IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 11, Issue 2 Ver. III (Mar - Apr. 2015), PP 64-71
www.iosrjournals.org

Mathematical Model Formulation and Comparison Study of Various Methods of Root- Finding Problems

Arup Dey
Department of Industrial and Production Engineering / Shahjalal University of Science and Technology

Abstract: *A number of methods are available to find the root of function. The study is aimed to compare Bisection method, Newton-Raphson method and Secant method in term of time, iteration needed to get root in a desire level of error. A mathematical polynomial equation for beam deflection developed and the point at which maximum deflection occurs find by Bisection method, Newton-Raphson method and Secant method. We compare these root finding methods by using the software "MATLAB R2008a". It would seem obvious that Newton's method is faster, since it converges more quickly. However, to compare performance, we must consider both cost and speed of convergence [1]. From the above observation it is seen that the Bisection method converge at 25th iteration while Newton-Raphson method and Secant method converge at 3rd and 4th iteration respectively. In Newton-Raphson method two functions evaluate per iteration and in Secant method only a single function (from 2nd) evaluate per iteration. Then it was conclude that among three methods Secant method is converge faster than others. And it is most effective method.*

Keywords: *Mathematical model, Algorithm, Root, Iteration, Function, Beam deflection, Bisection method, Newton-Raphson method and Secant method, Execution time, Flops.*

I. Introduction

A root-finding algorithm is a numerical method, or algorithm, for finding a value x such that $f(x)$=0, for a given function f. Such an x is called a root of the function $f(x)$.

Root finding methods are use in a wide variety of practical applications in Physics, Chemistry, Biosciences, Engineering and so on. A root finding problem is a mathematical model of a physical system. Numerical root finding methods use iteration producing a sequence of numbers that hopefully converge towards a limit which is the root of the function. First value of the series is called initial/seed value. Newton-Raphson method, False position method, Bisection method, Fixed point iteration, Secant method are widely used root finding methods.

Different methods converge at different rates. Rate of convergence depends on initial value. That is, some methods are faster in converging to the root than others. The rate of convergence could be linear, quadratic or others [2].

The study is at comparing the rate of performance. For this we develop a polynomial equation of beam deflection and roots finding methods are used to find the point where maximum deflection takes place.

II. Mathematical Model Formulation Of Problem

Problem

In this paper, we developed a mathematical model for beam deflection of a given load distribution to analysis different root finding methods. Then, use root finding techniques to find the value maximum deflection and where take place. Beam given in below figure:

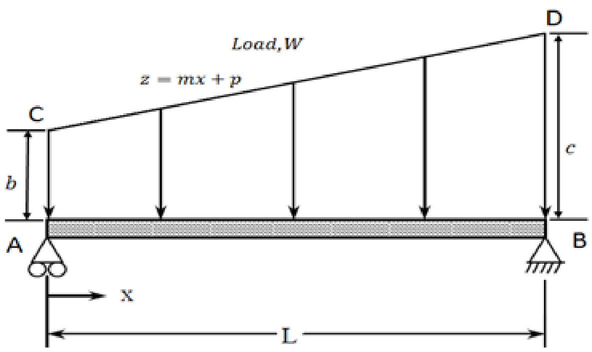

Fig-01: Simple supported beam

The dimensional parameters of the beam, $L = 48inches, b = 11inches, c = 18inches, W = 500lb$

Mathematical model formulation

Now consider point A is origin, and then the equation of CD is

$$z = mx + p$$

$$At\ x = 0, z = b\ and\ at\ x = L, z = c. so, from\ above\ equation, p = b\ and\ m = \frac{c-b}{L}.$$

$$z = \frac{c-b}{L}x + b = 0.14583x + 11$$

Now the distance of the centroid from point A, at which load acts is evaluate by using given formula[3].

