JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 33, No. 1, January–February 2010

Optimization of Human Perception Modeling Using Interval Analysis

E. van Kampen,* P. M. T. Zaal,† E. de Weerdt,‡ Q. P. Chu,§ and J. A. Mulder¶
Delft University of Technology, 2600 GB Delft, The Netherlands

DOI: 10.2514/1.46318

Estimating multimodal pilot model parameters from experimental data requires solving a global nonlinear optimization problem with many local minimums. With traditional parameter estimation techniques, the solution depends on the initial parameter estimate and a local optimum can be found instead of the global optimum. In this paper, the parameter optimization is performed by using the theory of interval analysis, which describes the properties of intervals of numbers instead of crisp numbers. Interval analysis has been shown to be an excellent tool for global nonlinear optimization and it can guarantee that the global minimum of the cost function is found. The interval optimization method is applied to data from an experiment investigating the role of optic flow and the influence of physical motion cues during control of self-motion. A comparison between gradient-based and interval optimization shows that the interval method can find the global minimum of the cost function, resulting in the optimal set of model parameters, whereas gradient-based methods often converge to a local minimum.

Nomenclature

e	=	tracking error signal, rad
f_d	=	disturbance forcing function, rad
f_t	=	target forcing function, rad
H_c	=	controlled system dynamics
$H(j\omega)$	=	frequency response function
H_{nm}	=	neuromuscular system dynamics
H_{oto}	=	otolith dynamics
H_{pe}	=	pilot visual error response
$H_{p\chi}$	=	pilot physical motion response
J	=	cost function
j	=	imaginary unit
K_m	=	motion perception gain
K_v	=	visual perception gain
n	=	pilot remnant signal, rad
P	=	parameter vector
T_{oto1}	=	otolith lead constant, s
T_{oto2}	=	otolith lag constant, s
T_v	=	visual lead time constant, s
u	=	pilot control signal, rad
V	=	forward speed, m/s
$[x]$	=	interval variable x
ϵ_p	=	minimum parameter box width
ϵ_J	=	minimum cost function box width
ζ_{nm}	=	neuromuscular damping
τ_v	=	visual perception time delay, s
τ_m	=	motion perception time delay, s
χ	=	track angle, rad
ω	=	frequency, rad/s
ω_{nm}	=	neuromuscular frequency, rad/s

I. Introduction

MODELING human perception and control is an important tool for the design of simulators and control systems [1–6]. Often the goal of human perception modeling is not to replicate the output of the human controller for a specific input, which can be done using any nondescriptive nonlinear and even adaptive model such as a neural network, but rather to describe and understand human control behavior in terms of recognizable elements such as gains and time delays. In either case, these models consist of parameters that need to be optimized to fit the estimated model output to the real output.

For a number of identification methods in the time and frequency domains, this optimization is nonlinear, characterized by the existence of multiple local minimums of the cost function [7]. Currently, linear gradient-based optimization methods are used to solve these nonlinear optimization problems, with the major drawback that one cannot guarantee that the global optimal solution has been found. Different initial values for the parameters can lead to different solutions, indicating the existence of multiple local minimums. Zaal et al. [8] apply a maximum likelihood estimation complemented with genetic algorithms to cope with the inherent nonlinearities. The genetic algorithms are used to find the initial parameter estimates for an unconstrained Gauss–Newton optimization. The chances of finding the global minimum of the cost function are greatly increased by using the genetic algorithms, but some suboptimal local minimums are still found, and it is not possible to guarantee that the global minimum will be found.

This paper introduces a global nonlinear optimization method based on the theory of interval analysis [9–11] into the field of human perception modeling. This method increases the performance of the existing models, but also opens the path for new types of descriptive models that include adaptive elements or nonlinearities such as thresholds. Interval analysis was developed by Moore [9] in the 1960s to study error propagation. It describes the mathematical properties of intervals of numbers instead of regular numbers and it has shown to be an excellent tool for global nonlinear optimization in a wide range of applications [12–15]. An experiment performed in the SIMONA Research Simulator investigating the role of optic flow and the influence of physical motion cues during control of self-motion is used to test this interval optimization procedure [16].

Presented as Paper 7108 at the AIAA Modeling and Simulation Technologies Conference and Exhibit, Honolulu, HI, 18–21 August 2008; received 13 July 2009; revision received 15 September 2009; accepted for publication 17 September 2009. Copyright © 2009 by Erik-Jan van Kampen. Published by the American Institute of Aeronautics and Astronautics, Inc., with permission. Copies of this paper may be made for personal or internal use, on condition that the copier pay the $10.00 per-copy fee to the Copyright Clearance Center, Inc., 222 Rosewood Drive, Danvers, MA 01923; include the code 0731-5090/10 and $10.00 in correspondence with the CCC.

*Ph.D. Student, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058; e.vankampen@tudelft.nl. Student Member AIAA.

†Ph.D. Student, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058; p.m.t.zaal@tudelft.nl. Student Member AIAA.

‡Ph.D. Student, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058; e.deweerdt@tudelft.nl. Student Member AIAA.

§Associate Professor, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058; q.p.chu@tudelft.nl. Member AIAA.

¶Professor, Control and Simulation Division, Faculty of Aerospace Engineering, P.O. Box 5058; j.a.mulder@tudelft.nl. Member AIAA.

First, this paper will discuss the parameter estimation problem in human perception modeling (Sec. II), including current techniques and drawbacks. Next, Sec. III introduces interval analysis, discussing the basic theory of interval arithmetic and the interval branch and bound optimization method. In Sec. IV, two examples are given of a simple parameter estimation problem and the differences between gradient-based algorithms and the interval algorithm are highlighted. Next, we will apply these methods to an optimization problem in the field of human perception modeling, comparing gradient-based optimization methods with interval optimization methods. The subsequent section describes the experiment that has been chosen as a testbed for the interval algorithm and gives the optimization results for this identification problem, followed by the conclusions in Sec. VI.

II. Parameter Estimation Problem in Human Perception Modeling

Although humans are nonlinear biological systems, in carefully designed control tasks and under constant conditions, they can be modeled by linear response functions and a remnant signal that accounts for nonlinear behavior [1]. In an aircraft or simulator, a pilot perceives different cues about the current state and the desired state of the controlled dynamics and uses this information to give a proper control action. Cues about the current state of the system dynamics can be central visual, peripheral visual, vestibular, proprioceptive, and somatosensory cues. The control output of the pilot changes the state of the system dynamics and the resulting cues, making the task a multiloop closed-loop control task.

A. Multimodal Pilot Model Identification

The closed-loop control task used in this paper to determine the performance of the interval analysis optimization method is given in Fig. 1 and is taken from previous research [16]. A pilot is controlling the track angle χ of a generic cart with system dynamics H_c. A target forcing function f_t represents the desired track angle and has to be followed by the pilot. A compensatory display shows the error e, which is the difference between the current and the desired track angle. Information about the current state of the system is perceived directly by the pilot via physical motion cues. The total pilot control signal u consists of the output of a linear error response function H_{pe}, the output of a linear state response function $H_{p\chi}$, and a remnant signal n that accounts for the nonlinear behavior.

The linear response functions are parametrized by gains and time constants, that is, the pilot equalization, and vestibular sensor dynamics, time delays, and neuromuscular dynamics. Different techniques are available to estimate the parameters of a multichannel pilot model from experimental data. The identifiability of the parameters depends on the type of control task, the control strategy of the pilot, the pilot model structure, and the parameter estimation technique.

Human operators adapt their control behavior to the controlled dynamics in such a way that the open-loop dynamics in the crossover region can be described by a single integrator and a time delay [1]. The system dynamics used in this research for controlling the track angle of a generic cart are given by

$$H_c = \frac{32}{s(s+8)} \tag{1}$$

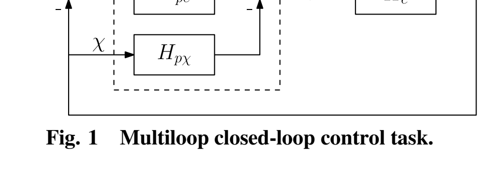


Fig. 1 Multiloop closed-loop control task.

