2012

GGOPT: an unconstrained non-linear optimizer

Bassingthwaighte, J.B.

GGOPT: an unconstrained non-linear optimizer; J.B. Bassingthwaighte, I.S. Chan, A.A. Goldstein, I.B. Russak; Comput Methods Programs Biomed. Author manuscript; available in PMC 2012 Jun 7

http://hdl.handle.net/10945/49684

Published in final edited form as:
Comput Methods Programs Biomed. 1988 ; 26(3): 275–281.

GGOPT: an unconstrained non-linear optimizer

J.B. Bassingthwaighte, **I.S. Chan**, **A.A. Goldstein**, and **I.B. Russak**
Center for Bioengineering, University of Washington, Seattle, WA and Department of Mathematics, Naval Postgraduate School, Monterey, CA, U.S.A

Abstract

GGOPT is a derivative-free non-linear optimizer for smooth functions with added noise. If the function values arise from observations or from extensive computations, these errors can be considerable. GGOPT uses an adjustable mesh together with linear least squares to find smoothed values of the function, gradient and Hessian at the center of the mesh. These values drive a descent method that estimates optimal parameters. The smoothed values usually result in increased accuracy.

Keywords

Optimization; Non-linear; Noisy function

1. Introduction

This modular software routine is designed to generate minimizing sequences for C^3 functions, with noisy values. It uses a variant of the method in [2], and is implemented in ANSII 77 standard FORTRAN.

The object of GGOPT is to increase the accuracy of optimization at the expense of additional computation. The algorithm uses an n-dimensional mesh of $1 + n + n^2$ points to estimate the function, gradient and Hessian at the center of the mesh by least squares. The accuracy of the fit depends on the location of the points of the mesh.

GGOPT consists of two phases. Phase I uses an ad hoc method for mesh spacing. On the test problems, and other problems as well, GGOPT is more efficient than the ZXMIN (an IMSL routine), and OPTIF when $n \leq 4$, slower otherwise. The approach used in OPTIF is described by Dennis and Schnabel [1]. Efficiency here means the number of function evaluations that are required to achieve a given accuracy. The single precision version of ZXMIN can handle only very small errors (approximately 10^{-5} relative error). The single precision version of GGOPT handles larger error. The algorithm of Phase II [3], which is not described here, provides mesh adjustments that result in still greater accuracy.

The noise or error in the data for GGOPT is assumed to be bounded in magnitude with a distribution whose covariance is a multiple of the identity matrix. The vector of parameter values, x, can be estimated within a certain accuracy, depending on the magnitude of the error. The method was developed with the assumption that the model function has bounded third derivatives for all values of f less than the value of f at the initial parameter x. The

Correspondence: James B. Bassingthwaighte, Center for Bioengineering, University of Washington WD-12, Seattle, WA 98195, U.S.A.

routine may be particularly useful for improving upon the minimization attained by another method, while the algorithm for Phase II may be tried to improve the accuracy of GGOPT.

2. Computational method

The algorithm estimates the function value f, the gradient vector **G** and the symmetric Hessian matrix **H** by generating a mesh around the current solution x_0 (a mesh is generated by perturbing the current solution x_0 as described in the algorithm, step 1 below) and using linear least squares. It uses the approximated gradient vector **G** and Hessian matrix **H** to estimate the step increments for the parameters.

The quadratic function $\mathbf{\Psi}(x)=f(x_0)+\mathbf{G}(x_0)^T\cdot(x-x_0)+\frac{1}{2}(x-x_0)^T\cdot\mathbf{H}(x_0)\cdot(x-x_0)$ is used to approximate $f(x)$ in a neighborhood of x_0. Let $y = f(x_1), f(x_2),\ldots, f(x_N))^T$ be the function values obtained at points $x_1, x_2,\ldots, x_N$ (points perturbed around x_0), where $N=1 + n + n^2$. We use second subscripts to denote components of the parameter vectors, i.e. x_{ij} indicates the jth component of the ith parameter vector. Let:

$$\theta=[f, g_1, \ldots, g_n, H_{11}, H_{21}, H_{22}, \ldots, H_{nn}]^T$$

and let matrix **A** of dimension $N\times(1 + n + n(n + 1)/2)$ be as shown below.