$$\bar{x}_L = \frac{\sum A_i \bar{x}_i}{\sum A_i}$$

$$= \frac{bL \times \frac{L}{2} + \frac{1}{2}(c-b)L \times \frac{2}{3}L}{\frac{1}{2}L(b+c)}$$

$$= \frac{1}{3}\left(\frac{b+2c}{b+c}\right)L$$

So, load per unit area, $W_\eta = \frac{W}{\frac{1}{2}L(b+c)} = \frac{500}{\frac{1}{2}\times 48(11+18)} = 0.71839lb/in^2$

Now let us find the deflection in the beam [4]. The deflection *y* as a function of *x* along the length of the beam is given by

$$\frac{d^2y}{dx^2} = \frac{M(x)}{EI}$$

$where,\quad M = Bending\ moment(Nm),\quad E = Young'smodulus\ (Nm^{-2},$
$I = 2nd\ moment\ of\ inertia(m^4)$

To find the bending moment, we need to first find the reaction force at the support. Let R_A and R_B be the reactions at the left and right support, respectively. Then from the sum of forces in the vertical direction,

$$R_A + R_B = W$$

And the moment about support A is zero as it is simply supported.

$$R_B L = W \times \frac{1}{3}\left(\frac{b+2c}{b+c}\right)L$$

$$R_B = W \times \frac{1}{3}\left(\frac{b+2c}{b+c}\right) = 270.11494lb \quad [Substituting\ value\ of\ W, b, c]$$

$$So, R_A = 500lb - 270.11494lb = 229.88506lb$$

The distance of the centroid from the right (Fig: 2) bending moment at any cross-section at a distance of *x* from the end A is then given by summing the moments at a cross-section of distance *x* from A as (Figure)

$$\bar{x}_x = \frac{xb \times \frac{x}{2} + \frac{1}{2}x \times \left(\frac{c-b}{L}x + b - b\right) \times \frac{1}{3}x}{\frac{1}{2}x(b + \frac{c-b}{L}x + b)}$$

$$= \left(\frac{\frac{1}{2}b + \frac{1}{6}\frac{c-b}{L}x}{b + \frac{c-b}{2L}x}\right)x$$

And

$$M(x) + W_\eta \times \frac{1}{2}x(b + \frac{c-b}{L}x + b) \times \left(\frac{\frac{1}{2}b + \frac{1}{6}\frac{c-b}{L}x}{b + \frac{c-b}{2L}x}\right)x - R_A x$$

$$M(x) = R_A x - W_\eta \times \left(\frac{1}{2}b + \frac{1}{6}\frac{c-b}{L}x\right)x^2$$

$$M(x) = R_A x - \frac{1}{2}W_\eta b x^2 - \frac{1}{6}\frac{c-b}{L}W_\eta x^3$$

$Substituting\ the\ value\ of\ R_A, W_\eta, L, b, c$

$$M(x) = 229.88506x - 3.95115x^2 - 0.01746x^3$$

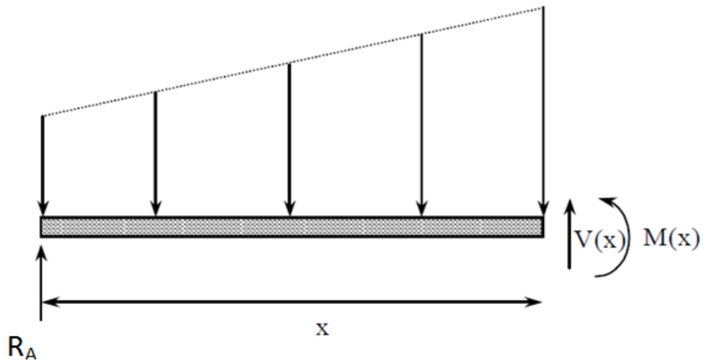

Fig-02: Free body diagram of beam

$$We\ know\ that, \frac{d^2y}{dx^2} = \frac{M(x)}{EI}$$

$$EI\frac{d^2y}{dx^2} = M(x) = 229.88506x - 3.95115x^2 - 0.01746x^3$$

$Integrating\ with\ respect\ to\ x\ gives$

$$EI\frac{dy}{dx} = 114.94253x^2 - 1.31705x^3 - 0.0043652x^4 + C_1$$

$Integrating\ once\ again\ \ with\ respect\ to\ x\ gives$

$$EIy = 38.31418x^3 - 0.32926x^4 - 0.00087304x^5 + C_1x + C_2$$

$$Now\ the\ boundary\ conditions\ at\ x = 0\ and\ x = 48\ are\ that\ displacement\ is\ zero. So,$$
$$C_2 = 0\ and\ \ C_1 = -4.72276 \times 10^4$$

