AN INVESTIGATION INTO A METHOD OF DESIGNING INSERTION LOSS LADDER FILTERS AVOIDING INCOMPATIBILITIES.

by

M.H. JOHNSON-ROBERTS.

The thesis is submitted for

'The Degree of Master of Philosophy' of the 'Council for National Academic Awards'.

The research was conducted in the

School of Mathematics, Statistics and Computing,

Thames Polytechnic, Woolwich, London, S.E.18.

MARCH 1977

Contents. Page

Contents

Page

Contents. Page

Acknowledgements.

I wish to express my sincere thanks for the help that my research supervisor, Dr.W. Saraga, has always given so generously. His excitement and enthusiasm for circuit theory are exhilarating and have increased my enjoyment of the research. Furthermore I am grateful to him for providing opportunities for me to study interesting topics and for his encouragement during all the stages of the research and its presentation in this thesis.

My thanks also go to Mr.K.B. Reed, my other research supervisor, for his help and encouragement.

I wish to thank the Director for the use of the many facilities of Thames Polytechnic including those of the Computer Centre, the Central Service Unit and the Library. Amongst the staff of these units I should particularly like to thank Mr.D.T.G. Perkins and Mr.M.T.Ibrahim and Mr. A.H.O. Saad, Mr.F.W.H. Morsley and Miss S.R. Bedwell.

I also wish to thank Mrs.E.B. McFall, who typed the whole thesis so well, particularly for her skilful and accurate typing of the many intricate mathematical expressions.

Abstract

In this thesis Saraga's design method for insertion loss filters is investigated, and an attempt made to assess its practicality. The basic filter design method (Darlington, Cauer) leads to numerical accuracy problems. Accepted methods for dealing with inaccuracy either replace the independent variable (frequency) by the more suitable "z-variable" (Szentirmai, Bingham) or introduce rules for polynomial manipulation based on multiplication in preference to summation (Musson, Norek). In contrast to these approaches, Saraga chooses the dependent variables (network functions) so as to avoid "incompatibilities". Saraga's method, applicable so far only to symmetrical filters (i.e. of odd degree n), had in the past been investigated only for n=7. In this thesis the mathematical results obtained by Saraga are extended and generalised,

practical design tests carried out for n=7(using artificially introduced inaccuracies to test the power of the method to overcome inaccuracies) are supplemented and extended to n=9. Various ways of comparing the practical results of different methods for overcoming numerical accuracy problems are discussed, and one particular method is chosen: to use the different methods to design the same nominal filter, with the same numerical accuracy which is reduced until one method breaks down. A comparison of Saraga's method with Szentirmai's/Bingham's is carried out (and also with Orchard's earlier method). The results are not conclusive; other methods of comparison may have to be used and the comparison will have to be applied to other filters (proposals for further work are made). Some programs developed previously (in a now obsolete language) had to be rewritten and some new filter design programs had to be developed. A sub-program for adjusting the numerical accuracy of any design program to a specified number of significant figures was also developed.

1. Introduction.

1.1. The computing accuracy required in the design of electric filter networks.

Electric filter networks have been used for signal processing in communications systems for many years. In the early days filters were designed semi-intuitively and later by means of the image parameter theory which frequently required trial-and-error modifications to be made to laboratory models. Today, exacting filters are usually designed using the insertion loss method, which involves sophisticated mathematics and computer programs (ref. 1 and ref. 2).

If the design computation is carried out by following "directly" the mathematical equations describing the design method, numerical difficulties frequently arise. To overcome these, high-accuracy arithmetic (20-30 or more significant digits) may have to be used, although final values for the elements are usually only required to four significant digits. Recent methods have attempted to overcome these difficulties in various ways. Two methods are in practical use. The method introduced by Szentirmai (ref. 3) and by Bingham (ref. 4) changes the square of the complex frequency variable p (the independent variable) by a bilinear transformation to avoid the clustering around certain points in the p-plane of the poles and zeros of the insertion voltage ratio, which is one of the causes of the inaccuracies in the direct method; the second method, introduced by Norek (ref. 5) and by Musson (ref. 6), uses polynomials as products of factors instead of in summation form* to avoid the loss of accuracy in the numerical evaluation of a polynomial near its zeros. A third method

---

* see footnote on next page.

proposed by Saraga (ref. 7) accepts that certain numerical values will be obtained inaccurately but reduces the required degree of accuracy by avoiding conflicting inaccuracies i.e. by avoiding incompatibilities as will be explained later. This method has so far been developed only for filters of order n=5 and n=7 but has not yet been fully investigated as far as its practical application is concerned.

1.2. The aim of this investigation.

The aim of the present research is to further investigate and develop Saraga's method and to study its practical value. Because preliminary investigations yielded encouraging results, it was decided to compare it with some of the established methods in order to assess its value for practical design purposes.

To set this in context, it is necessary to describe Saraga's method in some detail; and this has to be done against the background of the conventional insertion-loss design procedure.

1.3. Conventional insertion-loss filter design.

The loss, as a function of the frequency f, is considered here for a passive, purely-reactive, filter network, resistively

---

footnote from previous page.
*A polynomial A(p) in summation form is

$$A(p)=a_0+a_1p+a_2p^2+ \dots +a_ip^i+ \dots +a_np^n = \sum_{i=0}^{n} a_ip^i$$

where $a_0, a_1, a_2, \dots a_i, \dots a_n$ are constant coefficients. The same polynomial in product form is

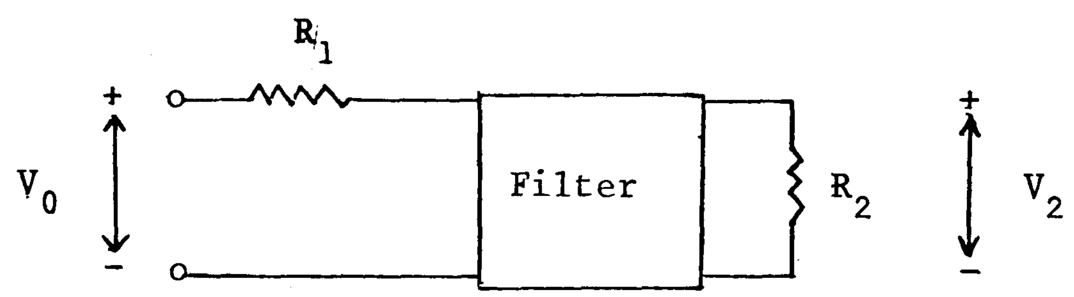
$$A(p)=C(p-p_1)(p-p_2) \dots (p-p_i) \dots (p-p_n)=C \prod_{i=1}^{n}(p-p_i)$$

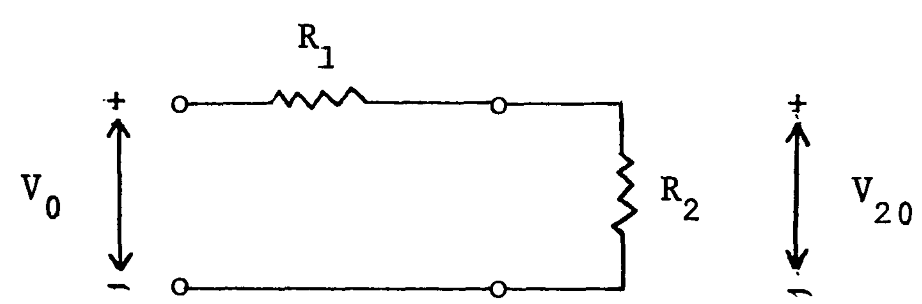
C being constant and $p_1, p_2, \dots p_i, \dots p_n$ being the zeros of A(p).

terminated at both ports. The function of interest is the voltage insertion ratio H(p) defined in the figure below, where p is the normalised complex frequency variable

$$p=\frac{\sigma+j\omega}{\omega_r}, \qquad \omega=2\pi f, \qquad \omega_r=2\pi f_r$$

and f_r is a conveniently chosen reference frequency (e.g. the end of the passband).





$$H(p)=\frac{V_{20}(p)}{V_2(p)}$$

For symmetrical filters R_2 is chosen to be equal to R_1 (for a definition of symmetry see page 6).

H(p) is a real rational function of p with zeros only in the left half of the p-plane. The loss L (in dB) is defined by

$$L=10\log_{10}\left|H(p)H(-p)\right|_{p=j\omega/\omega_r} \tag{1.1}$$

A second real rational function, the "characteristic function", K(p)* is introduced by the equation

$$H(p)H(-p)=1+K(p)K(-p) \tag{1.2}$$

To make possible the realisation of the filter in ladder form, it is necessary that the poles of K(p), which are identical with the poles of H(p), lie on the imaginary p-axis, i.e. the real

---

* A discussion of K(p) and H(p) is given on page 13.

frequency axis, and they will have been chosen as conjugate pairs, or at infinity, or at zero.

Symmetrical filters (i.e. filters which are electrically - not necessarily structurally - symmetrical with respect to the input port and the output port) only will be considered. In this case K(p) is an odd real rational function of p and equation (1.2) becomes

$$H(p)H(-p)=1-K^2(p)=\left[1+K(p)\right]\left[1-K(p)\right] \tag{1.3}$$

Thus equation (1.1) can also be written in the form

$$L=10\ \log_{10}\left|1-K^2(p)\right|_{p=j\omega/\omega_r} \tag{1.4}$$

The conventional design procedure consists of two stages, approximation and synthesis:-

(i) Approximation.

The design specification is most often stated in terms of a permissible maximum loss in the passband and of loss minima in the stopband. The usual procedure is to find, analytically, graphically or by a computational trial-and-error process (optimisation), a suitable function K(p) which, via equation (1.4), satisfies the loss specification.

The approximation stage is not the subject of this research and it will be assumed that a suitable K(p) function is given.

(ii) Synthesis.

(a) Determination of open- and short-circuit impedances from K(p).

K(p) can be written in the form

$$K(p)=N(p)/D(p) \tag{1.5}$$

where N(p) and D(p) are real polynomials in p and must be relatively prime. One of them must be odd and the other even; usually D(p) is chosen as the even polynomial (and this will be done in this thesis) but if D(p) is an odd polynomial, a similar argument will lead to similar expressions. Then from equations (1.3) and (1.5)

$$H(p)H(-p)=\frac{D^2(p)-N^2(p)}{D^2(p)}=\frac{[D(p)+N(p)][D(p)-N(p)]}{D^2(p)}$$

The numerator of the right hand side can be written in the form

$$D^2(p)-N^2(p)=[D(p)+N(p)]\,[D(p)-N(p)]=U(p)U(-p) \quad *$$

where $U(p)$ is a Hurwitz polynomial (i.e. all its zeros lie in the left half of the p-plane). Then

$$H(p)=\pm\frac{U(p)}{D(p)}$$

and $U(p)$ can be found <u>either</u> by factorising $D^2(p)-N^2(p)$ and using all the Hurwitz factors to form $U(p)$ <u>or</u> by factorising $D(p)+N(p)$ in the form

$$D(p)+N(p)=B_a(p)B_b(-p) \tag{1.6}$$

where both $B_a(p)$ and $B_b(p)$ are Hurwitz polynomials.**

Then

$$H(p)=\pm\frac{B_a(p)B_b(p)}{D(p)} \tag{1.7}$$

Having determined H(p) from K(p), the open-circuit impedance Z_o and the short-circuit impedance Z_s can both be found from expressions of the form

$$Z_o\ ,\ Z_s=\pm\left(\frac{H_o\pm K}{H_e}\right)^{\pm 1} \tag{1.8}$$

where H_e and H_o are the even and odd parts of H(p).

The present research is mainly concerned with investigating a new procedure by Saraga for finding Z_o and/or Z_s from K(p).

(b) <u>Realisation</u>.

In this stage of the design procedure, a suitable ladder filter

---

* see Appendix 1.

** see Appendix 2.

structure is chosen and the element values are found from the knowledge of Z_o and/or Z_s and from the knowledge of pole frequencies of $K(p)$. For a symmetrical low-pass filter the only freedom in the choice of the ladder network (apart from choosing its dual) lies in the sequence, from input to output, in which the attenuation poles are realised as series resonant circuits in the shunt arms or as parallel resonant circuits in the series arms. Because it will frequently be necessary to refer to the realisation stage, a brief description of the realisation of ladder networks is given in Appendix 3.

1.4. Computing inaccuracy problem in conventional filter design.

In the conventional method described above there are two main places where accuracy difficulties occur:-

(i) As explained in the previous section, the expression for $D(p)+N(p)$ has to be factorised to find the zeros of $H(p)$. Some of the roots are likely to be very close together around a finite non-zero frequency (in the case of "normalised" low pass filters usually around $p=j$, i.e. $\omega/\omega_r=1$). Therefore unless very high accuracy arithmetic is used, the zeros of $D(p)+N(p)$, and therefore $H(p)$ itself, are determined inaccurately.

(ii) In the realisation process, elements are calculated by evaluating the numerator and denominator polynomials of appropriate rational immittance (i.e. admittance or impedance) functions,taken in the summation form $\sum a_i p^i$. There is frequently a high loss of accuracy as the result of the subtraction of nearly equal quantities*. Thus the element value is obtained to a lower accuracy than the coefficients in the immittance from which it has been calculated;

* see footnote on next page.

when the immittance of the element is subtracted from the total immittance the resulting new immittance is less accurate than the previous one. The next element is therefore calculated less accurately still, and several such stages will often produce wrong results if arithmetic of too low an accuracy is used. That the results are wrong is demonstrated in such cases by the fact that quite different element values are obtained depending on whether the design is started from one end of the filter or the other.

---

footnote from previous page.

*Such a high loss of accuracy is evident in the evaluation of the polynomial

$$p^8+5.15p^6+9.815p^4+8.2165p^2+2.55255$$

at $p=j\sqrt{1.091537001}$, $j=\sqrt{-1}$. When, as frequently happens, nested multiplication is employed, i.e. the calculation is performed in the sequence

$$(((p^2+5.15)p^2+9.815)p^2+8.2165)p^2+2.55255.$$

The intermediate values in the calculation are shown in the following table for two cases: for case (a) an accuracy of 6 significant figures is used throughout, rounding at each step whereas for case (b) the maximum accuracy on a Texas SR50 calculator is retained, i.e. 10 significant figures are used as data and shown by the display but 12 significant figures are used for the calculations because the Texas carries an extra 2 guard figures internally.

Table 1. Evaluation of a polynomial.

Operation and operand	Value for case (a)	Value for case (b)
Enter p^2	-1.09154	-1.091537001
+5.15	4.05846	4.058462999
$\times p^2$	-4.42997	-4.429962531
+9.815	5.38503	5.385037469
$\times p^2$	-5.87798	-5.87796765
+8.2165	2.33852	2.33853235
$\times p^2$	-2.55259	-2.552594588
+2.55255	-0.0000400000	-0.0000445884

In adding the last value in column 1 to the preceding values which are in columns 2 and 3 and are negative, there is a loss of 5 significant figures and case (a) gives an answer accurate to only 1 significant figure.

In order to overcome the difficulties produced by these inaccuracies, various methods have been developed, for instance by Szentirmai and by Bingham (independently). A different method has been developed by Norek and by Musson (also independently), and another totally different method has been proposed by Saraga. The methods by Szentirmai/Bingham and Musson/Norek are in practical use, but that by Saraga is still being investigated and compared with the others.

The most widely used of these methods at present is the one developed by Szentirmai and Bingham. It is of interest to state very briefly the basic idea. Similar steps are performed to those for the traditional method but in the z-plane, instead of the p-plane, where for low pass filters $z^2 = 1 + \frac{1}{p^2}$. The great advantage is that whereas often the zeros of H in the p-plane are near $j=\sqrt{-1}$ (for normalised filters), they are scattered around zero in the z-plane. This means that the same number of significant figures for the positions of the zeros of H contain more information in the z-plane than in the p-plane, e.g. 1.00054 and 1.00053 to 6 significant figures are less useful than 0.000543765 and 0.000532941. The Szentirmai/Bingham method uses the z-plane for all stages of the design, including the realisation of the element values, and it is so good that it has in most design problems reduced the high accuracy previously required to only 10 significant figures.*

---

*The z variable was originally introduced by Orchard but only for the approximation part and the first part of the synthesis. He calculated the factors of the numerator of H in the z-plane and then converted them to the p-plane using for the last design stage, the conventional p-plane.

The method used by Musson and Norek uses polynomials in factor form instead of summation form to avoid a loss of accuracy in the evaluation. Thus, for example, the following function H(p) can be written in factor and summation forms:-

$$H(p)=(p^2+1)(p^2+1.05)(p^2+1.15)(p^2+1.175)$$

$$H(p)=p^8+4.375p^6+7.1675p^4+5.2113125p^2+1.4188125.$$

Evaluation of H(p) at $p^2=-1.1$ gives the correct value 0.00001875 for both forms when high accuracy arithmetic is used. If however only 4 significant figures are permitted the first expression gives the correct value, while the second gives the value of 0.001.

In a design, polynomials often have to be added to each other, or subtracted from each other many times. The Musson/Norek method requires the resulting polynomial to be obtained directly in product form. To achieve this, the derivatives of the polynomials are arranged as sums of products and Newton's root-finding method is then applied. Some details of the technique are given in Appendix 4. This filter design method requires many more computations to be performed than the conventional method or the Szentirmai/Bingham method.

Orchard (ref.8) compares the Musson/Norek method with the Szentirmai/Bingham method (i.e. the "product" method with the "transformed-variable" method) as follows:

> "The product method is conceptually simple and easy to program. It requires, however, a factoring of a linear combination of factored polynomials in every step of the design process, and hence it leads to even more lengthy computations than would multiple precision. The transformed-variable method ... is somewhat harder to understand and to program, but it is very fast to execute and the improvement in accuracy is greater than that obtainable with the product method."

He goes on to make the point that the two methods can be combined, carrying out all the computations in terms of factored polynomials in the transformed variable. Similarly, Saraga points out that his method could be combined with the Szentirmai/Bingham method.

2. Description of Saraga's method.

The basic relation between $H(p)$ and $K(p)$ is again considered when $K(p)$ is an odd function (symmetrical filters), from equation (1.3)

$$H(p)H(-p) = 1 - K^2(p) = \left[1 + K(p)\right]\left[1 - K(p)\right] \tag{2.1}$$

If $K(p)$ is given (e.g. specified by an approximation to the performance requirements), then $H(p)H(-p)$ is uniquely determined by (2.1) and $H(p)$ is also uniquely determined (apart from an irrelevant factor ± 1). This is now shown: from

$$K(p) = \frac{N(p)}{D(p)} \tag{2.2}$$

$$V(p) \equiv N(p) + D(p) \tag{2.3}$$

is obtained. This polynomial is factorised in the form

$$V(p) = B_a(p)B_b(-p)$$

where $B_a(p)$ and $B_b(p)$ are Hurwitz polynomials determined by factorising the polynomial into Hurwitz and anti-Hurwitz polynomials. Then $H(p)$ is obtained in the form

$$H(p) = \frac{B_a(p)B_b(p)}{D(p)} \equiv \frac{U(p)}{D(p)} \tag{2.4}$$

Note that

$$1 + K(p) = \frac{V(p)}{D(p)} \tag{2.5}$$

As long as the purely mathematical - as distinct from the numerical - aspects of the design are considered, after $B_a(p)$ and $B_b(-p)$ have been obtained from $K(p)$, it would be possible to consider $B_a(p)$ and $B_b(-p)$ as the basic information from which both $K(p)$ and $H(p)$ can be uniquely derived by means of

$$V(p) = B_a(p)B_b(-p) \tag{2.6}$$

$$U(p) = B_a(p)B_b(p) \tag{2.7}$$

$$K(p) = \frac{\text{Odd}\,[V(p)]}{\text{Even}\,[V(p)]} \tag{2.8}$$

$$H(p) = \frac{U(p)}{\text{Even}\,[V(p)]} \tag{2.9}$$

It is now necessary to consider the computational aspects of equations (2.6) to (2.9). Computational inaccuracies in the factorisation of $V(p)$ i.e. $N(p) + D(p)$ cause the inaccurate functions $B'_a(p)$ and $B'_b(-p)$ to be obtained (instead of $B_a(p)$ and $B_b(-p)$). In conventional filter design this inaccuracy is made sufficiently small by using high accuracy arithmetic to be acceptable, but it is necessary to investigate in detail what is actually done.

From $B'_a(p)$ and $B'_b(-p)$ an inaccurate function

$$U'(p) = B'_a(p)B'_b(p) \tag{2.10}$$

is formed and then $H(p)$ is taken as $\frac{U'(p)}{D(p)}$. Since $U'(p) \neq U(p)$ the function $\frac{U'(p)}{D(p)}$ is not identical with $H(p)$ and must be denoted as

$$H'(p) = \frac{U'(p)}{D(p)} \tag{2.11}$$

It should be noted that in (2.11) an inaccurate numerator is combined with an accurate denominator. It is necessary to investigate the consequences of such a combination.

It would have been possible to proceed in a different way. The functions $B'_a(p)$ and $B'_b(-p)$ could have been taken as <u>new</u> basic information (replacing the <u>unknown</u> exact $B_a(p)$ and $B_b(-p)$). Then, as shown in the footnote, equations (2.6) to (2.9) could be used to define new functions $K'(p)$ and $H'(p)$ (note that this function $H'(p)$ is not identical with $H'(p)$ defined by (2.11)). These functions $K'(p)$ and $H'(p)$ would satisfy equation (2.1) as is also shown in the footnote. It is obvious that $K(p)$ and $H'(p)$ defined by (2.11)

---

* $$V'(p) \equiv B'_a(p)B'_b(-p)$$

$$U'(p) \equiv B'_a(p)B'_b(p)$$

$$K'(p) \equiv \frac{\mathrm{odd}\,[V'(p)]}{\mathrm{even}[V'(p)]}$$

$$H'(p) \equiv \frac{U'(p)}{\mathrm{even}[V'(p)]}$$

The left hand side of equation (2.1) with $H(p)$ replaced by $H'(p)$ becomes

$$\frac{U'(p)U'(-p)}{\mathrm{even}\,[V'(p)]\mathrm{even}\,[V'(-p)]} = \frac{B'_a(p)B'_b(p)B'_a(-p)B'_b(-p)}{\{\mathrm{even}\,[V'(p)]\}^2}$$

and the right hand side with $K(p)$ replaced by $K'(p)$ becomes

$$\left[1 + \frac{\mathrm{odd}[V'(p)]}{\mathrm{even}[V'(p)]}\right]\left[1 - \frac{\mathrm{odd}[V'(p)]}{\mathrm{even}[V'(p)]}\right]$$

$$= \left[\frac{\mathrm{even}\,[V'(p)] + \mathrm{odd}[V'(p)]}{\mathrm{even}[V'(p)]}\right] \times \left[\frac{\mathrm{even}\,[V'(p)] - \mathrm{odd}\,[V'(p)]}{\mathrm{even}\,[V'(p)]}\right]$$

$$= \frac{V'(p)V'(-p)}{\{\mathrm{even}[V'(p)]\}^2} = \frac{B'_a(p)B'_b(-p)B'_a(-p)B'_b(p)}{\{\mathrm{even}[V'(p)]\}^2}$$

Equation (2.1) is therefore satisfied by this choice of $K'(p)$ and $H'(p)$.

would not satisfy equation (2.1).

It is interesting to consider all possible definitions of H'(p) and K'(p). These are

$$H'(p) \equiv \frac{U(p)}{D(p)} = H(p) \qquad H'(p) \equiv \frac{U'(p)}{D'(p)}$$

$$H'(p) \equiv \frac{U'(p)}{D(p)} \qquad H'(p) \equiv \frac{U(p)}{D(p)}$$

and
$$K'(p) \equiv \frac{N(p)}{D(p)} = K(p) \qquad K'(p) \equiv \frac{N'(p)}{D'(p)}$$

$$K'(p) \equiv \frac{N'(p)}{D(p)} \qquad K'(p) \equiv \frac{N(p)}{D'(p)}$$

where any pair of H'(p) and K'(p) might be taken together. However inspection shows that there are only 2 cases in which equation (2.1) is satisfied: the nominal case $H'(p) = \frac{U(p)}{D(p)}$, $K'(p) = \frac{N(p)}{D(p)}$ which is not available in practice because U(p) cannot be obtained exactly and the case

$$H'(p) = \frac{U'(p)}{D'(p)} \quad , \quad K'(p) = \frac{N'(p)}{D'(p)} \qquad (2.12)$$

These two cases will be considered as giving "compatible" design parameters - because equation (2.1) is satisfied - whereas all the other cases give incompatible information.

It is one of Saraga's basic ideas to avoid such incompatibilities. He argues as follows: the conventional method can only succeed, in spite of using incompatible design parameters by making these incompatibilities exceedingly small; this can in the conventional method only be achieved by using exceedingly high accuracy in the computations. In contrast to this,he conjectures that if only mutually compatible parameters are used the need for such high accuracy disappears. In the absence of incompatibilities the main consequence of inaccurate computation will be that the filter actually obtained will have a performance characteristic deviating from the nominal one. However fairly large deviations of this kind are usually acceptable, and the computing accuracy required to avoid unacceptable performance should be much smaller than that required for the design of a filter from the incompatible parameters.

The choice of $H'(p)$ and $K'(p)$ in accordance with equations (2.12) has been shown to satisfy equation (2.1) and therefore to eliminate the type of incompatibility considered above. However a second type of incompatibility occurs and how it arises will be considered below. As mentioned in section 1.3, the element values of the filters are found from the knowledge of the pole frequencies p_∞ of $K(p)$ (i.e. the zeros of $D(p)$) and the knowledge of the open and/or short circuit impedances as determined from expressions of the form (see equation (1.8))

$$Z_O,\ Z_S \;=\; \pm \left(\frac{H_O \pm K}{H_e}\right)^{\pm 1} \tag{2.13}$$

in which H_e and H_O are the even and odd parts of the rational function $H(p)$.

However since Z_O and/or Z_S are obtained inaccurately as Z_O' and Z_S' from H' or K', and since the poles p_∞' of H' and K' differ from the poles p_∞ of H and K, Z_O' and Z_S' are not compatible with the poles of H and K. This means that at the original pole frequencies $p = p_\infty$, $Z_O = Z_S$ but $Z_O' \neq Z_S'$. As a consequence of the inequality $Z_O' \neq Z_S'$ at $p = p_\infty$, different element values are obtained depending on which of the impedances Z_O' and Z_S' is used as "basic information" for obtaining these values. Therefore the question arises: which value is the correct one? In fact it is neither.

It can also be shown that whereas with correct Z_O and Z_S functions and the correct pole frequencies the same network is obtained whether the design starts from one end of the filter or the other, with an incompatibility between Z_O', Z_S' and p_∞ the two realisations starting at opposite ends do not "meet".

In order to avoid these incompatibilities the pole frequencies p_∞' of H' and K' might be determined and used instead of the pole frequencies p_∞. However, this would mean that the denominator polynomial D'(p) would have to be factorised to find the pole frequencies of K'(p). Not only would this introduce further inaccuracies but there would be the risk of the pole frequencies p_i being complex, not purely imaginary as is necessary for a LC ladder network realisation. For these reasons the following method is used instead of the methods outlined above.

A function K(p) is specified as before, however the aim of the design procedure, the determination of the open and short circuit impedances Z_O and Z_S is not achieved by determining the function H(p) but by using instead the impedances Z_a and Z_b of the equivalent lattice network.* The new method is described in this section first in terms of the accurate parameters; the effect of inaccuracies will be described later.

The lattice impedances Z_a and Z_b can be obtained from the polynomials $B_a(p)$ and $B_b(p)$ by means of the equations

---

* Any physically realisable purely reactive symmetrical filter possesses a physically realisable lattice equivalent where the lattice impedances Z_a and Z_b satisfy the reactance theorem.

$$Z_a = \frac{O_a}{E_a} \quad \text{and} \quad Z_b = \frac{E_b}{O_b}$$

or alternatively*

$$Z_a = \frac{E_a}{O_a} \quad \text{and} \quad Z_b = \frac{O_b}{E_b} \tag{2.14}$$

where

$$\left.\begin{aligned} E_a &= \text{Even}\left[B_a(p)\right] \\ O_a &= \text{Odd}\left[B_a(p)\right] \\ E_b &= \text{Even}\left[B_b(p)\right] \\ O_b &= \text{Odd}\left[B_b(p)\right] \end{aligned}\right\} \tag{2.15}$$

Moreover impedances Z_a and Z_b are related to the function $K(p)$ by the equation

$$K(p) = \frac{Z_a Z_b - 1}{Z_b - Z_a} \tag{2.16}$$

Once the lattice impedances Z_a, Z_b have been found, the open and short circuit impedances Z_o and Z_s can be determined by means of the equations

$$\left.\begin{aligned} Z_o &= \frac{Z_a + Z_b}{2} \\ Y_s &= \frac{Y_a + Y_b}{2} \end{aligned}\right\} \tag{2.17}$$

where

$$Y_a = \frac{1}{Z_a} \ , \quad Y_b = \frac{1}{Z_b} \ , \quad Y_s = \frac{1}{Z_s} \ . \tag{2.18}$$

However because of numerical inaccuracies, instead of the exact impedances Z_a and Z_b, inaccurate impedances Z'_a and Z'_b

---

* It does not matter which choice is taken.

will be obtained. Nevertheless a symmetrical network must exist for purely reactive lattice impedances Z'_a and Z'_b and thus an incompatibility of the type described as the first incompatibility between H(p) and K(p) cannot arise. However, Z'_a and Z'_b must also satisfy the reactance theorem. If this is the case (this will be discussed later) the only effect of the inaccuracies of Z'_a and Z'_b is that the actual loss/frequency curve will vary from the nominal one (this point has been discussed earlier).

Ultimately a ladder filter is required and therefore, as will now be shown, the second kind of incompatibility is still present. When equations (1.4) and (2.16) are written in terms of the inaccurate instead of the accurate parameters, they yield

$$L = 10 \log_{10} \left| 1 - K'^2(p) \right|_{p=j\omega/\omega_r} \tag{2.19}$$

and

$$K'(p) = \frac{Z'_a Z'_b - 1}{Z'_b - Z'_a} \tag{2.20}$$

respectively and show that the loss poles occur at the frequencies $p'_\infty = j\omega'_\infty$ at which

$$Z'_a = Z'_b \quad . \tag{2.21}$$

However these will in general not be the same as those for the poles p_∞ of K(p). Therefore, the second incompatibility still arises if Z'_o and Z'_s are obtained from Z'_a and Z'_b using equations (2.17) and (2.18), and the ladder filter is realised from Z'_o, Z'_s and p_∞. This difficulty, as mentioned before, could in principle be overcome by determining the p'_∞, at which equation (2.21) is satisfied. However the following method will be used instead. Some of the relevant parameters of Z'_a and Z'_b are altered to force equation (2.21) to be valid at the nominal frequencies of the attenuation poles, i.e. at $\omega=\omega_\infty$. This method is preferred for the two reasons that no further root finding is necessary and, of equal importance, it ensures that the p_∞ are purely imaginary which (as mentioned before) is a necessary condition for the network to be realisable in ladder form.

For a low-pass filter of (odd) order n, it can always be assumed that $(n-1)/2$ zeros of N+D lie on the left hand side of the p-plane and $(n+1)/2$ zeros on the right hand side, or vice versa. Therefore, impedances Z'_a and Z'_b, or rather the coefficients M'_i, $i=1,2,\ldots(n-1)/2$ and N'_i, $i=1,2,\ldots(n+1)/2$ of powers of p in the quadratic and linear factors of B'_a and B'_b, are improved by the process described below and produce new "improved" values for B'_a and B'_b. (For an exact definition of the coefficients M_i and N_i, in the special case n=9, see equations (3.3) and (3.6)). The process is applied as many times as is necessary to make the loss values, obtained in the way to be described, satisfy the design specifications. The latest values of the improved parameters B'_a and B'_b are used in place of the accurate parameters in equations (2.14), (2.15), and the lattice impedances so obtained will be called Z'_{ai}, Z'_{bi}. The loss is then obtained from equation (2.19) after using Z'_{ai} and Z'_{bi} instead of Z'_a and Z'_b in equation (2.20).

For the improvement process, the coefficients a'_i, $i = 1,2,\ldots \frac{n-1}{2}$ and b'_i, $i = 1,2,\ldots \frac{n+1}{2}$ of powers of p in the rational functions Z'_a and Z'_b are calculated in terms of the coefficients M'_i and N'_i; the coefficients k'_i, $i = 1,2,\ldots n$ of powers of p in the rational function $K'(p)$ are calculated in terms of the a'_i and b'_i coefficients and thence in terms of the M'_i and N'_i coefficients.* For a fuller account of the process, for the special case n=9, see section 3.1.

---

*The M'_i and N'_i coefficients are improved in preference to the a'_i and b'_i coefficients because it is easy to check that the M'_i and N'_i do not change their signs which means that the factors remain Hurwitz, i.e. that none of the roots move across the imaginary axis, whereas more work would be required to check that Z'_a and Z'_b still satisfied the reactance theorem.

In the resulting equations the inaccurate k_i', M_i', N_i' coefficients are written in terms of the related accurate coefficients k_i, M_i, N_i and their error terms Δk_i, ΔM_i, ΔN_i, i.e. $k_i' = k_i + \Delta k_i$, $M_i' = M_i + \Delta M_i$, $N_i' = N_i + \Delta N_i$, the expressions in the equations are then expanded and the second and higher order error terms are neglected. Similar equations hold for the accurate coefficients and are used to remove the most significant terms, the resulting n equations being linear in the unknown error terms ΔM_i, ΔN_i, and the known error terms Δk_i. These equations are then solved by Gauss' method with partial pivotting, see ref.9, and new inaccurate values are obtained for the M_i' and N_i' coefficients. As mentioned before, this process is repeated as many times as is necessary until the filter specification is satisfied.

The a_i, b_i coefficients are calculated from the latest values of the M_i' and N_i' coefficients and give the incompatible lattice impedances Z_{ai}', Z_{bi}'. Then for an n^{th} order filter, $\frac{n-1}{2}$ of the n coefficients belonging to Z_a' and Z_b' are altered to make the denominator polynomial of $K'(p)$ the same as that of $K(p)$. In this way the frequencies of the original attenuation poles will be compatible with those of $K'(p)$ without having to find the zeros of the polynomial denominator of $K'(p)$, which would again introduce inaccuracies. Various choices of the particular coefficients to be altered are possible but care is taken to choose those which lead to linear, not to non-linear, equations. The resulting equations, $\frac{n-1}{2}$ in number, are solved by Gauss' method with partial pivotting.

The compatible impedances Z_a' and Z_b' thus obtained, now called Z_{ac}' and Z_{bc}', are used to give the open and short circuit impedances of a ladder network by means of the equations (2.17) and (2.18), and together with the frequencies of the original attenuation poles, will lead, on applying the realisation techniques of appendix 3, to the element values of the ladder network.

3. Extension of Saraga's Method.

3.1. Symmetrical filters of order n=9.

Following the method used by Saraga for symmetrical filters of orders 5 and 7 (ref.7), the same method will now be applied to one of order 9.

Let $N=k_1p+k_3p^3+ \ldots k_9p^9$

and $D=1+k_2p^2+ \ldots +k_8p^8$

so that

$$K(p)=\frac{N}{D}=\frac{k_1p+k_3p^3+k_5p^5+k_7p^7+k_9p^9}{1+k_2p^2+k_4p^4+k_6p^6+k_8p^8} \tag{3.1}$$

Then N+D has to be factorised. Assuming it contains 4 Hurwitz and 5 anti-Hurwitz factors (it can contain alternatively 5 Hurwitz and 4 anti-Hurwitz factors but this means only that the expressions for Z_a and Z_b found later from equation (3.11) are interchanged and inverted).

$$N+D=(1+M_1p+M_2p^2)(1+M_3p+M_4p^2)(1-N_1p+N_2p^2)(1-N_3p+N_4p^2)(1-N_5p) \tag{3.2}$$

where all the M_i and N_i coefficients have positive values.

Now

$$B_a(p)=(1+M_1p+M_2p^2)(1+M_3p+M_4p^2) \tag{3.3}$$

$$=1+a_1p+a_2p^2+a_3p^3+a_4p^4 \tag{3.4}$$

$$=E_a+O_a \tag{3.5}$$

and

$$B_b(p)=(1-N_1p+N_2p^2)(1-N_3p+N_4p^2)(1-N_5p) \tag{3.6}$$

$$=1-b_1p+b_2p^2-b_3p^3+b_4p^4-b_5p^5 \tag{3.7}$$

$$=E_b-O_b \tag{3.8}$$

Equating coefficients of p in (3.3) and (3.4) gives

$$\left.\begin{aligned}
a_1 &= M_1+M_3\\
a_2 &= M_1M_3+M_2+M_4\\
a_3 &= M_1M_4+M_2M_3\\
a_4 &= M_2M_4
\end{aligned}\right\} \tag{3.9}$$

and in (3.6) and (3.7) gives

$$\left.\begin{aligned}
b_1 &= N_1+N_3+N_5\\
b_2 &= N_2+N_4+N_1N_3+N_5(N_1+N_3)\\
b_3 &= N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3)\\
b_4 &= N_2N_4+N_5(N_2N_3+N_1N_4)\\
b_5 &= N_2N_4N_5
\end{aligned}\right\} \tag{3.10}$$

The next step is to find Z_a and Z_b using equations (2.14), (3.4), (3.5), (3.7) and (3.8):

$$\left.\begin{aligned}
Z_a &= \frac{a_1p+a_3p^3}{1+a_2p^2+a_4p^4}\\
Z_b &= \frac{1+b_2p^2+b_4p^4}{b_1p+b_3p^3+b_5p^5}
\end{aligned}\right\} \tag{3.11}$$

After substituting into equation (2.16), the coefficients of p are equated with those in equation (3.1) and yield, for the numerator

$$\left.\begin{aligned}
k_1 &= a_1-b_1\\
k_3 &= a_3+a_1b_2-b_3-a_2b_1\\
k_5 &= a_3b_2+a_1b_4-a_4b_1-a_2b_3-b_5\\
k_7 &= a_3b_4-a_4b_3-a_2b_5\\
k_9 &= -a_4b_5
\end{aligned}\right\} \tag{3.12}$$

$$\left.\begin{aligned}
k_2&=a_2+b_2-a_1b_1\\
k_4&=a_4+b_4+a_2b_2-a_3b_1-a_1b_3\\
k_6&=a_2b_4+a_4b_2-a_3b_3-a_1b_5\\
k_8&=a_4b_4-a_3b_5
\end{aligned}\right\}\qquad(3.13)$$

The coefficients a_i and b_i are then replaced by M_i and N_i using equations (3.9) and (3.10) respectively: thus

$$\left.\begin{aligned}
k_1&=M_1+M_3-(N_1+N_3+N_5)\\
k_3&=M_1M_4+M_2M_3+(M_1+M_3)(N_2+N_4+N_1N_3+N_5(N_1+N_3))\\
&\quad-(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))-(M_1M_3+M_2+M_4)(N_1+N_3+N_5)\\
k_5&=(M_1M_4+M_2M_3)(N_2+N_4+N_1N_3+N_5(N_1+N_3))\\
&\quad+(M_1+M_3)(N_2N_4+N_5(N_2N_3+N_1N_4))-M_2M_4(N_1+N_3+N_5)\\
&\quad-(M_1M_3+M_2+M_4)(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))-N_2N_4N_5\\
k_7&=(M_1M_4+M_2M_3)(N_2N_4+N_5(N_2N_3+N_1N_4))\\
&\quad-M_2M_4(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))-(M_1M_3+M_2+M_4)N_2N_4N_5\\
k_9&=-M_2M_4N_2N_4N_5
\end{aligned}\right\}\qquad(3.14)$$

and

$$\left.\begin{aligned}
k_2&=M_1M_3+M_2+M_4+N_2+N_4+N_1N_3+N_5(N_1+N_3)-(M_1+M_3)(N_1+N_3+N_5)\\
k_4&=M_2M_4+N_2N_4+N_5(N_2N_3+N_1N_4)\\
&\quad+(M_1M_3+M_2+M_4)(N_2+N_4+N_1N_3+N_5(N_1+N_3))\\
&\quad-(M_1M_4+M_2M_3)(N_1+N_3+N_5)\\
&\quad-(M_1+M_3)(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))\\
k_6&=(M_1M_3+M_2+M_4)(N_2N_4+N_5(N_2N_3+N_1N_4))\\
&\quad+M_2M_4(N_2+N_4+N_1N_3+N_5(N_1+N_3))\\
&\quad-(M_1M_4+M_2M_3)(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))-(M_1+M_3)N_2N_4N_5\\
k_8&=M_2M_4(N_2N_4+N_5(N_2N_3+N_1N_4))-(M_1M_4+M_2M_3)N_2N_4N_5
\end{aligned}\right\}\qquad(3.15)$$

In practice inaccurate values M'_i $i=1,2,\ldots 4$, and N'_i $i=1,2,\ldots 5$, are found instead of M_i, N_i and lead to $B'_a(p)$, $B'_b(p)$, E'_a, O'_a, E'_b, O'_b and hence to Z'_a, Z'_b in place of the corresponding accurate parameters. Thus equations (3.3) to (3.15) are replaced by equations in terms of the inaccurate parameters, e.g. equations (3.3) to (3.5) are replaced by

$$\left.\begin{aligned} B'_a(p) &= (1+M'_1p+M'_2p^2)(1+M'_3p+M'_4p^2) \\ &= 1+a'_1p+a'_2p^2+a'_3p^3+a'_4p^4 \\ &= E'_a+O'_a \end{aligned}\right\} \quad (3.16)$$

If the loss calculated from equation (1.4) using K' instead of K, fails to satisfy the filter specification, the coefficients M'_i and N'_i must be improved in accuracy. To ensure that none of the roots move across the imaginary p axis, coefficients M'_i and N'_i are used in preference to a'_i and b'_i as explained in section 2. The nine equations based on (3.14) and (3.15), but in terms of the inaccurate parameters k'_i, M'_i, N'_i, can be rewritten in terms of the errors, Δk_i, ΔM_i, ΔN_i by substituting

$$\left.\begin{aligned} k'_i &= k_i+\Delta k_i \ , \quad i=1,2,\ldots 9 \\ M'_i &= M_i+\Delta M_i \ , \quad i=1,2,\ldots 4 \\ N'_i &= N_i+\Delta N_i \ , \quad i=1,2,\ldots 5 \end{aligned}\right\} \quad (3.17)$$

The most significant terms i.e. those terms not including error terms are then removed with the aid of equations (3.14) and (3.15). If second and higher order error terms are neglected, nine simultaneous equations which relate the unknowns ΔM_i, $i=1,2,\ldots 4$, ΔN_i, $i=1,2,\ldots 5$ to the known errors Δk_i, $i=1,2,\ldots 9$, are obtained, some of which are given:

$$\Delta k_1=\Delta M_1+\Delta M_3-\Delta N_1-\Delta N_3-\Delta N_5$$

$$\Delta k_3=\Delta M_1(M_4+N_2+N_4+N_1N_3+N_5(N_1+N_3)-M_3(N_1+N_3+N_5))$$

$$+\Delta M_2(M_3-(N_1+N_3+N_5))$$

$$+\Delta M_3(M_2+N_2+N_4+N_1N_3+N_5(N_1+N_3)-M_1(N_1+N_3+N_5))$$

$$+\Delta M_4(M_1-(N_1+N_3+N_5))$$

$$+\Delta N_1((M_1+M_3)(N_3+N_5)-(N_4+N_5N_3)-(M_1M_3+M_2+M_4))$$

$$+\Delta N_2((M_1+M_3)-(N_3+N_5))$$

$$+\Delta N_3((M_1+M_3)(N_1+N_5)-(N_2+N_5N_1)-(M_1M_3+M_2+M_4))$$

$$+\Delta N_4((M_1+M_3)-(N_1+N_5))$$

$$+\Delta N_5((M_1+M_3)(N_1+N_3)-(N_2+N_4+N_1N_3)-(M_1M_3+M_2+M_4))$$

$$
\begin{aligned}
\Delta k_5 = \Delta M_1 (& M_4 (N_2+N_4+N_1N_3+N_5(N_1+N_3))+N_2N_4+N_5(N_2N_3+N_1N_4) \\
& -M_3(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))) \\
+\Delta M_2 (& M_3(N_2+N_4+N_1N_3+N_5(N_1+N_3)) \\
& -M_4(N_1+N_3+N_5)-(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))) \\
+\Delta M_3 (& M_2(N_2+N_4+N_1N_3+N_5(N_1+N_3))+N_2N_4+N_5(N_2N_3+N_1N_4) \\
& -M_1(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))) \\
+\Delta M_4 (& M_1(N_2+N_4+N_1N_3+N_5(N_1+N_3))-M_2(N_1+N_3+N_5) \\
& -(N_2N_3+N_1N_4+N_5(N_2+N_4+N_1N_3))) \\
+\Delta N_1 (& (M_1M_4+M_2M_3)(N_3+N_5)+(M_1+M_3)N_5N_4 \\
& -M_2M_4-(M_1M_3+M_2+M_4)(N_4+N_5N_3)) \\
+\Delta N_2 (& (M_1M_4+M_2M_3)+(M_1+M_3)(N_4+N_5N_3) \\
& -(M_1M_3+M_2+M_4)(N_3+N_5)-N_4N_5) \\
+\Delta N_3 (& (M_1M_4+M_2M_3)(N_1+N_5)+(M_1+M_3)N_5N_2 \\
& -M_2M_4-(M_1M_3+M_2+M_4)(N_2+N_5N_1)) \\
+\Delta N_4 (& M_1M_4+M_2M_3+(M_1+m_3)(N_2+N_5N_1) \\
& -(M_1M_3+M_2+M_4)(N_1+N_5)-N_2N_5) \\
+\Delta N_5 (& (M_1M_4+M_2M_3)(N_1+N_3)+(M_1+M_3)(N_2N_3+N_1N_4) \\
& -M_2M_4-(M_1M_3+M_2+M_4)(N_2+N_4+N_1N_3)-N_2N_4)
\end{aligned}
$$

Similar equations can be obtained for all Δk_i where $i=2,4,6,7,8,9$.

