data system. Due to the number of data, prices for each year are kept in a separate file. Wholesale prices for asphalt and road oil, diesel (low sulfur), distillate fuel, gasoline, jet fuel, kerosene, liquefied petroleum gases, and residual fuel oil are aggregated into Census division prices (cents/gallon) and output to the NEMS database for other markup programs to access. Program last run May 11, 2001.

Program: **PRJ6007.NEMS.MARKUP.EIA782.READ**

Files Read:
PRJ6007.A.SASDB.STATE.AB9495
(IMP94B)
(IMP95B)
PRJ6007.A.SASDB.STATE.AB9697
(IMP96B)
(IMP97B)
PRJ6007.A.SASDB.STATE.AB9899

(IMP98B)
(IMP99B)
PRJ6007.A.SASDB.STATE.AB9899
(IMP00B)
PRJ6007.NEMS.MARKUP.SASDB
(REGIONS)
(PRODE)

This program updates retail prices on the NEMS database using the most recent data from the EIA-782A contained in IMP94B, IMP95B, IMP96, IMP97, IMP98, IMP99, IMP00. The State-level prices for gasoline, diesel (low sulfur), distillate, kerosene, jet fuel, high and low sulfur residual fuel, and liquefied petroleum gases for the most recent 5 years, 1996, 1997, 1998, 1999, and 2000 are aggregated into regional prices and added to data for previous years (using PRODE). The program is updated by revising files and incrementing YEAR in line 82. Contact Tammy Heppner (586-4748) of the Petroleum Marketing Division of the Office of Oil and Gas concerning the EIA-782 data which is available in March. Program last run on May 21, 2001.

Program: **PRJ6007.NEMS.MARKUP.GDP87**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(GDP87CH)

This program updates the chain weighted GDP deflators (1987 base) in the NEMS database. Program should be rerun after adding most recent year's deflator.

Program: **PRJ6007.NEMS.MARKUP.CREATE.CONFAC**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(CONVER84 through CONVER98)

This program provides annual conversion factors for each product. Factors for the most recent year of the SED's should be added to the data. These conversion factors are accessed by the SEPER.READ programs. The lpg factor is updated based on the upcoming SEDS (information available in April).

Program: **PRJ6007.NEMS.MARKUP.SEPER.READ**

Files Read: SSE6356.SEPER90S.PRN
PRJ6007.NEMS.MARKUP.SASDB
(CONVER84 through CONVER97)
(SEPER97) contains data from 1990-1997

This program reads State-level prices by sector from SEPER-1997 data query provided by Roy Stanley(586-5839) of Energy Markets and End-Use. The data is provided in an Excell file, after the top line is removed and the columns are formatted with 3.0 with right alignment, they are saved as prn files and ported to the mainframe via ftp. On the mainframe these files must be resaved as card. The SEPER.READ programs puts the data into the NEMS database. Prices are accessed for the following products:

gasoline (mg)
distillate (df)
kerosene (ks)
residual fuel (rf)
liquefied petroleum gases (lg)
jet fuel (jf)
asphalt and road oil (ar)
petrochemical feedstocks (fs)
naphtha feedstocks (fn)
other feedstocks (fo)

Program: **PRJ6007.NEMS.MARKUP.READ.F759**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(REGIONS)
(E759Y85 through E759Y99, E7592000)
PRJ6944.F759.MAST2000, file has four digit year

This program reads monthly volumes of distillate, low and high sulfur residual fuel consumed by electric utilities from EIA-759. The volumes are by State and are in gallons. The program annualizes the volumes and assigns them to Census divisions. Contact Melvin Johnson (202)287-1754. Program last run May 21, 2001.

Program: **PRJ6007.NEMS.MARKUP.READ.F423**

Files Read: PRJ6501.F423.EX1984.DATA
PRJ6501.F423.EX1985.DATA
PRJ6501.F423.EX1986.DATA
PRJ6501.F423.EX1987.DATA
PRJ6501.F423.EX1988.DATA
PRJ6501.F423.EX1989.DATA
PRJ6501.F423.EX1990.DATA
PRJ6501.F423.EX1991.DATA

PRJ6501.F423.EX1992.DATA
PRJ6501.F423.EX1993.DATA
PRJ6501.F423.EX1994.DATA
PRJ6501.F423.EX1995.DATA
PRJ6501.F423.EX1996.DATA
PRJ6501.F423.EX1997.DATA
PRJ6501.F423.EX1998.DATA
PRJ6501.F423.EX1999.DATA
PRJ6501.F423.EX2000.DATA
PRJ6007.NEMS.MARKUP.SASDB
(EUSTATE)
(UTDIV)

This program reads State-level quantities and costs of distillate, and high and low sulfur residual fuel consumed by electric utilities from FERC Form 423. The quantities and costs are converted to a trillion Btu basis and assigned to Census divisions. Contact Ken McClevey (202)287-1732. Program last run in May 21, 2001. Note: the survey has fewer and fewer respondents due to deregulation.

Program: **PRJ6007.NEMS.MARKUP.SEDSVOL.READ**

Files Read: PRJ6356.SEDS99A.PRN (spreadsheet saved as space delimited text, format right aligned number 3.0)
PRJ6356.SEDS99B.PRN (spreadsheet saved as space delimited text, format right aligned number 3.0)
PRJ6007.NEMS.MARKUP.SASDB
(ARVOL)
(GASVOL)
(DSVOL)
(KERVOL)
(LPGVOL)
(RSVOL)

This program reads the files that are a subset of the most recent SEDS file, provided by EMEU staff. The program updates members of the SAS database with the most recent volumes of the individual products by sector, and state.

Markup Estimating Programs

All "CALC" programs calculate markups as the difference between a products retail and wholesale price in each Census division and sector. The estimated markups are output to PRJ6007.NEMS.MARKUP.

SASDB in 1987 dollars per trillion Btu. The output file from each program contains the variables PRODPMM, SECTOR, CENDIV, MEAN, AND STDERR.

Census Division "99" represents national data and Sector "ZZ" represents all sectors. The following table describes the source programs for petroleum product markups:

Table F9. Markups Output

Products	Sectors	Markup Program	File Name on MARKUP.SASDB
DS	CM, IN, RS, TR, ZZ	PMM.CALC	MARKPMM
JF	TR, ZZ	PMM.CALC	MARKPMM
MG	CM, IN, TR, ZZ	PMM.CALC	MARKPMM
AR	IN, ZZ	ASRO.CALC	MARKAR
KS	CM, IN, RS, ZZ	KERO.CALC	MARKOTH1
LG	CM, IN, RS, TR, ZZ	LPG.CALC	MARKLPG
RL	CM, IN	RESID.CALC	MARKRL
RH	TR	RHTR.CALC	MARKRH
DS	EU	ELECUTIL.CALC	MARKELEC
RL	EU	ELECUTIL.CALC	MARKELEC
RH	EU	ELECUTIL.CALC	MARKELEC

Note: DS TR is 85 percent low sulfur diesel, DS IN is assumed 50 percent low sulfur.

Program: **PRJ6007.NEMS.MARKUP.PMM.CALC**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(PRODE)
(WHOLSALE)
(GDP87CH)
(GASVOL)
(CONVER84 through CONVER99)
(RGAL87)
(MGBTU)
(MARKPMM)

This program uses retail gasoline, diesel (low sulfur), distillate, and jet fuel prices and volumes originating from the EIA-782A. Commercial and industrial gasoline prices are estimated by weighting the EIA-782 end-user (transportation) prices with SEDS volumes for these two sectors. Commercial and industrial gasoline volumes for 1999 are used to represent 2000, since SEDS stops with 1999 data. Distillate and jet fuel markups are estimated as the average annual difference between end-use and wholesale prices for the years 1990 forward. Jet Fuel data from 1990 , Census divisions 3, 5, and 6 are dropped because the Gulf War resulted in a price spike . Due to a break in the data trend resulting in Clean Air Act requirements, gasoline markups are estimated based on data from 1992 forward. Program last run June 5, 2001.

To update: After updating "data" programs, revise this program to utilize more recent SEDS volumes by adding another data set that reads GASVOL00. Use CONVER00 and increment all year dependent formulas.

Program: **PRJ6007.NEMS.MARKUP.ASRO.CALC**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(PRODE)
(WHOLSALE)
(SEPER89)
(SEPER97)
(PRODE)
(GDP87CH)
(CONVER84 through CONVER99)
(RGAL82)
(ARBTU)
(ARGAL)
(MARKAR)

This program uses State-level asphalt and road oil prices to the industrial sector from SEPER's through 1997. End-use prices for 1998, 1999, and 2000 are estimated using annual changes in the EIA-782 high sulfur residual fuel prices. Prices for each Census division are calculated using 1997 volumes from the SEDS. Estimates for 1998 and 1999 are weighted using 1998 volumes(should have used 1999 volumes but too complicated for little impact). Markups are estimated as the average annual difference between end-use and wholesale prices for the years 1990 forward. Census division 8 data for 1990 is dropped as an outlier. Program last run June 7, 2001 but is usually run in late April.

To update: After updating "data" programs, revise this program to utilize more recent data. Use updated conversion factors, and increment all year dependent formulas.

Program: **PRJ6007.NEMS.MARKUP.KERO.CALC**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(PRODE)
(WHOLSALE)
(SEPER89)
(STATE97)
(PRODE)
(KEROVOL)
(GDP87CH)
(CONVER84 through CONVER99)
(KEROGAL)
(KEROBTU)
(MARKOTH1)

This program uses State-level kerosene prices to the residential, commercial, and industrial sectors from SEPER's through 1997. End-use prices for 1998, 1999, and 2000 are estimated using the 1997, 1998, 1999, and 2000 changes in the EIA-782 distillate prices. Markups are estimated as the average annual difference between kerosene end-use and distillate wholesale prices for the years 1990 forward. Program last run June 7, 2001. The kerosene historical price data series was reestimated in 2000.

To update: After updating "data" programs, revise this program to utilize more recent data. Use updated conversion factors, and increment all year dependent formulas.

Program: **PRJ6007.NEMS.MARKUP.RESID.CALC**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(PRODE)
(WHOLSALE)
(SEPER89)

(SEPER97)
(GDP87CH)
(CONVER84 through CONVER99)
(RSGAL)
(RSBTU)
(MARKRL)

This program uses State-level residual fuel prices to the commercial, and industrial sectors from SEPERS through 1997. End-use prices for 1998, 1999, and 2000 are estimated using the 1997, 1998, 1999, and 2000 changes in the EIA-782 low sulfur residual fuel prices. Price estimates are weighted using 1997 volumes. Markups are estimated as the average annual difference between low sulfur end-use prices and aggregate residual fuel wholesale prices for the years 1990 forward.

To update: After updating "data" programs, revise this program to utilize more recent data. Use updated conversion factors, and increment all year dependent formulas.

Program: **PRJ6007.NEMS.MARKUP.RHTR.CALC**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(PRODE)
(WHOLSALE)
(SEPER89)
(SEPER97)
(PRODE)
(REGIONS)
(EUSTATE)
(RSVOL)
(GDP87CH)
(CONVER84 through CONVER99)
(RSTRGAL)
(RSTRBTU)
(MARKRHTR)

This program uses State-level residual fuel prices to the transportation sector from SEPERS through 1997. End-use prices for 1998, 1999, and 2000 are estimated using average residual fuel prices to electric utilities according to SEPER's methodology. The average Census division prices to utilities are multiplied by the ratio of national high sulfur price from the EIA-782 to the national residual fuel price to utilities from F-423. Prices estimates for 2000 weighted using 1999 SEDS resid transportation volumes . Markups are estimated as the average annual difference between high sulfur end-use prices and aggregate residual fuel wholesale prices for the years 1990 forward.

To update: After updating "data" programs, revise this program to utilize more recent data. Use updated volumes and conversion factors, and increment all year dependent formulas.

Program: PRJ6007.NEMS.MARKUP.LPG.CALC

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(PRODE)
(WHOLSALE)
(SEPER89)
(SEPER97)
(PRODE)
(REGIONS)
(LPGVOL)
(GDP87CH)
(CONVER84 through CONVER99)
(LPGGAL)
(LPGBTU)
(MARKLG)

This program uses State-level SEPER's prices for liquefied petroleum gases to the commercial, industrial, transportation, and residential sectors for 1989 through 1993. Post-1993 prices for all sector's are the 782 sectoral prices, because the 782 began carrying sectoral data in 1994. Markups are estimated as the average annual difference between end-user LPG prices and wholesale prices for the years 1990 forward. Program last run July 1, 2001 but is usually run in late April.

To update: After updating "data" programs, revise this program to utilize more recent data and updated conversion factors.

Program: PRJ6007.NEMS.MARKUP.ELECUTIL.CALC

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(PRODE)
(WHOLSALE)
(SEPER89)
(STATE97)
(PRODE)
(REGIONS)
(UTDIV)
(E759Y84 through E759Y99, and E7592000)
(GDP87CH)
(CONVER84 through CONVER99)

(EURFRGAL)
(EURFBTU)
(MARKELEC)

This program uses State-level SEPER's prices for distillate, high sulfur, and low sulfur residual fuel to electric utilities through 1997. End-use prices for 1998, 1999, and 2000 are estimated using SEPER's methodology. The program breaks residual fuel consumption from the EIA-759 into high and low sulfur using a high/low sulfur breakout estimated from the FERC Form 423. The estimated high and low sulfur residual fuel volumes are used to estimate weighted average prices to the utility sector.

Markups are estimated as the average annual difference between prices to utilities and wholesale prices for the years 1990 forward. The average markup for commercial high sulfur residual fuel in Census division 8 excludes 1997 as an outlier. Note: the number of EI-759 respondents has declined significantly as non-utility generators have increased. It is uncertain how this affects prices.

To update: After updating "data" programs, revise this program to utilize more recent data. Add datasets CONVER00, and E759Y01.

Program: **PRJ6007.NEMS.MARKUP.CREATE.HISTORY**

Files Read: PRJ6007.NEMS.MARKUP.SASDB
(MGBTU)
(DSBTU)
(LPGBTU)
(RSBTU)
(RSTRBTU)
(KEROBTU)
(ARBTU)
(EURFBTU)
(HISTMARK)
(HSTDOL00)
PRJ6007.NEMS.MARKUP.MU1PRDS.HIST.D062300A

This program pulls together historical markups from 1990 forward, transposes them and outputs them to a file in 1987 dollars (HISTMARK) and another file in 2000 dollars (HSTDOL00). M85 uses the gasoline markup in cents per gallon, converted to btu's. E85 uses a markup estimated from retail data from Oxyfuel News. The E85 markups are estimated in **E85.wk4**.

Program: **PRJ6007.NEMS.MARKUP.CREATE.FLATFILE.AEO2002**

Files Read: PRJ6007.NEMS.MARKUP.SASDB

(MARKPMM)
(MARKLPG)
(MARKRL)
(MARKRHTR)
(MARKOTH1)
(MARKAR)
(MARKELEC)
(HISTMARK)
(HSTDOL00)
PRJ6007.MARKUP.NEMS.MU1AEO02.D081701A
PRJ6007.NEMS.MARKUP.DOL00.AEO02.D081701A

This program pulls together the markups for each product, transposes the data and outputs it to a file used by NEMS (MU1PRDS). M85 uses the gasoline markup in cents per gallon, converted to btu's. E85 uses a markup estimated from retail data from Oxyfuel News. The E85 markups are estimated in **E85.wk4**. Additional distribution costs are added to transportation distillate markups starting in 2006 to account for ultra-low-sulfur-diesel requirements. Converts the markups into 2000 dollars and outputs to a file. Save as lrecl=120.

To update: Update the GDPCH87 deflator, GDPDEF00 and year dependent statements.

F.6 Estimation of Taxes

In the PMM, taxes are added to the prices of gasoline, transportation distillate fuel (diesel), transportation liquefied petroleum gases (LPG), and jet fuel. Taxes are also estimated for M85 (transportation methanol) and E85 (transportation ethanol). Weighted averages of the most recently-available State and Federal taxes are developed for each Census division. The State taxes are fixed in real terms; the Federal taxes decline at the rate of inflation (i.e., Federal taxes are fixed in nominal terms). An additional 2 cents per gallon is added to the State gasoline taxes to approximate local taxes. Historical values are also calculated for gasoline, transportation distillate, jet fuel and LPG, which are then added to historical end-use prices excluding taxes in order to develop a series with taxes included. The State taxes, including both historical and projected series by sector, product, and year, are contained in following file which resides in the default input directory:

MU2PRDS

The Federal taxes are read into the PMM from file:

QDCRDCF

and are updated each forecast year by deflating the current value by the rate of inflation for that forecast year.

The following section traces the development of the taxes and lists the files used to produce both historical and forecast values. The historical data are developed on a monthly basis by State, then aggregated to volume-weighted annual averages by Census division. The outputs specified are members of the following SAS database unless otherwise stated:

PRJ6007.NEMS.TAX.SASDB

Program: PRJ6007.NEMS.TAX.Dyymm

Inputs: semi-annual tax information from the Federal Highway Administration and other sources

Outputs: STyymmS
STyymmM
where yymm refers to years 84 through 99, and months 01 and 07 (except for 1990 when the months are 01 and 08)

Sources:		
	gasoline	Federal Highway Administration, Table MF-121T
	diesel	Federal Highway Administration, Table MF-121T

State-level taxes are obtained from the Federal Highway Administration on a semi-annual basis (see www.fhwa.dot.gov/ohim/novmmfr.pdf), usually reflecting taxes as of January 1 and July 1, supplemented by information from EIA's *Petroleum Marketing Monthly* and the *Clean Fuels Report*.

We assume that taxes remain at the same level for the 6 months following each semi-annual report, unless information on effective dates of tax changes is provided. Thus, the tax level reported for say, 9207, is kept at the same amount for 9208-9212. The file STyymmS contains data for 1 month, while the file STyymmM reproduces the results for the subsequent 6 months, taking into account tax changes to the extent that they are known.

Program: PRJ6007.NEMS.TAX.DLPJTyy

Inputs: State-level LPG and jet fuel taxes collected once each year

Outputs: STLPJTyy

Sources:	Jet fuel	Petroleum Marketing Division, Maureen Klein 301-495-8440
	LPG	Federal Highway Administration, Table MF-121T

The annual tax data for jet fuel and LPG begin in 1995. A snapshot of annual taxes is reproduced for each month to create files with the same data for each month.

Program: PRJ6007.NEMS.TAX.M85E85

Inputs: State-level M85 and E85 taxes collected once each year

Outputs: STM8E8yy

Sources:	M85	Clean Fuels Report, Table 9
	E85	Clean Fuels Report, Table 9

The annual tax data for M85 and E85 begin in 1997.

Program: PRJ6007.NEMS.TAX.CALC95B

(Old Version): PRJ6007.NEMS.TAX.CALC94

Inputs: PRJ6007.NEMS.TAX.SASDB
PRJ6007.NEMS.MARKUP.SASDB
PRJ6007.A.SASDB.STATE.AB
PRJ6007.A.SASDB.STATE.AB9495
PRJ6007.A.SASDB.STATE.AB9697
PRJ6007.A.SASDB.STATE.AB9899

StyymmM for years 1984 to current
IMPSTATB, a file containing State-level refiner and marketer prices for petroleum products
IMPSTATA, a file containing State-level sales of petroleum products, which are used as weights in calculating weighted averages
IMPyyB, a file containing State-level refiner and marketer prices for petroleum products for years 1994 and later
IMPyyA, a file containing State-level sales of petroleum products, which are used as weights in calculating weighted averages, for years 1994 and later
REGIONS, maps States to Census divisions
GDP87, GDP deflators for conversion to constant dollars

Outputs: STGAS_MO
STDES_MO
STLPG_MO
STJET_MO

The CALC program pulls together the monthly tax information and calculates Federal, State, and total taxes by month and State in cents per gallon and dollars per million Btu for both nominal and constant 1987 dollars. The program was revised in 1997 to incorporate state taxes for LPG and jet fuel back to 1995. Sales volumes and regional designations are added to facilitate volume weighting across time and regions as required. Refiner and marketer volumes are used to weight LPG taxes as Census division 1 and 2 data were missing in the A (refiner only) data. Product files are calculated and output separately. The programs were adjusted to incorporate changes in the EIA-782 database starting with the 1994 data.

Program: **PRJ6007.NEMS.TAX.YRREG**
(Old version): **PRJ6007.NEMS.TAX.YRREG93**

Inputs: STGAS_MO
STDES_MO
STLPG_MO
STJET_MO

Outputs: STGAS_YR
STDES_YR
CDGAS_YR
CDDES_YR
STLPG_YR
STJET_YR
CDLPG_YR
CDJET_YR

The YRREG program takes the State monthly tax files created in the CALC program and develops weighted average annual taxes by State, then averages across Census divisions and outputs annual average taxes by Census division. The Census division annual averages are used to develop historical end-use prices including taxes by adding the calculated taxes to ex-tax prices from the *Petroleum Marketing Monthly*. The program was adjusted to incorporate changes in the EIA-782 database starting with the 1994 data.

Program: **PRJ6007.NEMS.TAX.FORECAST**

Inputs: STGAS_MO
STDES_MO
STGAS_YR
STDES_YR

Outputs: GASCAST
DIESCAST

The FORECAST program uses the latest tax information, combined with the most recent year of sales data, and develops Census division weighted-average taxes which are used for the forecast. The latest tax information is in the file:

PRJ6007.NEMS.TAX.DCURR

The outputs include both nominal, constant 1987, and constant dollar denominations for the current AEO year in both cents per gallon and dollars per million Btu.

Program: PRJ6007.NEMS.TAX.FORECAST.E85
PRJ6007.NEMS.TAX.FORECAST.JET
PRJ6007.NEMS.TAX.FORECAST.LPG
PRJ6007.NEMS.TAX.FORECAST.M85

These programs perform the same function as the FORECAST program above, but for the transportation fuels indicated. The comparable data files are the following:

Inputs: PRJ6007.NEMS.TAX.SASDB:STGAS_YR
PRJ6007.NEMS.TAX.SASDB:STLPGJT00
PRJ6007.NEMS.TAX.SASDB:STM8E800,STM8E800
PRJ6007.NEMS.MARKUP.SASDB:REGIONS
PRJ6007.NEMS.A.SASDB.STATE.AB9899:IMP99A
PRJ6007.NEMS.A.SASDB.STATE.AB9899:IMP99B (use for LPG because CD's 1 AND 2 EI-782A data are missing)
(Old versions): PRJ6007.NEMS.TAX.DLPGJET.Y94
PRJ6007.NEMS.TAX.DM85E85.Y94

Outputs: PRJ6007.NEMS.TAX.SASDB:LPGCAST
PRJ6007.NEMS.TAX.SASDB:JETCAST
PRJ6007.NEMS.TAX.SASDB:E85CAST
PRJ6007.NEMS.TAX.SASDB:M85CAST

Program: PRJ6007.NEMS.TAX.CREATE.HISTORY
(contains State tax data only)

Inputs: CDGAS_YR
CDDES_YR
CDLPG_YR
CDJET_YR

Outputs: PRJ6007.NEMS.TAX.HISTFILE

This program takes historical Census division taxes and creates a file that has the same format as the input tax file of the PMM (MU2PRDS), but only for the historical years.

Program: PRJ6007.NEMS.TAX.CREATE.FLATFILE
(contains State tax data only)

Inputs: GASCAST
DIESCAST
LPGCAST
JETCAST
E85CAST
M85CAST

Outputs: PRJ6007.NEMS.TAX.TAXFILE

This program takes the Census division taxes to be used in the forecast plus the historical taxes from HISTFILE and creates a file (MU2PRDS) that is used as input to the PMM. An additional 2 cents per gallon local tax estimate (in constant 1999 dollars) is added to the State gasoline tax estimates in this program. Save file as lrecl=120.

F.7 Gasoline Specifications

The PMM models the production and distribution of three different types of gasoline: conventional, oxygenated, and reformulated. The following specifications are included in PMM to differentiate between conventional and reformulated gasoline blends: octane, oxygen content, Reid vapor pressure (RVP), benzene content, aromatic content, sulfur content, olefin content, and the percent evaporated at 200 and 300 degrees Fahrenheit (E200 and E300).

The sulfur specification for gasoline is reduced to reflect recent regulations requiring the average annual sulfur content of all gasoline used in the United States to be phased-down to 30 ppm between the years 2004 and 2007. PMM assumes that RFG has an average annual sulfur content of 135 ppm in 2000 and will meet the 30 ppm requirement in 2004. The reduction in sulfur content between now and 2004 is assumed to reflect incentives for "early reduction." The regional assumptions for phasing-down the sulfur in conventional gasoline account for less stringent sulfur requirements for small refineries and refineries in the Rocky Mountain region. The 30 ppm annual average standard is not fully realized in conventional gasoline until 2008 due to allowances for small refineries.

Starting in 1998 the specifications for conventional gasoline reflect the Environmental Protection Agency's (EPA) "1990 baseline." These specifications prevent the quality of conventional gasoline from eroding over time, which is the intent of the EPA's "antidumping" requirements.

Oxygenated gasoline, which has been required during winter in many U.S. cities since October of 1992, requires an oxygen content of 2.7 percent by weight. Oxygenated gasoline is assumed to have specifications identical to conventional gasoline with the exception of a higher oxygen requirement. Some areas that require oxygenated gasoline will also require reformulated gasoline. For the sake of simplicity, the areas of overlap are assumed to require gasoline meeting the reformulated specifications.

Reformulated gasoline has been required in many areas of the United States since January 1995. Beginning in 1998, the EPA has certified reformulated gasoline using the "Complex Model," which allows refiners to specify reformulated gasoline based on emissions reductions either from their companies' 1990 baseline or from the EPA's 1990 baseline. In 2000 the Complex Model was tightened to require further emissions reductions. The PMM uses a set of specifications that meet these "Phase II" Complex Model requirements, but it does not attempt to determine the optimal specifications that meet the Complex Model.

The State of California currently uses its own set of performance based gasoline standards which instead of the Federal Complex Model standards. The PMM assumes that all West Coast refiners must meet the current California Air Resources Board "CARBII" requirements until 2002 when a new set of "CARBIII" requirements will take their place. The CARBIII standards reflect the removal of the oxygen requirement designed to compliment the State's plans to ban the oxygenate, methyl tertiary butyl ether (MTBE) by the end of 2002. Because MTBE is currently the main source of oxygen for gasoline in the State, California has petitioned the EPA for a waiver to the Federal RFG oxygen requirement. As EPA denied the waiver request, PMM reflects no oxygen waiver in the areas of California covered by the Federal reformulated gasoline program: Los Angeles, San Diego, Sacramento, and the recently added, San Joaquin Valley. In effect, these areas must use ethanol to meet the oxygen requirement. Other areas of California, do not have an oxygen requirement but use oxygenates because of their octane boosting, and volume extending properties. RFG in the areas with the Federal oxygen requirement is classified in the PMM as "RFG," while CARB gasoline in other areas is classified as "RFH."

AEO2001 reflects legislation which bans or limits the use of MTBE in twelve additional States: Arizona, Colorado, Connecticut, Illinois, Iowa, Kansas, Michigan, Minnesota, Nebraska, New York, South Dakota, and Washington.[15] Since the oxygen requirement on RFG is assumed to continue in these States, the MTBE ban is modeled as a requirement to produce ethanol blended gasoline in the CHGDMDS subroutine.

Arizona also has a reformulated gasoline program for the Phoenix area which is mandated by state law. Phoenix had previously been part of the Federal RFG program but opted out when State requirements were adopted. Phoenix is required to use CARB in the winter but may use either CARB or Federal RFG in the summer. Arizona is in a different model region than California and, for the sake of simplicity, is assumed to use RFG meeting Federal specifications.

Annual Average RVP Methodology

The annual average RVP limits are derived based on existing summertime requirements and estimated wintertime levels. The assumed summer and winter RVP specifications had been annualized by simple averaging using summer and winter weights provided by the EPA.[16] However the cost of reducing RVP may not be a linear function. Therefore, utilizing the Petroleum Market Model, a methodology based on marginal costs was developed to compute an RVP specification to represent the annual average. The PMM was configured to run two separate cases for summer and winter using the appropriate product specifications and demands for each season. The summer and winter marginal costs for gasoline obtained from the two runs were averaged (weighted by demand) to produce a target annual cost. The PMM (re-configured for annual demands) was then run iteratively, varying RVP until the model produced marginal gasoline costs that were significantly similar to the target cost. The resulting RVP specifications were then reduced by 0.3 psi to reflect pipeline requirements which are tighter than the EPA limits. The RVP specifications used in the PMM are shown in Table F10. The lower RVP specifications in PADD V reflect more stringent California limits which are imposed statewide.

Table F10. Estimated Annual Reid Vapor Pressure

Gasoline Market/Type	Simple Average		Annual RVP in PMM		
	PADD I-IV	PADD V	PADD I	PADD II-IV	PADD V
Conventional	10.2	9.2	9.7*	9.8*	9.2
Oxygenated	10.2	9.2	11.0	11.0	9.2
Reformulated	9.1	8.7	8.6	8.6	7.9

*Originally 10.0, adjusted to reflect "boutique fuels" with lower RVP requirements in some areas.

[15]Maine has passed legislation that provides a "goal" of phasing-out MTBE. Since the legislation is not binding, Maine is not included in AEO2002 assumptions.

[16]The summer weight of 0.396 and winter weight of 0.604 were provided by Dave Korotney of EPA (313-668-5507).

Source: Unpublished EPA data, July 2001.

Complex Model Standards for Motor Gasoline

The Environmental Protection Agency (EPA) has promulgated new regulations for reformulated motor gasoline that are designed to lower vehicle emission pollutants as required by the amended Clean Air Act of 1990.[17] The reformulated gasolines are designed to reduce vehicle emissions of toxic and ozone-forming compounds. Reformulated gasoline must be sold in certain regions where there are severe ozone problems as well as in areas with less severe ozone problems which opt into the program. Conventional gasoline may be sold elsewhere but it must not be more polluting than it was in 1990. These areas are discussed elsewhere in the documentation. Although the EPA has established some conventionally treated specifications, namely minimum oxygen content and maximum benzene content, the new conceptual aspect of the emission standards is that the reformulated gasoline must be blended in such a way that it meets maximum allowable emissions of volatile organic compounds (VOCs), nitrous oxides (NO_x), and toxics. These new motor gasoline standards are calculated by complex formulae based upon key properties of the gasoline blend. The regulations cover Phase I (1 January 1995 through 31 December 1999) and Phase II (1 January 2000 and indefinitely afterwards). The model uses a set of specifications that meet emissions requirements for Phase II of the Complex. Also, the refiner may meet the requirements for VOCs and NO_x on either a per gallon basis or on an average basis although some per gallon constraints still apply. The average basis has been incorporated into the model.

[17]Federal Register, Environmental Protection Agency, Regulation of Fuels and Fuel Additives;Standards for Reformulated and Conventional Gasoline; Final Rule, Part II, 40 CFR Part 80, (Washington, DC, 16 February 1994)

The NO_x and toxics emission standards for reformulated gasoline apply year-round whereas the VOCs standards apply only in the summer. The NO_x standard varies depending upon whether the VOCs standards apply, i.e. depending upon whether it is summer or winter. The VOCs standard for the north[18] is different from the VOCs standard applying to the south, greater volatility is allowed in the north. The Complex Model Averaged Standards are shown below in Table F11.