$$\begin{bmatrix} 1 & x_{11}-x_{01} & \cdots & x_{1n}-x_{0n} & \frac{1}{2}(x_{11}-x_{01})^2 & (x_{11}-x_{01})(x_{12}-x_{02}) & \cdots & \frac{1}{2}(x_{1n}-x_{0n})^2 \\ 1 & x_{21}-x_{01} & \cdots & x_{2n}-x_{0n} & \frac{1}{2}(x_{21}-x_{01})^2 & (x_{21}-x_{01})(x_{22}-x_{02}) & \cdots & \frac{1}{2}(x_{2n}-x_{02})^2 \\ . & . & \cdots & . & . & . & \cdots & . \\ . & . & \cdots & . & . & . & \cdots & . \\ . & . & \cdots & . & . & . & \cdots & . \\ 1 & x_{n1}-x_{01} & \cdots & x_{nn}-x_{0n} & \frac{1}{2}(x_{n1}-x_{01})^2 & (x_{n1}-x_{01})(x_{n2}-x_{02}) & \cdots & \frac{1}{2}(x_{nn}-x_{0n})^2| \end{bmatrix}$$

The problem of obtaining a least squares fit of $\Psi(x)$ to $f(x_1), f(x_2),\ldots, f(x_N)$ can then be formulated as the minimization of $\|\mathbf{A}\cdot\theta - y\|_2{}^2$; this gives the smoothed function values f, gradient vector **G** and Hessian matrix **H**, i.e. $\theta = (\mathbf{A}^T\cdot\mathbf{A})^{-1}\cdot\mathbf{A}^T\cdot y$, if $\mathbf{A}^T\cdot\mathbf{A}$ is non-singular.

One way of generating a non-singular **A** is to take the following mesh points:

$$\begin{gathered} x_0, \\ x_0+h_j\cdot I_j, x_0-h_j\cdot I_j \quad j=1,\ldots,n, \\ x_0+h_j\cdot I_j-h_k\cdot I_k \quad j=1,\ldots,n; k=1,\ldots,n; \\ j\neq k, \end{gathered}$$

where I_j denotes the jth column of the identity matrix, and h_j and h_k are lengths of mesh perturbed in jth and kth component of parameter vector respectively. Thus it will give a pattern consisting of $1 + n + n^2$ mesh points.

For example, let $n = 2$ and $x_0 = (0, 0)$. There are 7 mesh points, i.e. $x_1 = x_0 = (0, 0)$, $x_2 = (h_1, 0)$, $x_3 = (0, h_2)$, $x_4 = (-h_1, 0)$, $x_5 = (0, -h_2)$, $x_6 = (h_1, -h_2)$, $x_7 = (-h_1, h_2)$. The matrix **A** of dimension 7 × 6 is:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & h_1 & 0 & \frac{1}{2}h_1^2 & 0 & 0 \\ 1 & 0 & h_2 & 0 & 0 & \frac{1}{2}h_2^2 \\ 1 & -h_1 & 0 & \frac{1}{2}h_1^2 & 0 & 0 \\ 1 & 0 & -h_2 & 0 & 0 & \frac{1}{2}h_2^2 \\ 1 & h_1 & -h_2 & \frac{1}{2}h_1^2 & -h_1 \cdot h_2 & \frac{1}{2}h_2^2 \\ 1 & -h_1 & h_2 & \frac{1}{2}h_1^2 & -h_1 \cdot h_2 & \frac{1}{2}h_2^2 \end{bmatrix}$$

The algorithm is defined as follows:

1. Let the iteration index $k = 0$.
2. Find the right mesh size, h, in terms of ε, the relative error in $f(x^k)$: if

$$k=0, h^k=\varepsilon^{0.25} \cdot x, \text{else}$$
$$h^k=\frac{\varepsilon^{0.25} \cdot h^{k-1} \cdot f(x^k)}{f(x^k+h^{k-1})-f(x^k)}.$$

3. Generate a mesh matrix **A**.
4. Call function f at each of the mesh points, and store in a vector y.
5. Solve this overdetermined system of equations $\mathbf{A} \cdot \theta = y$ by least squares, where θ contains the information of smoothed function value, gradient and Hessian.
6. Get the Hessian matrix, **H**, and the gradient vector, **G**, from θ.
7. Do a Cholesky decomposition of **H**; if **H** is positive definite, then take a Newton step, $\Delta x = \mathbf{H}^{-1} \cdot \mathbf{G}$; else use a gradient step, $\Delta x = \mathbf{G}/\|\mathbf{G}\|$.
8. Find a step length, γ_k, by a line search, such that $f(x^k - \gamma_k \cdot \Delta x) < f(x^k)$.
9. Update: $x^{k+1} = x^k - \gamma_k \cdot \Delta x.$
10. Check convergence criteria; exit if any one of the convergence criteria is satisfied.
11. Set $k = k + 1$ and go to (2).

