THE HUBER CONCEPT IN DEVICE MODELING, CIRCUIT DIAGNOSIS AND DESIGN CENTERING

J.W. Bandler, S.H. Chen, R.M. Biernacki
and K. Madsen

SOS-93-16-V

May 1994

THE HUBER CONCEPT IN DEVICE MODELING, CIRCUIT DIAGNOSIS AND DESIGN CENTERING

J.W. Bandler, S.H. Chen, R.M. Biernacki

Simulation Optimization Systems Research Laboratory
and Department of Electrical and Computer Engineering
McMaster University, Hamilton, Canada L8S 4L7

K. Madsen

Institute of Mathematical Modelling
The Technical University of Denmark
DK-2800 Lyngby, Denmark

Introduction

circuit optimization must take into account model/measurement/statistical errors, variations and uncertainties

least-squares (ℓ_2) solutions are notoriously susceptible to the influence of gross errors: just a few "wild" data points can alter the results significantly

the ℓ_1 method is robust against gross errors; however, it inappropriately treats small variations in the data

neither the ℓ_1 nor ℓ_2 alone is capable of providing solutions which are robust against large errors *and* flexible w.r.t. small variations in the data

the Huber solution can provide a smooth model from data which contains many small variations and such a model is also robust against gross errors

implemented in the CAD system OSA90/hope which was used to produce the examples in this presentation

The Huber Function

$$\rho_k(f) = \begin{cases} f^2/2 & \text{if } |f| \le k \\ k\,|f| - k^2/2 & \text{if } |f| > k \end{cases}$$

f represents an error function

$k > 0$ is a threshold separating "large" and "small" errors

the definition of ρ_k ensures a smooth transition at k

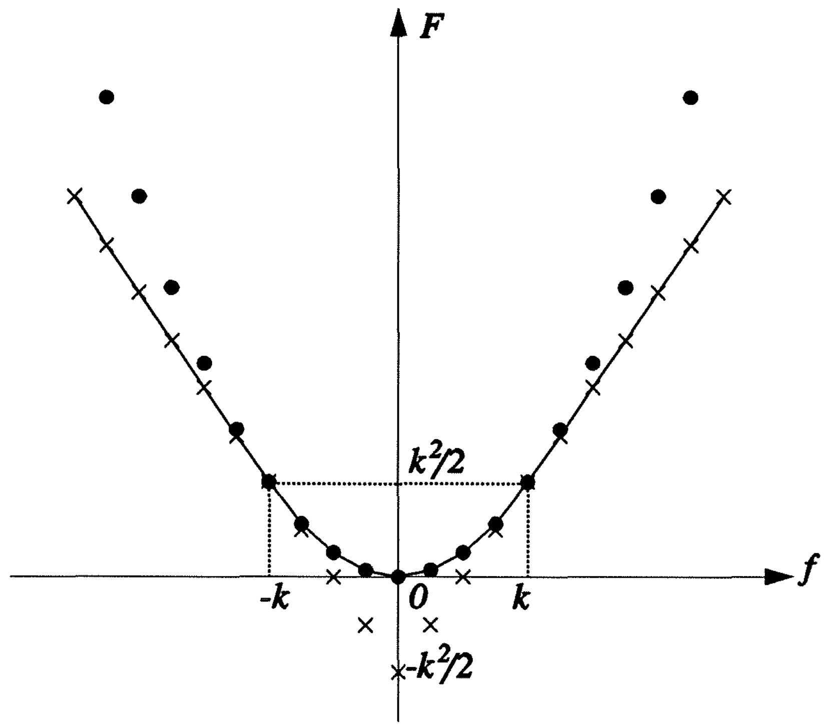
The Huber Norm

$$\sum_{j=1}^{m} \rho_k(f_j(\phi))$$

a hybrid of the ℓ_2 and the ℓ_1 norms

Huber Function as a Hybrid of ℓ_1 and ℓ_2

the Huber, ℓ_1 and ℓ_2 objective functions in the one-dimensional case



the large errors are treated in the ℓ_1 sense and the small errors are measured in terms of least squares

by selecting k we can control the proportion of errors treated in the ℓ_1 or ℓ_2 sense