$$So, EIy = 38.31418x^3 - 0.32926x^4 - 0.00087304x^5 - 4.72276 \times 10^4 x$$

Then, the vertical deflection in the beam is given by,

$$y = \frac{1}{EI}(38.31418x^3 - 0.32926x^4 - 0.00087304x^5 - 4.72276 \times 10^4 x)$$

But to find where the deflection maximum, we need to take the first derivative of the deflection to find where

$$\frac{dy}{dx} = 0$$
$$f(x) = \frac{dy}{dx} = 0$$

$f(x) = 114.94253x^2 - 1.31705x^3 - 4.36522 \times 10^{-3}x^4 - 4.72276 \times 10^4$

$f'(x) = 229.88506x - 3.95115x^2 - 0.174609x^3$

III. Methods

A number of methods are available to find the root of function. We compare only three (3) methods. These are:

- Bisection method
- Secant method
- Newton-Raphson method

The Bisection method

The Bisection Method is the most primitive method for finding real roots of function $f(x) = 0$ where f is a continuous function is a given interval. This method is also known as Binary-Search Method and Bolzano Method. Two initial guess is required to start the procedure. This method is based on the Intermediate value theorem: if function f(x) = 0 is continuous between $f(a)$and $f(b)$and have opposite signs and less than zero, then there is at least one root.

Steps of Bisection methods are given below [5]:

Step 1: Choose lower a upper b guesses for the root such that the function change sign over the interval. This can be check by ensuring that

$$f(a)f(b) < 0$$

Step 2: An estimate of the root x_r is determined by

$$x_r = \frac{a+b}{2}$$

Step 3: Make the following evaluation to determine in which subinterval the root lies:

a) If$f(a)f(x_r) < 0$, the root lie in the lower subinterval. Therefore, set $b = x_r$ and return to step 2.

b) If$f(a)f(x_r) > 0$, the root lie in the upper subinterval. Therefore, set $a = x_r$ and return to step 2.

c) If$f(a)f(x_r) = 0$, the root equal to x_r terminate iteration.

Convergence

Let $a_0 = a$and $b_0 = b$and $[a_n, b_n](n \geq 0)$ are the successive intervals in the Bisection process [6].
Clearly

$$a_0 \leq a_1 \leq a_2 \leq \cdots \leq b_0 = b$$

and

$$b_0 \geq b_1 \geq b_2 \geq \cdots \geq a_0 = a$$

Now the sequence $\{a_n\}$ is monotonic increasing and bounded above and the sequence $\{b_n\}$ is monotonic decreasing and bounded below. Hence both the sequence converges. Further,

$$b_n - a_n = \frac{b_{n-1} - a_{n-1}}{2} = \frac{b_{n-2} - a_{n-2}}{2^2} = \cdots = \frac{b_0 - a_0}{2^n}$$

Hence $\lim_{n\to\infty} a_n = \lim_{n\to\infty} b_n = \alpha$. Further taking limit in $f(a_n)f(b_n) \leq 0$ we get $[f(\alpha)]^n \leq 0$ and that implies $f(\alpha) = 0$. Hence, a_n and b_n converges to a root of $f(x) = 0$.
Let us apply the Bisection method to the interval$[a_n, b_n]$and calculate midpoint$c_n = \frac{a_n + b_n}{2}$. Then the root lies either in$[a_n, c_n]$ or $[c_n, b_n]$. In either case

$$|\alpha - c_n| \leq \frac{b_n - a_n}{2} = \frac{b-a}{2^{n+1}}$$

Hence $c_n \to \alpha$ as $n \to \infty$
Now

$$|e_{n+1}| = |\alpha - c_{n+1}| \leq \frac{b_{n+1} - a_{n+1}}{2} = \frac{1}{2}\frac{b_n - a_n}{2}$$

and

$$|e_n| = |\alpha - c_{n+1}| \leq \frac{b_n - a_n}{2}$$

Thus we find

$$|e_{n+1}| \sim \frac{1}{2}|e_n|$$

Hence the Bisection method converges linearly.