Table F11. Complex Model Standards

	Phase I 1995 - 1999	Phase II 2000+
VOC Reduction, %		
South	≥ 36.6	≥ 29.0
North	≥ 17.1	≥ 27.4
NO_x Reduction, %		
Summer	≥ 1.5	≥ 6.8
Winter	≥ 1.5	≥ 1.5
Toxics Reduction, %	≥ 16.5	≥ 21.5
Oxygen, wt%	≥ 2.1	≥ 2.1
Benzene, %	≤ 0.95	≤ 0.95

These standards were translated into conventionally configured specifications for blending motor gasoline. First, two winter specifications were developed, one for Phase I and one for Phase II. Of course, the VOCs standard was excluded from consideration. Then four summer specifications were created, a south set and a north set for Phase I and similarly for Phase II. Specifications for Phase I were used in prior forecasts for years up to 1999. PMM currently uses only the Phase II specifications as 2001 is the initial forecast year. The sulfur specification is adjusted to reflect the regulations requiring the reduction of sulfur in gasoline. RFG is assumed to reach the target of 30 ppm sulfur by 2004.

[18]For the sake of simplicity, we use the terms south and north to refer to EPA regions 1 and 2 respectively. Region 1 is covered by ASTM Class B while Region 2 is covered by Class C.

These sets were developed by use of a spreadsheet, developed by EPA, which calculates the VOCs, NO_x, and Toxics of a reformulated gasoline as a function of the 'conventional' properties of the gasoline, i.e. as a function of RVP, sulfur content, oxygen content, aromatics content, olefins content, benzene content, percent evaporation at 200 degrees Fahrenheit (E200), and percent evaporated at 300 degrees Fahrenheit (E300). The approach was to start with 'best informed guess' properties and use trial and error to gradually expand the allowable property limits. The blend properties cited as typical fuels in an EPA presentation[19] served as the starting values for both Phases I and II. The same starting point was used for both winter and summer. Table F12, following a chart developed by the EPA,[20] indicates the directional sensitivities of the properties on the standards. Of course, a more rigorous approach is possible in establishing the specification sets. For instance, one might perform incremental changes over the reformulated gasoline properties followed by computer runs to establish minimum cost specifications. However, this approach was not implemented due to resource constraints.

Table F12. Directional Emission Effects of Gasoline Property Changes

Property	VOC	NO_x	Air Toxics
RVP ↓	↓↓↓	–	↓
Sulfur ↓	↓	↓↓↓	↓↓
Aromatics ↓	↓	↓	↓↓
Olefins ↓	–	↓	–
E200 ↑	↓	↑	↓
E300 ↑	↓	–	–
Oxygen ↑	–	–	↓↓
Benzene ↓	–	–	↓↓↓

The PMM is an annual model, i.e. it does not have seasonality. A decision was made to develop, for PADDs I-IV, a single reformulated gasoline specification for Phase I simulation and a single specification for Phase II. This required several actions. The two summer sets for Phase I were linearly blended by compositing the projected gasoline sales-weighted south specifications to the appropriately weighted specifications of the north. The resulting two sets of specifications for Phase I, one for summer and one for winter, were then composited after weighting them according to summer sales and winter sales respectively. The Phase II specifications were collapsed to a single set in the same manner. The

[19]C.L. Gray, "Reformulated Gasoline Final Rulemaking and Renewable Oxygenate Proposal," Proceedings of The World Conference on Refinery Processing and Reformulated Gasoline, March 22-24, 1994, Information Resources, Inc.
[20]Ibid.

composites were calculated in a spreadsheet maintained by the Oil and Gas Division. This specification was adapted from the presentation made by Charles L. Gray at the conference cited above. The resulting reformulated gasoline specifications are shown in Table F13. It is, of course, a simple matter to convert the PMM blending stock distillation temperature values as needed.

Table F13. PMM Reformulated Gasoline Specifications

	Phase I PADDs I-IV	Phase II PADDs I-IV	Phase II with Reduced Sulfur PADDs I-IV
Max RVP, psia	8.7	8.6	8.6
Max S, ppm	305	108.75	30
Max Aro, %	25.0	25.0	25.0
Max Ole, %	12.0	12.0	12.0
Min E200, %	49.0	49.0	49.0
Min E300, %	87.0	87.0	87.0
Min Oxy, wt%	2.1	2.0	2.0
Max Ben, %	0.95	0.66	0.66

Data represent 2001 specifications to meet Complex Model standards. PMM adopted specifications in each forecast year based on the regulations in effect at the time. Therefore, in 2004 when the Tier 2 regulation kicks in, the maximum sulfur content is reduced to 30 ppm. The sulfur specification on the Complex Model is adjusted to meet Tier 2 gasoline requirements.

F.8 Estimation of Gasoline Market Shares

Within the PMM, total gasoline demand is disaggregated into demand for conventional, oxygenated, and reformulated gasolines by applying assumptions about the annual market shares for each type. Annual assumptions for each region account for the seasonal and city-by-city nature of the regulations. The market shares are assumed to remain constant at the actual 2000 level.

In 2003, the Census Division 9 market share for RFG is separated into two different catagories which represent CARB3 gasoline without an oxygen requirement (RFH) and CARB3 gasoline with the Federal oxygen requirement (RFG). This breakout into another product is needed to represent the planned MTBE ban in California in the absence of a waiver to the Federal RFG oxygen requirement. As EPA denied the State's request for a waiver, PMM assumes the Federal oxygen requirement will remain intact in the three areas of California bound by the Federal requirement; Los Angeles, Sacramento, San Diego, and the recently added, San Joaquin Valley. In effect, these areas must use ethanol to meet the oxygen requirement. The market shares assume that 59 percent of the gasoline in Census Division 9 will continue to meet the Federal 2.0 percent oxygen requirement, and 15 percent will meet California specifications with no oxygen requirement.

Although the shares are assumed to remain constant after 2003, the PMM structure allows for them to change over time based on alternative assumptions about the market penetration of new fuels. This allows for flexibility to analyze the impact of differing market share assumptions and to adjust the assumptions over time based on updated information about announced participation in the oxygenated and reformulated gasoline programs.

Limitation on MTBE Blended into Gasoline

MTBE is a gasoline blending component used primarily to meet the oxygen requirement of reformulated gasoline specified by the Clean Air Act Amendments of 1990. In the past few years, the use of MTBE has become a source of debate, because it has made its way from leaking pipelines and storage tanks into water supplies. Legislation to ban/limit the use of MTBE in California and twelve other States is modeled as a requirement to produce ethanol blended gasoline in the CHGDMDS subroutine. Ethanol blends are assumed to account for the following market percentages:

29.0 percent of RFG in Census Division 1
36.5 percent of RFG in Census Division 2
97.7 percent of RFG in Census Division 8
100.0 percent of RFG (with 2.0 percent oxygen requirement) in Census Division 9
100.0 percent of oxygenated gasoline in Census Division 4
100.0 percent of oxygenated gasoline in Census Division 8
100.0 percent of oxygenated gasoline in Census Division 9

Concerns for water quality have also prompted a number of congressional proposals that would ban MTBE and similar ethers in all States. The PMM can model future MTBE limits or bans in gasoline produced at the U.S. refineries. The MTBE limits are defined for conventional and reformulated gasolines (separately) on a national level, and can be activated in any forecast year. In the qdcrdcf.txt input file, the user can define the allowed volume percent of MTBE (combined with other undesirable ethers) for either or both conventional and reformulated gasolines, and the year the restrictions will go into effect. The constraints are defined mathematically below.

For reformulated gasoline (RFG):

$$\Sigma_{ethers}\, B(r)RFG(ethers) \;<=\; pct * Q(r)RFG \;+\; pct * Q(r)RFH \;+\; pct * Q(r)SSR$$

and for conventional gasoline (TRG):

$$\Sigma_{ethers}\, B(r)TRG(ethers) \;<=\; pct * Q(r)TRG \;+\; pct * Q(r)TRH \;+\; pct * Q(r)SST \;+\; pct * Q(r)SSE$$

where ethers include all or some of the following:

MTBE, ETBE, TAME, TAEE, THME, THEE

F.9 Diesel Specifications

PMM models three types of distillate fuel oil: heating oil (N2H), low-sulfur diesel (DSL), and ultra-low-sulfur-diesel (DSU). Both types of diesel fuel reflect specifications for sulfur, aromatics content, and API gravity. DSL reflects current highway diesel fuel requirements while DSU reflects the tighter "ultra-low-sulfur-diesel"(ULSD) requirement that will begin to be phased-in in 2006. DSL in Census Divisions 1 through 8 is assumed to meet Federal specifications including a maximum sulfur content of 500 parts per million (ppm) and a maximum aromatic content of 35 percent by volume.[21] DSL in Census Division 9 is assumed to meet California Air Resources Board (CARB) standards that limit sulfur content to 500 ppm and aromatics to 10 percent by volume.[22]

According to the "ultra-low-sulfur diesel"(ULSD) regulation finalized in December 2000, ULSD is highway diesel that contains no more than 15 ppm sulfur at the pump. In PMM this new product is assumed to contain 7 ppm sulfur at the refinery gate, reflecting the general consensus that refiners will need to produce diesel with a sulfur content below 10 ppm to allow for contamination during the distribution process.

F.10 Estimation of Diesel Market Shares

[21] Federal regulations require either a maximum 35 percent (volume) aromatics or a cetane index of 40.

[22] http://arbis.arb.ca.gov/diesel/diesregs.pdf

Demand for highway-grade diesel, both 500 ppm and ULSD combined, is assumed to be equivalent to total transportation distillate demand. Historically, highway-grade diesel supplied has nearly matched total transportation distillate sales, although some highway-grade diesel has gone to non-transportation uses such as agriculture and construction.

The ULSD regulation includes a phase-in period under the "80/20" rule, that requires the production of 80 percent ULSD and 20 percent 500 ppm highway diesel between June 2006 and June 2010, and a 100 percent requirement for ULSD thereafter. The phase-in path for ULSD is determined in the input file QDCRDCF.TXT. As NEMS is an annual average model, only a portion of the production of highway diesel in 2006 is subject to the 80/20 rule and the 100 percent requirement does not cover all highway diesel until 2011.

F.11 Estimation of Regional Conversion Coefficients

Differing regional definitions necessitate the conversions of certain variables from one regional structure to another. Regional conversions are not extensive in the PMM, but are needed for three refinery input prices, refinery fuel consumption, and cogeneration information. The factors are used to convert prices consumption, or cogeneration from other regions to the PADD level since the PMM was originally constructed by PADD. The PADD level information is applied to the current PMM regional configuration where PADD I is equivalent to PMM Region 1, PADD II, III, and IV are included in PMM Region 2, and PADD V is equivalent to PMM Region 3.

Conversions for Prices of Refinery Inputs

PMM receives prices for refinery inputs of natural gas in other regional configurations and must convert these into PADD level prices. Due to the proximity of refineries in PADDs II, III, and IV to the sources of natural gas supply, prices in these PADD's reflect wellhead natural gas prices in the corresponding Oil and Gas Production Regions. PADD's I and V use industrial prices in the corresponding Census divisions in order to capture the additional costs of moving the natural gas to the refineries. Table F14 shows the source of PADD level natural gas prices:

Table F14. Source of PMM Natural Gas Prices

Correlation of Prices

PADD	Input Price
I	Census Division 2 industrial price (PGIIN)
II – IV	Oil and Gas Production Region 3 and 5 wellhead prices (OGWPRNG), and quantity weighted average of Oil and Gas Production Regions 2, 4, & 8 wellhead price (OGWPRNG)
V	Census Division 9 industrial price (PGIIN)

PMM receives prices for refinery inputs of electricity by Census division. PADD level prices are derived by assuming prices in intersecting Census divisions. Table F15 shows the correlation between PADD and Census division electricity input prices:

Table F15. Source of PMM Electricity Prices

Correlation of Prices	
PADD	**Input Price**
I	Census Division 2 industrial prices (PELIN)
II - IV	Census Division 3, 7, and 8 industrial prices (PELIN)
V	Census Division 9 industrial prices (PELIN)

Conversions for Refinery Fuel Consumption

Refinery fuel consumption must be converted from the PADD to the Census division level. Each Census division consumption number will equal the consumption in the overlapping PADD's times a factor. The factors were developed using State-level refinery operating capacity and are shown in Table F16. The factors are interpreted as follows: The 0.8434 at the intersection of Census Division 2 and PADD I indicates that 84.3 percent of the PADD I refinery fuel consumption is estimated (using refinery operating capacity as estimator) to occur in Census Division 2. These values will change by small amounts as refinery capacities change, but the impact on model results will be small.

Example: Census Division 7 fuel consumption =

(PADD II consumption * .12) + (PADD III consumption * .93)

Table F16. PADD to Census Division Conversion Factors

	Census Division									
	1	2	3	4	5	6	7	8	9	SUM
PADD 1		0.84			0.16					1.00
PADD 2			0.60	0.19		0.09	0.12			1.00
PADD 3						0.07	0.93			1.00
PADD 4								1.00		1.00
PADD 5									1.00	1.00

Conversions for Cogeneration

Information including cogeneration levels (RFCGGEN(CD)), cogeneration capacity (RFCGCAP(CD)), refinery fuel consumption (RFCGFUEL(CD)), self-generation (RFCGSELF(CD)), and generation for grid (RFCGGRID(CD)) must also be converted from PADD level to Census divisions. The same factors and methodology developed for refinery fuel consumption (Table F16) are used to convert the cogeneration data.

Conversion coefficients for refinery fuel consumption and cogeneration information are estimated using &6007PRJ.PMM.CAPACITY.COEFS. Manual updates to the data file were made using data from Oil and Gas Journal survey and making some minor adjustments to the program that reads the data.

F.12 Unfinished Oil Imports Methodology

PADD's I and III are the primary recipients of unfinished oil imports into the United States. Of the four categories that EIA publishes, light gas oils are practically nil and will not be modeled in PMM.

The Oil Market Module used the most recent known values for imports of unfinished oils, which remained fixed throughout the forecast period. While simplistic, this methodology was difficult to improve on.

In the methodology described below, unfinished oil imports are estimated as a function of crude oil input to refineries. Only six observations were available for this equation (1989 was an outlier and was not used), so this methodology should be reviewed later, possibly re-estimating the equation using quarterly data. Total U.S. unfinished oil imports are estimated from the equation, then the PADD's I and III values are shared out.

U.S. Unfinished Oil Imports = -2856.7 + (0.2447 * Crude Inputs)
t-stats: (-157.5) (8.97)
R-squared: .95

PADD III Unfinished Oil Imports = U.S. Unfinished Oil Imports * 0.61
PADD III Naphthas = PADD III Total * 0.24
PADD III Heavy Gas Oils = PADD III Total * 0.33
PADD III Residuum = PADD III Total * 0.43

PADD I Unfinished Oil Imports = U.S. Unfinished Oil Imports * 0.38
PADD I Naphthas = PADD I Total * 0.12
PADD I Heavy Gas Oils = PADD I Total * 0.68
PADD I Residuum = PADD I Total * 0.20

F.13 Product Pipeline Capacities and Tariffs

Two sources were used to obtain the product pipeline data; (1) The NPC study [23] and (2) The North American Crude Oil Distribution (NACOD) model prepared by ICF for the Office of Strategic Petroleum Reserves (OSPR) during 1990-91. The NACOD data was received in LOTUS.WK3 spreadsheet format from Don Buck of OSPR.

NACOD data for the year 2000 were used for the petroleum product pipeline capacities and tariffs (1991$). The NPC study was used for LPG and NGL pipeline capacity data. The NACOD model defines 15 crude oil demand regions (including Canada and Puerto Rico/Virgin Islands) and the NPC study uses

[23]National Petroleum Council, *Petroleum Storage and Distribution, Volume 5, Petroleum Liquids Transportation,* (April 1989).

PADD regions. The links needed for PMM, as shown in Table F17, are based on PADDs for refining regions and Census divisions for demands.

Many of the arcs shown in Table F18 and Table F19 represent more than one pipeline. In some cases, we have retained more than one arc from a source to a destination in order to have a better representation of product movements.

Table F17. NACOD Regions and NEMS/PMM Census Regions

NACOD Regions		NEMS/PMM Regions	
Code	Locations	Code	Locations
1	New England	1	NE, New England
2	Includes MD,DE	2	MA, excludes MD,DE
3	WV to FL	5	SA, includes MD,DE
4	KS, OK	7	WSC, includes OK,KS
5	PADD II	3, 4	WNC,ENC, and KY, TN from 6
6	Texas Gulf Coast	7	WSC
7	LA Gulf Coast	7	WSC
8	West Texas, NM	7	WSC, excludes NM
9	AR, No. LA, No. MS, AL	6, 7	ESC,AR,LA,MS,AL
10	PADD IV, North- ID, MT	8	MNT
11	PADD IV, South- WY,UT,CO	8	MNT
12	Alaska		
13	Hawaii		
14	PADD V,	9	PAC, excludes NV,AZ

The product pipeline capacities, excluding LPG/NGL service, are shown in Table F19. The matrix formulation used in PMM allows for separate arcs for product movements. For example, to deliver a barrel of gasoline to Dorsey, Maryland (in Census Region 2) from PADD III, (Census Region 7), requires flow on the arc from Region 7 to Region 6 (capacity of 2280 Mbbl/cd) at a cost of $0.31/bbl, flow on the arc from Region 6 to Region 5 (capacity of 2526 Mbbl/cd) at a cost of $0.74/bbl, and flow on the arc from

Region 5 to Region 2 (capacity of 1392 Mbbl/cd) at a cost of $0.16/bbl. The total tariff is $1.21/bbl or 2.88 cents/gallon.

Table F18. Petroleum Product Pipeline Capacities and Tariffs[24]

Census region			
From	To	Capacity (Mbbl/cd)	Rate (Wt. avg $/bbl)
ENC, 3	MA, 2	157	1.32
ENC, 3	SA, 5	20	1.40
SA, 5	MA, 2	1392	0.16
ESC, 6	SA, 5	2526	0.74
WSC, 7	ENC, 3	328	0.56
WSC, 7	WNC, 4	280	0.86
WSC, 7	WNC, 4	717	0.80
WSC, 7	ESC, 6	2280	0.31
WSC, 7	MNT, 8	81	0.74
WSC, 7	MNT, 8	58	0.73
MNT, 8	WNC, 4	44	0.99
MNT, 8	PAC, 9	73	0.99

[24]Capacities and tariffs from NACOD model. Tariffs shown reflect 1987 dollars so that it's consistent throughout the PMM.

The LPG/NGL pipelines are shown in Table F19.

Table F19. LPG/NGL Pipelines Capacities and Tariffs[25]

Census Region		Capacity (Mbbl/cd)	Rate (Wt. avg $/bbl)
From	To		
ENC, 3	MA, 2	61	2.18
WNC, 4	ENC, 3	56	0.99 (estimated)
WNC, 4	SA, 5	57	0.99 (estimated)
ESC, 6	SA, 5	109	0.65
WSC, 7	ESC, 6	120	0.28
WSC, 7	WNC, 4	225	0.65
WSC, 7	WNC, 4	65	1.14
WSC, 7	MNT, 8	47	0.84
MNT, 8	WNC, 4	12	1.15

F.14 Cogeneration Methodology

Electricity consumption in the refinery is a function of the throughput of each unit. Sources of electricity consist of refinery power generation, utility purchases, refinery cogeneration, and merchant cogeneration. Power generators and cogenerators are modeled in the PMM Linear Program (LP) as separate units which are allowed to compete along with purchased electricity.

Refinery Cogeneration

The refinery cogeneration unit in the PMM LP was modeled using historical data as a guideline. Cogeneration activity for each refinery was aggregated to the PADD level for incorporation into the PMM LP. Cogeneration capacity, fuel consumption, and percent sales to the utility grid were estimated from the EIA-860B, Annual Nonutility Power Producer Report for 1998. The data covers all of SIC 29, not just SIC 2911. Cogeneration investment and operating costs were derived from the 1980 Office of Technology Assessment (OTA) report "Industrial Cogeneration."
Cogeneration capacity (including planned capacity) for each refining region was derived from the EIA-

[25]Capacities from NPC study, tariffs from NACOD model data

860B historical data base. It should be noted that the capacity provided in the data base is summertime capability and not nameplate capacity. The LP limits utilization to 90 percent of the summertime capability. Cogeneration capacity is allowed to expand when the value received from the additional product exceeds the investment and operating costs of the new unit. The value of adding capacity includes revenues from sales to the utility grid and the displacement of purchases of electricity. Investment costs are derived from the OTA report. The capacity expansion methodology is described in detail in Chapter 4.

Refinery cogeneration fuel consumption was derived from the NEMS Industrial Model for small cogeneration systems. A 1,000 kw capacity unit was assumed with an overall heat rate of 14,217 btu/kwh. Converted to fuel oil equivalent, consumption of 2.26 barrels of fuel oil produces approximately 1,000 kwh of electricity and 6,530 lbs of steam. Since the LP refinery consumes fuel in barrels of fuel oil equivalent, shares of individual fuels were determined from the historical data and computed post process. The shares are allocated as follows:

Oil	6.0%
Natural Gas	68.1%
Other Gaseous Fuels	25.2%
Other	0.7%.

In the past, shares of all petroleum based fuels were aggregated under Petroleum Products. This category has now been divided into Oil and Other Gaseous Fuels.

In general, refinery cogeneration units tend to be small, designed to supply the refinery's steam and electricity needs, with a small amount of leftover capacity sold to the grid. However, if it is profitable to sell cogeneration electricity, the LP will sell all of it. Likewise if it is not profitable, it will sell none of it. To model the situation more realistically, sales to the grid were modeled using percentages derived from the historical data base. The percentage of sales to the grid for each refining region (PADD) was calculated from the 1998 data as follows:

REGION	PERCENT SOLD TO GRID
1 (PADD I)	56.9
2 (PADD's II, III, and IV)	4.3
3 (PADD V)	20.1

The LP is forced to sell electricity back to the grid in these percentages at a price equal to the average price of electricity.

Fixed operating costs are calculated in the model as a function of cogeneration capacity while variable

operating costs are determined as a function of electricity generated. The following rates were determined from the OTA report.

Annual Fixed Cost $7.32/kw
Variable Cost $0.00565/kwh

Data from the EIA-860B report obtained from an Access query by Alan Beamon. The resulting data were manipulated in COGENALL.XLS and saved into COGEN.HISTORY.PRN which is read by the SAS program &PRJ6007.PMM.HIST.COGEN00. This program manipulates the data to get it into the format need by the LP. Note: cogenaration capacity by PADD from the last historical year is multiplied by 24 for input into the PMM file COGENER.DAT.

Merchant Cogeneration

Merchant cogeneration is also modeled in the PMM. Merchant cogenerator's are defined as non-refiner owned facilities located near refineries to provide energy to the open market and to the neighboring refinery. The PMM merchant cogeneration model parameters are based on the Central & South West Energy Inc. (CSWE) facility located adjacent to the Phillips Petroleum Company in Sweeny, Texas. CSWE supplies all of the refinery's steam and electricity requirements and receives up to three quarters of their fuel from refinery waste gases.[26] Electricity not used by Phillips, about two-thirds of total capacity, is sold on the open market.

Fuel consumption parameters for the PMM merchant plant are based on the Sweeny facility. The PMM merchant cogeneration unit consumes 1.90 barrels of fuel oil equivalent to produce 1,000 kwh of electricity and 5,200 lbs. of steam.

Initial capacity in PMM region 2 (PADD's II, III, and IV) is 330 Mw (Sweeny plant). Base capacity in all other regions is zero. Capacity expansion methodology is the same as in the refinery cogeneration model. Investment cost for a new cogeneration facility is $580 per kw of capacity. Annual fixed cost and variable operating cost are the same as for the refinery cogeneration model.

Unlike refinery cogeneration units, merchant facilities tend to be large units designed to sell a large portion of their electricity to the grid. The PMM merchant cogeneration model assumes 67 percent of electricity generated is sold to the grid in all regions, based on the Sweeny facility. The sale price is equal to the average of the generation price and the industrial price of electricity for each PMM region. Electricity prices are obtained from the Electricity Market Model.

[26]CarolAnn Giovando June 1998. *1998 Powerplant Awards Sweeny Cogeneration Facility*, Power.

F.15 Natural Gas Plant Fuel Consumption

The consumption of natural gas by natural gas processing plants is modeled as a function of dry gas production. Natural gas consumed at gas processing plants is calculated as a percentage of dry gas production using data from the *Natural Gas Annual 1992*. The ratios are calculated by PADD, except for PADD V where Alaska is computed separately from the rest of PADD V.

PADD I	1.36
PADD II	2.50
PADD III	2.43
PADD IV	2.61
PADD V	2.25
ALASKA	8.93

F.16 Crude Oil Exports/Total and Alaskan

Exports of crude oil have been historically linked to the level of domestic production. Crude oil exports are represented in the PMM by the following equation:

Exports= -179.23 + .043 (Totprod)
(-5.03) (6.14)

R squared = 0.60

where Totprod = total United States crude oil production in thousands of barrels per day.

In November 1995, the ban on exports of Alaskan North Slope (ANS) oil was lifted. Exports began in the spring of 1996. Since that time ANS exports have represented approximately 60 percent of total crude oil exports from the United States. Therefore the PMM methodology assumes that 60 percent of crude oil exports are from ANS.

F.17 Technology Improvement Option

A number of mechanisms for representing technological progress for key PMM refinery processing units has been implemented in the PMM. The first option allows the PMM to represent process technology improvements which will impact operating costs on any or all active processing units and/or processing modes. Thus, the user defines the processing unit(s), corresponding processing mode(s), and percentage change in variable operating cost (OVC) (positive or negative), along with a range of years over which these are phased in. The second option allows the user to define a set of processing units and corresponding output streams whose yields would change due to technology improvements. The year in

which the technology will come on and the corresponding yields are also included in the input data. The third option allows the user to upgrade the properties of intermediate streams beginning in any user-specified year. The user defines the stream ID, the spec ID(s), and the new spec value(s), along with the activation year for this change. The design for each of these options is modular in that the control data are located in a file separate from the current PMM refinery technology database, and the user defines the information needed to drive the technology change. Each of these options are summarized below.

For the first option, the user has the option to change the OVC data for all processing units (global), for any number of user-defined processing units, for both (with the user-defined unit data over riding the global data), for any set of processing modes, or for nothing at all. The user defines the period over which the OVC change is phased in, as well as the total percentage change (- for a decrease and + for an increase) that is desired over the period. For the user-specified option, the user also includes the 3-digit processing unit ID(s). The last record of data in the user-specified option must begin with a # symbol to signal the end of the list. To turn off either or both options, the phase-in begin and end years must be set to 0. The control data are located at the bottom of the PMM QDCRDCF data file. The format of this control data is as follows:

```
C Data for Global changes to reflect Technological Progress
C      |  Phase-in Period  | Chng over period
C      | BeginYR    EndYR  | Percent Chng
@      |      Y1        Y2  | TPCT_CHNG
               0         0      0.0000

C Data for Process Unit changes to reflect Technological Progress
C      | Phase-in Period   | Chng over period  | Name of ProcUnit/ Mode
C      | BeginYR    EndYR  | Percent Chng      | Three-letter ID
@      |      Y1        Y2  | TPCT_CHNG         | UNAMID      MNAMID
x              0         0      0.0000             FCC  75H
x              0         0      0.0000             ALK         C4A
#              0         0      0.0          ---         ---
```

To activate the option to *change stream yields* for a processing unit and mode of operation, the user is required to define the processing unit(s), mode(s) and stream(s) being affected, the corresponding new yield level(s), and an activation year (not phased in for this version). The number of processing units, modes, and streams must be included in the data file to act as controls for reading and processing the data. Up to ten modes and ten streams per mode can be changed for each processing unit defined. To deactivate this option, the number of processing units is set to zero. During the processing effort, the original yield and gain levels associated with the processing unit/mode combination are first retrieved from the LP matrix. Next, the yields corresponding to streams specified by the control data are updated based on the input data. Finally, the gain is recalculated and updated in the matrix. The data format representing an activated list of new yield levels is presented below:

```
C Data for Yield improvement to reflect Technological Progress
C    | Num/Name |      |       |    MAX MODES = 10
C    | ProcUnit | Tech | # of  |      1  new |      2  new |      3  new |      4  new
```

```
C   | 3-let ID | Year | modes | mode, coeff | mode, coeff | mode, coeff | mode, coeff|
@   | -------- | ---- | ----- | ----  ----- | ----  ----- | ----  ----- | ----  -----|
        1
       FCC       2005     2      80S           75H
                                  4             8
                                 RC8   .635    UC4   .064
                                 LC8   .099    UC3   .050
                                 UC4   .089    ZR8   .286
                                 COX   .051    ZR7   .286
                                 ---   .000    ZC7   .040
                                 ---   .000    ZC8   .040
                                 ---   .000    LC1   .054
                                 ---   .000    LC2   .055
```

To activate the option to *change spec values* of intermediate streams, the user is required to define the stream(s), spec ID(s), new spec levels, and an activation year (not phased in for this version). The number of streams and spec IDs must also be included in the data file to act as controls for reading and processing the data. Up to ten spec types can be changed for each stream defined. To deactivate this option, the number of streams is set to zero. During the processing effort, the coefficients corresponding to the product component stream columns (B* and F*) and the product spec requirement rows (Q*) are updated using the stream spec data. Note, however, that a special algorithm must be used when changing the gravity and sulfur specs for streams used for blending into products. The gravity spec (GRX) must be converted using the following equation before being used to update the LP matrix:

$$coef_{gravity} = 141.5 / (131.5 + CHNGSPC_{gravity})$$

The sulfur spec (SLX) data for streams used in blending gasoline simply must be divided by 100 (to convert from percent to decimal), while that used in blending distillate product must be converted based on the sulfur spec data, the gravity spec data, and a distillate sulfur adjustment factor (DISTSUL). Thus, the following equation is used to define the coefficient corresponding to the distillate component stream column (F*) and the distillate sulfur requirement (Q*) row:

$$coef_{sulfur} = (CHNGSPC_{sulfur} - DISTSUL) * 141.5 / (131.5 + CHNGSPC_{gravity})$$

The data format representing an activated list of new spec levels for intermediate streams is presented below:

```
C Data for Spec improvement to reflect Technological Progress
C
C  SELECT FROM THE FOLLOWING LIST OF SPEC IDS:
C      GASO: R0N, M0N, RVX, E2N, E3N, SLX, OLX, ARX, BZX, PON, POX
C      DIST: SLX, GRX, VBX, LMX, FLX, FZX, ARX
C  REQUIRED: WHEN CHANGING SLX, MUST ALSO CHANGE GRX, AND VICE VERSA
C
C   |      |          |       |    MAX NUM OF SPECS = 10
C   | Tech |  Stream  | # of  |      1  new |      2  new |      3  new |      4  new|
```

C	Year	3-let ID	specs	spec, value	spec, value	spec, value	spec, value
@	----	--------	-----	---- -----	---- -----	---- -----	---- -----
		5					
	2005	ZL8	2	BZX 2.50	RVX 10.00		
	2005	8LR	3	BZX 2.40	RVX 10.00	ARX 10.00	
	2005	ZR8	3	BZX 0.88	GRX 65.00	SLX 200.00	
	2005	OR8	3	BZX 2.56	GRX 75.00	SLX 80.00	
	2005	BR8	3	BZX 0.18	GRX 50.00	SLX 320.00	

F.18 GTL Representation in PMM

In the PMM, a gas-to-liquids (GTL) facility now can be built on the North Slope in Alaska. A minimum build requirement is 50,000 bbl/d GTL production (gas: ~500MMCFD or 180 BCF a year). The investment and operating costs do not change over time (in constant 1987 dollars). The natural gas supply is represented as a three-step supply curve. A price/quantity pair (currently $0.80 in 2000 dollars/mcf for 1000 bcf) for North Slope natural gas is provided by the NGTDM team for the variable AKNG_SUPCRV. The PMM uses this to generate the supply curve. GTL output streams can be transported from the North Slope, through Valdez, to any of the three PMM regions. GTL mixing losses (PMM input) due to transport with the oil along the Trans-Alaska Pipeline System (TAPS) are accounted for and added to the Alaska oil total to be processed in a US refinery. A maximum flow (oil plus gas) is defined along the TAPS pipeline. [A minimum flow could also be defined, but would force the build of a GTL facility if the oil flow is below the minimum.] GTL transport costs from the North Slope to Valdez (along TAPS) is calculated within the PMM as a function of a variable cost, a fixed cost (converted to a unit cost based on t-1 GTL and oil flow), and a subsidy factor (based on value of oil if total flow is below a pipeline minimum). The fixed and variable costs are determined within the PMM based on input data. GTL transport costs from Valdez to California via vessel is also defined (PMM input).