These controlled dynamics are a single integrator below a frequency of 8 rad/s and above this frequency a double integrator. With typical crossover frequencies of around 3 rad/s, the pilot is effectively controlling single integrator dynamics and is not required to generate lead. However, to achieve a higher performance, especially when the visual information is not very accurate, lead will be generated visually or through the vestibular system of the pilot [16]. Based on McRuer and Jex's precision model [1] and van der Vaart's multichannel model [17], the linear response functions (see Fig. 1) are then parametrized by

$$H_{\mathrm{pe}}(j\omega) = K_v(1 + j\omega T_v)e^{-j\omega\tau_v}H_{\mathrm{nm}}(j\omega) \tag{2}$$

and

$$H_{p\chi}(j\omega) = (j\omega)VH_{\mathrm{oto}}(j\omega)K_m e^{-j\omega\tau_m}H_{\mathrm{nm}}(j\omega) \tag{3}$$

In the visual perception channel H_{pe}, K_v is the visual perception gain, T_v the visual lead time constant, and τ_v the visual perception time delay. The control action of the pilot is limited by the neuromuscular dynamics H_{nm}, given by

$$H_{\mathrm{nm}}(j\omega) = \frac{\omega_{\mathrm{nm}}^2}{\omega_{\mathrm{nm}}^2 + 2\zeta_{\mathrm{nm}}\omega_{\mathrm{nm}}j\omega + (j\omega)^2} \tag{4}$$

with $\zeta_{\mathrm{nm}} = 0.18$ the neuromuscular damping and $\omega_{\mathrm{nm}} = 11$ the neuromuscular frequency. In this research, these values are fixed to keep the parameter estimation problem solvable in a reasonable amount of time.

For the control task discussed here, a change in track angle induces a lateral acceleration approximated by $\dot{\chi}V$, where V is the forward speed of the system dynamics. The lateral accelerations are the input to the otoliths of the pilot. The physical motion perception channel $H_{p\chi}$ includes the dynamics of the otoliths H_{oto}, the motion perception gain K_m, and a motion perception time delay τ_m. The otolith dynamics in the physical motion perception channel are given by

$$H_{\mathrm{oto}}(j\omega) = \frac{1 + j\omega T_{\mathrm{oto1}}}{1 + j\omega T_{\mathrm{oto2}}} \tag{5}$$

with $T_{\mathrm{oto1}} = 1.0$ s and $T_{\mathrm{oto2}} = 0.5$ s, the lead and lag time constants of the otoliths, respectively. These values are determined in previous research and fixed in the parameter estimation procedure [18]. Although there are more human sensors that sense linear accelerations, for example, proprioceptive and somatosensory sensors, they are left unmodeled in this research. This reduces the amount of parameters and results in a solvable and well-posed parameter estimation problem. The final parameter vector contains five parameters to be estimated:

$$P = [K_v \quad T_v \quad \tau_v \quad K_m \quad \tau_m]^T \tag{6}$$

B. Limitations of Current Parameter Estimation Techniques

In earlier research, parameter estimation techniques for pilot modeling were used that required two steps to estimate the parameters of a multimodal pilot model [3,4,19,20]. In the first step, nonparametric frequency response functions are estimated from the time-domain data. The nonparametric frequency response functions can be estimated by using Fourier coefficients or linear time-invariant models [e.g., autoregressive exogenous (ARX) models] and are used in the second step to fit a multimodal pilot model by adjusting the parameters. To fit the pilot model, a cost function is used that minimizes the difference between the identified frequency response functions from the first step and the frequency response functions from the pilot model, weighted by the variance of the estimate [20].

Depending on the control task and the structure of the pilot model, in many cases, the multimodal pilot models are able to model the total pilot response with different solutions of the parameter vector sufficiently well. For example, in a pure disturbance rejection task, if controlled dynamics require the pilot to generate lead, this can be modeled by the visual lead constant in the visual perception channel or by vestibular lead from the physical motion perception channel. Also, the time delays introduce many local minimums in the optimization problem. This results in a highly nonlinear nonconvex optimization problem, which is difficult to solve using the traditional gradient methods. Finding the global minimum is not guaranteed.

Additionally, for the identification of multiple frequency response functions, the two-step method requires inserting as many deterministic test signals at different locations in the control loop (e.g., a target and a disturbance) as the number of response functions to be identified. Also, the forcing functions need to be carefully designed, keeping in mind the requirements for identifiability of the frequency responses and the limitations of the pilot. This means the two-step method poses strong constraints on the design of the control task and often leads to tasks that are less representative of actual piloting tasks.

A new identification technique for estimating the parameters of a multimodal pilot model is based on maximum likelihood estimation (MLE) [21–23]. Using MLE, the parameters of a multimodal pilot model are directly estimated from the time-domain data using only one step. This increases the accuracy of the estimated parameters. Additionally, MLE significantly reduces the requirements on the forcing functions giving experimenters more freedom in the design of the control task. It is even possible to estimate the parameters of a two-channel model using only one forcing function [24]. This allows for control tasks that are more representative for actual piloting tasks.

Although MLE solves a lot of problems introduced by the two-step method, the multimodal pilot model in state-space form, needed for MLE, is highly nonlinear in the pilot model parameters, resulting in a highly nonlinear optimization problem with many local minimums. To increase the probability of finding the global minimum of the cost function, a genetic algorithm in combination with a Gauss–Newton optimization can be used [21]. Although this improves the performance of the MLE method, finding the global minimum of the cost function is still not guaranteed. Finding a local minimum results in a parameter set that does not represent the actual multimodal pilot control behavior.

This paper introduces a parameter estimation technique based on interval analysis that guarantees to find the global minimum of a cost function. The traditional two-step method, that is, a fit of the pilot model in the frequency domain, is used as proof of concept for this new parameter estimation method, as it requires less computer power. Expanding the method to a cost function in the time domain is relatively straightforward.

III. Interval Analysis

A. Interval Arithmetic

Interval analysis is the theory dealing with interval numbers and the arithmetic operations on them [9,25]. An interval number $[x]$ is defined by an ordered pair of real numbers $[x] = [a, b]$ with $a \leq b$. Operations applied to the ordinary number system can be extended to cover interval numbers, for example, the basic computational operations of addition, subtraction, multiplication, and division:

$$[a, b] + [c, d] = [a + c, b + d] \tag{7}$$

$$[a, b] - [c, d] = [a - d, b - c] \tag{8}$$

$$[a, b] \cdot [c, d] = [\min(ac, ad, bc, bd), \max(ac, ad, bc, bd)] \tag{9}$$

$$\frac{[a, b]}{[c, d]} = [a, b] \cdot [1/d, 1/c] \quad \text{if } 0 \notin [c, d] \tag{10}$$

The infimum and supremum of an interval are defined as

$$\inf([x]) = \min_{x\in[x]} x; \qquad \sup([x]) = \max_{x\in[x]} x \tag{11}$$

Although some people had been working on bounding rounding errors in the 1950s, the breakthrough for interval analysis came in the 1960s with Moore's book *Interval Analysis* [9].

Just as ordinary numbers can be grouped in vectors, interval numbers can be stacked in an interval vector, which is often called a box. An interval can be split into two or more subintervals. For interval vectors, these subdivisions are called subboxes. In the remainder of the text, interval variables and numbers will be denoted with square brackets.

The core of interval analysis is to use interval arithmetic to form an inclusion function $[f]([\mathbf{x}])$ of any function $f(\mathbf{x})$. This property of interval arithmetic follows from the inclusion function theorem given by Moore [9,26]:

Theorem III.1: If $[f]([x_1], [x_2], \ldots, [x_n])$ is a rational expression in the interval variables $[x_1], [x_2], \ldots, [x_n]$, that is, a finite combination of $[x_1], [x_2], \ldots, [x_n]$ and a finite set of constant intervals with interval arithmetic operations, then

$$[x_1]' \subset [x_1], [x_2]' \subset [x_2], \ldots, [x_n]' \subset [x_n] \tag{12}$$

implies

$$[f]([x]'_1, [x]'_2, \ldots, [x]'_n) \subset [f]([x_1], [x_2], \ldots, [x_n]) \tag{13}$$

for every set of interval numbers $[x_1], [x_2], \ldots, [x_n]$ for which the interval arithmetic operations in $[f]$ are defined.

