Asymptotic gain feedback model

Why do we need a feedback model if we can analyse our circuits using MNA and SPICE?

We need it for designing feedback circuits!

How does that work?

Two-step design approach for negative feedback amplifiers

1. Determine the requirements for the amplifier's transmission-1 matrix parameters A, B, C and D, and design the amplifier type and its ideal gain using:
   - load voltage (output parallel) sensing to fix A and/or C
   - load current (output series) sensing to fix B and/or D
   - a feedback network to generate a copy of the source signal from the load signal
   - source voltage (input series) comparison to fix A and/or B
   - source current (input parallel) comparison or to fix C and/or D
   - a nullor as ideal controller (error amplifier)
     - nullator sets the zero condition for comparison
     - norator provides the extra degree of freedom to satisfy this condition
2. Design a controller (error amplifier) such that the difference between the gain and the ideal gain is small enough.

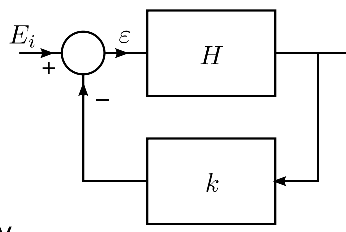
Still don't get it: you just increase the gain of the controller until the error is small enough!

Good luck with this trial-and-horror method!
Multiple cascaded amplifier stages may give a lot of gain AND a lot of phase shift, which may turn the negative (corrective) feedback into positive (regenerative) feedback!

With the aid of a feedback model we can investigate in which way and to which extend the performance aspects of the the amplifier depend on those of the controller.
By knowing this, we can derive performance specifications for the controller from those of the amplifier

Wow, that sounds like a great idea; must have been a smart person who discovered this!

Black presented a basic feedback model:



$$\frac{E_o}{E_i} = \frac{1}{k}\left(\frac{Hk}{1+Hk}\right)$$

E_i: Input quantity
E_o: Ouput quantity
H: Controller gain
k: Feedback gain = reciprocal value of ideal gain
Hk: Loop gain = measure for correspondence between the ideal gain and the gain

Okay, looks great let's use it!

All models are wrong, but some are useful!
(G. Box)
Models should be as simple as possible, but not simpler!
(A. Einstein)

The idea is great: if the loop gain approaches infinity, the gain approaches the gain.
However, in many cases this model is far too simple.

1. The loop gain is not the only contributer to the mismatch between the ideal gain and the actual gain:
   - Direct transfer through the feedback network and/or controller
   - Parasitic impedances in series of in parallel with elements of the feedback network
   - Parasitic impedances in series or in parallel with the signal source and/or the load
   - Finite common-mode impedance of a floating controller port
   - Finite common-mode rejection in the case of a floating controller port
2. As a result of interactions between the controller, the source, the load and the feedback network, the loop gain cannot easily be written as the product of the controller gain H and the feedback gain k.

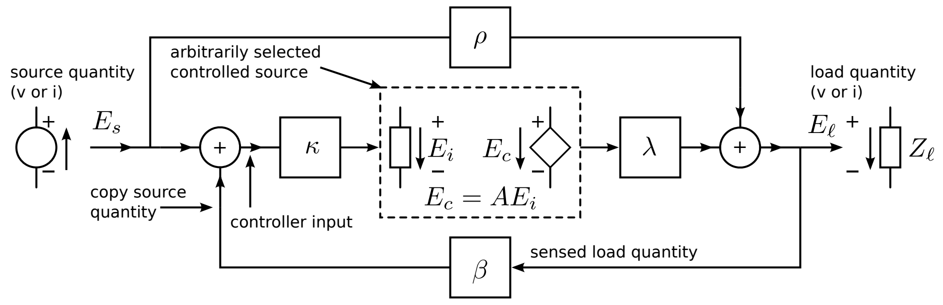
Hm, and now?