The following LP variables and coefficients are related to Alaska GTL (and oil) production, transport, and accounting:

TAAMHXZ	Volume of AMH crude transported from Alaska to Valdez for export to Canada
O@CRDEXP	Other Alaskan crude exports (from South Alaska)
YAAMH5(r)	Volume of AMH crude transported from North Slope to Valdez + to region (r)
YAALL5(r)	Volume of ALL crude transported from S. AK to region (r)
PADCRQ1	Total Alaskan crude production
TAALLTOT	Total ALL oil produced in Alaska
TAAMHTOT	Total AMH oil produced in Alaska
TAGTLTOT	Total GTLs transported from Alaska North Slope to Valdez along TAPS
TANSOTOT	Total oil transported from Alaska North Slope to Valdez along TAPS
WAGTLJ(r)	Total GTLs transported from Valdez to US refinery regions (r)
GTLLOSS	Percent of GTLs lost due to mixing with Alaska oil during transport along TAPS
H(r)SMD(mod)	Operating level for the SMD GTL processing unit
H(r)SOD(mod)	Operating level for the SOD GTL processing unit
N(r)NGKN1	Alaska NG supply curve, step 1
N(r)NGKN2	Alaska NG supply curve, step 2
N(r)NGKN3	Alaska NG supply curve, step 3
H(r)MPRFSL	Quantity of GTL stream SNL transferred from North Slope to Valdez
H(r)MPRFSP	Quantity of GTL stream SNP transferred from North Slope to Valdez

H(r)MPRFSE	Quantity of GTL stream SKE transferred from North Slope to Valdez
H(r)MPRFSX	Quantity of GTL stream SDX transferred from North Slope to Valdez

The following new and modified equations define the relationship between the variables defined above as related to GTL production in Alaska.

Equation 1
Total GTLs produced in Alaska and transported from the Alaska North Slope to Valdez equals the sum of the individual GTL stream types (produced by each GTL unit's mode of operation) going to each of the 3 PMM regions. The corresponding TAPS transport cost is applied to the TAGTLTOT variable.

CAGTLTOT: TAGTLTOT = $\sum_r \sum_{gtl}$ H(r)MPRF(gtl)
OBJ: - 3.20 * TAGTLTOT

Equation 2
Total GTLs being transported via a US flag light product vessel from Valdez to PADD (r) equals the total GTLs produced in Alaska, minus the loss due to mixing with Alaskan oil during transport. The corresponding transport cost is applied to the WAGTLJ(r) variable.

C(r)GTL: WAGTLJ(r) = (1. - GTLLOSS) * $\sum_{gtl}$ H(r)MPRF(gtl)
OBJ: - cst * WAGTLJ(r)

Equation 3
The mass balance equation for AMH Alaskan crude was modified to include a gain due to mixing of GTLs during transport. This gain is accounted for in the Alaska North Slope oil stream [YAAMH5(r)].

CAAMH: **TAAMHTOT** + **GLTLOSS * TAGTLTOT** =
TAAMHXZ + O@CRDEXP + $\sum_r$ YAAMH5(r)

Equation 4
Since total North Slope Alaska crude does not consist totally of AMH crude, a new variable (TANSOTOT) is created to represent total North Slope Alaska crude, as defined within a new balance row (CANSOTOT). The corresponding TAPS transportation cost for NORTHSlope crude is applied to the TANSOTOT variable. [Note: The OBJ row coefficient on the Y variables now represent other transportation costs from Valdez to the PMM regions.]

CANSOTOT: TANSOTOT = TAAMHXZ +
$\sum_r$ YAAMH5(r) - GLTLOSS * TAGTLTOT
OBJ: - 3.20 * TANSOTOT
OBJ: - coef(r) * YAAMH5(r)

Equations 5,6,7
Two new row constraints are added to account for maximum and minimum flow requirements on TAPS; and, 1 new row constraint to account for maximum NG production in Alaska for GTL use.

TAOILGTX: TANSOTOT + TAGTLTOT <= TAPSUL
TAOILGTN: TANSOTOT + TAGTLTOT >= TAPSLL
TANGKGTX: $\sum_r \sum_s$ N(r)NGKN(s) <= NGKUL

Equations 8,9
New balance rows were defined for AMH and ALL Alaska crude.

CAAMHTOT: TAAMHTOT = .9844 * PADCRQ1
CAALLTOT: TAALLTOT = .0156 * PADCRQ1

Equations 10-13
Other mass balance additions and changes include mass balance for the GTL stream at the refinery [B(r)(gtl)], mass balance for the GTL stream generated in Alaska [H(r)(gtl)], mass balance for the Alaskan natural gas stream [H(r)NGK], and a capacity limit on the transportation mode (J) [TVPJCP]:

B(r)(gtl): $(1. - GTLLOSS) * H(r)MPRF(gtl) = \sum_{unt} \sum_{mod} R(r)(unt)(mod) + \sum_{prd} F(r)(prd)(gtl) + \sum_{prd} B(r)(prd)(gtl) + T(r)(gtl)(str)$

H(r)(gtl): $\sum_{gtl} H(r)MPRF(gtl) = coef_{gtl} * \sum_{unt} \sum_{mod} H(r)(unt)(mod)$

H(r)NGK: $\sum_{s} N(r)NGKN(s) = coef_{ngk} * \sum_{unt} \sum_{mod} H(r)(unt)(mod)$

TVPJCP: $. . . + .002 * \sum_{r} WAGTL(r)$

where,

R(r)(unt)(mod)	Refinery unit (unt) operating level for mode (mod) in region (r)
B(r)(prd)(gtl)	Quantity of GTL (gtl) blended into mogas product (prd) in region (r)
F(r)(prd)(gtl)	Quantity of GTL (gtl) blended into distillate product (prd) in region (r)
T(r)(gtl)(str)	Quantity of GTL (gtl) transferred into blend component (str) in region (r)

Changes to the PMM

Three new subroutines (CHGAKTRN, RFGTLCAP, RPTAKGTL) were added and an old subroutine (CHGTRANS) was changed in the refine.f code to reflect the new GTL capability in Alaska. In addition, another subroutine was changed (ADDCAP) and one other added (CHGBLDLIM) to better handle the PMM ability to specify which processing units are allowed to build. Finally, a new set of data were added to the bottom of the rfinvest.txt data file to define GTL parameters and control flags. The data are included below, followed by a list of the GTL variables defined.

New GTL Data

```
================================================================================
PMM input data related to GTL
================================================================================
        Data:           Parameters related to GTL process
        Variables:      GTL_INCBLD    -- incremental GTL output levels for building (Mbbl/d)
                        GTL_FSTYR     -- first possible start year for facility to come on-line
                        GTL_DCLCAPCST -- annual decline rate for capital recovery costs
                        GTL_DCLOPRCST -- annual decline rate for fixed operating costs
        Source: Analyst's judgement
        Notes:
--------------------------------------------------------------------------------
DOCUMENTATION UPDATES:  DATE--AUTHOR--COMMENT
--------------------------------------------------------------------------------
NOTES:
--------------------------------------------------------------------------------
@
        50.0    GTL_INCBLD
        2005    GTL_FSTYR
        0.000   GTL_DCLCAPCST
        0.000   GTL_DCLOPRCST

================================================================================
        Data:           Parameters related to TAPS volumes
        Variables:      TAP_MAXCAP   -- maximum capacity on TAPS (Mbbl/d)
        `               TAP_MINTHRU  -- minimum economic throughput on TAPS (Mbbl/d)
                        TAP_MINSTVOL -- minimum incremental volume above min when subsidy
                        nonzero (Mbbl/d)
                        TAP_PGTLOIL  -- fraction of GTL volume transferred to oil in TAPS (fraction)
        Source:         TAP_MAXCAP   -- analyst’s judgment
                        TAP_MINTHRU  -- analyst’s judgment
                        TAP_MINSTVOL -- analyst's judgment
                        TAP_PGTLOIL  -- analyst's judgment
--------------------------------------------------------------------------------
DOCUMENTATION UPDATES:  DATE--AUTHOR--COMMENT
--------------------------------------------------------------------------------
NOTES:
--------------------------------------------------------------------------------
@
        3000.00TAP_MAXCAP
        400.00          TAP_MINTHRU
        100.00          TAP_MINSTVOL
        0.10            TAP_PGTOIL
```

==========

Data: Parameters related to TAPS costs/prices
Variables: TAP_FIXCST -- fixed transportation cost on TAPS (1000 $/day)
TAP_VARCHG -- variable transportation cost on TAPS ($/bbl)
TAP_OILIFT -- assumed oil lifting cost in Alaska ($/bbl)
TAP_OILADJ -- minimum upward adjustment of lift cost to set total costs (e.g., including profit) (fraction)
Source: TAP_FIXCST -- judgment based on Alaska Department of Natural Resources graph of tariff rates and volumes
TAP_VARCHG-- judgment based on Alaska Department of Natural Resources graph of tariff rates and volumes
TAP_OILIFT -- lifting cost for oil production in Alaska
TAP_OILADJ -- analyst's judgment

DOCUMENTATION UPDATES: DATE--AUTHOR--COMMENT

NOTES:

TAPS transportation costs (PMM sets oceanic shipping costs elsewhere)

TAPS tariff = (fixed cost / throughput) + variable charge

throughput = oil prod + GTL prod (mbbl/d)

for build decision:
GTL prod = current CAPgtlns + GTL_INCBLD
Oil prod = PCTAKAMH * 1000* ($XRFQTDCRD_{10}$ + $XRFQTDCRD_{11}$ +$XRFQTDCRD_{12}$) - QEXCRDIN
Oil prod = oil_prod(t) * (oil_prod(t) / oil_prod(t-1))**3.

in general
GTL prod = current CAPGTLNS
Oil prod = oil_prod(t) = PCTAKAMH*RFQTDCRD*1000 - QEXCRDIN

Parameters for calculating GTL subsidy from oil production
if (oil_prod(t) .le. (TAP_MINTHRU + TAP_MINSTVOL))
then subsidy = [(oil prod * oil price) - (lift cost * oil prod * (1.+min add)] /GTL prod
---
@ year $ value
1995 2055. TAP_FIXCST
1995 0.90 TAP_VARCHG
1999 10.00 TAP_OILIFT
0.20 TAP_OILADJ

==========

New GTL Variables

```
! PUT INTO PMMCOM1 INCLUDE

CAPGTLNS(MNUMYR)        ! TOTAL GTL CAPACITY (bbl/d)
TAP_FIXCST              ! FIXED TRANS CST ON TAPS, 1987 dollars
TAP_VARCHG              ! VAR TRANS CST ON TAPS, 1987 dollars/BBL
TAP_OILIFT              ! ASSUMED OIL LIFTING CST IN AK, 1987 dollars/BBL
TAP_OILADJ              ! MIN UPWARD ADJ OF LIFT CST, 1987 dollars/bbl
TAP_MAXCAP              ! MAX CAP ON TAPS, MMBBL/D
TAP_MINTHRU             ! MIN ECONOMIC THROUGHPUT ON TAPS, MMBBL/D
TAP_MINSTVOL            ! MIN INCR VOL ABOVE MINTHRU, MMBBL/D
GTL_INCBLD              ! INCR GTL OUTPUT BLD LEVEL, MBBL/D
PMMCAPI(MNUMPR,PUNITSN)       ! INITIAL REF UNIT CAPACITY, MBBL/D
GTL_FSTYR               ! FIRST POSSIBLE START YR FOR GTL BLD, 4-digit

! PUT INTO PMMOUT INCLUDE

AKNG_GTLCNS(MNUMYR)  ! CONSUMP OF NG IN AK FOR GTL PROD, Bcf
AKGTLPRD(MNUMYR)        ! GTL PRODUCTION IN AK, Bbl/d
AKGTLEXP(MNUMYR)        ! AK GTL EXPORTS, Bbl/d

! PUT INTO OGSMOUT INCLUDE

AKNG_SUPCRV(3,2,mnumyr) ! ALASKA NG SUPPLY CURVE, 1987 dollars/mcf, bcf
                        ! 3 - number of points, 2 - price(1),quantity(2)
```

APPENDIX G

Matrix Generator Documentation

APPENDIX G. Matrix Generator Documentation

G.1 Introduction

The purpose of this appendix is to describe the program which runs the Multi-Refining Model (MRM) (3-regions) and the Enhanced Refinery Model (ERM) (single region); and to provide detail on how it works. The program allows the user to simulate a single region (ERM) or a 3-region (MRM) representation of the entire refining industry in the United States.

The ERM model simulates a single PADD (Petroleum Administration for Defense District) and has no transportation of crude oil to the refinery region nor product from the refinery region. Instead, crude oil is supplied directly to the refinery gate and product demands are satisfied at the refinery gate. The ERM projects petroleum product prices, product demands, crude oils, alcohols, ethers, natural gas and fuel consumption, and capacity expansion in a PADD.

The 3-region MRM simulates multiple PADD regions. It is a collection of ERMs linked by a transportation network. It simulates the entire refining industry in the United States. The MRM simulates the operation of petroleum refineries in the United States, including the supply and transportation of crude oil to refineries, the regional processing of these raw materials into petroleum products, and the distribution of petroleum product to meet regional demands. In addition to the quantities that an ERM produces, the MRM identifies sources of supply for domestic and imported crude oils, alcohols, ethers and natural gas. The 3-region MRM models the 5 U.S. PADDs, aggregrated into 3 regions and labeled as follows: E = PADD I; B = PADDs II, III, IV; W = PADD V.

The program generates the matrix for the linear programming model representation of the ERM or MRM, solves it, writes the solution, reports, and packs the matrix for use by the analyst using the ANALYZE software.

G.2 Code

The program is written in Fortran and makes use of the OML (Optimization and Modeling Libraries) to read in the data files, generate the matrix representation of the model, solve the problem, store the solution, and pack the matrix for use with ANALYZE. In addition to the above, the program produces reports.

The program is data driven and the user provides key information such as the model he chooses to run, the location of the input data files, and some other options.

G.2.a Variables and Constraints

The model consists of variables or activities (columns), constraints (rows), and bounds on activities. A unique name has been assigned to each variable and constraint. In the naming of the variables and rows, indices are used. The following table displays the index set name and the number of elements in the set, gives a brief description of the set, and provides a partial listing of the set members.

Index	No. of Values	Description	Members
@	1	represents all regions	@
c	2	Constraint type	X: for max N: for min
d	9	Census divisions	1: New England 2: Mid Atlantic 3: East North Central 4: West North Central 5: South Atlantic 6: Est South Central 7: West South Central 8: Mountain 9: Pacific
e	2	Emission source	C: Emission from fuel combustion N: Emission from process unit (non-combustion)
m	15	Transportation mode, function of material and means of movement	4: U.S. flag residual oil (dirty tanker) 5: South Atlantic node W: crude pipeline from supply reg 3 to PADD III B: Barge residual oil (clean barge) I: West Texas to PADD II J: U.S. flag light products (clean tanker) O: U.S. flag LPG R: PADD III loop to PADD II S: PADD III capeline to PADD II T: Light product pipeline U: LPG, C4, CC5 pipeline V: Barge residual oil (dirty barge) X: Local transportation Y: Pipeline PADD II to demand region 6 Z: Psudo link
o	8	OGSM regions	0: OGSM 1A East Coast 1: OGSM 1B East West part 2: OGSM 2 Gulf Coast 3: OGSM 3 Midcontinent 4: OGSM 4 Permian Basin 5: OGSM 5 Rocky Mountain 6: OGSM 6 West Coast A: OGSM A Alaska North

Index	No. of Values	Description	Members
r	3	Refining regions	E: PADD I B: PADD II, PADD III, PADD IV W: PADD V
t	2	Type of transportation	V: Vessel P: Pipeline
x	5	Exporting regions	2: Export cd for PADD I (region E, cd 2) 3: Export cd for PADD II (region B, cd 3) 7: Export cd for PADD III (region B, cd 7) 8: Export cd for PADD IV (region B, cd 8) 9: Export cd for PADD V (region W, cd 9)
Nn	3	Negative shift in demand	N1, N2, N3
On	8	Natural Gas refinery supply steps	N1, N2, N3, N4, P5, P6, P7, P8
Pn	3	Positive shift in demand	P1, P2, P3
qm	11	Quality code for gasoline blending	AR: Aromatics BZ: Benzene E2: E 200 E3: E 300 M0: Motor octane PO: Percent oxygen OL: Olefine R0: Research octane RV: Reid vapor pressure SL: Sulfur RE: renewables component (due to required minimum contribution to oxygenates)
qd	7	Quality code for distillate	AR: Aromatics FL: Flash point FZ: Freezing point GR: Gravity LM: Luminometer number SL: Sulfur VB: Viscosity
Qs	3	Step label for crude oil imports	Q1, Q2, Q3
Rs	9	Step label for product imports	R1, ..., R9,
S1	1	Step label for product demands	S1
SX	1	Product exports	SX
Z9	1	Distress imports and exports	Z9

Index	No. of Values	Description	Members
crt	12	Crude groups by quality and origin	ALL: Alaskan, API 25-66, S<0.5, B<15 AMH: Alaskan, API 21-32, S<1.1, B>15 DLL: Domestic, API 25-66, S<0.5, B<15 DMH: Domestic, API 21-32, S<1.1, B>15 DHL: Domestic, API 29-56, S<1.99, B<15 DHH: Domestic, API 23-35, S<3.0, B>15 DHV: Domestic, API<23, S>0.7, B>15 FLL: Foreign, API 25-66, S<0.5, B<15 FMH: Foreign, API 21-32, S<1.1, B>15 FHL: Foreign API 29-56, S<1.99, B<15 FHH: Foreign, API 23-35, S<3.0, B>15 FHV: Foreign, API<23, S>0.7, B>15
dfo	6	Distillate fuel oil blends	JTA: Jet fuel N2H: Number 2 oil DSL: Low sulfur diesel DSU: Ultra low sulfur diesel N6I: Low sulfur resid N6B: High sulfur resid
emu	6	Emission type	CAR: Total carbon CO1: Carbon monoxide CO2: Carbon dioxide NOX: Nitrous oxides SOX: Sulfur oxides VOC: Volatile organic compounds
ist	795	Refinery intermediate streams	LNI: Light naphtha, (175-250) intermediate LNN: Light naphtha, (175-250) naphthenic LNP: Light naphtha, (175-250) paraffinic
mgb	2	Gasoline blends	TRG: Conventional gasoline RFG: Reformulated gasoline
mod	many	Operating mode	C2A: Ethyl alkylate C3A: Propyl alkylate C4A: Butyl alkylate
ncr	10	Non crude purchase	ARB: Atmospheric resid of type B CC3: Propane ETH: Ethanol HGM: Heavy gas oil medium sulfur IC4: Isobutane MET: Methanol MTB: M.T.B.E. NAT: Natural gasoline NC4: Normal butane NPP: Paraffinic naphtha

Index	No. of Values	Description	Members
pol	36	Policy type	LOS: Lost OVC: Other variable cost MSD: Maximum distillation feed, cat cracker MSR: Maximum low sulfur resid, cat cracker SVR: Maximum severity, cat cracker H00: Maximum 100 severity, HP reformer H05: Maximum 105 severity, HP reformer
prd	20	Products	AST: Asphalt COK: Coke DSL: Low sulfur diesel DSU: Ultra low sulfur diesel E85: 85% Ethanol and 15% TRG JTA: Jet fuel KER: Kerosene LPG: Liquefied petroleum gas M85: 85% Methanol and 15% TRG N2H: Number 2 oil N67: Low sulfur resid to utilities N68: High sulfur resid to utilities N6B: High sulfur resid N6I: Low sulfur resid OTH: Other PCF: Petrochemical feed stock RFG: Reformulated gasoline RFH: Reformulated high oxygen gasoline TRG: Conventional gasoline TRH: Conventional high oxygen gasoline
pri	13	Product imports	DSL, DSU, JTA, LPG, MET, MTB, N2H, N6B, N6I, OTH, PCF, RFG, TRG, SSR
prx	10	Product exports	COK, DSL, DSU, JTA, LPG, N2H, N6B, N6I, OTH, PCF, TRG
px9	16	Distress exports	AST, COK, DSL, DSU, JTA, LPG, N2H, N67, N68, N6B, N6I, OTH, PCF, RFG, RFH, TRG, TRH
pi9	18	Distress imports	AST, COK, DSL, DSU, E85, JTA, LPG, M85, N2H, N67, N68, N6B, N6I, OTH, PCF, RFG, RFH, TRG, TRH
unf	3	Unfinished oil	ARB: Atmospheric residual bottom type B HGM: Heavy gas oil medium sulfur NPP: Medium naphtha paraffin
uns	74 (mrm) 56 (erm)	Process unit (excludes merchant & gas plant in ERM)	ACU: Atmospheric crude distillation See Appendix A.3 for complete list of processes
uuu	3	Utility	KWH: Kilo-watt hour NGF: Natural gas liquids STM: Steam

In the naming of the columns and rows, the limit is a maximum of eight characters per name. The following two tables give the name of the variable (activity) and the row (constraint) represented.

The general name structure for columns is (v)(r)(abc)(def), where v is key code, r is region code, abc and def are 3 character names.

MRM	ERM	Name	Activity Represented
x	x	B(r)(mgb)(ist)	Blend stream (ist) to gasoline grade (mgb) in (r)
x		C(d)ETCR(s)	Ethanol from cellulose supply step (s) in (d)
x		C(d)ETHR(s)	Ethanol from corn supply step (s) in (d)
x		D@METS1	Total US demand for methanol
x	x	D(d)(prd)S1	Product (prd) demand in (d)
x		D(d)(prx)SX	Product (prx) exports from (d)
x		D(d)(px9)Z9	Distress product (px9) export from (d)
x	x	E(r)(uns)INV	Investment in new capacity for process (uns) in (r)
x	x	F(r)(dfo)(ist)	Blend stream (ist) to distillate fuel oil (dfo) in (r)
x		G(r)DGR	Dry gas residue in (r)
x		G(r)GPL01	Gas plant operations mode 01 in (r)
x		G(r)(ist)(prd)	Gas plant output transfer of stream (ist) to product (prd) in (r)
x		G(r)MOH01	Methanol plant operations in (r)
x		G(r)(ist)RFN	Transfer of gas plant stream (ist) to refinery in (r)
x		G(r)METDEM	Methanol production from methanol plant in (r)
x		G(r)SC2CC1	Shift of ethane to natural gas in gas plant in (r)
x		G(r)SC3CC1	Shift of propane to natural gas in gas plant in (r)
x		H(r)(uns)(mod)	Production of mode (mod) for process (uns) at merchant plant in (r)
x		H(r)(aa)(bb)(ist)*	Transfer from (aa) to (bb) of stream (ist) in (r)
x		I(d)(pi9)Z9	Distress product (pi9) imports to (d)
x		I(r)(pri)R(s)	Imported product (pri) step (s) to region (r)
x	x	K(r)(uns)CAP	Existing capacity for process (uns) in (r)
x	x	L(r)(uns)BLD	Addition to capacity for process (uns) in (r)

MRM	ERM	Name	Activity Represented
	x	N(r)(ncr)	Purchase noncrude and additives (ncr) in (r)
x		N(r)DGP	Dry gas supply in (r)
x		N(r)NGKN(s)	Alaska NG supply curve step (s) for GTL production for region (r)
x		N(r)NGRF(On)	Natural gas to refinery supply step (On) in (r)
x		NZAMH(On)	Export supply step (on) for Alaskan crude
x		O@CRDEXP	Alaskan crude exports
x		O@CRDSPR	SPR fill in US
x		P(o)DCRQ1	Domestic crude in (o)
	x	P(r)(crt)	Supply of crude (crt) to (r)
x		P(r)(crt)(Qs)	Supply step (Qs) of imported crude (crt) to (r)
x		PANGLQ1	Supply of natural gas liquids from Alaska North slope
x	x	Q(r)(mgb)	Spec vector, total volume for (mgb) in (r)
x	x	Q(r)(dfo)	Spec vector, total volume for (dfo) in (r)
x	x	R(r)ACU(crt)	Volume of crude (crt) processed by the ACU unit in (r)
x	x	R(r)(uns)(mod)	Refinery process (uns) operation for mode (mod) in (r)
x	x	R(r)CGNCGN R(r)CGXCGN	Refinery (CGN) and Merchant (CGX) cogeneration plant operation in region (r)
x	x	T(r)(ist)(ist)	transfer of stream (ist) to stream (ist) in (r)
x		T@UNFTOT T(r)UNF(ist)	Total unfinished oils in US Unfinished oils from stream (ist) in region (r)
x		TAALLTOT	Total Volume of ALL crude produced in Alaska
x		TAAMHTOT	Total Volume of AMH crude produced in Alaska
x		TAAMHXZ	Volume of AMH crude transported from Alaska to Valdez
x		TAGTLTOT	Total volume of GTL's transported from Alaska N. Slope to Valdez along TAPS
x		TANSOTOT	Total volume of crude transported from Alaska N. Slope to Valdez along TAPS
x	x	T(r)CBNTAX	Carbon tax in (r)
x		T(r)OVCOBJ	Refinery plant operating variable costs in region (r)
x		T(r)GPLOVC	Gas pland operating variable costs in region (r)
x		T(r)MCHOVC	Merchant plant operating variable costs in region (r)

MRM	ERM	Name	Activity Represented
x	x	U(r)(uuu)	Utility (uuu) purchased in (r)
x		VTVC(m)CP	Crude vessel transportation capacity for mode (m)
x		VTVP(m)CP	Product vessel transportation capacity for mode (m)
x		VTPC(r)(m)(d)	Crude pipeline transportation capacity from (r) to (d) using mode (m)
x		VTPP(r)(m)(d)	Product pipeline transportation capacity from (r) to (d) using mode (m)
x		VTPL(r)(m)(d)	LPG pipeline transportation capacity from (r) to (d) using mode (m)
x		WAGTLJ(r)	GTL transportation from Alaska (A) to region (r) using mode J
x		W(r)(prd)(m)(d)	Product (prd) transportation from (r) to (d) using mode (m)
x		W(d)(prd)(m)(r)	Product (prd) transportation from (d) to (r) using mode (m)
x		X(d)(prd)SPG	Recipe blends of product (prd) for oxygenated fuels and electric utility residual oils in (d)
x	x	X(r)(ist)(prd)	Recipe blends of product (prd) from stream (ist) in region (r)
x		X(d)(ist)(prd)	Splash blending of (prd) from stream (ist) in (d)
x		X(r)(yyyy)**	Recipe blends (yyyy) in region (r)
x		Y(o)(crt)(m)(r)	Crude (crt) transportation from (o) to (r) using mode (m)
x		Z(d)ETCTAX	Total supply of ethanol from cellulose in (d) for adding tax credit to objective function
x		Z(d)ETHTAX	Total supply of ethanol from corn in (d) for adding tax credit to objective function
x		Z(r)FLO(uns)	Total flow through processing unit (uns) in region (r)
		Z(r)RFGOXY	Total OXY in RFG in region (r)
x		Z@TOTCRD	Total foreign crude imports
x		ZZAMHTOT	Total export volume of Alaskan crude oil

* : (aa), (bb) = MP, GP, RF, where MP = Merchant plant, GP = Gas plant, RF = Refinery. For (ist), first and last character of (ist).

**: (yyyy) = recipe blends for categories of products such as AST, PCF, OTH.

The general name structure for rows is: (v)(r)(abc)(def), where v is key code, r is region code, abc and def are 3 character names.

MRM	ERM	Name	Constraint Represented
x		A(d)(prd)	Product demand accounting row in (d) for (prd)
x		A(d)RFG(yyy) A(d)TRG(yyy)	Gasoline blending accounting row in (d) for mogas (yyy)
x		A(x)PRDEXP A@PRDEXP	Product export accounting row in (x) and US
x		A(d)ETHTAX A(d)ETCTAX	Ethanol tax credit accounting row in (d)
x	x	A(r)INVST A@INVST	Investment accounting row in (r) and US
x	x	A(r)(prd)	Product (prd) refined accounting row in (r)
x	x	A(r)CRX(crt)	Crude oil (crt) accounting row in (r)
x		A@CRDAKA	Alaskan crude oil accounting row in US
x		A@CRDDCR	Domestic crude oil accounting row in US
x		A@CRDEXP	Crude oil export accounting row in US
x	x	A(r)CRDFCR A@CRDFCR	Foreign crude oil accounting row in (r) and US
x		A@CRDL48	Lower 48 crude oil accounting row in US
x		A@CRDSPR	SPR crude oil accounting row in US
x	x	A@CRDTOT	Total crude oil accounting row in US
x		A(r)ETHRFN	Ethanol to refinery accounting row in (r)
x	x	A(r)FUEL A@FUEL	Fuel use accounting row in (r) and US
x	x	A(r)FUM(xx) A@FUM(xx)	FUM accounting row in (r) for (xx) and US
x	x	A(r)FXOC A@FXOC	Fixed cost accounting row in (r) and US
x		A(d)GO8(yyy)	Gasoline blending accounting row in (d) for mogas (yyy)
x	x	A(r)G(I)(xxx) A@G(I)(xxx)	Gasoline blend components accounting row in (r) for (xxx); US total

MRM	ERM	Name	Constraint Represented
x	x	A(r)GAIN A@GAIN	Gain accounting row in (r)
x	x	A@KWHRFN	Refinery KWH usage accounting row in US
x		A(r)METIMP A@METIMP	Methanol imports accounting row in (r); US total
x		A@METDEM	Methanol demand accounting row in US
x		A@METM85	Accounting of methanol used for M85 splash blending in US
x		A@METPRD	Methanol production accounting row in US
x	x	A(r)METRFN	Accounting of methanol consumption by ETH refinery unit in (r)
x		A(r)MTBRFN	MTB refinery imorts accounting row in (r)
x	x	A(r)NGFTOT A@NGFTOT	Natural gas purchase accounting row in (r); US total
x		A(r)NGLRFN A@NGLRFN	Accounting of NGL transfer from gas plant to refinery in (r); US total
x	x	A(r)NGLPRD A@NGLPRD	NGL accounting in (r); US total
x	x	A(r)NGSH2P A@NGSH2P	Accounting of NGS consumption by H2P refinery unit in (r); US total
x		A(r)NGSMER A@NGSMER	Accounting of methanol transfer from gas plant to refinery in (r); US total
x		A(r)NGSMET A@NGSMET	Methanol plant production accounting row in (r); US total
x	x	A(r)NGSRFN	Accounting of NGF stream transfer to NGS stream in (r)
x	x	A(r)PETCOK A@PETCOK	Accounting of high & low sulfur coke production from recipe blending in (r) and US
x	x	A(r)SULSAL A@SULSAL	Accounting of sulfur production from recipe blending in (r) and US
x	x	B(r)(ist)	Balance for intermediate stream (ist) in (r)
x		CAALLTOT	Balance for Alaska crude type ALL
x		CAAMHTOT	Balance for Alaska crude type AMH
x		CAGTLTOT	Balance for GTL's produced in Alaska
x		CANSOTOT	Balance for Alaska N. Slope crude
x		C(o)(crt)	Crude balance for crude type (crt) in (o)

MRM	ERM	Name	Constraint Represented
x	x	C(r)(crt)	Crude balance for crude type (crt) in (r)
x		C(r)GTL	Balance for GTL's transported from Alaska to region (r)
x		CZAMH	Alaskan crude exports
x	x	D(d)(prd)	Final demand for product (prd) in (d)
x		D@MET	Total demand for methanol
x		E(r)(emu)(e)	Emission of (emu) from source (e) in (r)
x		F(r)UNF(unf)	Unfinished oil balance for (unf) in (r)
x	x	F@TOTCRD	Total crude balance for unfinished oil constraint in US
x		G(r)(ist)	Gas plant balance for stream (ist) in (r)
x		G(r)(pol)	Gas plant policy (pol) accounting row in (r)
x		H(r)(ist)	Merchant oxygenate plant balance row for (ist) in (r)
x		H(r)FUMCAP	Merchant oxygenate fuel balance row in (r)
x		H(r)(pol)	Merchant oxygenate plant policy (pol) accounting row in (r)
x	x	L(r)(uns)CAP	Process (uns) capacity in (r)
x		M(r)MTBRFG	MTBE (and others) limit in RFG motor gasolines in (r)
x		M(r)MTBTRG	MTBE (and others) limit in TRG motor gasolines in (r)
x	x	M(r)(prd) M(d)(prd)	Final product (prd) demand at refinery (r) or CD (d)
x		O(o)(crt)	Domestic crude oil (crt) accounting in (o)
x	x	OBJ	Objective function
x	x	P(r)(pol)	Policy (pol) constraint in (r)
x	x	P(r)CBNTAX	Carbon tax accounting row in (r)
x	x	Q(r)(prd)(qd)(c) Q(r)(prd)(qm)(c)	Product (prd) specification for quality (qd) constraint type (c) in (r) Product (prd) specification for quality (qm) constraint type (c) in (r)
x	x	S(r)(mgb)E	Sum row for blending gasolines (mgb) in (r)
x	x	S(r)(dfo)E	Sum row for blending fuel oils (dfo) in (r)
x	x	S(r)RFGOXY	Constraint on renewable OXY limits in (r)
x		TANGKGTX	Maximum NG production in Alaska for GTL use
x		TAOILGTN	Minimum flow requirement on TAPS in Alaska

MRM	ERM	Name	Constraint Represented
x		TAOILGTX	Maximum flow requirement on TAPS in Alaska
x		TVC(m)CP	Crude oil vessel transportation capacity limits for mode (m)
x		TVP(m)CP	Product vessel transportation capacity limits for mode (m)
x		TPC(r)(m)(d)	Crude pipeline transportation capacity balance row
x		TPL(r)(m)(d)	LPG pipeline transportation capacity balance row
x		TPP(r)(m)(d)	Product pipeline transportation capacity balance row
x	x	U(r)(uuu)	Utilities (uuu) in region r
x	x	Z(r)CAP(uns)	Balance row for total capacity of (uns) in (r)
x		Z(r)NGFSUM	Sum row for natural gas to refineries in (r)
x		ZZAMHSUM	Sum row for Alaskan crude export
x	x	Z@WOP	Current world oil price in value of RHS
x	x	Z@CRDTOT	Sum row for total crude in US
x	x	Z@YRITER	Iteration year
x		Z@IRACX	Sum row to force average refinery crude cost within specified range
x		Z@IRACN	Sum row to force average refinery crude cost within specified range
x	x	ZD(mmddyy)	Month, day, year of matrix generation

(I) = 00, 01, 02, ... , 12
(xx) = LPG, N2H, N6B, N6I, NGS, OTH, STG
(xxx) = TRG and RFG
(yyy) = TRG, TRH, and/or RFG, RFH
There are several other accounting rows.