In (2) above, h^k is estimated by a secant update that aims to maintain the differences of $f(x^k)$ with step h^k to be 1/4 of the significance of $f(x^k)$ itself.

3. Program instruction

To use the FORTRAN subroutine GGOPT, the user needs a FORTRAN program which calls GGOPT and a function subprogram. In the FORTRAN program, the user must include the following statement:

call GGOPT(f, n, x, maxit, grdtl, stptl, iout, fmin, epsilon, istop).

3.1. Subroutine arguments

Input

1. f: a function subprogram supplied by the user. It must be declared external in the calling program. f defines the objective function to be minimized and should be of the form: $f(x)$.
2. n: the number of parameters to be minimized.

3. x: vector of length n containing the initial guess of the parameters at entry.
4. *maxit*: maximum number of iterations allowed.
5. *grdtl*: convergence criterion, convergence condition satisfied if the Euclidean norm of the gradient is less than or equal to *grdtl.*
6. *stptl*: convergence criterion, convergence condition satisfied if the Euclidean norm of the relative parameter changes is less than or equal to *stptl.*
7. *iout*: logical unit of print output.
8. *fmin*: convergence criterion, convergence condition satisfied if the objective function to be minimized is less or equal to *fmin*; set to 0.0 if it is not known.
9. epsilon (ε): the relative error in the objective function.

Output

1. x: vector of length n containing the optimized parameters at the exit.
2. *grdtl*: the Euclidean norm of the gradients at exit.
3. *fmin*: the minimized objective function value at exit.
4. *istop*: the reason for termination: *istop* = 0, abnormal termination; *istop* = 1, the Euclidean norm of the gradients is less or equal to *grdtl; istop* = 2, the Euclidean norm of the relative parameter changes is less or equal to *stptl; istop* = 3, the number of iterations exceeds *maxit; istop* = 4, unable to locate a better solution than the current solution (current solution is the best approximation); *istop* =5, the objective function value is less or equal to *fmin.*

3.2. User-defined function

The user must supply a FORTRAN function, $f(x)$, that is to be minimized which may contain errors of round off and truncation. For example, fitting a complicated mathematical model to a set of the biological data, the function $f(x)$ will be the sum of the squares of the fitted residuals. Note that in our formulation, the model function can have significant errors. The format of the function is listed below:

```
function f(x)

dimension x(*)

(⋮)

FORTRAN statements

(⋮)

return

end
```

An important application of GGOPT is its use for optimizing a physical device. Consider a device whose output voltage depends on parameters given by voltages. The problem is to find the parameters which maximize the output voltage, i.e. minimize the negative output voltage. This is an analog computer. The user supplied function will be an interface from the analog computer to the FORTRAN subroutine.

3.3. Typical value for convergence criteria

The typical convergence criteria for the single precision version of GGOPT with error at machine run-off level are *maxit* = 20, *grdtl* = 0.0001, *stptl* = 0.00001, *fmin* = 0.

3.4. Control of output

The standard print output from this routine consists of printing the norm of $\nabla f(x)$, the function value $f(x)$, and the parameter values, *x,* at each iteration; as well as printing the reason for stopping. The logical unit of output can be controlled by input argument *iout.*

4. Sample runs

Three standard test problems with standard starting values [5] and an application of fitting tracer data were used for sample runs. The standard test problems are the Helical Valley function in three dimensions and the Rosenbrock and Jennrich-Sampson functions in two dimensions. Five different levels of relative random noise ($\varepsilon = 10^{-7}$, 10^{-4}, 10^{-3}, 10^{-2} and 5 × 10^{-2}) were added to these three functions for testing GGOPT in the presence of noise. The data were the indicator dilution curve, D-[^{3}H]glucose of an isolated rabbit heart obtained at the right ventricle outflow after an injection into the coronary arteries. A 3-region (capillary, interstitial fluid space and parenchymal cell) 2-barrier blood-tissue exchange model, BTEX30, was used to fit the data.