The Newton-Raphson method

Perhaps the most widely used of all root-locating formulas is the Newton-Raphson equation. This method may also be developed from the Taylor series expansion.
Taylor series expansion can be represented as

$$f(x_{n+1}) = f(x_n) + f'(x_n)(x_{n+1} - x_n) + \frac{f''(x_n)}{2!}(x_{n+1} - x_n)^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots(1)$$

An approximate version is obtained by truncating the series after the first derivative term:

$$f(x_{n+1}) \cong f(x_n) + f'(x_n)(x_{n+1} - x_n)$$

At the interaction with x axis $f(x_{n+1})$ would be equal to zero, or

$$f(x_n) + f'(x_n)(x_{n+1} - x_n) = 0 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(2)$$

Which can be solved for

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Which is called Newton-Raphson formula.

Convergence

Now, consider, $x_{n+1} = x_r$ where x_r is the value of the root [5]. So, $f(x_r) = 0$. We get from equation (1),

$$0 = f(x_n) + f'(x_n)(x_r - x_n) + \frac{f''(x_n)}{2!}(x_r - x_n)^2 \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots(3)$$

Equation (2) can be subtracted from equation (3). Then,

$$0 = f'(x_n)(x_r - x_{n+1}) + \frac{f''(x_n)}{2!}(x_r - x_n)^2 \ldots\ldots\ldots(4)$$

Now, realize that the error is equal to discrepancy between true value and current value. Then,

$$e_{n=}x_r - x_n$$
and
$$e_{n+1=}x_r - x_{n+1}$$

From equation (3), we get,

$$0 = f'(x_n)(e_{n+1}) + \frac{f''(x_n)}{2!}{e_n}^2$$

If we assume convergence, x_n should eventually be approximate by the root, x_r. Now,

$$e_{n+1} = \frac{-f''(x_r)}{f'(x_r)}{e_n}^2.$$

The error is roughly proportional to the square of the previous error. Hence the Newton-Raphson method convergence is quadratic.

The Secant method

As we have noticed, the main setback of the Newton-Raphson method is the requirement of finding the value of the derivative of f(x) per iteration. There are some functions that are either extremely difficult (if not impossible) or time consuming. For these, the derivative can be approximated by a backward finite divided difference [5],

$$f'(x_n) \cong \frac{f(x_{n-1}) - f(x_n)}{x_{n-1} - x_n}$$

This approximation can be substitute into the Newton-Raphson method formula,

$$x_{n+1} = \frac{f(x_{n-1})x_n - f(x_n)x_{n-1}}{f(x_{n-1}) - f(x_n)} \ldots\ldots\ldots\ldots(5)$$

$$x_{n+1} = x_n - \frac{x_{n-1} - x_n}{f(x_{n-1}) - f(x_n)} f(x_n)$$

This is Secant method formula.

Convergence

Say, $e_{n-1,}e_{n,}e_{n+1}$ are error at (n-1)th, nth and (n+1)th iterations respectively and $e_{n+1} = k\, {e_n}^p$, where k is constant and rate of convergence at rate p[7]. The true value of the function x_r and $e_{n=}x_r - x_n$

$$x_{n-1} = x_r - e_{n-1}$$
$$x_n = x_r - e_n$$
$$x_{n+1} = x_r - e_{n+1}$$

So, from equation (5). We get,

$$e_{n+1} = \frac{f(x_{n-1})e_n - f(x_n)e_{n-1}}{f(x_{n-1}) - f(x_n)} \ldots\ldots\ldots\ (6)$$

From the mean value theorem, $x = \eta_n$ in the interval x_n and x_r.