The following Table gives the dimensions of each model:

Model	Columns		Rows	
	Total	Fixed	Total	Fixed
MRM (3 regions)	11295	438	4102	3219
ERM (single region)	2274	38	914	709

The general structure for the model is as follows:

MRM Model Block Diagram

	Crude Trans.	Purchases Crude Oil, Other Inputs	Crude Distillation	Other Process Unit Operations	Capacity Expansion	Stream Transfers	Blending	Product Demand	Product Trans.	Row Type	RHS
Objective	-ct	-c	-o	-o	-i			+p	-pt	NC	Max
Crude Oil Balance	+1 +1	+1 +1	-1 -1							EQ	0
Intermediate Stream Balance			+y +y	-1 -1 +y +y		-1 +1 -1 +1	-1 -1			EQ	0
Utilities		+1	-u	-u +1						EQ	0
Policy Constraints				+z -z				+z - z		GE LE	0
Environmental Con-straints			+q	+q						GE LE	E
Unit Capacities			+1	+1	-1					LE	K
Quality Specifications							+q +q -Q			GE LE	0
Product Sales							-1	-1	-1 +1 +1 -1	EQ	0
Pipeline/Marine Capacities	+1 +1								+1 +1	LE	C
Bounds	Up/Lo/Fix	Up/Lo/Fix						Lo/Fix			

Legend: c = crude cost; p = price; Q = product specifications; y = yield; z = policy ratio; C = pipeline/marine capacity; u = utility consumption; q = stream quality; K = unit capacity; E = environmental quality limit; ct = crude transportation cost; pt = product transportation cost; o = operating cost; i = investment cost

G.2.b Subroutines

The program consists of several subroutines and a main program. The subroutines can be grouped as those that setup the OML environment, read in the data tables, form parts of the matrix representation of the model, solve the model, retrieve needed information for report writing, and write the reports. All the subroutines that generate part of the matrix representation of the model use input from data files in an OML format. These files have a .dat extension. Some subroutines use ASCII files as input and some others don't use any. The following table shows the subroutine names, the input data file names, the purpose of the subroutines, and the model that uses them.

Source Code	Data file	Purpose	Models
accunit.f	accunit.dat	Represents ACU unit	All
akaexp.f	akaexp.dat	Represents Alaskan exports	MRM
avoids.f	avoids.dat	Represents the avoids	MRM
cogener.f	cogener.dat cogenere.dat	Represents the cogeneration	MRM ERM
crdimprt.f	crdimprt.dat	Represents crude imports	MRM
demand.f	demand.dat	Represents demands	MRM
distblnd.f	distblnd.dat	Represents distillate blending	All
distress.f	distress.dat	Represents the distress imports and exports	MRM
domcrude.f	domcrude.dat	Represents crude inputs	MRM
emish.f	emish.dat	Represents emissions	MRM
ermcrude.f	ermcrude.dat	Represents crude inputs	ERM
ermother.f	ermother.dat	Represents non-crude inputs	ERM
ermprod.f	ermprod.dat	Represents product demands	ERM
ethanol.f	ethanol.dat	Represents ethanol supply and prices	MRM
fixcol.f	fixcol.dat	Fixes some columns	All (not used)
fuelmix.f	fuelmix.dat	Simulates fuel mixing	All
gasoblnd.f	gasoblnd.dat	Simulates gasoline blending	All
limpol.f	limpol.dat	Puts limits on policy rows	All
lplookup.f		Retrieves solution	MRM (5-reg only)
mrm.f	mrmparam, mrmpath main.dat ermparam pathe maine.dat	Program MPS2ANAL: Sets up the OML environment, reads in some main data, controls the program, calls subroutines to form matrix, solves problem, stores solution, writes reports, basis and packs matrix.	MRM ERM
mchproc.f	mchproc.dat	Simulates the merchant plant	MRM

Source Code	Data file	Purpose	Models
ngprod.f	ngprod.dat	Provides gas supply steps to refinery	MRM
nrfplant.f	nrfplant.dat	Simulates the non refinery plant activities	MRM
output.f		Prints a report	MRM (5-reg only)
prdexp.f	prdexp.dat	Simulates the product exports	MRM
prdimprt.f	prdimprt.dat	Simulates the product imports	MRM
recipes.f	recipes.dat	Specifies product recipe blends	All
refproc.f	refproc.dat	Simulates the refinery	All
setrows.f	setrows.dat	Sets some rows	All
splash.f	splash.dat	Simulates splash blending	MRM
stream.f	stream.dat	Simulates stream transfers	All
tabread.f		Reads data tables	All
transit3.f	transit.dat	Simulates the product and crude oil transportation for the 3-region representation of MRM	MRM
unfinished.f	unfinish.dat	Provide for unfinished oil imports	MRM
utility.f	utility.dat	Simulates utility purchased	MRM
utilitye.f	utilitye.dat	Simulates utility purchased	ERM

Most of the subroutines that constitute the program generate part of the matrix representation of the model. The following gives a representation of the submatrix generated by each subroutine in table form. Columns of the tables correspond to activities (variables), and rows of the tables to constraints. The symbols x, -x or +-x represent matrix coefficients.

accunit.f: This subroutine simulates the ACU unit. It creates the following submatrix:

	R(r)ACU(crt)
C(r)(crt)	-x
L(r)ACUCAP	x
A(r)CRDFCR*	x
A(r)STM	-x
A@CRDFCR*	x
A(r)CRX(crt)	x
A@CRDTOT	x
Z@CRDTOT*	x
F@TOTCRD	x
B(r)(ist)	+-x

	R(r)ACU(crt)
U(r)(uuu)	-x
P(r)(pol)**	+-x

* for (crt) = FLL, FMH, FHL, FHH, and FHV
** for (pol) = OVC, FRL

Bounds: None

akaexp.f: This subroutine simulates the Alaskan exports. It creates the following submatrix:

	NZAMH(i)	ZZAMHTOT	TAAMHXZ	PANGLQ1
OBJ	+-x*	x	-x	-x
CAAMH			-x	
CZAMH		-x	x	
BW(ist)				x
ZZAMHSUM	x	-x		
A@AKAEXP		x		
A@CRDEXP		x		
AANGLPRD				x
A@NGLPRD				x
AWNGLRFN				x

(i) = N1, N2, N3, P4, P5, P6
* : -x if i = N1, N2, N3; x if i = P4, P5, P6

Bounds: PANGLQ1, NZAMH(i)

avoids.f: This subroutine simulates the avoids. It is turned off. It creates the following submatrix:

	D(d)(prd)N(i)	D(d)(prd)P(i)
D(d)(prd)	x	-x
A(d)(prd)	-x	x
A@PRDDEM	-x	x
A@AVDNEG	x	
A@AVDPOS		x

(i) = 1, ..., 3

Bounds: D(d)(prd)N(i) and D(d)(prd)P(i)

cogener.f: This subroutine simulates the cogeneration unit. It creates the following submatrix

	E(r)CGNINV	K(r)CGNCAP	L(r)CGNBLD	R(r)CGNCGN
B(r)FUL				-x
L(r)CGNCAP	-x	-x	-x	x
OBJ	-x		-x	x
P(r)OVC				-x
U(r)(uuu)				x
A@FXOC	x		x	
A(r)FXOC	x		x	
A@INVST	x			
A(r)INVST	x			
A@KWHRFN				x

	E(r)CGXINV	K(r)CGXCAP	L(r)CGXBLD	R(r)CGXCGN
B(r)FUL				-x
L(r)CGXCAP	-x	-x	-x	x
OBJ	-x		-x	x
P(r)OVC				-x
U(r)(uuu)				x
A@FXOC	x		x	
A(r)FXOC	x		x	
A@INVST	x			
A(r)INVST	x			
A@KWHRFN				x

Bounds: E(r)CGNINV, K(r)CGNCAP, L(r)CGNBLD, E(r)CGXINV, K(r)CGXCAP, L(r)CGXBLD

crdimprt.f: This subroutine simulates the crude imports into the United States. It creates the following matrix:

	P(r)(crt)(Qs)
C(r)(crt)	x
OBJ	-x
Z@IRACN	x
Z@IRACX	x

(Qs) = supply step Q1,Q2,Q3

Bounds: P(r)(crt)Q(s)

demand.f: This subroutine simulates product demands. It creates the following submatrix:

	D(d)(prd)S1	D@METS1
D(d)(prd)	-x	
OBJ	x	
A(d)(prd)*	x	
A@METDEM		x
A@PRDDEM	x	
D@MET		-x

* for (prd) not equal to E85 or M85

Bounds: D@METS1 and D(r)(prd)S1

distblnd.f: This subroutine simulates the distillate blending. It creates the following matrix:

	F(r)(dfo)(ist)	Q(r)(prd)*
B(r)(ist)	+-x	
M(r)(prd)*		x
Q(r)(prd)*(qd)(c)	+-x	-x
S(r)(dfo)E	x	-x
U(r)STM		-x
A(r)(prd)*		x
A(r)STM		-x

* for (prd) = (dfo) only

Bounds: None

distress.f: This subroutine simulates product distresses. It creates the following submatrix:

	I(d)(pi9)Z9	D(d)(px9)Z9
OBJ	-x	-x
D(d)(pi9)	x	
A@ZZIMP	x	
D(d)(px9)		-x
A@ZZEXP		x

Bounds: None

domcrude.f: This subroutine simulates domestic crudes. It creates the following submatrix:

	P(o)DCRQ1	PADCRQ1	O@CRDEXP	O@CRDSPR
OBJ	-x	-x		
C(o)(crt)	x			
CAALL		x		
CAAMH		x	-x	
CBFHL				-x
A@CRDDCR	x	x		
A@CRDAKA		x		
A@CRDL48	x			
A@CRDEXP			x	
A@CRDSPR				x
A@CRDFCR				x
O(o)(crt)	x	x		

(o) : except A
Bounds: P(o)DCRQ1, PADCRQ1, O@CRDEXP, O@CRDSPR

emish.f: This subroutine simulates emissions. It creates the following submatrix:

	K(r)(uns)CAP	R(r)FUM(ist)
E(r)(emu)N	x	
E(r)(emu)C		x

Bounds: None

ermcrude.f: This subroutine simulates domestic crudes for ERM. It creates the following submatrix:

	P(r)(crt)*
C(r)(crt)*	x
OBJ	-x

* (crt) = FHH, FHL, FHV, FLL, FMH, DLL, DMH, DHL, DHH, DHV.
(r) = region G only (i.e., single region)

Bounds: P(r)(crt)

ermother.f: This subroutine simulates the non-crude inputs for ERM. It creates the following submatrix:

	N(r)(ist)*
B(r)(ist)*	x
OBJ	-x
A(r)(ist)*	x

* (ist) = (ncr) only
(r) = region G only (i.e., single region)

Bounds: N(r)(ist)

ermprod.f: This subroutine simulates product demands for ERM. It creates the following submatrix:

	D(r)(prd)S1
M(r)(prd)	-x
OBJ	x

(r) = region G only (i.e., single region)
Bounds: D(r)(prd)S1

ethanol.f: This subroutine represents ethanol supply and prices. It creates the following submatrix:

D(d)ETH	C(d)ETCR(i) x	Z(d)ETCTAX	C(d)ETHR(i) x	Z(d)ETHTAX	X(d)ETHE85
OBJ	-x	x	-x	x	-x
A(d)ETH	x		x		
A(d)ETHTAX			x	-x	
A@ETHPRD	x		x		
A(d)ETCTAX	x	-x			
A@ETCPRD	x				

i = 1, ..., 4

Bounds: C(d)ETHR(i), C(d)ETCR(i)

fixcols.f: This subroutine fixes some variables or activity. No submatrix is generated

Bounds: R(r)FCC(ist)

fuelmix.f: This subroutine simulates fuel mixing. It creates the following submatrix:

OBJ	R(r)FUM(mod)	T(r)CBNTAX -x	K(r)FUMCAP
P(r)CBNTAX		x	
A(r)FUM(mod)	x		
A@FUM(mod)	x		
A(r)FUEL			x
A@FUEL			x

Bounds: None

gasoblnd.f: This subroutine simulates gasoline blending. It creates the following submatrix:

B(r)(ist)	B(r)(mgb)(ist) -x	Q(r)(mgb)	Z(r)RFGOXY
M(r)(prd)		x	
Q(r)RFGREN			-x
Q(r)(prd)(qq)(c)	x	-x	
S(r)(mgb)E	x	-x	
S(r)RFGOXY			-x

	B(r)(mgb)(ist)	Q(r)(mgb)	Z(r)RFGOXY
U(r)KWH		-x	
A(r)(xxx)(mgb)	x		
A(r)(prd)*		x	
A@(xxx)(mgb)	x		
A@PRDRFN		x	
M(r)MTB(mgb)	x **		

* (prd) = (mgb) only; **ist=MTB,ETB,TAE,TAM,THE,THM only; (xxx) = GO1, ..., G12

Bounds: B(r)RFG(ist) and B(r)TRG(ist) where ist= ETB, MTB, TAE, TAM, THE, THM

limpol.f: This subroutine defines policy conditions. It creates the following submatrix:

	K(r)(uns)CAP	E(r)(uns)INV	L(r)(uns)BLD	Z(r)FLO(uns)
Z(r)CAP(uns)	x	x	x	-x
P(r)(pol)				-x

(uns) = specific units underpolicy controls (i.e., FCC, KRF,...)

Bounds: None

lplookup.f: Retrieves solution values and stores them in arrays.

main.f: This subroutine reads in the mrmparam file that has the information relative to the model to run; initializes the OML subroutine library environment; opens the database; specifies a problem in the database for processing; initializes the matrix processing; reads in the path file, the main.dat file that contains some global variables such as the refinery, the exporting, and demand regions code; calls the subroutines that generate the matrix; ends the matrix processing; writes out the MPS file; inserts the advanced basis; solves the matrix; puts the solution in output; writes the optimal basis; prints reports; packs the matrix; and closes the database.

mchproc.f: This subroutine represents the merchant plant. It creates the following submatrix

	K(r)(uns)CAP	E(r)(uns)INV	L(r)(uns)BLD	K(r)FUXCAP	H(r)(uns)(mod)
H(r)FUMCAP				x	x
L(r)(uns)CAP	-x	-x	-x		x
OBJ		-x	-x		
A@FXOC		x	x		
A(r)FXOC		x	x		
A@INVST		x			

	K(r)(uns)CAP	E(r)(uns)INV	L(r)(uns)BLD	K(r)FUXCAP	H(r)(uns)(mod)
A(r)INVST		x			

mchproc.f (Continued)

	H(r)(uns)(mod)	H(r)KWHMCH	H(r)(aa)(bb)(ist)*	T(r)MCHOVC	N(r)NGKN(s)	RHS
H(r)(ist)	+-x		+-x		x	
H(r)(uuu)	+-x					
H(r)(pol)	+-x					
G(r)(ist)			+-x			
B(r)(ist)			+-x			
H(r)KWH		x				
H(r)OVC				x		
TANGKGTX					x	x
A@NGLPRD			x			
OBJ		-x	-x	-x	-x	

mchproc.f (Continued)

	TANSOTOT	TAGTLTOT	RHS
TAOILGTN	x	x	x
TAOILGTX	x	x	x
CAAMH		x	
CAGTLTOT		-x	
CANSOTOT		-x	

*: first and last character of (ist); (aa) and (bb) = MP, GP, RF
where MP = Merchant plant, GP = Gas plant, and RF = Refinery

Bounds: K(r)(uns)(CAP, E(r)(uns)INV, L(r)(uns)BLD, H(r)GPMP(ist) and H(r)RFMP(ist) = 0;
N(r)NGKN(s)

ngprod.f: This subroutine represents the gas supply steps to refinery. It creates the following submatrix

	N(r)NGRF(ij)*
OBJ	+-x**
Z(r)NGFSUM	x

* (ij) = N1, N2, N3, N4, P5, P6, P7, P8
** +x for N1, N2, N3, N4 and -x for P5, P6, P7, P8

Bounds: N(r)NGRF(ij)

nrfplant.f: This subroutine simulates the non refinery plant. It creates the following submatrix:

	G(r)DGR	G(r)GPL01	T(r)GPLOVC	N(r)DGP
G(r)OVC	-x		x	
G(r)DGP	x	-x		
G(r)LOS	-x	-x		
G(r)(xxx)		x		
G(r)CC1	-x			x
OBJ			-x	

nrfplant.f (Continued)

G(r)(xxx)	G(r)NATOTH -x	G(r)NATPCF -x	G(r)(xxx)*LPG -x	G(r)(xxx)**RFN -x
OBJ	-x	-x	-x	-x
A(r)GPL(xxx)***	x	x	x	
A(r)NGLRFN				x
A@NGLRFN				x
B(r)(xxx)**				x
M(r)(prd)***	x	x	x	
A(r)NGLPRD	x	x	x	x
A@NGLPRD	x	x	x	x

nrfplant.f (Continued)

	G(r)SC2CC1	G(r)SC3CC1	G(r)METRFN	G(r)METDEM
G(r)CC1	x	x		
G(r)CC3		-x		
G(r)LOS	x	x		
G(r)PGS	-x			
OBJ	x	x	-x	-x
A(r)NGSMER			x	
A@NGSMER			x	
B(r)MET			x	
G(r)MET			-x	-x
D@MET				x

nrfplant.f (Continued)

	E(r)MOHINV	L(r)MOHBLD	K(r)MOHCAP	G(r)MOH01
L(r)MOHCAP	-x	-x	-x	x
G(r)MET				x
G(r)OVC				-x
U(r)NGF				-x
U(r)KWH				-x
A(r)NGSMET				x
A@NGSMET				x
A@METPRD				x
OBJ	-x	-x		
A(r)INVST	x			
A@INVST	x			
A(r)FXOC	x	x		
A@FXOC	x	x		

(xxx) : PGS, CC3, IC4, NC4, NAT
* : (xxx) except NAT ; **: (xxx) except PGS, CC3; *** LPG, OTH, PCF

Bounds: G(r)SC3CC1, N(r)DGP, E(r)MOHINV, K(r)MOHCAP, L(r)MOHBLD

output.f: This subroutine prints reports.

prdexp.f: Simulates product exports. It creates the following submatrix:

	D(z)(prx)*SX	D(z)COKSX	D(z)ASTSX
D(z)(prx)	-x	-x	-x
OBJ	x	x	
A@COKEXP		x	
A(d)PRDEXP	x		x
A@PRDEXP	x		x

*: All (prx) except COK and AST; (z) = export demand regions (d) 2,3,7,8,9

Bounds: D(z)(prx)SX, except for (prx) = COK

prdimprt.f : This subroutine simulates product imports. It creates the following submatrix:

	I(r)(pri)*R(s)	I(r)(pri)**R(s)
A@(pri)*IMP	x	
A(r)(pri)*IMP	x	
B(r)(pri)*	x	
A(r)(pri)RFN	x***	
A(r)PRDIMP		x
A@PRDIMP		x
M(r)(pri)**		x
OBJ	-x	-x

* : for (pri) = MET and MTB; **: for all (pri) except MET and MTB;
*** for (pri) = MTB only; (s) = import steps 1-3 or 1-9

Bounds: I(r)(pri)R(s)

recipes.f: This subroutine simulates product recipe blending. It creates the following submatrix:

	X(r)(xxx)(yyy)	X(r)(yyyy)
B(r)(ist)	-x	-x
M(r)(yyy)	x***	x
OBJ	x	-x*
U(r)STM		-x**
A(r)STM		-x**

	X(r)(xxx)(yyy)	X(r)(yyyy)
A(r)(yyy)	x***	x
A(r)SULSAL	x****	
A@SULSAL	x****	
A(r)PETCOK	x***	
A@PETCOK	x***	
A@PRDRFN	x***	x

(xxx) = AST, AVG, CKH, CKL, GOP, SUL ; (yyy) = AST, OTH, COK, PCF, SAL
(yyyy) = AST0, AST1, AVG0, GOP0;
: for (yyyy) = AVG; **: for (yyyy) = AST0; ***: for (yyy) = COK; **** for (yyy) = SAL

refproc.f: This subroutine simulates the refinery processes. It creates the following submatrix:

	E(r)(uns)INV	K(r)(uns)CAP	L(r)(uns)BLD	T(r)OVCOBJ	R(r)(uns)(mod)	RHS
OBJ	-x		-x	-x		
B(r)(ist)					+-x	
L(r)(uns)CAP	-x	-x	-x		x	
U(r)(uuu)					+-x	
P(r)(pol)				x*	+-x	
P(r)CBNTAX*****					-x	
M(r)(prd)******					-x	
A(r)INVST	x					
A@INVST	x					
A(r)FXOC	x		x			
A@FXOC	x		x			
A(r)GAIN					+-x	
A@GAIN					+-x	-x
A(r)METRFN					x**	
A(r)NGSH2P					x***	
A@NGSH2P					x***	
A@MTBPRD					x****	
A(r)STM					-x	

*: when (pol) = OVC; **: when (uns) = ETH; ***: when (uns) = H2P; ****: when (uns) = ETH and (mod) = MTB; *****: FUM only; ******: N6I, N6B only.

Bounds: K(r)(uns)CAP, E(r)(uns)INV, L(r)(uns)BLD, R(r)(uns)(mod)
Range: A@GAIN

setrows.f: This subroutine sets the row types (G, L, E, N) for rows P(r)(pol).

splash.f: This subroutine simulates splash blending. It creates the following submatrix:

	X(d)ETH(xxx)	X(d)METM85	X(d)(www)SPG	B(r)RFG(sss)	B(r)RFG(rrr)
D(d)(xxx)	x	-x****			
D(d)ETH	-x				
D(d)(yyy)	-x				
D(d)M85		x			
D(d)MET		-x			
A(d)G08(xxx)	x*				
A@ETH(xxx)	x				
A@METM85		x			
A(d)RFG(xxx)** not linked to col					
A(d)TRG(xxx)	x***				
D(d)(www)			x		
D(d)(ttt)			-x		
Q(r)RFGREN	x**			x	
S(r)RFGOXY	x**				x

(rrr) = ETB, MTB, TAE, TAM, THE, THM; (sss) = ETB, TAE, THE;
(ttt) = N6B, N6I; (www) = N67, N68; (xxx) = E85, RFG, RFH, TRH, TRG;
(yyy)=SSR when (xxx)=RFG, RFH and (yyy)=SST when (xxx)=TRH and
(yyy)=SSE when (xxx)=TRG;
*: (xxx) not E85; **: (xxx)=RFG, RFH only;
:(xxx)=TRG, TRH only; *:(xxx)=TRG only
Bounds: none.

stream.f: This subroutine simulates stream transfers. It creates the following submatrix:

	T(r)(ist)(ist)	T(r)(ist)(prd)
B(r)(ist)	+-x	-x
M(r)(prd)		x
A(r)(prd)		x
A@PRDRFN		x

tabread.f: This subroutine prints the data file names, the number of tables and lists the tables names that are read.

transit3.f: This subroutine simulates the transportation network for MRM 3-region

	Y(o)*(crt)(m)(r)	W(d)ETH(m)(r)	W(r)MET(m)(d)	W(s)(prd)(m)(d)	V(xxxxxx)
C(o)(crt)	-x				
C(r)(crt)	x				
OBJ	-x	-x	-x	-x	
TVC(m)CP	x				
TPC(o)*(m)(r)	x				
B(r)ETH		x			
D(d)ETH		-x			
D(r)ETH		x**			
A(r)ETHRFN		x			
B(r)MET			-x		
D(d)MET			x		
M(s)(prd)				-x	
M(d)(prd)				x	
TPL(s)(m)(d)				x****	
TPP(s)(m)(d)				x***	
D(d)(prd)				x	
TVP(m)CP				x	
(xxxxxx)					-x

transit3.f: (continued)

	Y(o)*(crt)(m)(r)	TANSOTOT	TAAMHXZ	TAGTLTOT	WAGTLJ(r)
C(r)GTL					x
CANSOTOT	x	-x	x		
OBJ		-x		-x	-x

* OGSM supply (o) and refinery (r) regions; ** for (r)=Census Division
*** for (prd) different than MET, ETH; **** for (prd) = LPG and PCF;
(s) : refinery (r) and demand (d) regions; (xxxxxx) = TVC5CP, TVPJCP, ...

Bounds: V(xxxxxx)

unfinished.f: This subroutine simulates the unfinished oil process. It creates the following submatrix:

	T(r)UNF(ist)	T@UNFTOT	RHS
B(r)(ist)	x		
F(r)UNF(ist)	-x	x	
OBJ	-x		
A(r)UNF	x		
A@UNFIMP	x		
A(r)UNFIMP	x		
F@TOTCRD		-x	x

utility.f: This subroutine represents the utility. It creates the following submatrix:

	U(r)(uuu)	R(r)KWGPGN	T(r)NGFNGS
U(r)(uuu)	x		-x***
OBJ	-x		
A@KWHRFN	x*	x	
B(r)NGS			x
A(r)NGSRFN			x
A@NGFTOT	x***		
A(r)NGFTOT	x***		
Z(r)NGFSUM	-x***		

*: for (uuu) = KWH ;***: for (uuu) = NGF

utilitye.f: This subroutine is the same as utility.f, except that it is for ERM, and (r)=refinery region G only (i.e., single region).

All the Fortran files are located on the EIA RS-6000 in the directory /default/source/ .

G.2.c Common Blocks

Variables shared by several subroutines are set up in common. There are four files that contain the common blocks used by the program. Some of the files consist of several common blocks.

The following table lists the common block names, gives a brief description and the location of the block.

Common	Description	Location
IPMMREAL	Common for real variables shared by subroutines that generate submatrices	/default/includes/ipmmtest
IPMMINT	Common for integer variables shared by subroutines that generate submatrices	/default/includes/ipmmtest
IPMMCHAR	Common for character variables shared by subroutines that generate submatrices	/default/includes/ipmmtest
LPTAB	Common used for solution retrieval and report writing	/refine/pmm_lp_gen/includes/lpout
OMLREAL	Common for real variables used to set LP mrmory size	/default/includes/omlspace
OMLINT	Common for integer variables used to set LP mrmory size	/default/includes/omlspace
DFINC2	Common for OML database functions	/default/includes/dfinc2
WCR	Common for the WHIZ optimizer	/default/includes/wfinc2

A list of the common blocks and the variables that constitute them is given in Appendix G-D.

G.3 Data

Most of the data that the program uses is provided in files with a .dat extension. There is a one to one correspondence between the Fortran files that form part of the matrix and the .dat data files (ex accunit.f gets its input from accunit.dat) . In each of the .dat files, the data is arranged in an OML format that consists of tables. Each table consists of a table name; row or stub and column or head names; and values at the intersection of rows and columns. In addition there are ASCII files. These ASCII files are for control of the program. The following is the description of each input file.

G.3.a Data sets

.dat files

accunit.dat

Table Name	Columns	Rows	Description
ACUCUTS	(crt)	(ist); FUL	Crude distillation yield
ACUPOL	OVC, LOS	(crt)	ACU policy table
ACUUTI	(uuu)*	(crt)	ACU utility consumption
INVLIM	MAX	(r) , @	Maximum investment

* except NGF

akaexp.dat

Table name	Columns	Rows	Description
EXPAKA	P, Q	N1, N2, N3, P4, P5, P6	Price and quantity of Alaskan crude exports.
NGLAKA	PER	PGS, CC3, NC4, IC4, and NAT	Yield of NGL
PRQAKA	VOL, TRP, EXPPRC	A	Volume, and transportation and expected cost for Alaskan crude exports

avoids.dat (no longer used)

Table	Columns	Rows	Description
SADELPX	FACTORS	N1, N2, N3, P1, P2, P3	Price differentiate
PRDAVOID	DUMMY	(prd)	Product list
SADELQ	N1, N2, N3, P1, P2, P3	(prd)	Demand shift quality fraction

cogener.dat

Table	Columns	Rows	Description
CGNCAP	CAP, PUL, BLD	(r)	Refinery cogeneration capacity, %utilization and build
CGNINV	INV, FXOC, CAPREC	(r)	Refinery cogeneration investment, fixed cost and capital recovery
CGNPOL	OVC	CGN	Refinery cogeneration policy
CGNREP	CGN	FUL	Refinery cogeneration yields
CGNUTI	(uuu)*	(r)	Refineiry cogeneration utility usage
SELCGN	SOLD	(r)	% cogeneration sold to grid from Refinery
VEPLAS	(r)	(year)	Electric utililty prices for Refinery cogen (87$/kwh)
CGXCAP	CAP, PUL, BLD	(r)	Merchant plant cogeneration capacity, %utilization and build
CGXINV	INV, FXOC, CAPREC	(r)	Merchant plant cogeneration investment, fixed cost and capital recovery
CGXPOL	OVC	CGN	Merchant plant cogeneration policy
CGXREP	CGX	FUL	Merchant plant cogeneration yields
CGXUTI	(uuu)*	(r)	Merchant plant cogeneration utility usage
SELCGX	SOLD	(r)	% cogeneration sold to grid from Merchant plant
VEPLWS	(r)	(year)	Electric utililty prices for Merchant cogen (87$/kwh)

* except NGF

cogenere.dat

Table	Columns	Rows	Description
CGNCAP	CAP, PUL, BLD	G	Cogeneration capacity, %utilization and build
CGNINV	INV, FXOC, CAPREC	G	Cogeneration investment, fixed cost and capital recovery
CGNPOL	OVC	CGN	Cogeneration policy
CGNREP	CGN	FUL	Cogeneration yields
CGNUTI	(uuu)*	G	Cogeneration utility usage
SELGEN	SOLD	G	% cogeneration sold to grid

* except NGF

crdimprt.dat

Table	Columns	Rows	Description
CRUDETYPE	DUMMY	(crt)*	Foreign crude types
ICR(crt)*(r)	C1, Q1, C2, Q2, C3, Q3	(year)	Price and quantity available for crude imports.