4.1. Helical Valley function

The Helical Valley function with error added is defined as:

$$f(x)=(1+U[-\varepsilon,\varepsilon])\cdot\left[100\cdot\left((x_3-10\cdot\theta)^2+\left(\sqrt{x_1^2+x_2^2}-1\right)^2\right)+x_3^2\right],$$

where $\theta = 1/(2\cdot\pi)\cdot\tan^{-1}(x_2/x_1)$ if $x_1 > 0$, $\theta = 1/(2\cdot\pi)\cdot\tan^{-1}(x_2/x_1) + 0.5$ if $x_1 < 0$ and U[$-\varepsilon$, ε] is a uniform random number generated between $-\varepsilon$ and ε

The minimal function value = 0.0 at $x = (1, 0, 0)$. Table 1 summarizes sample runs using the five different relative errors (ε), all with a standard starting value (− 1, 0, 0), where *ITN* is the number of iterations.

4.2. Rosenbrock function

The error-added Rosenbrock function is defined as:

$$f(x)=(1+U[-\varepsilon,\varepsilon])\cdot\left(100\cdot\left(x_2-x_1^2\right)^2+(1-x_1)^2\right).$$

The minimal function value = 0.0 at $x = (1, 1)$. Table 2 is the summary of the sample runs with the different relative errors (ε), using a standard starting value (− 1.2, 1).

4.3. Jennrich-Sampson function

The noise-added Jennrich-Sampson function is

$$f(x)=(1+U[-\varepsilon,\varepsilon])\times\left(\sum_{i=1}^{10}(2+2\cdot i-\exp(x_1\cdot i)-\exp(x_2\cdot i))^2\right).$$

The minimal function value = 124.362 at $x = (0.2578, 0.2578)$ when the relative error, ε is below or at the computer roundoff level. When the relative error is above the computer roundoff error, the minimal function value = 124.362 ± 124.362 · ε These function values should not be used to compare the runs with different relative errors, but the solution of x will be good for the comparison of the runs at different error levels. Table 3 is the summary of the test results with the five relative errors using a standard starting value (0.3, 0.4). For this starting value, both ZXMIN failed with relative error at 10^{-4}.

4.4. Fitting tracer data with blood tissue exchange model

Consider the application of GGOPT for fitting blood-tissue exchange model (BTEX30) to the recorded indicator dilution curves. In Fig. 1, the open circles show a deoxyglucose outflow dilution curve (normalized $C_D(t)$ curve) and the plus symbols are the albumin outflow dilution curve, which was used as an intravascular reference for the deoxyglucose curve. These outflow dilution curves were obtained by Kuikka et al. [4] from an isolated rabbit heart by injection of tracer-labeled albumin and deoxyglucose into the aortic root. The model is the solution to the 3-region convection–diffusion partial differential equations, which are required to describe the concentration gradients with position along the length of the capillary. Regions are denoted by subscripts: p for plasma in the capillary, isf for interstitial fluid space and pc for parenchymal cells. The concentration inside the capillary $C_p(x, t)$ is then given by the equation:

$$\frac{\partial C_p}{\partial t}=-\frac{F_p\cdot L}{V'_p}\cdot\frac{\partial C_p}{\partial x}-\frac{PS_g}{V'_p}\cdot(C_p-C_{isf}),$$

where V'_p is the volume of distribution for the solute in the capillary (ml · g^{-1}), $F_p \cdot L/V'_p$ is the velocity (cm ·s^{-1}) in the capillary, and L is an arbitrary capillary length.

The equations for the concentration in the interstitium, C_{isf}, and cell region, C_{pc}, are similar except that there are no flow terms. The equations:

$$\frac{\partial C_{isf}}{\partial t}=\frac{PS_g}{V'_{isf}}\cdot(C_p-C_{isf})-\frac{PS_{pc}}{V'_{isf}}\cdot(C_{isf}-C_{pc}),$$
$$\frac{\partial C_{pc}}{\partial t}=\frac{PS_{pc}}{V'_{pc}}\cdot(C_{isf}-C_{pc})-\frac{G_{pc}}{V'_{pc}}\cdot C_{pc},$$

where G_{pc} is an intracellular clearance (ml · g^{-1} · s^{-1}) and G_{pc}/V'_{pc} is equivalent to K_{seq}, a first order rate constant (s^{-1}) for intracellular sequestration or binding which is irreversible. V'_{isf} and V'_{pc} are interstitial and intracellular volumes of distribution.

