Programming Instructions for
Revenue Assurance Premium Calculations
For 2002

American Farm Bureau Insurance Services, Inc

September 14, 2001

RA Programming Instruction Changes

Date	Change Made
May 14, 2001	Original Release
May 18, 2001	Modified the per-acre enterprise unit premium calculations (Equations 14, 15 and 16) to incorporate the Short Rate Factor. Modified the per-acre whole-farm unit premiums calculations (Equations 22, 23, and 24) to incorporate the Short Rate Factor.
August 30, 2001	Added a note to set *rateou* equal to the written agreement rate when a written agreement is in place. Added a note about 80% and 85% coverage for basic and optional units being allowed where that coverage is available under the APH program for spring planted crops.
September 14, 2001	Modified the basic unit premium calculations (Equations 6, 7, and 8) to include the SR, FN and FO factors. Modified the per-acre enterprise unit premium calculations (Equations14, 15, and 16) to incorporate the SR, FN and FO factors. Modified the per-acre whole-farm unit premiums calculations (Equation 22, 23, and 24), to incorporate the SR, FN and FO factors.

1. Introduction

This document contains detailed instructions for calculating Revenue Assurance (RA) premiums in the 2002 crop year including fall crops planted in 2001. Unless otherwise indicated, results of arithmetic calculations should be rounded to 9 digits to the right of the decimal.

2. Data and Variable Definitions

Much of the data that is required for calculation of RA‘ premiums can be found on the actuarial pages of the APH program. A subset of the APH data will need to be found on the new RA actuarial page. The definitions of the data that are to be located on the RA actuarial page are given below. The variable names will be indexed by *j* for a crop and *i* for a unit of the crop in the equations. For example, *rate(j,i)* refers to the base premium unit rate for crop *j*, unit *i* where *j = c, s, w, cn, sf, b, ww* refers to crop corn, soybeans, spring wheat, canola, sunflower, barley, winter wheat..

yldREF	The reference yield for a crop in a county.
rateou	The base premium rate for an optional unit for the APH program at the 65% coverage level calculated using premium rate using the first eight steps of the continuous rating method. This rate will include the five percent surcharge for cupped yields (if appropriate), any adjustments required for high risk land involving the high risk land rating factor (associated Map Area, M13, rectype 11, field 20, Map Area, pos. 98), and the winter wheat endorsement factor.
rate	The base premium rate for a basic unit in the APH program to be used in the RA rating equations for all units.
srfctr	Combine the SR, FN, and FO option factors by multiplying them together and round to three decimals.
warate	Written agreement rate. This rate replaces *rateou* in determining RA premiums when a written agreement is in place on a unit.
minratefactor	A factor used to determine the minimum whole-farm premium rate
PP65	The APH prevented planting factor for a crop for 65% prevented planting coverage (option code PF, M13, rectype 11, field 46, Common Option Codes, pos 259)
PP70	The APH prevented planting factor for a crop for 70% prevented planting coverage (option code PT, M13, rectype 11, field 46, Common Option Codes, pos 259)

Other data used to calculate premiums are supplied by the farmer, supplied by the insurance agent, or supplied by the program. Data that is specific to each unit (basic or optional) is given below:

fyld Approved APH yield for the basic (or optional) unit following standard APH procedures. This yield is used to calculate coverage levels and revenue guarantees. (60% APH Adjusted Election, when applicable.) It is calculated following APH procedures that account for the effects of including yield floors, cups, caps, and possible substitution of T-yields. (M13 rectype 11, field 28, Yield, pos 128)

acre Acres in the crop on the basic (or optional) unit (M13 rectype 11, field 34, reported acres, pos 176)

share The farmer's share on a basic (or optional) unit of a crop (M13, rectype 11, field 38, insured share, pos. 210)

revb The selected per-acre revenue level for a basic (or optional) unit of a crop (M13, rectype 11, field 29, Dollar Amount of Insurance, pos. 138)

reve The selected per-acre revenue level for an enterprise unit (M13, rectype 11, field 29, Dollar Amount of Insurance, pos. 138)

revwf The selected per-acre revenue level for the whole-farm unit (M13, rectype 11, field 29, Dollar Amount of Insurance, pos. 138)

nsect Number of sections in which a crop is grown (M13, rectype 11, field 68, Number of sections, pos. 417)

The prices that are supplied by RMA are:

chip The projected harvest price of a crop (M13, rectype 11, field 36, Price Election Amount, pos. 194)

And the price volatilities that are supplied by RMA are:

cvp Price volatility of the crop

Variables that are either calculated by the program or supplied by the user and directly used to calculate premiums are:

cover Coverage level on a basic (or optional) unit (M13, rectype 11, field 31, Coverage Level Percent, pos. 158)

ecover Coverage level on an enterprise unit (M13, rectype 11, field 31, Coverage Level Percent, pos. 158)

covwf Coverage level on a whole-farm unit (M13, rectype 11, field 31, Coverage Level Percent, pos. 158)

Some other variables that are calculated by the program are:

premr Base premium rate for a basic (or optional) unit (M13, rectype 11, field 42, Base Premium Rate, pos. 233)

epremr Base premium rate for an enterprise unit (M13, rectype 11, field 42, Base Premium Rate, pos. 233)