Use a better model! TUDelft

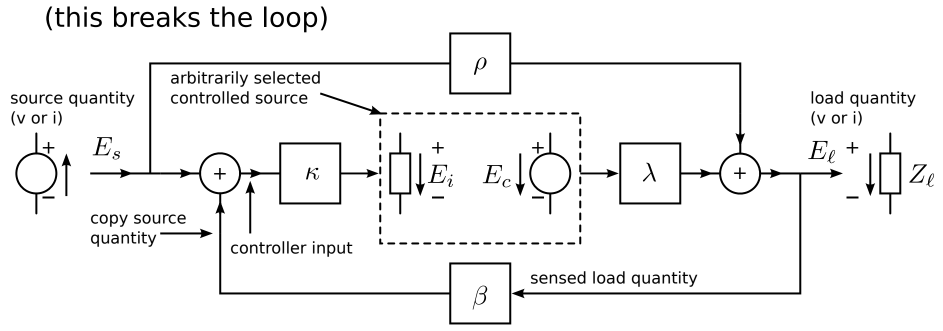
Please explain!

Superposition model

1. In a network comprising feedback, the gain of an arbitrarily selected controlled source is taken as loop gain reference



2. This controlled source is replaced with an independent source (this breaks the loop)



3. We now have a network with two independent sources and two responses and write:

$$\begin{pmatrix} E_\ell \\ E_i \end{pmatrix} = \begin{pmatrix} \rho & \lambda \\ \kappa & \lambda\beta\kappa \end{pmatrix} \begin{pmatrix} E_s \\ E_c \end{pmatrix}$$

4. Calculate the source-to-load transfer (gain) using: $E_c = AE_i$

$$A_f = \frac{E_\ell}{E_s} = \rho + \frac{A\lambda\kappa}{1-A\lambda\beta\kappa}$$

5. Calculate the asymptotic gain: $A_{f\infty} \triangleq \lim_{A\beta\to\infty} A_f$

$$A_{f\infty} = \rho - \frac{1}{\beta}$$

6. Rewrite the expression for the gain:

$$A_f = A_{f\infty}\frac{-L}{1-L} + \frac{\rho}{1-L}, \text{ where } L = \lambda A\beta\kappa$$

This is the asymptotic gain model

Looks like Black's model again!

Only identical if: $\rho = 0$, $\kappa = 1$, $\lambda = 1$ and if: $\beta = -\frac{1}{k}$

The latter one requires proper selection of the loop gain reference!

Quite confusing; what's the fun?

A_f as obtained from MNA!

It becomes very useful if the asymptotic gain equals the ideal gain!
The latter one one has been designed in the first design step: the design of the feedback network.

This is the case if the direct transfer equals zero and if the loop gain reference is selected such that if it is replaced with a nullor the controller becomes a nullor

$$\underbrace{A_f}_{\text{Gain}} \approx \underbrace{A_i}_{\text{Ideal gain}} \underbrace{\frac{\overbrace{-L}^{\text{Loop gain}}}{1-L}}_{\text{Servo function}}$$

We then have our two-step design:

1. Design of the ideal gain = design feedback network
2. Design of the controller such that its contribution to the loop gain is large enough (the servo function approaches unity).

What about the other terms κ, λ

They can provide information about the influence of the source and the load impedance on the loop gain:
- If E_i is taken at the input of the controller (at the nullator)
- and E_c is taken at the output of the controller (at the norator)
  - the source impedance plays no role in the loop gain if: $\kappa = 1$ and dimensionless
  - the load impedance plays no role in the loop gain if: $\lambda = 1$ and dimensionless

Does the ideal gain always equal the asymptotic gain?

No! (All models are wrong but ...)

The asymptotic gain equals the ideal gain if the controller obtains nullor properties when the selected controlled source is replaced with a nullor.

And what if this is not the case?

Then, the loopgain is NOT the unique measure for the correspondence between the gain and the ideal gain.

So this is all about the two-step design?!

Seems you've got the message!

So what's the problem?

