Zeitschrift für Operations Research, Band 23, 1979, Seite 197–206. ©Physica-Verlag, Würzburg.

On *R. W. Llewellyn*'s Rules to Identify Redundant Constraints: A Detailed Critique and Some Generalizations

By *J. Telgen*, Rotterdam[1])

Eingegangen am 14. April 1978
Revidierte Fassung eingegangen am 29. Januar 1979

Abstract: In his book "Linear Programming" [1964] *Llewellyn* devoted a chapter to simplifications and reductions of a linear programming problem by means of algebraic rules. These rules are claimed to be rather general. Here we give some counterexamples, where the rules of *Llewellyn* do not hold. Furthermore we give some general rules to identify redundant constraints in the case *Llewellyn* considers and show that the original rules of *Llewellyn* together with an extra condition are a variant of these general rules. Finally we consider the question whether or not the rules of *Llewellyn* should be used to identify redundant constraints.

Zusammenfassung: Bereits 1964 hat sich *Llewellyn* in seinem Buch „Linear Programming" mit der Vereinfachung von linearen Programmen durch Ermittlung redundanter Nebenbedingungen beschäftigt. Der von ihm für seine Regeln erhobene Anspruch der Allgemeingültigkeit wird in diesem Beitrag durch Gegenbeispiele widerlegt. Ferner werden allgemeine Regeln zur Identifikation redundanter Nebenbedingungen hergeleitet und gezeigt, daß diese die Regeln von *Llewellyn*, sofern man sie um eine zusätzliche Bedingung erweitert, umfassen.

1. Introduction

In his book "Linear Programming" published in 1964 *Llewellyn* devoted a chapter (Chapter 6; esp. 132–138) to simplifications and reductions in linear programming problems by means of algebraic rules. This publication has often been referenced as one of the first attempts to identify redundant constraints in LP problems, or more generally in systems of linear inequalities [see e.g. *Thompson/Tonge/Zionts; Gal*].

Here we are concerned with the rules *Llewellyn* gives to identify some special redundant constraints. The rules are intended to identify constraints that are redundant by the presence of one other constraint and the non-negativity constraints on all variables. Basically the same rules are reintroduced in *Zeleny* [1974].

We briefly review these rules in section 2. Unfortunately, the rules are not as general as is claimed by *Llewellyn*. Some counterexamples are given in section 3. In section

[1]) *J. Telgen*, Econometric Institute, Erasmus University Rotterdam, POB 1738, NL–Rotterdam.

4 a general criterion to identify redundant constraints [*Gal; Telgen*, 1977a] is used to obtain necessary and sufficient conditions for the case *Llewellyn* considers.

In section 5 it is shown that *Llewellyn*'s rules together with an extra condition on the signs of the coefficients, are a variant of the general criterion. Finally the question, whether or not the (extended) rules of *Llewellyn* should be used to identify redundant constraints, is considered both from a practical point of view and in relation with the theory of computational complexity.

2. *Llewellyn*'s Rules

We consider the system of linear inequalities

$$A\underline{x} \leqslant \underline{b} \tag{2.1}$$

$$\underline{x} \geqslant \underline{0} \tag{2.2}$$

where A is an $m \times n$-matrix; $\underline{x}$ and $\underline{0}$ are n-vectors and $\underline{b}$ is an m-vector. We denote by $\underline{\alpha}_i$ $(i = 1, \ldots, m)$ the i-th row of A.

Since redundancy is not defined for inconsistent systems, we will assume feasibility of the system (2.1) – (2.2) throughout.