wfpremr	Base premium rate for a whole-farm unit (M13, rectype 11, field 42, Base Premium Rate, pos. 233)
wfprem	Base per-acre premium for a whole-farm unit
avgrate	Weighted average BPR rate for an enterprise unit
erate	Adjusted average BPR rate for an enterprise unit
efyld	Weighted average APH yield for an enterprise unit
perlia	Percent of expected liability from a crop on a whole-farm unit
LP	Per-acre loaded premium for a basic (or optional) unit (M13 rectype 11, field 44, Loaded Premium Per Acre, pos. 249)
TLP	Total loaded premium for a basic (or optional) unit (M13 rectype 11, field 55, Total Premium, pos. 346)
LEP	Per-acre enterprise premium for a crop (M13 rectype 11, field 44, Loaded Premium Per Acre, pos. 249)
TLEP	Total loaded enterprise premium for a crop (M13, rectype 11, field 55, Total Premium, pos. 346)
LWFP	Per-acre loaded whole-farm unit premium (M13 rectype 11, field 44, Loaded Premium Per Acre, pos. 249)
TLWFP	Total loaded whole-farm unit premium (M13, rectype 11, field 55, Total Premium, pos. 346)
subfact	Premium subsidy factor on an optional or basic unit
subfacte	Premium subsidy factor on an enterprise unit
subfactwf	Premium subsidy factor on a whole-farm unit
psub	Premium subsidy on a basic (or optional) unit
psube	Premium subsidy on an enterprise unit
psubwf	Premium subsidy on a whole-farm unit
TLPsub	Subsidized premium (M13, rectype 11, field 61, Producer Premium, pos. 338)
TLEPsub	Subsidized enterprise premium (M13, rectype 11, field 61, Producer Premium, pos. 338)
WFPsub	Subsidized whole-farm premium (M13, rectype 11, field 61, Producer Premium, pos.338)
premb	Producer paid premium per acre for a basic (or optional) unit
preme	Producer paid premium per acre for an enterprise unit
premwf	Producer paid premium per acre for a whole-farm unit

3. Available Coverage Amounts

Revenue Assurance offers revenue guarantees that are 65%, 70%, and 75% of the product of projected harvest price and approved yield for basic and optional units. In addition, Revenue Assurance offers revenue guarantees that are 80% and 85% of the

product of projected harvest price and approved yield for spring planted crops in counties where 80% and 85% coverage levels are available under the APH program. These high coverage levels are not allowed for winter wheat basic and optional units.

Basic (or optional) units

The farmer selects a coverage level of 65%, 70%, or 75%, or, if allowed, 80% or 85%. The values for the per-acre revenue guarantees (rounded to the nearest cent) are then calculated for all crops $j = c, s, w, cn, sf, b, ww$ and all units i, $i = 1,\ldots,N(j)$:

(1)
$$revb(j,i) = cover(j) \cdot fyld(j,i) \cdot chip(j), i = 1,...,N(j), j = c,s,w,cn,sf,b,ww.$$

Enterprise units

The farmer selects a coverage level of 65%, 70%, 75%, 80%, or 85%. The values for the per-acre revenue guarantees (rounded to the nearest cent) are then calculated for all crops $j = c, s, w, cn, sf, b, ww$.

(2)
$$reve(j) = ecover(j) \frac{chip(j)\sum_{i=1}^{N(j)} share(j,i)acre(j,i)fyld(j,i)}{\sum_{i=1}^{N(j)} share(j,i)acre(j,i)}, j = c,s,w,cn,sf,b,ww.$$

Whole-farm unit

The farmer selects a coverage level of 65%, 70%, 75%, 80%, or 85%. The value for the per-acre revenue guarantee (rounded to the nearest cent) is then calculated. If less than the maximum number of crops are insured in the whole-farm unit (because a farmer does not plant a covered crop in a county) then the summations are done only over the included crops.

(3)
$$revwf = coverwf \left[\sum_{j=c,s,w,cn,sf,b} \frac{chip(j)\sum_{i=1}^{N(j)} share(j,i)acre(j,i)fyld(j,i)}{\sum_{j=c,s,w,cn,sf,b} \sum_{i=1}^{N(j)} share(j,i)acre(j,i)} \right]$$

4. Basic Unit Premiums

For each basic (or optional) unit the farmer supplies the state and county where the insured crops reside and values for *fyld*, and *cover*. In addition, the farmer decides whether or not to choose the harvest price option (M13 rectype 11, RA Fall Harvest Price option, pos. 268). The state, county and whether the farmer chooses the harvest price

option identifies which set of rating coefficients to use in equation (9). All single crop rating coefficients can be found in an Excel spreadsheet that accompanies this document. The counties that should be allocated to Southern Minnesota, Northern Minnesota, in which they apply are given in the Appendix.

The program should provide (or, alternatively, the user should supply) *yldREF* and *rateou*. YldREF is found on the FCI-35 page for RA. *rateou* is calculated using the steps used to calculate base premium rates for an optional unit under the APH program. These values must be allowed to vary by type and practice to account for the situation where a basic unit has more than one type of practice. The per-acre base premiums are then calculated using long, but straightforward, formulas.

The optional unit rate must first be multiplied by 0.9 as in equation (8) to calculate to determine *rate(j,i).* The result is rounded to eight digits.

(4) $$rate(j,i) = 0.9 \cdot rateou(j,i) \cdot, \; i = 1,...,N(j); ; \; j = c,s,w,cn,sf,b,ww$$

Base premium rate*

(5)
$$\begin{aligned} premr(j,i) = {} & beta(j,0) + beta(j,1)rate(j,i) + beta(j,2)rate(j,i)^2 + beta(j,3)cover(j) \\ & + beta(j,4)cover(j)^2 + beta(j,5)\frac{fyld(j,i)}{yldREF(j)} + beta(j,6)\left(\frac{fyld(j,i)}{yldREF(j)}\right)^2 \\ & + beta(j,7)cvp(j) + beta(j,8)cvp(j)^2 + beta(j,9)rate(j,i) \cdot cover(j) \\ & + beta(j,10)rate(j,i) \cdot \frac{fyld(j,i)}{yldREF(j)} + beta(j,11)rate(j,i) \cdot cvp(j) \\ & + beta(j,12)cover(j) \cdot \frac{fyld(j,i)}{yldREF(j)} + beta(j,13)cover(j) \cdot cvp(j) \\ & + beta(j,14)cvp(j) \cdot \frac{fyld(j,i)}{yldREF(j)} \quad i = 1,...,N(j); \; j = c,s,w,cn,sf,b,ww. \end{aligned}$$

This formula is used to calculate the base premium rates for each basic unit and for each optional unit for each crop.

Each individual calculation is rounded to 9 digits to the right of the decimal place, and the variable *premr* is rounded to 4 digits.