If we take $[x]'_1, [x]'_2, \ldots, [x]'_n$ to be the crisp numbers $x_1, x_2, \ldots, x_n$ and apply the theorem, then we obtain

$$f(x_1, x_2, \ldots, x_n) \subset [f]([x_1], [x_2], \ldots, [x_n]) \tag{14}$$

for $x_1 \subset [x_1], x_2 \subset [x_2], \ldots, x_n \subset [x_n]$. It states that, if the function arguments lie within the corresponding intervals, we can use interval arithmetic to produce an interval for the output of the function which is guaranteed to contain the crisp function output $f(\mathbf{x})$.

A property of interval arithmetic that can be deduced from Eqs. (7–10) is that the width of the interval can increase under certain computational operations. Some examples of interval arithmetic are

$$[0, 1] - [0, 1] = [-1, 1] \tag{15}$$

$$[-1, 2] \cdot [-3, 4] = [-6, 8] \tag{16}$$

$$\sin[\pi, 2\pi] = [-1, 0] \tag{17}$$

The first equation shows that in interval arithmetic $[x] - [x] \neq [0, 0]$, because the two terms are considered as different interval variables that follow the subtraction rule in Eq. (8). This property of interval arithmetic, known as the dependency problem, makes the result of a computation dependent on its formulation. This is shown in the following example:

$$[x] = [-1, 3] \tag{18}$$

$$[x]^2 - [x] = [-3, 10] \tag{19}$$

$$[x]([x] - 1) = [-6, 6] \tag{20}$$

$$\left([x] - \frac{1}{2}\right)^2 - \frac{1}{4} = [0, 6] \tag{21}$$

In the last formulation, the interval variable $[x]$ occurs only once, which results in the tightest bounds on the result of all the formulations. In fact, when the variable occurs only once, the interval computation will produce the exact range of values [9]. Note that the interval square operator here is not evaluated as an interval product, which would mean there are two instances of $[x]$, but as a special function that does not return negative lower or upper bounds.

B. Global Nonlinear Optimization Using Interval Analysis

Using the properties of intervals as described in the previous section, this section will show how intervals can be used to solve nonlinear optimization problems. Consider a nonlinear cost function J as a function of variables $x_1, \ldots, x_n$ and optimization parameters $p_1, \ldots, p_m$:

$$J = J(x_1, \ldots, x_n; p_1, \ldots, p_m) \tag{22}$$

The parameters and the resulting cost function can be replaced with intervals:

$$[J] = [J](x_1, \ldots, x_n; [p_1], \ldots, [p_m]) \tag{23}$$

where, for each parameter, the interval is set to cover the search space in which it needs to be optimized, so if, for example, p_1 represents a delay which is known to be in the range of 0.1–0.8 s, the interval for p_1 is set to

$$[p_1] = [0.1, 0.8] \tag{24}$$

The complete search space is defined by the set of all interval parameters, which can be combined into a single interval vector or box:

$$[P] = \begin{pmatrix} [p_1] \\ \cdots \\ [p_m] \end{pmatrix} \tag{25}$$

A global minimum can only be found if it resides within the chosen search space, defined by the interval boundaries. When the true global minimum is not inside the search space, a suboptimal local minimum will be found.

The optimization problem can now be written as

$$\min[J](x_1, \ldots, x_n; [P]) \quad \forall\ p_i \in [p_i] \qquad i = 1 \ldots m \tag{26}$$

The goal of the branch and bound algorithm is to efficiently remove subboxes from the initial parameter space, for which it is guaranteed that they do not contain the parameter combination for which the cost function has a global minimum. When the cost function is evaluated for a particular parameter box, the resulting interval can be wider than the range of values obtained when computing the cost function for each combination of parameters in the parameter box. This is caused by dependency (see Sec. III.A).

Let $[J_1]$ and $[J_2]$ be the cost function evaluation for parameter subboxes $[P_1]$ and $[P_2]$, respectively. Assume that $[P_1]$ and $[P_2]$ are formed by a bisection of the complete search space $[P]$. Because of dependency it is not possible to say that the true minimum value of J is in the interval with the lowest lower bound. So if $\inf([J_1]) < \inf([J_2])$, the true minimum value of J can still be in $[J_2]$.

Only if $\sup([J_1]) < \inf([J_2])$ it is guaranteed that the global minimum of the cost function is in the interval $[J_1]$ and thus formed by a combination of parameters inside $[P_1]$. This method of eliminating subboxes that cannot contain the global minimum is a fundamental element of the interval branch and bound algorithm.

The first step of the interval branch and bound algorithm is to define the parameter search space as a multidimensional interval box $[P]$. Next, the cost function is evaluated for this complete search space, resulting in $[J_P]$. This interval is likely to be very wide, but it is guaranteed that the global minimum is contained in it. As was explained earlier, it is not possible to say that the global minimum is equal to $\inf([J_P])$. The best estimate $J^*_{\min}$ for the global minimum of the cost function at this point in the algorithm is $J^*_{\min} = \sup([J_P])$.

The next step is to split the initial search space into subboxes. There are many possible ways to split an interval box and the theory does not state which method is best. Although the type of box-splitting influences the efficiency of the algorithm, the same global minimum will be found each time. Some possibilities are as follows:

1) Bisect in all dimensions, resulting in 2^m new subboxes, where m is the number of dimensions in the original interval box.

2) Bisect in the dimension with the widest interval, while not bisecting in the remaining dimensions, resulting in two new subboxes.

3) Split a random dimension into a random number of subintervals.

The new boxes obtained by splitting are put into a list ordered by the lower bound on the cost function evaluation of these boxes (see Fig. 2). Every time a new box is added to the list, the current best estimate of the cost function $J^*_{\min}$ can be lowered if the upper bound of the cost function corresponding to the added subbox is lower than the current best estimate. When $J^*_{\min}$ is lowered, a check is done to see if any of the boxes in the list have a lower bound on the cost function that is higher than the current best estimate. If so, then this box *and all following boxes* can be removed from the ordered list, thereby reducing the search space. In the example in Fig. 2, the subboxes $[P_6]$ and $[P_7]$ can be removed from the list.

Splitting, evaluating, and discarding of subboxes continues until the list of subboxes is empty. This requires some criteria to be set on the minimal width of the parameter subboxes:

$$\max(w([P_i])) < \varepsilon_P, \qquad w([x]) = \sup([x]) - \inf([x]) \tag{27}$$

When subboxes reach this criteria, they will not be split again and they will be put in a list of solutions. A different type of stopping criteria is the width of the evaluated cost function for a subbox:

$$w([J_i]) < \varepsilon_J \tag{28}$$

Usually, a combination of both stopping criteria is used, depending on the shape of the cost function. A flowchart of the basic interval branch and bound algorithm is given in Fig. 3.

C. Gradient Information in Branch and Bound

The branch and bound algorithm described previously can be extended to incorporate information about the gradient of the cost function with respect to the parameters if this information is available. Imagine an interval cost function $[J]$ as a function of a single interval parameter $[p]$, then the derivative $\mathrm{d}Jp([p_i])$ evaluated for a subbox $[p_i]$ of $[p]$ becomes

$$[\mathrm{d}Jp]([p_i]) = \frac{\mathrm{d}[J([p])]}{\mathrm{d}[p]}([p_i]) \tag{29}$$

which can fall in only one of the following categories:

1) $\inf([\mathrm{d}Jp]([p_i])) > 0$,

2) $\sup([\mathrm{d}Jp]([p_i])) < 0$,

3) $\inf([\mathrm{d}Jp]([p_i])) < 0$, $\sup([\mathrm{d}Jp]([p_i])) > 0$.

For case 1, the derivative is positive over the entire subbox, whereas for case 2 the derivative is negative over the entire subbox. For these cases, there cannot be a global minimum in the subbox, so it can be discarded. An exception occurs for case 1 when the subbox contains an edge of the initial search space, which makes it possible for the global minimum to lie on this edge. This exception is taken into account and only subboxes satisfying the first case while not containing an edge of the initial search space are discarded.

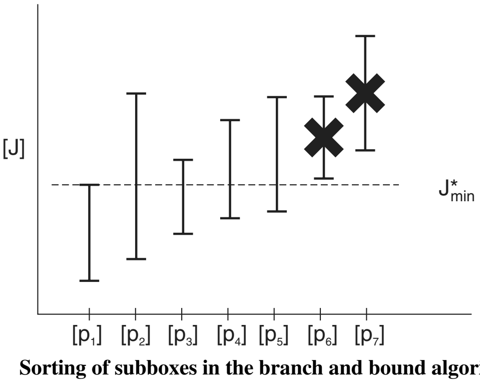


Fig. 2 Sorting of subboxes in the branch and bound algorithm.