*: for (crt) = FLL, FMH, FHL, FHH, FHV (i.e. foreign crudes)

demand.dat

Table	Columns	Rows	Description
CKSMIX	CKL, CKH	OBJ, CKL, CKH, COK	Coke price and conversion factor
PRODLIST	DUMMY	(prd)	List of products
(prd)*	(d)	(year)	Product (prd) demand
DEMMET	CHEM	(year)	MET demand by Chemical Industry

* RFH mapping and corresponding table renamed to RFHA due to duplicate table name elsewhere.

distblnd.dat

Table	Columns	Rows	Description
Q(r)DFO	(dfo)	(spec)	Distillate fuel oil blend specs
DFOUTI	STM	(dfo)	Distillate blend steam use.
DCC	(dfo)*	(ist)	Distillate recipe blend.
DCB	spec categories **	(ist)	Distillate blend intermediate stream quality specification.

* from Z:MAPDFOPD
** from Z:MAPDFOSP

distress.dat

Table	Columns	Rows	Description
ZPX	VALUE	(prd)	Distress code for pricing

domcrude.dat

Table	Columns	Rows	Description
DCRSUP	(o)	Y96	Historical crude supplies by OGSM region
DCRSHR	(o)	(crt)*	Domestic crude share by OGSM region

Table	Columns	Rows	Description
CREXP	CRDEXP, CRDSPR	VOL	Crude exports and SPR

*: (crt) except FLL, FMH, FHL, FHH, FHV

emish.dat

Table	Columns	Rows	Description
EMUNS	(emu)*	Process unit	Emission by process unit
EMFUM	(emu)	Fuel stream burned	Emission by fuel burned

* except CO2

ermcrude.dat

Table	Columns	Rows	Description
CRUDEG	CST, MIN, MAX	(crt)	Crude cost and volume limits

ermother.dat

Table	Columns	Rows	Description
OTHERG	CST, MIN, MAX	(ncr)	Non crude cost and volume limits

ermprod.dat

Table	Columns	Rows	Description
PRODUCTG	REV, MIN, MAX	(prd)	Product revenue and volume limits

ethanol.dat

Table	Columns	Rows	Description
SUPETH(d)	C1, R1, C2, R2, C3, R3, C4, R4	(year)	Ethanol supply curves
ETHTAX	TAXETH, TAXE85	(year)	Ethanol taxes; tax subsidy

fixcols.dat (no longer used)

Table	Columns	Rows	Description
FIXCOL	R	Dummy	First letter of column to fix
(r)RCOL	FCC	(mod)	Column to fix to zero

fuelmix.dat

Table	Columns	Rows	Description
GROUP	DUMMY	Fuel stream	List of fuel stream

gasoblnd.dat

Table	Columns	Rows	Description
Z:MAPGSLPD	ENSYS CODE	TRG, RFG	Map Ensys mogas ID to EIA mogas ID
Z:MAPGSLSP	ENSYS CODE	EIA codes	Map Ensys spec ID to EIA spec ID
Q(r)GSL	TRG, RFG	(spec)	TRG, RFG specs
(r)SSR	Y1, Y2, Y3, Y4, Y5	RFG(spec); YEAR	Gasoline specs for subspec SSR ; year of data
(r)SST	Y1, Y2, Y3, Y4, Y5	TRG(spec); YEAR	Gasoline specs for subspec SST; year of data
(r)SSE	Y1, Y2, Y3, Y4, Y5	TRG(spec); YEAR	Gasoline specs for subspec SSE; year of data
(r)RFH	Y1, Y2, Y3, Y4, Y5	RFG(spec); YEAR	Gasoline spec for RFH; year of data
(r)TRH	Y1, Y2, Y3, Y4, Y5	TRG(spec); YEAR	Gasoline spec for TRH; year of data
Z:GASGROUP	TEXT(1)	(ist)	List of blending streams specially grouped
GCB	Quality codes	(ist)	Gasoline blend intermediate stream quality specification.
GCC	Gasoline type	(ist)	Gasoline recipe blend.
MCO	Motor octane codes*	(ist)	Gasoline component base octane ratings
(xxx)BV	Motor octane codes**	(ist)	Gasoline component blending values
GSLUTI	KWH	(prd), SSE, SST, SSR	Gasoline utility use.
GSPETH	RE	RFGN	Gasoline specs for ETH

(spec) = 2 character quality code followed by X (maximum) or N (minimum).
(xxx) = many exist, however, PMM uses UNC and RFM defined by Z:MAPGSLPD (representing TRG and RFG, respectively).
* R00, R05, R15, R30, M00, M05, M15, M30, of which only R00 and M00 are used by the PMM.
** same as *, except column TEL added (but not used by the PMM).

limpol.dat

Table	Columns	Rows	Description
UNITPOL	DUMMY	(uns)	List of processes that have a limit on POL
LIM(uns)(r)	(ist)	DUM	Limit on (ist)

refmain.dat

Table	Columns	Rows	Description
EXPROD	DUMMY	(prx)	List of product exports
INVFACT	LOC, ENV	(r)	Location and environment factors
TRSOVC	OVC	(r)	Year $ conversion factor for operating cost
FORCRD	DUMMY	(crt)*	List of foreign crudes
YRDOLLAR	2000	1987	Year $ conversion factor
ZIRACFAC	DELTA	ZIRAC	Range of price differential for IRAC
WOP	WOP	(year)	World oil price (87$)
RFNREG	PAD	(r)	List of refinery regions vs PADD
RFNEXP	RFID	Linked list of refinery and export regions	List of exporting regions
DEMNDREG	REGION	Linked list of refinery and demand regions	List of demand region
USERYEAR	YEAR	Y96	Year to run model

* FFL, FMH, FHL, FHH, FHV

maine.dat

Table	Columns	Rows	Description
EXPROD	DUMMY	(prx)	List of product exports
INVFACT	LOC, ENV	G	Location and environment factors
TRSOVC	OVC	G	Conversion factor for operating cost
FORCRD	DUMMY	(crt)*	List of foreign crudes
YRDOLLAR	1991	1987	Conversion factor
ZIRACFAC	Delta	ZIRAC	Range of price differential for IRAC
WOP	WOP	(year)	World oil price
RFNREG	REFINERY	G	List of PADDs
DEMNDREG	REGION	GG	List of demand region
USERYEAR	YEAR	YR95	Year to run model

* FFL, FMH, FHL, FHH, FHV

mchproc.dat

Table	Columns	Rows	Description
MCHINV	INV, FXOC, CAPREC	(uns)	Merchant plant investment, fixed cost and capital recovering
(r)CAPMCH	CAP, PUL, BLD	(uns)	Merchant plant processes capacity, % utilization and build.
(uns)POL*	(pol)	(ist)	Merchant plant processes policy
(uns)CAP*	(uns)CAP	(ist)	Merchant plant process capacity
(uns)REP*	(mod)	(ist)	Merchant plant process yields
(uns)UTI*	(uuu)	(ist)	Merchant plant process utility usage
TRANSFER	Dummy	GP, MP, RF	Transfer allowed
RFTRANS	MP	(ist)	Refinery transfer to merchant plant
GPTRANS	MP	(ist)	Gas plant transfer to merchant plant
MPTRANS	GP, RF, GTLRF	(ist)	Merchant plant transfer to gas plant and refinery
(r)UAP	CST	(uuu)	Utility purchases

* uns defined by T:MCHINV

ngprod.dat

Table	Columns	Rows	Description
SPNGF	ALLREG	N1, .., N4, P5, ..., P8	Price steps for gas supply
SQNGF	MAX, MIN	N1, .., N4, P5, ..., P8	Quantity steps for gas supply
SCVAL	(r)	VOL	Volume limits on each step

nrfplant.dat

Table	Columns	Rows	Description
INVMOH	INV, CAPREC, FXOC	MOH	Non refinery plant process investment, capital recovery, and fixed cost.
MOHPLT	(r)01	CC1, MET, OVC, KWH	Production of methanol
MOHCAP	(r)01	CAP	Methanol capacity
GASPLT	(r)01	(ist), (pol)	Yield from gas plant
GASSHFT	SC2, SC3	CC1, LOS, OBJ	Shift from ethane and propane to methane
GASCAP	(r)01	FAC, CAP, LIM, PCU	Gas plant capacity limits

Table	Columns	Rows	Description
CC1CAP	(r)01	(year)	Dry gas production capacity

prdexp.dat

Table	Columns	Rows	Description
(x)PRDEXP	MINY1, MAXY1, MINY2, MAXY2, ... MINY5, MAXY5	(prx), YEAR	Limit on volume to export; year of data
EXPLIM	YRPC, FIX	1995	Limit on volume and yearly increase
MULTEXPR	MULT	PRICE	Price for exports as function of imports

(x) = export regions (CDs 2,3,7,8,9).

prdimprt.dat

Table	Columns	Rows	Description
PRODTYP	DUMMY	(pri)	List of product import
IMPLIM	MAX	@	Maximum imports into USA
IPR(pri)(r)	C1, R1, ..., C3, R3	(year)	Product import supply curve
NEMSRSD	R1B, R1PR	R1, ..., R9	Resid import supply curve

recipes.dat

Table	Columns	Rows	Description
RCPEIA	A, CST, JTA, N2H, SLP, CKH, CKL	KERSPG, SULSAL, CKHCOK, CKLCOK	Cost of sulfur and coke; kerosene split; unit conversions.
RCP	A, CST, component stream, STM	Recipe blended products	Recipe blends (fractions)

refproc.dat

Table	Columns	Rows	Description
(r)CAP	CAP, PUL, BLD	(uns)	CAP, PUL and BLD values
(uns)	(mod)	(ist), (uuu), CAP, (pol)	Refinery process yields, utility usage, capacity factor, policy
MATBAL	A, B	(ist)	Streams requiring material balance
INV	INV, FXOC, CAPREC	(uns)	Refinery processes investment, fixed cost and capital recovery
SCL	selected processing units	selected streams/utilities	selected factors applied to selected coeff in T: (uns)

setrows.dat

Table	Columns	Rows	Description
(r)POL	TYPE	(pol)	Row type

splash.dat

Table	Columns	Rows	Description
HOXETH	TRH, RFH, RFG,TRG	Gasoline stream	Ethanol recipe for splash blending
BLNSP(d)	KER, N67, N68	JTA, N2H, KER, N6I, N6B, N67, N68	Blend composition recipe
BLOX(d)YXX	E85, M85, TRH, RFH, RFG, TRG	Gasoline stream	Recipe blend composition
XETH	PO	XETH	Oxygen content of ethanol
SCB	PO	Oxygenate stream	Oxygen content of oxygenates

stream.dat

Table	Columns	Rows	Description
XSALE	DUMMY	(ist)(prd)	Linked list of stream transfer to products
TRS	MIN, MAX, CST	(ist)(ist)	Linked list of stream to stream transfers

transit.dat

Table	Columns	Rows	Description
MVCCAP	MAX	TVC(m)CP, TVP(m)CP	Marine vessel capacity for crude & product
BVPCAP	MAX	TVP(m)CP	Marine barge capacity for product
PLCCAP	MAX	TPC(o)(m)(r) TPP(r)**(m)(d) TPL(r)**(m)(d)	Pipeline capacity for (C) crude (P) product (L) LPG
TPCRLIST	DUMMY	(o)	Domestic crude supply regions for transportation
TPCR(o)	(crt); GTL	(m)(r); TAPS	Crude oil transportation cost from domestic supply region (o) to refinery region (r)
PLCRLIST	DUMMY	(o)	List of domestic crude oil supply regions for pipeline
PLCR(o)	(crt)	(m)(r)	Crude pipeline transportation cost from (o) to (r)
BVPR(r)	(prd)	(m)(d)	Product barge/truck transportation cost from (r) to (d)

Table	Columns	Rows	Description
TPPR(r)	(prd)	(m)(d)	Product marine transportation cost from (r) to (d)
TPME(r)	MET	(m)(d)	Methanol transportation cost from (r) to (d)
TPETLIST	DUMMY	(d)	List of Census Divisions for ethanol transportation source
TPET(d)	ETH	(m)(r)**	Ethanol transportation cost from (d) to (r)
PLPRLIST	DUMMY	(d)	List linked census divisions for product pipeline from region B to East Coast
PLPR(r)**	(prd), SSE, SST, SSR	(m)(d)	Product pipeline transportation cost from (r)** to (d)
PLLB(r)	LPG, PCF	(m)(d)	LPG transportation cost from (r) to (d)
PLNKLIST	DUMMY	(r)**	List of product pipeline originations.
PLNK(r)**	(prd), SSE, SST, SSR	(m)(r)**	Product pipeline transport connections and costs from (r)** to (r)**

** refinery (r) and/or demand (d) regions

unfinish.dat

Table	Columns	Rows	Description
UNFOIL	E, B, PD	streams: NPP, HGM, ARB	Types (3) of unfinished oil imports into the U.S.
UNFEQT	SLOPE, CONST	XYZ	eq. parameters that correlate unfinished oil imports to crude input

utility.dat

Table	Columns	Rows	Description
UTITRS	COEF	NGFNGS	BFOE natural gas.
(r)UAP	CST	(uuu)	Utility costs.
VALPNG	(r)	(year)	Industrial price of natural gas.
VPELIN	(r)	(year)	Industrial elec utility prices (87$/kwh)

utilitye.dat

Table	Columns	Rows	Description
UTITRS	COEF	NGFNGS	BFOE natural gas.
GUAP	CST	(uuu)	Utility costs.

G.3.b Other input files

1. **mrmparam** file

The mrmparam file[1] is a control file read by main.f to map input and output file names and instructions. In the mrmparam, the user chooses the model that he wants to run; the names for the actproblem, solution, path file, basis, mps file, and packed matrix; the model title; and the location and name of the starting basis and optimal basis. The following table lists the information that is to be supplied in the mrmparam file.

Variable name	Variable length	Variable purpose	Restrictions
MODELN	8	Model to be run	MRM3, ERM
ACTPROB	8	Act problem	
SOLNAME	8	Solution name	
PATHNF	7	Name of file where data files paths are stored.	
TITLE	40	Problem title	
INBASISN*	40	Location and name of advanced basis	
OUTBASIS	8	Name of optimal basis	
BASISN	8	Basis name	
MPSOUTN**	8	MPS file name	
PAKCN**	8	Packed matrix file name	.PCK extension

*: If no name or a file does not exist, the program will operate without an advanced basis.
**: if “NULL” or “null” is provided, the program will skip the part of the program that generates the file.

The above variables have to be provided in the order they are listed in the above Table and should start at column 18. Appendix G-A provides an example of an mrmparam file. The mrmparam file has to be in the subdirectory where the model is executed.

2. **mrmpath** file:

In this file the user provides the program with the location and name of the data files. The order in which the names appear is important. See Appendix G-B for an example of a path file. The path file has to be in the directory from which the model is executed. This file format is that of an OML table.

[1]Note that the default file used for the MRM is named “mrmparam,” while that used for the ERM is named “ermparam.” Before an ERM execution can be run, the ermparam name must be renamed to mrmparam.

3. **Advanced basis** file:

In this file the user provided an advanced basis to the model. If the user does not provide one or provides one whose name does not coincide with the basis name provided in the mrmparam, the program will not use it.

G.4 Submission of a Run

In order to run the model, one has to first compile and link the different FORTRAN source files to form an executable. Once the executable is created, the user submits a run in conjunction with an mrmparam and a path file. The execution of the program will solve the problem and create:

- An ACTFILE file
- An MPS file
- A SYSPRINT file (solution)
- An out basis file
- A packed matrix file
- Reports (only for MRM5)

The following are the files created by a run:

	ERM	**MRM**
ACTFILE	ACTFILE.act	ACTFILE.act
MPS file	Name provided by user in the mrmparam file	Name provided by user in the mrmparam file
Solution file	SYSPRINT	SYSPRINT
Out basis	Name provided by user in the mrmparam file	Name provided by user in the mrmparam file
Packed matrix	Name provided by user in the mrmparam file	Name provided by user in the mrmparam file
Report	None	reports5, fort.75

All the files used for MRM and ERM matrix generation reside within the NEMS default directories on EIA's NEMS NT servers, as defined next. The source files that encompass the program are on n:/default/source/. The user must create and link the object files to form the mrm executable to be located in the user's directory. The *Developer's Studio* (a debugging package for the PC) is used by the user to compile, link, execute, debug, and manage files. Instructions on how to use this package, in connection with the mrm, is defined below.

The mrmparam file and the mrmpath file used for the MRM model are located in the n:/default/scripts/ directory. The ermparam file and the ermpath file used for the ERM model are located in the n:/default/input/ directory. To run each of the models, their respective param and path files must be copied to the user's Debug directory. (Also remember that the ermparam file must be renamed to mrmparam in the user Debug directory prior to model execution.) The path files point to the default data files (*.dat) that are stored in n:/default/input/ directory.

Runs

First, connect to the NEMS-F5 via a terminal server client, and open a korn shell. You are now operating within a unix environment. All runs will be made using the *Developer's Studio*.

1. Set up and run the default (no changes):
    1. Create a scenario directory in your user directory, then create the following directories within the scenario directory: Source, Data, Debug
    2. In the Debug directory, *nemsco* the default **mrmparam** and **mrmpath** files. (If running ERM, *nemsco* the **ermparam** file and rename to **mrmparam**.)
    3. In the Debug directory, copy the following 4 files from the n:/default/oml/DLL directory: OMLDB32.DLL, OMLLIB32.DLL, OMLLP32.DLL, and OMLWIN32.DLL (or *32.DLL).
    4. In the scenario directory, *nemsco* mrmmps.dsp (a *Developer's Studio* project file).
    5. Double-click the *Developer's Studio* icon on your NEMS-F5 terminal server desktop.
       -- from the FILE, OPEN menu, set file type to *.dsp, and select the mrmmps.dsp file. (This creates two more files in the scenario directory: mrmmps.dsw and mrmmps.opt.)
       -- from the PROJECT, SETTINGS menu, click on the Debug tab. Highlight the path defined in the "Executable for debug session" display, and copy it to the "Working directory" display just below. HOWEVER, go to the "Working directory" display and delete "\mrmmps.exe" portion of the path. Click OK.
    6. To compile and link all default mrm code and DLLs, go to the BUILD menu, and select "Rebuild All." A message will appear in the lower window that the executible has been created.
    7. To execute the default, go the the BUILD menu, and select "Execute." A pop-up window will show the progress of the mrm run. When finished, the pop-up window will instruct the user to press any key to continue.
    8. The resulting mps file will be called mrm3mps (as defined in the mrmparam) and will be located in the Debug directory.

2. Make changes to a mrm data file and rerun mrm:
    1. In the Data directory, *nemsco* the data file you want to change.
    2. Edit, change, and resave.
    3. In the Debug directory, edit the **mrmpath** file.
       -- modify the path of the updated data file to point to the new location in the Data directory, and resave
       -- delete (remove) the following two files that were created from a previous mrm run: ACTFILE.act and SYSPRINT
    4. Double-click the *Developer's Studio* icon on your NEMS-F5 terminal server desktop.
       -- from the FILE, OPEN menu, set file type to *.dsw, and select the mrmmps.dsw file from your scenario directory (created in step 1.5 above)
    5. To execute [assuming source code has already been compiled], go the the BUILD menu, and select "Execute." A pop-up window will show the progress of the mrm run. When finished, the pop-up window will instruct the user to press any key to continue.
    6. The resulting mps file will be called mrm3mps (as defined in the mrmparam) and will be located in the Debug directory.
    7. REMEMBER: After a new data file is defaulted, the corresponding data file name in the **mrmpath** file must be updated to map the proper default version. Then, the **mrmpath** file must be defaulted.

3. Make changes to a mrm source code and rerun mrm:
    1. In the Source directory, *nemsco* the source code you want to change.
    2. Edit, change, and resave.
    3. In the Debug directory, delete (remove) the following two files that were created from a previous mrm run: ACTFILE.act and SYSPRINT
    4. Double-click the *Developer's Studio* icon on your NEMS-F5 terminal server desktop.
       -- from the FILE, OPEN menu, set file type to *.dsw, and select the mrmmps.dsw file from your scenario directory (created in step 1.5 above)
    5. [Need to do only once.] In the workspace window, double-click on the **mrmmps files** listing, and then on the **Source Files** listing.
       -- find the source code filename, right-click on the filename, and select **Properties** from this sub-menu: a pop-up window entitled "Source File Properties" appears
       -- within the pop-up window, modify the path of the source code (in the "Persist as" line) to point to the new location in the source directory
       -- close the pop-up window by clicking on the 'x' in this window

6. To recompile updated code and link to other code, go to the BUILD menu, and select "Rebuild All." A message will appear in the lower window that the executible has been created.
7. To execute, go the the BUILD menu, and select "Execute." A pop-up window will show the progress of the mrm run. When finished, the pop-up window will instruct the user to press any key to continue.
8. The resulting mps file will be called mrm3mps (as defined in the mrmparam) and will be located in the Debug directory.

Appendix G-A: Example of an mrmparam file.

mrmparam:

```
MODELN    'MRM3    '                                                !8
ACTPROB       'OMLMRM3 '                                           !8
SOLNAM    'MRM3SOL '                                               !8
PATHNF    'mrmpath'                                                !7
TITLE     'MRM3 MULTI  REGION  REFINERY                    '   !48
INBASISN  'inbasis3                                        '   !48
OUTBASISN     'outbasis3'                                          !9
BASISN    'MRM3     '                                              !9
MPSOUTN       'mrm3mps  '                                          !9
PACKN     'NULL     '                                              !9
```

ermparam:

```
MODELN    'ERM     '                                               !8
ACTPROB       'OMLERM  '                                           !8
SOLNAM    'ERMSOL  '                                               !8
PATHNF    'pathe  '                                                !7
TITLE     'ERM  SINGLE REGION  REFINERY                    '   !48
INBASISN  '/refine/pmm_lp_gen/data/erm/inbasise            '   !48
OUTBASISN     'outbasise'                                          !9
BASISN    'ERM      '                                              !9
MPSOUTN       'ermGmps  '                                          !9
PACKN     'ERM2.PCK '                                              !9
```

The mrmparam file will run the 3-region MRM model and the ermparam file will run the single region ERM model; the act problem will be “OMLMRM3" for the MRM model and “OMLERM “for the ERM model; the solution name will be “MRM3SOL” for the MRM model and “ERMSOL” for the ERM model; the data files path will be read from the file “mrmpath" for the MRM model and “pathe” for the ERM model; the title of the MRM model will be “MRM3, MULTIPLE REGIONS REFINERY” and the ERM model will be “ERM, SINGLE REGION REFINERY;” the advanced basis will be read from “inbasis3" for the MRM model and “inbasise” for the ERM model along the path defined; the optimal basis will be stored in the file “outbasis3" for the MRM model and “outbasise” for the ERM model; both the advanced and optimal basis will be named “MRM3" for the MRM model and “ERM for the ERM model; the mps file will be stored in “mrm3mps” for the MRM model and “ermGmps” for the ERM model; and the packed matrix will not be created (NULL) for the MRM model, but will be stored in “ERM2.PCK” for the ERM model.

Remember that the “ermparam” file must be **renamed** to “mrmparam” before an ERM execution can be run.

Appendix G-B: Example of a path file

mrmpath:

```
NAME          PATHDATA
         DATA    Z:PATHNAME
**        TEXT(6)
accunit     n:/default/input/accunit.dat.v1.6
avoids      n:/default/input/avoids.dat.v1.1
cogener     n:/default/input/cogener.dat.v1.7
crdimprt    n:/default/input/crdimprt.dat.v1.4
demand      n:/default/input/demand.dat.v1.4
distblnd    n:/default/input/distblnd.dat.v1.3
distress    n:/default/input/distress.dat.v1.2
domcrude    n:/default/input/domcrude.dat.v1.2
emish       n:/default/input/emish.dat.v1.1
ethanol     n:/default/input/ethanol.dat.v1.4
fixcols     n:/default/input/fixcols.dat.v1.3
fuelmix     n:/default/input/fuelmix.dat.v1.2
gasoblnd    n:/default/input/gasoblnd.dat.v1.7
main        n:/default/input/main.dat.v1.4
ngprod      n:/default/input/ngprod.dat.v1.2
nrfplant    n:/default/input/nrfplant.dat.v1.7
prdimprt    n:/default/input/prdimprt.dat.v1.3
recipes     n:/default/input/recipes.dat.v1.2
refproc     n:/default/input/refproc.dat.v1.18
setrows     n:/default/input/setrows.dat.v1.4
splash      n:/default/input/splash.dat.v1.4
stream      n:/default/input/stream.dat.v1.3
transit     n:/default/input/transit.dat.v1.6
utility     n:/default/input/utility.dat.v1.5
mchproc     n:/default/input/mchproc.dat.v1.5
limpol      n:/default/input/limpol.dat.v1.3
unfinished  n:/default/input/unfinish.dat.v1.2
prdexp      n:/default/input/prdexp.dat.v1.2
akaexp      n:/default/input/akaexp.dat.v1.1
foreign1
foreign2
fsu
intsetup
epsetup
bldhead
bldstub
ENDATA
```

pathe:

```
NAME          PATHDATA
         DATA    Z:PATHNAME
**        TEXT(6)
accunit     n:/default/input/accunit.dat.v1.4
avoids
cogener     n:/default/input/cogenere.dat.v1.1
crdimprt
ermprod     n:/default/input/ermprod.dat.v1.1
distblnd    n:/default/input/distblnd.dat.v1.2
distress
ermcrude    n:/default/input/ermcrude.dat.v1.1
```

```
emish
ethanol
fixcols
fuelmix     n:/default/input/fuelmix.dat.v1.2
gasoblnd    n:/default/input/gasoblnd.dat.v1.4
main        n:/default/input/maine.dat.v1.1
ngprod
ermother    n:/default/input/ermother.dat.v1.1
prdimprt
recipes     n:/default/input/recipes.dat.v1.2
refproc     n:/default/input/refproc.dat.v1.14
setrows     n:/default/input/setrows.dat.v1.3
splash
stream      n:/default/input/stream.dat.v1.3
transit
utility     n:/default/input/utilitye.dat.v1.1
mchproc
limpol      n:/default/input/limpol.dat.v1.2
unfinished
prdexp
akaexp
foreign1
foreign2
fsu
intsetup
epsetup
bldhead
bldstub
ENDATA
```