The next step is to add a prevented planting load to these base premiums and find the per-acre premiums.

*NOTE – In the case of a written agreement, use the written agreement rate in place of *rateou* in equation (4).

Loaded per-acre premiums

The base premium rates are increased for prevented planting coverage if the farmer opts for 65% or 70% prevented planting coverage. The per-acre premium is found by multiplying the premium rate by liability and rounding to two decimals.

At 60% prevented planting

(6) $$LP(j,i) = premr(j,i) \cdot revb(j,i) \cdot srfctr(j,i); \quad j = c,s,w,cn,sf,b,ww.$$

At 65% prevented planting

(7) $$LP(j,i) = premr(j,i) \cdot PP65(j) \cdot revb(j,i) \cdot srfctr(j,i); \quad j = c,s,w,cn,sf,b,ww.$$

At 70% prevented planting

(8) $$LP(j,i) = premr(j,i) \cdot PP70(j) \cdot revb(j,i) \cdot srfctr(j,i); \quad j = c,s,w,cn,sf,b,ww.$$

Total basic unit premiums

The total premium for a basic unit is given by the equation (13). The premium is rounded to the nearest whole-dollar amount.

(9)

$$TLP(j,i) = LP(j,i) \cdot acre(j,i) \cdot share(j,i), \; i = 1,\ldots,N(j); \quad j = c,s,w,cn,sf,b,ww.$$

Total optional unit premiums

The per-acre premium for an optional unit is found by treating the optional unit as a basic unit and then applying a 10% surcharge for all crops. This 10% surcharge applies to all RA crops. They are rounded to the nearest whole-dollar amount.

(10)

$$TLP(j,i) = 1.1LP(j,i) \cdot acre(j,i) \cdot share(j,i), i = 1,\ldots,N(j); \; j = c,s,w,cn,sf,b,ww.$$

5. Enterprise Unit Premiums

The premium for an enterprise unit is found by using the same coefficients that are used to find premiums for basic or optional units. Differences in the rating equations arise if a farmer has more than one basic unit or farms in more than one section of land. These two factors change the approved farm yield and APH rate used in the equations.

Before the premiums can be calculated, *avgrate,* and *efyld* must be calculated. These quantities are simply the acreage and share weighted average of the APH yields and APH premium rates for all units of a crop in a county.

$$(11) \qquad avgrate(j) = \frac{\sum_{i=1}^{N(j)} share(j,i)acre(j,i)rate(j,i)}{\sum_{i=1}^{N(j)} share(j,i)acre(j,i)} \quad j = c,s,w,cn,sf,b,ww.$$

$$(12) \qquad efyld(j) = \frac{\sum_{i=1}^{N(j)} share(j,i)acre(j,i)fyld(j,i)}{\sum_{i=1}^{N(j)} share(j,i)acre(j,i)} \quad j = c,s,w,cn,sf,b,ww.$$

avgrate is rounded to 9 digits to the right of the decimal. *efyld* is rounded to one digit to the right of the decimal.

We then need to adjust *avgrate* to reflect the number of sections.

$$erate(c) = avgrate(c) \cdot (1 - (nsect(c) - 1)\frac{0.4}{9}) \quad \text{if } nsect(c) \le 10,$$
$$\text{else} \quad erate(c) = 0.6avgrate(c)$$

$$erate(j) = avgrate(j) \cdot (1 - (nsect(j) - 1)\frac{0.5}{9}) \quad \text{if } nsect(j) \le 10, \quad ; j = s,w,cn,sf,b,ww.$$
$$\text{else} \quad erate(j) = 0.5avgrate(j)$$

erate is rounded to 4 digits.

And finally, if a farmer has multiple practices across within or between units then the values for *yldREF* to be used in equation (17) is the maximum applicable value.

Base premium rate for enterprise unit

(13)

$$epremr(j) = \text{beta}(j,0) + \text{beta}(j,1)erate(j) + \text{beta}(j,2)erate(j)^2 + \text{beta}(j,3)ecover(j)$$
$$+ \text{beta}(j,4)ecover(j)^2 + \text{beta}(j,5)\frac{efyld(j)}{yldREF(j)} + \text{beta}(j,6)(\frac{efyld(j)}{yldREF(j)})^2 + \text{beta}(j,7)cvp(j)$$
$$+ \text{beta}(j,8)cvp(j)^2 + \text{beta}(j,9)erate(j) \cdot ecover(j) + \text{beta}(j,10)erate(j) \cdot \frac{efyld(j)}{yldREF(j)}$$
$$+ \text{beta}(j,11)erate(j) \cdot cvp(j) + \text{betac}(j,12)ecover(j) \cdot \frac{efyld(j)}{yldREF(j)}$$
$$+ \text{betac}(j,13)ecover(j) \cdot cvp(j) + \text{beta}(j,14) \cdot \frac{efyld(j)}{yldREF(j)} \cdot cvp(j), j = c,s,w,cn,sf,b,ww.$$

Each individual calculation is rounded to 9 digits, and the variable *epremr* is rounded to 4 digits.

Per-acre enterprise unit premiums

The next step is to convert these rates into per-acre base premiums. This is done by multiplying the rate by the appropriate prevented planting factor and liability (the per-acre revenue guarantee for enterprise units) and rounding to two digits. An additional adjustment may be made when a short rate factor exists. An additional adjustment may be made when a short rate factor exists. Use the short rate factor *srfctr(j,i)* as defined by the Risk Management Agency.

At 60% prevented planting

(14) $$LEP(j,i) = epremr(j) \cdot reve(j) \cdot srfctr(j,i),\ j = c,s,w,cn,sf,b,ww.$$

At 65% prevented planting

(15) $$LEP(j,i) = epremr(j) \cdot PP65(j) \cdot reve(j) \cdot srfctr(j,i),\ j = c,s,w,cn,sf,b,ww.$$

At 70% prevented planting

(16) $$LEP(j,i) = epremr(j) \cdot PP70(j) \cdot reve(j) \cdot srfctr(j,i),\ j = c,s,w,cn,sf,b,ww.$$

Total loaded enterprise premiums

Now we need to multiply the loaded per-acre premium by the number of insured acres and share on the each unit. The result will be rounded to whole-dollars. The total enterprise premium is found by summing over all units. This will be rounded on a record by record basis (M13 ,rectype 13, field 55, total premium, pos 346.