In the early investigations, the equations were solved by means of an available ICL library program which used Gauss' method of pivotal condensation with partial pivotting, mentioned in section 2. Because various different accuracies were needed in the later investigations reported in section 5, the library program could not be used, but, as the method was satisfactory, a more general program, based upon it, was written.

The solutions of the equations are then used to give revised values for the M_i and N_i coefficients, and the loss is recalculated at the various ω values. The process is repeated until the filter specification is satisfied and when this occurs the coefficients a'_i and b'_i are calculated. For a ninth order filter five of them are fixed, say $b'_1, b'_2 \ldots b'_5$, and the values of the others ((n-1)/2 in number) a'_1, a'_2, a'_3 and a'_4 are chosen so that the denominator of $K'(p)$ will be the same as the denominator of $K(p)$ i.e. k'_2, k'_4, k'_6 and k'_8 are <u>forced</u> to be k_2, k_4, k_6 and k_8 respectively. Thus a'_1 to a'_4 are found from equations

$$\left.\begin{aligned}
k_2 &= a'_2 + b'_2 - a'_1 b'_1 \\
k_4 &= a'_4 + b'_4 + a'_2 b'_2 - a'_3 b'_1 - a'_1 b'_3 \\
k_6 &= a'_2 b'_4 + a'_4 b'_2 - a'_3 b'_3 - a'_1 b'_5 \\
k_8 &= a'_4 b'_4 - a'_3 b'_5
\end{aligned}\right\} \qquad (3.18)$$

In this way, the pair of compatible lattice impedances Z'_{ac}, Z'_{bc} for the particular choice of five fixed values and four variables out of the a_i, b_i coefficients are found which satisfy the filter specification for loss and have known resonance frequencies.

The open- and short-circuit impedances are found as mentioned in section 2 from the equations

$$\left.\begin{aligned} Z_o&=(Z_a+Z_b)/2 \\ Y_s&=(Y_a+Y_b)/2 \end{aligned}\right\} \qquad (3.19)$$

where Y=1/Z. Then the element values are realised by the method described in Appendix 3.

3.2 Symmetrical filters of any order.

This section extends the first part* of Saraga's method to any low-pass symmetrical filter of order n=2r+1, with r any positive integer. Proofs are given for some of the equations and are outlined for the remainder. The notation is explained and the generalised equations are given in tables I and II and then applied to a filter of order 9.

3.2.1 Derivation of generalised equations.

3.2.1.1. The k coefficients expressed in terms of the a and b coefficients.

The characteristic function K(p) can be written in the form

$$K(p) = \frac{k_1p+k_3p^3+\ldots\ldots+k_{2r+1}p^{2r+1}}{1+k_2p^2+\ldots\ldots+k_{2r}p^{2r}} = \frac{N}{D} \qquad (3.20)$$

---

* By first part is meant the discussion of relationships which in the case of n=9 are given by equations (3.12) to (3.15).

with N and D equal to the numerator and denominator polynomials respectively. The index r is restricted to an even value (a similar development can be traced if r is odd). The polynomial N+D is written as the product of two polynomials $B_a(p)$ and $B_b(-p)$, where $B_a(p)$ contains all the Hurwitz factors and $B_b(-p)$ contains all the anti-Hurwitz factors, i.e.

$$N+D=B_a(p)B_b(-p).$$

For a low pass filter of order 2r+1 it can always be assumed that r zeros of N+D lie on the left hand side of the p-plane and r+1 zeros on the right hand side, or vice versa. Therefore let

$$\left.\begin{aligned} B_a(p)&=1+a_1p+a_2p^2+\ldots\ldots+a_rp^r \\ &=E_a+O_a \\ \text{and}\qquad & \\ B_b(-p)&=1-b_1p+\ldots\ldots+(-1)^{r+1}b_{r+1}p^{r+1} \\ &=E_b-O_b \end{aligned}\right\} \qquad (3.21)$$

where E_a, E_b are the even parts and O_a, O_b the odd parts of $B_a(p)$, $B_b(p)$ respectively. The alternative form

$$B_a(p)=1+a_1p+\ldots\ldots+a_{r+1}p^{r+1}$$

$$B_b(-p)=1-b_1p+\ldots\ldots+(-1)^r b_rp^r$$

need not be discussed, as it would only lead to the interchange and inversion of the lattice impedances Z_a and Z_b defined by (3.22) below.

The equations (2.14) give the lattice impedances Z_a and Z_b

$$\left.\begin{aligned} Z_a &= \frac{O_a}{E_a} = \frac{a_1p+a_3p^3+\ldots\ldots+a_{r-1}p^{r-1}}{1+a_2p^2+\ldots\ldots\ldots+a_rp^r} \\ &\text{and} \\ Z_b &= \frac{E_b}{O_b} = \frac{1+b_2p^2+\ldots\ldots\ldots+b_rp^r}{b_1p+b_3p^3+\ldots\ldots+b_{r+1}p^{r+1}} \end{aligned}\right\} \tag{3.22}$$

The substitution of these lattice impedances into the equation

$$K(p) = \frac{Z_aZ_b-1}{Z_b-Z_a} \tag{3.23}$$

gives using equation (3.20)

$$\frac{k_1p+k_3p^3+\ldots+k_{2r+1}p^{2r+1}}{1+k_2p^2+\ldots\ldots+k_{2r}p^{2r}}$$

$$= \frac{\left\{(a_1p+a_3p^3+..+a_{r-1}p^{r-1})(1+b_2p^2+..+b_rp^r) - (1+a_2p^2+\ldots+a_rp^r)(b_1p+b_3p^3+\ldots+b_{r+1}p^{r+1})\right\}}{\left\{(1+a_2p^2+\ldots+a_rp^r)(1+b_2p^2+\ldots+b_rp^r) - (a_1p+a_3p^3+\ldots+a_{r-1}p^{r-1})(b_1p+b_3p^3+\ldots+b_{r+1}p^{r+1})\right\}}$$

Equating coefficients of powers of p in the numerator polynomials in the last two expressions leads to

$$\left.\begin{array}{ll} p & : k_1=a_1-b_1 \\ p^3 & : k_3=a_3+a_1b_2-a_2b_1-b_3 \\ p^5 & : k_5=a_5+a_3b_2+a_1b_4-b_5-a_2b_3-a_4b_1 \\ & \vdots \\ p^{2r+1} & : k_{2r+1}=-a_rb_{r+1} \end{array}\right\} \quad (3.24)$$

The general term can be considered as the coefficient of p^{2i+1} and can be written as

$$\left.\begin{array}{l} k_{2i+1}=a_{2i+1}+\sum_{j=1}^{i} a_{2i-2j+1}b_{2j}-b_{2i+1}-\sum_{j=1}^{i} a_{2j}b_{2i-2j+1} \\ \text{provided} \quad a_{r+1}=a_{r+2}=\ldots=0 \quad \text{i.e.} \quad a_t=0 \quad \text{for} \quad t\geq r+1 \\ \text{and} \quad b_{r+2}=b_{r+3}=\ldots 0 \quad \text{i.e.} \quad b_t=0 \quad \text{for} \quad t\geq r+2 \end{array}\right\} \quad (3.25)$$

Similarly equating denominators yields

$$\left.\begin{array}{lll} p^2 & : k_2=a_2+b_2-a_1b_1 & \\ p^4 & : k_4=a_4+a_2b_2+b_4-a_3b_1-a_1b_3 & \\ p^6 & : k_6=a_6+a_4b_2+a_2b_4+b_6-a_5b_1-a_3b_3-a_1b_5 & \\ & \vdots & \\ p^{2r-2} & : k_{2r-2}=a_rb_{r-2}+a_{r-2}b_r-a_{r-1}b_{r-1}-a_{r-3}b_{r+1} & \text{(assuming } r>2) \\ p^{2r} & : k_{2r}=a_rb_r-a_{r-1}b_{r+1} & \text{(assuming } r>1) \end{array}\right\} \quad (3.26)$$

For the general case;

$$\left.\begin{array}{l} p^{2i} \quad : k_{2i}=a_{2i}+\sum_{j=1}^{i-1} a_{2i-2j}b_{2j}+b_{2i}-\sum_{j=1}^{i} a_{2i-2j+1}b_{2j-1} \\ \\ \text{provided} \\ \\ \qquad a_t=0 \quad \text{for} \quad t\geqslant r+1 \quad \text{and} \quad b_t=0 \quad \text{for} \quad t\geqslant r+2. \end{array}\right\} \qquad (3.27)$$

Note that in order to obtain the equations (3.24) from equation (3.25) i takes the values $0,1,2,3,\ldots r-1,r$ whereas to obtain the equations (3.26) from equation (3.27) i takes the values $1,2,3,\ldots r-1,r$. Any summation with the upper limit smaller than the lower is taken as zero, e.g.

$$\sum_{j=1}^{0} a_{2i-2j+1}b_{2j}=0.$$

Such a case occurs for instance when $i=1$ in equation (3.27).

3.2.1.2. <u>The a coefficients expressed in terms of the M coefficients.</u>

The polynomial

$$B_a(p)=1+a_1p+\ldots+a_rp^r \qquad (3.28)$$

can be factorised into $\frac{1}{2}r$ quadratic factors i.e.

$$B_a(p)=\prod_{i=1}^{r/2}(1+M_{2i-1}p+M_{2i}p^2)$$

$$=(1+M_1p+M_2p^2)(1+M_3p+M_4p^2)\ldots(1+M_{r-1}p+M_rp^2) \qquad (3.29)$$

Equating coefficients of powers of p in (3.28) and (3.29) gives

$$p \quad : a_1=M_1+M_3+\ldots+M_{r-1}=\sum_{i=1}^{r/2}M_{2i-1}$$

$$p^2 \quad : a_2 = M_1M_3 + M_1M_5 + \ldots\ldots + M_1M_{r-1}$$

$$+M_3M_5 + \ldots\ldots + M_3M_{r-1}$$

$$+\ldots\ldots$$

$$\ldots + M_{r-3}M_{r-1}$$

$$+M_2 + M_4 + \ldots\ldots + M_r$$

$$= \tfrac{1}{2} \sum_{\substack{i=1 \\ i \neq j}}^{r/2} \sum_{j=1}^{r/2} M_{2i-1}M_{2j-1} + \sum_{i=1}^{r/2} M_{2i}$$

$$p^3 \quad : a_3 = \frac{1}{3!} \sum_{\substack{i=1 \\ i \neq j \neq k}}^{r/2} \sum_{j=1}^{r/2} \sum_{k=1}^{r/2} M_{2i-1}M_{2j-1}M_{2k-1} + \sum_{\substack{i=1 \\ i \neq j}}^{r/2} \sum_{j=1}^{r/2} M_{2i}M_{2j-1}$$

The notation $i \neq j \neq k$ is to be taken to mean that none of the counts can coincide in value in any term of the summation i.e. $i \neq k$ as well as $i \neq j$ and $j \neq k$. Similar meanings are attached to $i \neq j \neq k \neq \ell$ and $i \neq j \neq k \neq \ell \neq m$, etc.

$$p^4 \quad : a_4 = \frac{1}{4!} \sum_{i=1}^{r/2} \sum_{\substack{j=1 \\ i \neq j \neq k \neq \ell}}^{r/2} \sum_{k=1}^{r/2} \sum_{\ell=1}^{r/2} M_{2i-1}M_{2j-1}M_{2k-1}M_{2\ell-1}$$

$$+ \tfrac{1}{2} \sum_{i=1}^{r/2} \sum_{\substack{j=1 \\ i \neq j \neq k}}^{r/2} \sum_{k=1}^{r/2} M_{2i-1}M_{2j-1}M_{2k} + \tfrac{1}{2} \sum_{\substack{i=1 \\ i \neq j}}^{r/2} \sum_{j=1}^{r/2} M_{2i}M_{2j}$$

Before giving the two general terms $a_{2\alpha-1}$ and $a_{2\alpha}$, it is necessary to give examples to explain the notation for two formal expressions C and D that will be used. An expression of the following form will be considered:

C will be considered first

$$C = \underbrace{\sum_{i} \ldots\ldots\ldots \sum_{j}}_{\substack{\alpha-1 \text{ summation signs} \\ i \neq j}} \underbrace{M_{2i} \qquad M_{2j}}_{\alpha-1 \text{ terms}}$$

It is to be taken to mean: in the case of

$$\alpha=1 \quad : \quad C=0$$

$$\alpha=2 \quad : \quad C=\sum_{i} M_{2i}$$

$$\alpha=3 \quad : \quad C=\sum_{i}\sum_{j} M_{2i}M_{2j} \qquad i\neq j$$

$$\alpha=4 \quad : \quad C=\sum_{i}\sum_{j}\sum_{k} M_{2i}M_{2j}M_{2k} \qquad i\neq j\neq k$$

where i,j,k cannot take the same value in a term such as $M_{2i}M_{2j}M_{2k}$.

The expression D will now be considered:

$$D = \underbrace{\sum_{i}\cdots\cdots\sum_{j}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{k}\cdots\cdots\sum_{\ell}}_{2\beta-1 \text{ summation signs}} \underbrace{M_{2i}\cdots M_{2j}}_{\alpha-\beta \text{ terms}} \underbrace{M_{2k-1}\cdots M_{2\ell-1}}_{2\beta-1 \text{ terms}} \qquad i\neq j\neq k\neq i$$

where β will be used later as a running count in a summation.

It is to be taken to mean there are $\alpha-\beta$ summations, one for each of the $\alpha-\beta$ counts $i \ldots\ldots j$ of the even subscripted M coefficients, and $2\beta-1$ summations, one for each of the $2\beta-1$ counts $k\ldots\ldots\ell$ of the odd subscripted M coefficients. Two examples will be considered. For $\alpha=2$, $\beta=1$

$$D = \sum_{i}\sum_{k} M_{2i}M_{2k-1} \qquad i\neq k$$

For $\alpha=5$, $\beta=3$

$$D = \sum_{i}\sum_{j}\sum_{k}\sum_{\ell}\sum_{m}\sum_{n}\sum_{p} M_{2i}M_{2j}M_{2k-1}M_{2\ell-1}M_{2m-1}M_{2n-1}M_{2p-1} \qquad i\neq j\neq k\neq \ell\neq m\neq n\neq p$$

In this case no two of the count values i,j,k,ℓ,m,n,p can have the same values in the term $M_{2i}M_{2j}M_{2k-1}M_{2\ell-1}M_{2m-1}M_{2n-1}M_{2p-1}$ and there are two M coefficients with even subscripts and 5 with odd subscripts.

Then equating the coefficients of $p^{2\alpha-1}$ in equations (3.28) and (3.29) leads to:

$$a_{2\alpha-1} = \frac{1}{(\alpha-1)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-1 \text{ summation signs}} \sum_{k=1}^{r/2} \underbrace{M_{2i} \cdots\cdots\cdots M_{2j}}_{\alpha-1 \text{ terms}} M_{2k-1} + \cdots$$

$i \neq j \neq k$

$$\cdots + \frac{1}{(\alpha-\beta)!}\,\frac{1}{(2\beta-1)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{k=1}^{r/2} \cdots\cdots \sum_{\ell=1}^{r/2}}_{2\beta-1 \text{ summation signs}} \underbrace{M_{2i} \cdots\cdots\cdots M_{2j}}_{\alpha-\beta \text{ terms}} \underbrace{M_{2k-1} \cdots\cdots M_{2\ell-1}}_{2\beta-1 \text{ terms}} + \cdots$$

$i \neq j \neq k \neq \ell$

$$\cdots + \frac{1}{(2\alpha-1)!} \underbrace{\sum_{k=1}^{r/2} \cdots\cdots \sum_{\ell=1}^{r/2}}_{2\alpha-1 \text{ summation signs}} \underbrace{M_{2k-1} \cdots\cdots M_{2\ell-1}}_{2\alpha-1 \text{ terms}}$$

$k \neq \ell$

$$= \sum_{\beta=1}^{\alpha} \frac{1}{(\alpha-\beta)!}\,\frac{1}{(2\beta-1)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{k=1}^{r/2} \cdots\cdots \sum_{\ell=1}^{r/2}}_{2\beta-1 \text{ summation signs}} \underbrace{M_{2i} \cdots\cdots\cdots M_{2j}}_{\alpha-\beta \text{ terms}} \underbrace{M_{2k-1} \cdots\cdots M_{2\ell-1}}_{2\beta-1 \text{ terms}}$$

$i \neq j \neq k \neq \ell$

which is true for $\alpha=1,2,\ldots\frac{1}{2}r$ provided no two of the count values of $i\ldots j\ k\ldots\ell$ are the same in the term $M_{2i}\ldots M_{2j}\ M_{2k-1}\ldots M_{2\ell-1}$.

Let $q=\alpha-\beta$, $v=2\beta-1$ and change $i\to i_1$, $j\to i_q$, $k\to j_1$, $\ell\to j_v$ then

$$a_{2\alpha-1} = \sum_{\beta=1}^{\alpha} \frac{1}{q!\,v!} \sum_{i_1=1}^{c} \cdots\cdots \sum_{i_q=1}^{c} \sum_{j_1=1}^{d} \cdots\cdots \sum_{j_v=1}^{d} M_{2i_1} \cdots M_{2i_q} M_{2j_1-1} \cdots M_{2j_v-1}$$

$i_1 \neq i_q \neq j_1 \neq j_v$

when the limits of summation c and d equal $\frac{1}{2}r$.

For even powers of p the general term given by equating the coefficients of $p^{2\alpha}$ in equations (3.28) and (3.29) is:

$$a_{2\alpha} = \frac{1}{\alpha!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots\cdots \sum_{j=1}^{r/2}}_{\alpha \text{ summation signs}} \underbrace{M_{2i} \cdots\cdots M_{2j}}_{\alpha \text{ terms}} + \cdots$$

$i \neq j$

$$\cdots + \frac{1}{(\alpha-\beta)!\,(2\beta)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{k=1}^{r/2} \cdots\cdots \sum_{\ell=1}^{r/2}}_{2\beta \text{ summation signs}} \underbrace{M_{2i} \cdots M_{2j}}_{\alpha-\beta \text{ terms}} \underbrace{M_{2k-1} \cdots M_{2\ell-1}}_{2\beta \text{ terms}} + \cdots\cdots$$

$i \neq j \neq k \neq \ell$

$$\cdots + \frac{1}{(2\alpha)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots \sum_{j=1}^{r/2}}_{2\alpha \text{ summation signs}} \underbrace{M_{2i-1} \cdots\cdots M_{2j-1}}_{2\alpha \text{ terms}}$$

$i \neq j$

$$= \sum_{\beta=0}^{\alpha} \frac{1}{(\alpha-\beta)!\,(2\beta)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{k=1}^{r/2} \cdots\cdots\cdots \sum_{\ell=1}^{r/2}}_{2\beta \text{ summation signs}} \underbrace{M_{2i} \cdots\cdots M_{2j}}_{\alpha-\beta \text{ terms}} \underbrace{M_{2k-1} \cdots\cdots M_{2\ell-1}}_{2\beta \text{ terms}}$$

$$i \neq j \neq k \neq \ell$$

where $\alpha=1,2,\ldots \frac{1}{2}r$. Replacing i,j,k,ℓ by i_1, i_q, j_1, j_u ; $\alpha-\beta$ by q, 2β by u and the limits of the summation by c and d leads to the equation

$$a_{2\alpha} = \sum_{\beta=0}^{\alpha} \frac{1}{q!u!} \sum_{i_1=1}^{c} \cdots\cdots \sum_{i_q=1}^{c} \sum_{j_1=1}^{d} \cdots\cdots \sum_{j_u=1}^{d} M_{2i_1} \cdots\cdots M_{2i_q} M_{2j_1-1} \cdots\cdots M_{2j_u-1} \ .$$

$$i_1 \neq i_q \neq j_1 \neq j_u$$

3.2.1.3. The b coefficients expressed in terms of the N coefficients.

The polynomial

$$B_b(p) = 1+b_1p+b_2p^2 + \ldots + b_{r+1}p^{r+1}$$

can be factorised into $\frac{1}{2}r$ quadratic factors and 1 linear factor (as stated at the beginning of this section r has been limited to even values). Thus

$$B_b(p) = (1+N_1p+N_2p^2)(1+N_3p+N_4p^2) \ \ldots\ldots (1+N_{r-1}p+N_rp^2)(1+N_{r+1}p).$$

When similar techniques to those used to relate the a and M coefficients are applied, the following equations, valid for $\alpha=1,2,\ldots \frac{1}{2}r+1$ are obtained:

$$b_{2\alpha-1} = \frac{1}{(\alpha-1)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-1 \text{ summation signs}} \sum_{k=1}^{\frac{1}{2}r+1} \underbrace{N_{2i} \cdots\cdots N_{2j}}_{\alpha-1 \text{ terms}} \underbrace{N_{2k-1}}_{1 \text{ term}} + \cdots\cdots$$

$$i \neq j \neq k$$

$$\cdots\cdots + \frac{1}{(\alpha-\beta)!\,(2\beta-1)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{k=1}^{\frac{1}{2}r+1} \cdots\cdots\cdots \sum_{l=1}^{\frac{1}{2}r+1}}_{2\beta-1 \text{ summation signs}} \underbrace{N_{2i} \cdots\cdots N_{2j}}_{\alpha-\beta \text{ terms}} \underbrace{N_{2k-1} \cdots\cdots N_{2\ell-1}}_{2\beta-1 \text{ terms}} + \cdots\cdots$$

$$i \neq j \neq k \neq \ell$$

$$\cdots\cdots + \frac{1}{(2\alpha-1)!} \underbrace{\sum_{i=1}^{\frac{1}{2}r+1} \cdots\cdots\cdots \sum_{j=1}^{\frac{1}{2}r+1}}_{2\alpha-1 \text{ summation signs}} \underbrace{N_{2i-1} \cdots\cdots N_{2j-1}}_{2\alpha-1 \text{ terms}}$$

$$i \neq j$$

$$= \sum_{\beta=1}^{\alpha} \frac{1}{(\alpha-\beta)!(2\beta-1)!} \underbrace{\sum_{i=1}^{r/2} \cdots\cdots \sum_{j=1}^{r/2}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{k=1}^{\frac{1}{2}r+1} \cdots\cdots \sum_{\ell=1}^{\frac{1}{2}r+1}}_{2\beta-1 \text{ summation signs}} \underbrace{N_{2i} \cdots\cdots N_{2j}}_{\alpha-\beta \text{ terms}} \underbrace{N_{2k-1} \cdots\cdots N_{2\ell-1}}_{2\beta-1 \text{ terms}} .$$

$$i \neq j \neq k \neq \ell$$

The same substitutions will be used as before but in the present case the limits of the summations will be $c=\frac{1}{2}r$ and $d=\frac{1}{2}r+1$. This leads to

$$b_{2\alpha-1} = \sum_{\beta=1}^{\alpha} \frac{1}{q!v!} \sum_{i_1=1}^{c} \cdots\cdots \sum_{i_q=1}^{c} \sum_{j_1=1}^{d} \cdots\cdots \sum_{j_v=1}^{d} N_{2i_1} \cdots\cdots N_{2i_q} N_{2j_1-1} \cdots\cdots N_{2j_v-1} .$$

$$i_1 \neq i_q \neq j_1 \neq j_v$$

In a similar way

$$b_{2\alpha} = \sum_{\beta=0}^{\alpha} \frac{1}{q!u!} \sum_{i_1=1}^{c} \cdots\cdots \sum_{i_q=1}^{c} \sum_{j_1=1}^{d} \cdots\cdots \sum_{j_u=1}^{d} N_{2i_1} \cdots\cdots N_{2i_q} N_{2j_1-1} \cdots\cdots N_{2j_u-1} .$$

3.2.2 Tabulation and application of the generalised equations.

The generalised equations discussed in the previous section are summarised in the following two tables.

Note I

Summations $\sum_{a}^{b}$ with $b<a$ are to be treated as non-existent.

Note II

For the equations in table 3 relating the coefficients a to M and the coefficients b to N, none of the subscripts $i_1, i_2, \ldots i_{\alpha-\beta}, j_1, j_2, \ldots j_{2\beta-1}, j_{2\beta}$ is allowed to assume an identical value with another subscript in the same multiplicative term. Thus

$$\sum_{i_1=1}^{2} \sum_{i_2=1}^{2} \sum_{j_1=1}^{3} M_{2i_1} M_{2i_2} M_{2j_1-1} = M_2M_4M_5+M_4M_2M_5=2M_2M_4M_5$$

since terms like $M_2M_4M_3$ are not permitted because then $i_2=j_1=2$.

Table 2 Relationship between characteristic function $K(p)$ and lattice impedances Z_a and Z_b .

<table>
<tr><td colspan="2">

$$K(p) = \frac{k_1p+k_3p^3+ \ldots\ldots +k_{2r+1}p^{2r+1}}{1+k_2p^2+ \ldots\ldots +k_{2r}p^{2r}}$$

$$k_{2i+1}=a_{2i+1}-b_{2i+1}+\sum_{j=1}^{i} a_{2i-2j+1}b_{2j}-\sum_{j=1}^{i} a_{2j}b_{2i-2j+1} \qquad i=0,1,2,\ldots r \qquad (1)$$

$$k_{2i} = a_{2i}+b_{2i}+\sum_{j=1}^{i-1} a_{2i-2j}b_{2j}-\sum_{j=1}^{i} a_{2i-2j+1}b_{2j-1} \qquad i=1,2,\ldots r \qquad (2)$$

</td></tr>
<tr><td colspan="2">

$a_t=0$ for $t > r+1$; $b_t=0$ for $t > r+2$

</td></tr>
<tr><td>case I : r even</td><td>case II : r odd</td></tr>
<tr><td>

$$Z_a = \frac{a_1p+a_3p^3+ \ldots +a_{r-1}p^{r-1}}{1+a_2p^2+ \ldots\ldots +a_rp^r}$$

$$Z_b = \frac{1+b_2p^2+ \ldots\ldots +b_rp^r}{b_1p+b_3p^3+ \ldots +b_{r+1}p^{r+1}}$$

</td><td>

$$Z_a = \frac{a_1p+a_3p^3+ \ldots +a_rp^r}{1+a_2p^2+ \ldots +a_{r-1}p^{r-1}}$$

$$Z_b = \frac{1+b_2p^2+ \ldots +b_{r+1}p^{r+1}}{b_1p+b_3p^3+ \ldots +b_rp^r}$$

</td></tr>
</table>

Table 3 Relationship between the coefficients M and a, N and b.

The table should be read with notes I and II.

case I : r even	case II : r odd
$1+\sum_{j=1}^{r} a_j p^j = \prod_{i=1}^{r/2} (1+M_{2i-1}p+M_{2i}p^2)$	$1+\sum_{j=1}^{r} a_j p^j = (1+M_r p) \prod_{i=1}^{\frac{1}{2}(r-1)} (1+M_{2i-1}p+M_{2i}p^2)$
$1+\sum_{j=1}^{r+1} b_j p^j = (1+N_{r+1}p) \prod_{i=1}^{r/2} (1+N_{2i-1}p+N_{2i}p^2)$	$1+\sum_{j=1}^{r+1} b_j p^j = \prod_{i=1}^{\frac{1}{2}(r+1)} (1+N_{2i-1}p+N_{2i}p^2)$

For $\alpha=1,2,\ldots e$ with $q=\alpha-\beta$; $u=2\beta$; $v=2\beta-1$:

$$(1)\quad a_{2\alpha-1}=\sum_{\beta=1}^{\alpha}\frac{1}{q!v!}\sum_{i_1=1}^{c}\sum_{i_2=1}^{c}\cdots\sum_{i_q=1}^{c}\sum_{j_1=1}^{d}\sum_{j_2=1}^{d}\cdots\sum_{j_v=1}^{d} M_{2i_1}M_{2i_2}\cdots M_{2i_q}M_{2j_1-1}M_{2j_2-1}\cdots M_{2j_v-1}$$

$$(2)\quad a_{2\alpha}=\sum_{\beta=0}^{\alpha}\frac{1}{q!u!}\sum_{i_1=1}^{c}\sum_{i_2=1}^{c}\cdots\sum_{i_q=1}^{c}\sum_{j_1=1}^{d}\sum_{j_2=1}^{d}\cdots\sum_{j_u=1}^{d} M_{2i_1}M_{2i_2}\cdots M_{2i_q}M_{2j_1-1}M_{2j_2-1}\cdots M_{2j_u-1}$$

$$(3)\quad b_{2\alpha-1}=\sum_{\beta=1}^{\alpha}\frac{1}{q!v!}\sum_{i_1=1}^{c}\sum_{i_2=1}^{c}\cdots\sum_{i_q=1}^{c}\sum_{j_1=1}^{d}\sum_{j_2=1}^{d}\cdots\sum_{j_v=1}^{d} N_{2i_1}N_{2i_2}\cdots N_{2i_q}N_{2j_1-1}N_{2j_2-1}\cdots N_{2j_v-1}$$

$$(4)\quad b_{2\alpha}=\sum_{\beta=0}^{\alpha}\frac{1}{q!u!}\sum_{i_1=1}^{c}\sum_{i_2=1}^{c}\cdots\sum_{i_q=1}^{c}\sum_{j_1=1}^{d}\sum_{j_2=1}^{d}\cdots\sum_{j_u=1}^{d} N_{2i_1}N_{2i_2}\cdots N_{2i_q}N_{2j_1-1}N_{2j_2-1}\cdots N_{2j_u-1}$$

	case I : r even			case II : r odd		
	c	d	e	c	d	e
$a_{2\alpha-1}$	$\frac{1}{2}r$	$\frac{1}{2}r$	$\frac{1}{2}r$	$\frac{1}{2}(r-1)$	$\frac{1}{2}(r+1)$	$\frac{1}{2}(r+1)$
$a_{2\alpha}$	$\frac{1}{2}r$	$\frac{1}{2}r$	$\frac{1}{2}r$	$\frac{1}{2}(r-1)$	$\frac{1}{2}(r+1)$	$\frac{1}{2}(r-1)$
$b_{2\alpha-1}$	$\frac{1}{2}r$	$(\frac{1}{2}r)+1$	$(\frac{1}{2}r)+1$	$\frac{1}{2}(r+1)$	$\frac{1}{2}(r+1)$	$\frac{1}{2}(r+1)$
$b_{2\alpha}$	$\frac{1}{2}r$	$(\frac{1}{2}r)+1$	$\frac{1}{2}r$	$\frac{1}{2}(r+1)$	$\frac{1}{2}(r+1)$	$\frac{1}{2}(r+1)$

It may be noticed that equations (3) and (4) in the above table can be obtained from (1) and (2) respectively by replacing coefficients a by b and coefficients M by N, and taking the appropriate values for c, d and e.

To show the way in which the tables can be used in specific cases a filter of order $n=9$, i.e. $r=4$, will be considered. From table 2, $a_t=0$ for $t\geq 5$ and $b_t=0$ for $t\geq 6$. Substituting $i=0$ into (1) gives

$$k_1=a_1-b_1+\sum_{j=1}^{0} a_{1-2j}b_{2j}-\sum_{j=1}^{0} a_{2j}b_{1-2j}=a_1-b_1$$

and $i=2$, yields

$$k_5=a_5-b_5+\sum_{j=1}^{2} a_{5-2j}b_{2j}-\sum_{j=1}^{2} a_{2j}b_{5-2j}$$

$$=-b_5+a_3b_2+a_1b_4-a_2b_3-a_4b_1 \quad \text{since } a_5=0 \;.$$

When $i=3$ is substituted into equation (2) of table 2,

$$k_6=a_6+b_6+\sum_{j=1}^{2} a_{6-2j}b_{2j}-\sum_{j=1}^{3} a_{7-2j}b_{2j-1}$$

$$=a_4b_2+a_2b_4-a_3b_3-a_1b_5 \;.$$

Clearly these equations agree with the corresponding equations in (3.12) and (3.13).

From table 2

$$Z_a = \frac{a_1p+a_3p^3}{1+a_2p^2+a_4p^4}$$

$$Z_b = \frac{1+b_2p^2+b_4p^4}{b_1p+b_3p^3+b_5p^5}$$

and from table 3, equation (1),

$$a_{2\alpha-1} = \sum_{\beta=1}^{\alpha} \frac{1}{q!v!} \sum_{i_1=1}^{2} \cdot\cdot \sum_{i_q=1}^{2} \sum_{j_1=1}^{2} \cdot\cdot \sum_{j_v=1}^{2} M_{2i_1} \cdot\cdot M_{2i_q} M_{2j_1-1} \cdot\cdot M_{2j_v-1}$$

where $q=\alpha-\beta,\ \ u=2\beta,\ \ v=2\beta-1$.

With $\alpha=1$,

$$a_1 = \sum_{\beta=1}^{1} \frac{1}{(1-\beta)!(2\beta-1)!} \underbrace{\sum_{i_1=1}^{2} \cdots\cdots\cdots \sum_{i_q=1}^{2}}_{1-\beta \text{ summation signs}} \underbrace{\sum_{j_1=1}^{2} \cdots\cdots\cdots \sum_{j_v=1}^{2}}_{2\beta-1 \text{ summation signs}} \underbrace{M_{2i_1} \cdots\cdots M_{2i_q}}_{1-\beta \text{ terms}} \underbrace{M_{2j_1-1} \cdots\cdots M_{2j_v-1}}_{2\beta-1 \text{ terms}}$$

$$a_1 = \frac{1}{0!1!} \sum_{j_1=1}^{2} M_{2j_1-1} \ .$$

Since $0!=1$,

$$a_1 = M_1+M_3 \ .$$

With $\alpha=2$,

$$a_3 = \sum_{\beta=1}^{2} \frac{1}{(2-\beta)!(2\beta-1)!} \sum_{i_1=1}^{2} \cdot\cdot \sum_{i_{2-\beta}=1}^{2} \sum_{j_1=1}^{2} \cdot\cdot \sum_{j_{2\beta-1}=1}^{2} M_{2i_1} \cdot\cdot M_{2i_{2-\beta}} M_{2j_1-1} \cdot\cdot M_{2j_{2\beta-1}-1}$$

$$= \frac{1}{1!1!} \sum_{i_1=1}^{2} \sum_{j_1=1}^{2} M_{2i_1} M_{2j_1-1} + \frac{1}{0!3!} \sum_{j_1=1}^{2} \sum_{j_2=1}^{2} \sum_{j_3=1}^{2} M_{2j_1-1} M_{2j_2-1} M_{2j_3-1} \ .$$

The second term is non-existent because three different values of the count are needed and there are only two

$$a_3 = M_2M_3+M_4M_1 \ .$$

In equation (2) of table 3, α takes the values 1 and 2. As an example the expression for $a_{2\alpha}$ with $\alpha=1$ will be given.

$$a_2 = \sum_{\beta=0}^{1} \frac{1}{(1-\beta)!(2\beta)!} \underbrace{\sum_{i_1=1}^{2} \cdots\cdots\cdots \sum_{i_q=1}^{2}}_{1-\beta \text{ summation signs}} \underbrace{\sum_{j_1=1}^{2} \cdots\cdots\cdots \sum_{j_u=1}^{2}}_{2\beta \text{ summation signs}} \underbrace{M_{2i_1} \cdots\cdots M_{2i_q}}_{1-\beta \text{ terms}} \underbrace{M_{2j_1-1} \cdots\cdots M_{2j_u-1}}_{2\beta \text{ terms}}$$

$$= \frac{1}{1!0!} \sum_{i_1=1}^{2} M_{2i_1} + \frac{1}{0!2!} \sum_{j_1=1}^{2} \sum_{j_2=1}^{2} M_{2j_1-1} M_{2j_2-1}$$

$$= M_2+M_4+ \frac{1}{2} (M_1M_3+M_3M_1)$$

$$= M_2+M_4+M_1M_3 \ .$$

These equations agree with (3.9). Because the calculation with $\alpha=2$ has no unexpected features, it is not given here. However it might be of interest to apply the formula (3) for $b_{2\alpha-1}$ of table 3. Taking the appropriate e, c, d values from the table gives $\alpha=1,\ 2,\ 3$ and

$$b_{2\alpha-1} = \sum_{\beta=1}^{\alpha} \frac{1}{(\alpha-\beta)!\,(2\beta-1)!} \underbrace{\sum_{i_1=1}^{2} \cdots\cdots\cdots \sum_{i_q=1}^{2}}_{\alpha-\beta \text{ summation signs}} \underbrace{\sum_{j_1=1}^{3} \cdots\cdots\cdots \sum_{j_v=1}^{3}}_{2\beta-1 \text{ summation signs}} \underbrace{N_{2i_1} \cdots\cdots\cdots N_{2i_q}}_{\alpha-\beta \text{ terms}} \underbrace{N_{2j_1-1} \cdots\cdots\cdots N_{2j_v-1}}_{2\beta-1 \text{ terms}}$$

When $\alpha=1$,

$$b_1 = \frac{1}{0!1!} \sum_{j_1=1}^{3} N_{2j_1-1}$$

$$= N_1+N_3+N_5$$

This agrees with equation (3.10).

For $\alpha=3$,

$$b_5 = \sum_{\beta=1}^{3} \frac{1}{(3-\beta)!\,(2\beta-1)!} \sum_{i_1=1}^{2} \cdot\cdot \sum_{i_q=1}^{2} \sum_{j_1=1}^{3} \cdot\cdot \sum_{j_v=1}^{3} N_{2i_1} \cdot\cdot N_{2i_q} N_{2j_1-1} \cdot\cdot N_{2j_v-1}$$

$$= \frac{1}{2!1!} \sum_{i_1=1}^{2} \sum_{i_2=1}^{2} \sum_{j_1=1}^{3} N_{2i_1} N_{2i_2} N_{2j_1-1} + \frac{1}{1!3!} \sum_{i_1=1}^{2} \sum_{j_1=1}^{3} \sum_{j_2=1}^{3} \sum_{j_3=1}^{3} N_{2i_1} N_{2j_1-1} N_{2j_2-1} N_{2j_3-1}$$

$$+ \frac{1}{0!5!} \underbrace{\sum_{j_1=1}^{3} \cdots\cdots\cdots \sum_{j_5=1}^{3}}_{5 \text{ summation signs}} N_{2j_1-1} \cdots N_{2j_5-1}$$

No repetitions of the count values are allowed, i.e. the second and third terms cannot exist because in one case four different values of a count are required but there are only three values at most and in the other case five different are required from the three available. Therefore only the first term in the last equation exists which when expanded leads to

$$b_5 = \frac{1}{2} (N_2N_4N_5+N_4N_2N_5) \quad = \quad N_2N_4N_5 \; .$$

This agrees with equation (3.10). The expressions for a_4 and b_i, $i=2,3,4$ can be found in a similar way.

4. Initial design application of Saraga's method.

Whereas all algebraic expressions given in this thesis (and also in ref.7) refer to symmetrical low pass filters in general, irrespective of the shape of the loss frequency curve, the practical design examples studied in this thesis refer to Cauer type low pass filters, i.e. to low-pass filters with equi-ripple behaviour in the pass band and equal loss minima in the stop band.

4.1. Previous investigations.

Saraga's method had already been investigated in different ways for a symmetrical filter of order 7(ref.7)*. Such investigations had been carried out, mostly by this researcher, before the work reported in this thesis was started. In the investigations to be described in section 4.2, the starting point is a known function $K(p) = \frac{N(p)}{D(p)}$. Now in the conventional design process the positions of the zeros of $D(p)+N(p)$ are not given but have to be found from $K(p)$, where $N(p)$ and $D(p)$ are considered as "accurate". In the process of finding the zeros of $D(p)+N(p)$ the zeros would normally be obtained inaccurately (as discussed in detail in section 2) unless very special measures (such as double or triple length arithmetic and a great number of iterations) are taken. As explained before, in Saraga's method it

---

* The notation in that reference differs from that of section 3.2 of this thesis. For n=7 in ref.7,

$$N+D=(1+M_1p)(1+M_2p+M_3p^2)(1-N_1p+N_2p^2)(1-N_3p+N_4p^2)$$

whereas in this thesis

$$N+D=(1+M_1p+M_2p^2)(1+M_3p)(1-N_1p+N_2p^2)(1-N_3p+N_4p^2)$$

i.e. the first two factors have been interchanged. This has been done because for the general algebra it is more convenient to write the linear factor after the corresponding quadratic factors.

is possible to work with fairly inaccurate zeros of D+N and to carry out any improvement that is necessary not on these zeros but on the lattice impedences Z_a and Z_b, at a later stage of the design procedure. Therefore, in an investigation of Saraga's method "inaccurate zeros" of D+N are required as a starting point. It was thought that, rather than actually finding such "inaccurate zeros" for each investigation which would also imply that the inaccuracy had some random character, it would be more convenient and more useful to introduce defined inaccuracies. The first approach adopted was to start with "accurate" zeros. These were obtained by double or triple-length arithmetic including a large number of iterations (later, during the work for this thesis they were obtained from ref.11). Then these zeros were artificially deteriorated by multiplying the real part and the imaginary part of each zero, by a factor η. At the beginning of the investigation $\eta=1.1$ was used.