Appendix G-C:Makefile
(not part of the Developer's Studio Methodology)

```
FLAGS= -c -g -C -qcharlen=16384 -qmaxmem=-1 -qnoprint \
      -qflttrap=zerodivide -qsave
INC= -I/refine/pmm_lp_gen/includes/ -I/default/includes
LIBS= -lwhiz -ldb -lutil -L/usr/lpp/MPS/oml/lib
DIR= /default/objects/
SIR= /default/source/
LIR= /default/objects/
OMLSPA = /refine/pmm_lp_gen/includes/omlspace \
       /refine/pmm_lp_gen/includes/ipmmtest
OBJS= $(LIR)mrm.v1.2.o \
      $(LIR)extpnt.v1.2.o \
      $(LIR)epadjust.v1.2.o \
      $(LIR)bldhead.v1.2.o \
      $(LIR)bldstub.v1.2.o \
      $(LIR)intsetup.v1.2.o \
      $(LIR)foreign.v1.2.o \
      $(LIR)fsu.v1.2.o \
      $(LIR)akaexp.v1.2.o \
      $(LIR)prdexp.v1.4.o \
      $(LIR)unfinished.v1.2.o \
      $(LIR)lplookup.v1.2.o \
      $(LIR)output.v1.2.o \
      $(LIR)ermcrude.v1.2.o \
      $(LIR)ermother.v1.2.o \
      $(LIR)ermprod.v1.2.o \
      $(LIR)tabread.v1.2.o \
      $(LIR)limpol.v1.2.o \
      $(LIR)mchproc.v1.4.o \
      $(LIR)crdimprt.v1.2.o \
      $(LIR)accunit.v1.6.o \
      $(LIR)refproc.v1.6.o \
      $(LIR)nrfplant.v1.5.o \
      $(LIR)stream.v1.2.o \
      $(LIR)distblnd.v1.3.o \
      $(LIR)gasoblnd.v1.4.o \
      $(LIR)domcrude.v1.3.o \
      $(LIR)demand.v1.3.o \
      $(LIR)utility.v1.3.o \
      $(LIR)utilitye.v1.2.o \
      $(LIR)cogener.v1.4.o \
      $(LIR)fuelmix.v1.2.o \
      $(LIR)recipes.v1.2.o \
      $(LIR)ethanol.v1.9.o \
      $(LIR)ngprod.v1.2.o \
      $(LIR)setrows.v1.2.o \
      $(LIR)fixcols.v1.2.o \
      $(LIR)transit3.v1.4.o \
      $(LIR)transit5.v1.2.o \
      $(LIR)prdimprt.v1.2.o \
      $(LIR)splash.v1.3.o \
      $(LIR)distress.v1.2.o \
      $(LIR)avoids.v1.2.o \
      $(DIR)omlanal.v1.6.o \
      $(LIR)emish.v1.2.o \

mrm: $(OBJS)
      xlf -o mrm -g -C $(OBJS) $(LIBS) -bloadmap:loadmap
```

```
$(LIR)mrm.v1.3.o: $(SIR)mrm.f $(OMLSPA)
	xlf $(FLAGS) -o $(LIR)mrm.v1.3.o $(INC) $(SIR)mrm.f
$(LIR)extpnt.v1.2.o: $(SIR)extpnt.f
	xlf $(FLAGS) -o $(LIR)extpnt.v1.2.o $(INC) $(SIR)extpnt.f
$(LIR)epadjust.v1.2.o: $(SIR)epadjust.f
	xlf $(FLAGS) -o $(LIR)epadjust.v1.2.o $(INC) $(SIR)epadjust.f
$(LIR)bldhead.v1.2.o: $(SIR)bldhead.f
	xlf $(FLAGS) -o $(LIR)bldhead.v1.2.o $(INC) $(SIR)bldhead.f
$(LIR)bldstub.v1.2.o: $(SIR)bldstub.f
	xlf $(FLAGS) -o $(LIR)bldstub.v1.2.o $(INC) $(SIR)bldstub.f
$(LIR)intsetup.v1.2.o: $(SIR)intsetup.f
	xlf $(FLAGS) -o $(LIR)intsetup.v1.2.o $(INC) $(SIR)intsetup.f
$(LIR)foreign.v1.2.o: $(SIR)foreign.f
	xlf $(FLAGS) -o $(LIR)foreign.v1.2.o $(INC) $(SIR)foreign.f
$(LIR)fsu.v1.2.o: $(SIR)fsu.f
	xlf $(FLAGS) -o $(LIR)fsu.v1.2.o $(INC) $(SIR)fsu.f
$(LIR)akaexp.v1.2.o: $(SIR)akaexp.f
	xlf $(FLAGS) -o $(LIR)akaexp.v1.2.o $(INC) $(SIR)akaexp.f
$(LIR)prdexp.v1.4.o: $(SIR)prdexp.f
	xlf $(FLAGS) -o $(LIR)prdexp.v1.4.o $(INC) $(SIR)prdexp.f
$(LIR)unfinished.v1.2.o: $(SIR)unfinished.f
	xlf $(FLAGS) -o $(LIR)unfinished.v1.2.o $(INC) $(SIR)unfinished.f
$(LIR)lplookup.v1.2.o: $(SIR)lplookup.f
	xlf $(FLAGS) -o $(LIR)lplookup.v1.2.o $(INC) $(SIR)lplookup.f
$(LIR)output.v1.2.o: $(SIR)output.f
	xlf $(FLAGS) -o $(LIR)output.v1.2.o $(INC) $(SIR)output.f
$(LIR)limpol.v1.2.o: $(SIR)limpol.f
	xlf $(FLAGS) -o $(LIR)limpol.v1.2.o $(INC) $(SIR)limpol.f
$(LIR)tabread.v1.2.o: $(SIR)tabread.f
	xlf $(FLAGS) -o $(LIR)tabread.v1.2.o $(INC) $(SIR)tabread.f
$(LIR)mchproc.v1.4.o: $(SIR)mchproc.f
	xlf $(FLAGS) -o $(LIR)mchproc.v1.4.o $(INC) $(SIR)mchproc.f
$(LIR)ermcrude.v1.2.o: $(SIR)ermcrude.f
	xlf $(FLAGS) -o $(LIR)ermcrude.v1.2.o $(INC) $(SIR)ermcrude.f
$(LIR)ermother.v1.2.o: $(SIR)ermother.f
	xlf $(FLAGS) -o $(LIR)ermother.v1.2.o $(INC) $(SIR)ermother.f
$(LIR)ermprod.v1.2.o: $(SIR)ermprod.f
	xlf $(FLAGS) -o $(LIR)ermprod.v1.2.o $(INC) $(SIR)ermprod.f
$(LIR)crdimprt.v1.2.o: $(SIR)crdimprt.f
	xlf $(FLAGS) -o $(LIR)crdimprt.v1.2.o $(INC) $(SIR)crdimprt.f
$(LIR)accunit.v1.6.o: $(SIR)accunit.f
	xlf $(FLAGS) -o $(LIR)accunit.v1.6.o $(INC) $(SIR)accunit.f
$(LIR)refproc.v1.6.o: $(SIR)refproc.f
	xlf $(FLAGS) -o $(LIR)refproc.v1.6.o $(INC) $(SIR)refproc.f
$(LIR)nrfplant.v1.5.o: $(SIR)nrfplant.f
	xlf $(FLAGS) -o $(LIR)nrfplant.v1.5.o $(INC) $(SIR)nrfplant.f
$(LIR)stream.v1.2.o: $(SIR)stream.f
	xlf $(FLAGS) -o $(LIR)stream.v1.2.o $(INC) $(SIR)stream.f
$(LIR)distblnd.v1.3.o: $(SIR)distblnd.f
	xlf $(FLAGS) -o $(LIR)distblnd.v1.3.o $(INC) $(SIR)distblnd.f
$(LIR)gasoblnd.v1.4.o: $(SIR)gasoblnd.f
	xlf $(FLAGS) -o $(LIR)gasoblnd.v1.4.o $(INC) $(SIR)gasoblnd.f
$(LIR)domcrude.v1.3.o: $(SIR)domcrude.f
	xlf $(FLAGS) -o $(LIR)domcrude.v1.3.o $(INC) $(SIR)domcrude.f
$(LIR)demand.v1.3.o: $(SIR)demand.f
	xlf $(FLAGS) -o $(LIR)demand.v1.3.o $(INC) $(SIR)demand.f
$(LIR)utility.v1.3.o: $(SIR)utility.f
	xlf $(FLAGS) -o $(LIR)utility.v1.3.o $(INC) $(SIR)utility.f
$(LIR)utilitye.v1.2.o: $(SIR)utilitye.f
	xlf $(FLAGS) -o $(LIR)utilitye.v1.2.o $(INC) $(SIR)utilitye.f
$(LIR)cogener.v1.4.o: $(SIR)cogener.f
	xlf $(FLAGS) -o $(LIR)cogener.v1.4.o $(INC) $(SIR)cogener.f
$(LIR)fuelmix.v1.2.o: $(SIR)fuelmix.f
	xlf $(FLAGS) -o $(LIR)fuelmix.v1.2.o $(INC) $(SIR)fuelmix.f
$(LIR)recipes.v1.2.o: $(SIR)recipes.f
```

```
	xlf $(FLAGS) -o $(LIR)recipes.v1.2.o $(INC) $(SIR)recipes.f
$(LIR)ethanol.v1.9.o: $(SIR)ethanol.f
	xlf $(FLAGS) -o $(LIR)ethanol.v1.9.o $(INC) $(SIR)ethanol.f
$(LIR)ngprod.v1.2.o: $(SIR)ngprod.f
	xlf $(FLAGS) -o $(LIR)ngprod.v1.2.o $(INC) $(SIR)ngprod.f
$(LIR)setrows.v1.2.o: $(SIR)setrows.f
	xlf $(FLAGS) -o $(LIR)setrows.v1.2.o $(INC) $(SIR)setrows.f
$(LIR)fixcols.v1.2.o: $(SIR)fixcols.f
	xlf $(FLAGS) -o $(LIR)fixcols.v1.2.o $(INC) $(SIR)fixcols.f
$(LIR)transit3.v1.4.o: $(SIR)transit3.f
	xlf $(FLAGS) -o $(LIR)transit3.v1.4.o $(INC) $(SIR)transit3.f
$(LIR)transit5.v1.2.o: $(SIR)transit5.f
	xlf $(FLAGS) -o $(LIR)transit5.v1.2.o $(INC) $(SIR)transit5.f
$(LIR)prdimprt.v1.2.o: $(SIR)prdimprt.f
	xlf $(FLAGS) -o $(LIR)prdimprt.v1.2.o $(INC) $(SIR)prdimprt.f
$(LIR)splash.v1.3.o: $(SIR)splash.f
	xlf $(FLAGS) -o $(LIR)splash.v1.3.o $(INC) $(SIR)splash.f
$(LIR)distress.v1.2.o: $(SIR)distress.f
	xlf $(FLAGS) -o $(LIR)distress.v1.2.o $(INC) $(SIR)distress.f
$(LIR)avoids.v1.2.o: $(SIR)avoids.f
	xlf $(FLAGS) -o $(LIR)avoids.v1.2.o $(INC) $(SIR)avoids.f
$(LIR)emish.v1.2.o  : $(SIR)emish.f
	xlf $(FLAGS) -o $(LIR)emish.v1.2.o $(INC) $(SIR)emish.f
```

Appendix G-D: Common Blocks

File: ipmmtest

```
!  $Header: N:/default/includes/RCS/ipmmtest,v 1.3 1999/12/21 14:13:29 DSA Exp $
     COMMON /IPMMREAL/MAX, MIN, VALUE, INFINITY,                    &
           WOP,                                  &
           REALURYR, YRDOLLAR, ENV, LOC, OVCOST
     REAL*8          MAX(1000)
     REAL*8          MIN(1000)
     REAL*8          VALUE(1000)
     REAL*8          INFINITY
     REAL*8          WOP(26)
     REAL*8          REALURYR
     REAL*8          YRDOLLAR
     REAL*8          ENV(5)
     REAL*8          LOC(5)
     REAL*8          OVCOST(5)
     COMMON /IPMMINT/COUNT, DUMMY,                                  &
           I, IRET, IROWS, J, JCOLS, K, KROWS, NUMREG,EXPREGSS,     &
           USERYEAR, DMDREG, NUMYRS, DBPROBG, COMTAB1, COMTAB2
     INTEGER*4       COUNT
     INTEGER*4       DUMMY
     INTEGER*4       I
     INTEGER*4       IRET
     INTEGER*4       IROWS
     INTEGER*4       J
     INTEGER*4       JCOLS
     INTEGER*4       K
     INTEGER*4       KROWS
     INTEGER*4       NUMREG
     INTEGER*4       EXPREGSS
     INTEGER*4       USERYEAR
     INTEGER*4       DMDREG
     INTEGER*4       NUMYRS
     INTEGER*4       DBPROBG(22)
     INTEGER*4       COMTAB1(13)
     INTEGER*4       COMTAB2(13)
     COMMON /IPMMCHAR/HEAD, STUB, PROCESS, BND, CNAME,              &
       DECKNAME, LRNAME, OBJ, RHS, RNAME, RCHAR5, REGION,           &
       EXPREG, DEMNDREG, DEMNDPAD, PATH, FRCRD, EXPRD
     CHARACTER*8                   HEAD(1000)
     CHARACTER*8                   STUB(1000)
     CHARACTER*8                   PROCESS(1000)
     CHARACTER*8                   BND
     CHARACTER*8                   CNAME
     CHARACTER*8                   DECKNAME
     CHARACTER*8                   LRNAME
     CHARACTER*8                   OBJ
     CHARACTER*8                   RHS
     CHARACTER*8                   RNAME
     CHARACTER*5                   RCHAR5
     CHARACTER*1                   REGION(5)
     CHARACTER*2                   EXPREG(5)
     CHARACTER*1                   DEMNDREG(9)
     CHARACTER*1                   DEMNDPAD(9)
     CHARACTER*48          PATH(40)
     CHARACTER*8                   FRCRD(5)
     CHARACTER*8                   EXPRD(11)
```

File: lpout

Common block used for report writing

```
C  LPOUT COMMON BLOCK FOR USE IN OML MATRIX SOLUTION
                                !
      REAL        PRDPRC(9,18)
      REAL        BASECAP(5,27)
      REAL        BASEUTL(5,27)
      REAL        INVCAP(5,27)
      REAL        INVUTL(5,27)
      REAL        TOTCAP(5,27)
      REAL        TOTUTL(5,27)
      REAL        TOTSUP(6)
      REAL        PUTL(5,27)
      REAL        RFCRDCR(6)
      REAL        RFCRDAKA(6)
      REAL        RFCRDL48(6)
      REAL        RFQICRD(6)
      REAL        RFIMCR(6)
      REAL        RFQISPR(6)
      REAL        RFIMTP(6)
      REAL        RFQEXCRD(6)
      REAL        RFPQNGL(6)
      REAL        RFQNGLRF(6)
      REAL        NGLRF(6)
      REAL        NGLMK(6)
      REAL        RFQPRCG(6)
      REAL        RFPQIPRDT(6)
      REAL        RFPQUFC(6)
      REAL        RFQEXPRDT(6)
      REAL        RFQPRDDM(11)
      REAL        QCDUPD(6)
      REAL        RFQDINPOT(6)
      REAL        RFETHD(6)
      REAL        RFMTBI(6)
      REAL        RFMETI(6)
      REAL        RFMETD(6)
      REAL        RFMETCHM(6)
      REAL        RFMETM85(6)
      REAL        RFMETETH(6)
      REAL        RFETHE85(6)
      REAL        RFETHETB(6)
      REAL        RFETHMGS(6)
      REAL        RFCRDOTH(6)
      REAL        OTHOXY(6)
      REAL        QPRDIMD(11,23)
      REAL        QPRDEXD(6,23)
      COMMON/LPTAB/PRDPRC,BASECAP,BASEUTL,INVCAP,INVUTL,TOTCAP,          &
      TOTUTL,PUTL,RFCRDDCR,RFCrtAKA,RFCrtL48,RFQICRD,RFIMCR,          &
      RFQEXCRD,RFPQNGL,RFQNGLRF,NGLRF,NGLMK,RFQPRCG,RFPQIPRDT,          &
      RFPQUFC,RFQEXPRDT,RFQPRDDM,QCDUPD,RFQDINPOT,RFETHD,RFMTBI,          &
      RFMETI,RFMETD,RFMETCHM,RFMETM85,RFMETETH,RFETHE85,TOTSUP,          &
      RFQISPR,RFIMTP,RFETHETB,RFETHMGS,OTHOXY,QPRDIMD,QPRDEXD,          &
      RFCRDOTH
```

File: DFINC2

Common block used for OML database functions

```
! $Header: N:/default/includes/RCS/dfinc2,v 1.5 2000/05/18 15:12:11 DSA Exp $
      INTERFACE
      INTEGER*4 FUNCTION DFINIT  (TITLE,TBUF,LEN)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFINIT
      !DEC$ATTRIBUTES REFERENCE ::  TITLE,TBUF,LEN
      CHARACTER*8 TITLE
      CHARACTER*1 TBUF(2)
      INTEGER*4 LEN
      END FUNCTION

      INTEGER*4 FUNCTION SetConsolePrintMode(IVAL)
      !DEC$ATTRIBUTES STDCALL, ALIAS: '_SetConsolePrintMode@4' ::  SetConsolePrintMode
      !DEC$ATTRIBUTES VALUE ::  IVAL
      INTEGER*4 IVAL
      END FUNCTION

      INTEGER*4 FUNCTION DFOPEN  (DBFILE,ACTFILE)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFOPEN
      !DEC$ATTRIBUTES REFERENCE ::  DBFILE,ACTFILE
      INTEGER*4 DBFILE(2)
      CHARACTER*8 ACTFILE
      END FUNCTION

      INTEGER*4 FUNCTION DFCLOSE (DBFILE)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFCLOSE
      !DEC$ATTRIBUTES REFERENCE ::  DBFILE
      INTEGER*4 DBFILE(2)
      END FUNCTION

      INTEGER*4 FUNCTION DFDEL   (DBFILE)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFDEL
      !DEC$ATTRIBUTES REFERENCE ::  DBFILE
      INTEGER*4 DBFILE(2)
      END FUNCTION

      INTEGER*4 FUNCTION DFLIST  (DBFILE,MASK,PNAME)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFLIST
      !DEC$ATTRIBUTES REFERENCE ::  DBFILE,MASK,PNAME
      INTEGER*4 DBFILE(2)
      CHARACTER*8 MASK
      CHARACTER*8 PNAME
      END FUNCTION

      INTEGER*4 FUNCTION DFPINIT (DB,DBFILE,ACTPROB)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPINIT
      !DEC$ATTRIBUTES REFERENCE ::  DB,DBFILE,ACTPROB
      INTEGER*4 DB(2)
      INTEGER*4 DBFILE(2)
      CHARACTER*8 ACTPROB
      END FUNCTION

      INTEGER*4 FUNCTION DFPDEL  (DB)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPDEL
      !DEC$ATTRIBUTES REFERENCE ::  DB
      INTEGER*4 DB(2)
      END FUNCTION

      INTEGER*4 FUNCTION DFPCPY  (TO,FROM)
      !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPCPY
```

```
!DEC$ATTRIBUTES REFERENCE ::  TO,FROM
INTEGER*4 TO
INTEGER*4 FROM
END FUNCTION

INTEGER*4 FUNCTION DFPLIST (DB,MASK,TNAME,TYPE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPLIST
!DEC$ATTRIBUTES REFERENCE ::  DB,MASK,TNAME,TYPE
INTEGER*4 DB(2)
CHARACTER*8 MASK
CHARACTER*8 TNAME
INTEGER*4 TYPE
END FUNCTION

INTEGER*4 FUNCTION DFPTDEL (DB)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPTDEL
!DEC$ATTRIBUTES REFERENCE ::  DB
INTEGER*4 DB(2)
END FUNCTION

INTEGER*4 FUNCTION DFPTCPY (TO,FROM)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPTCPY
!DEC$ATTRIBUTES REFERENCE ::  TO,FROM
INTEGER*4 TO
INTEGER*4 FROM
END FUNCTION

INTEGER*4 FUNCTION DFPMDEL (DB)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPMDEL
!DEC$ATTRIBUTES REFERENCE ::  DB
INTEGER*4 DB(2)
END FUNCTION

INTEGER*4 FUNCTION DFPMCPY (TO,FROM)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPMCPY
!DEC$ATTRIBUTES REFERENCE ::  TO,FROM
INTEGER*4 TO
INTEGER*4 FROM
END FUNCTION

INTEGER*4 FUNCTION DFPSDEL (DB)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPSDEL
!DEC$ATTRIBUTES REFERENCE ::  DB
INTEGER*4 DB(2)
END FUNCTION

INTEGER*4 FUNCTION DFPSCPY (TO,FROM)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPSCPY
!DEC$ATTRIBUTES REFERENCE ::  TO,FROM
INTEGER*4 TO
INTEGER*4 FROM
END FUNCTION

INTEGER*4 FUNCTION DFPRTBI (DB,FILENAME,DECKNAME,PRNTSW,TABLIST)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFPRTBI
!DEC$ATTRIBUTES REFERENCE ::  DB,FILENAME,DECKNAME,PRNTSW,TABLIST
INTEGER*4 DB(2)
CHARACTER*8 FILENAME
CHARACTER*8 DECKNAME
INTEGER*4 PRNTSW
CHARACTER*8 TABLIST
END FUNCTION

INTEGER*4 FUNCTION DFTINIT (TABLE,DB,TABNAME,TYPE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTINIT
```

```
!DEC$ATTRIBUTES REFERENCE ::  TABLE,DB,TABNAME,TYPE
INTEGER*4 TABLE(2)
INTEGER*4 DB(2)
CHARACTER*8 TABNAME
INTEGER*4 TYPE
END FUNCTION

INTEGER*4 FUNCTION DFTNEW  (TABLE,DB,TABNAME,TYPE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTNEW
!DEC$ATTRIBUTES REFERENCE ::  TABLE,DB,TABNAME,TYPE
INTEGER*4 TABLE(2)
INTEGER*4 DB(2)
CHARACTER*8 TABNAME
INTEGER*4 TYPE
END FUNCTION

INTEGER*4 FUNCTION DFTDIM  (TABLE,NROWS,NCOLS)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTDIM
!DEC$ATTRIBUTES REFERENCE ::  TABLE,NROWS,NCOLS
INTEGER*4 TABLE(2)
INTEGER*4 NROWS
INTEGER*4 NCOLS
END FUNCTION

INTEGER*4 FUNCTION DFTDEL  (TABLE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTDEL
!DEC$ATTRIBUTES REFERENCE ::  TABLE
INTEGER*4 TABLE(2)
END FUNCTION

INTEGER*4 FUNCTION DFTCPY  (TO,FROM)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTCPY
!DEC$ATTRIBUTES REFERENCE ::  TO,FROM
INTEGER*4 TO
INTEGER*4 FROM
END FUNCTION

INTEGER*4 FUNCTION DFTSNDX (TABLE,STUB)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTSNDX
!DEC$ATTRIBUTES REFERENCE ::  TABLE,STUB
INTEGER*4 TABLE(2)
CHARACTER*8 STUB
END FUNCTION

INTEGER*4 FUNCTION DFTHNDX (TABLE,HEAD)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTHNDX
!DEC$ATTRIBUTES REFERENCE ::  TABLE,HEAD
INTEGER*4 TABLE(2)
CHARACTER*8 HEAD
END FUNCTION

INTEGER*4 FUNCTION DFTSMSK (TABLE,START,MASK)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTSMSK
!DEC$ATTRIBUTES REFERENCE ::  TABLE,START,MASK
INTEGER*4 TABLE(2)
INTEGER*4 START
CHARACTER*8 MASK
END FUNCTION

INTEGER*4 FUNCTION DFTHMSK (TABLE,START,MASK)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTHMSK
!DEC$ATTRIBUTES REFERENCE ::  TABLE,START,MASK
INTEGER*4 TABLE(2)
INTEGER*4 START
CHARACTER*8 MASK
```

```
END FUNCTION

INTEGER*4 FUNCTION DFTNDX  (TABLE,RNAME,CNAME,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTNDX
!DEC$ATTRIBUTES REFERENCE ::  TABLE,RNAME,CNAME,VALUE
INTEGER*4 TABLE(2)
INTEGER*4 RNAME
INTEGER*4 CNAME
INTEGER*4 VALUE(1)
END FUNCTION

INTEGER*4 FUNCTION DFTGETVC (TABLE,RNAME,CNAME,VALUE,COUNT)
!DEC$ATTRIBUTES STDCALL,REFERENCE,ALIAS: '_dftget@20' ::  DFTGETVC
!DEC$ATTRIBUTES REFERENCE ::  TABLE,RNAME,CNAME,VALUE,COUNT
INTEGER*4 TABLE(2)
INTEGER*4 RNAME
CHARACTER*8 CNAME(1)
REAL*8    VALUE(1)
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION DFTGETV  (TABLE,RNAME,CNAME,VALUE,COUNT)
!DEC$ATTRIBUTES STDCALL,REFERENCE,ALIAS: '_dftget@20' ::  DFTGETV
!DEC$ATTRIBUTES REFERENCE ::  TABLE,RNAME,CNAME,VALUE,COUNT
INTEGER*4 TABLE(2)
INTEGER*4 RNAME
INTEGER*4 CNAME
REAL*8    VALUE(1)
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION DFTGETN  (TABLE,RNAME,CNAME,VALUE,COUNT)
!DEC$ATTRIBUTES STDCALL,REFERENCE,ALIAS: '_dftget@20' ::  DFTGETN
!DEC$ATTRIBUTES REFERENCE ::  TABLE,RNAME,CNAME,VALUE,COUNT
INTEGER*4 TABLE(2)
INTEGER*4 RNAME
INTEGER*4 CNAME
CHARACTER*8 VALUE(1)
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION DFTPUT  (TABLE,RNAME,CNAME,VALUE,COUNT)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTPUT
!DEC$ATTRIBUTES REFERENCE ::  TABLE,RNAME,CNAME,VALUE,COUNT
INTEGER*4 TABLE(2)
INTEGER*4 RNAME
INTEGER*4 CNAME
INTEGER*4 VALUE
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION DFTSADD (TABLE,INDEX,STUB)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTSADD
!DEC$ATTRIBUTES REFERENCE ::  TABLE,INDEX,STUB
INTEGER*4 TABLE(2)
INTEGER*4 INDEX
CHARACTER*8 STUB
END FUNCTION

INTEGER*4 FUNCTION DFTHADD (TABLE,INDEX,HEAD)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTHADD
!DEC$ATTRIBUTES REFERENCE ::  TABLE,INDEX,HEAD
INTEGER*4 TABLE(2)
INTEGER*4 INDEX
CHARACTER*8 HEAD
```

```
END FUNCTION

INTEGER*4 FUNCTION DFTSDEL (TABLE,STUB)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTSDEL
!DEC$ATTRIBUTES REFERENCE ::  TABLE,STUB
INTEGER*4 TABLE(2)
INTEGER*4 STUB
END FUNCTION

INTEGER*4 FUNCTION DFTHDEL (TABLE,HEAD)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTHDEL
!DEC$ATTRIBUTES REFERENCE ::  TABLE,HEAD
INTEGER*4 TABLE(2)
INTEGER*4 HEAD
END FUNCTION

INTEGER*4 FUNCTION DFTDISP (TABLE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFTDISP
!DEC$ATTRIBUTES REFERENCE ::  TABLE
INTEGER*4 TABLE(2)
END FUNCTION

INTEGER*4 FUNCTION DFMINIT (DB,MODE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMINIT
!DEC$ATTRIBUTES REFERENCE ::  DB,MODE
INTEGER*4 DB(2)
INTEGER*4 MODE
END FUNCTION

INTEGER*4 FUNCTION DFMEND  ()
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMEND
END FUNCTION

INTEGER*4 FUNCTION DFMSTAT (STAT)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMSTAT
!DEC$ATTRIBUTES REFERENCE ::  STAT
INTEGER*4 STAT(9)
END FUNCTION

INTEGER*4 FUNCTION DFMCRTP (RNAME,TYPE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCRTP
!DEC$ATTRIBUTES REFERENCE ::  RNAME,TYPE
CHARACTER*8 RNAME
CHARACTER*8 TYPE
END FUNCTION

INTEGER*4 FUNCTION DFMCRSC (RNAME,SCALE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCRSC
!DEC$ATTRIBUTES REFERENCE ::  RNAME,SCALE
CHARACTER*8 RNAME
REAL*8 SCALE
END FUNCTION

INTEGER*4 FUNCTION DFMCCSC (CNAME,SCALE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCCSC
!DEC$ATTRIBUTES REFERENCE ::  CNAME,SCALE
CHARACTER*8 CNAME
REAL*8 SCALE
END FUNCTION

INTEGER*4 FUNCTION DFMCVAL (CNAME,RNAME,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCVAL
!DEC$ATTRIBUTES REFERENCE ::  CNAME,RNAME,VALUE
CHARACTER*8 CNAME
CHARACTER*8 RNAME
```

```
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION DFMMVAL (CNAME,RNAME,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMMVAL
!DEC$ATTRIBUTES REFERENCE ::  CNAME,RNAME,VALUE
CHARACTER*8 CNAME
CHARACTER*8 RNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION DFMCCLR (CNAME,FLAG)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCCLR
!DEC$ATTRIBUTES REFERENCE ::  CNAME,FLAG
CHARACTER*8 CNAME
INTEGER*4 FLAG
END FUNCTION

INTEGER*4 FUNCTION DFMCRHS (CNAME,RNAME,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCRHS
!DEC$ATTRIBUTES REFERENCE ::  CNAME,RNAME,VALUE
CHARACTER*8 CNAME
CHARACTER*8 RNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION DFMCRNG (RNGNAME,RNAME,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCRNG
!DEC$ATTRIBUTES REFERENCE ::  RNGNAME,RNAME,VALUE
CHARACTER*8 RNGNAME
CHARACTER*8 RNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION DFMCBND (BNDNAME,CNAME,LVALUE,UVALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMCBND
!DEC$ATTRIBUTES REFERENCE ::  BNDNAME,CNAME,LVALUE,UVALUE
CHARACTER*8 BNDNAME
CHARACTER*8 CNAME
REAL*8 LVALUE
REAL*8 UVALUE
END FUNCTION

INTEGER*4 FUNCTION DFMRRTP (RNAME,TYPE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMRRTP
!DEC$ATTRIBUTES REFERENCE ::  RNAME,TYPE
CHARACTER*8 RNAME
CHARACTER*8 TYPE
END FUNCTION

INTEGER*4 FUNCTION DFMRRSC (RNAME,SCALE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMRRSC
!DEC$ATTRIBUTES REFERENCE ::  RNAME,SCALE
CHARACTER*8 RNAME
REAL*8 SCALE
END FUNCTION

INTEGER*4 FUNCTION DFMRCSC (CNAME,SCALE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMRCSC
!DEC$ATTRIBUTES REFERENCE ::  CNAME,SCALE
CHARACTER*8 CNAME
REAL*8 SCALE
END FUNCTION

INTEGER*4 FUNCTION DFMRVAL (CNAME,RNAME,VALUE)
```

```
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMRVAL
!DEC$ATTRIBUTES REFERENCE ::  CNAME,RNAME,VALUE
CHARACTER*8 CNAME
CHARACTER*8 RNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION DFMRRHS (CNAME,RNAME,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMRRHS
!DEC$ATTRIBUTES REFERENCE ::  CNAME,RNAME,VALUE
CHARACTER*8 CNAME
CHARACTER*8 RNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION DFMRRNG (RNGNAME,RNAME,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMRRNG
!DEC$ATTRIBUTES REFERENCE ::  RNGNAME,RNAME,VALUE
CHARACTER*8 RNGNAME
CHARACTER*8 RNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION DFMRBND (BNDNAME,CNAME,LVALUE,UVALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMRBND
!DEC$ATTRIBUTES REFERENCE ::  BNDNAME,CNAME,LVALUE,UVALUE
CHARACTER*8 BNDNAME
CHARACTER*8 CNAME
REAL*8 LVALUE
REAL*8 UVALUE
END FUNCTION

INTEGER*4 FUNCTION DFMIROW (NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMIROW
!DEC$ATTRIBUTES REFERENCE ::  NAME
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMICOL (NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMICOL
!DEC$ATTRIBUTES REFERENCE ::  NAME
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMIRHS (NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMIRHS
!DEC$ATTRIBUTES REFERENCE ::  NAME
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMIRNG (NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMIRNG
!DEC$ATTRIBUTES REFERENCE ::  NAME
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMIBND (NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMIBND
!DEC$ATTRIBUTES REFERENCE ::  NAME
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMNROW (INDEX,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMNROW
!DEC$ATTRIBUTES REFERENCE ::  INDEX,NAME
```

```
INTEGER*4 INDEX
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMNCOL (INDEX,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMNCOL
!DEC$ATTRIBUTES REFERENCE ::  INDEX,NAME
INTEGER*4 INDEX
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMNRHS (INDEX,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMNRHS
!DEC$ATTRIBUTES REFERENCE ::  INDEX,NAME
INTEGER*4 INDEX
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMNRNG (INDEX,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMNRNG
!DEC$ATTRIBUTES REFERENCE ::  INDEX,NAME
INTEGER*4 INDEX
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMNBND (INDEX,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMNBND
!DEC$ATTRIBUTES REFERENCE ::  INDEX,NAME
INTEGER*4 INDEX
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMLROW (MASK,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMLROW
!DEC$ATTRIBUTES REFERENCE ::  MASK,NAME
CHARACTER*8 MASK
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMLCOL (MASK,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMLCOL
!DEC$ATTRIBUTES REFERENCE ::  MASK,NAME
CHARACTER*8 MASK
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMLRHS (MASK,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMLRHS
!DEC$ATTRIBUTES REFERENCE ::  MASK,NAME
CHARACTER*8 MASK
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMLRNG (MASK,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMLRNG
!DEC$ATTRIBUTES REFERENCE ::  MASK,NAME
CHARACTER*8 MASK
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION DFMLBND (MASK,NAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMLBND
!DEC$ATTRIBUTES REFERENCE ::  MASK,NAME
CHARACTER*8 MASK
CHARACTER*8 NAME
```

```
END FUNCTION

INTEGER*4 FUNCTION DFMLVAL (CNAME,START,INDEX,VALUE,COUNT)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMLVAL
!DEC$ATTRIBUTES REFERENCE ::  CNAME,START,INDEX,VALUE,COUNT
CHARACTER*8 CNAME
INTEGER*4 START
INTEGER*4 INDEX(*)
REAL*8 VALUE(*)
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION DFMTAB  (TABLE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFMTAB
!DEC$ATTRIBUTES REFERENCE ::  TABLE
INTEGER*4 TABLE
END FUNCTION

INTEGER*4 FUNCTION DFSINIT (DB,CNAME)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFSINIT
!DEC$ATTRIBUTES REFERENCE ::  DB,CNAME
INTEGER*4 DB(2)
CHARACTER*8 CNAME
END FUNCTION

INTEGER*4 FUNCTION DFSROW  (NAME,SELECT,STAT,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFSROW
!DEC$ATTRIBUTES REFERENCE ::  NAME,SELECT,STAT,VALUE
CHARACTER*8 NAME
CHARACTER*8 SELECT
character*2 stat
REAL*8 VALUE(5)
END FUNCTION

INTEGER*4 FUNCTION DFSCOL  (NAME,SELECT,STAT,VALUE)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFSCOL
!DEC$ATTRIBUTES REFERENCE ::  NAME,SELECT,STAT,VALUE
CHARACTER*8 NAME
CHARACTER*8 SELECT
character*2 STAT
REAL*8 VALUE(5)
END FUNCTION

INTEGER*4 FUNCTION DFSEND  ()
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFSEND
END FUNCTION

SUBROUTINE DFNCPY  (TO,FROM)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNCPY
!DEC$ATTRIBUTES REFERENCE ::  TO,FROM
INTEGER*4 TO
INTEGER*4 FROM
END SUBROUTINE

SUBROUTINE DFNCAT  (NAME1,NAME2)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNCAT
!DEC$ATTRIBUTES REFERENCE ::  NAME1,NAME2
CHARACTER*8 NAME1
CHARACTER*8 NAME2
END SUBROUTINE

SUBROUTINE DFNFILL (NAME1,NAME2)
!DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNFILL
!DEC$ATTRIBUTES REFERENCE ::  NAME1,NAME2
CHARACTER*8 NAME1
```

```
    CHARACTER*8 NAME2
    END SUBROUTINE

    SUBROUTINE DFNINCR (NAME)
    !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNINCR
    !DEC$ATTRIBUTES REFERENCE ::  NAME
    CHARACTER*8 NAME
    END SUBROUTINE

    SUBROUTINE DFNMASK (NAME1,NAME2)
    !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNMASK
    !DEC$ATTRIBUTES REFERENCE ::  NAME1,NAME2
    CHARACTER*8 NAME1
    CHARACTER*8 NAME2
    END SUBROUTINE

    SUBROUTINE DFNSQOZ (NAME)
    !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNSQOZ
    !DEC$ATTRIBUTES REFERENCE ::  NAME
    CHARACTER*8 NAME
    END SUBROUTINE

    SUBROUTINE DFNSHFT (NAME,N)
    !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNSHFT
    !DEC$ATTRIBUTES REFERENCE ::  NAME,N
    CHARACTER*8 NAME
    INTEGER*4 N
    END SUBROUTINE

    INTEGER*4 FUNCTION DFNCMP  (NAME1,NAME2)
    !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNCMP
    !DEC$ATTRIBUTES REFERENCE ::  NAME1,NAME2
    CHARACTER*8 NAME1
    CHARACTER*8 NAME2
    END FUNCTION

    INTEGER*4 FUNCTION DFNMCMP (MASK,NAME)
    !DEC$ATTRIBUTES STDCALL,REFERENCE ::  DFNMCMP
    !DEC$ATTRIBUTES REFERENCE ::  MASK,NAME
    CHARACTER*8 MASK
    CHARACTER*8 NAME
    END FUNCTION

    END INTERFACE
```