(17) $$TLEP(j) = \sum_{i=1}^{N(j)} \text{round}(LEP(j) \cdot acre(j,i) share(j,i), 0);\ j = c,s,w,cn,sf,b,ww.$$

6. Whole-Farm Unit Premium

Calculation of whole-farm premium follows the same procedure as calculation of premium for the other unit structures. However, because there are up to six crops involved, the equations for whole-farm premiums are significantly longer. To facilitate programming, the rating coefficients and rating factors (variables) that are multiplied together and then added to come up with the whole-farm premium are presented as columns below.

The values for the coefficients in **betawf** depend on which crops are in the whole-farm unit and on whether the farmer chooses the harvest price option. Thus, a state that has corn, soybeans, and wheat eligible for whole-farm RA coverage will have eight sets of coefficients: two for a corn-soybean whole-farm unit, two for a corn-wheat whole-farm unit, two for a soybean-wheat whole-farm unit, and two for a corn-soybean-wheat whole-farm unit. There are eight sets each for Iowa, Southern Minnesota, Northern Minnesota, Eastern South Dakota, Western South Dakota, and Idaho. There are two sets of coefficients for states with two crops eligible for whole-farm RA coverage, which includes Arkansas, Colorado, Illinois, Indiana, Kansas, Kentucky, Michigan, Missouri, Ohio, Oklahoma, and Tennessee. For North Dakota, which has six crops eligible for RA coverage, there are a total of 114 sets of whole-farm rating coefficients. The sets of whole-farm coefficients are given in an Excel spreadsheet that accompanies these programming instructions.

There are six additional rating factors used to calculate whole-farm rates. These are *perlia*(*j*), which is calculated as

$$perlia(j) = \frac{\sum_{i=1}^{N_j} chip(j)fyld(j,i)acre(j,i)share(j,i)}{\sum_{j=1}^{6}\sum_{i=1}^{N_j} chip(j)fyld(j,i)acre(j,i)share(j,i)} \quad j = c,s,w,cn,sf,b. \tag{18}$$

perlia should be rounded to four digits. If a crop is not grown, then set all *acre(j,i)* for that crop equal to zero in this equation.

Whole-farm base premium rate

Table 1. Whole-farm rating coefficients and rating factors (variables).

Coefficient	Variable
betawf(0)	1.0
betawf(1)	erate(c)
betawf(2)	erate(s)
betawf(3)	erate(w)
betawf(4)	erate(cn)
betawf(5)	erate(sf)
betawf(6)	erate(b)
betawf(7)	$erate(c)^2$
betawf(8)	$erate(s)^2$
betawf(9)	$erate(w)^2$
betawf(10)	$erate(cn)^2$
betawf(11)	$erate(sf)^2$
betawf(12)	$erate(b)^2$
betawf(13)	erate(c) x erate(s)
betawf(14)	erate(c) x erate(w)
betawf(15)	erate(c) x erate(cn)
betawf(16)	erate(c) x erate(sf)
betawf(17)	erate(c) x erate(b)
betawf(18)	erate(s) x erate(w)
betawf(19)	erate(s) x erate(cn)
betawf(20)	erate(s) x erate(sf)
betawf(21)	erate(s) x erate(b)
betawf(22)	erate(w) x erate(cn)
betawf(23)	erate(w) x erate(sf)
betawf(24)	erate(w) x erate(b)
betawf(25)	erate(cn) x erate(sf)
betawf(26)	erate(cn) x erate(b)
betawf(27)	erate(sf) x erate(b)
betawf(28)	covwf
betawf(29)	$covwf^2$
betawf(30)	covwf x erate(c)
betawf(31)	covwf x erate(s)
betawf(32)	covwf x erate(w)
betawf(33)	covwf x erate(cn)
betawf(34)	covwf x erate(sf)
betawf(35)	covwf x erate(b)

betawf(36) perlia(c)
betawf(37) perlia(s)
betawf(38) perlia(w)
betawf(39) perlia(cn)
betawf(40) perlia(sf)
betawf(41) perlia(b)
betawf(42) perlia(c)^2
betawf(43) perlia(s)^2
betawf(44) perlia(w)^2
betawf(45) perlia(cn)^2
betawf(46) perlia(sf)^2
betawf(47) perlia(b)^2
betawf(48) perlia(c)^3
betawf(49) perlia(s)^3
betawf(50) perlia(w)^3
betawf(51) perlia(cn)^3
betawf(52) perlia(sf)^3
betawf(53) perlia(b)^3
betawf(54) perlia(c) x erate(c)
betawf(55) perlia(c) x erate(s)
betawf(56) perlia(c) x erate(w)
betawf(57) perlia(c) x erate(cn)
betawf(58) perlia(c) x erate(sf)
betawf(59) perlia(c) x erate(b)
betawf(60) perlia(s) x erate(c)
betawf(61) perlia(s) x erate(s)
betawf(62) perlia(s) x erate(w)
betawf(63) perlia(s) x erate(cn)
betawf(64) perlia(s) x erate(sf)
betawf(65) perlia(s) x erate(b)
betawf(66) perlia(w) x erate(c)
betawf(67) perlia(w) x erate(s)
betawf(68) perlia(w) x erate(w)
betawf(69) perlia(w) x erate(cn)
betawf(70) perlia(w) x erate(sf)
betawf(71) perlia(w) x erate(b)
betawf(72) perlia(cn) x erate(c)
betawf(73) perlia(cn) x erate(s)
betawf(74) perlia(cn) x erate(w)
betawf(75) perlia(cn) x erate(cn)
betawf(76) perlia(cn) x erate(sf)
betawf(77) perlia(cn) x erate(b)
betawf(78) perlia(sf) x erate(c)
betawf(79) perlia(sf) x erate(s)