The parameters, PS_g, PS_{pc}, G_{pc}, V'_{isf} and V'_{pc} were obtained by fitting the solution $C_p(x = L, t)$ to the dilution curve. The data were obtained at discrete time points, $C_D(t_i)$ for i=1, …, m, and the objective function to be minimized is defined as the coefficient of variation, with m − 5 degrees of freedom (the number of data points minus the number of free parameters):

$$\frac{\sqrt{\frac{1}{m-5}\sum_{j=1}^{i=m}(C_{\mathrm{D}}(t_i)-C_{\mathrm{p}}(x=L,t_i))^2}}{\frac{1}{m}\sum_{i=1}^{i=m}C_{\mathrm{D}}(t_i)}.$$

The testing of various relative errors, ε, added to the model function $C_{\mathrm{p}}(x = L, t_i)$, is of less relevance in this test because of the relatively large noise levels of the data, $C_{\mathrm{D}}(t_i)$. Model solutions will usually have errors that are orders of magnitude less than the residual differences between model function and data, in this case represented by a coefficient of variation of 0.072 in the final fitting.

The objective function, the coefficient of variation, the measurement of the fitting deviation between the model and the data, was reduced from 0.96 to 0.072. We chose a set of (arbitrary but with reasonably physiological sense) starting parameter values, $PS_{\mathrm{g}} = 0.02$, $PS_{\mathrm{pc}} = 0.02$, $G_{\mathrm{pc}} = 0.02 \text{ ml} \cdot \text{g}^{-1} \cdot \text{s}^{-1}$, $V'_{\mathrm{isf}}=0.3$, and $V'_{\mathrm{pc}}=0.5 \text{ ml} \cdot \text{g}^{-1}$. The estimates of these transport parameters were obtained by GGOPT at $PS_{\mathrm{g}} = 0.0338$, $PS_{\mathrm{pc}}= 0.0185$, $G_{\mathrm{pc}} = 0.00675 \text{ ml} \cdot \text{g}^{-1} \cdot \text{s}^{-1}$, $V'_{\mathrm{isf}}=0.338$, and $V'_{\mathrm{pc}}=0.469 \text{ ml} \cdot \text{g}^{-1}$. In Fig. 1, the open circles curve is the deoxyglucose outflow dilution curve, the solid line is the model solution optimized by GGOPT. The dotted line is the model solution at the initial guessed parameter values. The fitting of the model to the data was very consistent over the whole curve.

Acknowledgments

This work was supported by grant RR 01243 and EB08407 from the National Institutes of Health.

References

1. Dennis, JE.; Schnabel, RB. Numerical Methods of Unconstrained Optimization and Nonlinear Equation. Prentice-Hall; New York: 1983.
2. Glad T, Goldstein AA. Optimization of functions whose values are subject to small errors. BIT. 1977; 17:160–169.
3. Goldstein AA, Chan IS, Bassingthwaighte JB, Russak IB. On the minimization of functions with errors, (descriptive manuscript and code available on request).
4. Kuikka J, Levin M, Bassingthwaighte JB. Multiple tracer dilution estimates of D-, and 2-deoxy-D-glucose uptake by the heart. Am J Physiol (Heart Circ Physiol 19). 1986; 250:H29–H42.
5. More JJ, Garbow BS, Hillstrom KE. Testing unconstrained optimization software. ACM TOMS. 1981; 7/1:17–41.