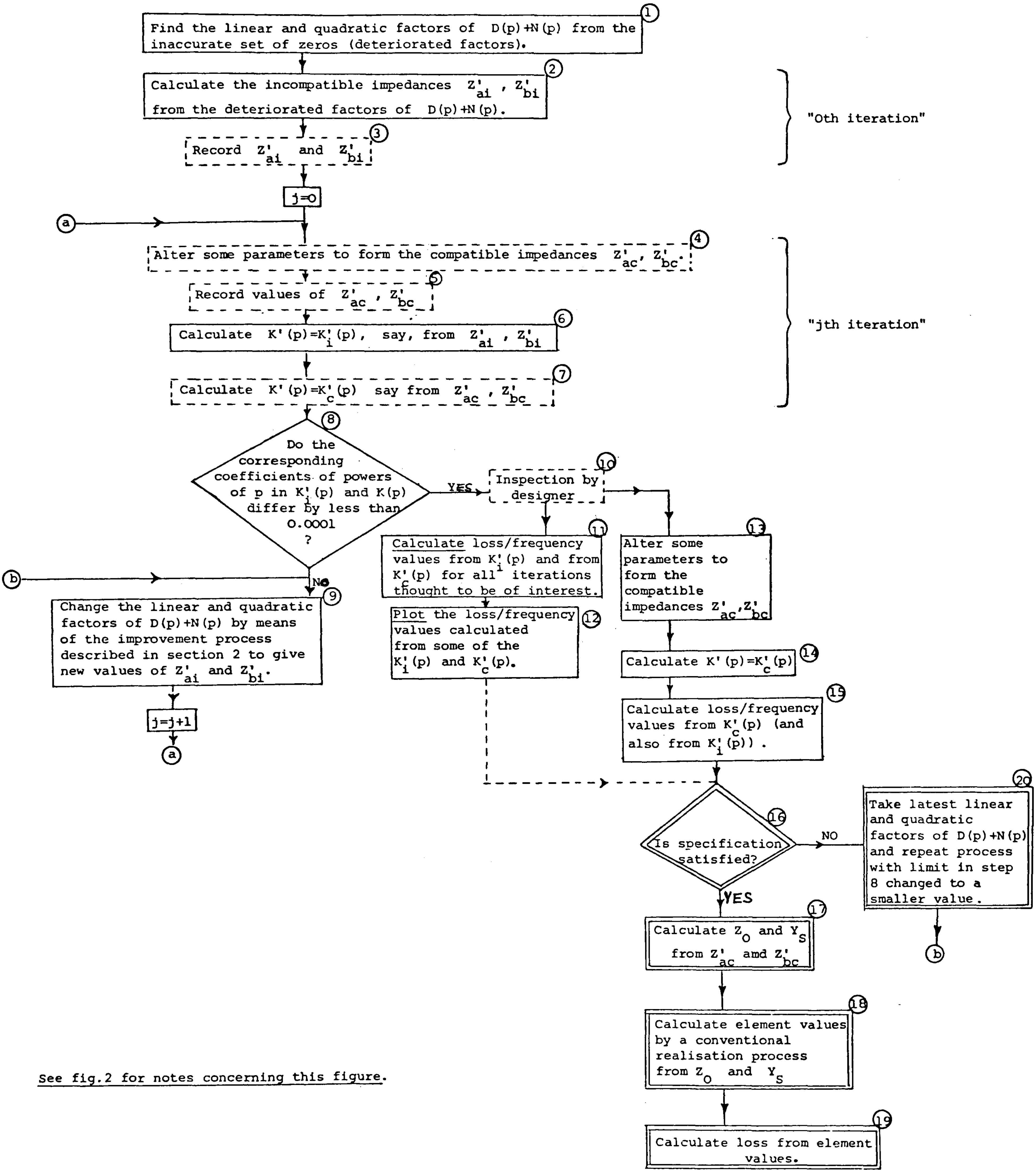
Therefore the investigation described in ref.7 started with an accurately known function $K(p) = \frac{N(p)}{D(p)}$ and a set of zeros of N+D with exactly defined inaccuracies.

Four of the ways in which Saraga's method can be studied are shown in fig.1 and defined as "options" 1 to 4 in fig.2. Also the techniques used in the steps in fig.1 are indicated in fig.2. Options 1 and 2 can be used to investigate the method i.e. the performance (loss/frequency values) is calculated after some of the parameters have been changed. Options 3 and 4 can be used to

Fig.1. Steps in the investigation of Saraga's method.

Given $K(p) = \frac{N(p)}{D(p)}$

and a set of zeros of $D(p)+N(p)$ with exactly defined inaccuracies.



See fig.2 for notes concerning this figure.

Fig.2 Notes for fig.1

Note 1 Symbols such as $\downarrow$ⓐ and ⓐ$\rightarrow$ mean connect $\downarrow$ⓐ to ⓐ$\rightarrow$.

Note 2 Double length arithmetic instead of single length arithmetic or lower accuracy, was used in step 18 only i.e. for the conventional realisation method which was used to find the element values.

Note 3 Different forms of Saraga's method are defined in the following way:

Option 1 The steps in blocks 1 to 12.

Option 2 The steps in blocks 1 to 9 and 13 to 15, but excluding 3,4,5,7,10 the blocks with broken line borders.

Option 3 As Option 1 but with steps 16 to 20 in addition which are enclosed in blocks with double borders.

Option 4 As Option 2 but with steps 16 to 20 in addition.

Note 4 In the following table those equation numbers and section numbers which refer to the detailed description of the method for a filter of order n=9 are enclosed in square brackets [ ] whereas those which refer to the general description are enclosed in "curly" brackets { }.

Block number(s)	(Equations and sections used).
2	{(2.14),(2.15)} ; [(3.2)-(3.11)].
4, 13	{Last 2 paragraphs of section 2}; [find a'_1 to a'_4 from (3.18)].
6	{(2.20)}
7, 14	{(2.20)}
9	{Near end of section 2}; [section 3.1 after equation (3.16)].
11, 15	{(2.19)}
17	{(2.17),(2.18)}; [3.19] .

design a filter i.e. the element values are found (as well as the performance). Options 1 and 3 are similar because when they are used, many calculations are carried out and many results are obtained, and this means that the method can be studied in detail. On the other hand options 2 and 4 are more efficient because when they are used only the necessary calculations are performed.

As mentioned in blocks 4 and 13 of fig.1, some of the parameters are altered to give a pair of impedances Z'_{ac}, Z'_{bc} which will produce the required poles in the loss function, i.e. Z'_{ac}, Z'_{bc} are forced to be "compatible" with the frequencies of the poles of the original $K(p)$ function. Referring now to block 8, the value 0.0001 was chosen for the test so that a great deal would be learnt when option 1 was used: all the functions $K'_c(p)$ that could be of interest are calculated and by the time the test is satisfied the differences in the loss/frequency values calculated from the original $K(p)$ and from $K'_i(p)$ and $K'_c(p)$ are negligible. However when option 2 is used, another larger value such as 0.01 might be more useful because it is only required that the loss calculated from $K'(p)$ should satisfy the specification (involving lower and/or upper limits) it need not be near to the loss calculated from the original $K(p)$.

The count j (set after block 3 and incremented after block 9) is used to indicate the number of times block 9 has been used to improve the incompatible impedances Z'_{ai} and Z'_{bi}, i.e. the number of iterations performed to produce the particular Z'_{ai}, Z'_{bi}, the associated Z'_{ac}, Z'_{bc} calculated in block 4 and the corresponding K'_i, K'_c calculated in blocks 6 and 7.

As indicated in blocks 11 and 12 of fig.1, some sets of loss/ frequency values calculated from the $K_i'(p)$ and $K_c'(p)$ (of later iterations) were not plotted. This was because such loss/ frequency values were almost indistinguishable from the "original" loss/frequency values calculated from $K(p)$ on a scale in the pass band of 0.02 dB/cm and in the stopband of 5 dB/cm against a normalised frequency scale of 0.1/cm (the normalisation being such that the passband ends at unity).

Higher precision arithmetic is used for the realisation of the element values by a conventional realisation process (block 18 of fig.1) as mentioned in note 2 of fig.2. This ensures that the differences between the results calculated from the "compatible" and "incompatible" values are meaningful and not attributable to rounding errors.

When the compatible impedances Z'_{ac} and Z'_{bc} were used, block 15 options 2 and 4 of Saraga's method gave identical results with those from block 19 of option 4 for the loss/frequency values for a given iteration (to recall: in block 15 the loss is computed from $K'_c(p)$ and in block 19 from the element values). In view of this agreement it was tentatively inferred that the loss could be predicted using option 2 by itself and this option has the advantage of being far shorter than option 4. In fact instead of option 2, option 1 was used because it gives much more information which was useful for the study as a whole but for practical purposes it is often justified to consider the comparison as if it were between options 2 and 4. Therefore it was regarded as appropriate strategy to investigate Saraga's method by option 1 alone in the expectation that if the loss calculated from $K'_c(p)$ (i.e. from Z'_{ac} and Z'_{bc}) satisfied the design requirements, the element values of a filter with the same performance could be obtained using option 4 of Saraga's method.

It is of interest to find that when the incompatible impedances were used blocks 15 and 19 gave different results (i.e. different loss/frequency values); therefore the performance of the final filter could not be assessed by inspection of the results of option 2. The reason that the results from block 19 differ from those of block 15 (in the case of incompatible impedances) is that, as mentioned earlier, it is required that $Z'_a = Z'_b$ at the frequencies of the attenuation poles (i.e. where $K(p) \to \infty$); this relationship is not satisfied for the incompatible impedances Z'_{ai} and Z'_{bi} at the nominal pole frequencies of $K(p)$ which were used for the realisation.*

4.2. Present investigations.

4.2.1. n=7

Before the work for this thesis began, investigations had been carried out (as mentioned in Section 4.1) using option 1 fig.1 of Saraga's method. In this case all the zeros of $D(p)+N(p)$ i.e. the zeros of $B_a(p)$ and of $B_b(-p)$ (see equation (1.6)) had been increased by a factor of 1.1. The work for this thesis started with further investigations using option 1 but with the real and imaginary parts of these zeros altered in various ways by factors of 1.1 and 0.9 The zeros of $B_a(p)$ are $-\alpha_1+j\beta_1$, $-\alpha_2$ and those of $B_b(p)$ are $-\alpha_3 \pm j\beta_3$, $-\alpha_4 \pm j\beta_4$** and the inaccurate zeros, i.e. the zeros of $B'_a(p)$, $B'_b(p)$ (which are taken as initial data for the cases studied using option 1 fig.1 of Saraga's method) are given in Table 4 in terms of the α_i and β_i. The attenuation poles are Ω_1, Ω_2 and Ω_3.

---

* Alternatively, if agreement between the loss frequency values obtained from blocks 15 and 19 were required, an equation in p^2 obtained from $Z'_{ai} = Z'_{bi}$ could be solved (of comparatively low order, namely, $\frac{1}{2}(n-1)$). The p-values thus obtained could then be used in the conventional realisation process as the frequencies of the loss poles, provided they all were purely imaginary. The loss/frequency values found by analysing the element values obtained in this way should then agree with the results of block 15 but would not agree with those of block 19 .

** Note that these are the zeros of $B_b(p)$ and not of $B_b(-p)$.

Table 4. Filter type C0715, θ=56.*

$\Omega_1 = 1.206218, \quad \Omega_2 = 1.402707, \quad \Omega_3 = 2.248546.$

Case \ Iteration number →	$B'_a(p)$	$B'_b(p)$	Fig. numbers for plots of loss/frequency values: 0	1	2	3
(i)	$-0.9\alpha_1 \pm j0.9\beta_1$ $-0.9\alpha_2$	$-0.9\alpha_3 \pm j0.9\beta_3$ $-0.9\alpha_4 \pm j0.9\beta_4$**	-	3	4	-
(ii)	$-1.1\alpha_1 \pm j1.1\beta_1$ $-1.1\alpha_2$	$-0.9\alpha_3 \pm j0.9\beta_3$ $-0.9\alpha_4 \pm j0.9\beta_4$**	5	6	7	8
(iii)	$-1.1\alpha_1 \pm j0.9\beta_1$ $-1.1\alpha_2$	$-1.1\alpha_3 \pm j0.9\beta_3$ $-1.1\alpha_4 \pm j0.9\beta_4$**	9	10	11	12
(iv)	$-1.1\alpha_1 \pm j0.9\beta_1$ $-1.1\alpha_2$	$-0.9\alpha_3 \pm j1.1\beta_3$ $-0.9\alpha_4 \pm j1.1\beta_4$	13	14	15	-
(v)	$-1.1\alpha_1 \pm j1.1\beta_1$ $-1.1\alpha_2$	$-1.1\alpha_3 \pm j1.1\beta_3$ $-0.9\alpha_4 \pm j0.9\beta_4$**	16	17	18	19

$-\alpha_1 \pm j\beta_1 = -\ .1631560004 \pm j.9274774275$	$-\alpha_3 \pm j\beta_3 = -\ .04176841557 \pm j1.0249489572$
$-\alpha_2 = -.5179729584$	$-\alpha_4 \pm j\beta_4 = -\ .3717975189 \pm j.624350288$

---

* In this table and table 5 the type number is that used by Saal (ref.10).

** The value for $0.9\beta_4$ was incorrectly used as 0.56195259 instead of 0.5619 <u>1</u> 5259 but as the error was less than 0.01% it was decided not to repeat the calculations.

The loss/frequency values calculated from the accurate K(p) functions are called "ideal" and shown in figs. 3 and 4. The loss/frequency curves for the incompatible and compatible parameters are shown in figs.3 to 19 as indicated in table 4.

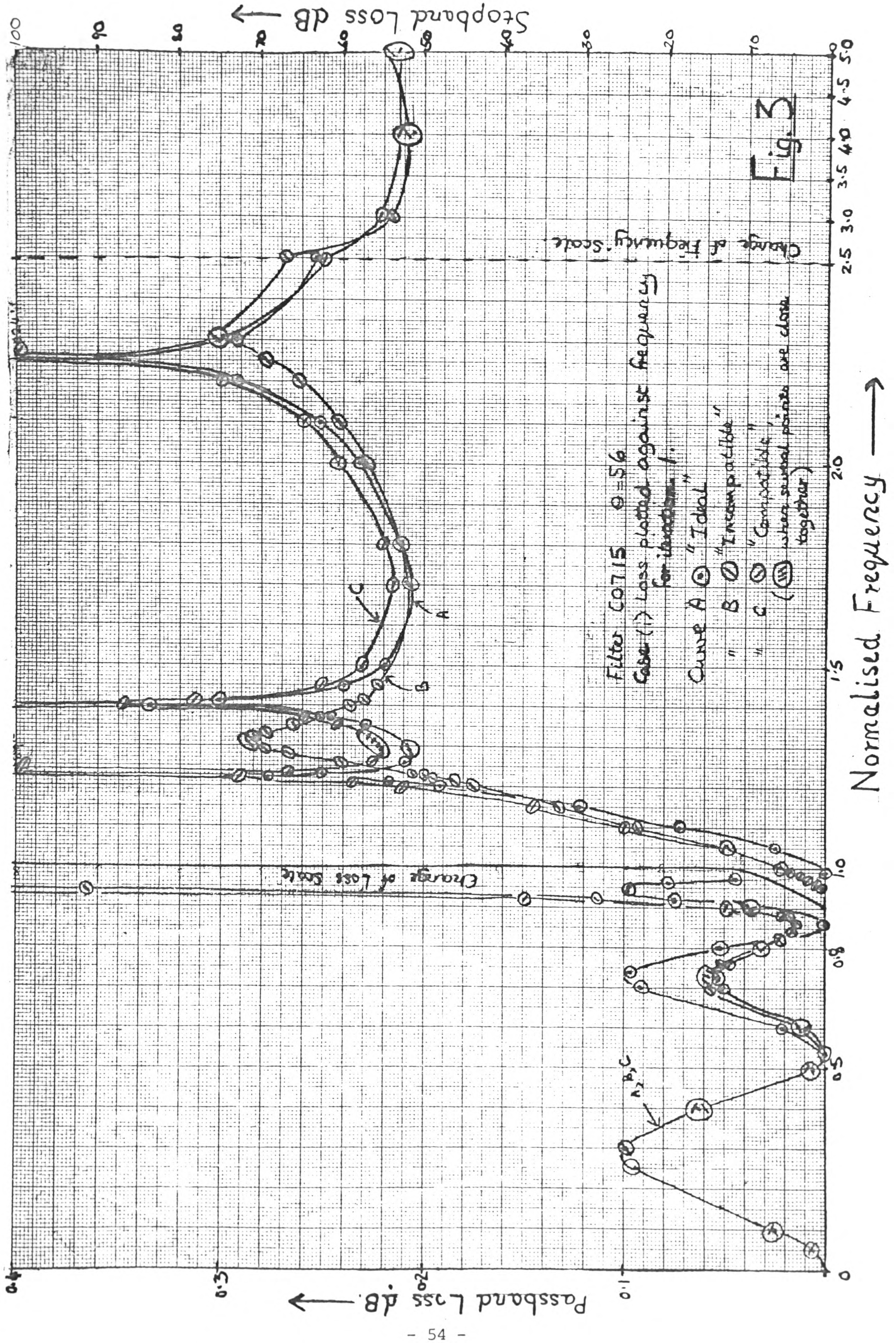
In case (i) all zeros of D+N are multiplied by a common factor 0.9. This is equivalent to changing the frequency scale.* However, this applies only to the 0th iteration, because during the improvement process the coefficients in the inaccurate K'(p) are compared with the coefficients in the original K(p) which is ideal with reference to the original frequency scale.

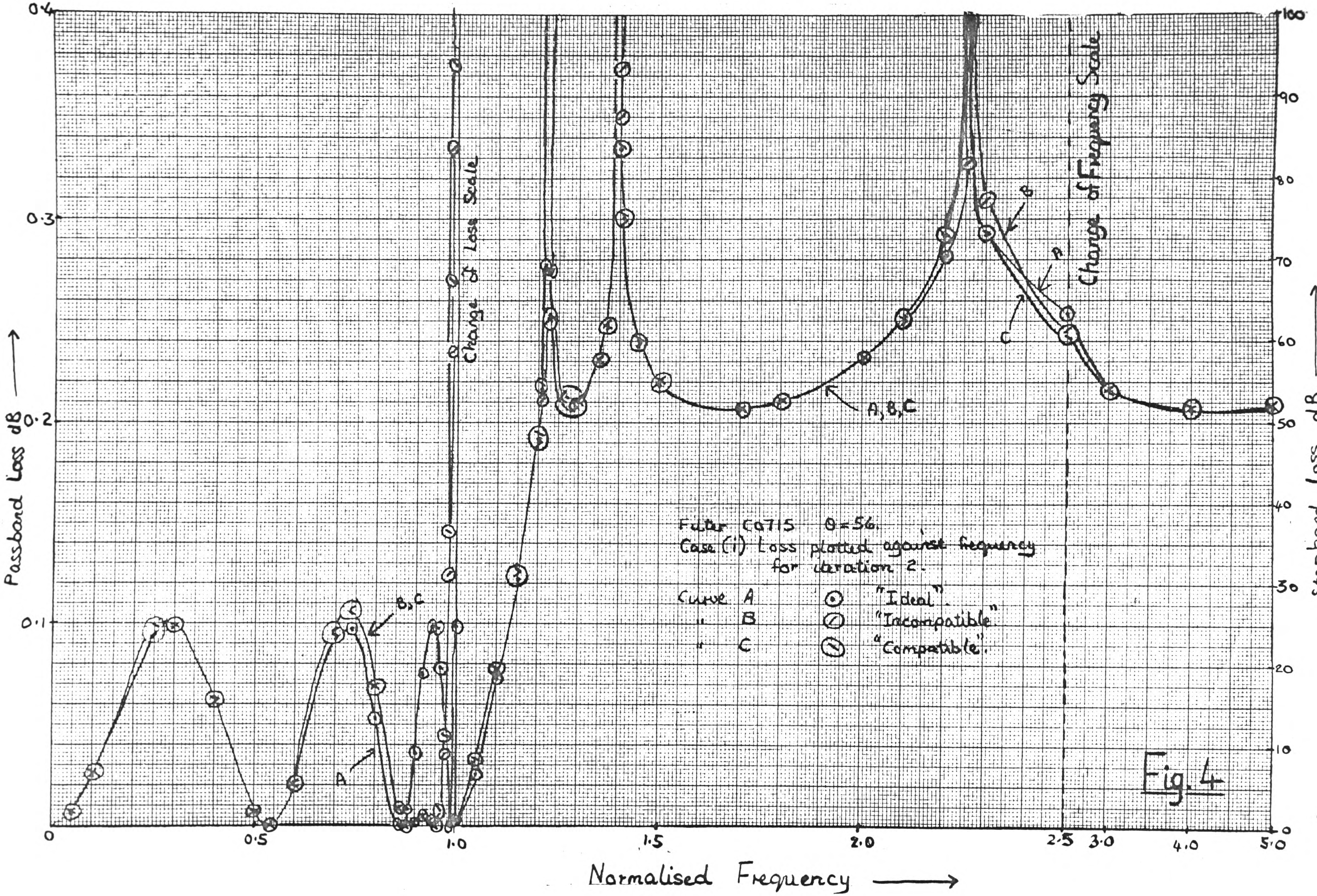
A general conclusion is that the stop bands are much better for "compatible" than for "incompatible" cases, but the passbands are mostly slightly worse (although some were improved); this is understandable since the operation of making Z'_{ai} and Z'_{bi} compatible is concerned with the stopband behaviour. It also appeared that in the "compatible" case an acceptable filter can be obtained after fewer iterations.

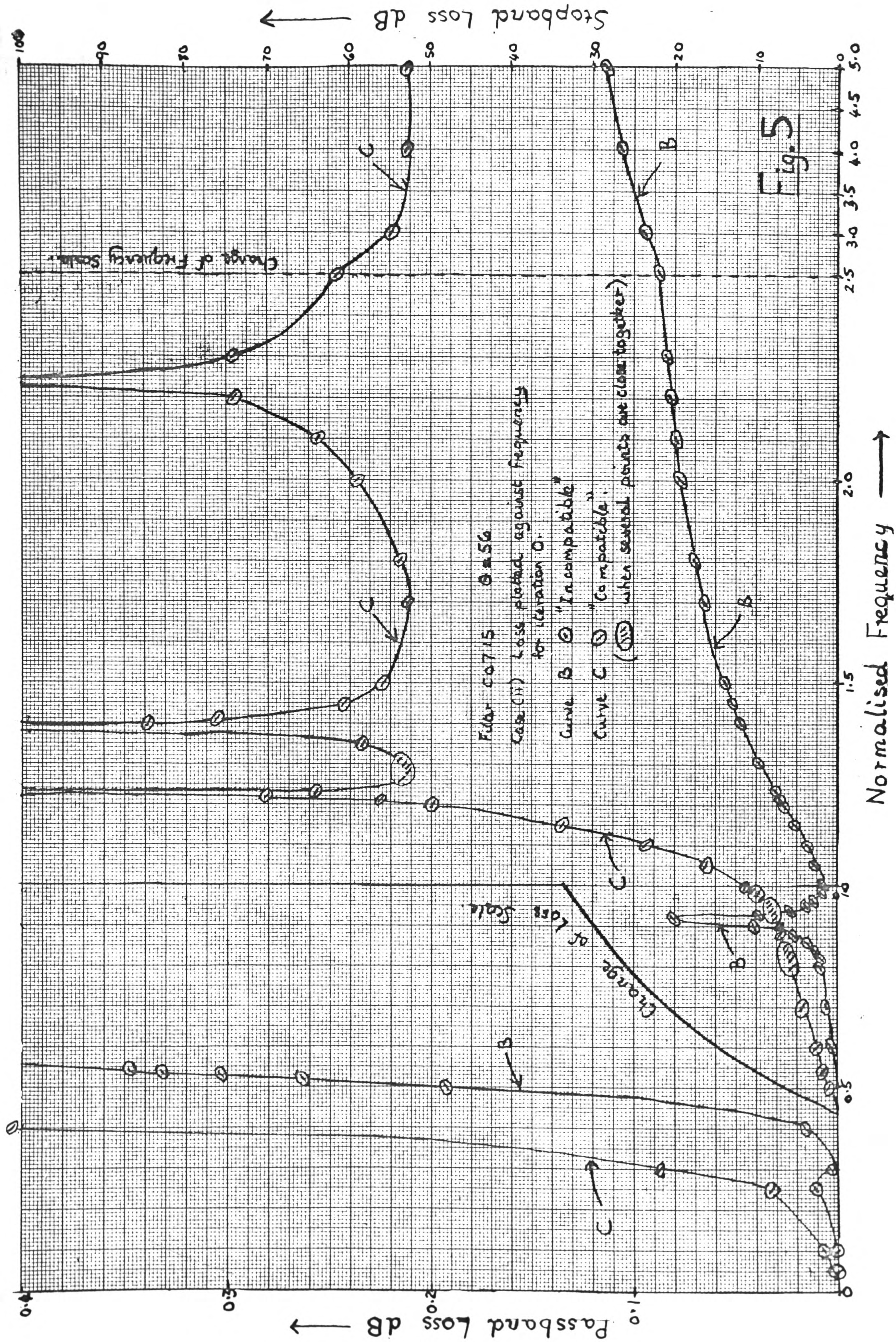
These encouraging results led to the method being investigated for a filter of order 9.

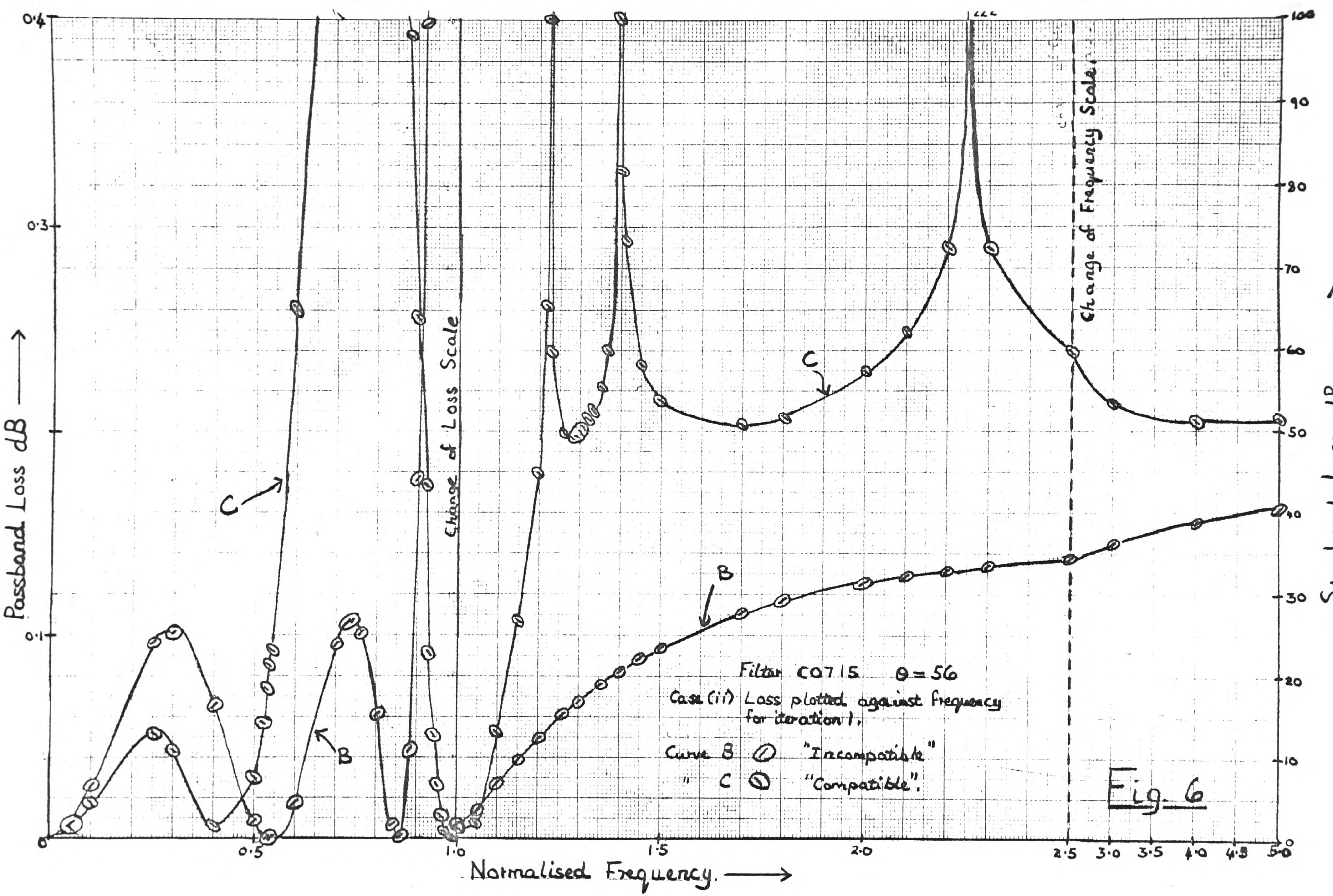
---

* This was foreseen and the results were used as a check.


Stopband Loss dB
Passband Loss dB
Normalised Frequency
Fig. 3
Filter C0715 θ = 56
Case (1) Loss plotted against frequency for iteration 1.
Curve A "Ideal"
" B "Incompatible"
" C "Compatible"
(when several points are close together)
Change of Frequency Scale
Change of Loss Scale
A, B, C

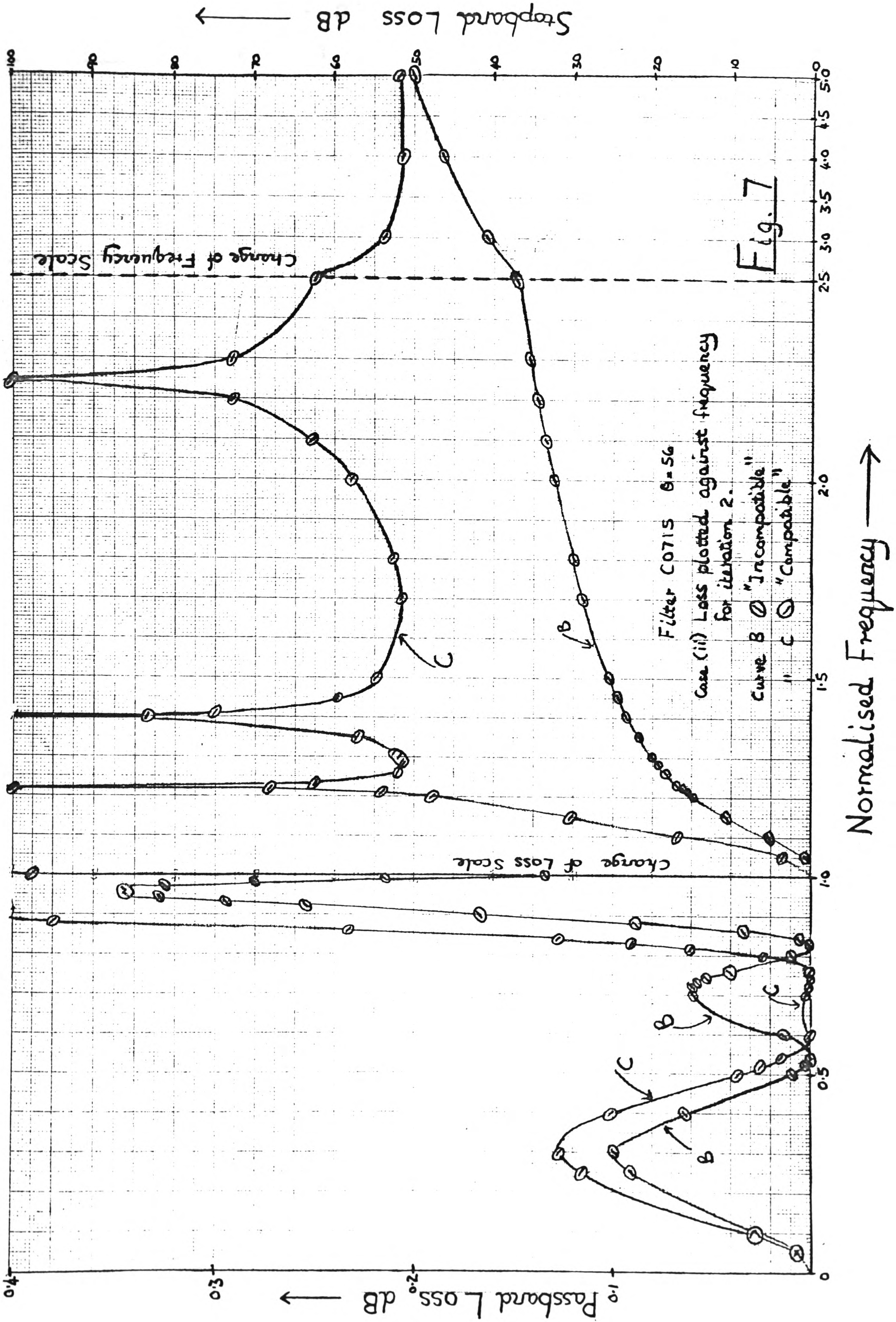

Passboard Loss dB
Stopband Loss dB.
Normalised Frequency
Change of Loss Scale
Change of Frequency Scale
Filter C0715 θ=56.
Case (i) Loss plotted against frequency
for iteration 2.
Curve A "Ideal".
" B "Incompatible".
" C "Compatible".
A, B, C
B, C
A
B
C
Fig. 4


Stopband Loss dB
Passband Loss dB
Normalised Frequency
Fig. 5
Filter C0715 θ = 56
Case (ii) Loss plotted against frequency for iteration 0.
Curve B ⊙ "Incompatible"
Curve C ⊘ "Compatible"
(when several points are close together)
Change of Frequency Scale.
Change of Loss Scale.
B
C



Passband Loss dB
Stopband Loss dB
Normalised Frequency
Change of Loss Scale
Change of Frequency Scale
222
B
C
Filter C0715 θ = 56
Case (ii) Loss plotted against frequency for iteration 1.
Curve B "Incompatible"
" C "Compatible".

Fig. 6


Stopband Loss dB
Passband Loss dB
Normalised Frequency
Fig. 7
Filter C0715 θ=56
Case (ii) Loss plotted against frequency for iteration 2.
Curve B "Incompatible"
" C "Compatible"
Change of Frequency Scale
Change of Loss Scale
B
C

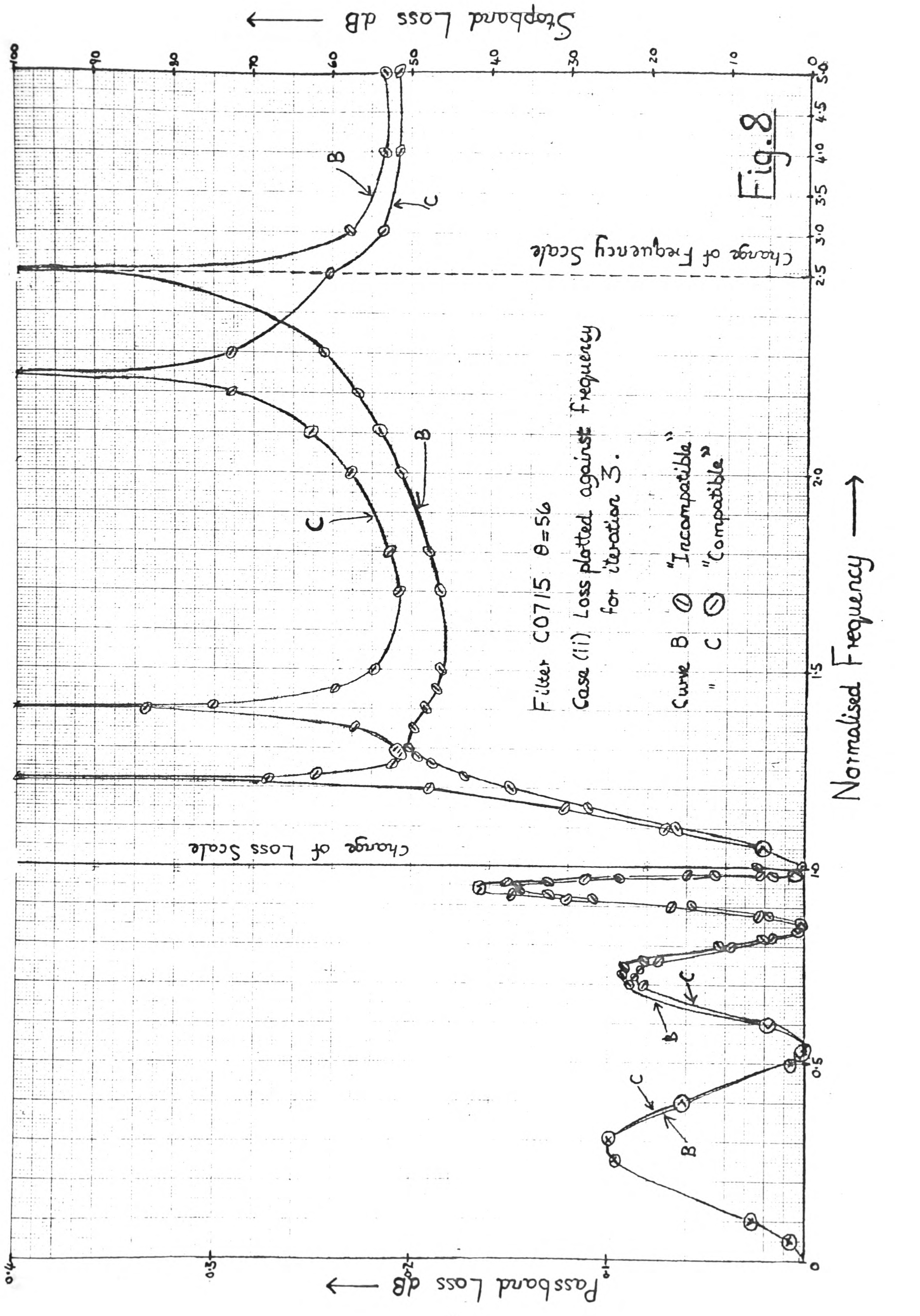


Fig. 8
Filter C0715 θ=56
Case (ii) Loss plotted against frequency for iteration 3.
Curve B "Incompatible"
" C "Compatible"
Stopband Loss dB
Passband Loss dB
Normalised Frequency
Change of Frequency Scale
Change of Loss Scale
B
C

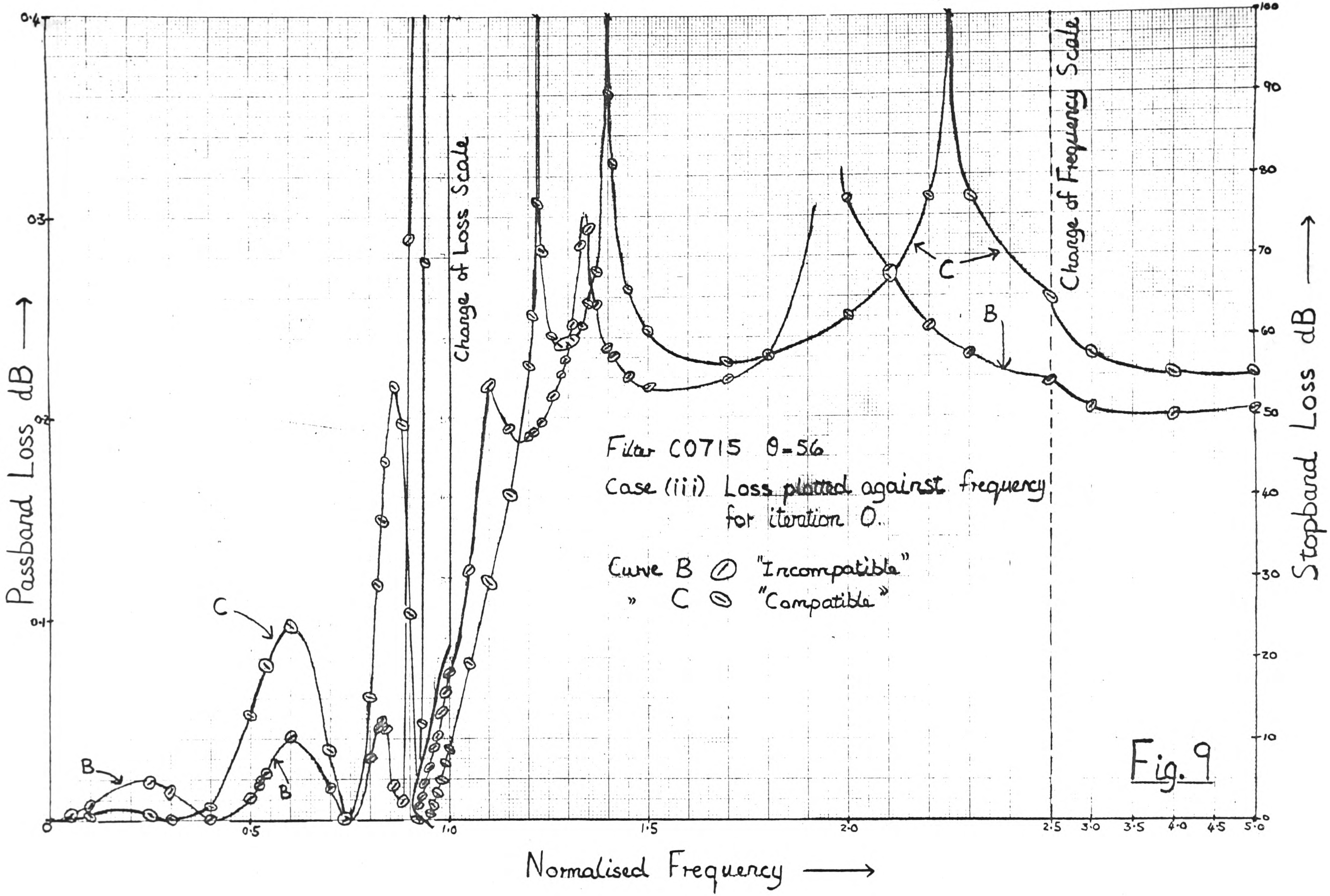

Passband Loss dB
Stopband Loss dB
Normalised Frequency
Change of Loss Scale
Change of Frequency Scale
Filter C0715 θ=56
Case (iii) Loss ~~plotted~~ against frequency for iteration 0.
Curve B "Incompatible"
" C "Compatible"
B
C
0
0·1
0·2
0·3
0·4
0·5
1·0
1·5
2·0
2·5
3·0
3·5
4·0
4·5
5·0
10
20
30
40
50
60
70
80
90
100

