THE USE OF LINEAR PROGRAMMING IN POWER GENERATION DECISIONS

RG HOEKSTRA

Associate Consultant, Bosch Projects (Pty) Ltd, P.O.Box 2009, Durban 4000, South Africa

Introduction

In a simple sugar mill, with only a limited number of similar boilers and similar turbo-alternators with no alternative uses of generated power and no problems of either a surplus or a shortage of bagasse, it is relatively easy to make the correct operating decisions. However, in bigger sugar mills with many years of evolution behind them, the problem of making the best decision is more complicated. There might be two boiler pressure ranges, a variety of boilers and of back pressure and condensing turbo-alternators, the possibility of selling export power, possibly at different prices to different clients, and of using or selling bagasse as a by-product instead of burning it as a boiler fuel. In addition, many constraints will have to be observed, such as capacities of the various boilers and turbo-alternators, the size of the markets for power and bagasse export and the amount of exhaust steam which the process section can absorb without blow-off. Calculation of an optimum policy within all those constraints can become a daunting task.

The answer is to perform such calculations by linear programming. This facility is now available in the better spreadsheet packages, thus making it more accessible, user-friendly and able to slot in with spreadsheet simulations of other parts of the factory.

What is linear programming?

A typical linear equation is of the form:

$3*x_1 + 4.1*x_2 - 2*x_3 = 9$

The symbols x_1, x_2, …. are the variables or unknown quantities in the equation. Note that they must appear to the power of 1. Such an equation will, in two dimensions, give a straight line, and in three dimensions a flat surface, hence the name linear.

More generally, a set of m linear equations in n unknowns will be in the form of:

$a_{11}*x_1 + a_{12}*x_2 + \ldots\ldots\ldots + a_{1n}*x_n = c_1$

$a_{21}*x_1 + a_{22}*x_2 + \ldots\ldots\ldots + a_{2n}*x_n = c_2$

....

$a_{m1}*x_1 + a_{m2}*x_2 + \ldots\ldots\ldots + a_{mn}*x_n = c_m$

The fixed values a_{11}, a_{12}, ... , a_{mn} are the coefficients and the c_1, c_2, ..., c_m the constants.

The object of setting up such equations is to solve all the unknown x-values, which are the answer to the particular problem.

With such sets of linear equations, there are three possibilities:

1. There is the same number of equations as unknowns, i.e. m = n.

This is the usual case, with which we are all familiar from high school algebra. There will be one and only one solution to the set of equations.

2. There are more equations than unknown variables, i.e. m > n.

The set is over-specified, and usually there will not be a proper solution, because the excessive equations will contradict each other.

3. There are fewer equations than unknown variables, i.e. m < n.

There will be an infinite number of solutions which can satisfy the equations. This is an interesting case, because within that infinity of possible solutions there should be a most advantageous or optimum set of values for the unknowns. This is where linear programming comes in.

In short, linear programming problems have the following characteristics:

- There will be a number of underlying linear relationships or equations, which represent the mathematical or physical logic of the system under discussion. For example: $x_2 + x_3 = 7$
- Besides the above-mentioned equalities, there can also be inequalities or constraints, in that one or a combination of variables may not exceed a given fixed value. For example: $x_1 + 3*x_4 <= 10.2$
- The unknowns or variables are all non-negative, meaning they must each be greater than or equal to zero, i.e. $x_i >= 0$ for all i within [1, ..., n]. This will rule out solutions that may be mathematically correct but infeasible in practice, such as a negative mass.
- There must be a combination of some of the x-unknowns with appropriate cost or value coefficients, to represent the value to the user. This is called the objective function, and the purpose of the linear program is to choose that combination of unknown x-values, subject to the constraints of equations, inequalities and non-negative values, which will yield the maximum or optimum value. For example: Maximise ($5.2*x_3 + 10.3*x_4$)

There are many textbooks on linear programming, of which Gass (1969) is an example.

Discussion of example

Power-house flow diagram:

The basic flow diagram of the power house of a sugar factory chosen as example is shown in Figure 1.