betawf(80) perlia(sf) x erate(w)
betawf(81) perlia(sf) x erate(cn)
betawf(82) perlia(sf) x erate(sf)
betawf(83) perlia(sf) x erate(b)
betawf(84) perlia(b) x erate(c)
betawf(85) perlia(b) x erate(s)
betawf(86) perlia(b) x erate(w)
betawf(87) perlia(b) x erate(cn)
betawf(88) perlia(b) x erate(sf)
betawf(89) perlia(b) x erate(b)
betawf(90) perlia(c)^2 x erate(c)
betawf(91) perlia(c)^2 x erate(s)
betawf(92) perlia(c)^2 x erate(w)
betawf(93) perlia(c)^2 x erate(cn)
betawf(94) perlia(c)^2 x erate(sf)
betawf(95) perlia(c)^2 x erate(b)
betawf(96) perlia(s)^2 x erate(c)
betawf(97) perlia(s)^2 x erate(s)
betawf(98) perlia(s)^2 x erate(w)
betawf(99) perlia(s)^2 x erate(cn)
betawf(100) perlia(s)^2 x erate(sf)
betawf(101) perlia(s)^2 x erate(b)
betawf(102) perlia(w)^2 x erate(c)
betawf(103) perlia(w)^2 x erate(s)
betawf(104) perlia(w)^2 x erate(w)
betawf(105) perlia(w)^2 x erate(cn)
betawf(106) perlia(w)^2 x erate(sf)
betawf(107) perlia(w)^2 x erate(b)
betawf(108) perlia(cn)^2 x erate(c)
betawf(109) perlia(cn)^2 x erate(s)
betawf(110) perlia(cn)^2 x erate(w)
betawf(111) perlia(cn)^2 x erate(cn)
betawf(112) perlia(cn)^2 x erate(sf)
betawf(113) perlia(cn)^2 x erate(b)
betawf(114) perlia(sf)^2 x erate(c)
betawf(115) perlia(sf)^2 x erate(s)
betawf(116) perlia(sf)^2 x erate(w)
betawf(117) perlia(sf)^2 x erate(cn)
betawf(118) perlia(sf)^2 x erate(sf)
betawf(119) perlia(sf)^2 x erate(b)
betawf(120) perlia(b)^2 x erate(c)
betawf(121) perlia(b)^2 x erate(s)
betawf(122) perlia(b)^2 x erate(w)
betawf(123) perlia(b)^2 x erate(cn)

betawf(124) perlia(b)^2 x erate(sf)
betawf(125) perlia(b)^2 x erate(b)
betawf(126) perlia(c)^2 x covwf
betawf(127) perlia(s)^2 x covwf
betawf(128) perlia(w)^2 x covwf
betawf(129) perlia(cn)^2 x covwf
betawf(130) perlia(sf)^2 x covwf
betawf(131) perlia(b)^2 x covwf
betawf(132) perlia(c)^3 x covwf
betawf(133) perlia(s)^3 x covwf
betawf(134) perlia(w)^3 x covwf
betawf(135) perlia(cn)^3 x covwf
betawf(136) perlia(sf)^3 x covwf
betawf(137) perlia(b)^3 x covwf
betawf(138) efyld(c)/yldREF(c)
betawf(139) efyld(s)/yldREF(s)
betawf(140) efyld(w)/yldREF(w)
betawf(141) efyld(cn)/yldREF(cn)
betawf(142) efyld(sf)/yldREF(sf)
betawf(143) efyld(b)/yldREF(b)
betawf(144) (efyld(c)/yldREF(c))^2
betawf(145) (efyld(s)/yldREF(s))^2
betawf(146) (efyld(w)/yldREF(w))^2
betawf(147) (efyld(cn)/yldREF(cn))^2
betawf(148) (efyld(sf)/yldREF(sf))^2
betawf(149) (efyld(b)/yldREF(b))^2
betawf(150) perlia(c)/perlia(s)
betawf(151) perlia(c)/perlia(w)
betawf(152) perlia(c)/perlia(cn)
betawf(153) perlia(c)/perlia(sf)
betawf(154) perlia(c)/perlia(b)
betawf(155) perlia(s)/perlia(w)
betawf(156) perlia(s)/perlia(cn)
betawf(157) perlia(s)/perlia(sf)
betawf(158) perlia(s)/perlia(b)
betawf(159) perlia(w)/perlia(cn)
betawf(160) perlia(w)/perlia(sf)
betawf(161) perlia(w)/perlia(b)
betawf(162) perlia(cn)/perlia(sf)
betawf(163) perlia(cn)/perlia(b)
betawf(164) perlia(sf)/perlia(b)
betawf(165) (perlia(c)/perlia(s))^2
betawf(166) ((perlia(c)/perlia(w))^2
betawf(167) (perlia(c)/perlia(cn))^2