File: WFINC2

Common block used for the WHIZ optimizer

```
! $Header: N:/default/includes/RCS/wfinc2,v 1.3 2001/10/09 17:35:35 PKC Exp $
    TYPE CR
    SEQUENCE
!
    CHARACTER*8 XNAME
    INTEGER     XCORE
    INTEGER     XCORELEN
!
    CHARACTER*8 XACTCASE
    CHARACTER*8 XACTFILE
    CHARACTER*8 XACTPROB
    CHARACTER*8 XBOUND
    CHARACTER*8 XCHOBJ
```

```
      CHARACTER*8 XCHRHS
      CHARACTER*8 XCMASK
      CHARACTER*8 XDATA
      CHARACTER*8 XINSERT
      CHARACTER*8 XLOADB
      CHARACTER*8 XMINMAX
      CHARACTER*8 XOBJ
      CHARACTER*8 XPUNCH
      CHARACTER*8 XRANGE
      CHARACTER*8 XRHS
      CHARACTER*8 XRMASK
      CHARACTER*8 XRNGFILE
      CHARACTER*8 XRNGPRNT
      CHARACTER*8 XSAVEB
      CHARACTER*8 XSOLFILE
      CHARACTER*8 XSOLPRNT
      CHARACTER*8 XSOLSTAT
      CHARACTER*8 XTABCASE
!
      REAL*8     XEPS
      REAL*8     XFUNCT
      REAL*8     XINTGOAL
      REAL*8     XMAXVAR
      REAL*8     XPARAM
      REAL*8     XPENCOST
      REAL*8     XPHI
      REAL*8     XRSTOP
      REAL*8     XSIF
      REAL*8     XSSCALE
      REAL*8     XTHETA
      REAL*8     XTOLCHK
      REAL*8     XTOLCHZR
      REAL*8     XTOLCKRW
      REAL*8     XTOLCLRT
      REAL*8     XTOLDJ
      REAL*8     XTOLERR
      REAL*8     XTOLFSTM
      REAL*8     XTOLINV
      REAL*8     XTOLLMAX
      REAL*8     XTOLPERT
      REAL*8     XTOLPIV
      REAL*8     XTOLREL
      REAL*8     XTOLRMAX
      REAL*8     XTOLRWRT
      REAL*8     XTOLUREL
      REAL*8     XTOLV
      REAL*8     XTOLZE
      REAL*8     XZERO
!
      INTEGER    XCHECKSW
      INTEGER    XCLOCKSW
      INTEGER    XCOMPERR
      INTEGER    XCOMPSW
      INTEGER    XCRASHSW
      INTEGER    XCYCLESW
      INTEGER    XDEGEND
      INTEGER    XDEGMAX
      INTEGER    XDEMAND
      INTEGER    XDETAIL
      INTEGER    XDOATTN
      INTEGER    XDODELTM
      INTEGER    XDOFEAS
      INTEGER    XDOFREQ1
      INTEGER    XDONFS
      INTEGER    XDOUNB
```

```
    INTEGER    XDUAL
    INTEGER    XEQUIL
    INTEGER    XFEAS
    INTEGER    XFREQINV
    INTEGER    XFREQLOG
    INTEGER    XFREQSAV
    INTEGER    XFREQSUM
    INTEGER    XINVERT
    INTEGER    XITERNO
    INTEGER    XJ
    INTEGER    XLUDENSE
    INTEGER    XLUFILL
    INTEGER    XLUINV
    INTEGER    XM
    INTEGER    XMAJERR
    INTEGER    XMAXITER
    INTEGER    XMAXPASS
    INTEGER    XMAXTIME
    INTEGER    XMINERR
    INTEGER    XNEGDJ
    INTEGER    XNET
    INTEGER    XNIF
    INTEGER    XNOWT
    INTEGER    XRUNMODE
    INTEGER    XRYANOSB
    INTEGER    XSCALESW
    INTEGER    XSCRNSW
    INTEGER    XSLPNZ
    INTEGER    XTIMESAV
    INTEGER    XTRACE
    INTEGER    XTRAN
    INTEGER    XUNIQUES
    INTEGER    XUNBDNDX
    INTEGER    XUSEFREE
    END TYPE CR
!
    INTERFACE
!
      subroutine getwcr ( icrloc )
    !DEC$ATTRIBUTES STDCALL, REFERENCE ::  getwcr
    !DEC$ATTRIBUTES REFERENCE ::  icrloc
      integer*4 icrloc
      end subroutine

    INTEGER*4 FUNCTION WFINIT  ( TITLE, TBUF, LEN )
    !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFINIT
    !DEC$ATTRIBUTES REFERENCE ::  TITLE, TBUF, LEN
    CHARACTER*8 TITLE
    real*8 TBUF(*)
    INTEGER*4 LEN
    END FUNCTION

    INTEGER*4 FUNCTION WFDEF   ( MODEL, LEN, MODLNAME )
    !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFDEF
    !DEC$ATTRIBUTES REFERENCE ::  MODEL, LEN, MODLNAME
    REAL*8 MODEL(*)
    INTEGER*4 LEN
    CHARACTER*8 MODLNAME
    END FUNCTION

    INTEGER*4 FUNCTION WFSET   ( MODEL )
    !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFSET
    !DEC$ATTRIBUTES REFERENCE ::  MODEL
    REAL*8 MODEL(*)
    END FUNCTION
```

```
INTEGER*4 FUNCTION WFMPSIN ( ACTFILE, ACTPROB, FILENAME,  &
    DECKANME, PRNTSW, REVISE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFMPSIN
!DEC$ATTRIBUTES REFERENCE ::  ACTFILE, ACTPROB, FILENAME
!DEC$ATTRIBUTES REFERENCE ::   DECKANME, PRNTSW, REVISE
CHARACTER*8 ACTFILE
CHARACTER*8 ACTPROB
CHARACTER*8 FILENAME
CHARACTER*8 DECKANME
INTEGER*4 PRNTSW
INTEGER*4 REVISE
END FUNCTION

INTEGER*4 FUNCTION WFMPSOU ( ACTFILE, ACTPROB, FILENAME, &
 DECKANME, ONECOEF )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFMPSOU
!DEC$ATTRIBUTES REFERENCE ::  ACTFILE, ACTPROB, FILENAME
!DEC$ATTRIBUTES REFERENCE ::  DECKANME, ONECOEF
CHARACTER*8 ACTFILE
CHARACTER*8 ACTPROB
CHARACTER*8 FILENAME
CHARACTER*8 DECKANME
INTEGER*4 ONECOEF
END FUNCTION

INTEGER*4 FUNCTION WFLOAD  ( ACTFILE, ACTPROB )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFLOAD
!DEC$ATTRIBUTES REFERENCE ::  ACTFILE, ACTPROB
CHARACTER*8 ACTFILE
CHARACTER*8 ACTPROB
END FUNCTION

INTEGER*4 FUNCTION WFOPT   ( )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFOPT
END FUNCTION

INTEGER*4 FUNCTION WFEND   ( )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFEND
END FUNCTION

INTEGER*4 FUNCTION WFCVAL  ( COLNAME, ROWNAME, VALUE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCVAL
!DEC$ATTRIBUTES REFERENCE ::  COLNAME, ROWNAME, VALUE
character*8 COLNAME
character*8 ROWNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION WFCRHS  ( ROWNAME, VALUE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCRHS
!DEC$ATTRIBUTES REFERENCE ::  ROWNAME, VALUE
character*8 ROWNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION WFCRNG  ( ROWNAME, VALUE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCRNG
!DEC$ATTRIBUTES REFERENCE ::  ROWNAME, VALUE
CHARACTER*8 ROWNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION WFCBND  ( COLNAME, LOBOUND, UPBOUND )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCBND
```

```
!DEC$ATTRIBUTES REFERENCE ::  COLNAME, LOBOUND, UPBOUND
character*8 COLNAME
REAL*8 LOBOUND
REAL*8 UPBOUND
END FUNCTION

INTEGER*4 FUNCTION WFRVAL  ( COLNAME, ROWNAME, VALUE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRVAL
!DEC$ATTRIBUTES REFERENCE ::  COLNAME, ROWNAME, VALUE
character*8 COLNAME
character*8 ROWNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION WFRRHS  ( ROWNAME, VALUE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRRHS
!DEC$ATTRIBUTES REFERENCE ::  ROWNAME, VALUE
character*8 ROWNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION WFRRNG  ( ROWNAME, VALUE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRRNG
!DEC$ATTRIBUTES REFERENCE ::  ROWNAME, VALUE
CHARACTER*8 ROWNAME
REAL*8 VALUE
END FUNCTION

INTEGER*4 FUNCTION WFRBND  ( COLNAME, LOBOUND, UPBOUND )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRBND
!DEC$ATTRIBUTES REFERENCE ::  COLNAME, LOBOUND, UPBOUND
character*8 COLNAME
REAL*8 LOBOUND
REAL*8 UPBOUND
END FUNCTION

INTEGER*4 FUNCTION WFRLVAL ( COLNAME, START, INDEX, VALUE, COUNT)
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRLVAL
!DEC$ATTRIBUTES REFERENCE ::  COLNAME, START, INDEX, VALUE, COUNT
INTEGER*4 COLNAME
INTEGER*4 START
INTEGER*4 INDEX(*)
REAL*8 VALUE(*)
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION WFRNAME ( INDEX, NAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRNAME
!DEC$ATTRIBUTES REFERENCE ::  INDEX, NAME
INTEGER*4 INDEX
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION WFCNAME ( INDEX, NAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCNAME
!DEC$ATTRIBUTES REFERENCE ::  INDEX, NAME
INTEGER*4 INDEX
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION WFRMASK ( MASK, NAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRMASK
!DEC$ATTRIBUTES REFERENCE ::  MASK, NAME
CHARACTER*8 MASK
CHARACTER*8 NAME
```

```
END FUNCTION

INTEGER*4 FUNCTION WFCMASK ( MASK, NAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCMASK
!DEC$ATTRIBUTES REFERENCE ::  MASK, NAME
CHARACTER*8 MASK
CHARACTER*8 NAME
END FUNCTION

INTEGER*4 FUNCTION WFSROW  ( NAME, SELECT, STAT, SOLVAL )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFSROW
!DEC$ATTRIBUTES REFERENCE ::  NAME, SELECT, STAT, SOLVAL
CHARACTER*8 NAME
CHARACTER*8 SELECT
character*2 STAT
REAL*8 SOLVAL(2)
END FUNCTION

INTEGER*4 FUNCTION WFSCOL  ( NAME, SELECT, STAT, SOLVAL )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFSCOL
!DEC$ATTRIBUTES REFERENCE ::  NAME, SELECT, STAT, SOLVAL
CHARACTER*8 NAME
CHARACTER*8 SELECT
character*2 STAT
REAL*8 SOLVAL(2)
END FUNCTION

INTEGER*4 FUNCTION WFRROW  ( NAME, SELECT, STAT, SOLVAL )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRROW
!DEC$ATTRIBUTES REFERENCE ::  NAME, SELECT, STAT, SOLVAL
INTEGER*4 NAME
CHARACTER*8 SELECT
character*2 STAT
REAL*8 SOLVAL
END FUNCTION

INTEGER*4 FUNCTION WFRCOL  ( NAME, SELECT, STAT, SOLVAL )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRCOL
!DEC$ATTRIBUTES REFERENCE ::  NAME, SELECT, STAT, SOLVAL
INTEGER*4 NAME
CHARACTER*8 SELECT
character*2 STAT
REAL*8 SOLVAL
END FUNCTION

INTEGER*4 FUNCTION WFRLIST ( LIST, COUNT )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRLIST
!DEC$ATTRIBUTES REFERENCE ::  LIST, COUNT
CHARACTER*8 LIST
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION WFCLIST ( LIST, COUNT )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCLIST
!DEC$ATTRIBUTES REFERENCE ::  LIST, COUNT
CHARACTER*8 LIST
INTEGER*4 COUNT
END FUNCTION

INTEGER*4 FUNCTION WFSPRT  ( ACTIVE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFSPRT
!DEC$ATTRIBUTES REFERENCE ::  ACTIVE
INTEGER*4 ACTIVE
END FUNCTION
```

```
INTEGER*4 FUNCTION WFSFILE ( FILENAME, ACTIVE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFSFILE
!DEC$ATTRIBUTES REFERENCE ::  FILENAME, ACTIVE
CHARACTER*8 FILENAME
INTEGER*4 ACTIVE
END FUNCTION

INTEGER*4 FUNCTION WFRCORD ( ACTFILE, ACTPROB, CASENAME, ACTIVE )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRCORD
!DEC$ATTRIBUTES REFERENCE ::  ACTFILE, ACTPROB, CASENAME, ACTIVE
CHARACTER*8 ACTFILE
CHARACTER*8 ACTPROB
CHARACTER*8 CASENAME
INTEGER*4 ACTIVE
END FUNCTION

INTEGER*4 FUNCTION WFRPRT  ( )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRPRT
END FUNCTION

INTEGER*4 FUNCTION WFRFILE ( FILENAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRFILE
!DEC$ATTRIBUTES REFERENCE ::  FILENAME
CHARACTER*8 FILENAME(*)
END FUNCTION

INTEGER*4 FUNCTION WFSAVEB ( FILENAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFSAVEB
!DEC$ATTRIBUTES REFERENCE ::  FILENAME
CHARACTER*8 FILENAME(*)
END FUNCTION

INTEGER*4 FUNCTION WFLOADB ( FILENAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFLOADB
!DEC$ATTRIBUTES REFERENCE ::  FILENAME
CHARACTER*8 FILENAME(*)
END FUNCTION

INTEGER*4 FUNCTION WFPUNCH ( FILENAME, DECKANME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFPUNCH
!DEC$ATTRIBUTES REFERENCE ::  FILENAME, DECKANME
CHARACTER*8 FILENAME
CHARACTER*8 DECKANME
END FUNCTION

INTEGER*4 FUNCTION WFINSRT ( FILENAME, DECKANME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFINSRT
!DEC$ATTRIBUTES REFERENCE ::  FILENAME, DECKANME
CHARACTER*8 FILENAME
CHARACTER*8 DECKANME
END FUNCTION

INTEGER*4 FUNCTION WFMGDMP ( MODEL, WMGNAME, KEEPFAC )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFMGDMP
!DEC$ATTRIBUTES REFERENCE ::  MODEL, WMGNAME, KEEPFAC
REAL*8 MODEL(*)
CHARACTER*8 WMGNAME
INTEGER*4 KEEPFAC
END FUNCTION

INTEGER*4 FUNCTION WFMGLOD ( MODEL, WMGNAME, MODLNAME )
!DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFMGLOD
!DEC$ATTRIBUTES REFERENCE ::  MODEL, WMGNAME, MODLNAME
REAL*8 MODEL(*)
CHARACTER*8 WMGNAME
```

```
      CHARACTER*8 MODLNAME
      END FUNCTION

      INTEGER*4 FUNCTION WFCRGET ( NAME, VALUE, VLEN, ALEN )
      !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFCRGET
      !DEC$ATTRIBUTES REFERENCE ::  NAME, VALUE, VLEN, ALEN
      CHARACTER*8 NAME
      INTEGER*4 VALUE
      INTEGER*4 VLEN
      INTEGER*4 ALEN
      END FUNCTION

      INTEGER*4 FUNCTION WFCRPUT (COMMAND,NAME,VALUE,STRING,VERBOSE)
      !DEC$ATTRIBUTES STDCALL, REFERENCE :: WFCRPUT
      !DEC$ATTRIBUTES REFERENCE :: COMMAND,NAME,VALUE,STRING,VERBOSE
      CHARACTER*8 COMMAND
      CHARACTER*8 NAME
      INTEGER*4 VALUE
      INTEGER*4 STRING
      INTEGER*4 VERBOSE
      END FUNCTION

      INTEGER*4 FUNCTION WFALLOC ( MODEL, LEN )
      !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFALLOC
      !DEC$ATTRIBUTES REFERENCE ::  MODEL, LEN
      REAL*8 MODEL(*)
      INTEGER*4 LEN
      END FUNCTION

      SUBROUTINE WFFREE         ( MODEL, LEN )
      !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFFREE
      !DEC$ATTRIBUTES REFERENCE ::  MODEL, LEN
      REAL*8 MODEL(*)
      INTEGER*4 LEN
      END SUBROUTINE

      INTEGER*4 FUNCTION WFRDCR  ( CRFILE )
      !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFRDCR
      !DEC$ATTRIBUTES REFERENCE ::  CRFILE
      CHARACTER*8 CRFILE
      END FUNCTION

      INTEGER*4 FUNCTION WFWRCR  ( CRFILE )
      !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFWRCR
      !DEC$ATTRIBUTES REFERENCE ::  CRFILE
      CHARACTER*8 CRFILE
      END FUNCTION

      SUBROUTINE WFSYSPR        ( FMT )
      !DEC$ATTRIBUTES STDCALL, REFERENCE ::  WFSYSPR
      !DEC$ATTRIBUTES REFERENCE ::  FMT
      CHARACTER*8 FMT
      END SUBROUTINE
!
      subroutine PICK (CASEKEY, PICKRG, PICRTN, I)
      !DEC$ATTRIBUTES STDCALL, REFERENCE :: pick
      integer*2 casekey(*)
      integer*4  pickrg(*)
      integer*4  picrtn(*)
      integer*4  i
      END SUBROUTINE

      subroutine FIND (CASEKEY, Ival, SL, Jval, I)
      !DEC$ATTRIBUTES STDCALL, REFERENCE :: find
      integer*2 casekey(*)
```

```
integer*4  sl(*)
integer*4  i,ival,jval
END SUBROUTINE

END INTERFACE
integer icrloc
TYPE (CR) :: OML
POINTER (icrloc,OML)
common/wcr/icrloc
```

Appendix G-E: MRM Data Conversion

The current data used in the MRM (and ERM) were updated for the AEO2002 using the Feb2001 data (referred to as ENSYS01Feb) provided by EnSys Co. Due to the differences in data format (OMNI vs OML) and naming conventions, the ENSYS01Feb data had to undergo convertion and new mapping data tables had to be added to the MRM files prior to being used by the MRM.

The following provides a summary of the effort it took to convert the ENSYS01Feb OMNI tables into OML RTB tables (to be read by the matrix generation code).

1. The code to convert the OMNI tables to RTB tables is called *omni2rtb.c* and can be found in directory m:/ogs/pmm_prj/ensys/ensys01-feb/source01/ (on the EIA NT server).

2. To compile and execute the omni2rtb type:
   compile: cc -o <exec filename> <source code filename>
   execute: <exec filename> <input data path/filename> <output filename>
   For example, if all files are in the same directory:
   cc -o omniexec omni2rtb.c
   omniexec v0.ensrymeq.gdb omniout

3. Technology database development since 1994, according to ENSYS:

   - Implementation of weight balancing on every process vector
   - Implementation of rigorous sulfur balancing on every process vector (down to the ppm level)
   - Conversion of the technology database to spreadsheet basis (Excel)
   - Implementation of model compositing system. This current task is directed to RYM-type regional/prototypical refinery fuels/regulatory studies and takes advantage of the fact that, in any one case, crude slate is essentially fixed. This enables a much smaller matrix to be generated, with yields and qualities crude slate dependent, from the un-composited model tables. Resulting model matrix has far fewer prototypical streams, less susceptibility to over-optimization (and attendant criticism), more manageable when ratio controls are used. (The task to develop this system is proving to be a rigorous diagnostic on model structure, non-composited and composited, and as such is helping eliminate any errors.)

4. Recent new processing units related to technology improvements included in refproc.dat:
   - Atmospheric Residuum Desulfurization (ARD)
   - Catalitic Desulfurization (CDT)
   - Low Conversion Hydrocracker (HCL)
   - Catalytic Naphtha Hydrotreater (SYG)
   - Hydrodesulfurizer 1 (HD1, HL1)
   - Hydrodesulfurizer 2 (HD2, HS2)
   - Hydrocracker (partial) (MAK)
   - Catalytic Fluidized Bed (MOD)
   - Middle Distillate Hydro (MDH, orignally identified by Ensys as MOH)

- Catalytic Fixed-Bed Hydroprocessor (OCT)
- Caustic SOx Scrubber (SOX)
- Methanol to Olefins Fluid Bed (MTO)
- Shell Middle Distillate (SMD)
- Syntroleum GTL Process (SOD)

5. The updated OMNI data files provided by ENSYS and used to update the mrm data tables are located in the directory m:/ogs/pmm_prj/ensys/ensys01-feb/data01/ , and include:

   R-GCB.DAT
   R-MPROC.DAT
   R-NPROC1.DAT
   R-NPROC2.DAT
   R-PROC1.DAT
   R-PROC2.DAT

   The following list of *.dat and tables were updated using the new ensys01Feb data:

stream.dat:	DATA	T:TRS
distblnd.dat:	DATA	T:DCC
	DATA	T:DCB
gasoblnd.dat:	DATA	T:MCO
	DATA	T:REGBV
	DATA	T:LOGBV
	DATA	T:PRMBV
	DATA	T:PRCBV
	DATA	T:UNLBV
	DATA	T:UNCBV
	DATA	T:RFMBV
	DATA	T:RGBBV
	DATA	T:RGCBV
	DATA	T:GCB
	DATA	T:GCC
recipes.dat:	DATA	T:RCP
refproc.dat:	DATA	T:MATBAL
	DATA	T:HLO
	DATA	T:RST
	DATA	T:SDA
	DATA	T:KRD
	DATA	T:KRF
	DATA	T:VBR
	DATA	T:NDS
	DATA	T:DDS
	DATA	T:FDS
	DATA	T:RDS
	DATA	T:RFH
	DATA	T:RFL
	DATA	T:RFC
	DATA	T:SPL
	DATA	T:HCN
	DATA	T:OLE

DATA T:SYD
DATA T:JFP
DATA T:FCC
DATA T:FGS
DATA T:TCG
DATA T:TCN
DATA T:TCV
DATA T:HCR
DATA T:HCV
DATA T:ETS
DATA T:C24
DATA T:H56
DATA T:C4T
DATA T:FEX
DATA T:HDN
DATA T:JPS
DATA T:DEW
DATA T:ALK
DATA T:CPL
DATA T:DIP
DATA T:ETH
DATA T:CYC
DATA T:ALM
DATA T:DIM
DATA T:ARP
DATA T:RES
DATA T:LUB
DATA T:PHI
DATA T:TRI
DATA T:C4I
DATA T:C4S
DATA T:H2P
DATA T:H2X
DATA T:SUL
DATA T:VCU
DATA T:KWG
DATA T:STG
DATA T:REL
DATA T:PFA
DATA T:OCT
DATA T:CDT
DATA T:SYG
DATA T:MTO
DATA T:MOD
DATA T:MDH (ENSYS table MOH)
DATA T:SMD
DATA T:SOD
DATA T:HD1
DATA T:HD2
DATA T:HL1

	DATA	T:HS2
	DATA	T:JFP
	DATA	T:ETB
	DATA	T:HMP
	DATA	T:BID
	DATA	T:AVC
	DATA	T:MAK
	DATA	T:MER
	DATA	T:SOX
	DATA	T:HCL
	DATA	T:ARD
	DATA	T:INVGEN
	DATA	T:INVUNT
	DATA	T:SCL
	DATA	T:FUM
refproc.dat:	DATA	T:C4X
	DATA	T:OLX
	DATA	T:ETX
	DATA	T:FUX
	DATA	T:STX
	DATA	T:SMD
	DATA	T:SOD

Some data error corrections or updates were made to the converted RTB files and/or the mrm input files. These corrections were recorded in the *A.readme* and *A1.readme* files in the data01 directory indentified above.

Appendix G-F: Refinery Processes

Refinery Process	Abbreviation
Atmospheric crude distillation	ACU
Alkylation	ALK
Alkynax	ALM*
Atmospheric Residuum Desulfurization	ARD****
Aromatic recovery	ARP
Catalytic Distillation Technology	CDT
Cogeneration unit	CGN**
Polymerization	CPL
Cyclar	CYC
Butane isomerization	C4I
Butane splitter	C4S**
Akylation feed butene isomerizer	C4T*
Butane isomerization	C4X***
C2E to C4E dimerization	C24*
Distillate desulfurizer	DDS
Dimersol	DIM
Gas oil dewaxer	DEW
Di-isopropyl ether	DIP*
Etherol	ETH, ETM
Cryogenic C2 fractionation	ETS*
Etherrol unit	ETX***
FCC gasoline fractionation units	FGS
FCC feed hydrofiner	FDS
Fluid cat cracker	FCC
Mid-distillate furfural treating	FEX
Fuel plant	FUM**
Fuel plant	FUX***
Gas oil hydrocraker	HCR
Residum hydrocracker	HCV

Refinery Process	Abbreviation
Low Conversion Hydrocracking	HCL****
Naphta hydrocracker	HCN
Hydrodesulfurizer 1	HD1, HL1****
Hydrodesulfurizer 2	HD2, HS2****
High density jet fuel hydrotreating	HDN
Hydogen/fuel gas reformer hydrogen	HLO**
H2-stream reformer	H2P
H2-partial oxidizer	H2X
Hydrogeneration normal pentenes/hexenes	H56*
High density jet fuel prefactionation	JFP
Recut for JTA	JPS**
Delayed-coker	KRD
Fluid/flexi-coker	KRF
Power generation	KWG**
Lub and wax	LUB
Hydrocracker (partial)	MAK
Olefin Saturation Process	MDH****
Catalytic Fluidized Bed	MOD****
Methanol to Olefins Fluid Bed Process	MTO****
Naphta hydrotreater	NDS
OCTGAIN Catalytic Hydroprocessor	OCT
C2-C5 dehydrogenation	OLE*
C2-C5 dehydrogenation	OLX***
Pentane/hexane isomerization	PHI
Residuum desulfurizer	RDS
HP semi-regenerative reformer	RFH
LP cyclic reformer	RFL
LP continuous reformer	RFC
Solvent deasphalting	SDA
Shell Middle Distillate	SMD
Syntroleum GTL Process	SOD
Caustic SOx Scrubber	SOX****

Refinery Process	Abbreviation
Naphta splitter	SPL
Steam generation, lbs/hr	STG**
Steam generation, lbs/hr	STX***
Sulfur, short tons/day	SUL
Mid-distillate deep hydrotreater	SYD
Catalytic Naphtha Hydrotreater	SYG****
Thermal cracker C2-C4 feed	TCG*
Thermal craker naphta feed	TCN*
Thermal craker gas oil feed	TCV*
Total recycle isomerization	TRI*
Vaccum distillation	VCU
Visbreaker/thermal craker	VBR

* : Processes involved in reformulated gasoline manufacturing
**: Utilities and pseudo-units
***: Processes represented in OXY-Refineries
****: Not made available for *AEO2001*-- new, not ready for production

APPENDIX H

Historical Data Processing

APPENDIX H. Historical Data Processing

H.1 Processing Data for PMM History File

Data Dump Programs:

1) PSA Query: In Access open c:\OGIRS3.6\modify ogirs2. Click on get PSADATA. This runs a query and produces tblStacey. To add or delete elements in the query change tblParkListStacey and rerun. Open PSADATA.XLS in Excell save as space delimited text file (PSA00.PRN). In FTP/TCP download ASCII file to mainframe to be read by SAS files. (May need to save as card on mainframe.)

   The last version of OGIRS was missing some of the elements of refinery production: nfrpp1-p5(naphtha feedstocks), otrpp1-p5 (other oils for feedstock), msrpp1-us(miscellaneous products for nonfuel use), pfrp-us(total petrochemical feedstocks).

2) PRJ6007.NEMS.MARKUP.PRICE.HISTFILE: reads markup and tax files and creates a file of historical end-use prices.

Input files:	PRJ6007.NEMS.MARKUP.SASDB
	PRJ6007.NEMS.TAX.SASDB
Output file:	PRJ6007.PRICES.HIST.SASDB: PBTU0001

3) PRJ6007.NEMS.READ.SEDS: reads flat-file of SEDS sectoral consumption data, converts to thousand barrels per day, estimates “other” consumption, and outputs to a SASDB.

Input files:	PRJ6007.SEDS99A.DATA provided by EMEU (Roy Stanley)
	PRJ6007.SEDS99B.DATA
	PRJ6007.NEMS.MARKUP.SASDB
Output file:	PRJ6007.HISTORY.SASDB:SEDS.

4) Manually update PRJ6007.FUELUSED from fuel consumption data in Table 47. of the PSA. Natural gas and coal consumption have been published only every 2 years. The table will be published every year starting in 2000.

5) Update PFPRICES.XLS to estimate petrochemical feedstock prices (see SEPER APPENDIX).

p naphtha= 1.23* WOP and other = 1.607 * WOP, uses 1999 prices weighted by 1999 & 2000 volumes to estimate average pf price from PSA.

6) Manually update PRJ6007.ELCG.FLATFILE.DATA(MSPRICE) using estimates from PFPRICES.XLS.

7) Crude oil Wellhead Production and Prices are provided by the OGSM modeling team and are added to the history file during STEO data processing (see Section H.2).

8) Updated cogeneration data provided by the EMM modeling team and added during STEO data processing (Section H.2).

Data Processing Files:

1) PRJ6007.PMM.PROCESS.PSADATA - loads PRJ6007.PSA00.PRN which is created from a query of OGIRS(See PSA Query). The data is transposed, manipulated, and put data into SASDB.

Input files: PRJ6007.PSA00.PRN
Output file: PRJ6007.PMM.HISTORY.DATA.SASDB:PSDATA

2) PRJ6007.PMM.PROCESS.RFFUEL - reads refinery fuel consumption data and outputs to SASDB.