Fig. 9

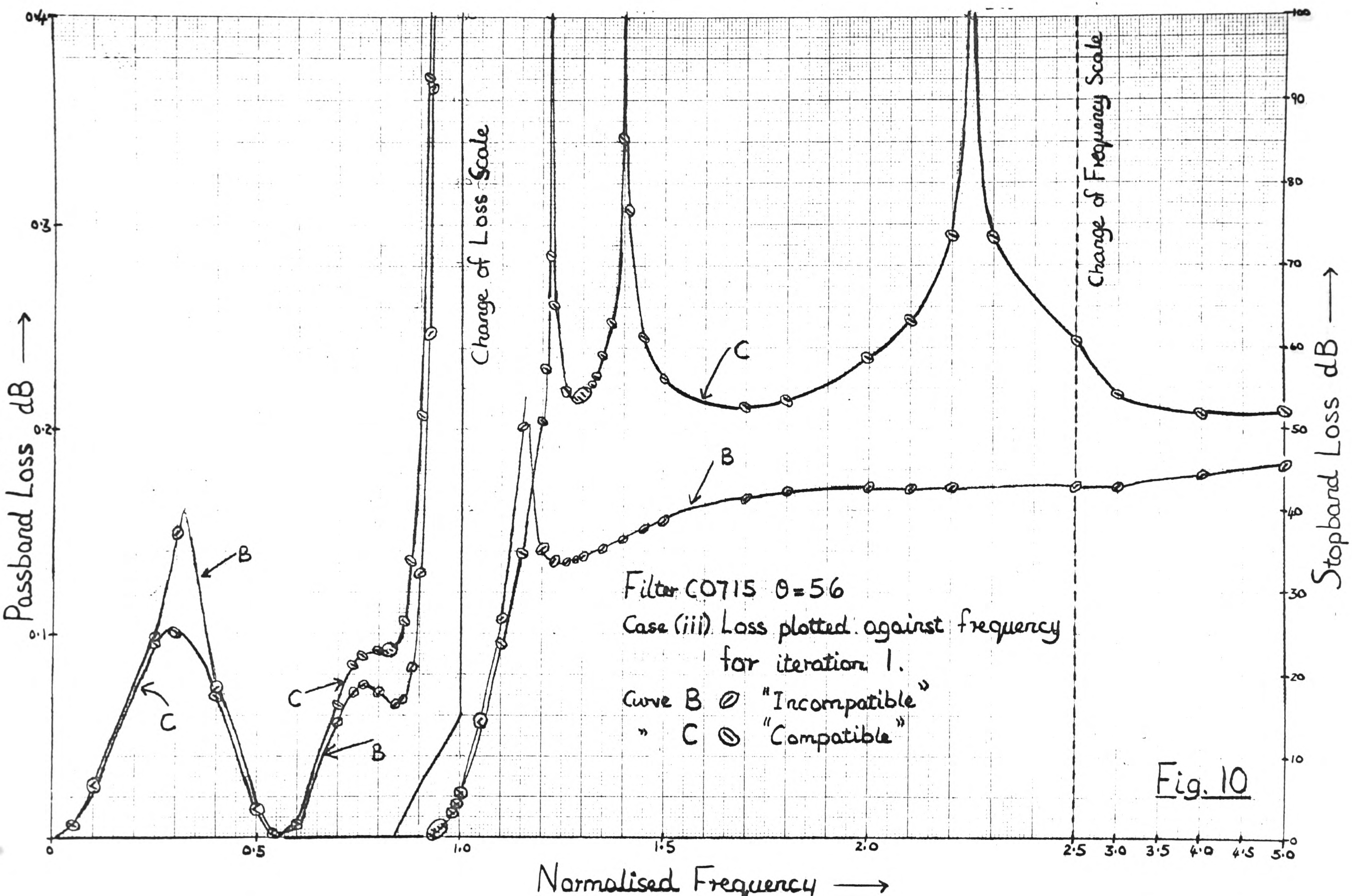

Passband Loss dB →
Stopband Loss dB →
Normalised Frequency →
Change of Loss Scale
Change of Frequency Scale
Filter CO715 θ=56
Case (iii) Loss plotted against frequency
for iteration 1.
Curve B "Incompatible"
" C "Compatible"
B
C
0
0·1
0·2
0·3
0·4
0·5
1·0
1·5
2·0
2·5
3·0
3·5
4·0
4·5
5·0
10
20
30
40
50
60
70
80
90
100

Fig. 10

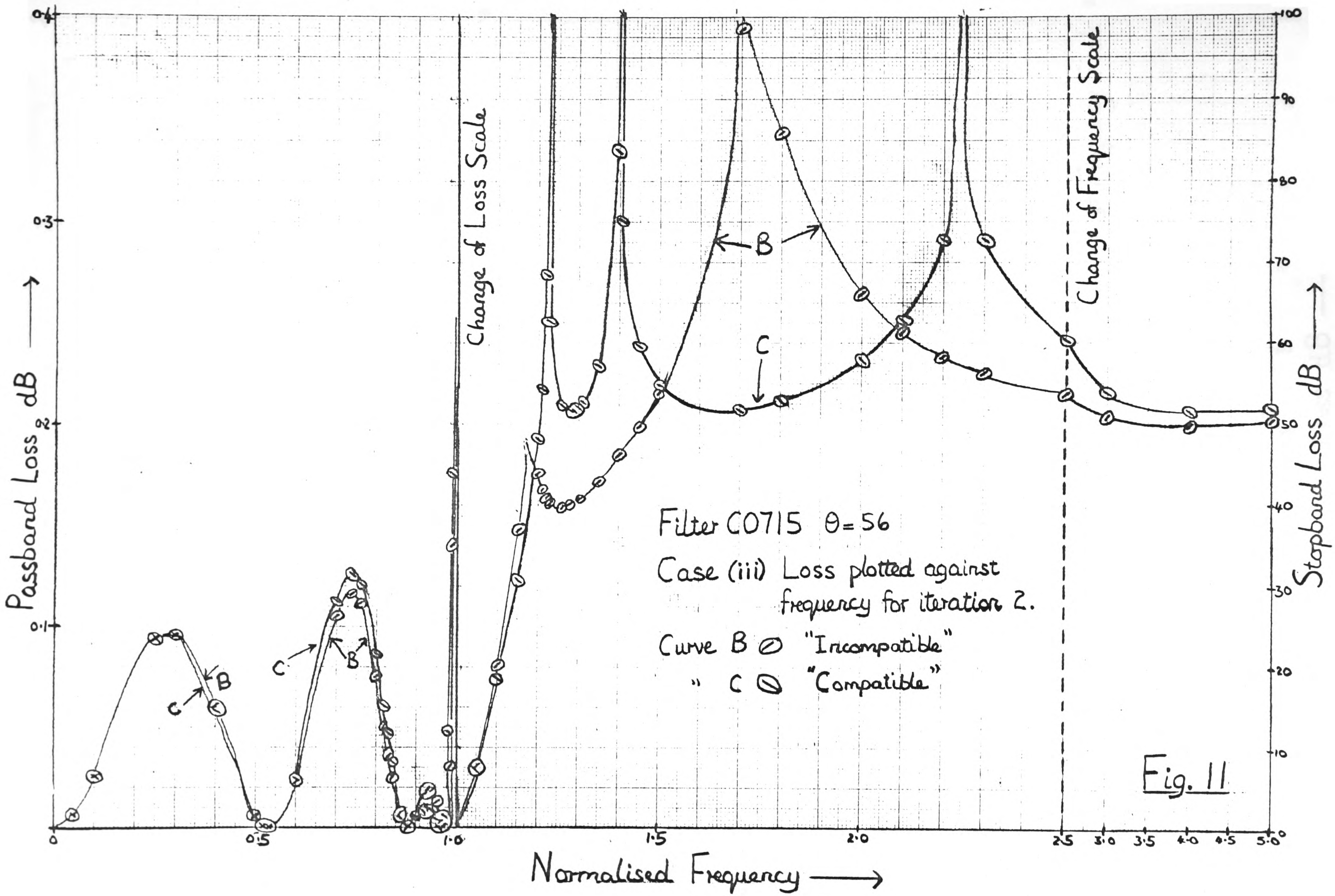

Filter C0715 θ = 56
Case (iii) Loss plotted against frequency for iteration 2.
Curve B "Incompatible"
" C "Compatible"
Change of Loss Scale
Change of Frequency Scale
Passband Loss dB
Stopband Loss dB
Normalised Frequency
B
C
0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
0.1
0.2
0.3
0.4
10
20
30
40
50
60
70
80
90
100

Fig. 11

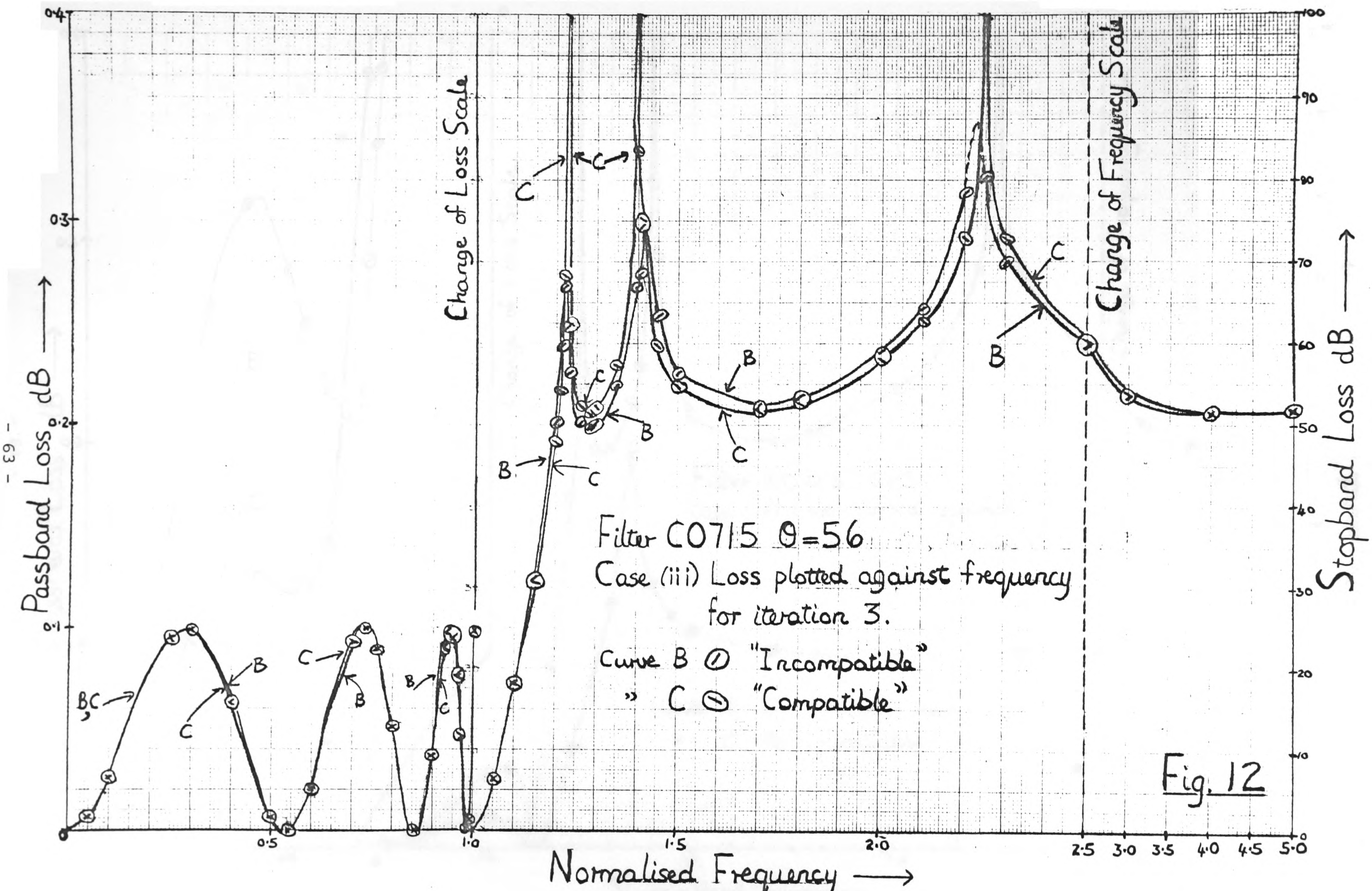

Passband Loss, dB
Stopband Loss dB
Normalised Frequency
Change of Loss Scale
Change of Frequency Scale
Filter C0715 θ=56
Case (iii) Loss plotted against frequency
for iteration 3.
Curve B "Incompatible"
" C "Compatible"
0.4
0.3
0.2
0.1
0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
100
90
80
70
60
50
40
30
20
10
B
C
BC

Fig. 12

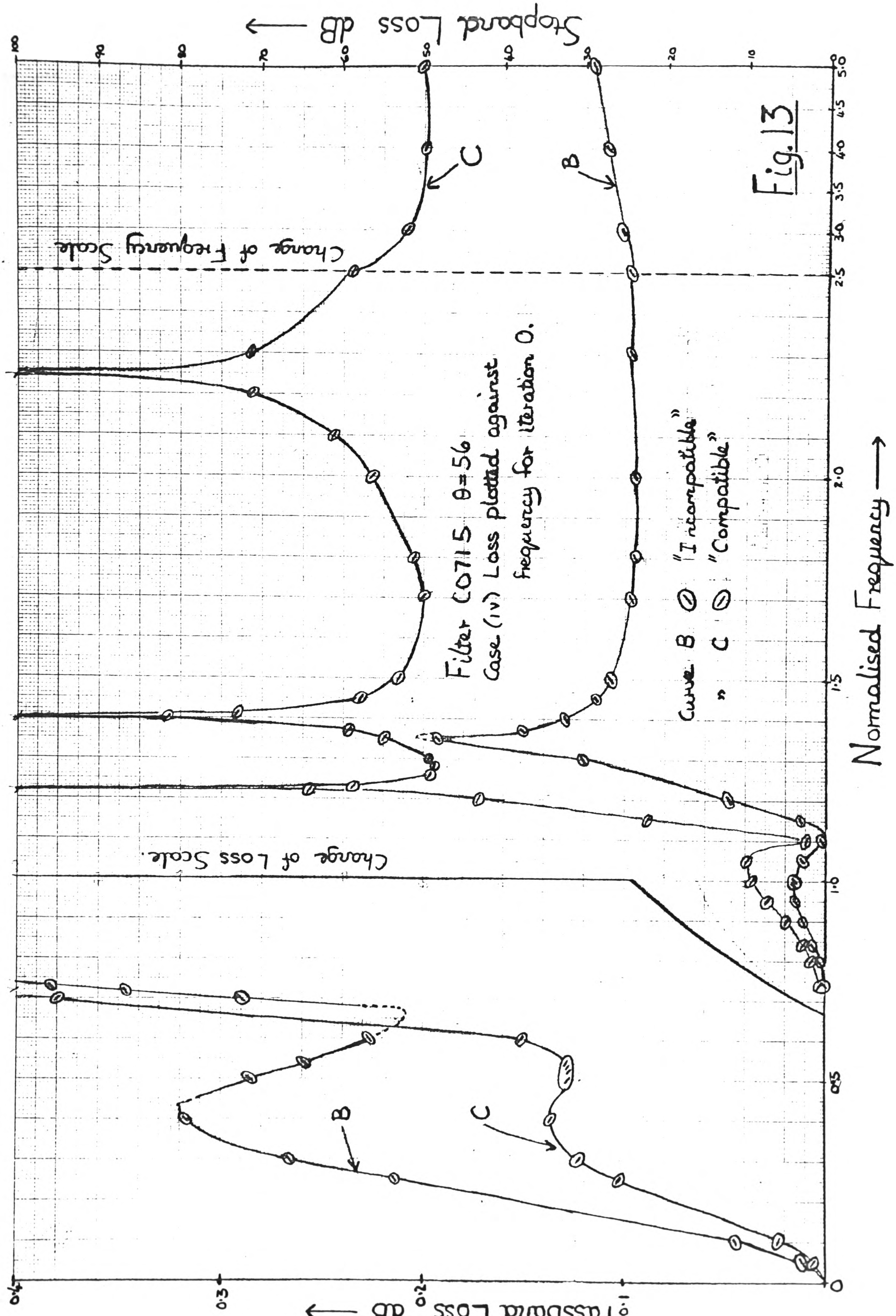
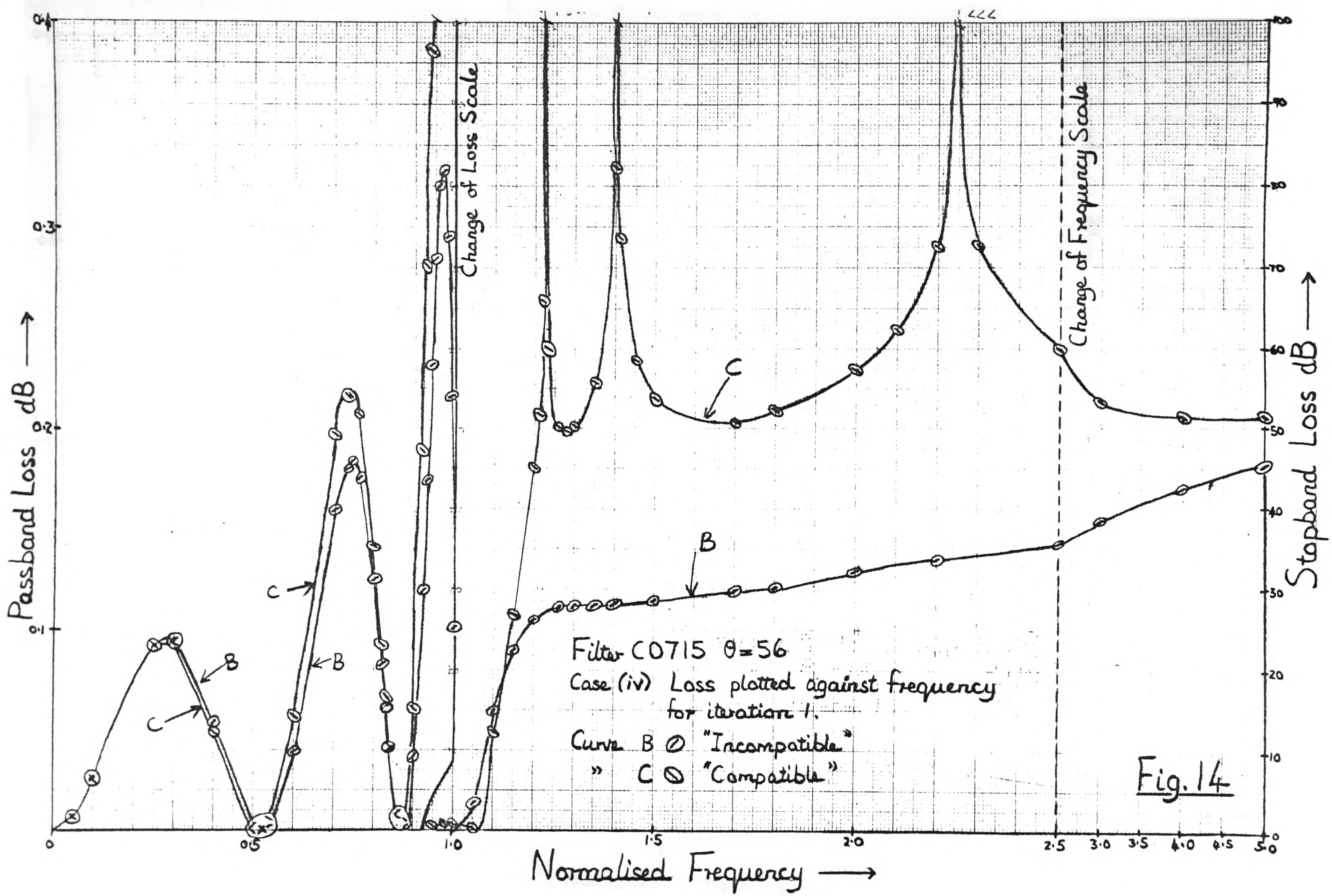
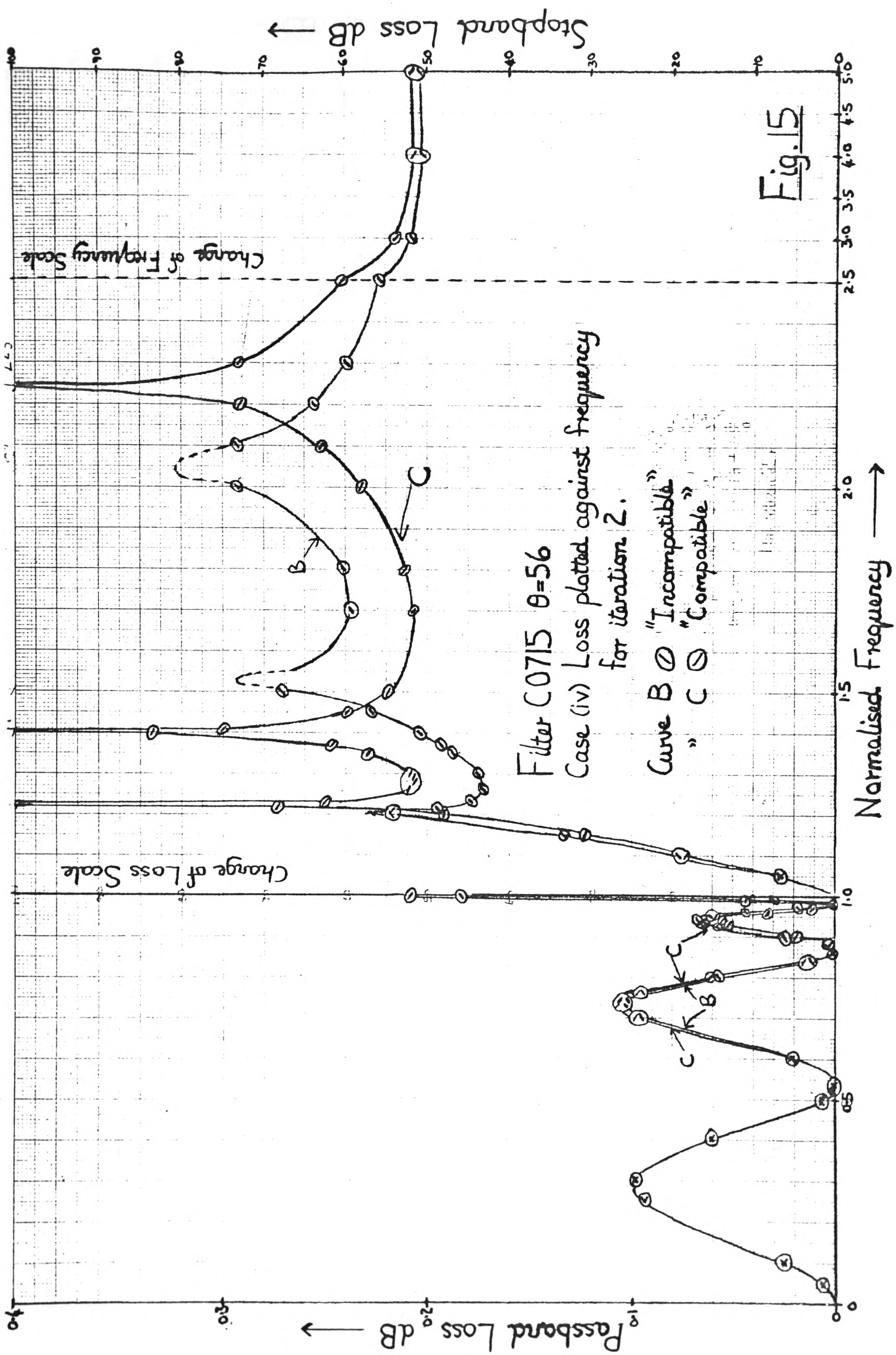
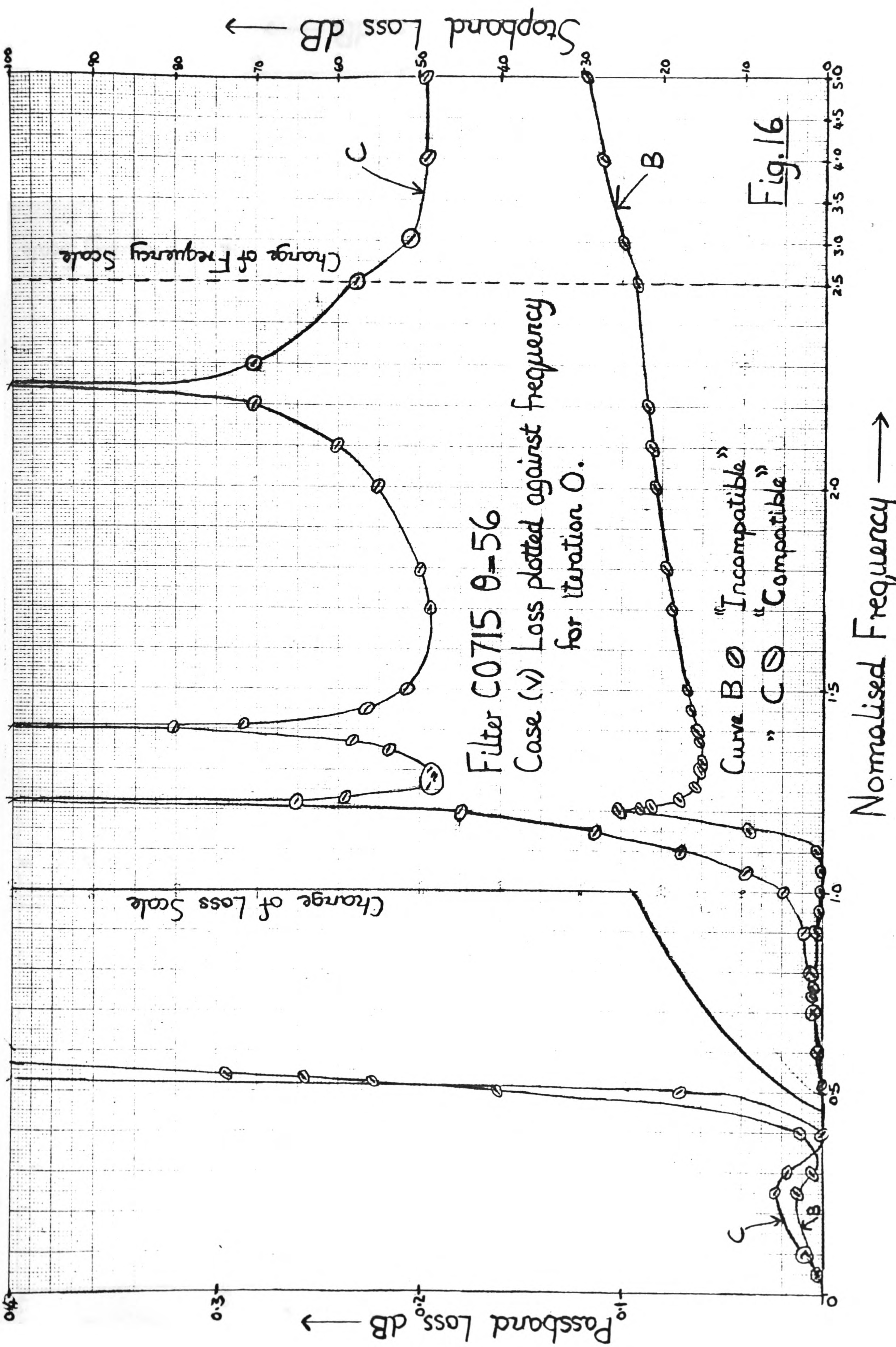


Fig. 13
Filter C0715 θ=56
Case (iv) Loss plotted against
frequency for iteration 0.
Curve B "Incompatible"
" C "Compatible"
Change of Frequency Scale
Change of Loss Scale.
Stopband Loss dB
Passband Loss dB
Normalised Frequency
B
C


Passband Loss dB →
Stopband Loss dB →
Normalised Frequency →
Change of Loss Scale
Change of Frequency Scale
Filter C0715 θ=56
Case (iv) Loss plotted against frequency
for iteration 1.
Curve B ⊘ "Incompatible"
" C ⊘ "Compatible"
B
C
0
0·5
1·0
1·5
2·0
2·5
3·0
3·5
4·0
4·5
5·0
0·1
0·2
0·3
0·4
10
20
30
40
50
60
70
80
90
100
Fig. 14



Stopband Loss dB
Passband Loss dB
Normalised Frequency
Change of Frequency Scale
Change of Loss Scale
Filter C0715 θ=56
Case (iv) Loss plotted against frequency for iteration 2.
Curve B "Incompatible"
" C "Compatible"
B
C
Fig. 15


Stopband Loss dB
Passband Loss dB
Normalised Frequency
Change of Frequency Scale
Change of Loss Scale
Filter C0715 θ=56
Case (v) Loss plotted against frequency
for iteration 0.
Curve B ⊘ "Incompatible"
" C ⊘ "Compatible"
B
C
100
90
80
70
60
50
40
30
20
10
0
0.4
0.3
0.2
0.1
0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
Fig. 16

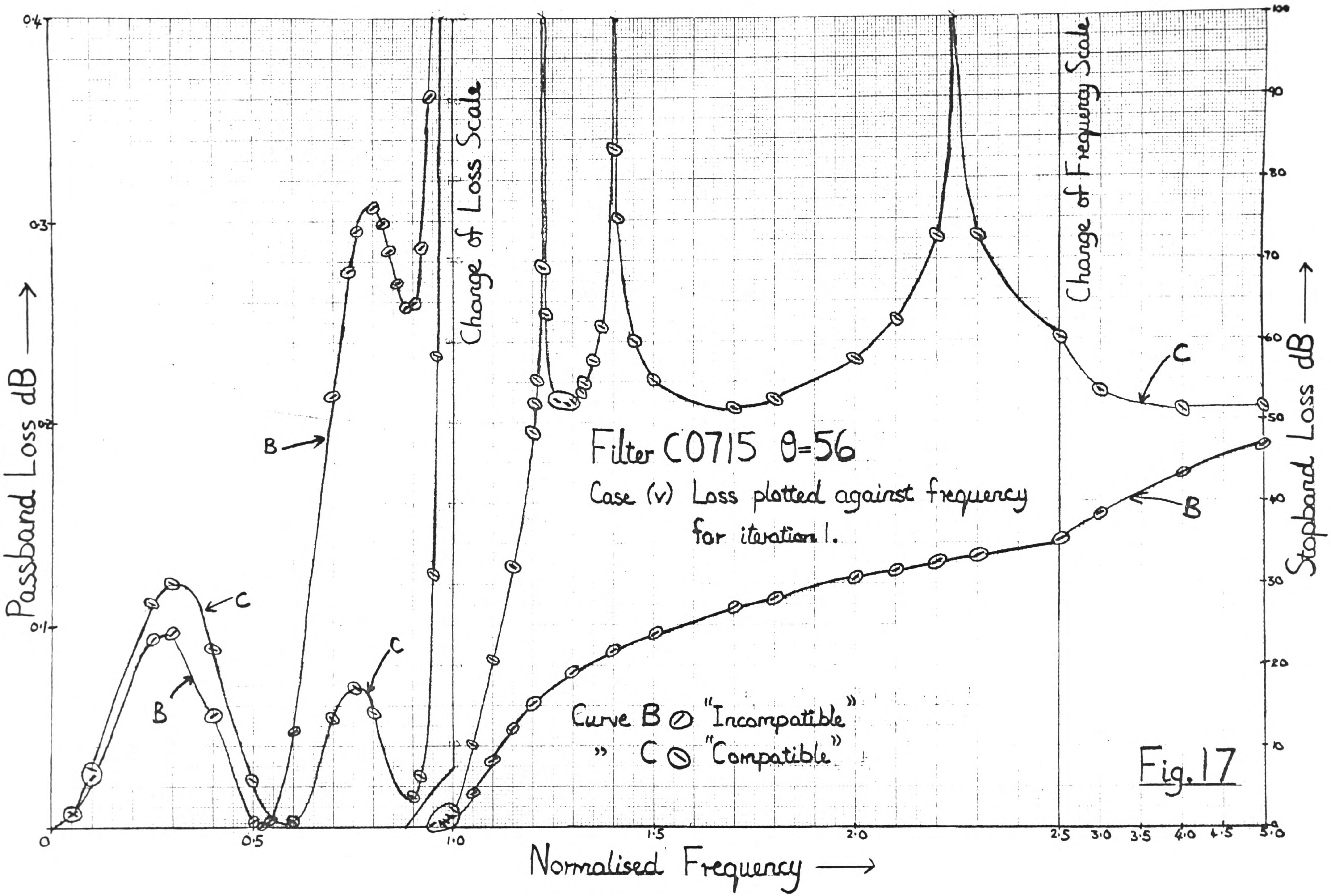
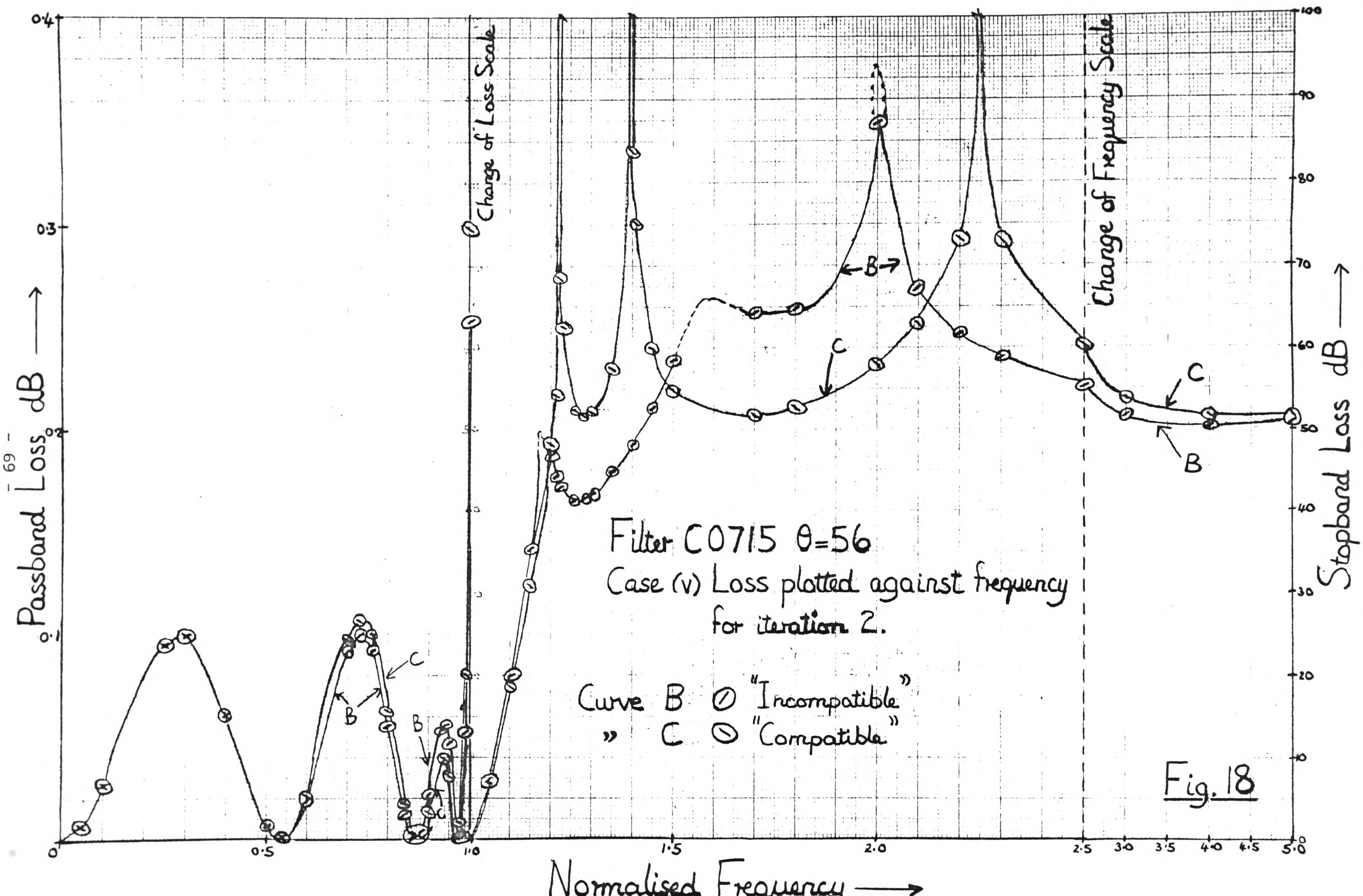

Filter C0715 θ=56
Case (v) Loss plotted against frequency for iteration 1.
Curve B ⊘ "Incompatible"
" C ⊘ "Compatible"
Change of Loss Scale
Change of Frequency Scale
B
C
Passband Loss dB
Stopband Loss dB
Normalised Frequency
0
0.1
0.2
0.3
0.4
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
10
20
30
40
50
60
70
80
90
100

Fig. 17


Filter C0715 θ=56
Case (v) Loss plotted against frequency
for iteration 2.
Curve B "Incompatible"
" C "Compatible"
Passband Loss dB
Stopband Loss dB
Normalised Frequency
Change of Loss Scale
Change of Frequency Scale
B
C
0
0·5
1·0
1·5
2·0
2·5
3·0
3·5
4·0
4·5
5·0
0·1
0·2
0·3
0·4
10
20
30
40
50
60
70
80
90
100
Fig. 18

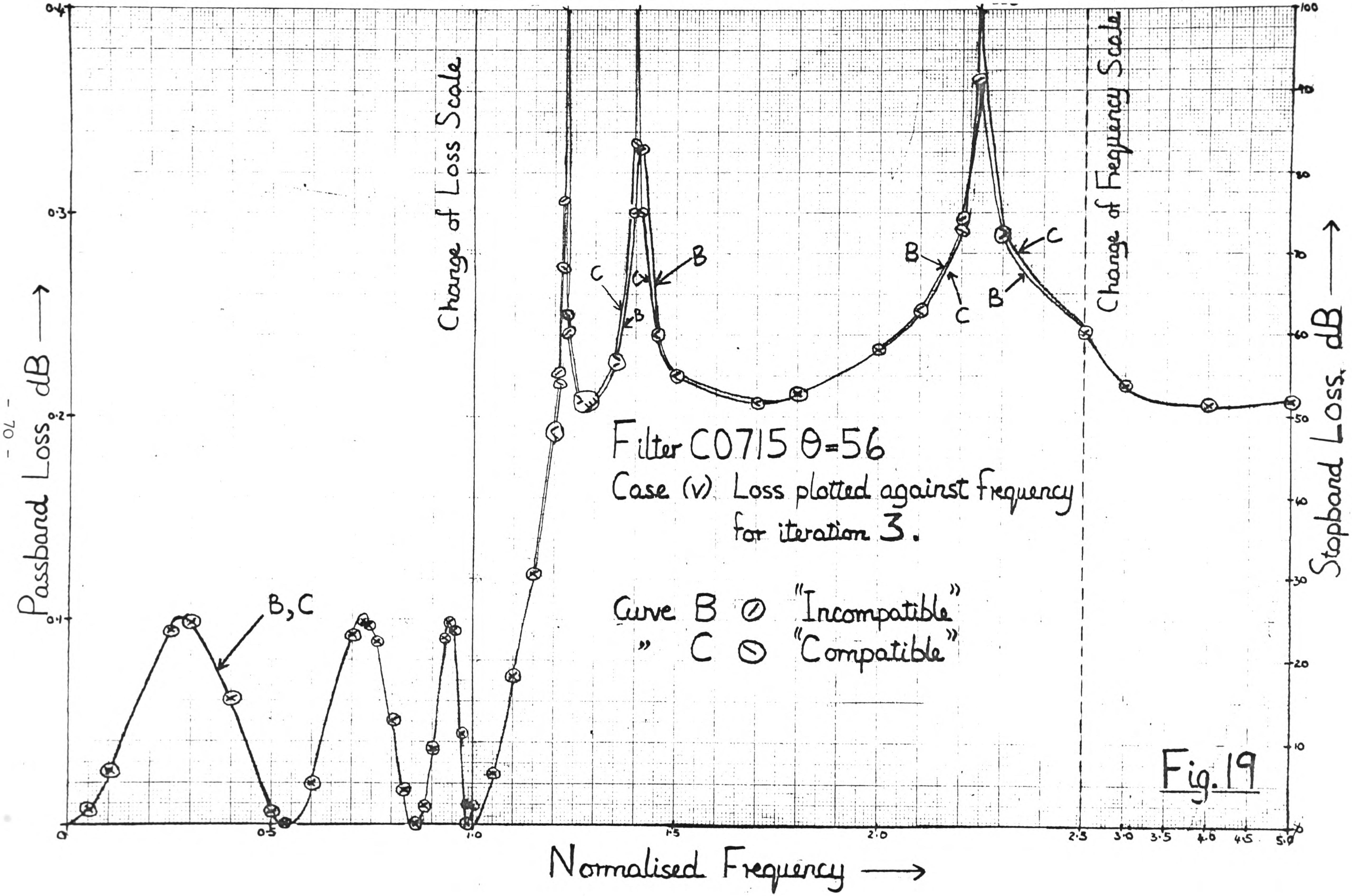

Filter C0715 θ=56
Case (v) Loss plotted against frequency for iteration 3.
Curve B ⊘ "Incompatible"
" C ⊖ "Compatible"
Change of Loss Scale
Change of Frequency Scale
Passband Loss, dB
Stopband Loss, dB
Normalised Frequency
B, C
B
C
0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
5.0
0.1
0.2
0.3
0.4
10
20
30
40
50
60
70
80
90
100

Fig. 19

4.2.2. n=9

Saraga's method was investigated further, a filter of order 9 being used. Once again only option 1 fig.1 of Saraga's method was employed. The values of p at which the zeros of $D(p)+N(p)$ occur were taken from published tables (ref.11) - where they are tabulated in the form of zeros of $H(p)$ - and inaccuracies were introduced by multiplying, in turn, each zero by 1.01 and 1.1, i.e. by adding 1% and 10% to each. The case of 1% inaccuracies revealed after a single iteration that the errors were too small to be of interest. The case of 10%, on the other hand, required more than 11 iterations, and two zeros were interchanged, showing the error was too large and therefore unsuitable as an initial error. For these reasons the process was repeated with the more realistic multiplication factors of 0.95 and 1.05.

Table 5 gives:-

(a) the accurate values of the zeros of $B_a(p)$, $-\alpha_1 \pm j\beta_1$ and $-\alpha_2 \pm j\beta_2$, and the zeros of $B_b(p)$, $-\alpha_3 \pm j\beta_3$, $-\alpha_4 \pm j\beta_4$ and $-\alpha_5$;

(b) the new starting values, i.e. the above zeros modified by factors 0.95 and 1.05 ;

(c) the numbers of the figures showing the corresponding loss frequency plots ;

(d) the values of the attenuation poles Ω_1, Ω_2, Ω_3 and Ω_4 .

Table 5. Filter type CO915, θ=70.

Ω_1 = 1.0691285656, Ω_2 = 1.1193168508, Ω_3 = 1.3023470398, Ω_4 = 2.0944721998.

Case \ Iteration number	$B'_a(p)$	$B'_b(p)$	Fig. numbers for plots of loss/frequency values for: Incompatible parameters		Compatible parameters	
			0 to 2	3,4	0 to 2	3,4
(i)	$-1.05\alpha_1 \pm j1.05\beta_1$ $-1.05\alpha_2 \pm j1.05\beta_2$	$-1.05\alpha_3 \pm j1.05\beta_3$ $-1.05\alpha_4 \pm j1.05\beta_4$ $-1.05\alpha_5$	20	21	22	23*
(ii)	$-0.95\alpha_1 \pm j0.95\beta_1$ $-0.95\alpha_2 \pm j0.95\beta_2$	$-0.95\alpha_3 \pm j0.95\beta_3$ $-0.95\alpha_4 \pm j0.95\beta_4$ $-0.95\alpha_5$	24	25	26	27
(iii)	$-0.95\alpha_1 \pm j0.95\beta_1$ $-1.05\alpha_2 \pm j1.05\beta_2$	$-0.95\alpha_3 \pm j0.95\beta_3$ $-1.05\alpha_4 \pm j1.05\beta_4$ $-0.95\alpha_5$	28	29	30	31
(iv)	$-1.05\alpha_1 \pm j1.05\beta_1$ $-0.95\alpha_2 \pm j0.95\beta_2$	$-1.05\alpha_3 \pm j1.05\beta_3$ $-0.95\alpha_4 \pm j0.95\beta_4$ $-1.05\alpha_5$	32	33	34**	35

$-\alpha_1+j\beta_1$=-0.3519690945±j0.5907712658 $-\alpha_2+j\beta_2$= -0.0573133222±j0.9781899505	$-\alpha_3 \pm j\beta_3$= -0.1586637177±j0.8776774257 $-\alpha_4 \pm j\beta_4$= -0.0137793011±j1.0084930305 $-\alpha_5$= -0.4905954745

The "ideal" loss/frequency values, calculated from the function K(p), are shown in each figure.

---

* The loss/frequency values for iteration number 5 are also plotted in fig.23.

** Loss/frequency values for iteration 0, compatible parameters case (iv) were not plotted.

The loss/frequency points for a filter of order $n=9$ were plotted automatically, by means of a "Kingmatic Drafting Machine" at Imperial College (instead of by hand, as for $n=7$) but the curves which connected the points were drawn by hand so that, if necessary, extra points could be entered later in areas of particular interest.