betawf(168) $(\text{perlia(c)}/\text{perlia(sf)})^2$
betawf(169) $(\text{perlia(c)}/\text{perlia(b)})^2$
betawf(170) $(\text{perlia(s)}/\text{perlia(w)})^2$
betawf(171) $(\text{perlia(s)}/\text{perlia(cn)})^2$
betawf(172) $(\text{perlia(s)}/\text{perlia(sf)})^2$
betawf(173) $(\text{perlia(s)}/\text{perlia(b)})^2$
betawf(174) $(\text{perlia(w)}/\text{perlia(cn)})^2$
betawf(175) $(\text{perlia(w)}/\text{perlia(sf)})^2$
betawf(176) $(\text{perlia(w)}/\text{perlia(b)})^2$
betawf(177) $(\text{perlia(cn)}/\text{perlia(sf)})^2$
betawf(178) $(\text{perlia(cn)}/\text{perlia(b)})^2$
betawf(179) $(\text{perlia(sf)}/\text{perlia(b)})^2$
betawf(180) cvp(c)
betawf(181) cvp(s)
betawf(182) cvp(w)
betawf(183) cvp(cn)
betawf(184) cvp(sf)
betawf(185) cvp(b)
betawf(186) cvp(c)^2
betawf(187) cvp(s)^2
betawf(188) cvp(w)^2
betawf(189) cvp(cn)^2
betawf(190) cvp(sf)^2
betawf(191) cvp(b)^2
betawf(192) cvp(c) x erate(c)
betawf(193) cvp(c) x erate(s)
betawf(194) cvp(c) x erate(w)
betawf(195) cvp(c) x erate(cn)
betawf(196) cvp(c) x erate(sf)
betawf(197) cvp(c) x erate(b)
betawf(198) cvp(s) x erate(c)
betawf(199) cvp(s) x erate(s)
betawf(200) cvp(s) x erate(w)
betawf(201) cvp(s) x erate(cn)
betawf(202) cvp(s) x erate(sf)
betawf(203) cvp(s) x erate(b)
betawf(204) cvp(w) x erate(c)
betawf(205) cvp(w) x erate(s)
betawf(206) cvp(w) x erate(w)
betawf(207) cvp(w) x erate(cn)
betawf(208) cvp(w) x erate(sf)
betawf(209) cvp(w) x erate(b)
betawf(210) cvp(cn) x erate(c)
betawf(211) cvp(cn) x erate(s)

betawf(212) cvp(cn) x erate(w)
betawf(213) cvp(cn) x erate(cn)
betawf(214) cvp(cn) x erate(sf)
betawf(215) cvp(cn) x erate(b)
betawf(216) cvp(sf) x erate(c)
betawf(217) cvp(sf) x erate(s)
betawf(218) cvp(sf) x erate(w)
betawf(219) cvp(sf) x erate(cn)
betawf(220) cvp(sf) x erate(sf)
betawf(221) cvp(sf) x erate(b)
betawf(222) cvp(b) x erate(c)
betawf(223) cvp(b) x erate(s)
betawf(224) cvp(b) x erate(w)
betawf(225) cvp(b) x erate(cn)
betawf(226) cvp(b) x erate(sf)
betawf(227) cvp(b) x erate(b)
betawf(228) $cvp(c)^2$ x erate(c)
betawf(229) $cvp(c)^2$ x erate(s)
betawf(230) $cvp(c)^2$ x erate(w)
betawf(231) $cvp(c)^2$ x erate(cn)
betawf(232) $cvp(c)^2$ x erate(sf)
betawf(233) $cvp(c)^2$ x erate(b)
betawf(234) $cvp(s)^2$ x erate(c)
betawf(235) $cvp(s)^2$ x erate(s)
betawf(236) $cvp(s)^2$ x erate(w)
betawf(237) $cvp(s)^2$ x erate(cn)
betawf(238) $cvp(s)^2$ x erate(sf)
betawf(239) $cvp(s)^2$ x erate(b)
betawf(240) $cvp(w)^2$ x erate(c)
betawf(241) $cvp(w)^2$ x erate(s)
betawf(242) $cvp(w)^2$ x erate(w)
betawf(243) $cvp(w)^2$ x erate(cn)
betawf(244) $cvp(w)^2$ x erate(sf)
betawf(245) $cvp(w)^2$ x erate(b)
betawf(246) $cvp(cn)^2$ x erate(c)
betawf(247) $cvp(cn)^2$ x erate(s)
betawf(248) $cvp(cn)^2$ x erate(w)
betawf(249) $cvp(cn)^2$ x erate(cn)
betawf(250) $cvp(cn)^2$ x erate(sf)
betawf(251) $cvp(cn)^2$ x erate(b)
betawf(252) $cvp(sf)^2$ x erate(c)
betawf(253) $cvp(sf)^2$ x erate(s)
betawf(254) $cvp(sf)^2$ x erate(w)
betawf(255) $cvp(sf)^2$ x erate(cn)

betawf(256) cvp(sf)^2 x erate(sf)
betawf(257) cvp(sf)^2 x erate(b)
betawf(258) cvp(b)^2 x erate(c)
betawf(259) cvp(b)^2 x erate(s)
betawf(260) cvp(b)^2 x erate(w)
betawf(261) cvp(b)^2 x erate(cn)
betawf(262) cvp(b)^2 x erate(sf)
betawf(263) cvp(b)^2 x erate(b)
betawf(264) perlia(c) x cvp(c)
betawf(265) perlia(c) x cvp(s)
betawf(266) perlia(c) x cvp(w)
betawf(267) perlia(c) x cvp(cn)
betawf(268) perlia(c) x cvp(sf)
betawf(269) perlia(c) x cvp(b)
betawf(270) perlia(s) x cvp(c)
betawf(271) perlia(s) x cvp(s)
betawf(272) perlia(s) x cvp(w)
betawf(273) perlia(s) x cvp(cn)
betawf(274) perlia(s) x cvp(sf)
betawf(275) perlia(s) x cvp(b)
betawf(276) perlia(w) x cvp(c)
betawf(277) perlia(w) x cvp(s)
betawf(278) perlia(w) x cvp(w)
betawf(279) perlia(w) x cvp(cn)
betawf(280) perlia(w) x cvp(sf)
betawf(281) perlia(w) x cvp(b)
betawf(282) perlia(cn) x cvp(c)
betawf(283) perlia(cn) x cvp(s)
betawf(284) perlia(cn) x cvp(w)
betawf(285) perlia(cn) x cvp(cn)
betawf(286) perlia(cn) x cvp(sf)
betawf(287) perlia(cn) x cvp(b)
betawf(288) perlia(sf) x cvp(c)
betawf(289) perlia(sf) x cvp(s)
betawf(290) perlia(sf) x cvp(w)
betawf(291) perlia(sf) x cvp(cn)
betawf(292) perlia(sf) x cvp(sf)
betawf(293) perlia(sf) x cvp(b)
betawf(294) perlia(c)^2 x cvp(c)
betawf(295) perlia(c)^2 x cvp(s)
betawf(296) perlia(c)^2 x cvp(w)
betawf(297) perlia(c)^2 x cvp(cn)
betawf(298) perlia(c)^2 x cvp(sf)
betawf(299) perlia(c)^2 x cvp(b)