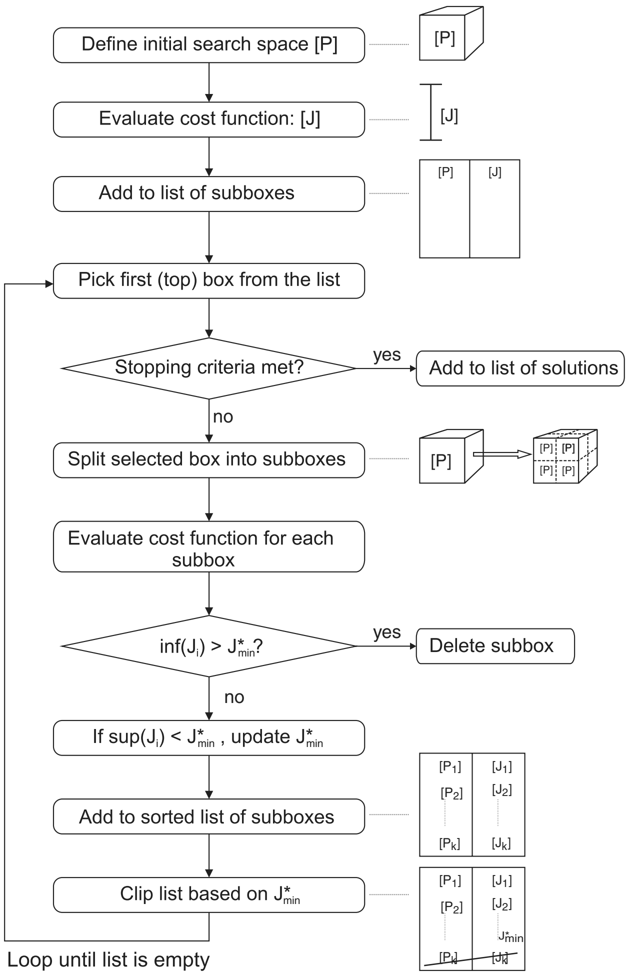

Define initial search space [P]
[P]
Evaluate cost function: [J]
[J]
Add to list of subboxes
[P]
[J]
Pick first (top) box from the list
Stopping criteria met?
yes
Add to list of solutions
no
Split selected box into subboxes
[P]
[P]
[P]
[P]
[P]
Evaluate cost function for each subbox
inf(J$_i$) > J$^*_{min}$?
yes
Delete subbox
no
If sup(J$_i$) < J$^*_{min}$, update J$^*_{min}$
Add to sorted list of subboxes
[P$_1$]
[J$_1$]
[P$_2$]
[J$_2$]
[P$_k$]
[J$_k$]
Clip list based on J$^*_{min}$
[P$_1$]
[J$_1$]
[P$_2$]
[J$_2$]
J$^*_{min}$
[P$_k$]
[J$_k$]
Loop until list is empty

Fig. 3 Schematic flowchart of the interval branch and bound algorithm.

When the interval derivative contains both negative and positive parts, the subbox can (but does not have to) contain either a local minimum or a local maximum, so the subbox cannot be deleted, but it stays in the branch and bound loop, where it will be split into smaller boxes that perhaps can be deleted based on their derivatives.

D. Box Contraction

A further extension of the branch and bound algorithm, also based on the gradient information, is to contract parts of a subbox that cannot contain the solution (Fig. 4). The procedure is as follows:

1) Compute the derivative of the cost function with respect to one of the parameters evaluated at the lower bound of the parameter interval.

2) Compute the second derivative of the cost function with respect to one parameter evaluated over the complete parameter interval.

3) Use these derivatives to determine the subinterval where a zero of the first derivative can occur, Eqs. (30–32). Note that Eqs. (30) and (31) are crisp equations.

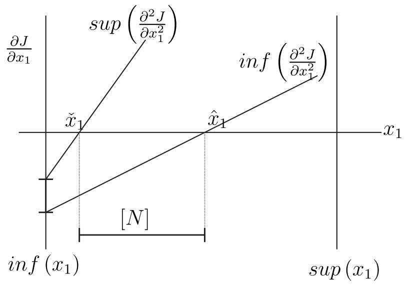

$\frac{\partial J}{\partial x_1}$
$sup\left(\frac{\partial^2 J}{\partial x_1^2}\right)$
$inf\left(\frac{\partial^2 J}{\partial x_1^2}\right)$
$\check{x}_1$
$\hat{x}_1$
x_1
$[N]$
$inf\,(x_1)$
$sup\,(x_1)$

Fig. 4 Box contraction based on gradient information.

4) Intersect this subinterval with the original interval to obtain a reduced or contracted interval, Eq. (33).

5) When the amount of contraction is above a user-defined percentage, the process can be repeated.

$$\sup\left(\left.\frac{\partial J}{\partial x_1}\right|_{\inf(x_1)}\right) + \sup\left(\left.\frac{\partial^2 J}{\partial x_1^2}\right|_{[x_1]}\right)(\check{x}_1 - \inf(x_1)) = 0 \tag{30}$$

$$\inf\left(\left.\frac{\partial J}{\partial x_1}\right|_{\inf(x_1)}\right) + \inf\left(\left.\frac{\partial^2 J}{\partial x_1^2}\right|_{[x_1]}\right)(\hat{x}_1 - \inf(x_1)) = 0 \tag{31}$$

$$[N] = [\check{x}_1, \hat{x}_1] \tag{32}$$

$$[x_1] = [x_1] \cap [N] \tag{33}$$

IV. Examples of Parameter Optimization Using Interval Analysis

Before applying the interval optimization method to the field of human perception modeling, this section gives two simple generic parameter optimization examples, showing the differences between gradient-based optimization methods and the interval optimization method. The motivation for showing these simplified examples is that it is easy to show, in two dimensions, the effects of different initial conditions for the gradient-based optimization. For the more realistic and higher dimensional problem in Sec. II, it is not possible to show these effects with the same clarity.

A. Example of an Optimization with a Complex Cost Function

This example optimization problem has two parameters, p_1 and p_2. Reference output data d are generated by computing the system output for some chosen values p_1^* and p_2^*. For this example, we set $p_1^* = 2.2$ and $p_2^* = 3.6$:

$$d = p_1^* e^{-j\omega p_2^*} \tag{34}$$

The goal is to find p_1^* and p_2^*, using only the value of d and description of the system equation [Eq. (34)]. The system output for any combination of crisp p_1 and p_2 is written as

$$y(p_1, p_2) = p_1 e^{-j\omega p_2} \tag{35}$$

which can be written in interval notation as

$$[y]([p_1], [p_2]) = [p_1] e^{-j\omega[p_2]} \tag{36}$$

Next, a search space is defined, which is the starting point for the branch and bound algorithm described in Sec. III.B and also forms the boundary for the constraint gradient-based optimization method:

$$[P_0] = \begin{pmatrix} [p_1] \\ [p_2] \end{pmatrix} = \begin{pmatrix} [0, 7] \\ [0.5, 6.7] \end{pmatrix} \tag{37}$$

The cost function that is to be minimized is given in Eq. (38), together with its interval notation, and is visualized in Fig. 5:

$$\begin{aligned} J(p_1, p_2) &= \text{abs}(d - y(p_1, p_2)); \\ [J]([p_1], [p_2]) &= \text{abs}(d - [y]([p_1], [p_2])) \end{aligned} \tag{38}$$

Clearly, using gradient-based optimization algorithms such as MATLAB's `fmincon` and `fminsearch` will have problems finding the global minimum of the cost function when the initial point is outside of the convex area that holds the global minimum. Choosing the correct initial starting point for the optimization can be difficult and usually the optimization is repeated for different initial conditions until the *expected* global minimum has been found. For