Llewellyn states two rules to identify some redundant constraints from (2.1). A redundant constraint is not defined in his book[2]) but we will suppose that *Llewellyn* implicitely used a widespread definition [see *Telgen*, 1977b] like:

The k-th constraint

$$\underline{\alpha}_k \underline{x} \leqslant b_k \tag{2.3}$$

is redundant in the system (2.1) – (2.2) if and only if

$$\underline{\alpha}_k \underline{x} \leqslant b_k \qquad \forall\, x \in \mathbf{R}^n \;\left|\; \begin{array}{ll} \underline{\alpha}_i \underline{x} \leqslant b_i & i = 1, \ldots, m \quad i \neq k \\ \underline{x} \geqslant \underline{0}. & \end{array}\right. \tag{2.4}$$

The rules, *Llewellyn* gives, are concerned with the situation in which a constraint in (2.1) is redundant by virtue of one other constraint from (2.1) and all constraints (2.2). In terms of the definition given above, it is the case in which the k-th constraint is redundant because for some $s \in (1, \ldots, m)$ $s \neq k$:

$$\underline{\alpha}_k \underline{x} \leqslant b_k \qquad \forall\, x \in \mathbf{R}^n \;\left|\; \begin{array}{l} \underline{\alpha}_s \underline{x} \leqslant b_s \\ \underline{x} \geqslant \underline{0}. \end{array}\right. \tag{2.5}$$

---

[2]) *Llewellyn* [1964, p. 135]: "... the constraint is redundant and can be dropped from the problem without affecting the optimum solution"; this is hardly a definition.

For this case *Llewellyn* states two rules:
rule 1: Given two inequalities with b_k and $b_s \geqslant 0$ where

$$\frac{b_k}{a_{kj}} \geqslant \frac{b_s}{a_{sj}} \qquad \forall j \in (1, \ldots, n) \tag{2.6}$$

then the k-th inequality is redundant.
rule 2: Given two inequalities with b_k and $b_s < 0$ where

$$\frac{b_k}{a_{kj}} \leqslant \frac{b_s}{a_{sj}} \qquad \forall j \in (1, \ldots, n) \tag{2.7}$$

then the k-th inequality is redundant.

Some examples [modified from *Llewellyn*, 135–137] are given in the following figures:

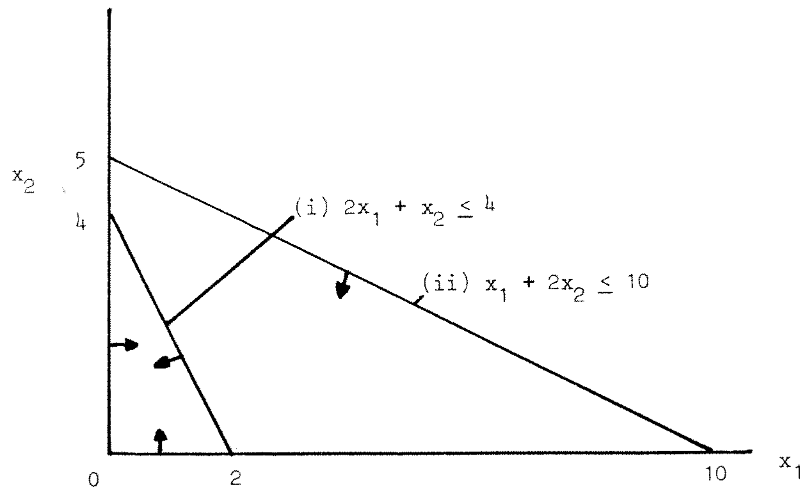

Fig. 1: Inequality (ii) is redundant according to rule 1

Then *Llewellyn* [1964, p. 136] states: "While it is easy to see the justification for the two rules just given after studying the figures, the reader should be sure he appreciates the fact that the rules are algebraic. It can be shown that these rules hold for problems of any dimension and can be applied to pairs of inequality constraints regardless of the signs of the coefficients of the variables in the constraints and even hold if some of the coefficients equal zero."