betawf(300)	perlia(s)^2 x cvp(c)
betawf(301)	perlia(s)^2 x cvp(s)
betawf(302)	perlia(s)^2 x cvp(w)
betawf(303)	perlia(s)^2 x cvp(cn)
betawf(304)	perlia(s)^2 x cvp(sf)
betawf(305)	perlia(s)^2 x cvp(b)
betawf(306)	perlia(w)^2 x cvp(c)
betawf(307)	perlia(w)^2 x cvp(s)
betawf(308)	perlia(w)^2 x cvp(w)
betawf(309)	perlia(w)^2 x cvp(cn)
betawf(310)	perlia(w)^2 x cvp(sf)
betawf(311)	perlia(w)^2 x cvp(b)
betawf(312)	perlia(cn)^2 x cvp(c)
betawf(313)	perlia(cn)^2 x cvp(s)
betawf(314)	perlia(cn)^2 x cvp(w)
betawf(315)	perlia(cn)^2 x cvp(cn)
betawf(316)	perlia(cn)^2 x cvp(sf)
betawf(317)	perlia(cn)^2 x cvp(b)
betawf(318)	perlia(sf)^2 x cvp(c)
betawf(319)	perlia(sf)^2 x cvp(s)
betawf(320)	perlia(sf)^2 x cvp(w)
betawf(321)	perlia(sf)^2 x cvp(cn)
betawf(322)	perlia(sf)^2 x cvp(sf)
betawf(323)	perlia(sf)^2 x cvp(b)
betawf(324)	perlia(b)^2 x cvp(c)
betawf(325)	perlia(b)^2 x cvp(s)
betawf(326)	perlia(b)^2 x cvp(w)
betawf(327)	perlia(b)^2 x cvp(cn)
betawf(328)	perlia(b)^2 x cvp(sf)
betawf(329)	perlia(b)^2 x cvp(b)

The whole-farm premium rate (*wfpremr*) is found by multiplying each coefficient by the corresponding value of the variable and then summing the results. Each individual calculation should be rounded to 9 digits. The sum should be rounded to 4 digits. If a crop is not used, care must be taken to avoid divide-by-zero errors in the rating variables.

Checking to See if Maximum Whole-Farm Discount is Exceeded

RA whole-farm premium rates cannot be less than *minratefactor* times the average premium rate had the producer bought enterprise unit coverage, where *minratefactor* = .5, if two crops are included in the whole-farm unit, = .475 if three crops, = .45 if four crops, = .425 if five crops, and = .4 if six crops are include. To determine if this limit has been exceeded we need to use the whole-farm coverage level, *covwf*, in the enterprise unit

premium equations for the crops in the whole-farm unit. The enterprise equations with *covwf* is reproduced below. Each individual calculation is rounded to 9 digits to the right of the decimal place, and the variable *epremrw* is rounded to 4 digits. *epremrw*(*j*) is rounded to four digits.

(19)

$$\begin{aligned} epremrw(j) \;=\; & \text{beta}(j,0) + \text{beta}(j,1)erate(j) + \text{beta}(j,2)erate(j)^2 + \text{beta}(j,3)covwf \\ & + \text{beta}(j,4)covwf^2 + \text{beta}(j,5)\frac{efyld(j)}{yldREF(j)} + \text{beta}(j,6)\left(\frac{efyld(j)}{yldREF(j)}\right)^2 + \text{beta}(j,7)cvp(j) \\ & + \text{beta}(j,8)cvp(j)^2 + \text{beta}(j,9)erate(j)\cdot covwf + \text{beta}(j,10)erate(j)\cdot\frac{efyld(j)}{yldREF(j)} \\ & + \text{beta}(j,11)erate(j)\cdot cvp(j) + \beta(j,12)covwf\cdot\frac{efyld(j)}{yldREF(j)} \\ & + \beta(j,13)covwf\cdot cvp(j) + \text{beta}(j,14)\cdot\frac{efyld(j)}{yldREF(j)}\cdot cvp(j), j \;=\; c,s,w,cn,sf,b. \end{aligned}$$

Now we need to take the weighted average of *epremrw* to determine if the maximum discount has been exceeded.

(20)
$$wfpremre \;=\; \frac{\sum_{j=c,s,w,cn,sf,b}\left(\sum_{i=i}^{N(i)}\text{round}(epremrw(j)\cdot acre(j,i)share(j,i),0)\right)}{\left(\sum_{j=c,s,w,cn,sf,b}\sum_{i=i}^{N(i)}acre(j,i)share(j,i)\right.}$$

Now set *wfpremr* equal to *minratefactor* times the weighted average of the enterprise unit premium rate if the maximum discount is exceeded, otherwise leave it alone. The product of *minratefactor* times the weighted average of the enterprise unit premium rate is rounded to four digits before the comparison is done.

(21) $$wfpremr \;=\; \max(wfpremr,\ \text{round}(minratefactor\cdot wfpremre),4)$$

Per-acre whole-farm unit premiums

The next step is to add the prevented planting load. The resulting per-acre premium is rounded to two digits. The prevented planting load is the share and acreage weighted average of the prevented planting load for all crops. An additional adjustment may be made when a short rate factor exists. Use the short rate factor *srfctr(j)* as defined by the Risk Management Agency. For whole-farm units, the short rate load is the share and acreage weighted average of the short rate load for all crops..