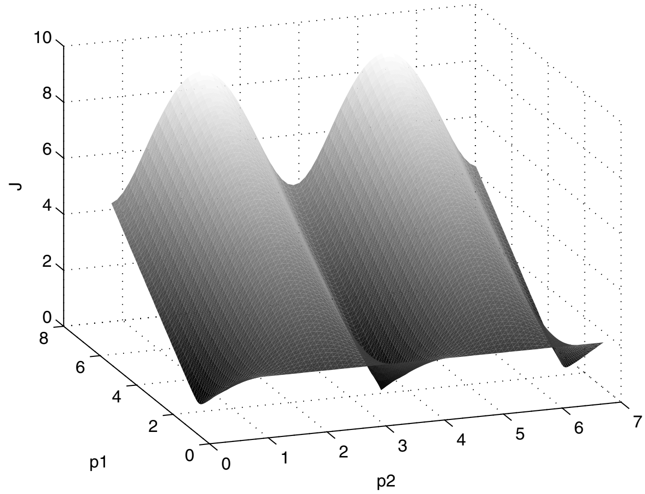


Fig. 5 Cost function with multiple local minimums.

this example, both the constrained and unconstrained algorithms give five solutions for the minimum of the cost function. Note that it is not possible to guarantee that no lower minimum is found when starting at another initial point. This property of gradient-based optimization methods is illustrated by Fig. 6, which shows the minimums that are found by starting from many different initial points. When the interval branch and bound optimization method from Fig. 3 is used, starting with the initial parameter space defined by Eq. (37), *only one minimum* is found, which is guaranteed to be the global minimum for the interval ranges selected:

$$[P_{\text{final}}] = \begin{pmatrix} [p_1]_{\text{final}} \\ [p_2]_{\text{final}} \end{pmatrix} = \begin{pmatrix} [2.1999, 2.2001] \\ [3.5999, 3.6001] \end{pmatrix} \tag{39}$$

The cost function value corresponding to this solution is

$$[J_{\text{final}}] = [0, 5.29\text{e} - 007] \tag{40}$$

B. Example of an Optimization Including a Threshold

There is no limitation on the model structure when using the interval method; the cost function is even allowed to be discontinuous to any order, which is another advantage compared to gradient-based methods. Also, there does not need to be an analytical expression for the cost function, for example, in the case where the cost function is a function of measurement data stored in lookup tables [13]. In the next example, a threshold is put into the model structure, which is defined in the time domain and has two parameters, p_1 and p_2. Reference output data d are generated by computing the system output for some chosen values p_1^* and p_2^*. For this example, we set $p_1^* = 2.2$ and $p_2^* = 0.4$:

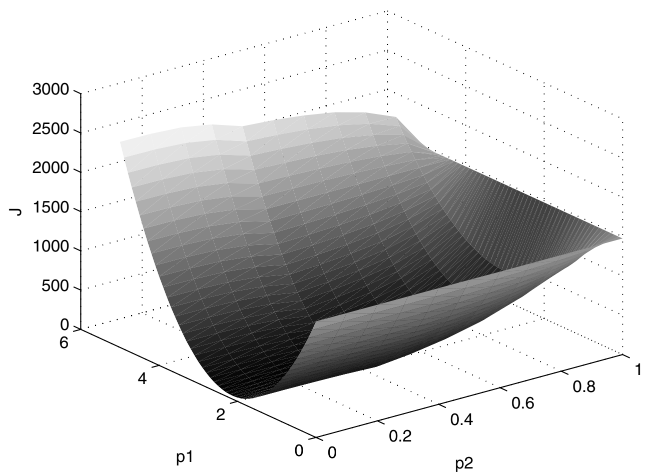


Fig. 7 Cost function with multiple local minimums.

$$d = \begin{cases} p_1^* x & x \geq p_2^* \\ 0 & x < p_2^* \end{cases} \tag{41}$$

where x is a vector of input data in the time domain. The goal is to find p_1^* and p_2^*, using only the value of d and description of the system equation [Eq. (41)]. The system output for any combination of crisp p_1 and p_2 is written as

$$y = \begin{cases} p_1 x & x \geq p_2 \\ 0 & x < p_2 \end{cases} \tag{42}$$

which can be written in interval notation as

$$[y] = \begin{cases} [p_1] x & x \geq [p_2] \\ 0 & x < [p_2] \end{cases} \tag{43}$$

Next, a search space is defined, which is the starting point for the branch and bound algorithm described in Sec. III.B and also forms the boundary for the constraint gradient-based optimization method:

$$[P_0] = \begin{pmatrix} [p_1] \\ [p_2] \end{pmatrix} = \begin{pmatrix} [0, 5] \\ [0, 1] \end{pmatrix} \tag{44}$$

The cost function that is to be minimized is given in Eq. (45), together with its interval notation, and is visualized in Fig. 7:

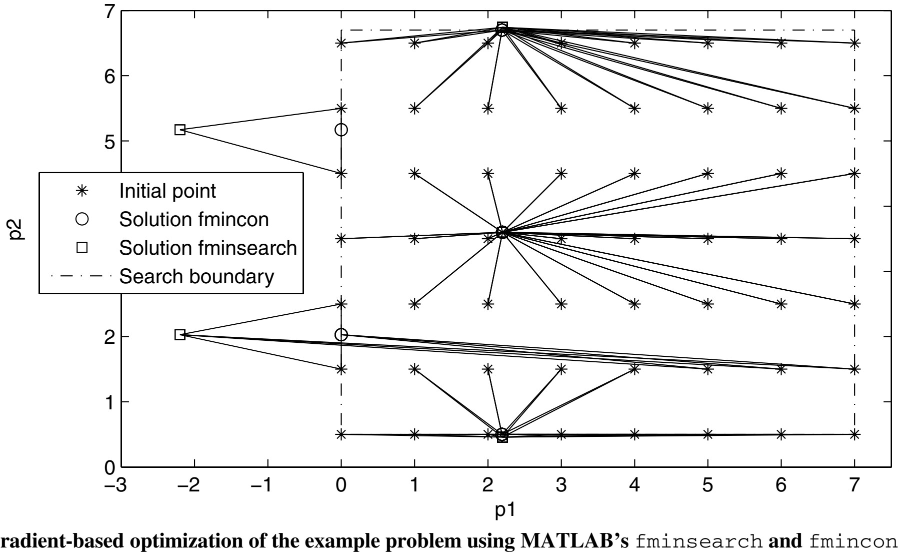


Fig. 6 Gradient-based optimization of the example problem using MATLAB's `fminsearch` and `fmincon` functions.

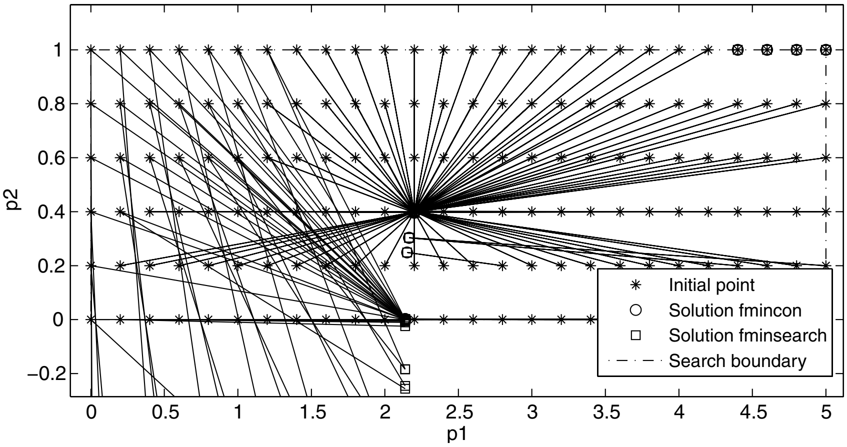


Fig. 8 Gradient-based optimization of the example problem using MATLAB's `fminsearch` and `fmincon` functions.

$$J = \sum (d - y)^2; \qquad [J] = \sum (d - [y])^2 \tag{45}$$

As in the previous example, gradient-based optimization algorithms such as MATLAB's `fmincon` and `fminsearch` have problems finding the global minimum of the cost function, illustrated by Fig. 8, which shows the minimums that are found by starting from many different initial points. When the interval branch and bound optimization method is used, the correct global minimum is found:

$$[P_{\text{final}}] = \begin{pmatrix} [p_1]_{\text{final}} \\ [p_2]_{\text{final}} \end{pmatrix} = \begin{pmatrix} [2.1999, 2.2001] \\ [0.3952, 0.4035] \end{pmatrix} \tag{46}$$

V. Experimental Results

In this section, the optimization results for the human perception modeling problem described in Sec. II are given for data from an experiment. First, the setup of the experiment will be described briefly. Next, the results for the interval optimization, for two experiment conditions, are presented in Sec. V.B. Finally, the solution to the same problem obtained by a gradient-based optimization method is given in Sec. V.C.