Input files: PRJ6007.FUELUSED manually updated
and PRJ6007.PMM.MISCELL.SASDB:BTU1

Output file: PRJ6007.PMM.MISCELL.HIST.SASDB:RFFUELCD in BTU's

3) PRJ6007.PMM.PROCESS.PRCDATA - reads prices and petrochemical prices, transposes for regional headings and puts in SASDB. Data set "test1" must be updated for current year for CD4 and CD8. Adds a 2 cents per gallon (nominal) local gasoline tax.

Input files: PRJ6007.PRICES.HIST.SASDB:PBTU0001
PRJ6007.ELCG.FLATFILE.DATA(MSPRICE)

Output file: PRJ6007.PRICES.HIST.SASDB:PRDPRIC

4) PRJ6007.PMM.PROCESS.WPRCDATA - reads wholesale prices transposes for regional headings and puts in SASDB. Data set "test1" must be updated for current year for CD4 and CD8.

Input files: PRJ6007.PRICES.HIST.SASDB:PBTU0001

Output file: PRJ6007.PRICES.HIST.SASDB:WPRDPRIC

5) PRJ6007.PMM.PROCESS.DEMAND - reads sectoral demand from SEDS, estimates most recent 2 year's sectoral demand by using current PSA total demand times the sector ratios. Type in PSA total product supplied number from Table 3.

Input files: PRJ6007.HISTORY.SASDB:SEDS

Output File: PRJ6007.PRODTN.HIST.SASDB:DEMAND

6) PRJ6007.PMM.PROCESS.MISCELL - includes data for miscellaneous inputs, to be manually updated each year. See ETHBLND.XLS for ethanol blending (RFETHMGS).

Output file: PRJ6007.PMM.MISCELL.SASDB:MISCELL

Creating PMM Flat-file:

1) PRJ6007.PMM.MAKE3.HIST.FLATFILE - reads SASDBs and STEO estimates and outputs them to a flat-file. Must run STEO programs in Appendix H.2 to update STEO benchmarking.

Input files: PRJ6007.PMM.HISTORY.DATA.SASDB:PSDATA
PRJ6007.MISCELL.HIST.SASDB:RFFUELCD
PRJ6007.PRODTN.HIST.SASDB:CRDOGSM (updated manually later)
PRJ6007.PRODTN.HIST.SASDB:DEMAND
PRJ6007.PRICES.HIST.SASDB:CRDPRIC (updated manually later)
PRJ6007.PMM.HISTORY.DATA.SASDB:COGEN (updated later)
PRJ6007.PRICES.HIST.SASDB:PRDPRIC
PRJ6007.PMM.MISCELL.SASDB:MISCELL
PRJ6007.NEMSSTEO.COMP.BASE
Output file: PRJ6007.RFHIST6
** lrecl=120

The crude oil and cogeneration information is usually updated later via cut and paste.

Cogeneration Data Processing:

Historical cogeneration data is derived from the EIA-860B, Annual Nonutility Power Producer Report. The data is obtained from an Access query by Alan Beamon of the Electricity Market Modeling team. The data is provided in the form of an Excell spreadsheet 1999.XLS which contains refinery cogeneration capacity, generation, fuel consumption, and sales to grid data as well as planned new capacity by census division. The spreadsheet is saved as formatted text (space delimited). The .PRN text file cog99c.prn is then transported to the EIA mainframe computer and read by the SAS program &PRJ6007.PMM.HIST.COGEN. The program combines the most recent years data with old data, manipulates it into the format needed by the LP and outputs to &PRJ6007.PMM.HISTORY.DATA.SASDB:COGEN0001. Note: cogeneration capacity by PADD from the last historical year is multiplied by 24(hours) for input into the PMM file COGENER.DAT.

H.2 Processing Data for STEO Years

The *Short-Term Energy Outlook* (STEO) published by the Energy Information Administration provides quarterly projections 2 years into the future. Since the STEO is EIA's official forecast during this period, the Petroleum Market Model was set up to produce numbers that very closely match the STEO when a switch in the PMM is turned on. The information in this section describes the programs that are used to generate this data for the PMM. The data is then added to the input file that contains the history data described in Section H.1.

The files below are on mainframe account 6007, PRJ, unless otherwise stated. Input data are read from and output data are written to

&PRJ6007.NEMSSTEO.COMP.BASE.SASDB

unless otherwise indicated. Files 1 and 2 only need to be updated once a year when new historical price and supply data are available. The other files can be updated with data from any STEO projection. Note that some files produce output for other files, so the order of execution is important. For example, program 6 inputs data from programs 3 and 5.

1. NEMS.HIST.PRICES.MARGINS

This file creates a data file “RETMARG” which contains historical product prices, crude prices, and margins.

Intput files:	many	from PRJ6007.NEMS.MARKUP.SASDB
	many	from PRJ6007.NEMS.TAX.SASDB
	many	from PRJ6007.A.SASDB
Output file:	RETMARG	to PRJ6007.A.SASDB

2. NEMSSTEO.HISTMARG

This file calculates differentials that are used to estimated regional prices from U.S. average STEO prices. Differentials are also calculated from crude oil prices for those fuels that STEO does not price.

Input file:	RETMARG	from PRJ6007.A.SASDB
Output files:	RCSPRD88	
	CDSPRD88	
	RFSPRD88	

3. NEMSSTEO.HISTFACT

This file calculates factors from petroleum supply data which are used to apportion U.S. data to the PADD level.

Input file:	PSDATA	from PRJ6007.PMM.HISTORY.DATA.SASDB
Output file:	SUPPFACT	

4. NEMSSTEO.PRICE.DATA

STEO prices (taken from STEO report, Table 4) are typed in to this program and the differentials from file 1 are input, then regional PMM product prices are estimated.

Input files:	CDSPRD88	from PRJ6007.NEMSSTEO.COMP.BASE.SASDB
	RCSPRD88	
	RFSPRD88	
	GDP87CH	from PRJ6007.NEMS.MARKUP.SASDB
Output files:	PRICES	
	RRAC87	

5. NEMSSTEO.READ.STEODB.OTHINP

This program reads in data from the STEO database and outputs data needed for later processing (monyy = sep01). Also need some data from PSA table 3 (Other hydrocarbons/Oxygenates & Motor gasoline blending components).

Input files:	ALLBBB	from PRJ6489.STIFSIII.monyy.SASDATA
Output files:	CAPUTIL	(STEO capacity and utilization rates)
	OTHIMP	(imports of unfinished oils, ethers, and gasoline blending components)
	OTHINPUT	(other input data)

6. NEMSSTEO.SUPPLY.PUBDATA

In this file, supply data are typed in from the STEO publication (STEO table 5).

Input files:	SUPPFACT	from PRJ6007.NEMSSTEO.COMP.BASE.SASDB
	OTHIMP	
	CAPUTIL	
Output files:	USSUP	
	CRUDEINP	
	CRDPROD	
	PADSUP	

7. NEMSSTEO.READ.STEODB.PRODSUP

This program uses data from the STEO database and file 5, and develops product supplied data for the PMM (monyy = sep01).

Input files:	ALLBBB	from PRJ6489.STIFSIII.monyy.SASDATA
	USSUP	from PRJ6007.NEMS.STEO.COMP.BASE.SASDB
Output files:	PRODSUP	
	PRODSECT	

8. NEMSSTEO.READ.STEODB.REFPROD

This program uses data from the STEO database and file 2, and develops refinery production numbers (monyy = sep01).

Input files:	ALLBBB	from PRJ6489.STIFSIII.monyy.SASDATA
	SUPPFACT	from PRJ6007.NEMS.STEO.COMP.BASE.SASDB

Output:	REFPROD	

9. NEMSSTEO.RFFUEL

This program estimates refinery fuel consumption based on historical values and STEO crude input estimates.

Input:	CRUDEINP	from PRJ6007.NEMS.STEO.COMP.BASE.SASDB
	RFFUELCD	from PRJ6007.FUELUSE.HIST.SASDB
Output:	FUELUSE	

10. NEMSSTEO.WELLPR.D091100

This file estimates regional crude wellhead prices based on wellhead price equations in the PMM provided by the Oil and Gas Supply Team and STEO's world oil price projections.

Input:	GDP87CH	from PRJ6007.NEMS.MARKUP.SASDB
	flatfile	PRJ6007.WELLPR.D091100
	flatfile	PRJ6007.WELLPROD.D091100
Output:	WELLREG	
	CRDPROD	

11. NEMSSTEO.INPUT.DSTCAP.DSTADD

This file updates historical and STEO year distillation capacity and additions. NOTE: This routine replaces output files HCAP and HADD, so use different output filenames when testing.

Input:	ADDHCAP	from data input cards
	ADDHADD	from data input cards
	STEOCAP	from data input cards
	STEOADD	from data input cards
Output:	HCAP	to PRJ6007.PMM.HISTORY.DATA.SASDB
	HADD	
	ELCGCAP	
	ELCGADD	

12. NEMSSTEO.ALLDATA

This program reads in all the relevant data created by the previous programs, drops the last historical year,[1] and writes out the data needed to input into the PMM history file.

Input:	Output:
PADSUP	SPADSUP
REFPRD	SREFPRD
USSUP	SUSSUP
OTHINPUT	SOTHINP
PRODSUP	SPRODSUP
PRODSECT	SPRDSECT
FUELUSE	SFUELUSE
PRICES	SPRICES
WELLREG	SWELLREG
CRDPROD	SCRDPROD

13. PMM.MAKE3.HIST.FLATFILE

This program reads in the output data from **12** above and produces a file containing PMM output data for the history and STEO years. (NOTE: had to manually update data for RFETH*** for 2000, 2001, 2002 data in RFHIST6 file—saved into RFHIST6.mod.)

Output:	flatfile	to PRJ6007.RFHIST6

[1]The last historical year was included to provide a check that the estimates for the forecast years were reasonable.

H.3 Processing Other Historical Data

In addition to developing an input history file (described in Appendix H-1), the PMM team utilizes other historical data to develop some inputs and to support analysis of the model results. This section describes the updating of these data, which is usually done on an annual basis.

Crude Oil Price Data

EIA-14: Crude oil prices are used to calculate historical margins which are used to analyze the margins estimated in the model. The series used is the refiner acquisition cost (RAC) of imported crude oil which comes from Form EIA-14. The domestic and composite series are also maintained but the imported RAC is the series used in the margin calculations (Contact Elizabeth Scott 202-586-1258). Program needs updating to read a four digit year variable.

The file to update the crude prices each year is:

PRJ6007.EIA14.READ.FROZEN where yy represents the year

Input files: PRJ6105.ANNUAL.FROZEN.Dyymm where yymm represents year and month
Use command (SL &6105prj.) Because the files are on the old system.

Output files: (all located in PRJ6007.A.SASDB)

YMCRyy	monthly prices for year yy
QTCRyy	quarterly prices for year yy
YRCRyy	annual prices for year yy
YMCRUDE	monthly prices from 1974 through year yy
QTCRUDE	quarterly prices from 1978 through year yy
YRCRUDE	annual prices from 1978 through year yy

A new file must be created each year, changing the yy=s of the previous year to the current year.

EIA-856: The EIA-856 survey data were used in the past for analyzing results from the PMM. The EIA-856 collects prices by crude stream for imported crude oil. Differentials between these prices and the world oil price (the refiner acquisition cost of imported oil from the EIA-14) can be calculated and compared with model results. The crude streams can also be segregated by PADD regions and the aggregate crude types found in the PMM. The primary file for reading the EIA-856 data and performing various calculations is

PRJ6007.EIA856.READ95.PORT (Missing)

Petroleum Product Price Data

Data on petroleum product prices is obtained from the EIA-782 surveys. The EIA-782A survey contains only refiner data, the EIA-782B survey includes petroleum marketers. Prices are produced monthly and updated for an annual publication. One file reads the monthly data, and a second file is set up to read the numbers from the annual data. The file names are as follows:

PRJ6007.A.IMP00.ADF	reads monthly data for 2000 (used for AEO due to timeliness)
PRJ6007.A.IMP99.ADF	reads final monthly data for 1999

Each year the program from the previous years data should be rerun using the final, RLSE3, data. The program should be modified to read the preliminary,RLSE2, data from the most recent year, run again, and resaved with a new name.

Input files: PRJ6015.E782P.R.DATA.ADF.D1999mm.RLSE3 (for final annual update of the monthly data, where mm represent months)
PRJ6015.E782P.R.DATA.ADF.D2000mm.RLSE2 (for monthly data where mm represents months)

Output files: IMP00A contains refiner data only
IMP00B contains refiner and marketer data combined
(these files are found in PRJ6007.A.SASDB.STATE.AByyyy, where yyyy represents two consecutive years (i.e., two years of data are placed in one SASDB, then a new SASDB is created for the next two years, and so on.))

Historical Prices and Margins

Historical wholesale and end-use prices from the EIA-782 are aggregated and presented in tabular form by product type and Census division. The end-use transportation prices include State and Federal taxes but for jet fuel and LPGs the State taxes are not included prior to 1995.

Differentials with the world oil price (the refiner acquisition cost of imported oil from the EIA-14) are also calculated by product type and Census division and presented in tabular form for analyzing similar margin calculations from the PMM. The following program has been written to perform these calculations and develop the tables. The margins include the 2-cents-per-gallon local tax that is currently being added to gasoline price projections.

PRJ6007.NEMS.HIST.PRICES.MARGINS

Summary

Once the monthly prices from the EIA-782 are finalized for a particular year, set up and run the program A.IMPyy.ADF for that year (yy is the year). Set up and run the EIA14.READyy.FROZEN for that year. Then set up and run NEMS.HIST.PRICES.MARGINS to get the price and margin tables. Update with the annual data as it becomes available.

APPENDIX I

Ethanol Supply Model

Appendix I. Ethanol Supply Model

I.1 Model Purpose

The objective of the ethanol supply model is to provide the NEMS Petroleum Market Module (PMM) with supply curves for corn and cellulose based ethanol, thus allowing the PMM to forecast transportation ethanol demand throughout the NEMS forecast period. To be consistent with the market clearing mechanism adopted for NEMS, the model provides ethanol prices in the form of annual price-quantity curves. The curves, derived from an ethanol production cost function, represent the prices of ethanol at which associated quantities of transportation ethanol are expected to be available for production of E85 and ETBE, and for blending with gasoline.

The delivered ethanol prices are provided to the PMM linear program in the form of a unique supply curve for each of the nine U.S. Census Divisions. The majority of ethanol currently produced in the United States is made from corn and is produced in the East North Central Census Division (NEMS region 3), and the West North Central Census Division (NEMS region 4). Smaller amounts are available in the East South Atlantic (NEMS region 6), the Mountain (NEMS region 8), and the Pacific (NEMS region 9) Census Divisions. The PMM also models planned cellulose based ethanol production beginning in 2003 in the Middle Atlantic (NEMS region 2) and West South Central (NEMS region 7) Census Divisions. The majority of growth in cellulose ethanol production, however, is forecasted for Census Divisions 3 and 4, where large quantities of corn stover (the most likely biomass feedstock) are available, and in Census Division 9, where ethanol demand is expected to grow in response to the California ban on MTBE in reformulated gasoline.

I.2 Corn-Based Ethanol

Fundamental Assumptions

The corn-based ethanol model uses a process costing approach to model the impacts of net feedstock production costs plus the capital, operating, and process energy costs associated with converting the corn feedstocks to ethanol. In other words, each of the above factors contributes a part of the total price of ethanol.

To determine the delivered ethanol price, the contribution of the net cost of corn feedstock production must be factored in to the total unit price of ethanol. Net cost of corn feedstock is the price of corn less the price of the co-products produced in the conversion of corn to ethanol. Conversion of corn to ethanol is accomplished by either a wet milling or dry milling process. The co-products produced from the wet

milling process are corn gluten feed (CGF), corn gluten meal (CGM), and corn oil, while the dry milling process produces distillers dried grains (DDGS). Prices for CGF, CGM, and DDGS are obtained from the USDA Feed Grains Data Delivery System (http://www.ers.usda.gov/db/feedgrains/default.asp?ERSTab=3). The price of corn oil is obtained from the October 2000 Oil Crops Situation and Outlook. Prices were converted to 2000 dollars using the chained GDP price index as reported in Table 2 of USDA Baseline Projections to 2009.

The price of corn is projected from *Analysis of Ethanol Production Under a Renewable Fuels Requirement*, Office of Energy Policy and New Uses, USDA, September 1, 2000. This paper estimates the effect on agricultural markets of a renewable fuels requirement for gasoline, starting at 1.3 percent and rising to 3.3 percentby volume in 2010. The USDA constructed a baseline agricultural market forecast and an alternative forecast under the renewable fuels standard. Corn prices in 2000 dollars and quantity of corn input to ethanol production are included in the forecasts. This gives two points of corn price as a function of corn input to ethanol production to which a line was fitted.

Wet milling accounts for about 60 percent of current total ethanol production, while new ethanol facilities are projected to be dry milling plants.[1] Therefore, ethanol production levels below the baseline forecast use a net feedstock price calculated assuming 60 percent of co-products are CGM, CGF, and corn oil, and 40 percent are DDGS. For ethanol production above baseline, co-products are assumed to be 100 percent DDGS. The variability of the market price for the feedstock corn and the conversion by-products and the variable influences of competitive uses for corn (e.g., for producing corn syrup) give rise to broad fluctuations in net corn feedstock prices. As ethanol production from corn increases, land becomes scarcer, causing both land and feedstock costs to increase. These factors are included in the USDA model. The USDA forecasts also include ethanol production from sorghum. Sorghum ethanol output is an order of magnitude smaller than corn ethanol output and so is not modeled in PMM. The USDA projections end in 2010, so net feedstock prices for 2011 to 2020 were set equal to the 2010 estimates. The net feedstock costs were then converted to 1987 dollars by applying a GDP deflator of 1.379.

In addition to feedstock prices and quantities, the model requires capital cost, feedstock conversion cost (non-energy operating cost), and energy cost data. The cost data were derived from several sources which are documented in the Inventory of Variables, Data, and Parameters section of this report. Note that with this theoretical approach, only the agricultural, or feedstock production costs are modeled as a function of the total quantity of ethanol produced. The conversion plant process costs, (capital, operating, and process energy) are independent of production quantities. The feedstock production cost components are estimated statistically, whereas the conversion process costs are determined from engineering concepts and data. Actual ethanol conversion process data are, for the most part, proprietary.

Capital and conversion costs were assumed to be constant across all Census Divisions and for all forecast

[1]Urbanchuk, J.M. 1998. "Review of Alternative Ethanol Supply Curves Used in the Energy Information Administration's National Energy Modeling System."

years. Energy costs vary across Census Divisions as a function of fuel price. The model assumes wet milling plants consume 20 percent natural gas and 80 percent coal, while dry milling plants consume 50 percent natural gas and 50 percent coal.[2] Natural gas prices are obtained from the NEMS Natural Gas Transmission and Distribution model, and coal prices are from the NEMS Coal Market Model. It was also assumed that the quantity of energy needed for ethanol conversion decreases through the forecast period as ethanol plants become more efficient over time. The supply/price curves in Census Divisions 3, 4, and 8 also include credits of $0.14, $0.08, and $0.18 per gallon, respectively. These credits are weighted averages of producer incentives and State tax credits offered by various States within these Census Divisions. The ethanol blender's Federal excise tax credit, which is currently $0.053 per gallon of gasohol (10 percent ethanol, 90 percent gasoline), is modeled separately in the PMM.

Key Computations and Equations

Net feedstock costs are calculated in a separate spreadsheet by subtracting USDA projected co-product prices from the price of corn. These costs are tabulated as a function of ethanol production quantity in the PMM input file WETOHIN. This file also contains the capital cost, conversion cost (non-energy operating cost), and energy cost data. Linear interpolations are performed to calculate intermediate yearly values for the quantity of energy consumed at the ethanol plant, and the net feedstock price for each ethanol quantity step. Once the data is read and the interpolations are performed, the ethanol price is calculated from the following equation:

$$PETOH_{cd,t,e} = FC_{t,e} + CAPCST*CCF + OPCST + QEN_t*PEN_{cd,t} - STSUB_{cd} \quad \textbf{(I-1)}$$

where:

$PETOH_{cd,t,e}$ = Delivered price of ethanol produced in Census Division *cd* in year *t* for volume step *e* ($/gal),

$FC_{t,e}$ = Feedstock corn production cost in year *t* for volume step *e* ($/gal),

$CAPCST$ = Capital cost for conversion technology ($/gal),

CCF = Capital cost factor (dimensionless),

$OPCST$ = Operating costs, exclusive of energy ($/gal),

QEN_t = Quantity of energy needed to convert corn to ethanol in year *t*

[2]Wang, M. et al 1997. "Fuel-Cycle Fossil Energy Use and Greenhouse Gas Emissions of Fuel Ethanol Produced from U.S. Midwest Corn".

(MMBtu/gal),

$PEN_{cd,t}$ = Price of energy used in the corn-to-ethanol conversion process in Census Division *cd* in year *t* ($/MMBtu),

$STSUB_{cd}$ = State incentive for ethanol production in Census Division *cd* ($/gal).

The price of energy on the first two steps of the supply curve (current production) is calculated assuming 60 percent of the existing plants are wet mills (consuming 20 percent natural gas and 80 percent coal), and 40 percent are dry mills (consuming 50 percent natural gas and 50 percent coal). The price of energy for the last two steps of the supply curve is calculated assuming all new facilities will be dry mill, consuming 50 percent natural gas and 50 percent coal. The price of energy in each case is calculated as follows:

Steps 1 and 2:

$$PEN_{cd,t} = (0.32*PNGIN_{cd,t}) + (0.68*PCLIN_{cd,t}) \quad \textbf{(I-2)}$$

Steps 3 and 4:

$$PEN_{cd,t} = (0.50*PNGIN_{cd,t}) + (0.50*PCLIN_{cd,t}) \quad \textbf{(I-3)}$$

where:

$PNGIN_{cd,t}$ = Industrial price of natural gas for Census Division *cd* in year *t* ($/MMBtu),

$PCLIN_{cd,t}$ = Industrial price of coal for Census Division *cd* in year *t* ($/MMBtu).

The capital cost factor (CCF) used in equation I-1, which is based on a 8-year amortization period, is calculated as follows:

$$CFF = MC\_RMPUAANS_t * (1 + MC\_RMPUAANS_t)^8 / ((1 + MC\_RMPUAANS_t)^8 - 1) \quad \textbf{(I-4)}$$

where:

$MC\_RMPUAANS_t$ = yield on AA-grade utility bonds in year *t* (a Macroeconomic Activity Module output variable).

Inventory of Variables, Data, and Parameters

MODEL INPUT: *CAPCST*

DEFINITION: Capital cost for conversion technology for crop *I* in year *t*.

The current value is $1.00 per gallon on steps one and two, and $2.00 per gallon on steps three and four of the supply curves. Costs are the same for all years. Located in the WETOHIN input data file.

SOURCE: Walsh, M. et al 1997. *Evolution of the Fuel Ethanol Industry: Feedstock Availability and Price*. Oak Ridge National Laboratory, Oak Ridge, TN.

MODEL INPUT: *OPCST*

DEFINITION: Operating costs, exclusive of energy, for conversion technology of corn.

Value is $0.27/gal. for 2000 thru 2020. Located in the WETOHIN input data file.

SOURCE: USDA/ERS. 1988. *Ethanol: Economic and Policy Tradeoffs*. Agricultural Economic Report No. 585. Resources and Technology Division, Economic Research Service, U.S. Department of Agriculture, Washington, D.C.

MODEL INPUT: *PCLIN*

DEFINITION: Price of coal for industrial use in Census Division *cd* in year *t*.

Located in the Price common block, (MPBLK).

SOURCE: Generated by the Coal Market Model.

MODEL INPUT: *PNGIN*

DEFINITION: Price of natural gas for industrial use in Census Division *cd* in year *t*.

Located in the Price common block, (MPBLK).

SOURCE: Generated by the Natural Gas Transmission and Distribution Model.

MODEL INPUT: *QEN*
DEFINITION: Quantity of energy needed to convert corn to ethanol in year *t*.

Values, in million Btu per gallon, are as follows: 0.050 in 1990, 0.041 in 2000, 0.037 in 2005, 0.035 in 2020. This decreasing trend is based on the assumption that energy required decreases linearly over time. Located in the WETOHIN input data file.

SOURCE: Marland, G. and A.F. Turhollow. 1991. "CO_2 Emissions From the Production and Combustion of Fuel Ethanol from Corn." *Energy*, 16(11/12):1307-1316.

I.3 Cellulose-Based Ethanol

Theoretical Approach

The cellulose ethanol model also uses a process costing approach to model the impacts of net feedstock production costs plus the capital, operating, and process energy costs associated with converting the corn feedstocks to ethanol. As with the corn model, each of the above factors contributes a part of the total price of ethanol.

Biomass feedstock supply is not modeled in the Petroleum Market Model ethanol model. Biomass price/quantity data are obtained from the Renewable Fuels Model of NEMS and are used as input to the ethanol model. The "Model Documentation: Renewable Fuels Module of the National Energy Modeling System", DOE/EIA-M069(2000) contains a complete description of the approach and assumptions used in generating the biomass feedstock supply functions.

Briefly, the biomass use in NEMS is modeled as two distinct markets, the captive and non-captive biomass markets. The captive market pertains to users with dedicated biomass supplies that obtain energy by burning biomass byproducts resulting from the manufacturing process. The noncaptive market is defined to include the commercial, transportation, and electric utility sectors, as well as the resources marketed in the industrial sector. There is an additional noncaptive market serving residential uses of biomass.

EIA developed a fairly simple model structure consisting of one supply schedule per region. This schedule defines the quantity and cost relationships of biomass resources accessible by all noncaptive, non-residential consumers. It is based on an aggregation of supply/price information from U.S. Forest Service and forest product experts. The wood portion of the cost-supply schedule is static throughout the model period. Energy crop cost-supply schedules are also developed and superimposed onto the wood total.

Fundamental Assumptions

A basic assumption for the biomass feedstock is that the supply price for noncaptive biomass energy is the same across all sectors. Biomass feedstock costs are input from the NEMS Renewable Fuels Model at the Census Division level. Biomass usage by the PMM ethanol model is fed back to the Renewable Fuels Model.

An important modeling consideration for cellulose ethanol production is the imposition of a constraint on the amount of ethanol production capacity assumed for the early years of the forecast. Ethanol from cellulose is relatively new technology and ethanol production from cellulose is currently at the demonstration level. A constraint on cellulose ethanol production prevents unrealistically large increases

in production capacity from occurring suddenly in response to favorable market prices. Cellulose ethanol production capacity is allowed to grow between 5 and 20 percent per year from 2004 to 2013, depending on region and technology assumptions. After 2013, production capacity growth is constrained by the marketed quantities of gasoline into which ethanol is blended.

In addition to feedstock prices and quantities input from the Renewable Fuels Model, the ethanol model requires feedstock conversion and energy cost data, and capital and operating cost data. The conversion and capital cost data were derived from the Oak Ridge National Laboratory Report *Evolution of the Fuel Ethanol Industry: Feedstock Availability and Price*, Marie Walsh et.al., June, 1997 for year 2000 as follows:

Capital Cost	0.398 $/gal
Operating Cost	0.269 $/gal
Power Credit	-0.090 $/gal
Transportation Cost	0.048 $/gal
Total Conversion Cost	0.625 $/gal

The *AEO2002* reference case assumes that cellulose conversion technology will improve over time such that full utilization of hemicellulose sugars, combined with modest reductions in cellulase enzyme costs will yield a savings of $0.18 per gallon over year 2000 cost by 2015.[3] A high-technology case, used for the *AEO2002* High Renewables side case, assumes a reduction of $0.35 per gallon over the same period. A low-technology case assumes a reduction of $0.08 per gallon by 2015. Conversion costs are constant in all cases from 2015 to 2020. Ethanol production costs are assumed to be constant across the United States. However, feedstock availability and price varies from Census Division to Census Division. The largest growth in ethanol production is expected in Census Divisions 3 and 4 where Midwestern corn stover would be desirable raw material because of its large volume, competitive price and proximity to current ethanol production plants. The feedstocks available in Census Division 9 are forest residue and rice straw. Feedstock conversion efficiency also improves over the forecast. Under the reference case, the yield increases from 85 gallons per ton in 2000 to 103 gallons per ton by 2015. The low-technology and high-technology yields in 2015 are assumed to be 85 gallons per ton and 120 gallons per ton, respectively. Currently, most State producer and tax incentives (limited by production volume) are at their maximum. Therefore, no State subsidies are assumed in the cellulose ethanol supply/price curves. As with corn-based ethanol, the ethanol blender's Federal excise tax credit, which is currently $0.053 per gallon of gasohol (10 percent ethanol, 90 percent gasoline), is modeled in the PMM.

Key Computations and Equations

The main computations performed by the cellulose portion of the ethanol model involve the derivation of an ethanol supply-price curve for each Census Division. The computations consist of three major steps:

[3]National Renewable Energy Laboratory 1999. *Bioethanol Multi-Year Technical Plan*.

1. Reading in ethanol component cost data from the PMM input file WETOHIN.
2. Obtaining biomass feedstock prices at the census division level from the Renewable Fuels Model.
3. Derivation of delivered ethanol prices, calculated as a function of the biomass feedstock price and the ethanol conversion costs.

Total Conversion Costs are calculated as follows:

$$TOTCONV_t = CAPCSTCL_t * CAPRSK_t + OPCSTCL_t + PWCRDCL_t + TRANSCL_t \quad \textbf{(I-5)}$$

where:

$TOTCONV_t$ = Total ethanol conversion cost for year *t*,

$CAPCSTCL_t$ = Capital cost for year *t*,

$CAPRSK_t$ = Risk premium required for new technology for year *t*

$OPCSTCL_t$ = Operating cost for year *t*,

$PWRCDCL_t$ = Power credit for co-products combusted and sold as power for year *t*

$TRANSCL_t$ = Transportation cost for year *t*

The delivered ethanol price equation is as follows:

$$PETOH_{cd,t} = FC_{cd,t} + TOTCONV_t \quad \textbf{(I-6)}$$

where:

$PETOH_{cd,y}$ = Delivered price of cellulose ethanol in Census Division *cd* in year *t*

$FC_{cd,y}$ = Biomass feedstock cost for Census Division *cd* in year *t*.

Inventory of Variables, Data, and Parameters

MODEL INPUT: *CAPCSTCL*

DEFINITION: Capital cost for conversion technology for cellulose ethanol.

Located in the WETOHIN input data file.

SOURCE: Marie Walsh et. al., *Evolution of the Fuel Ethanol Industry: Feedstock Availability and Price*. Oak Ridge National Laboratory, June 1997.

MODEL INPUT: *CAPRSK*

DEFINITION: Risk premium that investors require to invest in new technology.

Located in PMM FORTRAN source file refeth.f. Assumed to be zero for *AEO2002*.

MODEL INPUT: *OPCSTCL*

DEFINITION: Operating cost for conversion technology for cellulose ethanol.

Located in the WETOHIN input data file.

SOURCE: Marie Walsh et. al., *Evolution of the Fuel Ethanol Industry: Feedstock Availability and Price*. Oak Ridge National Laboratory, June 1997.

MODEL INPUT: *PWRCDCL*

DEFINITION: Power credit for co-products combusted and sold as power.

Located in the WETOHIN input data file.

SOURCE: Marie Walsh et. al., *Evolution of the Fuel Ethanol Industry: Feedstock Availability and Price*. Oak Ridge National Laboratory, June 1997.

MODEL INPUT: *TRANSCL*

DEFINITION: Transportation cost for cellulose ethanol.

Located in the WETOHIN input data file.

MODEL INPUT: *FC*

DEFINITION: Biomass feedstock cost for Census Division *cd* in year *y*.

Biomass feedstock costs are input from the Renewable Fuels Model under the variable PBMET.

SOURCE: National Energy Modeling System common block WRENEW.