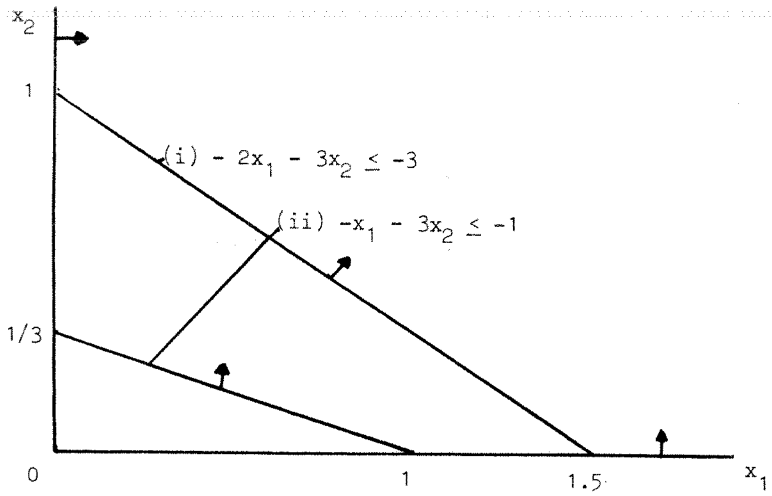

Fig. 2: Inequality (ii) is redundant according to rule 2

3. Counterexamples

To see that *Llewellyn* is not correct in the quoted sentences it is enough to give a counterexample for each of the two rules. For rule 1 consider the system:

$$\left.\begin{array}{ll} \text{(i)} & x_1 + x_2 \leqslant 2 \\ \text{(ii)} & x_1 - x_2 \leqslant 1 \\ & x_1,\ x_2 \geqslant 0 \end{array}\right\} \tag{3.1}$$

which is graphically shown in figure 3.

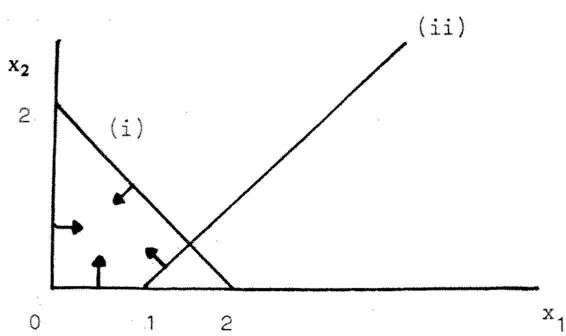

Fig. 3: The system (3.1)

Clearly there is no redundant constraint in system (3.1). But if we use *Llewellyn*'s rule 1, then $2/1 \geqslant 1/1$ and $2/1 \geqslant 1/-1$ and so constraint (i) is said to be redundant!

For rule 2 consider the system:

$$\left.\begin{array}{ll}\text{(i)} & -x_1 - x_2 \leqslant -3 \\ \text{(ii)} & -x_1 + x_2 \leqslant -2 \\ & x_1,\ x_2 \geqslant 0\end{array}\right\} \qquad (3.2)$$

which is graphically shown in figure 4.

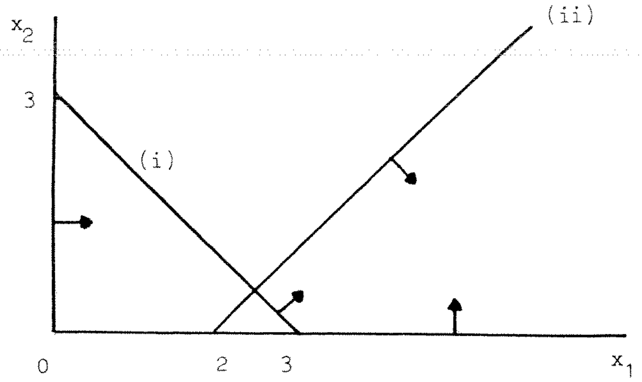

Fig. 4: The system (3.2)

Clearly constraint (ii) is not redundant, but using rule 2 we find that $-2/-1 \leqslant -3/-1$ and $-2/1 \leqslant -3/-1$ and so constraint (ii) is redundant according to *Llewellyn*'s rules!