The factory has two boiler pressure ranges, for convenience called high and medium pressure (HP and MP) respectively. Within each pressure range, there are boilers, direct-drive turbines, back-pressure turbo-alternators and a condensing turbo-alternator. For simplicity, the boilers within each pressure range are considered to be one large boiler, and likewise the direct drive turbines and the turbo-alternators. There are two markets of different sizes and prices for selling export power. The bagasse can be sold for by-product use, but this might require the purchase of coal to provide sufficient steam.

Given fixed operating values:

Note that all given values which follow are shown in bold type in Figure 1.

A listing of all the given values appears in Appendix 1.

On the assumption that the factory crushes at a given rate, various operating values which are relevant to the overall energy optimisation problem will be considered as given and fixed. These are: The steam sundries and losses and the steam for the direct drive turbines (t/h) for each of the pressure ranges; the LP steam demand; the bagasse supply and the power demands (MW) for the factory and the village.

Given equipment characteristics:

These are: Steam/MW ratio for back-pressure and for condensing TA's and the steam/bagasse ratio for each of the boiler pressure ranges; and the coal/bagasse heating value ratio. Also required are the enthalpies of the HP, MP and LP steam at the given respective pressures and temperatures and enthalpies of the direct drive turbine exhausts.

Given constraints:

These constraints can be upper and/or lower limits with respect to their associated variables. Among upper limits are the maximum boiler steam production (t/h) and the maximum power generating capacity (MW) of the back-pressure and the condensing TA's, for each of the pressure ranges. On the sales side are the maximum bagasse sales (t/h) and the maximum power (MW) sales to customers A and B. Lower limits apply to the minimum let-down rates (t/h) from HP to MP and from MP to LP steam.

Given cost and pricing data:

The objective, as expected, is to maximise profit, and the objective function will therefore have to contain all the variables that affect profit, together with their relevant coefficients.

These include the cost of coal and the incomes from sales of bagasse (R/t) as by-product and from export of power (R/MWh) to customers A and B. Note that the coal cost must have a negative coefficient because it is a cost and works against profit maximisation.

Equations and inequalities:

A listing of all the inequalities and equations is given in Appendix 2.

The logical relationships of the unknown variables must be expressed in equations and, where constraints are applicable, as inequalities.

The inequalities consist of expressing each of the aforementioned constraints in relation to its associated variable.

The equations consist of:

The steam/MW relationships for the back-pressure and the condensing TA's for each of the pressure ranges;

The Supply = Demand relationships for HP, MP and LP steam (t/h) and for power (MW);

The Input = Output mass (t/h) and heat (MJ/h) balances across each of the desuperheaters and across the combined TA's.

The bagasse used for fuel is what is left over from sales and, if not sufficient, the balance is made up by coal, with adjustment for differing heating values.