Again a general conclusion was that most of the passbands are slightly worse and the stopbands are much better for the "compatible" than for the "incompatible" case. These results gave further encouragement for the use of the method and it was decided to test it under more realistic circumstances and compare the results with established filter-design methods.

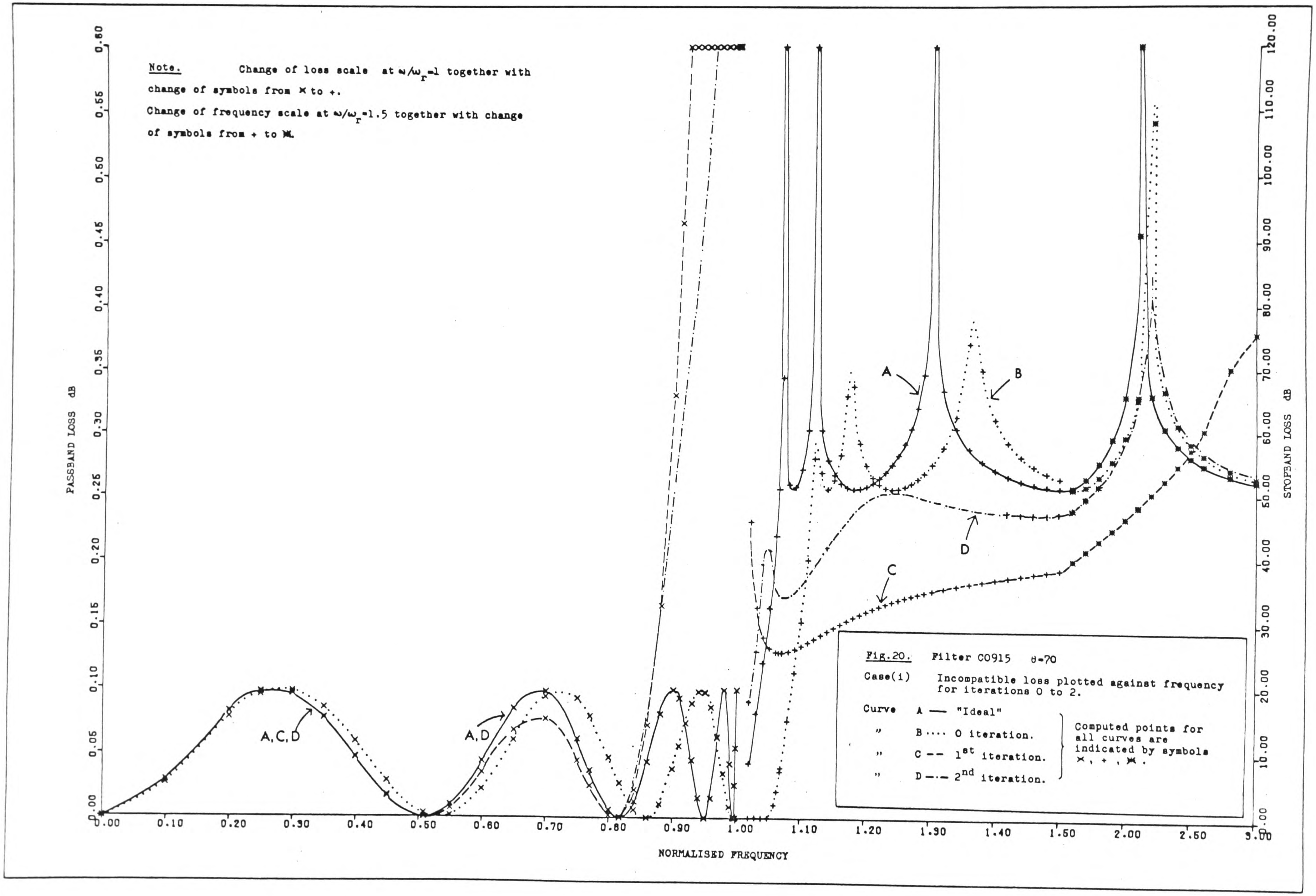


Fig.20. Filter C0915 $\theta=70$

Case(i) Incompatible loss plotted against frequency for iterations 0 to 2.

Curve A —— "Ideal"
" B ···· 0 iteration.
" C – – 1[st] iteration.
" D –·– 2[nd] iteration.

Computed points for all curves are indicated by symbols ×, + , ⋈.

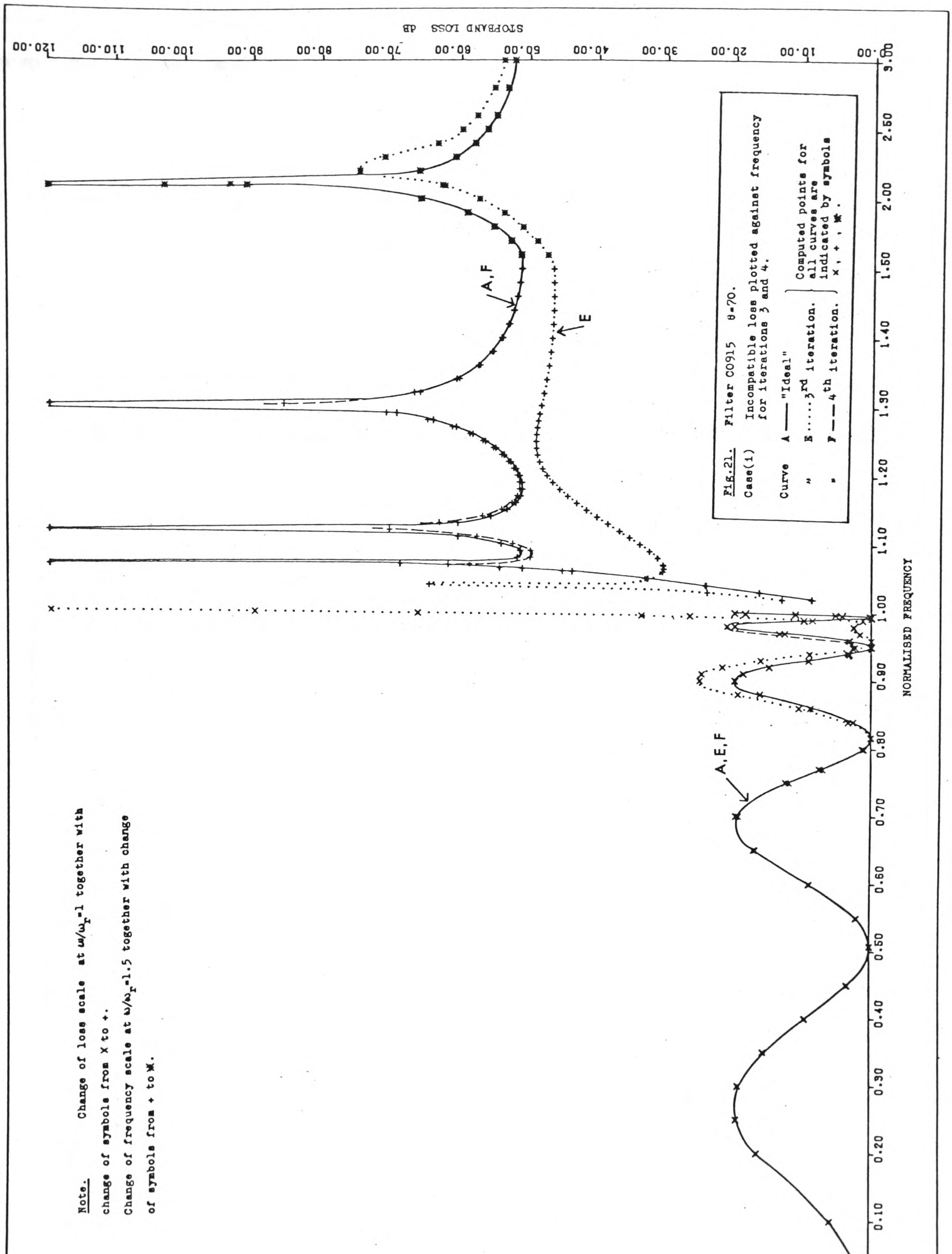


Fig.21. Filter C0915 θ=70.
Case(1) Incompatible loss plotted against frequency for iterations 3 and 4.
Curve A —— "Ideal"
" E ..... 3rd iteration.
" F — — 4th iteration.

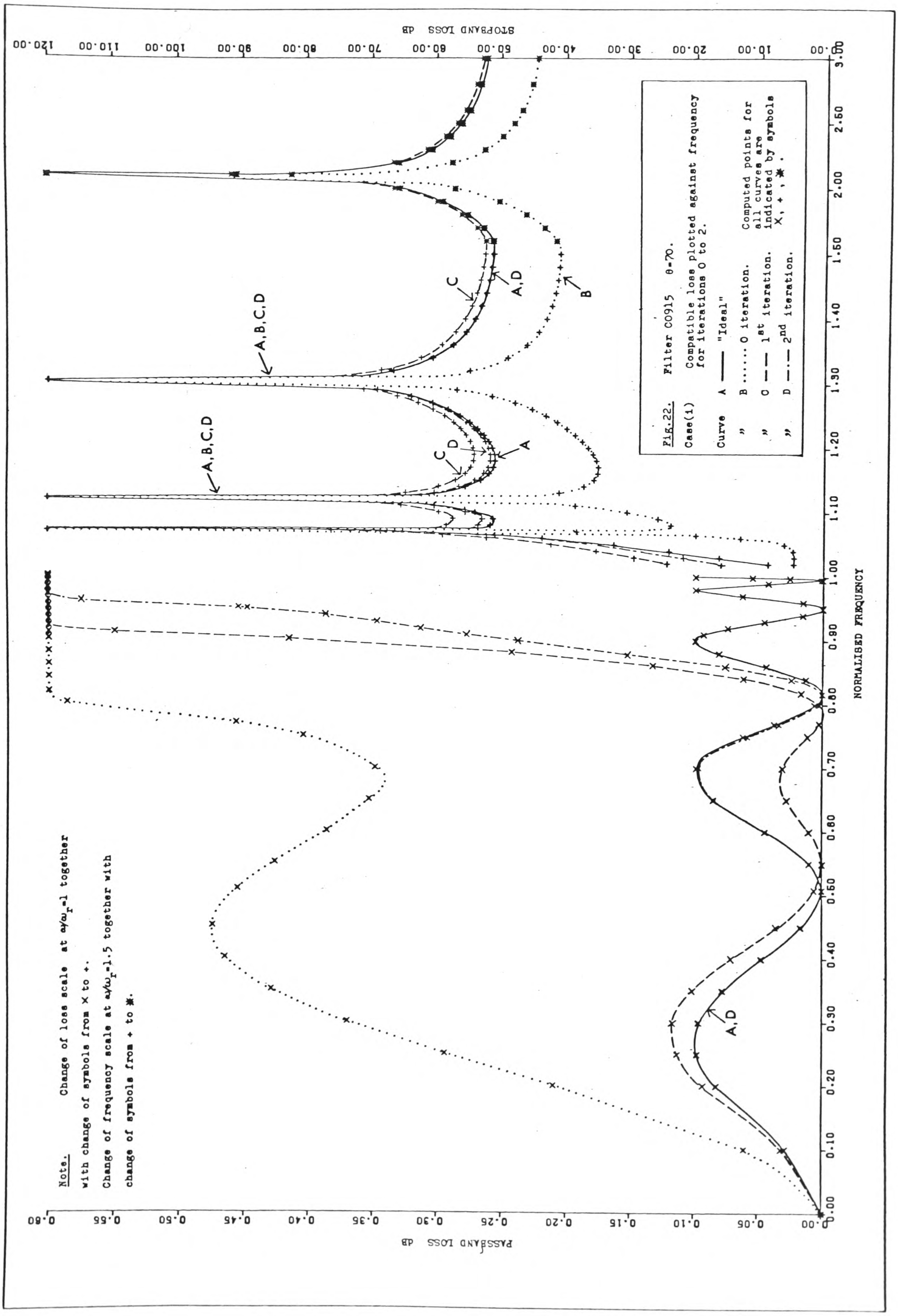


Fig.22. Filter CO915 θ-70.
Case(1) Compatible loss plotted against frequency for iterations 0 to 2.
Curve A —— "Ideal"
" B ...... 0 iteration.
" C — — 1st iteration.
" D —·— 2nd iteration.
Computed points for all curves are indicated by symbols X, +, ✱.
Note. Change of loss scale at ω/ω_r=1 together with change of symbols from × to +.
Change of frequency scale at ω/ω_r=1.5 together with change of symbols from + to ✱.
STOPBAND LOSS dB
PASSBAND LOSS dB
NORMALISED FREQUENCY
A,B,C,D
A,B,C,D
C
D
A
C
A,D
B
A,D
0.00
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.80
0.90
1.00
1.10
1.20
1.30
1.40
1.50
1.60
2.00
2.50
3.00
10.00
20.00
30.00
40.00
50.00
60.00
70.00
80.00
90.00
100.00
110.00
120.00
0.05
0.15
0.20
0.25
0.30
0.35
0.40
0.45
0.50
0.55
0.60

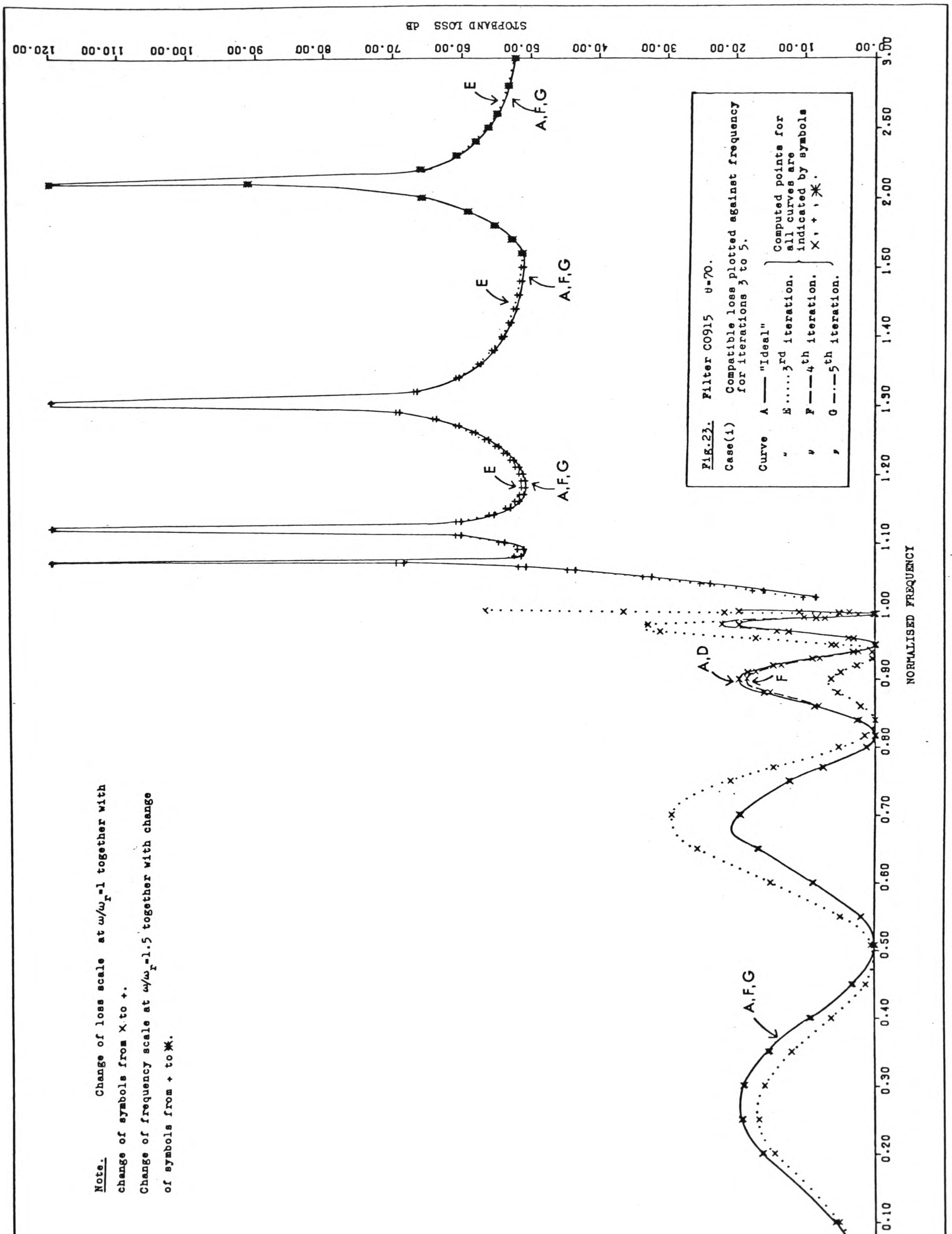


Fig.23. Filter C0915 $\theta=70$.

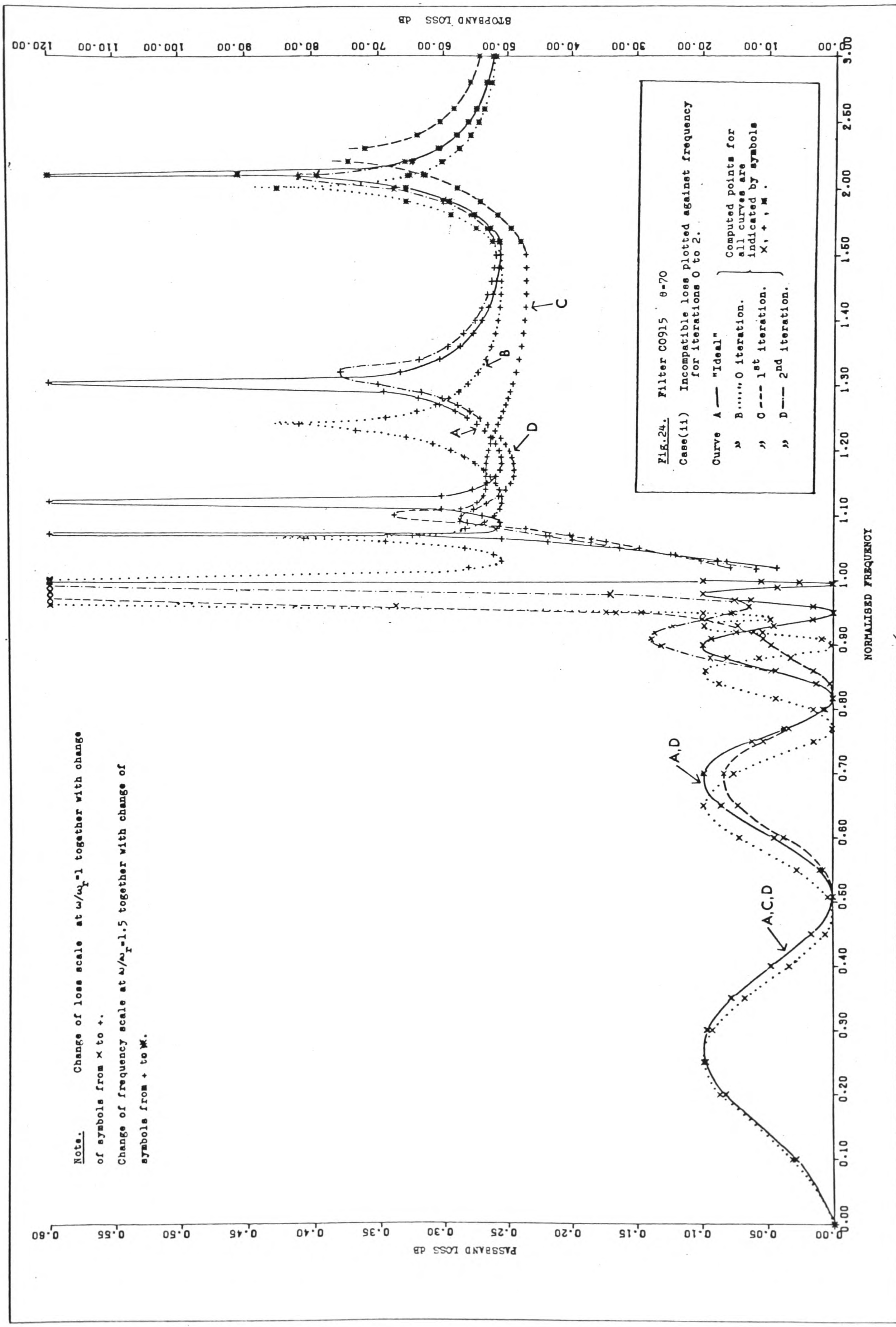
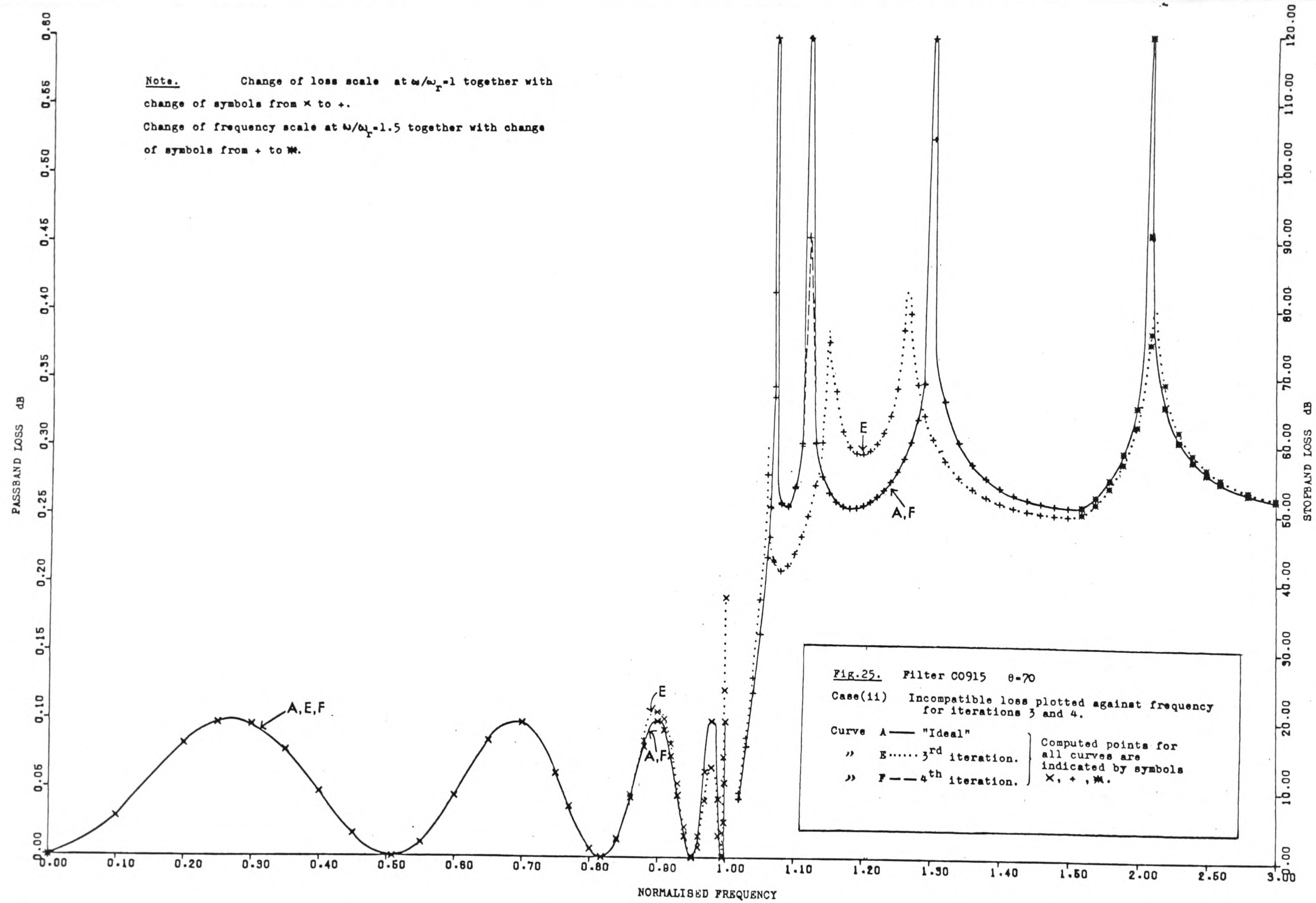


Fig.24. Filter CO915 θ=70
Case(ii) Incompatible loss plotted against frequency for iterations 0 to 2.



Note. Change of loss scale at $\omega/\omega_r = 1$ together with change of symbols from × to +.
Change of frequency scale at $\omega/\omega_r = 1.5$ together with change of symbols from + to ✻.

Fig.25. Filter C0915 θ=70

Case(ii) Incompatible loss plotted against frequency for iterations 3 and 4.

Curve A —— "Ideal"
" E ...... 3rd iteration.
" F — — 4th iteration.

Computed points for all curves are indicated by symbols ×, +, ✻.

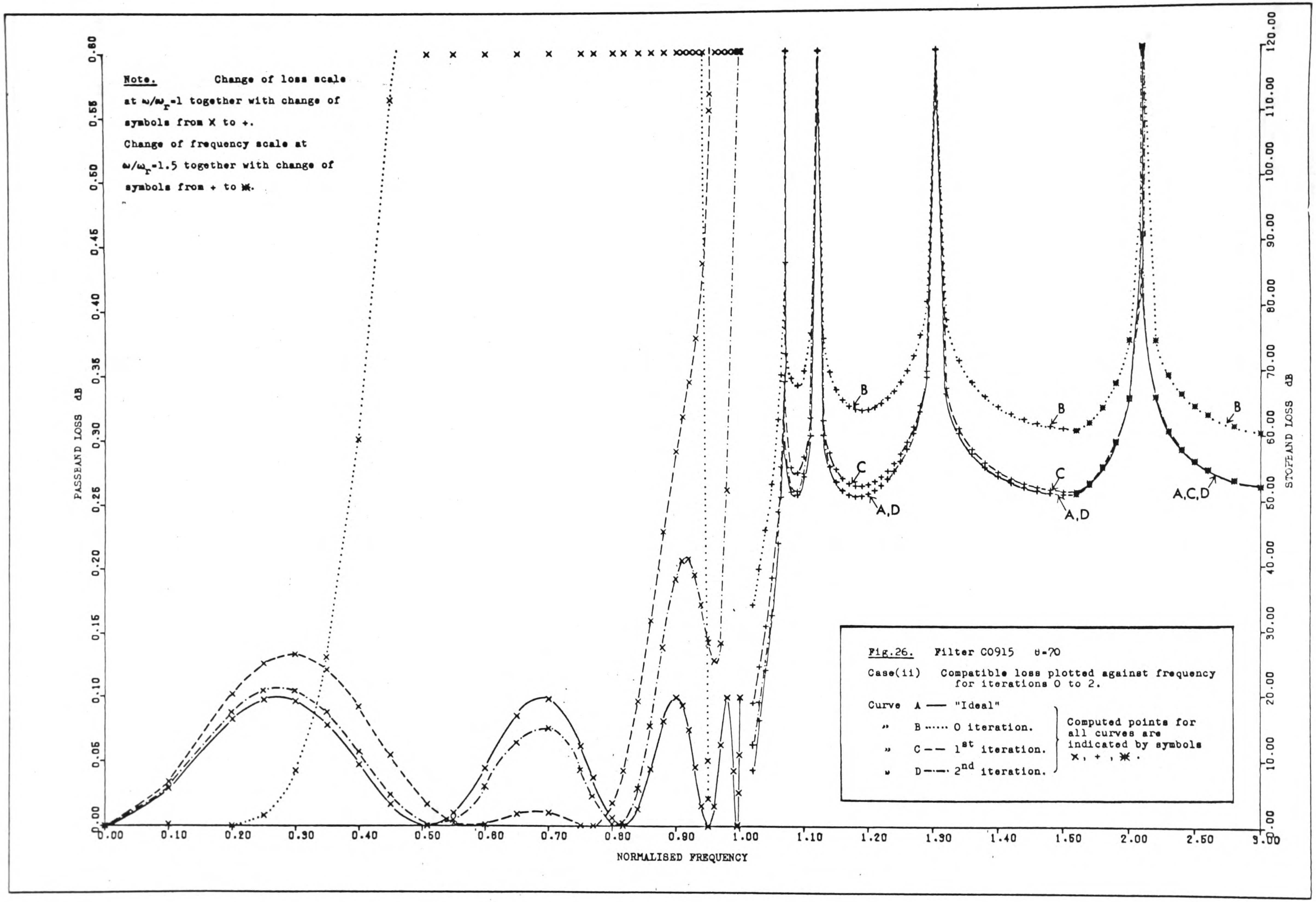


Fig.26. Filter C0915 θ=70
Case(ii) Compatible loss plotted against frequency for iterations 0 to 2.

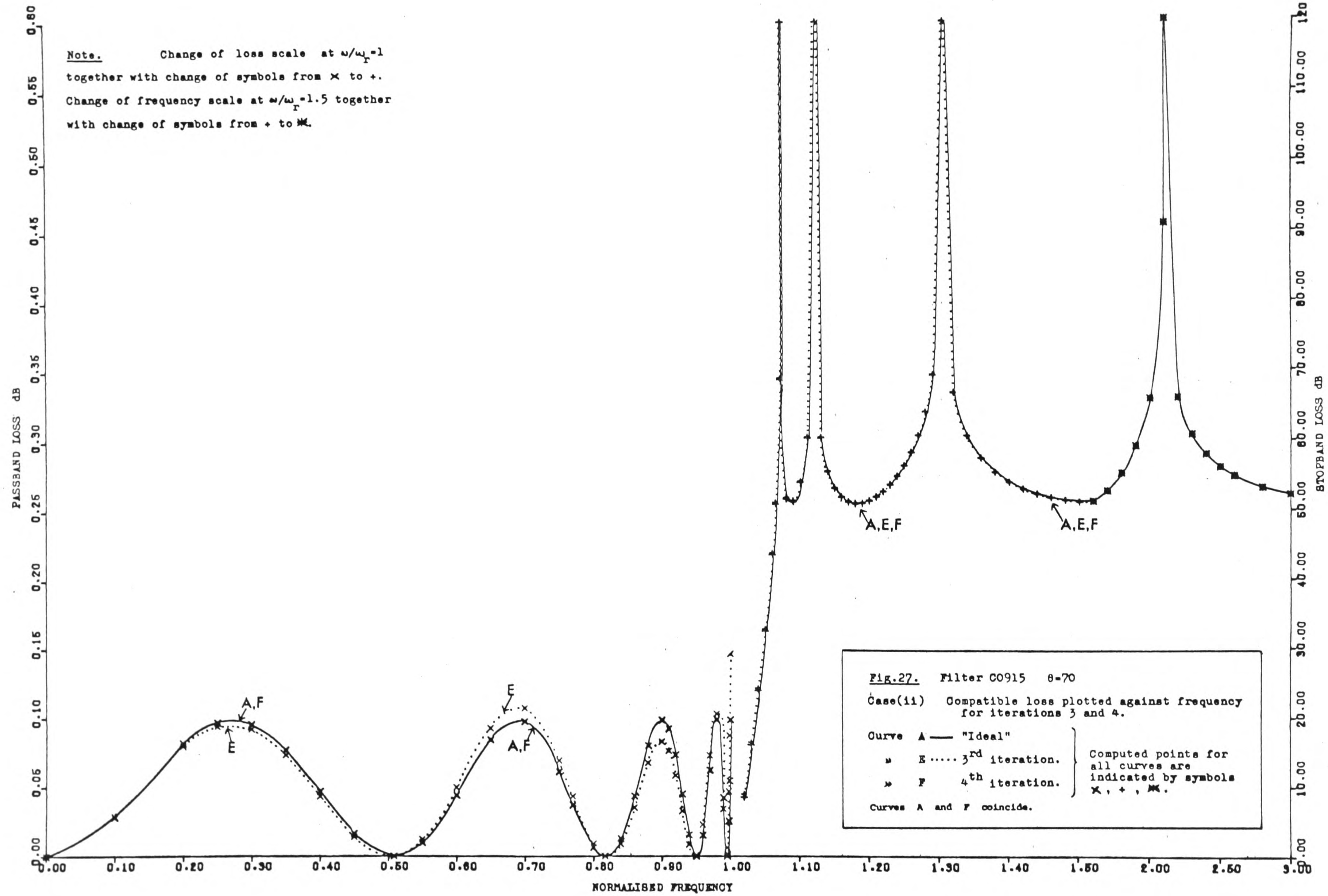


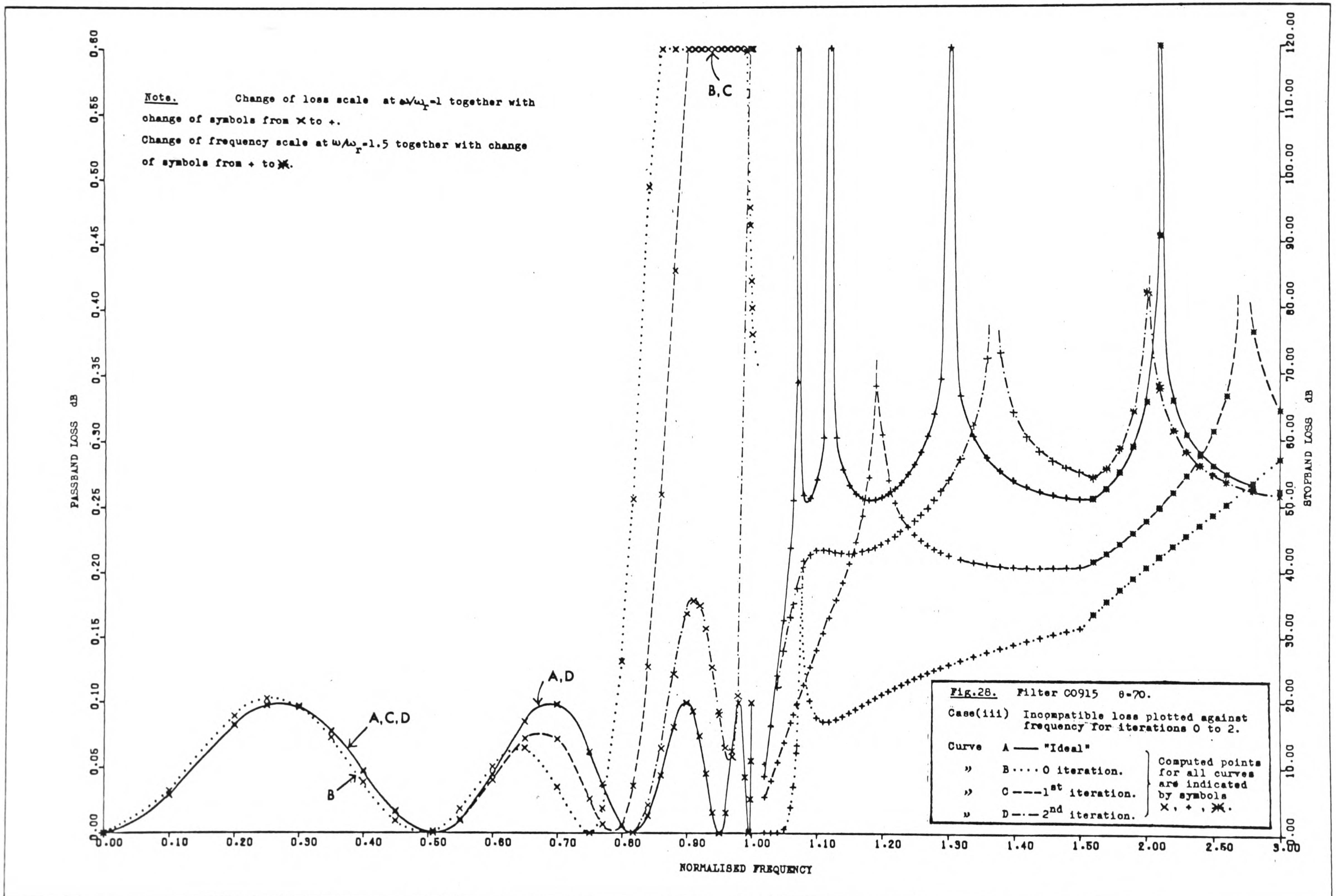
Fig.27. Filter C0915 θ=70

Case(ii) Compatible loss plotted against frequency for iterations 3 and 4.

Curve A —— "Ideal"
" E ····· 3rd iteration.
" F 4th iteration.

Computed points for all curves are indicated by symbols ×, +, ✕.

Curves A and F coincide.


Note. Change of loss scale at $\omega/\omega_r=1$ together with change of symbols from × to +.
Change of frequency scale at $\omega/\omega_r=1.5$ together with change of symbols from + to ⋇.
PASSBAND LOSS dB
STOPBAND LOSS dB
NORMALISED FREQUENCY
A,C,D
B
A,D
B,C
Fig.28. Filter C0915 θ=70.
Case(iii) Incompatible loss plotted against frequency for iterations 0 to 2.
Curve A —— "Ideal"
" B ···· 0 iteration.
" C ---- 1st iteration.
" D -·-· 2nd iteration.
Computed points for all curves are indicated by symbols ×, +, ⋇.

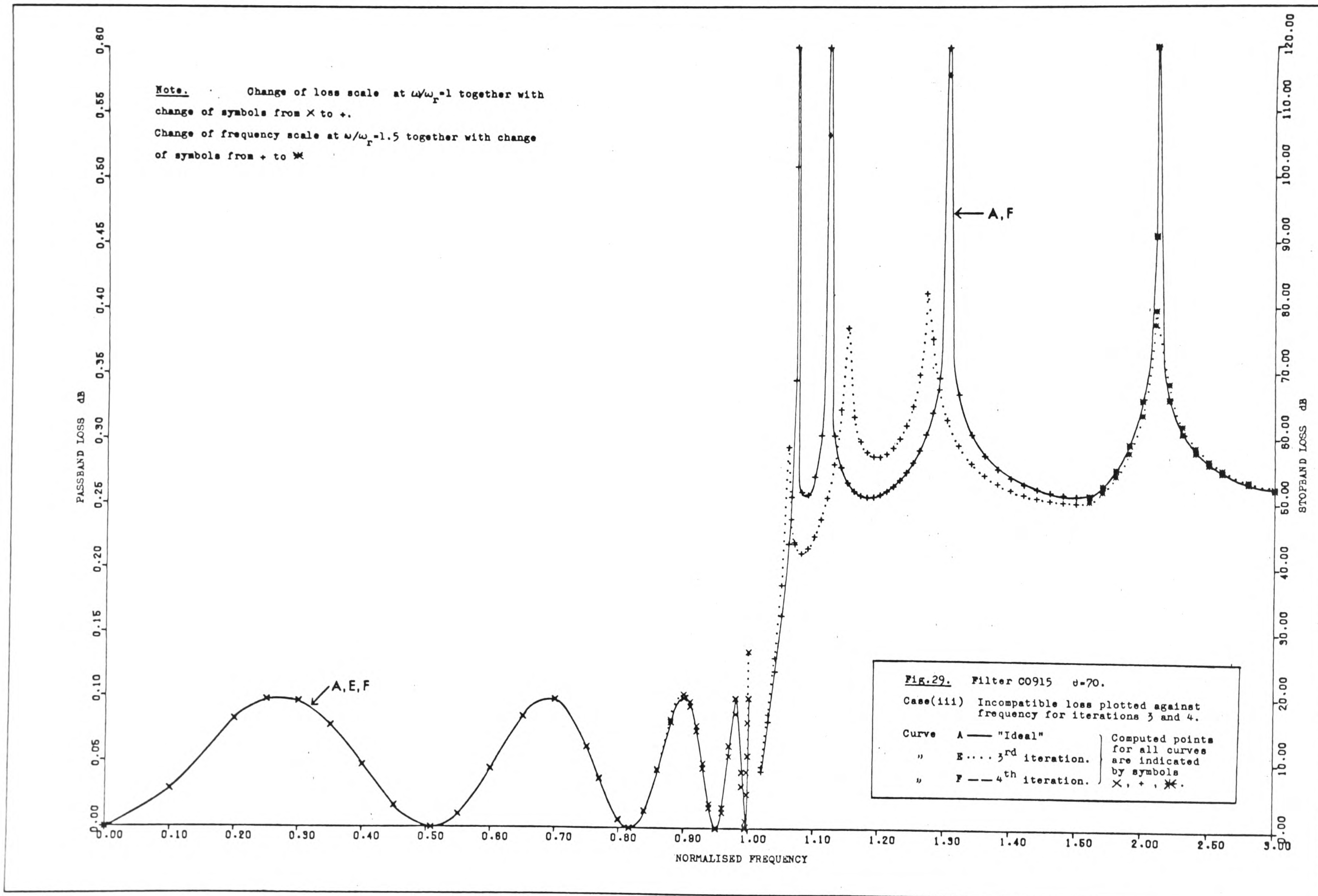


Fig.29. Filter C0915 $\theta = 70$.

Case(iii) Incompatible loss plotted against frequency for iterations 3 and 4.

Curve A —— "Ideal"
" E .... 3rd iteration.
" F —— 4th iteration.

Computed points for all curves are indicated by symbols ×, +, ✳.

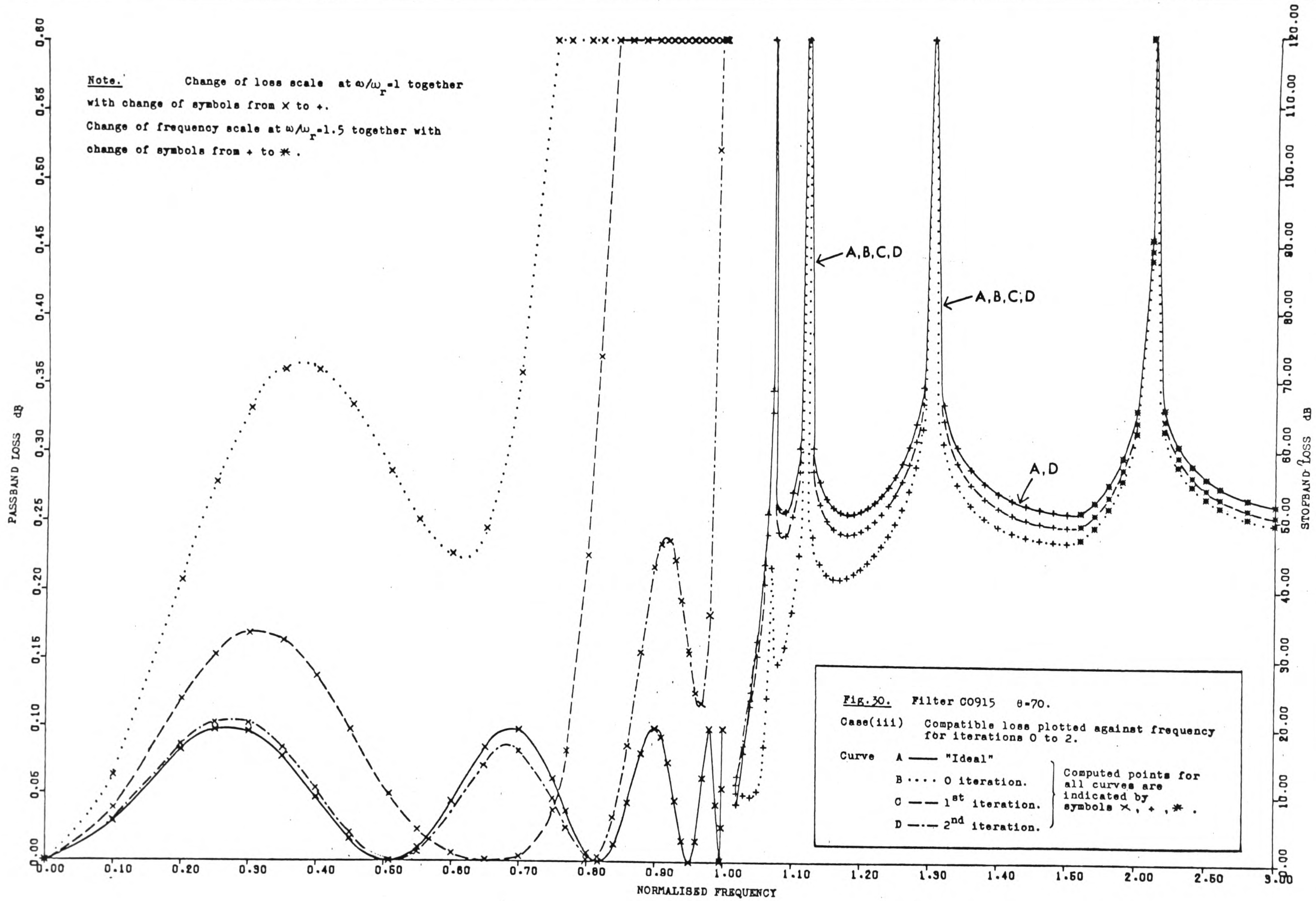


Fig. 30. Filter C0915 θ=70.