$$f^{'}(\xi\eta_n) = \frac{f(x_n) - f(x_r)}{x_n - x_r}$$

Or

$$f^{'}(\eta_n) = \frac{f(x_n)}{e_n} \text{or} f(x_n) = e_n f^{'}(\eta_n)$$

Similarly, $f(x_{n-1}) = e_{n-1} f^{'}(\eta_{n-1})$

Now, substituting the value in equation (5). We get,

$$e_{n+1} = e_{n-1} e_n \frac{f^{'}(\eta_{n-1}) - f^{'}(\eta_n)}{f(x_{n-1}) - f(x_n)}$$

i.e$e_{n+1} = k e_{n-1} e_n$

Also,

$$e_{n+1} = k\, {e_n}^p$$

From above two equation,

$${e_n}^p = k e_{n-1} e_n = k\, {e_n}^{\frac{1}{p}} e_n = k\, {e_n}^{\frac{1+p}{p}}$$

So,

$$p = \frac{1+p}{p}\ or\ p^2 - p - 1 = 0.$$

i.e. $p > 0, p = \frac{1+\sqrt{5}}{2} = 1.618$

Hence, convergence is super linear.

IV. Convergence Rates Of Bisection, Newton-Raphson And Secant Methods

Rate of convergence of Bisection, Secant, Newton-Raphson method are respectively linear, super linear and quadric. Bisection method converge rate is low and Newton-Raphson method converge rate is higher method. Bisection method will always converge. This shows that Newton-Raphson method converges quadratically. By implication, the quadratic convergence we mean that the accuracy gets doubled at each iteration. But there is no guarantee that Newton- Raphson method converges. If the initial values are not close enough to the root, then there is no guarantee that the Secant method converges. If f is differentiable on that interval and there is a point where $f^{'} = 0$ on the interval, then the algorithm may not converge [9].

V. Comparisons analysis

Comparisons of roots finding methods based on

- No. of iterations needed to get desire result
- Execution time
- Operations counting

No. of iterations

The Bisection method, Newton-Raphson method and Secant method were applied to a function:
$f(x) = 114.94253x^2 - 1.31705x^3 - 4.36522 \times 10^{-3}x^4 - 4.72276 \times 10^4$

On interval [0, 48], using the software *"MATLAB R2008a"*. The results are presented in table 1 to 3.

Table 01: Iteration data for Bisection method with $a = 0, b = 48, E_t = 0.0000001$

$Iteration, i$	x_r	$Error, E_i$ (%)	$Iteration, i$	x_r	$Error, E_i$ (%)
0	0.000000	-	13	24.228516	0.024184
1	24.000000	100.000000	14	24.225586	0.012093
2	36.000000	33.333333	15	24.224121	0.006047
3	30.000000	20.000000	16	24.224854	0.003023
4	27.000000	11.111111	17	24.225220	0.001512
5	25.500000	5.882353	18	24.225037	0.000756
6	24.750000	3.030303	19	24.225128	0.000378

7	24.375000	1.538462	20	24.225174	0.000189
8	24.187500	0.775194	21	24.225197	0.000094
9	24.281250	0.386100	22	24.225208	0.000047
10	24.234375	0.193424	23	24.225214	0.000024
11	24.210938	0.096805	24	24.225211	0.000012
12	24.222656	0.048379	25	24.225213	0.000006

Table 1 shows the iteration data obtained for Bisection method with the aid of *"MATLAB R2008a"*. It was observed that the function converges to 24.225213 at the 25th iterations with error level of 0.000006.

Table 02: Iteration data for Newton-Raphson with $x_1 = 24, E_t = 0.0000001$

$Iteration, i$	x_{i+1}	$Error, E_i$ (%)
0	24.000000	-
1	24.225292	0.929988
2	24.225214	0.000324
3	24.225214	0.000000

From Table 2, we noticed that the function converges to 24.225214 at the 3rd iteration with error 0.000000.