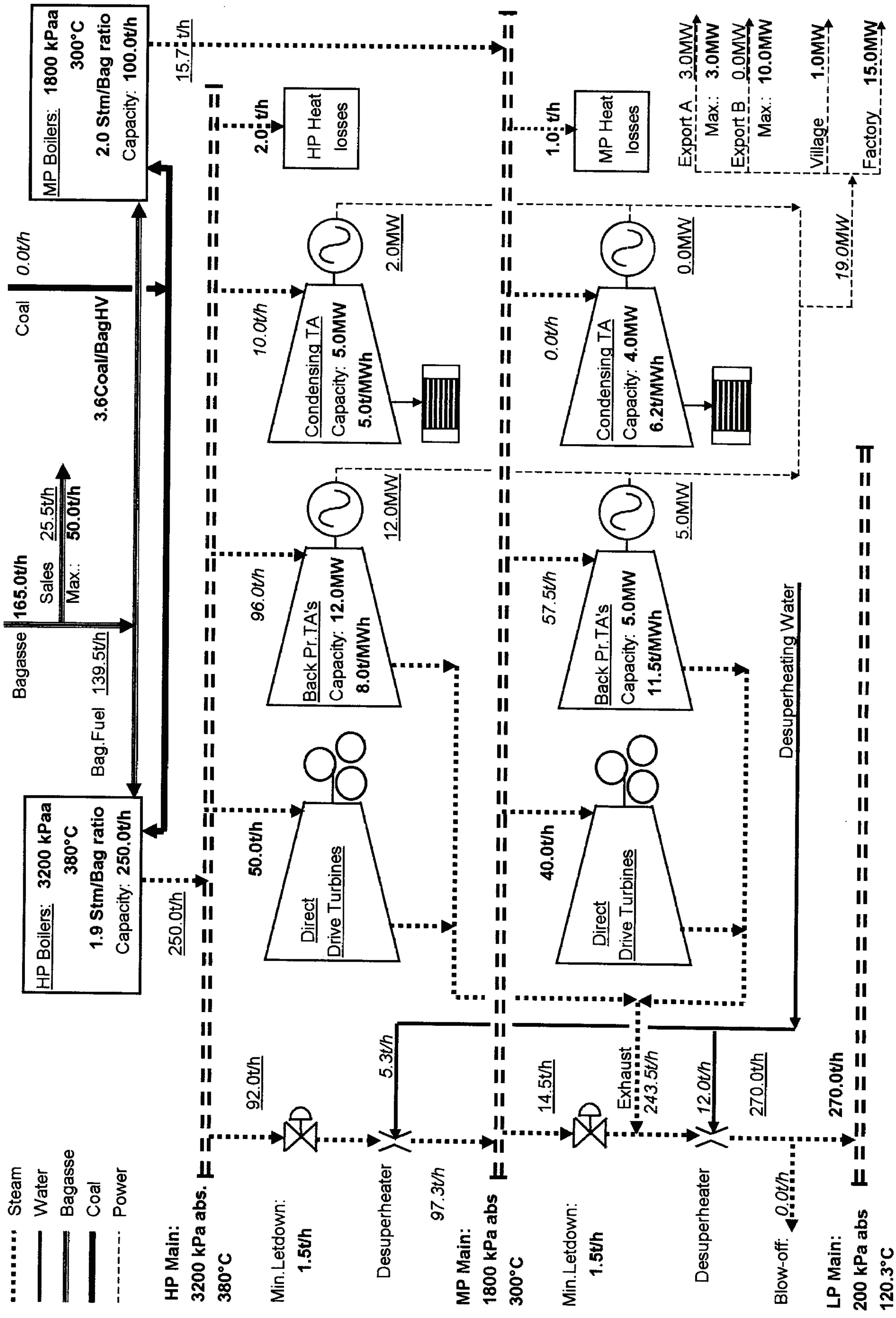
The resulting solution to the problem is shown in Appendix 2.

To make the results easier to follow, the flow diagram in Figure 1 shows the solution values underlined, compared to the bold type used for the given values.

Discussion:

- Some of the constraints turn out to be slack, others are tight, i.e. the variables are operating at their limits.
- The computing time is rapid. In a problem of this size, with a 400 MHz processor and an Excel 95 spreadsheet, the solution is reached within a second.
- The model illustrated is fairly simple, in that it considers all operating units within a given category as one large unit. For greater accuracy, it is possible to consider individual turbo-alternators, boilers, etc., with their own respective capacities and efficiencies, albeit in a more complicated program.
- Other equipment and features can also be added, such as a topping turbo-alternator, steam export to a by-product plant or import of power.
- For simplicity, each turbo-alternator was given a fixed steam/MW ratio, regardless of its actual MW output. A more accurate approach is the application of Willans lines, whereby the t/h steam required is a linear function of the MW output of the TA, with an intercept on the t/h axis. The slope of the line has dimensions of t/MWh, and the intercept represents the "overhead" t/h of steam to represent fixed energy losses.
- With pieces of equipment, such as the above-mentioned TA's, the overhead steam supply is required only when an output (MW) is produced. If optimum operation indicates that such a unit should not be used, the overhead t/h of steam does not apply. With pure linear programming, this cannot be simulated, but fortunately some versions have a facility to constrain chosen variables to integers. By letting a (0, 1) integer variable, say ta_use, represent whether or not the equipment is in use, an equation of the following form can be used to give TA steam consumption in terms of its MW output and its usage status:

  ta_steam = ta_use*Intercept + mw_gen*Slope



Figures in bold are the known, given data. Underlined figures were directly calculated by the Linear Program. Figures in italics were derived from these.

Figure 1. Flow Diagram of Power House of Example Sugar Factory.

Illustration of response of linear program to varying of parameters.