At 60% prevented planting

(22) $$LWFP(j,i) = wfpremr \cdot revwf \cdot srfctr(j,i)$$

At 65% prevented planting

(23)

$$LWFP(j,i) = wfpremr \cdot revwf \cdot srfctr(j,i) \cdot \frac{\sum_{j=c,s,w,cn,sf,b} PP65(j)\left(\sum_{i=i}^{N(i)} acre(j,i)share(j,i)\right)}{\sum_{j=c,s,w,cn,sf,b} \sum_{i=i}^{N(i)} acre(j,i)share(j,i)}$$

At 70% prevented planting

(24)

$$LWFP(j,i) = wfpremr \cdot revwf \cdot srfctr(j,i) \cdot \frac{\sum_{j=c,s,w,cn,sf,b} PP70(j)\left(\sum_{i=i}^{N(i)} acre(j,i)share(j,i)\right)}{\sum_{j=c,s,w,cn,sf,b} \sum_{i=i}^{N(i)} acre(j,i)share(j,i)}$$

Total whole-farm unit premiums

Total loaded premium is then found by multiplying *LWFP* by insured acres and share on each unit (or record) and then summing up over all records. This will be rounded on a record by record basis (M13, rectype 13, field 55, total premium, pos 346).

(25) $$TLWFP = \sum_{j=c,s,w,cn,sf,b} \sum_{i=i}^{N(i)} round(LWFP(j,i) \cdot acre(j,i) \cdot share(j,i),0)$$

7. Premium Subsidy

RA Premium subsidies are calculated by multiplying the total loaded premium by a premium subsidy factor. The premium subsidy factor depends on the coverage level percent. If the coverage level is equal to 0.65 or 0.70 then *subfact*, *subfacte*, and

subfactwf (whichever is appropriate) equals 0.59. If the coverage level is greater than or equal to 0.75 then *subfact*, *subfacte*, and *subfactwf* (whichever is appropriate) equals 0.55. If the coverage level is equal to 0.80 then *subfacte* and *subfactwf* (whichever is appropriate) equals 0.48. If the coverage level is equal to 0.85 then *subfacte* and *subfactwf* (whichever is appropriate) equals 0.38.

Optional and basic units

The RA premium subsidy equals the RA subsidy factor times the total loaded RA premium

$$psub(j,i) = \text{round}(subfact(j) \cdot TLP(j,i),0)), \quad i = 1,\ldots,N(j);\ j = c,s,w,cn,sf,b,ww \tag{26}$$

Enterprise unit

The premium subsidy for RA enterprise units is calculated on a record by record basis and then summed. Note that the record premium needs to be rounded before it is multiplied by the RA premium subsidy factor.

$$psube(j) = \sum_{i=1}^{N(j)} \text{round}(subfacte(j) \cdot \text{round}(LEP(j) \cdot acre(j,i) \cdot share(j,i)),0),0)), \quad j = c,s,w,cn,sf,b,ww \tag{27}$$

Whole-farm unit

The premium subsidy for RA whole-farm units is calculated on a record by record basis and then summed. Note that the record premium needs to be rounded before it is multiplied by the RA premium subsidy factor.

(28)

$$psubwf = \sum_{c,s,w,cn,sf,b} \sum_{i=1}^{N(j)} \text{round}(subfactwf \cdot round(LWFP \cdot acre(j,i) \cdot share(j,i)),0),0)),$$

8. Producer Paid Premiums

The following equations are used to calculate the subsidized producer-paid premiums for each unit structure. Because both the unsubsidized and subsidized premiums are rounded to whole-dollar amounts, there will be no need to round producer paid premium. The producer paid premiums are calculated on a record-by-record basis.

Optional and basic units

(29)

$$TLPsub(j,i) = TLP(j,i) - psub(j,i), i = 1,\ldots,N(j); j = c,s,w,cn,sf,b,ww.$$

Enterprise unit

(30)

$$TLEPsub(j,i) = \text{round}(LEP(j) \cdot acre(j,i) \cdot share(j,i)),0) - \text{round}(subfacte(j) \cdot \text{round}(LEP(j) \cdot acre(j,i) \cdot share(j,i)),0),0)$$
$$i = 1,...,N(j); j = c,s,w,cn,sf,b,ww.$$

Whole-farm unit

(31)

$$TLWFPsub(j,i) = \text{round}(LWFP(j) \cdot acre(j,i) \cdot share(j,i)),0) - \text{round}(subfactwf(j) \cdot \text{round}(LWFP(j) \cdot acre(j,i) \cdot share(j,i)),0),0)$$
$$i = 1,...,N(j); j = c,s,w,cn,,sf,b.$$

10. Appendix. Identification of Counties in Southern Minnesota and South Dakota

Table A1. Minnesota counties that make up Southern Minnesota and South Dakota counties that make up Eastern South Dakota.

FIPS	Southern Minn	FIPS	E. South Dakota
27011	Big Stone	46009	Bon Homme
27013	Blue Earth	46011	Brookings
27015	Brown	46027	Clay
27019	Carver	46029	Codington
27023	Chippewa	46035	Davison
27033	Cottonwood	46039	Deuel
27037	Dakota	46051	Grant
27039	Dodge	46057	Hamlin
27041	Douglas	46061	Hanson
27043	Faribault	46067	Hutchinson
27045	Fillmore	46077	Kingsbury
27047	Freeborn	46079	Lake
27049	Goodhue	46083	Lincoln
27051	Grant	46087	McCook
27053	Hennepin	46097	Miner
27055	Houston	46099	Minnehaha
27063	Jackson	46101	Moody
27067	Kandiyohi	46111	Sanborn
27073	Lac Qui Parle	46125	Turner
27079	Le Sueur	46127	Union
27081	Lincoln	46135	Yankton
27083	Lyon		
27085	McLeod		
27091	Martin		
27093	Meeker		
27099	Mower		
27101	Murray		
27103	Nicollet		
27105	Nobles		
27109	Olmsted		
27117	Pipestone		
27121	Pope		
27127	Redwood		
27129	Renville		

27131	**Rice**
27133	**Rock**
27139	**Scott**
27143	**Sibley**
27145	**Stearns**
27147	**Steele**
27149	**Stevens**
27151	**Swift**
27155	**Traverse**
27157	**Wabasha**
27161	**Waseca**
27165	**Watonwan**
27169	**Winona**
27171	**Wright**
27173	**Yellow Medicine**