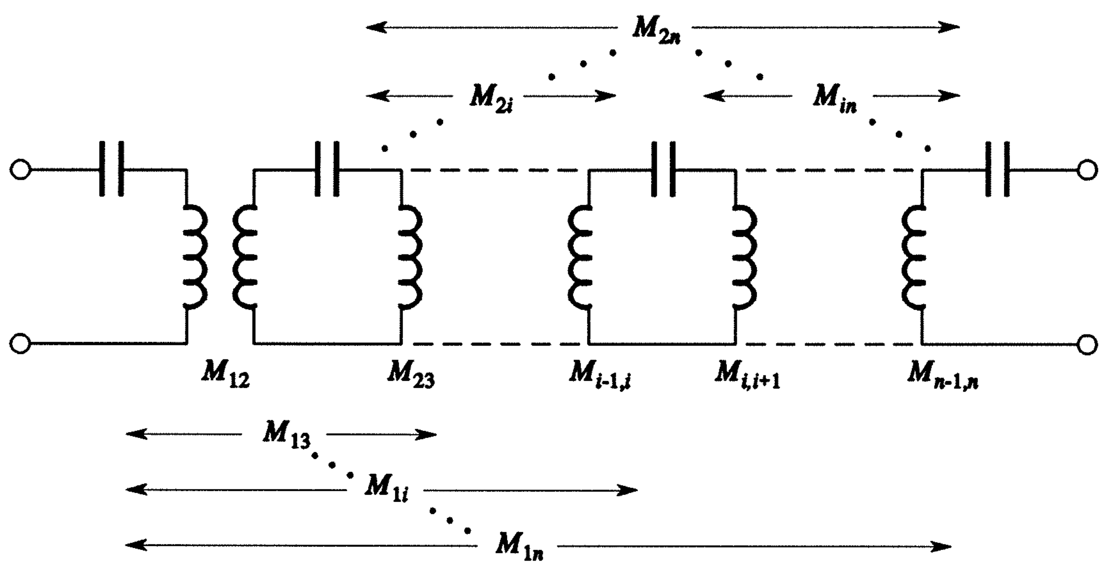
One-Sided Huber Function

we extend the Huber concept by introducing a "one-sided" Huber function for design optimization with upper and/or lower specifications

we define the "one-sided" Huber function as

$$\rho_k^+(f) = \begin{cases} 0 & \text{if } f \le 0 \\ \rho_k(f) & \text{if } f > 0 \end{cases}$$

A 6th Order Multicavity Filter



the input reflection coefficient is used as simulated measurement

two large errors are deliberately introduced into data

the task is to identify the parameters from the contaminated data

Results of Parameter Identification for the Multicavity Filter - Case A

the two large errors are the only errors contained in the data

Couplings	M_{12}	M_{45}	M_{16}
Actual Values	0.859956	0.526602	0.087293
Starting Point	0.819006	0.511264	0.093863
ℓ_2	-11%	7.3%	278%
ℓ_1	0.05%	-0.06%	-0.01%
Huber	0.02%	0.01%	-1.2%

the ℓ_2 solution is hopelessly corrupted by the wild data points

Results of Parameter Identification for the Multicavity Filter - Case B

the data is truncated to the first two significant digits to emulate the limited accuracy of measurement equipment

Couplings	M_{12}	M_{45}	M_{16}
Actual Values	0.859956	0.526602	0.087293
Starting Point	0.819006	0.511264	0.093863
ℓ_1	0.51%	-2.9%	-14%
Huber	0.15%	-0.01%	-8.3%