As an example of how the linear programming system can cope with different values of parameters, Appendix 3 shows the effect of changes in the bagasse sales price on the calculated optimum outcomes of fuel tonnages, MW power exports and objective function. In so doing, various constraints change from slack to tight or vice versa.

Integration into overall factory energy and mass balance.

Such a linear programming section can be integrated into an overall mass and energy balance for the complete factory. One of the most important inputs around which the power generation decision logic revolves is the LP steam demand from the process. With the ability to simulate the process section, the LP steam demand need no longer be regarded as absolutely fixed, but can be varied by changing the amount of imbibition water, the choice of juice heaters, no. of evaporator effects, pan boiling scheme, etc. This will greatly extend the range of problems which can be optimised. For example, if bagasse surplus is a problem, it could be advantageous to increase the LP steam demand by making changes to the process section.

Ideally, the amount of imbibition water should be included as one of the linear program variables to optimise, but calculation of its effect on LP steam demand would mean bringing much of the process section into the linear program, thus disproportionately complicating the system. It would therefore be better to change the process operating conditions through the appropriate input values to the factory mass and energy balance model, re-run the linear program, and compare results.

It follows that the process section should preferably be based on the same spreadsheet package as the linear program. Proprietary process software packages of the "black box" type are likely to be awkward to integrate.

Applications of linear programming to power generation decisions.

- Planning of new or expanded factories.
- De-bottlenecking of existing factories.
- Feasibility investigation of sales or by-product projects.
- Planning of optimum routine operating procedures.
- Decision-making for unexpected events, such as equipment failure or a change in the marketing parameters.

REFERENCES

Gass, S.I. (1969). *Linear Programming*. 3rd Edition McGraw-Hill

Appendix 1.
Given and known data for input to linear program.

Notation.

The names of the unknowns to be solved are entirely in lower case letters.
The names of the given fixed values and constraints contain capitals.
Suffixes after the Names :
_h : Refers to High Pressure (HP) range: 3200 kPaa, 380°C
_m : Refers to Medium Pressure (MP) range: 1800 kPaa, 300°C

Given Fixed Operating Values:

Description	Units	Name	Value:
Steam Sundries & Losses	t/h	Sundries_h	2.0
		Sundries_m	1.0
Steam to Direct Drive Turbines	t/h	Tu_h	50.0
		Tu_m	40.0
LP Steam Demand	t/h	LPDemand	270.0
Bagasse Supply	t/h	BagSupply	165.0
Factory Power Demand	MW	FactoryMW	15.00
Village Power Demand		VillageMW	1.00

Given Equipment Characteristics:

Description	Units	Name	Value:
Steam/MW Ratio for BackPr.TA's	t/MWh	StmMWBpta_h	8.0
		StmMWBpta_m	11.5
Steam/MW Ratio for Cond.TA's	t/MWh	StmMWCta_h	5.0
		StmMWCta_m	6.2
Steam/Bagasse ratio	t/t	SteamBag_h	1.9
		SteamBag_m	2.0
Coal/Bagasse Heating Value Ratio	MJ/MJ	Coal_BagHV	3.6

Thermal Properties:

Stream	Pressure kPa abs.	Temp. °C	Enthalpy of Stream Name	kJ/kg	Condition
HP Steam	3200	380	StmEnth_h	3180	Superh.
MP Steam	1800	300	StmEnth_m	3031	Superh.
LP Steam	200	120.3	StmEnth_lp	2707	Saturated
Exh.from Direct	200	259.0	TuEnth_h	2988	Superh.
Drive Turbines	200	187.7	TuEnth_m	2845	Superh.
Desuperh.Water for HP Steam		105	DesupWEnth	440	

Given Constraints:

Description	Units	Name	Value:	Upper/Lower:
Steam from Boilers	t/h	BoilerCap_h	250.0	Upper
		BoilerCap_m	100.0	Upper
Steam to Back Pressure TA's	MW	BptaMWCap_h	12.00	Upper
		BptaMWCap_m	5.00	Upper
Steam to Condensing TA's	MW	CtaMWCap_h	5.00	Upper
		CtaMWCap_m	4.00	Upper
Steam Let down from HP to MP	t/h	LetDnMin_h	1.5	Lower
Steam Let down from MP to LP		LetDnMin_m	1.5	Lower
Bagasse Sales	t/h	BagSalesMax	50.0	Upper
Power Export: Customer a	MW	ExpMWMax_A	3.00	Upper
Power Export: Customer b		ExpMWMax_B	10.00	Upper

GivenCost and Pricing Data:

Coal Cost	R/t	CoalCost	R 300.00
Income for Bagasse Sales	R/t	BagPrice	R 50.00
Income for Power Export to A	R/MWh	MWPrice_A	R 200.00
Income for Power Export to B		MWPrice_B	R 120.00

Appendix 2.