A. Experiment Setup

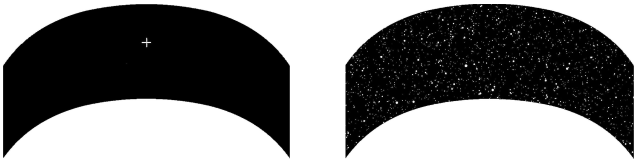
In the past years, experiments were performed at the Delft University of Technology Faculty of Aerospace Engineering to investigate the role of optic flow and the influence of physical motion cues during control of self-motion by modeling human perception and control behavior [16,27]. The data of one of these experiments are used to determine the performance of the interval analysis parameter estimation procedure. The experiment was performed on the SIMONA Research Simulator of the Delft University of Technology (see Fig. 9). This simulator features a 6-degree-of-freedom hydraulic motion system with an actuator stroke of 1.15 m. The visual system consists of three projectors that generate a 180 deg collimated outside visual.

Fig. 9 SIMONA Research Simulator.

The control task and the multimodal pilot model used in this experiment were discussed in Sec. II. In the experiment, two visual configurations were used for the outside visual of the simulator (see Fig. 10). The baseline visual condition explicitly shows the track angle error e, that is, the difference between the target forcing function f_t and the track angle χ by a cross (see Fig. 1). In the second visual condition, the subjects were moving through an optic flowfield consisting of Gaussian blobs, randomly scattered around a three-dimensional grid. When moving through this optic flowfield, the focus of radial outflow provides the subjects with an implicit presentation of the error.

The two visual conditions were combined with three physical motion conditions: no motion, reduced motion, and full motion. In the full motion condition, the accelerations of the model were translated one-to-one into accelerations of the simulator. For the reduced motion condition, the accelerations were scaled by a factor of 0.5. A classical washout filter was used to prevent the simulator from drifting to the limits of the motion system, however, this had a negligible effect on the accelerations in the frequency range of interest. The experiment had a full factorial design, resulting in six conditions in total. In this paper, the interval analysis parameter estimation technique is used on the data of two conditions from one subject, that is, baseline with full motion and optic flow with full motion.

The subjects were controlling the track angle of a cart. The physical motion in the experiment was in correspondence with the lateral movement of the cart. This means that only the lateral degree of freedom of the motion system, sway, was used during the experiment. The velocity V of the cart was 3 m/s. In combination with a motion filter, this ensured that the simulator did not reach any motion bounds during the experimental runs. With full motion, the subjects also perceive information on the current track angle of the system dynamics by means of the lateral motion cues sensed by the subjects' otoliths. This results in the two-channel pilot model given in Fig. 1.



a) Baseline b) Optic flow

Fig. 10 Visual conditions used in the experiment.

Table 1 Search space

K_v	[0.0001, 2.0000]
T_v	[0.0001, 6.0000]
τ_v	[0.0001, 1.0000]
K_m	[0.0001, 2.0000]
τ_m	[0.0001, 1.0000]
T_{oto1}	[1.0000, 1.0000] [a]
T_{oto2}	[0.5000, 0.5000] [a]
ζ_{nm}	[0.1800, 0.1800] [a]
ω_{nm}	[11.0000, 11.0000] [a]

[a] Fixed parameter.

Table 2 Solution interval optimization for the baseline condition

K_v	[0.6057, 0.6059]
T_v	[0.1518, 0.1522]
τ_v	[0.2620, 0.2622]
K_m	[0.0011, 0.0012]
τ_m	[0.3047, 0.3048]
T_{oto1}	[1.0000, 1.0000]
T_{oto2}	[0.5000, 0.5000]
ζ_{nm}	[0.1800, 0.1800]
ω_{nm}	[11.0000, 11.0000]
J	[18.3237, 18.3731]

Table 3 Solution interval optimization for the flow condition

K_v	[0.6697, 0.6698]
T_v	[0.0001, 0.0002]
τ_v	[0.3681, 0.3683]
K_m	[0.0028, 0.0029]
τ_m	[0.3899, 0.3901]
T_{oto1}	[1.0000, 1.0000]
T_{oto2}	[0.5000, 0.5000]
ζ_{nm}	[0.1800, 0.1800]
ω_{nm}	[11.0000, 11.0000]
J	[70.0542, 70.1089]

A two-step method is used to estimate the parameters of the pilot model. From the time-domain data, the frequency response functions of the pilot are estimated using a multi-input single-output ARX model [20]. These frequency response functions are then used in the second step to estimate the pilot model parameters using the interval analysis parameter estimation procedure. Additionally, a gradient-based optimization is used to serve as a baseline for the interval analysis method. The cost function used in the optimization methods is given by

$$J = \frac{|\hat{H}_{\text{pe}}(j\omega) - \tilde{H}_{\text{pe}}(j\omega, P)|^2}{\hat{\sigma}^2_{|\hat{H}_{\text{pe}}|}} + \frac{|\hat{H}_{p\chi}(j\omega) - \tilde{H}_{p\chi}(j\omega, P)|^2}{\hat{\sigma}^2_{|\hat{H}_{p\chi}|}} \tag{47}$$

where $\hat{}$ indicates the frequency response of the ARX estimate and $\tilde{}$ indicates the frequency response of the parameter model.

Five parameters of the pilot model will be estimated [see Eq. (6)]. Theoretically. it is possible to optimize for more than these five parameters, for example, by including the neuromuscular dynamics. The interval algorithm will still find the global minimum of the cost function, but the computation time will increase exponentially when extra parameters are introduced. In this paper, the problem is limited to five parameters, as the advantages of the interval method can already be seen for this somewhat easier optimization problem. It should be noted that the nature of interval optimization, that is, using the branch and bound algorithm, makes it very easy to distribute the computational load, for example, using a parallel computing cluster.

A search area needs to be defined that forms the initial interval box and sets up constraints for the gradient-based optimization. The search space is chosen as in Table 1.

B. Interval-Based Optimization

The results for the interval optimization shown in this section are obtained by applying the standard branch and bound algorithm from Fig. 3, supplemented by gradient-based box deletion (Sec. III.C), and box contractions (Sec. III.D). During the branch and bound algorithm, only subboxes that cannot contain the global minimum are deleted, so the parameter box that is left at the end is guaranteed to contain the global minimum of the cost function. The stopping criteria is set as

Table 4 Solution of the gradient-based optimization method for the baseline condition and several initial starting points

	$x_0(0)$	$x_0(1/4)$	$x_0(1/2)$	$x_0(3/4)$	$x_0(1)$	Interval
K_v	0.6058	0.6058	0.6058	0.6058	0.6058	[0.6057, 0.6059]
T_v	0.0921	0.0921	0.0921	0.0921	0.0921	[0.1518, 0.1522]
τ_v	0.2621	0.2621	0.2621	0.2621	0.2621	[0.2620, 0.2622]
K_m	0.0001	0.0005	0.0001	0.0004	0.0004	[0.0011, 0.0012]
τ_m	0.0001	0.6706	0.0001	1.0000	1.0000	[0.3047, 0.3048]
T_{oto1}	1.0000	1.0000	1.0000	1.0000	1.0000	[1.0000, 1.0000]
T_{oto2}	0.5000	0.5000	0.5000	0.5000	0.5000	[0.5000, 0.5000]
ζ_{nm}	0.1800	0.1800	0.1800	0.1800	0.1800	[0.1800, 0.1800]
ω_{nm}	11.0000	11.0000	11.0000	11.0000	11.0000	[11.0000, 11.0000]
J	24.2229	23.0905	24.2229	23.4161	23.4161	[18.3237, 18.3731]

Table 5 Solution of the gradient-based optimization method for the flow condition and several initial starting points