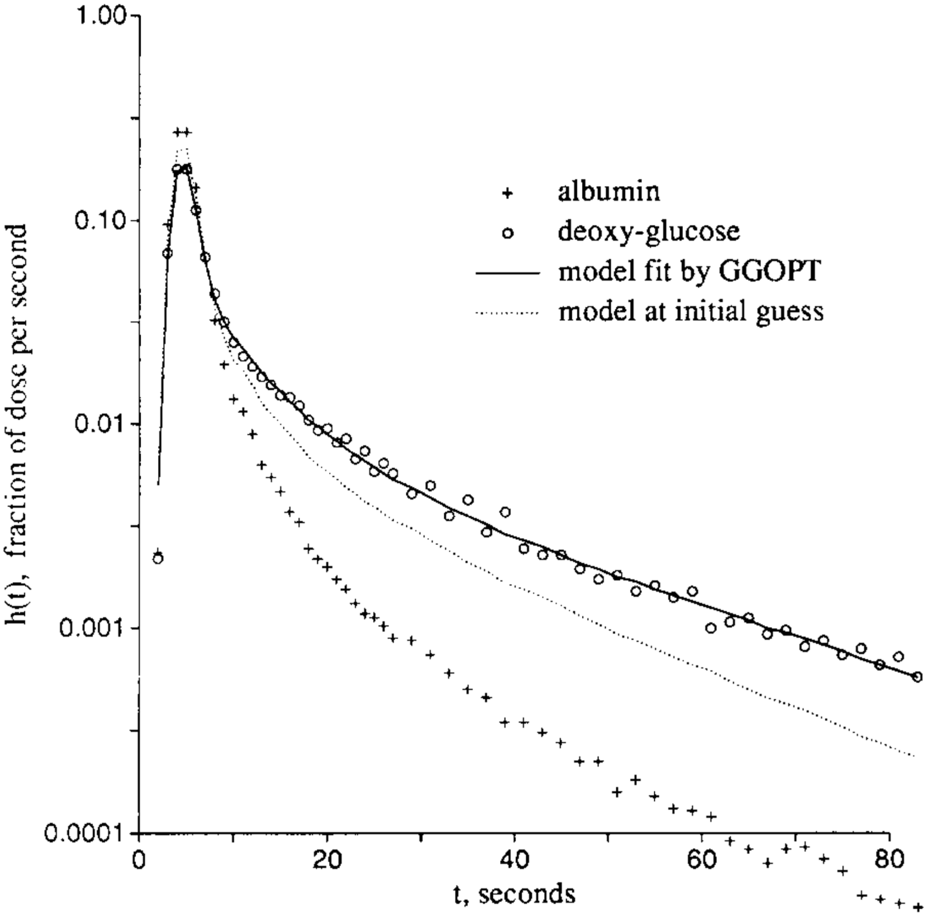

Fig. 1.

Fitting of the blood–tissue exchange model to the deoxyglucose outflow dilution curve of an isolated rabbit heart. The albumin dilution curve defines transport through the intravascular region without exchange with the tissue. From the initial solution (dotted line) using guessed parameter values, 11 iterations were used to reach the final solution (solid line).

TABLE 1

Results of Helical Valley function run

ε	*ITN*	$f(x)$	x_1	x_2	x_3
0.1E−6	10	0.109671E−18	0.100000E + 01	0.208957E−09	0.327986E−09
0.1E−3	13	0.110068E−13	0.100000E + 01	−0.65673E−07	−0.10486E−06
0.1E−2	22	0.369896E−09	0.100000E + 01	0.121430E−04	0.191907E−04
0.1E−1	27	0.242588E−07	0.999986E + 01	−0.24414E−04	−0.42822E−04
0.5E−1	29	0.215065E−04	0.100003E + 01	−0.29952E−02	−0.47317E−02

TABLE 2

Results of Rosenbrock function run

ε	*ITN*	$f(x)$	x_1	x_2
0.1E−06	22	0.000000E−00	0.100000E + 01	0.100000E + 01
0.1E−03	31	0.327410E−10	0.100001E + 01	0.100001E + 01
0.1E−02	26	0.336418E−08	0.100006E + 01	0.100011E + 01
0.1E−01	47	0.354608E−06	0.999406E + 00	0.998808E + 00
0.5E−01	44	0.428770E−02	0.932850E + 00	0.870257E + 00

TABLE 3

Results of Jennrich and Sampson function runs

ε	*ITN*	$f(x)$	x_1	x_2
0.1E−06	10	124.366	0.258766	0.256845
0.1E−03	7	124.355	0.257960	0.257597
0.1E−02	8	124.282	0.258399	0.256323
0.1E−01	36	124.283	0.251413	0.258590
0.5E−01	22	124.213	0.230784	0.272401