Moreover, if $b_k = b_s = 0$ both the k-th and the s-th constraint may be declared redundant by application of *Llewellyn*'s rule 1! From simple geometrical considerations it will be clear that any one of two constraints can be redundant, only if they are coincident or if the feasible region is of lower dimension than the solution space. This last condition is fulfilled only if (for $b_k = b_s = 0$)

$$\left.\begin{array}{l} a_{kj} \geqslant 0 \\ a_{sj} \geqslant 0 \end{array}\right\} \qquad \forall j \in (1, \ldots, n). \qquad (3.3)$$

4. Generalizing the Method

In *Gal* [1975] and *Telgen* [1977a] it is proved that there is a direct correspondence for $k = 1, \ldots, m$, k fixed, between redundancy of the k-th constraint and the prob-

lem:

$$\hat{u}_k = \min (b_k - \underline{\alpha}_k \underline{x})$$
$$\text{s.t. } \underline{\alpha}_i \underline{x} \leqslant b_i \qquad i = 1, \ldots, m \qquad i \neq k. \tag{4.1}$$
$$\underline{x} \geqslant \underline{0}$$

If $\hat{u}_k \geqslant 0$ the k-th constraint is redundant, otherwise it is not. In the case *Llewellyn* [1964] considers, the k-th constraint should be redundant subject to the s-th constraint and all nonnegativity constraints i.e.

$$\tilde{u}_k = \min (b_k - \underline{\alpha}_k \underline{x}) \tag{4.2}$$
$$\text{s.t. } \underline{\alpha}_s \underline{x} \leqslant b_s$$
$$\underline{x} \geqslant \underline{0}.$$

Since it is trivial that $\tilde{u}_k \leqslant \hat{u}_k$, it will be clear that, if $\tilde{u}_k \geqslant 0$ then also $\hat{u}_k \geqslant 0$ and the k-th constraint is redundant.

The linear programming problem (4.2) is quite easy to solve.

First note that, since we assumed that the original system is feasible, (4.2) has a feasible solution. Because (4.2) has only one constraint, an optimal solution is at hand or can be obtained in only one pivot step.
An optimal solution is at hand if the conditions for one of the following two theorems hold.

Theorem 1. The problem (4.2) has a finite optimal solution with value $\tilde{u}_k = b_k$ if

$$b_s \geqslant 0$$
$$a_{kj} \leqslant 0 \qquad \forall j \in (1, \ldots, n). \tag{4.3}$$

Theorem 2. The problem (4.2) has an unbounded optimal solution if

$$\exists j \in (1, \ldots, n) \text{ such that } \begin{cases} a_{kj} > 0 \\ a_{sj} \leqslant 0. \end{cases} \tag{4.4}$$

The proof of these theorems follows directly from the correctness of the simplex method and is omitted here.

Excluding these cases, the optimal solution can be reached in one simplex iteration, by pivoting on element a_{sp}. We know that a finite optimal solution exists if the coefficients after pivoting (indicated by a prime) satisfy the conditions (4.3):

$$a'_{kj} = a_{kj} - \frac{a_{kp}}{a_{sp}} \cdot a_{sj} \leqslant 0 \qquad \forall j \in (1, \ldots, n) \qquad j \neq p \tag{4.5}$$

$$a'_{kp} = -\frac{a_{kp}}{a_{sp}} \leqslant 0 \tag{4.6}$$

$$b'_s = \frac{b_s}{a_{sp}} \geqslant 0. \tag{4.7}$$

If $b_s \geqslant 0$ we choose a_{sp} such that

$$\frac{a_{kp}}{a_{sp}} = \max_j \left\{ \frac{a_{kj}}{a_{sj}} \,\middle|\, a_{sj} > 0 \right\} \tag{4.8}$$

and check for

$$\begin{aligned} \frac{a_{kp}}{a_{sp}} &\leqslant \frac{a_{kj}}{a_{sj}} & \forall j \text{ with } a_{sj} < 0. \\ a_{kj} &\leqslant 0 & \forall j \text{ with } a_{sj} = 0 \end{aligned} \tag{4.9}$$