	$x_0(0)$	$x_0(1/4)$	$x_0(1/2)$	$x_0(3/4)$	$x_0(1)$	Interval
K_v	0.6697	0.6697	0.6697	0.6697	0.6697	[0.6697, 0.6698]
T_v	0.0001	0.0001	0.0001	0.0001	0.0001	[0.0001, 0.0002]
τ_v	0.3682	0.3682	0.3682	0.3682	0.3682	[0.3681, 0.3683]
K_m	0.0011	0.0011	0.0011	0.0001	0.0001	[0.0028, 0.0029]
τ_m	0.0001	0.0001	0.0001	0.9922	0.9922	[0.3899, 0.3901]
T_{oto1}	1.0000	1.0000	1.0000	1.0000	1.0000	[1.0000, 1.0000]
T_{oto2}	0.5000	0.5000	0.5000	0.5000	0.5000	[0.5000, 0.5000]
ζ_{nm}	0.1800	0.1800	0.1800	0.1800	0.1800	[0.1800, 0.1800]
ω_{nm}	11.0000	11.0000	11.0000	11.0000	11.0000	[11.0000, 11.0000]
J	93.9653	93.9653	93.9653	97.9477	97.9477	[70.0542, 70.1089]

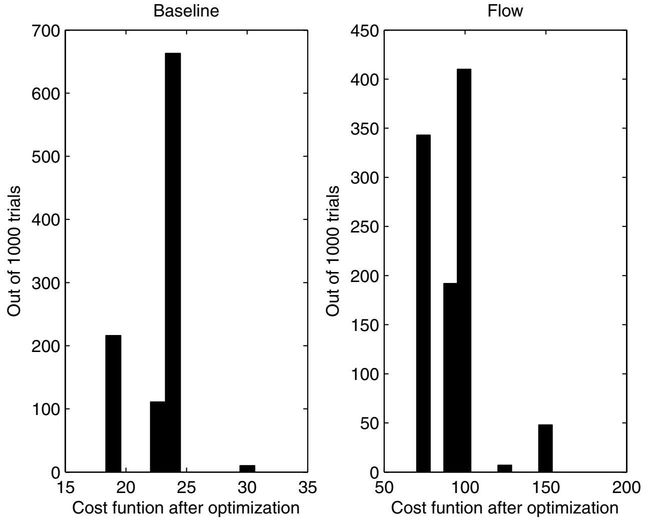


Fig. 11 Minimum cost function after gradient-based optimization for 1000 random initial points, for both the baseline and the flow condition.

$$\sup([p_i]) - \inf([p_i]) < 0.00001 \qquad i = 1, \ldots, 5 \qquad (48)$$

where $[p_i]$ are the parameters from Eq. (6). Many boxes can end up satisfying this stopping criteria, depending on the steepness of the cost function near the global minimum and the type of box splitting that is used. To obtain the location of the global minimum, the hull of all these boxes must be taken, where the hull of two intervals $[x]$ and $[y]$ is defined as

$$\text{hull}\,([x],[y]) = [\min(\inf([x]), \inf([y])), \max(\sup([x]), \sup([y]))] \qquad (49)$$

The hulls of the remaining boxes for the baseline condition and the flow condition are shown in Tables 2 and 3, respectively. It can be seen that the diameter or width of the individual parameters is on the order of 0.0002, while the width of the cost function is around 0.05. This difference in widths indicates the presence of the interval dependency problem, which was explained in Sec. III. A cost function evaluation for crisp (i.e., width is zero) parameters leads of course to a crisp cost function, but when interval parameters are used, dependency can cause a small interval around one of the parameters to result in a relatively wide interval (in this case, 250 times as wide) around the cost function.

C. Gradient-Based Optimization

The gradient-based optimization requires an initial point in the parameter space from which the optimization is started. This initial guess requires knowledge about the optimization problem, but even for someone experienced in the field of human perception modeling it can be hard to find a good initial guess and it usually takes several tries until a point is found that is believed to lead to the global minimum. Depending on the type of problem, points in the parameter space that are close to each other can lead to very different solutions, making the selection of the initial guess even harder.

In Tables 4 and 5, the results for the gradient-based optimization are given for five sets of initial conditions, where, for example, $x_0(1/4)$ indicates that each parameter is set to a quarter of the total search space as defined by Table 1, so $x_0(0)$ is the lower bound and $x_0(1)$ the upper bound on the search space. For comparison, the results from the interval optimization in Tables 2 and 3 are given in the last column.

From these tables, one can see that the gradient-based methods have not found the global minimum of the cost function, because the minimum cost function found by the interval method is about 20% lower. Some of the parameters, for example, K_v and τ_v, are converged to their correct values, but others are stuck in a local minimum. If the solution from the interval method is used as the initial guess for the gradient optimization, then the gradient method cannot find a better combination of parameters than the initial guess, proving that the solution from the interval method is at least a local minimum, but it is of course guaranteed that it is also the global minimum.

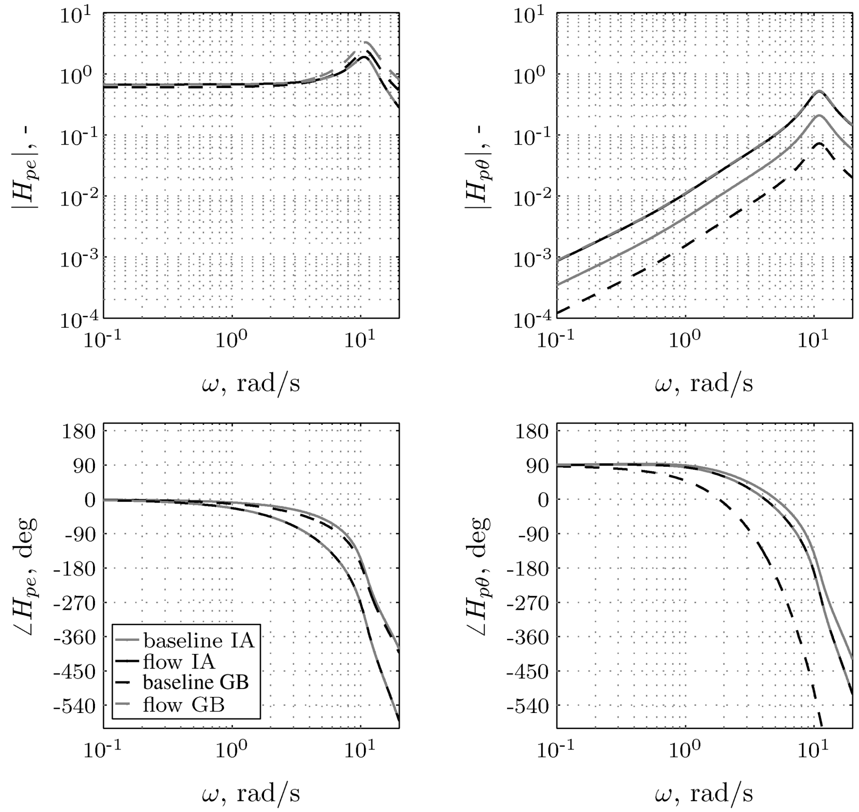


Fig. 12 Pilot frequency response functions for the baseline and flow conditions [interval analysis (IA); gradient-based (GB)].

To show more clearly the importance of the initial guess for the gradient method, Fig. 11 shows histograms for the minimum cost function value found for 1000 different initial guesses for both the baseline and flow conditions. For the baseline condition, 194 random initial conditions lead to the same minimum that is found by the interval method; for the flow condition, this is 250. In total, six different local minimums are found for the baseline condition and eight for the flow condition. Although it seems plausible for these types of problems that the global minimum can be found by performing 1000 randomly initialized gradient optimizations and then select the lowest cost function, there is no guarantee that this is always possible.

D. Frequency Response Functions

Figure 12 gives the pilot frequency response functions H_{pe} and $H_{p\chi}$ (see Fig. 1) for the baseline and flow conditions. For the frequency response function resulting from the interval analysis optimization, the parameters are chosen as the midpoint of the intervals (see Tables 2 and 3). For the frequency response functions resulting from the gradient-based optimization, the solution is chosen that was found the most frequently without going to a parameter bound (see Fig. 11).

It can be seen that, for the baseline condition, the two parameter sets from the two different optimization methods result in two completely different visual and vestibular response functions. Comparing the interval analysis and the gradient-based solution, it can be observed that the gradient-based solution has a lead constant that is too low, a vestibular gain that is too low, and a vestibular time delay that is too high. For the flow condition, the two optimization methods provide the same solution.