Case(iii) Compatible loss plotted against frequency for iterations 0 to 2.

Curve A —— "Ideal"
B ···· 0 iteration.
C — — 1^{st} iteration.
D —·— 2^{nd} iteration.

Computed points for all curves are indicated by symbols ×, +, ✱.

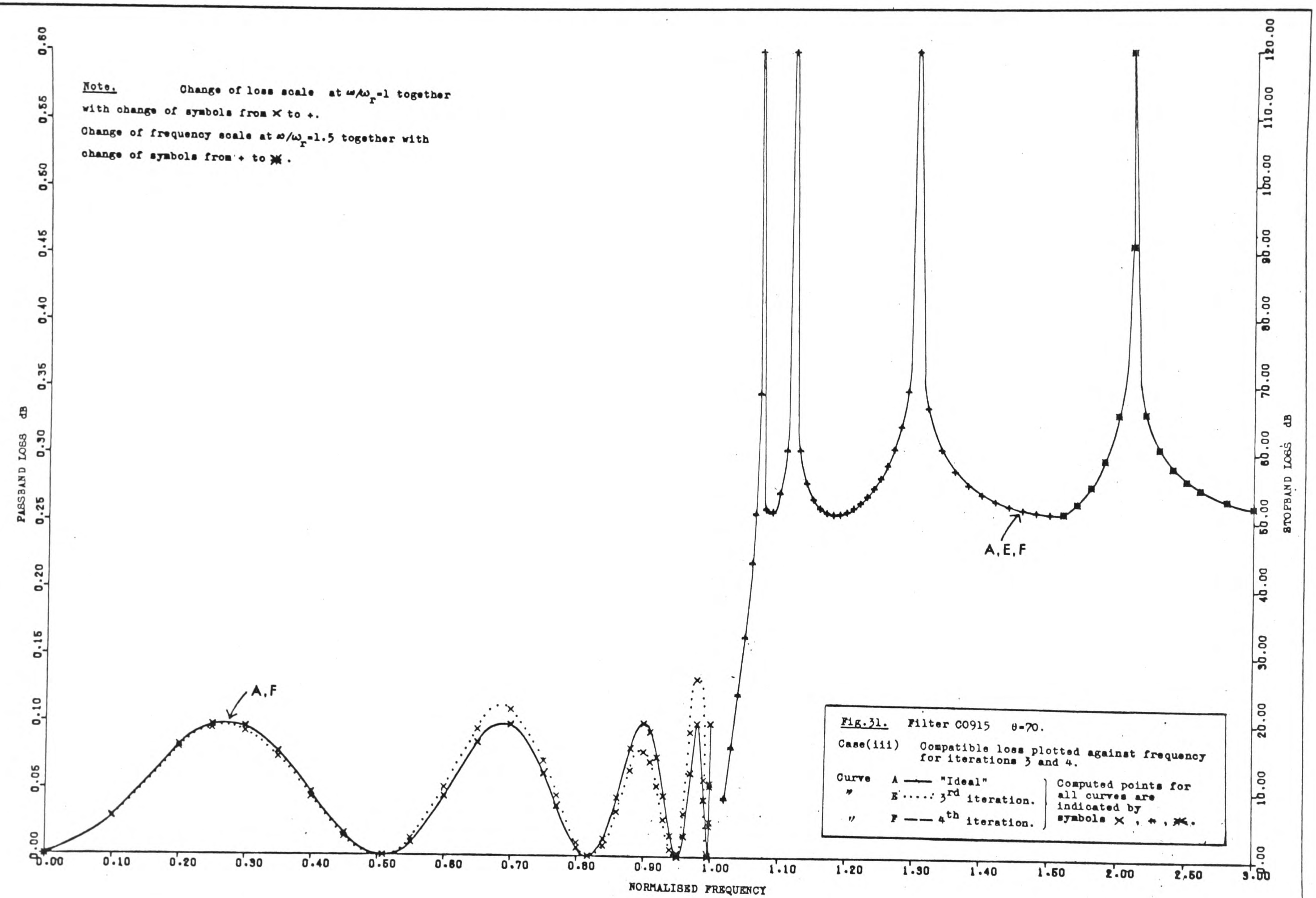


Fig.31. Filter C0915 θ=70.

Case(iii) Compatible loss plotted against frequency for iterations 3 and 4.

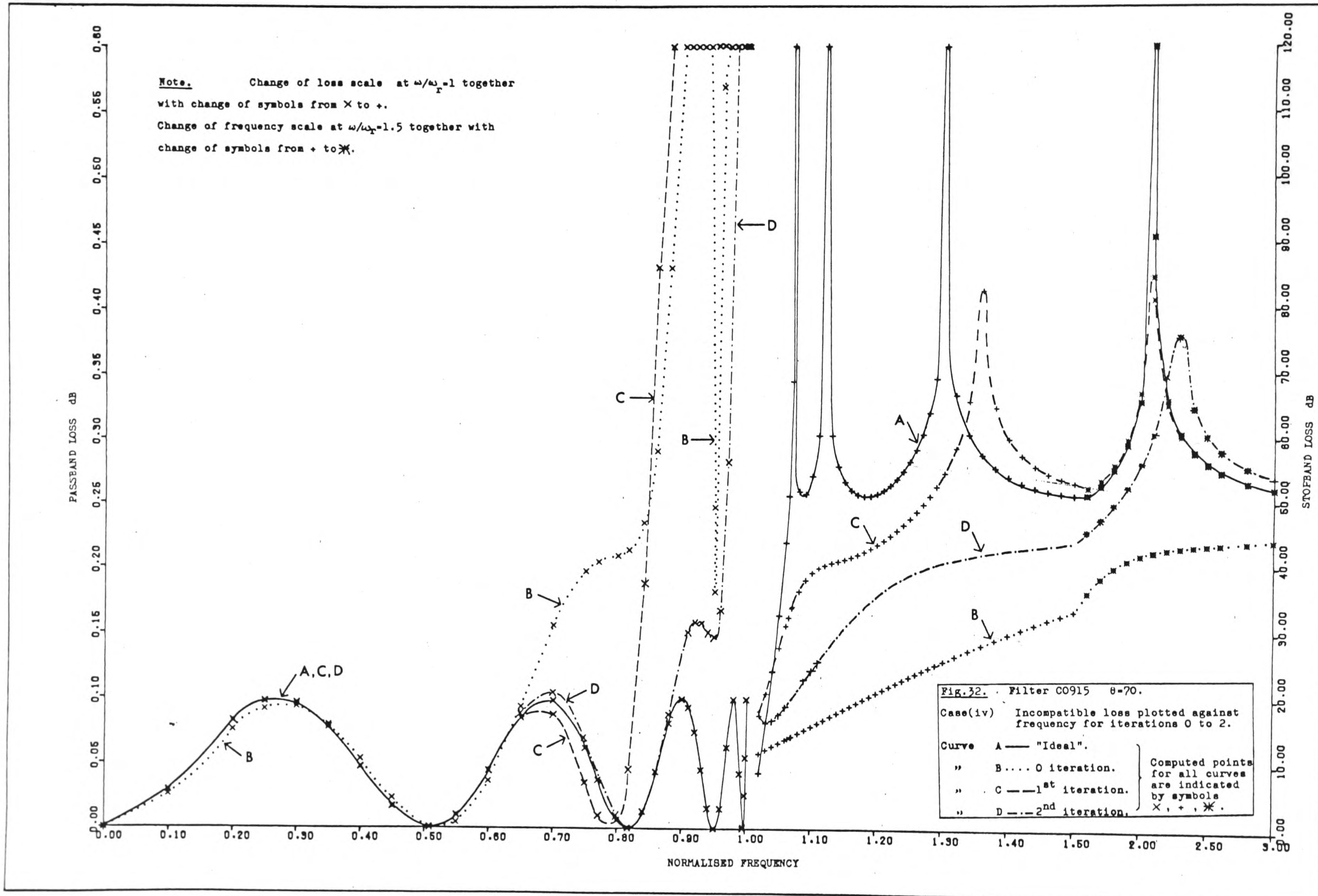
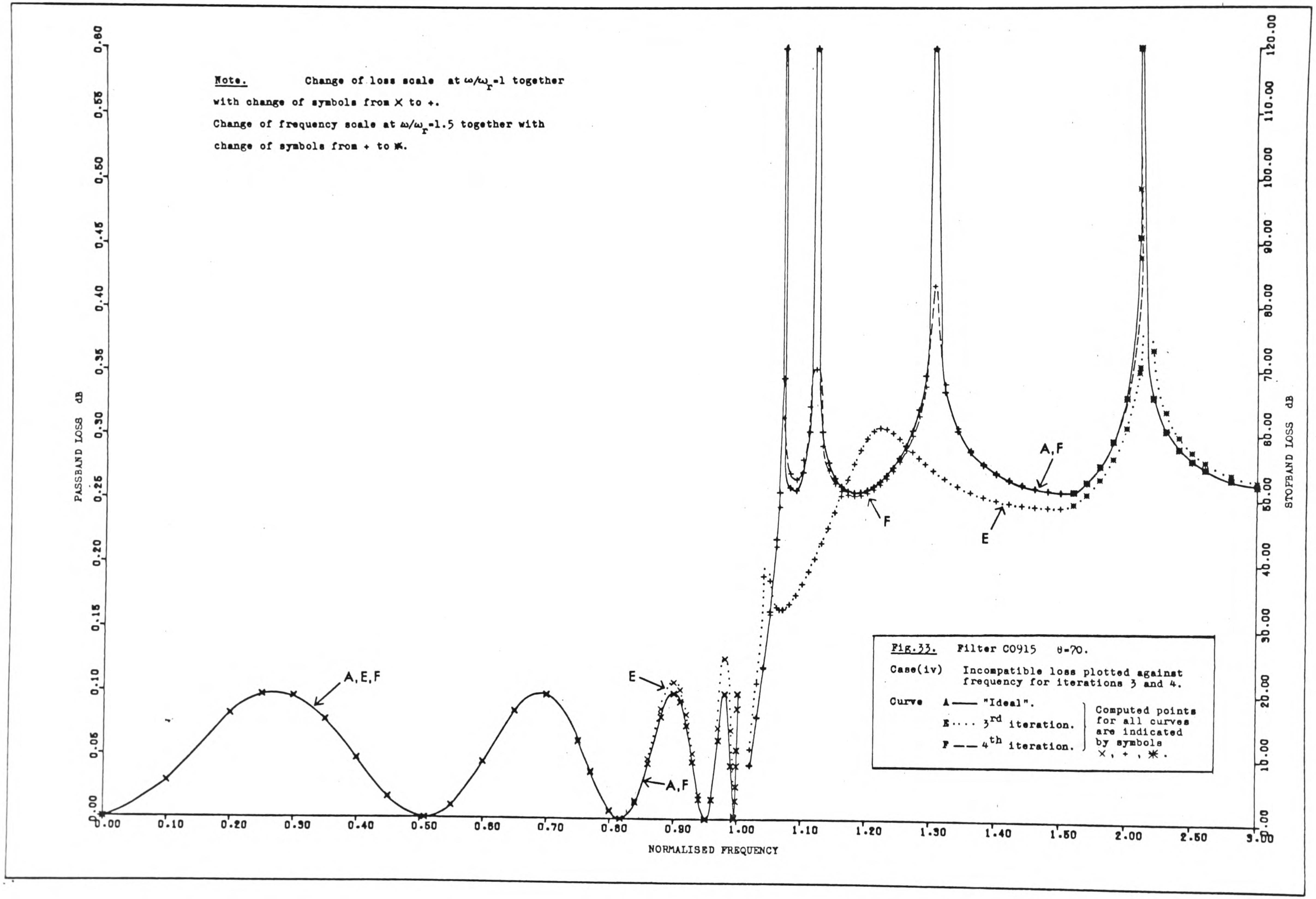
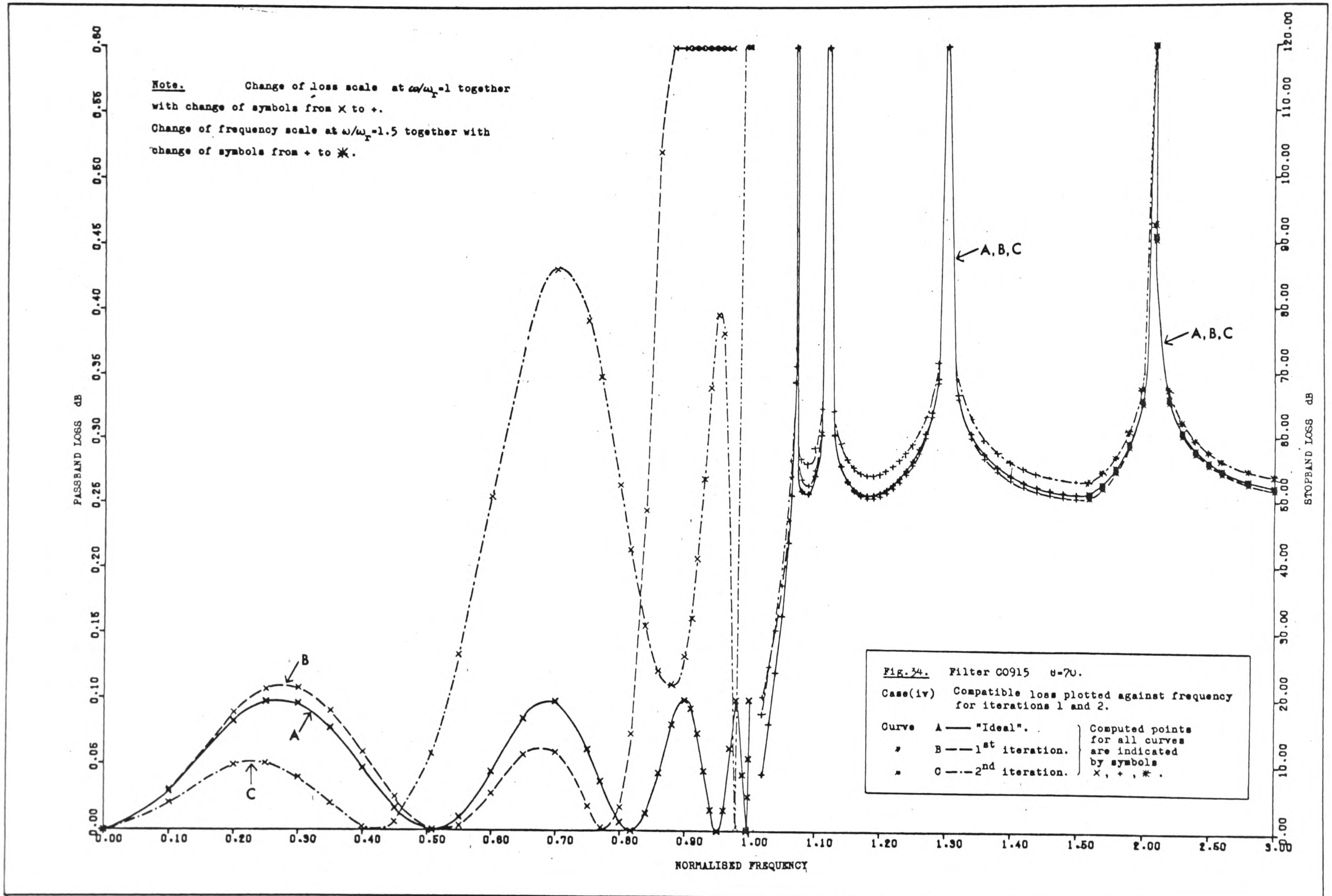


Fig.32. Filter C0915 θ=70.
Case(iv) Incompatible loss plotted against frequency for iterations 0 to 2.


Note. Change of loss scale at $\omega/\omega_r = 1$ together
with change of symbols from × to +.
Change of frequency scale at $\omega/\omega_r = 1.5$ together with
change of symbols from + to ✱.
PASSBAND LOSS dB
STOPBAND LOSS dB
NORMALISED FREQUENCY
A, E, F
E
A, F
F
E
A, F
Fig.33. Filter C0915 θ=70.
Case(iv) Incompatible loss plotted against frequency for iterations 3 and 4.
Curve A —— "Ideal".
E .... 3rd iteration.
F — — 4th iteration.
Computed points for all curves are indicated by symbols ×, +, ✱.



Note. Change of loss scale at $\omega/\omega_r = 1$ together with change of symbols from × to +.
Change of frequency scale at $\omega/\omega_r = 1.5$ together with change of symbols from + to ✱.

Fig. 34. Filter C0915 θ=70.
Case(iv) Compatible loss plotted against frequency for iterations 1 and 2.

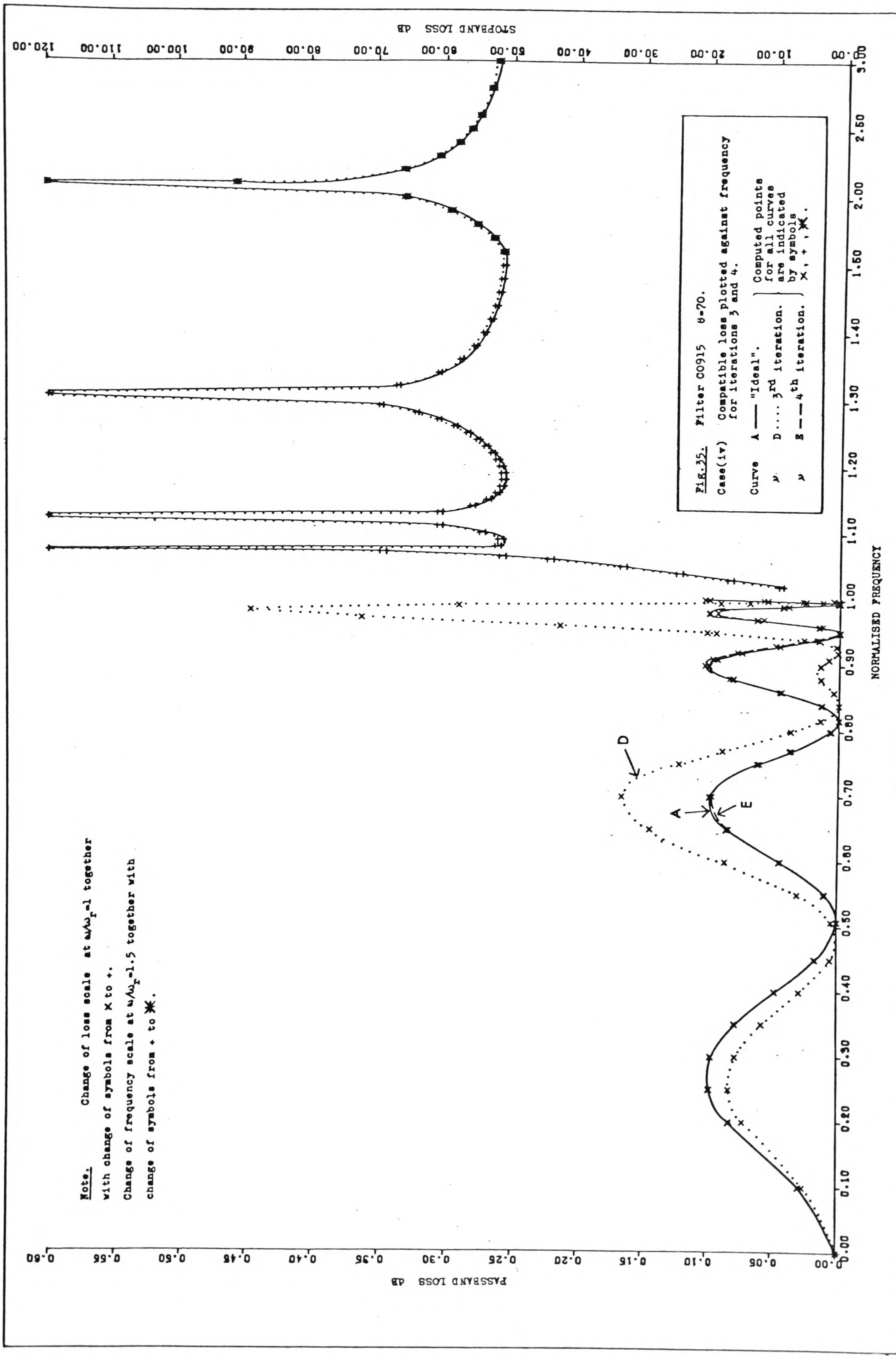


Fig.35. Filter C0915 θ-70.
Case(iv) Compatible loss plotted against frequency for iterations 3 and 4.
Curve A —— "Ideal".
" D .... 3rd iteration.
" E — — 4th iteration.
Computed points for all curves are indicated by symbols ×, +, ✱.
Note. Change of loss scale at ω/ω_r=1 together with change of symbols from × to +.
Change of frequency scale at ω/ω_r=1.5 together with change of symbols from + to ✱.
STOPBAND LOSS dB
PASSBAND LOSS dB
NORMALISED FREQUENCY
A
D
E

5. Comparison with other design methods.

Since the main purpose of the study of Saraga's method was to assess its practical value in filter design in relation to the existing methods, a comparison of the various methods had to be the conclusion of the present investigation. It was decided that, after the results described in section 3 and in section 4.2 had been obtained, a stage had been reached where such a comparison should be attempted, in preference to further developing or generalising the method. Initially a discussion of both the possible criteria and possible methods of comparison is required.

5.1. Choice of criteria for comparison.

5.1.1. Computing accuracy

The most important comparison is that of the accuracy of the element values, because inaccuracy can reach a level where the values are useless and then either higher-accuracy arithmetic is necessary for the computations involved in the design, or a different, "better", method of design has to be used, if one exists. As stated earlier, 20, 30 or more significant figures were often required when a filter was designed by the conventional method in order to produce element values accurate to only 4 significant figures, and the later method by Szentirmai/Bingham reduced the accuracy needs: only single length arithmetic had to be used, for instance 7-8 significant decimal figures on a Ferranti Mercury computer. Later the ICL 1902A was used and in its case single-length arithmetic (Fortran) means 11-12 significant figures which was more than that required for the Szentirmai/Bingham method.

In this investigation of the comparative accuracy of different design methods, these different methods are applied to the same filter function and to the same performance specification. If all methods produce satisfactory element values, two approaches are possible:-

(i) reduce the accuracy of working, and redesign by all the methods; repeat this until at least one method gives results which are inadequate.

(ii) Choose a more "exacting" filter specification using

either (a) a filter of the same order but with, for instance, a steeper transition band and smaller maximum passband loss*,

or (b) a higher-order filter,

and then, employing the same constant accuracy throughout, repeat the process (several times if necessary) until at least one method gives unsatisfactory results.

The particular choice of the criteria for the comparison was affected by the stage reached in the development of Saraga's method. Filters of order $n=7$, 9 had been investigated (section 4.1 and 4.2), also filters of order $n=7$ had been designed and with a little more (programming) work filters of order $n=9$ could also be designed. Furthermore, for the first part of Saraga's method general equations and expressions had been prepared for filters of <u>any</u> order (see section 3.2), but no attempt had been made to prepare general expressions for the later part of the method (see section 3.1 for filter of order $n=9$). Moreover, when some

---

* For a given n-value, only two of the three parameters, maximum passband loss, minimum stopband loss, and the steepness of the transition band, can be chosen independently.

programs had to be redeveloped (as described in section 6) the advantage of a facility for choosing the number of significant digits that should be used for the arithmetic in each design investigation was pointed out to me, and it proved possible to provide such a facility; all the later-written programs depended on it*.

The advantage of using approach (i) was that a stage would be reached when at least one method failed whatever order of filter was used. For convenience a filter of order n=9 could be chosen for the tests. The disadvantage in following approach (ii)(a) was that there was no certainty that it would have been possible to choose a filter of order 9 (or lower) with a sufficiently steep transition band and small passband loss so that at least one method failed and yet all the element values were positive. If approach (ii)(b) is used and if, for example, it had been proved to be necessary to choose a filter of order 11 or higher for the comparison of the methods, the algebra and new programs would have had to be prepared for Saraga's method although no extra preparations would have been necessary for the established methods as the programs for them had been written generally for any odd order of filter. For these reasons it was decided that approach (i) should be used.

Approach (i) would indicate the best method (if any) for the given ninth order filter with "low accuracy" arithmetic i.e.

---

*This was suggested by my supervisor Dr.W. Saraga.

arithmetic in which only a few significant digits were used. Further it was hoped that the following inference would be valid, namely, the method indicated as best in the above circumstances would with higher accuracy arithmetic, i.e. using more significant digits in the arithmetic calculations, also be the "best method" for a higher order filter.

5.1.2. Computer time.

Other criteria for the comparison between the methods were also considered, for instance, instead of the accuracy of the methods, the computer time taken for the different methods could form the basis of the comparison. Although the computer times were small for all the methods, it is of interest to consider some of the work which would be involved in ensuring that such a comparison was fair. It would be necessary to rewrite the programs in very restricted forms so that the essential parts of the processes were compared and not lost among the inessential parts. Thus the final numerical answers only should be printed i.e. without any text to annotate the results and without printing the data or intermediate results; this, under normal conditions, would be inadvisable as such values are useful as checks. To make the program as efficient as possible with respect to time, flexibility has to be avoided. A flexible program might, for instance, require an extra parameter to be read and tested by the computer so as to determine which one of the three interrelated parameters, maximum reflection factor, maximum passband loss or worst return loss*, is given as datum whereas a restricted program would not allow for a choice of parameters. Moreover as it takes longer to change from one subroutine to another than to use instructions consecutively, subroutines would have to be avoided and consequently inelegant programs with many repetitions of similar instructions would be needed. Furthermore the results would depend on the computer and the compiler used and whether only a machine version of the program was used or whether the time of compilation was also included.

5.1.3. Computer Storage.

The methods might be compared with respect to the amount of computer storage that is required by the programs and data. In this case restricted, inflexible programs would again be required. However subroutines should be used as much as possible and repetitions of instructions should be avoided so that the programs are as short as possible. As the storage used for all the methods is small, this comparison is not important either.

---

* The maximum reflection factor ρ, maximum passband loss A_D and worst return loss A_r are related by the equations

$$A_D = 10\log_{10}\left(\frac{1}{1-\rho^2}\right)$$

$$A_r = 20\log_{10}\left|\frac{1}{\rho}\right|$$

Therefore once a value is given for one of the parameters, A_D, A_r, ρ the other two (ignoring the sign of ρ) can be calculated, if required.

5.1.4. Miscellaneous.

Another basis of comparison could be the ease of learning, understanding and applying the methods but it would be difficult to pursue this meaningfully because there are too many different circumstances which could affect the results. For instance, they would depend on who is to use the method (e.g. filter designer, programmer, mathematician, trainee, etc.) and on whether the program or only the data had to be prepared.

5.2. Methods which could be compared.

Basically, there are four methods to be compared:

(1) The conventional filter design method; it starts from $K=N/D$; then D^2-N^2 or $D+N$ or $D-N$ is factorised and H is found using some or all of these factors (in some of them p may be replaced by $-p$); from H and K the impedances Z_O, Z_S are found, and the filter is realised from Z_O and/or Z_S. All the computations are carried out in the p-plane* (Saal).

(2a) Orchard's method. Similar to (1), but with the p-plane partly replaced by the z-plane (the factorisation is performed in the z-plane).

(2b) Szentirmai's/Bingham's method. Similar to (1), but with the p-plane fully replaced by the z-plane.

(3) Musson's/Norek's method. Similar to (1), but the polynomials instead of being used in summation form are used in factor form. It would also be possible to apply this technique to methods (2a) and (2b).

---

* It is worth mentioning that graphical curve approximation by means of templates is always performed in the z-plane.

(4) Saraga's method. This method proceeds from K via factorisation of D+N to lattice impedances Z_a and Z_b, then to Z_O, Z_S. From this stage onwards one of the realisation techniques mentioned above has to be used. This method has so far only been used in the p-plane but it could in principle also be carried out in the z-plane. Furthermore the computational technique of method (3) could be incorporated.

As far as this thesis is concerned it was decided to restrict the investigation to a comparison of Saraga's method with (1) and (2), i.e. without using the Musson/Norek technique.

The fact that Saraga's method does not contain a realisation procedure of its own, might suggest that only the design procedure from the start (with K) to the stage when Z_O, Z_S are obtained should be compared. Unfortunately, from a practical engineering point of view a meaningful comparison of slightly different expressions for Z_O (and/or Z_S) obtained by different methods is not feasible; the significance of such differences can be assessed only by comparing the final filter element values. Thus it is necessary to include network realisation in the comparison and the question arises, whether the realisation for two different methods should be carried out for both methods by the same procedure or whether the realisation should be carried out "in the style" of the design method under consideration i.e. in Saal's, Orchard's and Saraga's method in the p-plane, but in the Bingham/Szentirmai method in the z-plane.

Ideally the Saraga method should be compared with method (1) and both versions of method (2). It is possible to take either of the following two extreme views (or any intermediate position):

(a) Conventional filter design methods were originally developed in the p-plane. In order to overcome the accuracy problems discussed above, the z-plane methods (methods (2a) and (2b)) were - successfully - introduced. Saraga's method constitutes an attempt to overcome the accuracy problems in a completely different way, by using network functions different from those used in the conventional p-plane method. Therefore it could be argued that the basic test of Saraga's method should be, in the p-plane, to determine whether or not Saraga's method is better than the conventional p-plane method, i.e. method 1. In this comparison the same-length-arithmetic e.g. "single-length"* or less accurate arithmetic should be used for both methods.

(b) In contrast to the view taken in (a), it could be argued that, because at present most engineers use the Szentirmai/Bingham method (i.e. the z-plane), Saraga's method (either in the p-plane or converted to the z-plane**) should be compared with the Szentirmai/Bingham method. Again the same-length-arithmetic should be used for both methods.

It will be realised that the comparison suggested under (b) constitutes a much more difficult test for Saraga's method than the comparison suggested under (a).

---

* In this context "single-length" arithmetic may mean for example 11-12 significant figures or 7-8 significant figures depending on the computer or calculator used; here it will be assumed that 11-12 significant figures are used.

** The conversion of Saraga's method from the p-plane to the z-plane has never been attempted and might turn out to be a major undertaking.

This short survey indicates the large variety of comparisons that could be made. However, bearing in mind the duration of a C.N.A.A. registration period and the time that had passed up to this stage, it was decided for the immediate purposes of this thesis to include only limited comparisons that had already been completed. These are between Saraga's method in the p-plane and Orchard's method (method (2a)) and Szentirmai's/Bingham's method (method (2b)). Had more time been available other comparisons would have been explored and embodied in this thesis. In particular, a comparison between Saraga's method and Saal's method,- which, as mentioned above, could be considered as the basic test to prove the practical value of Saraga's method - has not been carried out, as part of the work for the thesis. However, some of the work reported in ref.7 does in fact constitute such a comparison, which clearly showed the advantages of the new method.

5.3. Starting points for the methods actually compared.

The original investigations concerning some aspects of Saraga's method were based on the following argument. In the conventional insertion loss filter design method, the zeros of $D(p)+N(p)$ have to be found so accurately that the incompatibility between $H'(p)$ (derived from these zeros) and the original $K(p) = \frac{N(p)}{D(p)}$ is negligibly small so as not to affect the usability, for the ultimate element realisation, of Z_{O1}, Z_{O2} and Z_{S1}, Z_{S2} derived from $H'(p)$ and $K(p)$. In contrast to this, Saraga claims that with his method fairly inaccurate zeros of $D+N$ are acceptable because compatibility between all relevant parameters is enforced. Therefore, if this argument is correct, Saraga's method should lead to satisfactory results, even if fairly inaccurate zeros are used.

As mentioned in section 4.2.2., for a ninth order filter such inaccurate zeros were roughly simulated by inserting an artificial inaccuracy of 5% in the frequencies of the zeros of $D(p)+N(p)$ or rather in the zeros of $H(p)$ (which are related to those of $D(p)+N(p)$) taken from the Christian and Eisenmann tables ref.11.* Having shown that the method worked satisfactorily under such conditions, it had to be investigated under less artificial conditions and compared with some of the established methods listed in section 5.2.

Saraga's method was compared with the methods (2a) and (2b) of section 5.2, as mentioned before. Methods (2a) and (2b) are applied to filters with equi-ripple loss in the passbands and can be used if either the maximum passband loss A_D or the reflection coefficient ρ has been chosen and all the frequencies of the loss poles which are also the frequencies of the poles of function $K(p)$, have been found by one of the approximation methods mentioned in section 1.3. The relationship between A_D and ρ is (see also footnote, section 5.12) $A_D = 10 \log_{10}\left(\frac{1}{1-\rho^2}\right)$. Sometimes also the parameter $t = \frac{\rho}{\sqrt{1-\rho^2}}$ is used. All design methods can in principle be applied to any specified rational function $K(p)$ which for symmetrical filters has to be an odd function: however Orchard's method and Szentirmai's/Bingham's method are usually used only for

---

* These Tables give primarily the zeros of $H(p)$, all of which lie in the left half p-plane, but it is known, and indicated in these Tables, that, if alternate zeros (considering their imaginary parts) are replaced by their mirror images in the imaginary axis (of the p-plane) then the zeros of $D+N$ are obtained.

equi-ripple cases and the programs used in this thesis are restricted to those cases. The program for Saraga's method was written in a more general way so that it accepts any odd rational function $K(p)$ and not only those $K(p)$ functions which give an equi-ripple loss in the passband. It is therefore necessary in this case to specify the frequencies Ω_{oi}, $i=1,2,..4$ of the zeros and $\Omega_{\infty i}$, $i=1,2,..4$ of the poles and the multiplicative constant C, thus

$$K(p) = Cp \prod_{i=1}^{4} \left(\frac{p^2+\Omega_{oi}^2}{p^2+\Omega_{\infty i}^2} \right) .$$

The Christian and Eisenmann tables (see ref.11) were extremely useful as they gave all the data required for each of the methods. It was assumed that the data were "equally accurate" for all the methods. The data and configuration of the filter used to compare the methods are shown in fig.36.

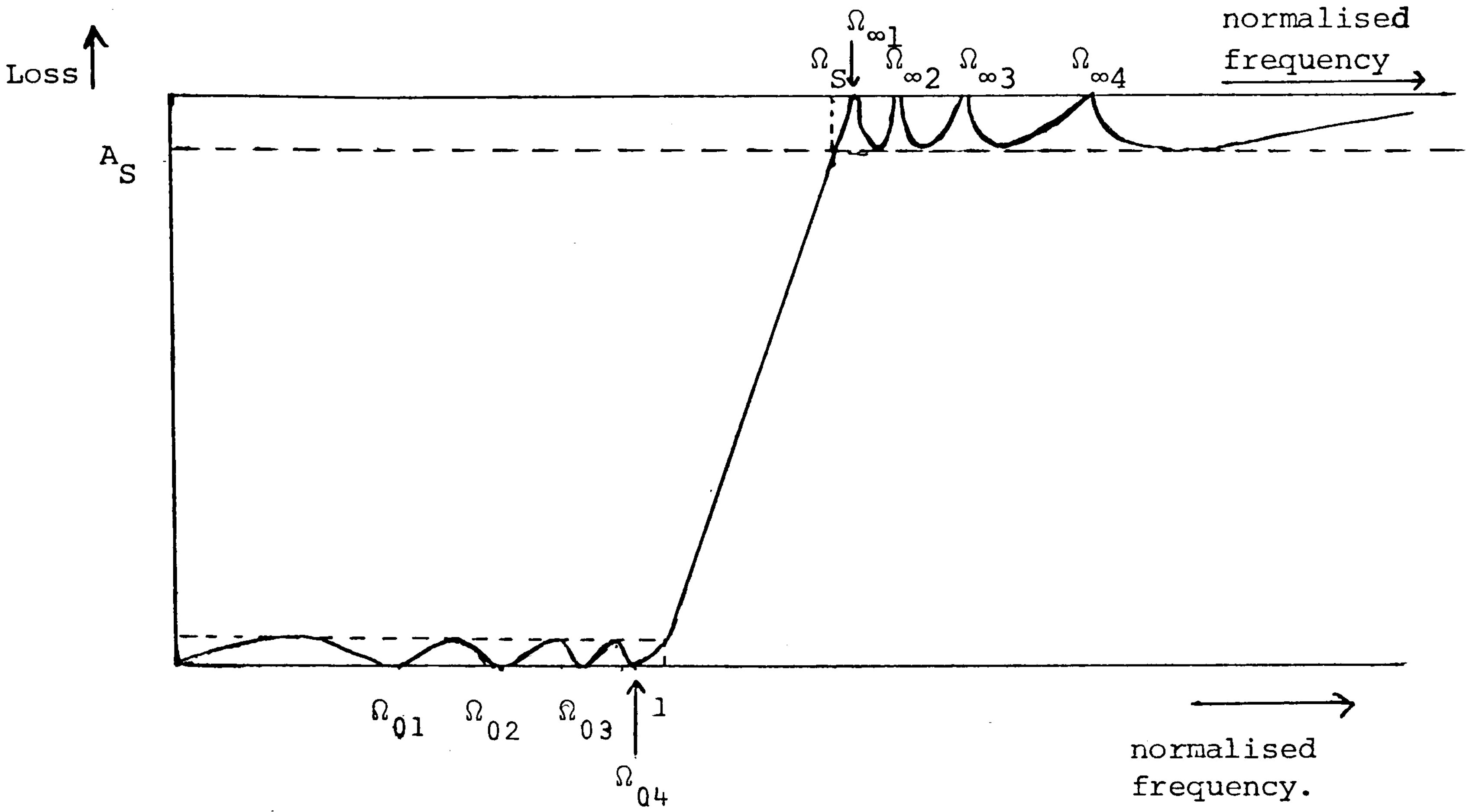
5.4. Results.

5.4.1. Arrangement of results.

As explained in the previous section, different data were taken for the different filter design methods. The data shown in fig.37 were taken for both Orchard's design method and for Szentirmai's/ Bingham's method whereas the data in fig.38 were taken for Saraga's method.

For the last part of each method, i.e. the realisation, the impedances Z_{O1}, Z_{S1}, Z_{O2} and Z_{S2} are used. The filter can be realised from each impedance separately (except the input and output shunt capacitors: the input shunt capacitor cannot be realised from the output short-circuit impedance, and vice versa). Thus, for each method four values are obtained for every circuit element (apart

Fig.36. Data and configuration of filter CO915 θ=70



A_S=51.01 dB

Ω_S=1.064178

ρ=15%

C=59.113259

i	Ω_{oi}	$\Omega_{\infty i}$
1	0.508089	1.0691285656
2	0.817124	1.1193168508
3	0.950739	1.3023470398
4	0.995370	2.0944721998

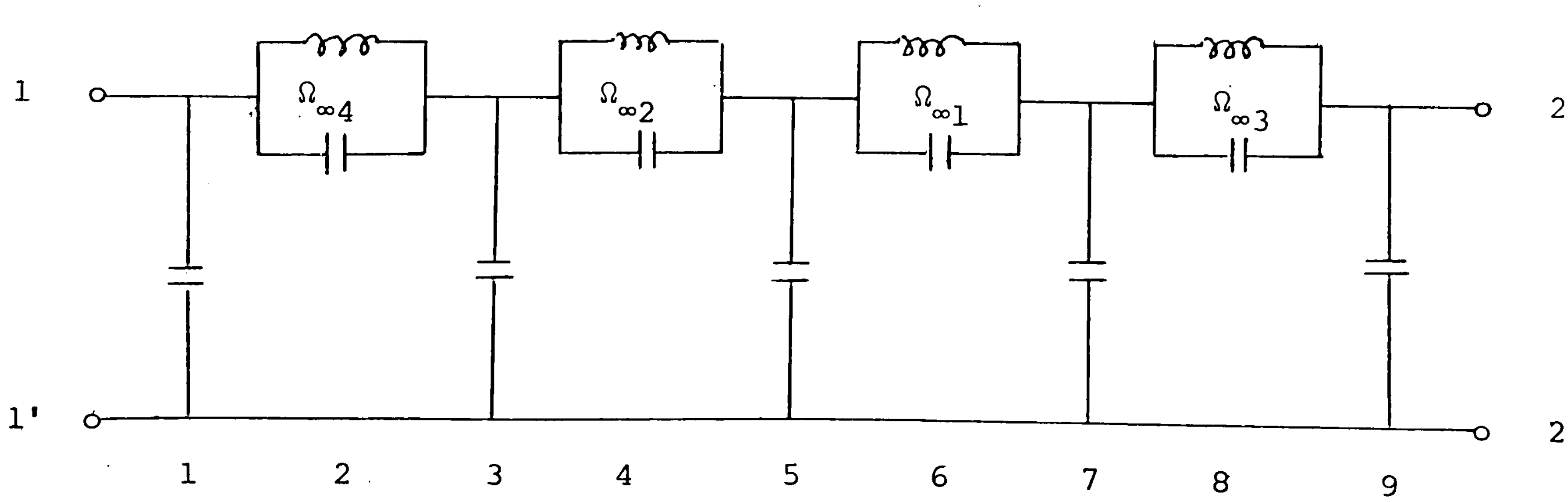


Fig.37. Data used for Orchard's method and Szentirmai's/Bingham's method.

$\rho = 15\%$

i	$\Omega_{\infty i}$
1	1.0691285656
2	1.1193168508
3	1.3023470398
4	2.0944721998

Fig.38. Data used for Saraga's method.

$C = 59.113259$

i	Ω_{oi}	$\Omega_{\infty i}$
1	0.508089	1.0691285656
2	0.817124	1.1193168508
3	0.950739	1.3023470398
4	0.995370	2.0944721998

from the end shunt capacitors for which only three values are obtained). For each element the results are given in two rows, the higher row containing the element values realised from port 1, i.e. from the impedances Z_{O1} and Z_{S1}, and the lower row containing the element values realised from port 2, i.e. from impedances Z_{O2}, Z_{S2}. Moreover for each method the values realised from the open-circuit impedances are given in the left-hand column and those from the short-circuit impedances are given in the right-hand column. This is all indicated by the positioning of the symbols for the impedances, Z_{O1}, Z_{S1}, Z_{O2}, Z_{S2}, at the top of the columns. The inductor values are not shown because the errors in their values are of the same order as those for the associated capacitors with which they resonate at accurately known frequencies.

The element values obtained when using "single-length" arithmetic (11-12 significant figures) are shown in table 6, and those obtained when the results of each arithmetic operation are truncated to 8 significant figures are shown in table 7. For the "single-length" calculations three sets of results were obtained for Saraga's method for the case in which the values are compatible. The reason for three sets of results occurring is rather accidental but it illustrates the effect of small numerical inaccuracies. Therefore all the columns have been retained and a detailed explanation is given in appendix 5.

Table 6. Comparison of results for filter fig.36 using "single-length" arithmetic.