ℓ_1 is more affected by small variations in the data

Huber solution less affected by small variations in the data

Results of Parameter Identification for the Multicavity Filter - Case C

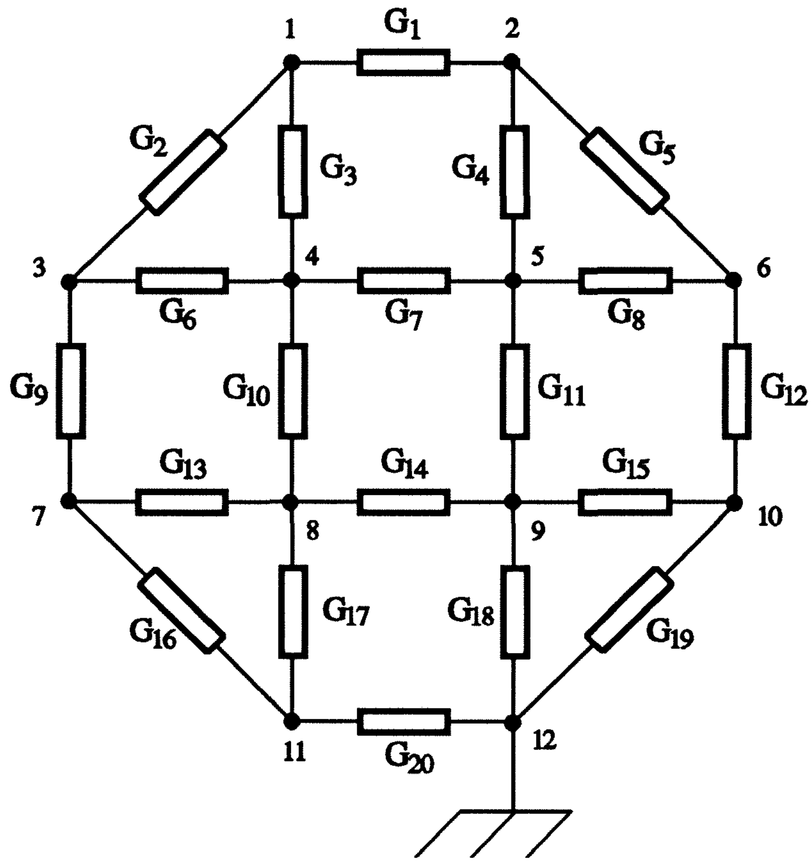
small errors randomly generated from the uniform distribution [-0.01 0.01] are introduced into the data

Couplings	M_{12}	M_{45}	M_{16}
Actual Values	0.859956	0.526602	0.087293
Starting Point	0.819006	0.511264	0.093863
ℓ_1	1.8%	-4.1%	-43%
Huber	0.41%	0.04%	-27%

ℓ_1 is more affected by small variations in the data

Huber solution less affected by small variations in the data

A Resistive Mesh Circuit



used to demonstrate the ℓ_1 approach to analog fault location

we present new results which take into account data truncation errors representing limited accuracy of measurement equipment

Analog Fault Location of the Resistive Mesh Circuit

ℓ_1 optimization attempts to suppress as many parameter deviations as possible to exactly zero

this may lead to an incorrect solution

two faults were assumed, namely G_2 and G_{18}

simulated node voltage measurements were generated at the accessible nodes

these voltages were truncated to the first two significant digits

Results of Fault Location of the Resistive Mesh Circuit

Element	Nominal Value	Actual Value	Percentage Deviation		
			Actual	ℓ_1	Huber
G_2	1.0	0.50	-50.0	-47.55	-54.40
G_3	1.0	1.05	5.0	-25.45	-3.68
G_{16}	1.0	0.95	-5.0	-20.24	-3.53
G_{17}	1.0	1.05	5.0	0.00	-0.81
G_{18}	1.0	0.50	-50.0	-8.90	-49.97
G_{19}	1.0	0.95	-5.0	-25.32	-4.74
G_{20}	1.0	0.95	-5.0	-20.73	-5.98

the nominal parameter values are used as the starting point for optimization

ℓ_1 optimization fails to isolate the faults

Huber optimization successfully isolates the faults

Robustness Against "Bad" Starting Points in Optimization

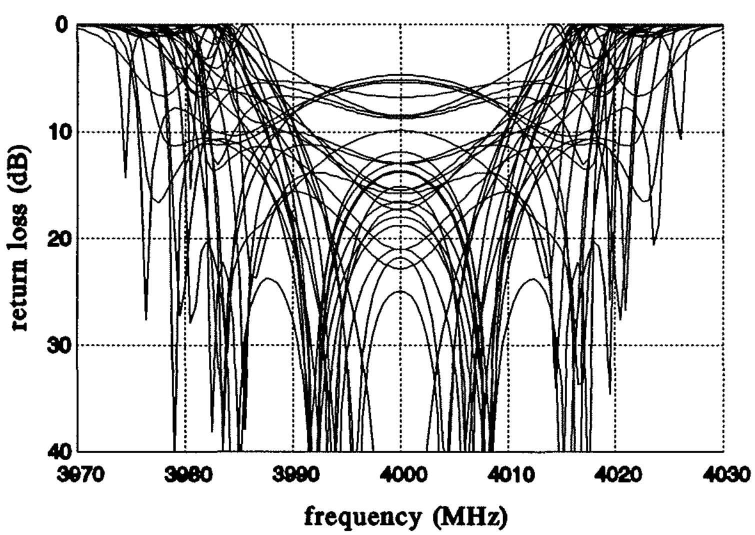
we show that the one-sided Huber function can be used in a "preprocessing" optimization to overcome bad starting points