1. The the controller does seldom behave as a natural two-port
2. There often exist local feedback loops in the controller
3. The loop gain reference should not be selected inside such a local loop

The ideal gain is obtained with a natural two-port (nullor) as controller

According to network theory, a network can always be modeled as a two-port if:
1. It has only three nodes
2. The ports are terminates with one-ports
3. The network is a natural two-port

So this model turns out to be useless anyway?

No, its very useful!
1. The source-to-load transfer calculated from the asymptotic gain model equals the one calculated from modified nodal analysis
2. Correspondence between the asymptotic gain and the ideal gain tells us about the meaning of the loop gain; no other model provides such a check!
3. At an early stage of the design this correspondence can be made very good by using relative simple device models that provide sufficient design information.

Sounds nice but ...

Examples will help you

Pffff, finally!

Example 10.1: Model of Black
Controller type matches amplifier type:
Hk can easily be identified in the loop gain expression.

Example 10.2: Model of Black
Controller type opposite to amplifier type:
Hk cannot easily be identified in the loop gain expression;
source and load impedance enter the expression.

Example 10.3: Asymptotic-gain model
Selection of loop gain reference such that the ideal gain corresponds with the asymptotic gain.

Example 10.4: Asymptotic-gain model
Influence of parasitic impedance on the asymptotic gain (two-port conditions).

Section 10.3.4: Hand calculation of the loop gain of a passive feedback voltage amplifier.

But there is more

We have seen that:
1. Parallel feedback at an amplifier port reduces the port impedance
   - In single-loop feedback amplifiers the ideal value of the port impedance will become zero
2. Series feedback at an amplifier port increases the port impedance
   - In single-loop feedback amplifiers the ideal value of the port impedance will become infinite

The superposition model and the asymptotic gain model can be used to study the effect of feedback on the port impedance

So what's use of that?

1. In multiple-loop negative feedback amplifiers it will help us to find requirements for the loop gain under two conditions:
   a. The port is left open:
      parallel feedback is effective but series feedback is not
   b. The port is shorted:
      series feedback is effective but parallel feedback is not
   Both loop gains are a measure for the mismatch between the port impedance and its ideal value which is determined by the feedback networks.
2. Manufacturers of operational amplifiers often specify the 'closed-loop' output impedance of an OpAmp, rather than its output impedance.
   With the aid of the superposition model we are able to relate one to the other.
   See section 10.3.6, example 10.8

$$Z_f = \rho\frac{1-L_{sc}}{1-L_o}$$

Z_f = port impedance with feedback
ρ = port impedance with loop gain reference set to zero (no feedback)
L_{sc} = loop gain with the port shorted
L_o = loop gain with the port left open

And what about SLiCAP?

SLiCAP can be used for the analysis and design of negative feedback amplifiers

Use gainType() to define the type of transfer you want to evaluate:
- 'gain': source-to-load transfer
  requires definition of:
  * signal source
  * detector
- 'asymptotic': source-to-load transfer (loop gain reference replaced with a nullor)
  requires definition of:
  * signal source
  * detector
  * loop gain reference
- 'direct': direct transfer from source to load (gain of loop gain reference set to zero)
  requires definition of:
  * signal source
  * detector
  * loop gain reference
- 'loopgain': loop gain as defined in the asymptotic gain model
  requires definition of:
  * loop gain reference
- 'servo': servo function as defined in the asymptotic gain model
  requires definition of:
  * loop gain reference

Associated functions:

source	Define the signal source
detector	Define the signal detector
lgRef	Define the loop gain reference
gainType	Define the gain type
controlled	Returns a list of controlled sources that can be used as loop gain reference
depVars	Returns a list of nodal voltages and branch currents that can be used as detector
indepVars	Returns a list of independent sources that can be used as signal source

SLiCAP help:

For opening the SLiCAP documentation, type 'Help' in the MATLAB Command Window

For displaying help for a specific function, type 'help <functionName>' in the MATLAB Command Window