Method	Orchard		Szentirmai/Bingham		Saraga: incompatible		Saraga:	compatible			
	z_{O1} z_{O2}	z_{S1} z_{S2}	z_{O1} z_{O2}	z_{S1} z_{S2}	z_{O1} z_{O2}	z_{S1} z_{S2}	z_{O1} z_{O2}	z_{O1} z_{O2}	z_{S1} z_{S2}	z_{O1} z_{O2}	z_{S1} z_{S2}
c_1	1.0404773090 1.0404842623	1.0404773089	1.040477309011862 1.040477269823725	1.040477309201008	1.0404733583 1.0404579523	1.0404733582	1.0404733582 1.0404928272	1.0404733582 1.0404583615	1.0404733582	1.0404733582 1.0405011308	1.0404733583
c_2	0.18391080046 0.18391091915	0.18391080033 0.18391080034	0.1839108003805449 0.1839107997893734	0.1839108003332512 0.1839107994965161	0.18391090991 0.18391064718	0.18391091017 0.18391090962	0.18391091011 0.18391124277	0.18391091010 0.18391065385	0.18391091013 0.18391091004	0.18391091013 0.18391138402	0.18391090994 0.18391090962
c_3	1.3539581920 1.3539526295	1.3539584800 1.353958229	1.353958230217955 1.353958262639594	1.353958230378111 1.353958232444369	1.3539569071 1.3539689332	1.3539563504 1.3539565320	1.3539559944 1.3539408480	1.3539562149 1.3539686059	1.3539564152 1.3539565275	1.3539563262 1.3539341608	1.3539566702 1.3539565301
c_4	1.1920623762 1.1920691041	1.1920620174 1.1920631968	1.192063206879367 1.192063173817388	1.192063210226308 1.192063203517932	1.1920597426 1.1920493384	1.1920631158 1.1920623637	1.1920683743 1.1920789068	1.1920658655 1.1920496672	1.1920629080 1.1920623874	1.1920639353 1.1920859030	1.1920617694 1.1920623786
c_5	0.84026800048 0.84025112095	0.84025306111 0.84025244920	0.8402524468401679 0.8402524519406142	0.8402524373159394 0.8402524385164724	0.84026620664 0.84025386930	0.84025080445 0.84025094261	0.84019393130 0.84024722657	0.84021795297 0.84025380194	0.84025069005 0.84025093835	0.84024012178 0.84024565815	0.84025108745 0.84025093412
c_6	1.7146367717 1.7150142157	1.7150154764 1.7150134229	1.715013352513494 1.715013406268184	1.715013416265094 1.715013425112772	1.7146642601 1.7150114315	1.7150115405 1.7150132877	1.7163601302 1.7150154883	1.7157924796 1.7150115222	1.7150123139 1.7150132545	1.7152642720 1.7150165076	1.7150146064 1.7150132544
c_7	1.0105822362 1.0102804745	1.0102798264 1.0102804985	1.010280550442672 1.010280503367226	1.010280505535519 1.010280501868408	1.0105614015 1.0102797882	1.0102799404 1.0102797054	1.0092001852 1.0102796422	1.0096545947 1.0102797706	1.0102799958 1.0102797158	1.0100776718 1.0102795882	1.0102794984 1.0102797163
c_8	0.73989370101 0.73998860405	0.73998876935 0.73998860455	0.7399885892082238 0.7399886038474505	0.7399885992490453 0.7399886034545488	0.73990099704 0.73998969358	0.73998968094 0.73998969543	0.74032756541 0.73998969507	0.74018509797 0.73998969512	0.73998962909 0.73998969461	0.74005317410 0.73998969717	0.73998971729 0.73998969470
c_9	0.68340187494 0.68371931093	0.68371931061	0.6837192614038940 0.6837193109313375	0.6837193112587556	0.68341785166 0.68371500552	0.68371500524	0.68485217377 0.68371500527	0.68437347902 0.68371500517	0.68371500534	0.68392800371 0.68371500493	0.68371500536
Column number	(i)	(ii)	(iii)	(iv)	(v)	(vi)	(vii)	(viii)	(ix)	(x)	(xi)

Table 7. Comparison of results for filter fig.36 using 8 figure arithmetic.

Method	Orchard		Szentirmai/Bingham		Saraga: incompatible		Saraga: compatible	
	z_{O1} z_{O2}	z_{S1} z_{S2}	z_{O1} z_{O2}	z_{S1} z_{S2}	z_{O1} z_{O2}	z_{S1} z_{S2}	z_{O1} z_{O2}	z_{S1} z_{S2}
c_1	1.0404775000 1.0581708000	1.0404777000	1.04047790000905 1.03983340000 5392	1.040481800009445	1.0404705000 1.1112969000	1.0404705000	1.0404703000 1.0375007000	1.0404705000
c_2	0.18391080000 0.18416462000	0.18391059000 0.18390895000	0.183910860001 5606 0.183900070001633	0.183909780002491 0.183901700002058	0.18391136000 0.18435148000	0.18391112000 0.18390951000	0.18391217000 0.18385922000	0.18391129000 0.18391284000
c_3	1.3518213000 1.3398268000	1.3544473000 1.3539992000	1.353959600004372 1.354482299997101	1.353963900007926 1.354048499997162	1.3535423000 1.2989630000	1.3546215000 1.3539627000	1.3516603000 1.3563454000	1.3547132000 1.3539236000
c_4	1.2183794000 1.2074384000	1.1896587000 1.1919800000	1.192059900000259 1.191517200000618	1.192129200004075 1.192010600003357	1.1974361000 1.2591208000	1.1890370000 1.1920083000	1.2140205000 1.1895872000	1.1887283000 1.1921205000
c_5	0.65564569000 0.83677734000	0.84171834000 0.84023443000	0.840277060005169 0.840376210006070 3	0.840125250004348 3 0.840145010006381 1	0.78898684000 0.82477897000	0.84146390000 0.84027971001	0.69963524000 0.84086178000	0.84151484000 0.84031651000
c_6	4.7150515000 1.7172014000	1.7187979000 1.7151897000	1.714430500010877 1.714939099998247	1.715151099997314 1.715292900000093	2.8271803000 1.7247163000	1.7209460000 1.7150898000	4.1989401000 1.7147206000	1.7218710000 1.7149861000
c_7	0.72405976000 1.0101908000	1.0087437000 1.0102479000	1.010748900007172 1.010284600007764	1.010354500001824 1.010254100008438	0.70525586000 1.0099278000	1.0088484000 1.0102463000	0.69646395000 1.0102580000	1.0087575000 1.0102619000
c_8	0.62285021001 0.73999023000	0.74040893000 0.73998919001	0.739840690002893 0 0.739988799999991 8	0.739922180000576 1 0.739979650003078 8	0.70029030000 0.73999641000	0.74027769001 0.73999647000	0.63692236000 0.73999680000	0.74028665001 0.73999518000
c_9	1.1574277000 0.68371919000	0.68371974000	0.683226320004905 5 0.683719870001368 6	0.683726770002976 9	1.0406162000 0.68371057000	0.68371019000	1.1402937000 0.68370980000	0.68371021000
Column number	(i)	(ii)	(iii)	(iv)	(v)	(vi)	(vii)	(viii)

5.4.2. Discussion of the results.

5.4.2.1. General discussion.

On comparing the results, it is evident that those obtained by Szentirmai's/Bingham's method are better than those obtained by means of either Orchard's or Saraga's method. This was to a certain extent expected (as discussed above) because Szentirmai's/Bingham's method uses the z-plane not only in that part of the design method where Saraga's method tries a different approach to overcome the inaccuracy problem but in the realisation part whereas the same network realisation in the p-plane was used in the investigation of both Orchard's and Saraga's methods. Therefore the Szentirmai/Bingham results will not be considered for the main comparisons which will consist of comparisons of the results from Orchard's and Saraga's methods. Nevertheless a few comments will be made on some aspects of the results obtained by Szentirmai's/Bingham's method.

For the comparisons of Saraga's and Orchard's method the network element values are obtained by identical realisation techniques in the p-plane from Z_O, Z_S. Consequently the comparison concerns the relative advantages of the use of the z-plane for part of the design procedure (Orchard's method) and "incompatibility-avoidance" (Saraga's method).

This comparison however reveals no decisive differences for the particular example presented in tables 6 and 7. Therefore it would appear that following the discussion in section 5.1.1., an attempt should be made to produce more significant differences by either designing a more difficult filter or by further reducing the accuracy (however the comparison of the results of single length and 8 figure arithmetic is disappointing in this respect as no clear tendency can be discerned). The lack of significance of the differences between the results obtained by Saraga's and Orchard's methods (and also some unexplained obscurities in the results obtained by Szentirmai's/ Bingham's method) will be discussed in the following section.

It is of course possible that a different example might have given much more decisive results. However in the absence of such further examples and as a result of the knowledge acquired from a consideration of the investigations as a whole it seems imperative for any future work to compare first of all Saraga's and Saal's method. In this way it would be possible to establish without any doubt whether or not Saraga's method constitutes an improvement on Saal's method in terms of the comparison criteria discussed above. Only if the answer is definite and positive, does it become meaningful to compare Saraga's method with Orchard's and Szentirmai's/Bingham's method.

5.4.2.2. Detailed comparisons.

5.4.2.2.1. Numerical techniques.

In the preceding section a "global" assessment has been given of the results obtained by designing the filter of fig.36. Before it is possible to give a detailed comparison, it is necessary to define the three versions of the numerical techniques used for the different comparisons. The reason, for not using throughout the technique that was the quickest to apply, will also be explained.

The three techniques for the comparison of two associated values consisted of finding

(i) the maximum number of significant figures which agreed when the numbers were rounded,

(ii) the difference between them,

(iii) the "relative difference" between them. In this case the difference itself (and not the absolute difference)* was divided by the number that was likely to be the element value (usually taken to 4 significant figures and judged by looking at all the values of the particular element calculated by all the methods including Szentirmai's/Bingham's from all the impedances).

As an example of the manner in which these techniques were applied consider the values of C_6, obtained from Z_{O1}, Z_{O2} using Orchard's method, which are given in table 6. These are 1.7146367717 and 1.7150142157 respectively. The application of technique (i) gives a maximum agreement of 4 significant figures as both figures become 1.715 (when rounded). When the comparison is between the values obtained from Z_{O1}, Z_{O2} (in this sequence) the difference by technique (ii) is taken as -3.77444×10^{-4} since the second value (i.e. that calculated from Z_{O2}) is subtracted from the first value (i.e. that calculated from Z_{O1}). The "relative difference" by technique (iii) is taken as

$$\frac{-3.77444\times10^{-4}}{1.715} \simeq -2.201\times10^{-4} .$$

*The difference itself was chosen instead of the absolute difference as it is interesting to see whether the error is in the same direction all the time or whether it changes its direction.

The numerical technique (i) is quickest to apply, and is useful if the results agree to a large number of figures for one method (Szentirmai/Bingham) and to fewer figures for the others, but not useful when the comparison values are close. It also has to be used with great care; consider the values for C_5, calculated from Z_{01}, Z_{02} by Szentirmai's/Bingham's method and given in table 6, which are

0.8402524468401....

0.8402524519406....

When these are rounded to 8 significant figures they both become 0.84025245; when rounded to 7 significant figures they become 0.8402524 and 0.8402525 respectively; when these values given in the table are rounded directly to 6 significant figures they both become 0.840252. This is somewhat unsatisfactory as they can be said to agree when rounded either to 6 or to 8 significant figures but not when rounded to 7 significant figures. In this case the figures are said (by me) to agree to 8 significant figures.

Another example of the way in which comparisons by technique (i) can be misleading is seen when 1.516 and 1.517 are compared with each other and when 1.514 and 1.515 are compared. The former pair is said to agree to 3 significant figures (1.52) but the latter pair is said to agree to only 2 significant figures (1.5) although in both cases the differences are -0.001, i.e. the inaccuracies are about the same size.

Comparisons using techniques (ii) and (iii) are more useful but take longer to perform.

5.4.2.2.2. Some comparisons in terms of technique (i).

The results for the Szentirmai/Bingham method are so much better than the others that, as mentioned earlier, they are excluded from the later detailed comparisons. The four values given for each element in table 6 ("single-length" arithmetic) for the Szentirmai/Bingham method agree to 7 significant figures whilst for the other methods there is about 6 figure agreement for the elements calculated from Z_{S1} and Z_{S2} but some elements calculated from Z_{O1} and Z_{O2} agree to only 2 significant figures. The corresponding results for Szentirmai's/Bingham's method in table 7 ("8-figure" arithmetic) agree to at least 3 significant figures with the exception of the 2 figure agreement of the values for element C_9. However there is no agreement for some of the element values calculated from Z_{O1} and Z_{O2} by the other methods with "8-figure" arithmetic (although the values calculated from Z_{S1} and Z_{S2} agree to at least 2 significant figures).

Usually the lower order impedances give better results: this effect is confirmed in the given example, in which the Z_{S1} and Z_{S2} impedances are of lower order than Z_{O1} and Z_{O2}, for the p-plane methods but not for the z-plane method.

On applying technique (i) to tables 6 and 7, it is seen that the results in table 6 (where "single-length" arithmetic i.e. 11/12 significant figures were used for the calculations) are about 3 significant figures better than those in table 7 (where 8 significant figures were used in the calculations), and this seems reasonable.

Some of the results are unexpected and one of the most surprising is the inconsistency in the rates of losing accuracy in calculating the element values from the different impedances. Before this can be demonstrated, it is necessary to recall the realisation process. The element values are calculated from Z_{O1} in the sequence $C_1, C_2, C_3, \ldots C_8, C_9$; i.e. the value of C_1 is calculated first, then the element C_1 is removed from Z_{O1} (actually from the admittance $Y_{O1} = 1/Z_{O1}$) and the resulting impedance Z_1, say, is less accurate than Z_{O1}. The next element C_2 is found from this, then the resonant elements C_2, L_2 are removed from Z_1 to yield a new impedance which is less accurate still and the process is repeated until all the elements have been found. When impedance Z_{O2} instead of Z_{O1} is used, the elements are calculated in the opposite sequence i.e. C_9 first, then C_8, and so on until C_1 has been found: therefore C_9 is more accurate than C_8 and C_8 than C_7, etc. As an example of the surprising result that the set of element values calculated from Z_{O2} are better overall than those calculated from Z_{O1}, consider the comparison of the results from Saraga's compatible method with the results from Szentirmai's/Bingham's method when "8-figure" arithmetic is used (in this comparison the Szentirmai/Bingham results can be regarded as "accurate"). For Z_{O1} the elements C_1, C_2 agree to 5 significant figures; C_3 to 3; C_4 to 2; C_5, C_6 to 0; C_7 to 1 (only just: 1.01... and .69...); C_8 to 0; C_9 to 1 (only just: .68... and 1.14...), whereas for Z_{O2}, elements C_9 and C_8 agree to 4 significant figures; C_7 to 5; C_6 to 4; C_5 to 2; C_4 to 3; C_3 to 2; C_2 to 4; and C_1 to 3 significant figures. Similar observations can be made for the other p-plane methods. The different rates of losing accuracy when the elements are calculated from Z_{O1} and from Z_{O2} is not yet understood, but this point will not be pursued here.

Surprise is also caused by the greater similarity between the element values, of a particular element, calculated by the different p-plane methods from the same impedance than between the element values calculated by the same method from the four different impedances when 8 figure arithmetic is used for the calculations. Such similarities of element values are particularly noticeable when the values are "wrong", for instance when Z_{01} is used; this is demonstrated in table 8. The table consists of some values taken from table 7 but with the figures rounded to 4 decimal places. The entries for the column headed "rough approximation to correct value" are obtained by comparing the results calculated from Z_{S1}, Z_{S2}, Z_{02} for all the p-plane methods and taking a value (to 2 decimal places) near to most of them.

Table 8 Element values calculated with 8-figure arithmetic from Z_{01}.

Method / Element	Orchard	Saraga		Rough approximation to "correct" value (taken from table 6).
		incompatible	compatible	
C_5	0.6556	0.7890	0.6996	0.84
C_6	4.7151	2.8272	4.1989	1.72
C_7	0.7241	0.7053	0.6965	1.01
C_8	0.6229	0.7003	0.6369	0.74

These results are surprising because it was thought that results which are so far from the correct values are random numbers, for the following reason: the repeated loss of accuracy at so many steps in the realisation process has already been mentioned and leads to the assumption that all accuracy has been lost by the time that the input impedance Z has been obtained by the removal of the elements nearer to port 1, i.e. C_1 to C_5, from Z_{01}. It was therefore expected that the results would be random. However the values for C_6 are all greater than the "correct" value and furthermore the values 4.198.. and 4.715.. are surprisingly near to each other (for random numbers).

The above comment also applies to the elements C_5, C_7, C_8 because for each of them the values change in the same direction and are, for each element, much closer to each other than to the "correct" value. This makes it appear that there is a systematic reason for these errors which is independent of the method used, but no explanation has so far been found.

The same effect to a much diminished extent can be seen with respect to the element values calculated from Z_{S1}, Z_{S2}, i.e. the values from Z_{S1} for all the p-plane methods are closer to each other than to those from Z_{S2} for each particular method. (A similar comment is true for the element values calculated from Z_{S2} for all the p-plane methods). As an example of this effect the various results for element C_4 are given in table 9, the values being picked out from table 7 but rounded to 6 decimal figures. Similar remarks can be made for most of the other element values.

Table 9. Values for element C_4 calculated with 8-figure arithmetic from Z_{S1}, Z_{S2}.

Method	Value when C_4 is calculated from		Differences between values from Z_{S1} and Z_{S2} for the same method.
	Z_{S1}	Z_{S2}	
Orchard	1.189659	1.191980	-0.0023
Saraga "incompatible"	1.189037	1.192008	-0.0030
Saraga "compatible"	1.188728	1.192121	-0.0034
Maximum difference for values from the same impedance	0.0009	0.0001	
Range	(1.1887,1.1897)	(1.1919,1.1922)	

These results seem even more peculiar when it is remembered that because $Z_{O1} = Z_{O2}$, $Z_{S1} = Z_{S2}$ for a symmetrical filter, the same numerical expressions are taken for Z_{S1} and Z_{S2} and also for Z_{O1} and Z_{O2} but different ones are taken for each method. It therefore seemed reasonable to expect that the results would be more similar for the same method with different impedances than for the different methods with the same kind of impedance. However these effects are not apparent in the case of the element values calculated using single-length arithmetic; again, the reason for this is not understood.

5.4.2.2.3. Comparisons in terms of techniques (ii) and (iii).

The differences and relative differences have been calculated for the element values found from Z_{O1} and Z_{O2} for all the p-plane methods and are shown in appendix 6 in table 6.1 (elements found with single-length arithmetic) and in table 6.2 (elements found with 8-figure arithmetic). Similarly, differences have been calculated between the results from Z_{S1} and Z_{S2}, between those from Z_{S1} and Z_{O1} and also between those from Z_{S2} and Z_{O2} and are given in tables 6.3 to 6.8 (appendix 6). The maximum of the relative errors with respect to the elements found by each method is noted for each of the comparisons. Comments summarising the numerical information in the tables of appendix 6 are given here in table 10. The three sets of results (described in section 5.4.1. and appendix 5) for Saraga's compatible method are called the results of (a), (b), (c) of Saraga's compatible methods where

(a) consists of the results in columns (vii) and (ix) of table 6,

(b) consists of the results in columns (viii) and (ix) of table 6,

and (c) consists of the results in columns (x) and (xi) of table 6.

Table 10. Remarks based on study of differences and relative differences for element values of tables 6 and 7.

Comparison between results calculated from	Remarks for calculations involving	
	"single-length" arithmetic	"8-figure" arithmetic
z_{O1}, z_{O2}	See table 6.1. The results (c) of Saraga's "compatible" method are best at end 2 where the relative differences are largest which is valuable, but worst at end 1. Orchard's method gives the best results at end 1 and not too bad results at end 2.	See table 6.2. Saraga's "compatible" method gives some of the elements with smallest relative differences. Saraga's "incompatible" method gives the best elements for the elements nearest end 2 but the worst nearest end 1. It gives the best i.e. smallest maximum differences. Orchard's method gives a mixture of the best and worst element values.
z_{S1}, z_{S2}	See table 6.3. The results (a) (or(b)) for Saraga's "compatible" method are best. Orchard's method gives the worst results.	See table 6.4. Saraga's "incompatible" method gives good results; however, the "compatible" results are least good. Orchard's method gives a mixture of the best and worst results together with the best maximum relative difference.
z_{O1}, z_{S1}	See table 6.5. The results (c) of Saraga's "compatible" method are best both because it has more of the best values and the smallest maximum relative difference, but the results (a) are overall worse than any of the results of the other methods.	See table 6.6. The results for Saraga's "incompatible" method are best for all the elements except C_2 and C_7, and the maximum relative difference is also the best. The maximum relative difference is worst for the results of Orchard's method but otherwise there is little to choose between the results of Saraga's "compatible" and Orchard's methods.
z_{O2}, z_{S2}	See table 6.7. Orchard's method gives the best results and the best maximum relative difference. The results (c) of Saraga's "compatible" method are worst.	See table 6.8. The results of Saraga's "compatible" method are best for six of the eight relative differences listed and worst for one element but the maximum relative difference is smallest. The results of Saraga's "incompatible" method are worst.

5.4.2.3. Summary of discussion of results.

From the above discussion, it is obvious that the comparison tests completed so far are quite insufficient to reach any final conclusions. These tests can only be used as a basis for further investigations and for modifications of the comparisons carried out so far.

6. Development of programs.

Whilst the programming development work for this research is described, no attempt is made to produce users' manuals for the various programs. The material is treated in varying depth mainly according to its interest and not to the effort involved.

6.1. Conversion of programs written in Mercury Autocode

Programs already existed (written mainly by this researcher) for the analysis of ladder networks and also for the design of low-pass filters by each of the following three methods; the Orchard version of the conventional method, the Bingham method (both of these for filters of any order) and the Saraga method (limited to filters of order $n=7$). Unfortunately the programs had been written in Mercury Autocode for use on the now obsolete Mercury computers manufactured by Ferranti and therefore they had to be changed for use on a present day computer, in particular, the Thames Polytechnic 1902A computer. Some programs were converted into Extended Mercury Autocode (EMA) and others into Fortran.

6.1.1. Conversion to Extended Mercury Autocode.

Moreover as the programs were on the outdated 5-hole paper tape they had to be read into the computer and either the output had to be punched using 8-hole paper tape for editing on a teleprinter or edited direct on the computer. The latter method was chosen as it should be quicker and more accurate, although at least 3 computer runs were necessary. The first two runs use the XMED editor program and in the first the instructions are listed and numbered. In the second the instructions are altered by means of editing instructions and new EMA instructions fed into the computer from punched cards. The third run is needed to test the program and the EMA compiler has to be loaded into the computer before the altered program can be "executed". If the results are wrong, or there are any error messages, the second and third runs have to be repeated.

Two examples of the kind of Mercury Autocode instructions that had to be changed to convert the programs to EMA will now be discussed. Some changes were necessary because in Mercury Autocode for the extra accurate (double-length) numbers to be read and printed by means of a library program the "backing" store had to be used whereas in EMA there were instructions for reading and printing the numbers directly, and consequently library programs were not provided. Therefore the many pieces of program dealing with reading and printing had to be rewritten for the design programs because they all used double-length numbers. For example the Mercury Autocode instructions to print $\left(\frac{p}{2}\right)$ double-length numbers, stored in the pairs of locations,

$C_L, C_{L+1};\ C_{L+2}, C_{L+3};\ \ldots\ C_{L+p-2}, C_{L+p-1};$

were:

```
Ø7(100)CL,P
A'  = 100
M   =  10
N   =  20
U   =   1
π2  =  P/2 + 0.05
V   =  ØINT PT(π2)
PRESERVE
DOWN 2/1-512
RESTORE
```

and became, for the EMA version of the program:

```
N  =  L+P-2
M  =  L(2)N
NEWLINE
PRINT [((CM, C(M+1)))]10,20
REPEAT
```

Other changes to the programs were required because the EMA language does not include the powerful RMP instruction (read more program) of the Mercury Autocode Language. Therefore each of the numerous RMP instructions had to be changed to END.

The above examples indicate the nature of the work involved; it was very tedious and needed much care and attention. Moreover the resulting programs have two major disadvantages. One is that on each occasion EMA programs are to be used, the EMA compiler has to be loaded first. The time taken to do this is non-productive and as no one else is likely to need the EMA language, computer runs on such programs had to be given a low priority. This proved to be very inconvenient, both when altering and testing the resulting program and when actually using the programs for investigations; moreso as the results of one computer run often gave results which then had to be used to select the data for the next run. The second disadvantage is that as the Mercury Autocode language was not of interest to anyone else at the Thames Centre, it could not be supported i.e. no help could be obtained when unexplained errors arose, other than those which were"obviously" operating ones. This meant that when warning messages such as "unexpected cue found - EIVS" arose and when one element value in a design program was not printed and the causes could not be found from a programming point of view, the matter had to be left unresolved. More seriously still it cast doubts on results obtained for new data. There could even be undiscovered faults in the compiler.

The general analysis program for a ladder filter program of any order and two programs for part of the Saraga method for a filter of order 7 appeared to be satisfactory: not only did the results agree with those obtained previously on a Mercury computer but also no unexplained messages arose. It was confidently expected that results of future runs on these programs would be equally reliable. However the other programs, some of which had been converted to EMA and given rise to unexplained error messages, were rewritten in Fortran, despite the work involved.

6.1.2. Conversion to Fortran.

The conversion of large and complicated programs from Mercury Autocode to Fortran is not straightforward. To understand the reasons for this it is necessary to first appreciate some of the limitations of Mercury Autocode used on a Mercury computer.

One of the most important limitations is that programs are split up into chapters (similar to subroutines) of one standard size. If there are too few instructions in a chapter, the program is inefficient with respect to machine storage, for the rest of the chapter is wasted. It is inefficient with respect to machine time to change the chapters* too often because the time to do so is large compared with the time of executing arithmetical instructions. If there are too many instructions in a chapter it has to be split and in some cases this presents a difficult problem. In theory it is possible to jump from one chapter to a particular instruction in another, by an "Across instruction", but the use of too many across instructions in a program leads to "execution" faults. It is possible to avoid the use of an excess of "across instructions" because all of those used to change from (different parts of) a particular chapter, say 10, to (different parts of) the same other chapter, say 7, can be replaced by one "across instruction". Three modifications are then needed for each of the original "across instructions" : (1) a parameter has to be set in Chapter 10, (2) a jump instruction has to be inserted in Chapter 10 so that the one "across instruction", which arranges for the change to Chapter 7, is reached, (3) a special kind of jump instruction has to be inserted in Chapter 7; it tests the parameter and then arranges that the appropriate instruction is reached. Programs relying on such clumsy techniques are difficult to follow and cannot be translated into Fortran blindly one instruction at a time,without making the Fortran program equally clumsy and inefficient. Therefore the program has to be studied

---

* Most instructions were executed (obeyed) in the sequence in which they were written but jump instructions could be used to tell the computer which instruction to obey next, provided it was in the same chapter. However if the next instruction to be obeyed was in a different chapter, an "across instruction" was used and the "chapter had to be changed" i.e. the chapter containing the next instruction had to be brought to the "working store".

in depth and the algebra rewritten for at least some parts.

The need to rewrite the programs gave the opportunity for introducing a new facility (suggested by Dr. Saraga, see second footnote in section 5.1) namely, that of variable length arithmetic. He foresaw that it would be of interest to study filter design methods using arithmetic of different accuracy in the calculations e.g. 10-figure arithmetic, 8-figure arithmetic. Accordingly a subroutine R(A), described in section 6.3, was devised and written by this researcher and used in the Fortran versions of the programs. The penalties paid for this facility are that the Fortran statements (instructions) are clumsier and the programs less efficient with respect to computer time and storage than they would otherwise be. Expressions such as:

$$b_j = -g\, b_{j+1} - h\, b_{j+2} + a_j$$

can be written in ordinary Fortran as:

```
B(J) = -G * B(J+1) - H * B(J+2) + A(J)
```

but using the subroutine are written as:

```
B(J) = R(R(R(-G * B(J+1))-R(H * B(J+2))) + A(J)).
```

Lengthier expressions are even more intricate. No attempt was made to ensure that the arithmetic in the individual instructions was carried out in the same sequence in these Fortran programs as in the earlier Mercury Autocode programs.

Testing and debugging programs is often a time consuming task. The Fortran programs were produced under pressure, due to shortage of time and were not tested as thoroughly as the original Mercury Autocode programs: although the programs were meant to be used to design passive ladder filters of many different configurations they were mainly tested for the configurations of the filters that were to be studied. The tests gave results which agreed, within small rounding errors, with those obtained earlier from the Mercury Autocode programs. The need to get the numerical results as quickly as possible meant that no attempt was made to improve the programs or to present the results in an elegant form.

6.2. New programs for filters of order n=9.

A few details will now be given of the programs written to investigate Saraga's design method for ninth order filters; they were all written in Fortran by this researcher. The later programs used the variable length arithmetic facility developed previously (see section 6.1.2).

The programming work which was required to prepare the results described in section 4.2.2 had to be split between two programs because the Thames computer was to be used for the main calculations, but the Imperial College Kingmatic plotter was to be used to prepare the graphs automatically. To use the plotter the results had to be placed on magnetic tape in a particular way. This was effected with the aid of some Imperial College subroutines. The reasons for the way the work was split between the two programs will be explained later after the programs have been briefly described.

One of the programs used the improvement method and algebra described in section 3.1 and most of the option 1 defined in section 4.1 i.e. the steps shown in blocks 1 to 9 of fig.1. Summarising the essential points: the program makes the computer calculate and print the values of the inaccurate "incompatible" lattice impedances Z'_{ai}, Z'_{bi} obtained in each iteration of the improvement method, and the coefficients of the inaccurate "incompatible" function $K'(p)$ obtained from them. The improvement method was applied to the incompatible parameters only, but after each iteration the associated "compatible" parameters were calculated and they, together with the "compatible" $K'(p)$ found from them, were printed.

This program was used in two different ways: the one used most frequently will be described first. The first part of the input data consists of one set of the variously artificially deteriorated zeros of $D(p)+N(p)$* given in section 4.2.2. The coefficients of the numerator and denominator polynomials of the accurate function $K(p)$ (it will be explained later how these were obtained) completed the input data required by the program. The improvement method was then used repeatedly until the square of the difference between the coefficients of $K'(p)$, calculated in the latest iteration, and the corresponding coefficients of the accurate $K(p)$ was less than 0.00000001 for all the coefficients.** The accurate $K(p)$ function was required (as mentioned before) and rather than writing an extra program to calculate it, the program already described was applied in a special way, the second

---

* The accurate zeros of $H(p)$ are taken from the Christian and Eisenmann tables (ref.11) and it is possible to identify which of these zeros belong to the set of zeros of $D(p)+N(p)$. The others belong to $D(p)-N(p)$ and their mirror images in the imaginary p-axis complete the set of zeros of $D(p)+N(p)$.

** This was not the best possible test (because absolute errors rather than relative errors were assessed) but was satisfactory as the size of the coefficients of $K(p)$ were known to be near 1 in value.

of the two ways mentioned above, which will now be delineated: the exact zeros of $D(p)+N(p)$ were given as data. Usually the coefficients of the $K(p)$ function were also given as data but being unknown in this case, zero values were given instead. Only the first part of the program had to be used and the values of the accurate $K(p)$ function were then calculated and printed instead of the inaccurate "incompatible" function $K'(p)$.

The second of the programs used the various $K'(p)$ functions obtained in the first program and made the computer calculate the loss values for the specified frequency values and stored them on magnetic tape in the way needed for later use on the Kingmatic plotter at Imperial College. The program included some Imperial College subroutines and arranged for the setting of parameters needed in the use of the Kingmatic plotter. The parameters were used to determine the positions of the axes, their scales, the colour of the pen and the symbols to be used to plot the loss/frequency points of the different functions $K'(p)$ on the different scales.

Usually between 100 and 150 points were plotted for each $K'(p)$ function. Therefore, if the first program had been used instead of the second for the calculation of the loss/frequency values, 100-150 loss values, accurate to at least 4 significant figures, would have had to be punched instead of the 10 coefficients of $K(p)$, to single-length accuracy (11-12 significant figures).

Another method of transferring the data was considered, namely to store the results on magnetic tape using the ICL 1902A at Thames and then use the magnetic tape via Imperial College on the ULCC 6600. However the amount of data was insufficient to justify the work that member(s) of the Thames Computer Centre would have had to do to make such a procedure possible.

A short description follows of the new programs that had to be written before Saraga's method could be compared with established methods and the results described in Section 5.4 obtained. As mentioned in section 5.3 the design had to be started at an earlier stage than for the preceding investigations. The zeros and poles of the rational function $K(p) = N(p)/D(p)$ were given and the roots of the polynomial equation $N(p)+D(p) = 0$ had to be found instead of being given as for the earlier investigations. To do this Bairstow's root-finding method was used. A program was therefore written to calculate $N(p)$ and $D(p)$ from the zeros and poles of $K(p)$ and to find the roots of $N(p)+D(p) = 0$. For the comparison of the design methods the element values of the ladder network were required, whereas in the early investigations the loss/frequency values were required. Therefore a small program had to be written to calculate the open-and short-circuit impedances from the lattice impedances. The circuit was then designed by the same realisation program as for the conventional design methods. Furthermore the main program used in the investigation of Saraga's method (for a filter of order 9) had to be rewritten using the "variable accuracy" subroutine R(A) mentioned earlier, because the results were to be compared (for all the design methods) when specified numbers of significant figures were used in all the computations.

Rewriting the programs using the "variable accuracy" subroutine was not straightforward because the earlier program N9B (see table 11) uses a library program (as a "subprogram") to solve linear simultaneous equations. Unfortunately the library program is written in PLAN, a "lower level" language and therefore this library program could not be used with variable accuracy arithmetic, unless a special subroutine were written in PLAN and the library program were altered extensively (an enormous amount of work even if the author were familiar with the PLAN language and if ICL were willing to let us have the details of the program).

Table 11 Programs on main part of Saraga's method.

Program name	Means of solving the linear equations	arithmetic	filter element values shown in section 5.4.1. "incompatible"	"compatible"
N9B	Library program (PLAN)	single-length only	Table 6 columns (v),(vi)	Table 6 columns (vii),(viii),(ix)
N9G	"Subprogram 1" (Fortran) (existed previously)	single-length only		Table 6 columns (x),(xi)
N9GR	"Subprogram 2" (Fortran) same algebra as "sub-program 1" but different sequences of arithmetical operations for some intermediate calculations	variable-length	Table 7 columns (v),(vi)	Table 7 columns (vii), (viii)
N9GR2	"Subprogram 3" (Fortran) same algebra as "subprogram 1" and same sequences of arithmetical operations for all intermediate calculations	variable-length		

Other library programs could not be used either, because even those written in Fortran relied on subroutines written in PLAN. Therefore a new subprogram for solving linear simultaneous equations had to be developed using the "variable accuracy" subroutine R(A)*. As the library program was based on Gauss' method with partial pivotting and as it worked satisfactorily, the same numerical method was used

* Details of subroutine R(A) are given in section 6.3.

for the new Fortran program. Previously a program (subprogram 1, table 11) which later formed part of program N9G, had been developed by the writer for other purposes and used single-length arithmetic. A new Fortran program N9GR was developed using variable accuracy arithmetic and included subprogram 2 (also new) for solving the linear simultaneous equations. To test this new program (N9GR) a "dummy" subroutine R(A) (see later) was used to produce single-length arithmetic in the calculations so that the results could be checked against results from program N9G. However the results were not identical because the internal sequences of the calculations were sometimes different and had an effect on the rounding errors. Later, out of interest, another version of the program N9GR2, was written. In writing it, care was taken to ensure that the same sequences of calculations were reproduced as in the earlier program N9G. The limited time available prevented filters being designed with the aid of the latest version (N9GR2) of the program. Table 11 not only contains a summary of the programs existing for the main part of Saraga's method but indicates the position of the results obtained by means of the various programs.

6.3. New facilities for reducing computational accuracy to a chosen number of significant digits.

6.3.1. General.

Using Fortran on the ICL 1902A computer, it is easy to arrange for the arithmetic to be performed with 11-12 significant figures (i.e. 37 bits) by using single-length arithmetic or with 22-33 significant figures (i.e. 74 bits) by using double-length arithmetic*.

* These details are taken from ref.12: single-length arithmetic for Fortran uses "double-length" arithmetic, and double-length arithmetic for Fortran uses "quadruple-length" arithmetic in the computer.

To achieve other accuracies for the arithmetical operations, a subroutine R(A) was written to reduce the accuracy of the numbers by rounding them to a required number of significant digits. This subroutine was applied to the data and also after every single arithmetical operation of addition, subtraction, multiplication and division. An example of a statement using the subroutine has been given in section 6.1.2. The subroutine will be described in detail in the next section.

For the ordinary arithmetical operations the subroutine R(A) can be used but for "macro" Fortran instructions, in which several arithmetical operations are done automatically,e.g. for a square root, it is not possible to use R(A) because the intermediate operations would be calculated too accurately and consequently the value of the square root would be too accurate. Therefore a special subroutine had to be written to obtain the square roots with the required reduced accuracy: details of it are given in appendix 7.

6.3.2. Subroutine R(A).

The subroutine R(A) was written, as mentioned above, so that numbers could be rounded to a required number of digits: it rounds the number A to its N most significant digits. A general description of the three steps, on which it depends, is illustrated with numerical examples.

(1) This step is based on the relationship between the characteristic of the common logarithm of a number A. Either let M be the number of digits in A before its decimal point when $A \geqslant 1$ and $A \leqslant -1$ or let $-M$ be the number of leading zeros after the decimal point when $-1 < A < 1$. Thus

$$M = \text{int.pt.} \left\{\log_{10}|A|\right\} + 1 \qquad (6.1)$$

for all $A \geqslant 1$ and $A \leqslant -1$, or

$$M = \text{int.pt.} \left\{\log_{10}|A|\right\} \qquad (6.2)$$

for $-1 < A < 1$.

When $A = -9085$, equation (6.1) yields

$$M = \text{int.pt.} \left\{\log|-9085|\right\} + 1 = 4$$

and when $A = 0.0002$, then $\log_{10}A = \bar{4}.3010 = -3.6990$, and so equation (6.2) yields $M = -3$.

(2) In this step the number A is multiplied by 10^{N-M} to give a modified number with exactly N digits before its decimal point.

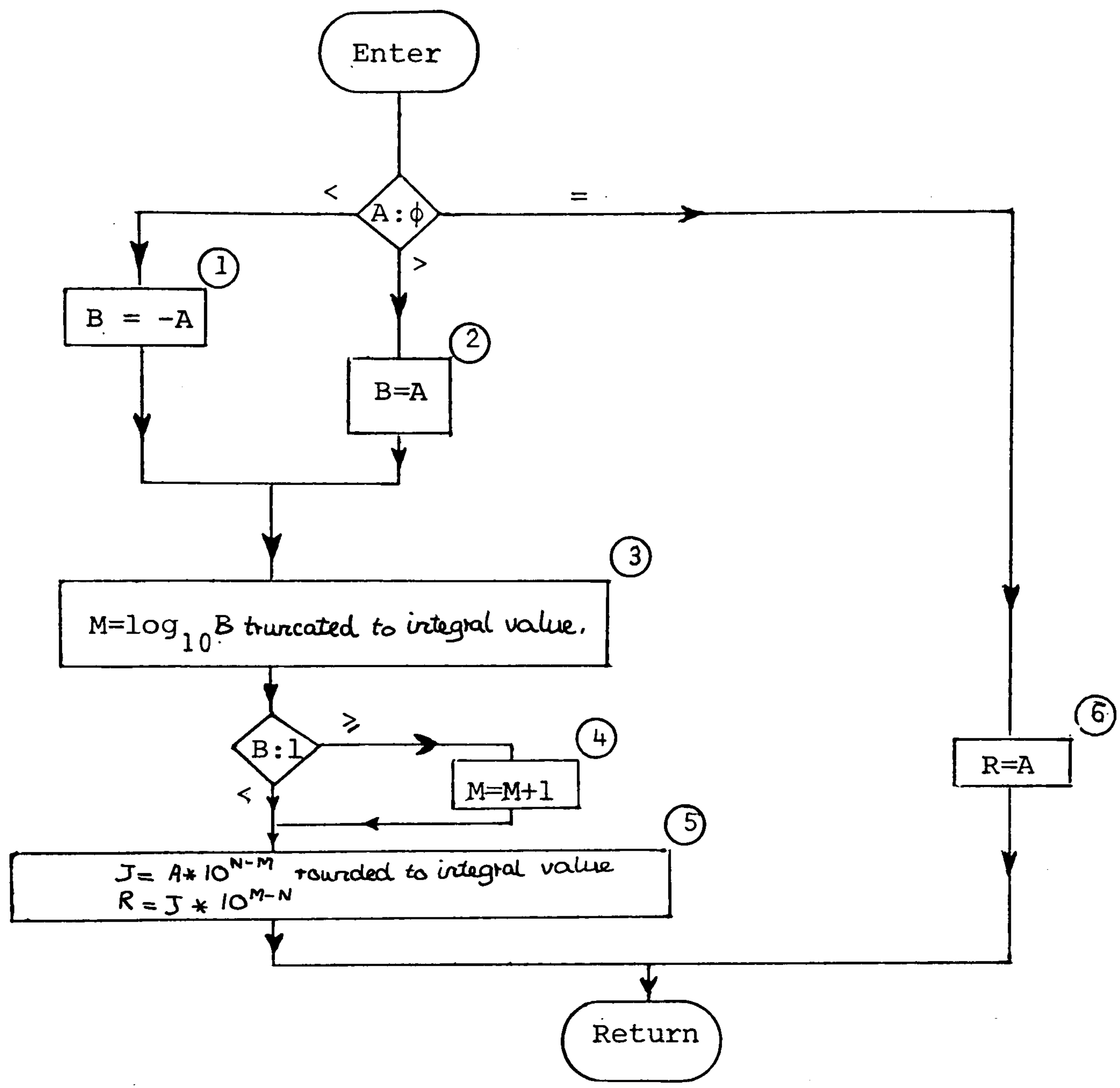
(3) This modified number is then rounded to the nearest whole number and multiplied by 10^{M-N} in order to give the original number rounded to the N required digits.

To illustrate these steps, consider the following examples in which the numbers 19378547.5, 3.976294 and 0.001568364 are to be rounded to 4 significant figures. Then $N = 4$ for all the numbers and M takes the values

8, 1 (by equation 6.1) and -2 (by equation (6.2)) respectively. In step (2) the multiplication factors 10^{N-M} are 10^{-4}, 10^{3} and 10^{6} and lead to the modified numbers

1937.85475, 3976.294 and 1568.364, respectively. In step (3) these modified numbers are rounded to 1938, 3976, 1568 and then multiplied by the factor 10^{M-N}, i.e. 10^{4}, 10^{-3}, 10^{-6}, to yield 19380000, 3.976, 0.001568. Clearly, the original numbers have been successfully rounded to 4 significant digits.

Fig. 39 Flow diagram for subroutine R(A)



In the flow diagram of this subroutine (see fig.39) $B=|A|$ (unless A=O) and blocks 3 and 4 together with the test in the diamond shaped box near them lead to the appropriate M value (which agrees with that given by equation (6.1) or (6.2) of step 1). Block 5 covers steps 2 and 3*.

When programs had to be used with true single-length working, the programs were run with a 'dummy' subroutine R(A), which consisted only of the statement R=A (apart from the control statements).

---

* The actual subroutine used to produce the results in section 5.4 calculated with "8-figure" arithmetic was slightly different. A parameter, NA was introduced to cater for positive and negative numbers, but it is now realised that it was unnecessary and can be avoided by writing block 5 in the form given in fig.39.