6th-order multicavity filter

30 starting points generated using uniform distribution centered at a "good" starting point

±30% spread of the parameter values

the input return loss of the filter at the 30 starting points



One-sided Huber Preprocessing of Arbitrary Starting Points

from a "bad" starting point, a minimax optimizer can be trapped by the initial large errors

we have exploited the potential of using one-sided Huber preprocessing to overcome bad starting points in large-scale multiplexer optimization

here we expand our investigation by testing several starting points for optimization

we compare minimax optimization with and without one-sided Huber preprocessing from these randomly generated starting points

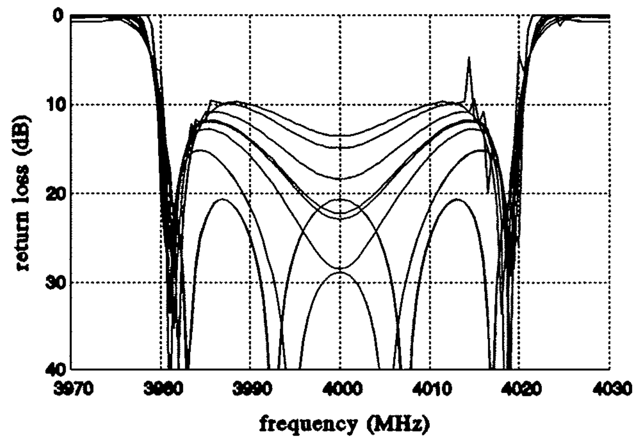
from each starting point, we perform:

(1) direct minimax optimization

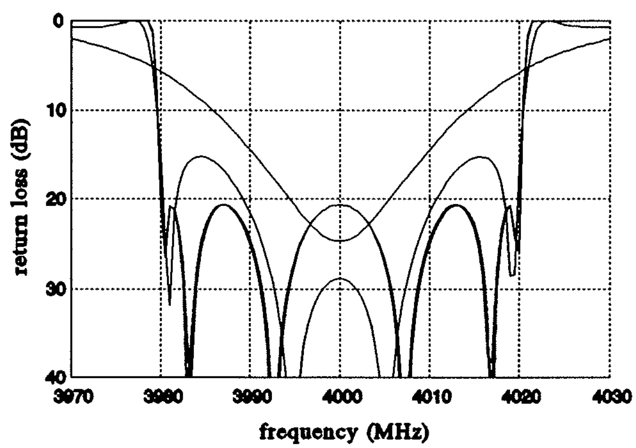
(2) one-sided Huber optimization (preprocessing) followed by minimax optimization

Results of One-sided Huber Preprocessing

without Huber preprocessing



with Huber preprocessing



one-sided Huber preprocessing produces more focused results

Statistical Device Modeling

parameter extraction/statistical postprocessing

first, we extract model parameters for individual devices from device measurements

then the sample of model parameters is postprocessed to estimate the statistics

for postprocessing we normally apply least-squares estimators

wild points severely degrade the least-squares estimates; in our earlier work using the ℓ_2 estimator the wild points had to be manually excluded

the Huber function can be used as an automatic robust statistical estimator in place of least-squares estimators

applying Huber estimators to the same data we obtained similar results but without excluding any points

Statistical Estimation

the error functions to estimate mean values

$$f_j(\bar{\phi}) = \bar{\phi} - \phi^j$$

the error functions to estimate standard deviations

$$f_j(V_\phi) = V_\phi - (\phi^j - \bar{\phi})^2$$

where

ϕ^j	the extracted value of a parameter of the jth device
j	1, 2, ..., N
N	the total number of devices
V_ϕ	the estimated variance from which we can calculate the standard deviation σ_ϕ

One-sided Huber Formulation for Yield Optimization

we present a one-sided Huber approach to yield optimization of linear and nonlinear circuits

we consider a number of statistical outcomes of circuit parameters denoted by $\boldsymbol{\phi}^i$

for each outcome we create a generalized ℓ_p function $v(\boldsymbol{\phi}^i)$

we have formulated yield optimization as a one-sided ℓ_1 problem (*Bandler and Chen, 1988*)