VI. Conclusions

This paper introduces a global nonlinear optimization method based on the theory of interval analysis to solve the nonlinear optimization problems occurring in human perception modeling. This method is applied to an experiment, performed on the SIMONA Research Simulator, investigating the role of optic flow and the influence of physical motion cues during control of self-motion by modeling human perception and control behavior.

Results from the interval optimization method are compared to a gradient-based optimization algorithm, which is currently one of the most frequently used methods in human perception modeling. With the interval inclusion theorem and the branch and bound algorithm that only removes boxes from the search space that cannot contain a minimum of the cost function, it is theoretically proved that the global minimum must be in the subboxes returned by the branch and bound algorithm. The results of the simulator experiment investigated here corroborate this conclusion by showing the lowest cost function minimum when compared to gradient-based methods. In the experiment, five parameters are optimized and the global minimum is searched for in a predefined search area of the five-dimensional parameter space. For 1000 gradient-based optimizations, each starting with a random initial starting point, no solutions are found that are better than the solution found with the interval method. A simple two-parameter optimization problem with a complex cost function is used to visualize the disadvantage of the gradient-based methods of getting stuck in local cost function minimums. A second two-parameter example shows how thresholds can be included in the model structure.

The interval optimization method poses no limitations on the structure of the human perception model, providing opportunities to add more nonlinearities to the model such as thresholds. With ongoing research into reducing the computational load of the interval optimization method, it promises to be a useful tool in human perception modeling.

References

[1] McRuer, D. T., and Jex, H. R., "A Review of Quasi-Linear Pilot Models," *IEEE Transactions on Human Factors in Electronics*, Vol. HFE-8, No. 3, 1967, pp. 231–249.
doi:10.1109/THFE.1967.234304

[2] McRuer, D. T., and Klein, R. H., "Comparison of Human Driver Dynamics in Simulators with Complex and Simple Visual Displays and in an Automobile on the Road," *Proceedings of the Eleventh Annual Conference on Manual Control*, NASA Ames Research Center, May 1975, pp. 684–692.

[3] Stapleford, R. L., Peters, R. A., and Alex, F. R., "Experiments and a Model for Pilot Dynamics with Visual and Motion Inputs," NASA Contractor Rept. CR-1325, 1969.

[4] Jex, H. R., Magdaleno, R. E., and Junker, A. M., "Roll Tracking Effects of *G*-Vector Tilt and Various Types of Motion Washout," *Fourteenth Annual Conference on Manual Control*, Univ. of Southern California, Los Angeles, April 1978, pp. 463–502.

[5] Weir, D. H., and McRuer, D. T., "Pilot Dynamics for Instrument Approach Tasks: Full Panel Multiloop and Flight Director Operations," NASA, Tech. Rept. CR-2019, May 1972.

[6] Steurs, M., Mulder, M., and van Paassen, M. M., "A Cybernetic Approach to Assess Flight Simulator Fidelity," AIAA Modeling and Simulation Technologies Conference and Exhibit, AIAA Paper 2004-5442, Aug. 2004.

[7] Mulder, M., "Cybernetics of Tunnel-in-the-Sky Displays," Ph.D. Dissertation, Faculty of Aerospace Engineering, Delft Univ. of Technology, Delft, The Netherlands, 1999.

[8] Zaal, P. M. T., Pool, D. M., Chu, Q. P., Van Paassen, M. M., Mulder, M., and Mulder, J. A., "Modeling Human Multimodal Perception and Control Using Genetic Maximum Likelihood Estimation," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 4, July–Aug. 2009, pp. 1089–1099.
doi:10.2514/1.42843

[9] Moore, R. E., *Interval Analysis*, Prentice–Hall, Upper Saddle River, NJ, 1966, Chap. 1, pp. 8–25.

[10] Hansen, E., and Walster, G. W., *Global Optimization Using Interval Analysis*, 2nd ed., Marcel Dekker, New York, 2004, Chap. 2, pp. 35–41.

[11] Jaulin, L., Kieffer, M., Didrit, O., and Walter, E., *Applied Interval Analysis*, Springer–Verlag, Berlin, 2001, Chap. 2, pp. 11–44.

[12] Juliana, S., "Re-Entry Flight Clearance," Ph.D. Thesis, Delft Univ. of Technology, Delft, The Netherlands, Sept. 2006.

[13] van Kampen, E., Chu, Q. P., Mulder, J. A., and van Emden, M. H., "Nonlinear Aircraft Trim Using Interval Analysis," AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA Paper 2007-6766, Aug. 2007.

[14] de Weerdt, E., Chu, Q. P., and Mulder, J. A., "Continuous State-Action Space Advantage-Learning Using Interval Analysis and Neural Networks," AIAA Guidance, Navigation, and Control Conference and Exhibit, AIAA Paper 2007-6522, Aug. 2007.

[15] deWeerdt, E., van Kampen, E., Chu, Q., and Mulder, J., "Integer Ambiguity Resolution Using Interval Analysis," *Navigation: Journal of the Institute of Navigation*, Vol. 55, No. 4, 2008, pp. 293–307.

[16] Zaal, P. M. T., Nieuwenhuizen, F. M., Mulder, M., and van Paassen, M. M., "Perception of Visual and Motion Cues During Control of Self-Motion in Optic Flow Environments," AIAA Modeling and Simulation Technologies Conference and Exhibit, AIAA Paper 2006-6627, Aug. 2006.

[17] van der Vaart, J. C., "Modelling of Perception and Action in Compensatory Manual Control Tasks," Ph.D. Dissertation, Faculty of Aerospace Engineering, Delft Univ. of Technology, Delft, The Netherlands, 1992.

[18] Hosman, R. J. A. W., "Pilot's Perception and Control of Aircraft Motions," Ph.D. Dissertation, Faculty of Aerospace Engineering, Delft Univ. of Technology, Delft, The Netherlands, 1996.

[19] van Paassen, M. M., and Mulder, M., "Identification of Human Operator Control Behaviour in Multiple-Loop Tracking Tasks," *Proceedings of the Seventh IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design and Evaluation of Man-Machine Systems*, Pergamon, Oxford, England, U.K., Sept. 1998, pp. 515–520.

[20] Nieuwenhuizen, F. M., Zaal, P. M. T., Mulder, M., Van Paassen, M. M., and Mulder, J. A., "Modeling Human Multi-Channel Motion Perception and Control Using Linear Time-Invariant Models," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 4, 2008, pp. 999–1013.
doi:10.2514/1.32307

[21] Zaal, P. M. T., Mulder, M., Van Paassen, M. M., and Mulder, J. A., "Maximum Likelihood Estimation of Multi-Modal Pilot Control Behavior in a Target-Following Task," *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, 2008, pp. 1085–1090.
doi:10.1109/ICSMC.2008.4811426

[22] Pool, D. M., Zaal, P. M. T., Mulder, M., Van Paassen, M. M., and Mulder, J. A., "Parameter Estimation of Multimodal Pilot Models for Manual Target-Following Tasks," *Proceedings of the 27th European Annual Conference on Human Decision-Making and Manual Control*, Delft Univ. of Technology, Delft, The Netherlands, 2008.

[23] Berger, T., Zaal, P. M. T., Mulder, M., and Van Paassen, M. M., "Time Domain Pilot Model Identification Using Modified Maximum Likelihood Estimation," AIAA Modeling and Simulation Technologies Conference and Exhibit, AIAA Paper 2008-7109, Aug. 2008.

[24] Zaal, P. M. T., Pool, D. M., Mulder, M., and Van Paassen, M. M., "New Types of Target Inputs for Multi-Modal Pilot Model Identification," AIAA Modeling and Simulation Technologies Conference and Exhibit, AIAA Paper 2008-7106, Aug. 2008.

[25] Hickey, T., Ju, Q., and van Emden, M., "Interval Arithmetic: From Principles to Implementation," *Journal of the Association for Computing Machinery*, Vol. 48, No. 5, 2001, pp. 1038–1068. doi:10.1145/502102.502106

[26] Moore, R., "Methods and Applications of Interval Analysis," Society for Industrial and Applied Mathematics, Philadelphia, 1979, Chap. 2, pp. 9–18.

[27] Berntsen, M. F. F., Mulder, M., and van Paassen, M. M., "Modelling Human Visual Perception and Control of the Direction of Self-Motion," AIAA Modeling and Simulation Technologies Conference and Exhibit, AIAA Paper 2005-5893, Aug. 2005.