Unknown variables, inequalities, equations and objective function.

Notation.

The names of the unknowns to be solved are entirely in lower case letters.

The names of the given fixed values and constraints are contain capitals.

Suffixes :	_h : Refers to High Pressure (HP) range: 3200 kPaa, 380°C
	_m : Refers to Medium Pressure (MP) range: 1800 kPaa, 300°C

Unknowns to be solved by the Linear Program:

Description	Units	Name:	Value on Solution:
Steam from Boilers	t/h	boiler_h	250.0
		boiler_m	15.7
Power from BackPressure TA's	MW	bptamw_h	12.00
		bptamw_m	5.00
Power from Condensing TA's	MW	ctamw_h	2.00
		ctamw_m	0.00
Total MW Produced	MW	totmw	19.00
Steam Let down from HP to MP	t/h	letdn_h	92.0
Steam Let down from MP to LP		letdn_m	14.5
LP Steam production	t/h	lp_prod	270.0
Bagasse Sales	t/h	bagsales	25.5
Bagasse fuel	t/h	bagfuel	139.5
Power Export: Customer A	MW	expmw_a	3.00
Power Export: Customer B	MW	expmw_b	0.00

Other Unknowns, following from the Linear Program Solution:

Description	Units	Name:	Value on Solution:
Desuperheating Water	t/h	desupw_h	5.3
		desupw_m	12.0
Desuperheated Letdown Steam	t/h	dsltdn_h	97.3
LP Steam Blow-off	t/h	lp_blowoff	0.0
Coal burnt as fuel	t/h	coalfuel	0.0
Steam to Back Pressure TA's	t/h	bpta_h	96.0
		bpta_m	57.5
Steam to Condensing TA's	t/h	cta_h	10.0
		cta_m	0.0
Combined Exhaust Flow	t/h	exhaust	243.5

Inequalities (Constraints):

Boilers:			
t/h	boiler_h	<=	BoilerCap_h
	boiler_m	<=	BoilerCap_m
BackPressureTurbo-Alternators:			
MW	bptamw_h	<=	BptaMWCap_h
	bptamw_m	<=	BptaMWCap_m
Condensing Turbo-Alternators:			
MW	ctamw_h	<=	CtaMWCap_h
	ctamw_m	<=	CtaMWCap_m
Steam let-downs:			
t/h	letdn_h	>=	LetDnMin_h
	letdn_m	>=	LetDnMin_m
	lp_prod	>=	LPDemand
Sales and Exports:			
t/h	bagsales	<=	BagasseMax
t/h	bagsales	<=	BagSalesMax
MW	expmw_a	<=	ExpMWMax_A
MW	expmw_b	<=	ExpMWMax_B

Equalities (Equations):