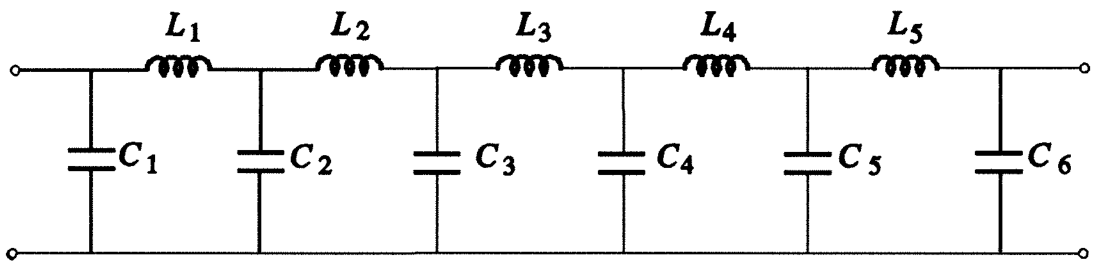
here we formulate yield optimization as a one-sided Huber problem: the objective function is defined as

$$U(\boldsymbol{\phi}^0) = \sum_{i=1}^{N} \rho_k^+(\alpha_i v(\boldsymbol{\phi}^i))$$

where

- $\boldsymbol{\phi}^0$ the nominal circuit parameters
- α_i a positive multiplier associated with the ith outcome
- N the total number of outcomes

Yield Optimization of an LC Filter



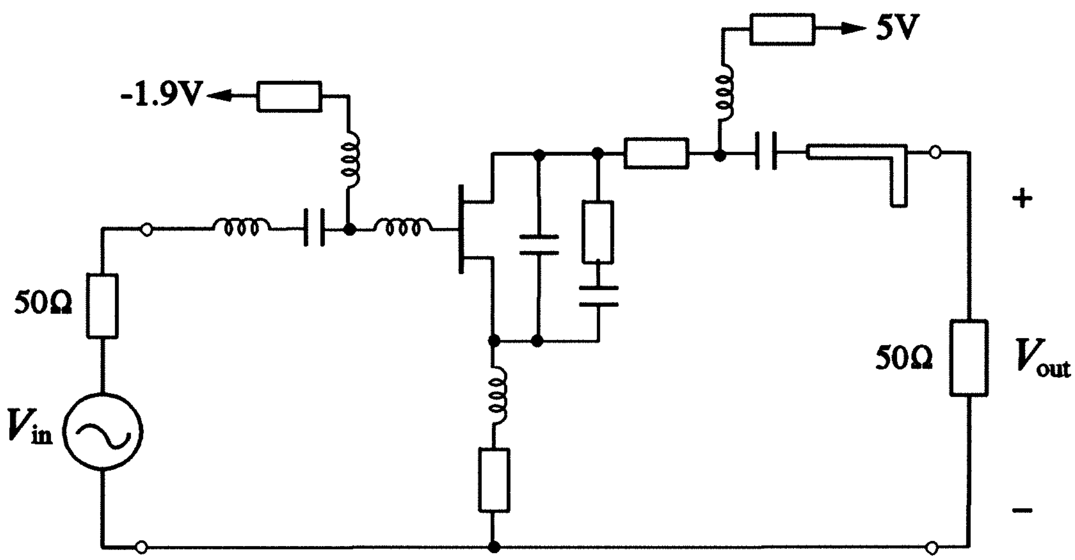
one-sided ℓ_1 method needed 160 CPU seconds (11 iterations)

one-sided Huber yield optimization with k=0.2 finished in 123 CPU seconds (9 iterations)

both optimizations produced 75% yield

Sun SPARCstation 10

Yield Optimization of a Nonlinear Frequency Doubler



uniform distributions in the linear matching subcircuits; normal distributions for the intrinsic FET

one-sided ℓ_1 centering finished in 17 iterations and 337 CPU seconds and increased yield from 28% to 76%

one-sided Huber centering finished in 29 iterations and 574 CPU seconds and increased yield from 28% to 77%

Sun SPARCstation 10

Conclusions

exciting applications of a novel Huber approach to

- parameter identification
- analog fault location
- preprocessing of arbitrary starting points
- statistical modeling
- statistical design centering

the Huber approach demonstrates robustness and consistency in the presence of large and small errors, both deterministic and statistical

it combines the advantages of ℓ_1 and ℓ_2 techniques and overcomes their respective shortcomings

the Huber concept is consistent with practical engineering intuition

the Huber method will have a far-reaching and profound impact on modeling, design, design validation, fault diagnosis and statistical modeling and design