If (4.9) does not hold then some $a'_{kj} > 0$ and $a'_{sj} = \frac{a_{sj}}{a_{sp}} < 0$, so theorem 2 applies. If (4.9) holds, then theorem 1 applies and

$$\tilde{u}_k = b'_k = b_k - \frac{a_{kp}}{a_{sp}} b_s. \tag{4.10}$$

If $b_s < 0$ we choose a_{sp} such that

$$\frac{a_{kp}}{a_{sp}} = \min_j \left\{ \frac{a_{kj}}{a_{sj}} \,\middle|\, a_{sj} < 0 \right\} \tag{4.11}$$

and check for

$$\begin{aligned} \frac{a_{kp}}{a_{sp}} &\geqslant \frac{a_{kj}}{a_{sj}} & \forall j \text{ with } a_{sj} > 0. \\ a_{kj} &\leqslant 0 & \forall j \text{ with } a_{sj} = 0 \end{aligned} \tag{4.12}$$

Analogeously, if (4.12) does not hold, the problem (4.2) has an unbounded solution and if (4.12) holds the optimal solution is reached with objective function value as determined by (4.10).

5. *Llewellyn*'s Rules as a Variant of the General Method

It is easily seen that the rules *Llewellyn* gives do not ensure that the conditions given in the preceding section hold. However, we will show that together with an extra

condition *Llewellyn*'s rules are a sufficient condition to identify redundant constraints.

For *Llewellyn*'s rule 1 we require all coefficients to be non-negative: choose p according to (4.8), then (4.9) certainly holds, and the k-th constraint is redundant according to (4.10), if

$$b_k - \frac{a_{kp}}{a_{sp}} \cdot b_s \geqslant 0 \tag{5.1}$$

which yields

$$\frac{b_k}{b_s} \geqslant \frac{a_{kp}}{a_{sp}} \; . \tag{5.2}$$

Combining (5.2) and (4.8) yields:

$$\frac{b_k}{b_s} \geqslant \frac{a_{kj}}{a_{sj}} \qquad \forall j \in (1, \ldots, n) \tag{5.3}$$

which is equivalent to *Llewellyn*'s rule 1.

In a similar way we may derive *Llewellyn*'s rule 2, by requiring all coefficients to be nonpositive: choose p according to (4.11) then (4.12) is satisfied, and from (4.10) we see that the k-th constraint is redundant if

$$\frac{b_k}{b_s} \leqslant \frac{a_{kp}}{a_{sp}} \; . \tag{5.4}$$

Combining (5.4) and (4.11) yields:

$$\frac{b_k}{b_s} \leqslant \frac{a_{kj}}{a_{sj}} \qquad \forall j \in (1, \ldots, n) \tag{5.5}$$

which is equivalent to *Llewellyn*'s rule 2.
Another generalization of *Llewellyn*'s rules is given in *Eckhardt* [1971]. There it is proved that the k-th constraint is redundant if and only if there is some basic feasible solution to the system (2.1) – (2.2), an index s and some $\mu \geqslant 0$, such that

$$\begin{aligned} a_{kj} &\leqslant \mu \cdot a_{sj} \qquad \forall j \in (1, \ldots, n). \\ b_k &\geqslant \mu \cdot b_s \end{aligned} \tag{5.6}$$

This implies that, without changing the basis, the k-th constraint may be identified as being redundant if both b_k and $b_s \geqslant 0$ and there exists some $\mu \geqslant 0$ such that (5.6) holds. *Llewellyn*'s rules are a special case of (5.6) since $\mu = a_{kp}/a_{sp}$ yields (4.5) and (4.10), from which *Llewellyn*'s rules arise if all coefficients are either nonnegative or nonpositive, as shown before.

Llewellyn's rules may be valid in other situations as well, but these situations may

not be as easy to check as nonnegativity or nonpositivity.