Steam/MW Relationships for BackPressure TA's:			
t/h	bpta_h	=	bptamw_h*StmMWBpta_h
	bpta_m	=	bptamw_m*StmMWBpta_m
Steam/MW Relationships for Condensing TA's:			
t/h	cta_h	=	ctamw_h*StmMWCta_h
	cta_m	=	ctamw_m*StmMWCta_m
HP Steam Supply and Demand:			
t/h	boiler_h	=	bpta_h + cta_h + letdn_h + Tu_h + Sundries_h
Letdown from HP to MP:			
t/h	dsletdn_h	=	letdn_h + desupw_h
MJ/h	dsltdn_h*StmEnth_m	=	letdn_h*StmEnth_h + desupw_h*DesupWEnth
MP Steam Supply and Demand:			
t/h	dsletdn_h + boiler_m	=	bpta_m + cta_m + letdn_m + Tu_m + Sundries_
Total Exhaust Steam:			
t/h	exhaust	=	bpta_h + bpta_m + Tu_h + Tu_m
MJ/h	exhhtrate	=	(bpta_h*StmEnth_h - 3600*bptamw_h)
			+ (bpta_m*StmEnth_m - 3600*bptamw_m)
			+ Tu_h*TuEnth_h + Tu_m*TuEnth_m
Letdown from MP to LP:			
t/h	lp_prod	=	letdn_m + desupw_m + exhaust
MJ/h	lp_prod*StmEnth_lp	=	letdn_m*StmEnth_m + desupw_m*DesupWEnth
			+ exhhtrate
LP Steam Blow-off:			
t/h	lp_blowoff	=	lp_prod - LPDemand
Power Supply and Demand:			
MW	bptamw_h + ctamw_h + bptamw_m + ctamw_m =		
	FactoryMW + VillageMW + expmw_a + expmw_b		
Fuel Supply to Boilers:			
t/h	selfsuftbag	=	(boiler_h/SteamBag_h + boiler_m/SteamBag_m)
t/h	bagavail	=	BagSupply - bagsales
t/h	bagfuel	<=	selfsuftbag
t/h	bagfuel	<=	bagavail
t/h	coalfuel	=	(selfsuftbag - bagfuel)/Coal_BagHV

Objective Function:

Maximise(expmw_a*MWPrice_A + expmw_b*MWPrice_B
+ bagsales*BagPrice - coalfuel*CoalCost)
= R 1,877.31 /h on Solution

Appendix 3

Effect of varying the price of bagasse on the linear programming solution:

As an illustration of how the linear program can be used, one of the parameters, namely the price of bagasse (Rand/ton), will be varied from either side of the base case, which is at R50.00/ton bagasse.

The main changes in the output are summarised in Table 1 below:

Discussion:

Case A: R90.00/ton:

At this price the bagasse is valuable enough to sell up to the limit of its market size of 50 tons/h. The price for Power Export A is sufficiently high to justify using coal fuel in the place of the resulting bagasse shortfall, but only for generation by back-pressure TA's, with their more favourable (lower) steam/MW ratios.

Case B: R80.00/ton:

The bagasse price no longer justifies substitution by coal for boiler fuel, and consequently the bagasse sales drop and bagasse usage for fuel rises. The amounts of power generated remain as for Case A.

Case C: R50.00/ton:

This is the base or example case, per Figure 1 and Appendices 1 and 2.

At this price it becomes economical to sacrifice further bagasse sales to generate on the HP condensing TA to the limit of the market for Power Export A.

Case D: R40.00/ton:

Here the full capacity of the HP condensing TA is used to generate the full Power Export A and part of the Export B allocation. The MP condensing TA is not used, as its Steam/MW ratio is not sufficiently favourable (low) to justify diverting bagasse sales to provide the necessary steam.

Case E: R30.00/ton:

The bagasse price is now sufficiently low to use bagasse for steam generation, to the limit of the total TA capacity. Only such bagasse surplus as remains gets sold.

Table 1. Effect of varying the bagasse price on the linear programming solution.

Case No.:		**A**	**B**	**C****	**D**	**E**	
Bagasse Price	R/ton	R 90.00	R 80.00	R 50.00	R 40.00	R 30.00	
Selected Solution Variables:							**Upper Limit**
Bagasse Sales	t/h	**50.0**	30.8	25.5	17.6	5.2	50.0
Bagasse Fuel	t/h	115.0	134.0	139.5	147.4	159.8	165.0
Coal Fuel	t/h	5.3	0.0	0.0	0.0	0.0	None
MP Boiler Steam	t/h	5.2	5.2	15.7	31.7	56.4	100.0
HP Cond.TA	t/h	0.0	0.0	10.0	25.0	25.0	
	MW	0.0	0.0	2.0	**5.0**	**5.0**	5.0
MP Cond.TA	t/h	0.0	0.0	0.0	0.0	24.8	
	MW	0.0	0.0	0.0	0.0	**4.0**	4.0
Power Export A	MW	1.0	1.0	**3.0**	**3.0**	**3.0**	3.0
Power Export B	MW	0.0	0.0	0.0	3.0	7.0	10.0
Objective Function	R/h	3103	2667	1877	1665	1596	

** BaseCase

Bold type indicates tight constraint on the variable.