7. Scope for further research.

As discussed in section 5.4.2., the work reported in this thesis does not constitute a full assessment of Saraga's method, and therefore conclusions regarding its practical value when compared with the established methods cannot be drawn. On the basis of the results obtained so far, it would appear that in order to reach such conclusions it would be desirable to carry out the following investigations.

One of the serious limitations of the present investigations has been mentioned in section 5.2. and concerns the comparison of Saraga's method with other methods. The results were by no means conclusive and indicated different and sometimes conflicting trends. Consequently the following basic test needs to be performed. Saal's method should be used to design a filter based on the data given in fig.38. The resulting filter should be compared with the filters already designed by means of both versions ("compatible" and "incompatible") of Saraga's method. Such a test is important because both methods use the p-plane only and the later steps in the design procedure (realisation), which are not part of Saraga'a method, are performed by means of exactly the same techniques for both methods. Therefore any improvement (or otherwise) in the results should be directly attributable to Saraga's method.

Another starting point for further work concerns section 5. The writer feels that since some of the results seem to be difficult to explain, further tests are imperative. Furthermore many of the numerical results should be checked yet again before any final interpretation is attempted. Thus it would be useful to obtain further results which might reveal inconsistencies (if they exist) and/or confirm the tendencies noticed in the present results.

One of the checks could be to use a different program for the main part of Saraga's method. In section 6 it was mentioned that the program N9GR (which allowed for variable length arithmetic) did not give exactly the same results when single length arithmetic was used as the program N9G (which allowed for single length arithmetic only). It was also mentioned that the results were different because the sequences of the arithmetical operations were different in the two programs and that a program N9GR2 (which allowed for variable length arithmetic) was written later and gave identical results (when single length arithmetic was used) with those of program N9G. It would therefore be very interesting to see the filter element values when program N9GR2 instead of N9GR was used with 8 figure arithmetic and the programs for the rest of the design process were the same as in the earlier designs. Furthermore if all the design methods were used with 6 figure arithmetic, the results could be compared and inspected to see if the same tendencies were displayed by them as by the results (obtained with 8-figure arithmetic) shown in table 7 of section 5.

It is thought that Saraga's method might lead to noticeably better results compared with most of the comparison methods of section 5.3 if more "difficult" data than those of fig.36 were taken, i.e. data for a more "difficult" filter with a steeper transition band and smaller maximum passband loss than the filter CO915θ=70 of fig.36. The data would need to be chosen carefully if taken from ref.11 as some data given there lead to negative element values which are not of interest in this context. The reasons for suggesting that the results might be more favourable will now be indicated. In filter design (section 2) the p values of the zeros of the polynomial $D(p) + N(p)$ have to be found and the more "difficult" the filter, the closer together some of the zeros will be (i.e. "clustering" effect) and the more inaccurate are the values that are found likely to be. At a later stage in the design, "compatibility" has to be enforced by altering certain a_i and b_i parameters and the alterations should be much larger than for the filters already investigated (based on the data of fig. 36) where the changes were only of the size of rounding errors. It is to be hoped in such a more "difficult" filter that the advantage of enforcing compatibility between the parameters will be clearly seen in the results and not lost amongst the rounding errors as appears to happen in the case of the investigations described in this thesis.

It would be advantageous to modify the programs in some respects, for instance, to reduce the amount of data that have to be punched into cards from the printed output of the "earlier" programs in order to use the "later" programs. It is desirable to make the computer read and print either a title for the data or a parameter so as to avoid the risk of sets of data being run in a different sequence from that expected and this remaining undetected.

The investigations and modifications suggested above are of a routine character and should be almost straightforward but would be time consuming. The following investigations would be of quite a different character and would be of much greater interest from a mathematical point of view. As mentioned in section 1 Saraga points out that it should be possible to apply the principles of his method in the z-plane (i.e. to combine his method with that of Szentirmai/ Bingham). Furthermore the techniques of Musson's/Norek's method could also be included. The value of such steps is not known because both (or either) Szentimai's/Bingham's and Musson's/Norek's methods, might lead to such accurate values that the benefits of enforcing compatibility might be lost amongst the rounding errors.

It would be desirable and interesting from a mathematical point of view to continue the work of section 3.2.1., i.e. to obtain general mathematical expressions, for any n, for the second part of Saraga's method (such expressions for the first part are given in section 3.2.). However, the reasons for completing this would be stronger if the method were shown to be of practical value.

8. Conclusion

8.1. Background.

Saraga's method was completely new from a 'philosophical' point of view: Szentirmai's/Bingham's method is based on an appropriate choice of the independent variable (z instead of p). In this way the clustering of the poles and zeros of the relevant network functions, which occurs in the p-plane but not in the z-plane, is avoided. Musson and Norek deal with the rules of calculation: preference being given to multiplication rather than addition. In contrast, in Saraga's method network functions, i.e. the dependent variables are chosen in such a way, that even when inaccuracies occur, these do not lead to incompatibilities (or where they occur, they can be removed).

Early investigations (ref.7) showed the new method to be attractive and it was decided that it should be investigated more thoroughly. The initial aims were to develop the theory for higher values of n and to apply it in the same - somewhat artificial - way (artificial deterioration in the zeros of $1 + K(p)$ or $N(p) + D(p)$) in which it was applied in the case of the earlier investigations with lower n values ($n=7$). It was planned to apply the design method afterwards in a more realistic situation.

8.2. Investigations on and extensions of Saraga's method.

Investigations for filters of order $n=5$ and $n=7$ were reported (ref.7) before the beginning of the work for this thesis, and as a first step the investigations on the filter of order $n=7$ were continued (the results are shown in fig.3 to 19), then the method was extended to filters of order $n=9$ and a filter of such an order was investigated (the results are shown in fig.20 to 35). The results of all these investigations which started from the artificially deteriorated zeros of $1+K(p)$ were favourable. Furthermore it proved possible to generalise the first part of the methods to symmetrical filters of any order.

8.3 Later investigations.

As all the earlier investigations gave encouraging results, it was decided that Saraga's method should be tested in a more practical situation, i.e. as if the filters were to be designed for practical use. Therefore instead of artificially deteriorating the zeros of $1+K(p)$ (the accurate values of which were obtained from tables), in the later investigations the zeros were regarded as unknown and were obtained by a normal iteration procedure.

Furthermore, a difficulty concerning the formulation of a comparison in a "practical" context arose from the fact that Saraga's method is concerned only with that part of the filter design method which starts with a chosen $K(p)$ and obtains the short-circuit and open-circuit immittances of the filter. In a practical context the required final result of the design procedure is the set of element values. However, the circuit realisation, i.e. the computation of the element values from the filter immittances, is beset with a similar accuracy problem to that in the first part of the design procedure. Therefore the question arises of which method should be used for the realisation of the filter immittances in the case of Saraga's method. In the work for this thesis, for Saraga's filters, the conventional p-plane method was employed, with the same arithmetical accuracy as that used for the first part of the design. Therefore the comparison with Szentirmai's/Bingham's method is rather unfair on Saraga's method as Szentirmai's/Bingham's method uses the "less-accuracy-requiring" z-plane method not only for the first part of the design procedure (with reference to which various methods were compared) but also for the realisation part where Saraga's method, is unduly handicapped in these comparisons by the use of the p-plane.*

---

* The comparison was carried out in this way in spite of this "unfairness", because two apparently obvious ways for overcoming the unfairness, namely to realise either Szentirmai's/Bingham's filter in the p-plane or Saraga's filter in the z-plane are neither straightforward in practice nor conceptually fully acceptable. In view of the original optimism about Saraga's method (see section 4.2.1 page 53 and section 4.2.2 page 73 it was thought that the comparison which had actually been carried out would be justified in spite of its lack of fairness.

In section 5 a number of different possible comparisons are discussed. But only a limited number could be undertaken, and since the early results from Saraga's method looked so attractive an optimistic attitude was taken and the most ambitious comparison was carried out; a comparison of Saraga's method with Szentirmai's/Bingham's, in spite of the "unfairness" mentioned above.

Different choices of criteria for the comparison were considered Obviously, accuracy of computation and accuracy of results were the main criterion, since a reduction of the computing accuracy required to obtain the element values to an acceptable accuracy was the main aim of Saraga's method. Specifically the accuracy of the element values produced by the different design methods when a predetermined number of significant digits were used in the calculations, were compared.

Basically the same filter data (see fig.36-38) were used for the different design methods (Orchard's, Szentirmai's/Bingham's, and several versions of Saraga's compatible and incompatible methods). The results were tabulated (see tables 6 and 7 and appendix 6) and were studied in detail. They were found to be inconclusive, and a number of possible reasons for this outcome are mentioned in the thesis; some of these reasons seem probable although others are only tentative. Some comments on further tests and investigations which should be performed are given in sections 5 and 7. It is very much regretted that it has not been possible to complete a thorough assessment nor to reach a conclusion. As mentioned in section 7, the writer thinks that a more "difficult" example should be chosen, that Saal's method should be included in the comparisons and further that the calculations should be performed with other numbers of significant digits.

8.4. Computer programs.

Existing programs for Saraga's method (for filters of order 7) and for "comparison" methods by Orchard and Szentirmai/Bingham had to be converted from the Mercury Autocode language for the obsolete Mercury computer to a form that could be used on a present-day computer. At first an attempt was made to convert the programs to Extended Mercury Autocode (EMA) for use on the Thames ICL 1902A computer but the use of the EMA programs was not to be encouraged for the various reasons given in chapter 6. Therefore the programs mentioned above were rewritten in Fortran and new programs were written for Saraga's method for filters of order n=9. Also a Fortran program was written to use the automatic graph plotting routines and fig.20 to 35 (n=9) were prepared automatically on a "Kingmatic" plotter. Before the rewriting was commenced a subroutine (used in all the later programs) was written to round numbers to a predetermined number of significant figures so that the calculations could be performed with "any" arithmetical accuracy.

Bibliography

1. DARLINGTON, S.:

   Synthesis of reactance four-poles which produce prescribed insertion loss characteristics. J.Math. and Phys. vol.18 Sept.1939, Bell Telephone System Monograph B 1186, pp.50-51.

2. SAAL, R. and ULBRICH, E.:

   On the design of filters by synthesis. I.R.E. Trans. C.T. vol.C.T.5 Dec. 1958, pp. 284-327.

3. SZENTIRMAI, G.:

   Theoretical basis of a digital computer program package for filter synthesis. Proc. 1st. Allerton Conf. on Circuit and System Theory, 1963, pp.37-49.

4. BINGHAM, J.A.C.:

   A new method of solving the accuracy problem in filter design. I.E.E.E. Trans., 1964, pp.327-341.

5. NOREK, C.:

   Product Method for the Calculation of the effective loss L,C filters. Proc. Int. Symp. on Network Theory, Sept.1968 University of Belgrade, Belgrade, Yugoslavia, pp.353-365.

6. SKWIRZYNSKI, J.K.:

   On synthesis of filters: the product method of Musson and Norek. I.E.E.E. International Symposium on Circuit Theory, Dec.1969,pp.7-8.

7. SARAGA, W.:

   A new approach to insertion loss ladder filter design. Sept.1966 Conference on Electrical Networks, University of Newcastle upon Tyne.

8. ORCHARD, H.J. and TEMES, G.C.:

   "Passive Filters with lumped elements" in the book "Modern Filter Theory and Design" edited by Temes G.C. and Mitra S.K.: John Wiley and Sons, 1973 pp. 77-120.

9. FOX, L.:

An introduction to numerical linear algebra, Clarendon Press 1964, Ch.3, pp.60-65.

10. SAAL, R.:

Der Entwurf von Filtern mit Hilfe des Kataloges Normierter Tief Pässe, published by Telefunken, 1961.

11. CHRISTIAN, E. and EISENMANN, E.:

Filter design tables and graphs. John Wiley and Sons Inc.

12. ICL technical publication 4095 "Central Processors" 1900 series.

Appendix 1

Comment on factorisation of $D^2(p) - N^2(p)$.

The factorisation of $D^2(p) - N^2(p)$ into the product $B(p)B(-p)$ where $B(p)$ is assumed to be a "strictly" Hurwitz polynomial (which is defined here as having no zeros on the imaginary axis), presupposes that the real polynomial (in p) $D^2(p) - N^2(p)$ does not contain conjugate pairs of zeros on the imaginary axis one zero of each pair being associated with $B(p)$ and the other with $B(-p)$. The occurrence of such zeros of $D^2(p) - N^2(p)$ i.e. of $H(p)H(-p)$ would be inadmissible for passive resistively terminated networks since the zeros of $H(p)$ are transfer function poles which must lie in the (restricted) left half of the p-plane.

This same point can also be shown by the following argument. Since $K(p)$ is an odd function of p,

$$K(p) = pA(p^2)$$

where $A(p^2)$ is a function in p^2. When $p = j\omega$,

$$K(j\omega) = j\omega A(-\omega^2)$$

which is purely imaginary and therefore

$$K(j\omega) \neq \pm 1, \text{ for any } \omega . \qquad (1)$$

Consideration of equation (1.3) shows that where $H(p)H(-p) = 0$, $K(p) = \pm 1$. Since from (1) $K(j\omega) \neq \pm 1$, $H(j\omega)H(-j\omega) \neq 0$ for any ω.

Appendix 2

Proof that D(p) + N(p) can be factorised into the product of a Hurwitz and an anti-Hurwitz polynomial.

In section 1 it was shown that

$$D^2(p) - N^2(p) = \{D(p) + N(p)\}\{D(p) - N(p)\} = B(p)B(-p) \qquad (1)$$

and in appendix 1 it was shown that $D^2(p) - N^2(p)$ does not contain zeros on the imaginary axis in the p-plane.

Therefore D(p) + N(p) can only contain factors with zeros in the left half p-plane (if any) and factors with zeros in the right half p-plane (if any) but there cannot be any factors with zeros on the imaginary axis. Thus it is possible to write

$$D(p) + N(p) = B_a(p)B_b(-p)$$

where $B_a(p)$, $B_b(p)$ are Hurwitz polynomials and it is possible for $B_a(p) \equiv 1$ or $B_b(p) \equiv 1$ when all the zeros lie either in the right half or in the left half p-plane.

Appendix 3.

The conventional realisation of ladder filters.

The starting point is the open circuit impedance Z_{O1} (or the short circuit impedance Z_{S1}) and the list of required attenuation poles (i.e. the zeros of the denominator D of H(p)). The first step is to choose a ladder structure which corresponds to the particular H(p) function. This means that, corresponding to each attenuation pole at a finite frequency there must be in the ladder structure a parallel resonant circuit as a series arm, or a series resonant circuit as a shunt arm, tuned to the frequency of this attenuation pole. Furthermore the two "asymptotic" circuits to which the chosen ladder circuit degenerates at zero and infinite frequencies must provide the right number of attenuation poles at these extreme frequencies (i.e. in accordance with the number of poles of H(p) at zero and infinite frequencies).

For the second step let us assume that we have a circuit as shown in Fig.1. In order to find L_1 we note that as $\omega \to \infty$, $Z_{O1} \to Z_{L_1}$.

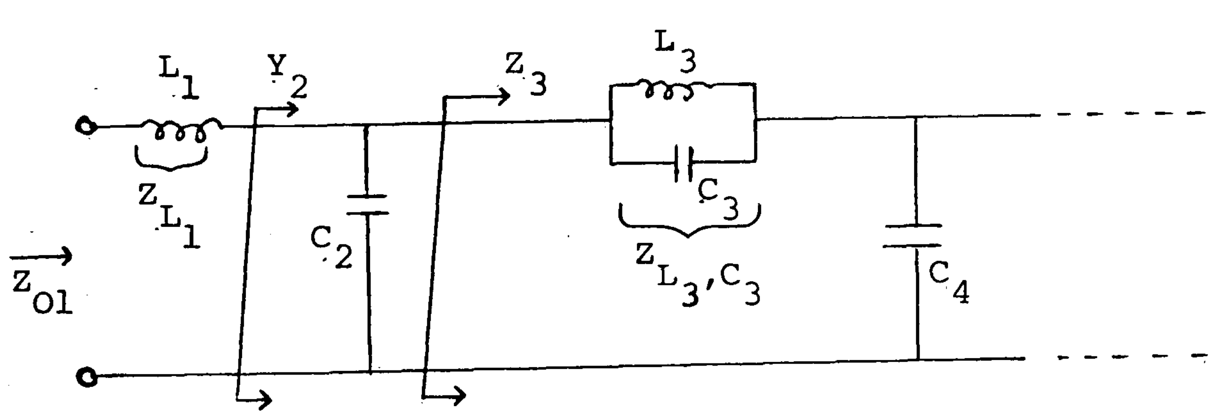


Fig.1.

In this way L_1 can be found and $Z_{L_1}(j\omega)$ can be subtracted from $Z_{O1}(j\omega)$ to obtain $Z_2(j\omega)$. The admittance of the arm containing L_3 and C_3 is, of course, zero at its frequency of resonance $\omega_{3\infty}$ and this means that the rest of the circuit has no effect and all the signal passes through the arm containing C_2. Thus evaluating $Y_2(j\omega) = \dfrac{1}{Z_2(j\omega)}$ at $\omega = \omega_{3\infty}$

gives the admittance Y_{C_2} and hence C_2. Next Y_{C_2} is subtracted from Y_2 to give $Y_3 = \frac{1}{Z_3}$, completing the process known as the partial removal of a pole. The next stage is to totally remove the pole at $\omega_{3\infty}$. This is arranged by setting the residue of Z_{L_3,C_3}, the impedance of the arm containing L_3 and C_3, equal to the residue of Z_3 at $\omega = \omega_{3\infty}$. Thus C_3 is found, and hence L_3 since $\omega_{3\infty} = 1/(L_3C_3)^{\frac{1}{2}}$. This method can be continued.

If, instead of a parallel tuned circuit in a series arm, a series tuned circuit in a shunt arm has to be evaluated, the method is essentially the same, except that the impedance and admittance have to be interchanged.

The whole filter can be realised from one end (provided that $Z_{O1}(Z_{S1})$ is used if the filter at port 2 ends with a shunt (series) arm). Alternatively, the filter can be realised from both ends. If there is no significant loss of accuracy, then, where the two realisation procedures meet, the same element values will be obtained by both procedures.

Appendix 4.

Some details of the Norek/Musson method.

As mentioned in section 1.4, this method deals with polynomials in their factorised form. In this appendix the algebra, based on ref.6 but with different symbols is described and proved in the case of adding two polynomials $A(p)$ and $B(p)$ and obtaining the resulting polynomial $C(p)$ directly in its factorised form. The roots of polynomial $C(p)$ are found by Newton's method and therefore the polynomials $A(p)$ and $B(p)$ and their differentials have to be evaluated repeatedly near the roots of $C(p)$.

It is worth considering first, the differential of one of the polynomials in factor form and then applying the result to the sum of the polynomials.

Let
$$A(p) = C_1 \prod_{i=1}^{n} (p-a_i) \tag{1}$$

with C constant and zeros a_i, $i=1,2,\ldots n$. Differentiating with respect to p gives

$$A'(p) = C_1\Big\{(p-a_2)(p-a_3)\ldots(p-a_n) + (p-a_1)(p-a_3)\ldots(p-a_n) + \ldots + (p-a_1)(p-a_2)\ldots(p-a_{n-1})\Big\} \tag{2}$$

$$A'(p) = C_1\left\{\frac{(p-a_1)(p-a_2)(p-a_3)\ldots(p-a_n)}{(p-a_1)} + \frac{(p-a_1)(p-a_2)(p-a_3)\ldots(p-a_n)}{(p-a_2)} + \ldots + \frac{(p-a_1)(p-a_2)\ldots(p-a_{n-1})(p-a_n)}{(p-a_n)}\right\}$$

$$= C_1\left\{\prod_{i=1}^{n}(p-a_i)\right\}\left\{\frac{1}{p-a_1} + \frac{1}{p-a_2} + \ldots \frac{1}{p-a_n}\right\}$$

$$A'(p) = C_1\left\{\prod_{i=1}^{n}(p-a_i)\right\}\left\{\sum_{i=1}^{n}\frac{1}{(p-a_i)}\right\} \tag{3}$$

Similarly $B(p) = C_2 \prod_{i=1}^{m} (p-b_i)$ where C_2 is a constant and b_i, $i=1,2,\ldots m$ are the zeros of $B(p)$. Then it is required to find $C(p)=A(p)+B(p)$ in the form

$$C(p) = C_3 \prod_{i=1}^{\max(n,m)} (p-c_i)$$

where C_3 is a constant and c_i, $i=1,2,\ldots$ maximum of n and m, are the zeros of $C(p)$ and

$$C(p) = C_1 \prod_{i=1}^{n} (p-a_i) + C_2 \prod_{i=1}^{m} (p-b_i) \tag{4}$$

Applying the result given by expression (3) for $A(p)$ to $B(p)$ and using equation (4), gives the polynomial $C(p)$ differentiated with respect to p, as

$$C'(p) = C_1 \left\{ \prod_{i=1}^{n} (p-a_i) \right\} \left\{ \sum_{i=1}^{n} \frac{1}{(p-a_i)} \right\} + C_2 \left\{ \prod_{i=1}^{m} (p-b_i) \right\} \left\{ \sum_{i=1}^{m} \frac{1}{(p-b_i)} \right\} \tag{5}$$

Whereas in theory Newton's formula can be used repeatedly

$$p_n = p_o - \frac{C(p_o)}{C'(p_o)}$$

where p_o is the "old" root and p_n is the "new" improved value until a root is perfected and then the next root is improved, in practice the roots are too close together and there is the danger of relocating the same root instead of locating a different root. Therefore, instead of using the polynomial $C(p)$, the rational function $G(p) = \frac{C(p)}{D(p)}$ is used where the polynomial $D(p)$ is the product of the factors whose roots are the roots so far found of the polynomial $C(p)$. Then

$$G'(p) = \frac{C'(p)}{D(p)} - \frac{C(p)D'(p)}{D^2(p)}$$

and the Newton correction to the approximate root p_i, is

$$\frac{-G(p_i)}{G'(p_i)} = \frac{-C(p_i)}{C'(p_i) - \dfrac{C(p_i)D'(p_i)}{D(p_i)}} .$$

The polynomials $C(p_i)$ and $C'(p_i)$ are calculated from equations (4) and (5) and the polynomials $D(p_i)$, $D'(p_i)$ are calculated from equations similar in form to those of (1) and (3).

Appendix 5. Reasons for existence of columns (vii) to (xi) of table 6, section 5.4.

Three sets of results occur for the "single-length" calculations using Saraga's method as mentioned in Section 5.4. Two of the sets arise because the coefficients of the numerator of Z_O and the denominator of Z_S should be identical but their coefficients of p^4 were not, as explained below. The coefficient of p^4 in the numerator of Z_O was calculated from the expression (the sequence of the calculation is indicated by the position of the brackets):-

$$G_1 = (((a_1b_3+a_3b_1)+a_2b_2)+(a_4+b_4))$$

and gave 27.211883704175932052. Similarly the coefficient of p^4 in the denominator of Z_S was calculated from the expression:-

$$G_2 = (((a_4+b_4)+a_2b_2)+(a_1b_3+a_3b_1))$$

and gave 27.211883704410411155.

These two different values for the coefficient of p^4 in Z_O and Z_S were used to produce the results in column (vii) of table 6. In column (viii) of table 6 only the second value was used; thus the numerator of Z_O and the denominator of Z_S were the same. It should be noted that the two different figures given by, G_1, G_2 for this one datum value only differ in the 12th significant digit and yet the values for the element values differ in some cases in the 5th significant digit e.g. see the numbers calculated from Z_o, for C_6 in columns (vii) and (viii) and in one case, C_8, even in the 4th significant digit. This indicates the possible effect of rounding errors on the realisation part. Therefore it has nothing to do with Saraga's method itself; it might also have occurred in the other methods but this was not investigated.

The set of results given in columns (x) and (xi) differ from those of columns (viii) and (ix) (although the data were the same for (x) and (viii) and for (xi) and (ix) because a different subroutine was used to solve the linear equations.

Appendix 6. Tables of differences and relative differences of element values in tables 6 and 7, section 5.4.1.

The differences in the tables of this appendix are shown rounded to only 3 significant figures and the exact sizes of the element values needed for the calculation of the"relative" differences are not really important but they have been chosen, after looking at the results of table 6, as likely values (to 4 significant figures).

The best, i.e. the smallest, of the"relative"differences for each particular element is inside its own rectangular box with broken sides (or inside a larger one when several consecutive"relative"differences for elements calculated by the same method from the same impedance are best). The worst, i.e. largest, of the"relative" differences for each particular element is surrounded by an "oval" shape with unbroken sides. In each set of comparisons the modulus of maximum (or worst) "relative" difference for each method is also noted; the "relative" difference was defined in section 5.4.2.2.1. as the difference (not the absolute or modulus of the difference) divided by the element value.

Table 6.1. Differences and "relative" differences between elements calculated from Z_{01} and Z_{02} with "single-length" arithmetic.

method		differences					"relative" differences				
		Orchard	Saraga "incompatible"	Saraga "compatible"			Orchard	Saraga "incompatible"	Saraga "compatible"		
element	approximate element value			(a)	(b)	(c)			(a)	(b)	(c)
c_1	1.040	-6.95×10^{-6}	1.54×10^{-5}	1.95×10^{-5}	1.50×10^{-5}	-2.78×10^{-5}	-6.69×10^{-6}	1.48×10^{-5}	-1.87×10^{-5}	1.44×10^{-5}	-2.67×10^{-5}
c_2	0.1839	-1.19×10^{-7}	2.63×10^{-7}	-3.33×10^{-7}	2.56×10^{-7}	-4.74×10^{-7}	-6.45×10^{-7}	1.43×10^{-6}	-1.81×10^{-6}	1.39×10^{-6}	-2.58×10^{-6}
c_3	1.354	5.56×10^{-6}	-1.20×10^{-5}	1.51×10^{-5}	-1.24×10^{-5}	2.22×10^{-5}	4.11×10^{-6}	-8.88×10^{-6}	1.12×10^{-5}	-9.15×10^{-6}	1.64×10^{-5}
c_4	1.192	-6.73×10^{-6}	1.04×10^{-5}	-1.05×10^{-5}	1.62×10^{-5}	-2.20×10^{-5}	-5.64×10^{-6}	8.73×10^{-6}	-8.84×10^{-6}	1.36×10^{-5}	-1.84×10^{-5}
c_5	0.8403	1.69×10^{-5}	1.23×10^{-5}	-5.33×10^{-5}	-3.58×10^{-5}	-5.54×10^{-6}	2.01×10^{-5}	1.47×10^{-5}	-6.34×10^{-5}	-4.27×10^{-5}	-6.59×10^{-6}
c_6	1.715	-3.77×10^{-4}	-3.47×10^{-4}	1.34×10^{-3}	7.81×10^{-4}	2.48×10^{-4}	-2.20×10^{-4}	-2.02×10^{-4}	7.84×10^{-4}	4.55×10^{-4}	1.44×10^{-4}
c_7	1.010	3.02×10^{-4}	2.82×10^{-4}	-1.08×10^{-3}	-6.25×10^{-4}	-2.02×10^{-4}	2.99×10^{-4}	2.79×10^{-4}	-1.07×10^{-3}	-6.19×10^{-4}	-2.00×10^{-4}
c_8	0.7400	-9.49×10^{-5}	-8.87×10^{-5}	3.38×10^{-4}	1.95×10^{-4}	6.35×10^{-5}	-1.28×10^{-4}	-1.20×10^{-4}	4.57×10^{-4}	2.64×10^{-4}	8.58×10^{-5}
c_9	0.6837	-3.17×10^{-4}	-2.97×10^{-4}	1.14×10^{-3}	6.58×10^{-4}	2.13×10^{-4}	-4.64×10^{-4}	-4.35×10^{-4}	1.66×10^{-3}	9.63×10^{-4}	3.12×10^{-4}
Maximum modulus of "relative" differences							4.64×10^{-4}	4.35×10^{-4}	1.66×10^{-3}	9.63×10^{-4}	3.12×10^{-4}

Table 6.2. Differences and "relative" differences between elements calculated from Z_{01} and Z_{02} with "8-figure" arithmet

method		differences			"relative" differences		
element	approximate element value	Orchard	Saraga "incompatible"	Saraga "compatible"	Orchard	Saraga "incompatible"	Saraga "compatible"
C_1	1.040	-0.0177	-0.0708	0.00297	-0.0170	-0.0681	0.00286
C_2	0.1839	-2.54×10^{-4}	-4.40×10^{-4}	5.30×10^{-5}	-1.38×10^{-3}	-2.39×10^{-3}	2.88×10^{-4}
C_3	1.354	0.0120	0.0546	-0.00469	8.86×10^{-3}	4.03×10^{-2}	-3.46×10^{-3}
C_4	1.192	0.0109	-0.0617	0.0244	9.18×10^{-3}	-5.17×10^{-2}	2.05×10^{-2}
C_5	0.8403	-0.181	-0.0358	-0.141	-2.16×10^{-1}	-4.26×10^{-2}	-1.68×10^{-1}
C_6	1.715	3.00	1.10	2.48	1.75	6.43×10^{-1}	1.45
C_7	1.010	-0.286	-0.305	-0.314	-2.83×10^{-1}	-3.02×10^{-1}	-3.11×10^{-1}
C_8	0.7400	-0.117	-0.0397	-0.103	-1.58×10^{-1}	-5.37×10^{-2}	-1.39×10^{-1}
C_9	0.6837	0.474	0.357	0.457	6.93×10^{-1}	5.22×10^{-1}	6.68×10^{-1}
Maximum modulus of "relative" differences					1.75	0.643	1.45

Table 6.3. Differences and "relative" differences between elements calculated from Z_{S1} and Z_{S2} with "single-length" arithmetic.

method		differences				"relative" differences			
element	approximate element value	Orchard	Saraga "incompatible"	Saraga "compatible" (a) = (b)	Saraga "compatible" (c)	Orchard	Saraga "incompatible"	Saraga "compatible" (a) = (b)	Saraga "compatible" (c)
c_2	0.1839	-1×10^{-11}	5.5×10^{-10}	9×10^{-11}	3.2×10^{-10}	-5.44×10^{-11}	2.99×10^{-9}	4.89×10^{-10}	1.74×10^{-9}
c_3	1.354	2.51×10^{-7}	-1.82×10^{-7}	-1.12×10^{-7}	1.4×10^{-7}	1.85×10^{-7}	-1.34×10^{-7}	-8.29×10^{-8}	1.03×10^{-7}
c_4	1.192	-1.18×10^{-6}	7.52×10^{-7}	5.21×10^{-7}	-6.09×10^{-7}	-9.89×10^{-7}	6.31×10^{-7}	4.37×10^{-7}	-5.11×10^{-7}
c_5	0.8403	6.12×10^{-7}	-1.38×10^{-7}	-2.48×10^{-7}	1.53×10^{-7}	7.28×10^{-7}	-1.64×10^{-7}	-2.95×10^{-7}	1.82×10^{-7}
c_6	1.715	2.05×10^{-6}	-1.75×10^{-6}	-9.41×10^{-7}	1.35×10^{-6}	1.20×10^{-6}	-1.02×10^{-6}	-5.48×10^{-7}	7.88×10^{-7}
c_7	1.010	-6.72×10^{-7}	2.35×10^{-7}	2.8×10^{-7}	-2.18×10^{-7}	-6.65×10^{-7}	2.33×10^{-7}	2.77×10^{-7}	-2.16×10^{-7}
c_8	0.7400	1.65×10^{-7}	-1.45×10^{-8}	-6.55×10^{-8}	2.26×10^{-8}	2.23×10^{-7}	-1.96×10^{-8}	-8.85×10^{-8}	3.05×10^{-8}
Maximum modulus of "relative" differences						1.20×10^{-6}	1.02×10^{-6}	5.49×10^{-7}	7.88×10^{-7}

Table 6.4. Differences and "relative" differences between elements calculated from Z_{S1} and Z_{S2} with "8-figure" arithmet

method		differences			"relative" differences		
element	approximate element value	Orchard	Saraga "incompatible"	Saraga "compatible"	Orchard	Saraga "incompatible"	Sarage "compatible"
C_2	0.1839	1.64×10^{-6}	1.61×10^{-6}	-1.55×10^{-6}	8.92×10^{-6}	8.75×10^{-6}	-8.43×10^{-6}
C_3	1.354	4.48×10^{-4}	6.59×10^{-4}	7.90×10^{-4}	3.31×10^{-4}	4.87×10^{-4}	5.83×10^{-4}
C_4	1.192	-2.32×10^{-3}	-2.97×10^{-3}	-3.39×10^{-3}	-1.95×10^{-3}	-2.49×10^{-3}	-2.85×10^{-3}
C_5	0.8403	1.48×10^{-3}	1.18×10^{-3}	1.20×10^{-3}	1.77×10^{-3}	1.41×10^{-3}	1.43×10^{-3}
C_6	1.715	3.61×10^{-3}	5.86×10^{-3}	6.88×10^{-3}	2.10×10^{-3}	3.41×10^{-3}	4.01×10^{-3}
C_7	1.010	-1.50×10^{-3}	-1.40×10^{-3}	-1.50×10^{-3}	-1.49×10^{-3}	-1.38×10^{-3}	-1.49×10^{-3} *
C_8	0.7400	4.20×10^{-4}	2.81×10^{-4}	2.91×10^{-4}	5.67×10^{-4}	3.80×10^{-4}	3.94×10^{-4}
Maximum modulus of "relative" differences					2.10×10^{-3}	3.41×10^{-3}	4.01×10^{-3}

*Selection of worst element by looking at more accurate values.

Table 6.5. Differences and "relative" differences between elements calculated from z_{01} and z_{S1} with "single-length" arithmetic.

method		differences					"relative" differences				
		Orchard	Saraga "incompatible"	Saraga "compatible"			Orchard	Saraga "incompatible"	Saraga "compatible"		
element	approximate element value			(a)	(b)	(c)			(a)	(b)	(c)
c_1	1.040	1×10^{-10}	1×10^{-10}	0	0	-1×10^{-10}	9.62×10^{-11}	9.62×10^{-11}	0	0	-9.62×10^{-11}
c_2	0.1839	1.3×10^{-10}	-2.6×10^{-10}	-2×10^{-11}	-3×10^{-11}	1.9×10^{-10}	7.07×10^{-10}	-1.41×10^{-9}	-1.09×10^{-10}	-1.63×10^{-10}	1.03×10^{-9}
c_3	1.354	-2.88×10^{-7}	5.57×10^{-7}	-4.21×10^{-7}	-2×10^{-7}	-3.44×10^{-7}	-2.13×10^{-7}	4.11×10^{-7}	-3.11×10^{-7}	-1.48×10^{-7}	-2.54×10^{-7}
c_4	1.192	3.59×10^{-7}	-3.37×10^{-6}	5.47×10^{-6}	2.96×10^{-6}	2.17×10^{-6}	3.01×10^{-7}	-2.83×10^{-6}	4.59×10^{-6}	2.48×10^{-6}	1.82×10^{-6}
c_5	0.8403	1.49×10^{-5}	1.54×10^{-5}	-5.68×10^{-5}	-3.27×10^{-5}	-1.10×10^{-5}	1.78×10^{-5}	1.83×10^{-5}	-6.75×10^{-5}	-3.90×10^{-5}	-1.30×10^{-5}
c_6	1.715	-3.79×10^{-4}	-3.47×10^{-4}	1.35×10^{-3}	7.80×10^{-4}	2.50×10^{-4}	-2.21×10^{-4}	-2.02×10^{-4}	7.86×10^{-4}	4.55×10^{-4}	1.46×10^{-4}
c_7	1.010	3.02×10^{-4}	2.81×10^{-4}	-1.08×10^{-3}	-6.25×10^{-4}	-2.02×10^{-4}	2.99×10^{-4}	2.79×10^{-4}	1.07×10^{-3}	-6.19×10^{-4}	-2.00×10^{-4}
c_8	0.7400	-9.51×10^{-5}	-8.87×10^{-5}	3.38×10^{-4}	1.95×10^{-4}	6.35×10^{-5}	-1.28×10^{-4}	-1.20×10^{-4}	4.57×10^{-4}	2.64×10^{-4}	8.58×10^{-5}
Maximum modulus of "relative" differences							2.99×10^{-4}	2.79×10^{-4}	1.07×10^{-3}	6.19×10^{-4}	2.00×10^{-4}

Table 6.6 Differences and "relative" differences between elements calculated from Z_{01} and Z_{S1} with "8-figure" arithmet

method		differences			"relative" differences		
element	approximate element value	Orchard	Saraga "incompatible"	Saraga "compatible"	Orchard	Saraga "incompatible"	Saraga "compatible"
C_1	1.040	-2×10^{-7}	0	-2×10^{-7}	-1.92×10^{-7}	0	-1.92×10^{-7}
C_2	0.1839	2.1×10^{-7}	2.4×10^{-7}	8.8×10^{-7}	1.14×10^{-6}	1.31×10^{-6}	4.79×10^{-6}
C_3	1.354	-2.63×10^{-3}	-1.08×10^{-3}	-3.05×10^{-3}	-1.94×10^{-3}	-7.97×10^{-4}	-2.25×10^{-3}
C_4	1.192	2.87×10^{-2}	8.40×10^{-3}	2.53×10^{-2}	2.41×10^{-2}	7.05×10^{-3}	2.12×10^{-2}
C_5	0.8403	-1.86×10^{-1}	-5.25×10^{-2}	-1.42×10^{-1}	-2.21×10^{-1}	-6.25×10^{-2}	-1.69×10^{-1}
C_6	1.715	3.00	1.11	2.48	1.75×10^{0}	6.45×10^{-1}	1.44×10^{0}
C_7	1.010	-0.285	-0.304	-0.312	-2.82×10^{-1}	-3.01×10^{-1}	-3.09×10^{-1}
C_8	0.7400	-0.118	-0.04	-0.103	-1.59×10^{-1}	-5.40×10^{-2}	-1.40×10^{-1}
Maximum modulus of "relative" differences					1.75	0.645	1.44

Table 6.7. Differences and "relative" differences between elements calculated from z_{O2} and z_{S2} with "single-length" arithmetic.

method		differences					"relative" differences				
		Orchard	Saraga "incompatible"	Saraga "compatible"			Orchard	Saraga "incompatible"	Saraga "compatible"		
element	approximation element value			(a)	(b)	(c)			(a)	(b)	(c)
c_2	0.1839	-1.19×10^{-7}	-2.62×10^{-7}	3.33×10^{-7}	-2.56×10^{-7}	4.74×10^{-7}	-6.46×10^{-7}	-1.43×10^{-6}	1.81×10^{-6}	-1.39×10^{-6}	2.58×10^{-6}
c_3	1.354	-5.60×10^{-6}	1.24×10^{-5}	-1.57×10^{-5}	1.21×10^{-5}	-2.24×10^{-5}	-4.14×10^{-6}	9.16×10^{-6}	-1.16×10^{-5}	8.92×10^{-6}	-1.65×10^{-5}
c_4	1.192	5.91×10^{-6}	-1.30×10^{-5}	1.65×10^{-5}	-1.27×10^{-5}	2.35×10^{-5}	4.96×10^{-6}	-1.09×10^{-5}	1.39×10^{-5}	-1.07×10^{-5}	1.97×10^{-5}
c_5	0.8403	-1.33×10^{-6}	2.93×10^{-6}	-3.71×10^{-6}	$+2.86 \times 10^{-6}$	-5.28×10^{-6}	-1.58×10^{-6}	3.48×10^{-6}	-4.42×10^{-6}	$+3.41 \times 10^{-6}$	-6.28×10^{-6}
c_6	1.715	7.93×10^{-7}	-1.86×10^{-6}	2.23×10^{-6}	-1.73×10^{-6}	3.25×10^{-6}	4.62×10^{-7}	-1.08×10^{-6}	1.30×10^{-6}	-1.01×10^{-6}	1.90×10^{-6}
c_7	1.010	-2.4×10^{-8}	8.3×10^{-8}	-7.4×10^{-8}	5.5×10^{-8}	-1.28×10^{-7}	-2.38×10^{-8}	8.2×10^{-8}	-7.3×10^{-8}	5.4×10^{-8}	-1.27×10^{-7}
c_8	0.7400	-5.0×10^{-10}	-1.85×10^{-9}	4.6×10^{-10}	5.1×10^{-10}	2.5×10^{-9}	-6.76×10^{-10}	-2.5×10^{-9}	6.22×10^{-10}	6.89×10^{-10}	3.3×10^{-9}
c_9	0.6837	3.2×10^{-10}	2.8×10^{-10}	-7×10^{-11}	-1.7×10^{-10}	-4.3×10^{-10}	$+4.68 \times 10^{-10}$	$+4.10 \times 10^{-10}$	-1×10^{-10}	-2.49×10^{-10}	-6.29×10^{-10}
Maximum modulus of "relative" differences							4.96×10^{-6}	1.09×10^{-5}	1.39×10^{-5}	1.07×10^{-5}	1.97×10^{-5}

Table 6.8. Differences and "relative" differences between elements calculated from Z_{O2} and Z_{S2} with "8-figure" arithmet

method		differences			"relative" differences		
element	approximate element value	Orchard	Saraga "incompatible"	Saraga "compatible"	Orchard	Saraga "incompatible"	Saraga "compatible"
C_2	0.1839	2.56×10^{-4}	4.42×10^{-4}	-5.36×10^{-5}	1.39×10^{-3}	2.40×10^{-3}	-2.92×10^{-4}
C_3	1.354	-1.42×10^{-2}	-5.50×10^{-2}	2.42×10^{-3}	-1.05×10^{-2}	-4.06×10^{-2}	1.79×10^{-3}
C_4	1.192	1.55×10^{-2}	6.71×10^{-2}	-2.53×10^{-3}	1.30×10^{-2}	5.63×10^{-2}	-2.13×10^{-3}
C_5	0.8403	-3.46×10^{-3}	-1.55×10^{-2}	5.45×10^{-4}	-4.11×10^{-3}	-1.84×10^{-2}	6.49×10^{-4}
C_6	1.715	2.01×10^{-3}	9.63×10^{-3}	-2.66×10^{-4}	1.17×10^{-3}	5.61×10^{-3}	-1.55×10^{-4}
C_7	1.010	-5.71×10^{-5}	-3.19×10^{-4}	-3.9×10^{-6}	-5.65×10^{-5}	-3.15×10^{-4}	-3.86×10^{-6}
C_8	0.7400	1.04×10^{-6}	-6×10^{-8}	1.62×10^{-6}	1.41×10^{-6}	-8.11×10^{-8}	2.19×10^{-6}
C_9	0.6837	-5.5×10^{-7}	3.8×10^{-7}	-4.1×10^{-7}	-8.04×10^{-7}	5.56×10^{-7}	-6.00×10^{-7}
Maximum modulus of "relative" differences					1.30×10^{-2}	5.63×10^{-2}	2.13×10^{-3}

Appendix 7. The square root subroutine SSQRT.

The square root subroutine SSQRT was written so that the computer would find the square roots of numbers using arithmetic of a chosen reduced accuracy. It is based on the well-known iterative formula

$$b_{n+1} = \frac{\left(\frac{a}{b_n} + b_n\right)}{2}$$

where the square root of a positive number, a, is required and the improved approximation b_{n+1} is calculated from the previous approximation b_n. To obtain the most accurate answer working with the prescribed accuracy the following test is used.* The value of the expression $|a-b^2_{n+1}|$ is calculated for the latest approximation b_{n+1} and compared with that for the previous approximation $|a-b^2_n|$. The process is repeated until

$$|a-b^2_{n+1}| > |a-b^2_n|$$

and then b_n is taken for the value of $\sqrt{a}$ as it is the most accurate root that can be obtained with the accuracy prescribed for the arithmetical operations.

---

* This text was advised years ago by Dr.Wilkinson (N.P.L.) in the case of finding the most accurate root of a polynomial equation by Bairstow's method.