6. Conclusion

Determining all constraints, which are redundant by virtue of one other constraint and non-negativity constraints on all variables, requires solving problem (4.2) for all possible combinations of s and k. This means that by solving $m\,(m-1)$ problems (4.2) only a limited number of redundant constraints can be identified.

Even for this limited class of redundant constraints (redundant by virtue of one other constraint and nonnegativity constraints on all variables) *Llewellyn's* rules only have limited appliance (may only be applied if all coefficients have the same signs).

However, the number of calculations required to check the general conditions (4.8) – (4.9) or (4.11) – (4.12) is practically the same as the number required to apply *Llewellyn's* rules. Therefore one should use the general conditions instead of *Llewellyn's* rules.

If the system of linear inequalities corresponds to a linear programming problem, which is solved by the simplex method, then the tableau of coefficients changes in every iteration. In this case *Llewellyn's* rules or the general conditions may be applied to any new tableau and also to the dual problem, which may enlarge their scope considerably.

However both *Llewellyn's* rules and the general conditions require the availability of all coefficients of the tableau in updated form, which is a rather expensive demand in practical linear programming codes.

Therefore from a practical point of view, it seems best to apply them only to some special problems, in which one might expect a reasonable number of constraints to be identified as being redundant by these rules.

The question, whether or not *Llewellyn's* rules or the general conditions derived in section 4, should be applied to identify redundant constraints may also be considered from a viewpoint based on the theory of computational complexity [see e.g. *Aho/Hopcroft/Ullman*].

Observe that problem (4.2) is a linear programming problem with one constraint in n variables. From the discussion in section 4 we can see, that such an LP problem can be solved in $0\,(n)$ computations. So the total of $m\,(m-1)$ problems may also be solved in a number of computations, which is bounded from above by a polynomial function in the size of the problem. From the theory of computational complexity it is known that this implies, that this problem is in the class P, which means that it is rather easy to solve. Therefore, from the viewpoint of complexity theory, all constraints should be checked for being redundant by virtue of one other constraint and nonnegativity constraints on all variables, if the problem to be solved does not belong to P (and hence is more difficult than any problem in P). Note, that this implies that linear programming problems too should be checked for this kind of redundancy.[3]) If the problem to be solved does belong to P, the degrees of the corresponding polynomial functions should be compared in more detail.

[3]) Assuming that LP is "somewhere in between" P and NP-complete problems.

A similar argument holds for the question whether or not to check for redundancy in general: determining all redundant constraints can be done by solving the problem (4.1) for m different constraints k. The problem (4.1) is a linear programming problem with $(m - 1)$ constraints in n variables and for that problem the complexity status is as yet unknown, although it is generally considered to be more difficult than any problem in P. Therefore in any problem, which is more difficult than linear programming, all redundant constraints should be identified [see also *Telgen*, 1977a].

References

Aho, A.V., J.E. Hopcroft and *J.D. Ullman*: The design and analysis of computer algorithms. Reading, Mass. 1974.

Eckhardt, U.: Redundante Ungleichungen bei linearen Ungleichungssystemen. Unternehmensforschung **12**, 1971, 279–286.

Gal, T.: Zur Identifikation redundanter Nebenbedingungen in linearen Programmen. Zeitschrift für OR **19**, 1975, 19–28.

Llewellyn, R.W.: Linear programming. New York 1964.

Telgen, J.: On redundancy in systems of linear inequalities. Report 7718. Econometric Institute, Erasmus University Rotterdam, 1977a.

– : Redundant and non binding constraints in linear programming problems. Report 7720. Econometric Institute, Erasmus University Rotterdam, 1977b.

Thompson, G.L., F.M. Tonge and *S. Zionts*: Techniques for removing non-binding constraints and extraneous variables from linear programming problems. MS **12** (7), 1966, 588–608.

Zeleny, M.: Linear multiobjective programming. Lecture notes no. 95. Berlin–Heidelberg–New York 1974.