Table 03: Iteration data for Secant Method with $x_0 = 20,\ x_1 = 30, E_t = 0.0000001$

$Iteration, i$	x_{i+1}	$Error, E_i$ (%)
0	30.000000	-
1	24.216635	19.277883
2	24.225320	0.035863
3	24.225214	0.000436
4	24.225214	0.000000

Table 3 revealed that the function converges to 24.225214 at the 4th iteration with error 0.000000.
From foresaid table, number of iteration for Bisection method is too much high compare to other two methods.
Now, we compare Newton-Raphson method and Secant method based on execution time and operation counting.

Execution time

The execution time of a given task is defined as the time spent by the system executing that task, including the time spent executing run-time or system services on its behalf**.**
Table 4 to 5 shows time for iteration in both methods.

Table 04: Execution time comparison for three (03) iterations

Serial No.	**Newton-Raphson method**	**Secant method**
	Execution Time	**Execution Time**
1	0.002178	0.001876
2	0.002145	0.002012
3	0.002153	0.002028
4	0.002074	0.002038
5	0.002126	0.002038

Table 05: Execution time comparison for four (04) iterations

Serial No.	**Newton-Raphson method**	**Secant method**
	Execution Time	**Execution Time**
1	0.002645	0.002472
2	0.002698	0.002077
3	0.002532	0.002032
4	0.002637	0.002111
5	0.002792	0.002235

From above two tables, execution time for Newton-Raphson method is higher than Secant method. The secant method requires only one function evaluation per iteration, since the value of $f(x_{i-1})$ can be stored from the previous iteration. Newton's method requires one function evaluation and one evaluation of the derivative per iteration.

Operation counting

The performance depends on the amount of floating point operations (or flops) involved in the algorithms. On modern computer use math coprocessors, the time consumed to perform addition/subtraction and multiplication/division is about the same [5]. Therefore totaling up these operations provides insight into which parts of the algorithm are most time consuming and how computation time increases as the system get larger.

Table 06: No. of flops in various equations

Equation	$f(x) = 114.94253x^2 - 1.31705x^3 - 4.36522 \times 10^{-3}x^4 - 4.72276 \times 10^4$
Addition/Subtraction flops	3
Multiplication/division flops	9
Equation	$f'(x) = 229.88506x - 3.95115x^2 - 0.174609x^3$
Addition/Subtraction flops	2
Multiplication/division flops	6
Equation	$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$
Addition/Subtraction flops	1
Multiplication/division flops	1
Equation	$x_{n+1} = x_n - \frac{x_{n-1} - x_n}{f(x_{n-1}) - f(x_n)} f(x_n)$
Addition/Subtraction flops	3
Multiplication/division flops	2

Table 07:No. of flops in Newton-Raphson method

Iteration	Addition/Subtraction flops	Multiplication/division flops
1	6	16
2	6	16
3	6	16
.	.	.
.	.	.
.	.	.
n	6	16

So, total flops= $6n + 16n = 22n$

Table 08: No. of flops in secant method

Iteration	Addition/Subtraction flops	Multiplication/division flops
1	9	20
2	6	11
3	6	11
.	.	.
.	.	.
.	.	.
n	6	11

So, total flops= $9 + 6(n-1) + 20 + 11(n-1) = 17n + 12$

In first two iteration No. of flops is higher in Secant method. But then from 3rd iteration no. flops is higher in Newton-Raphson method. Comparing the results of the three methods under investigation, we observed that the rates of convergence of the methods are in the following order:

Secant method > Newton-Raphson method > Bisection method.

VI. Conclusion

Based on our analysis and results, we now conclude that among three methods Secant method is most effective. Newton-Raphson method is convergence quicker than others two method, but requires two function evaluation per iteration. Secant method requires only a single function evaluation per iteration. We also conclude that Bisection provide grantee of convergence, its convergence rate is too slow and difficult to use for system of equation.

References

[1]. Ron, Amos (2010), Lecture notes on CS412: Solving Equations and Polynomial Interpolation.

[2]. Ehiwario, J.C., Aghamie, S.O. (2014), Comparative Study of Bisection, Newton-Raphson and Secant Methods of Root- Finding Problems. IOSR Journal of Engineering, vol. 04 pp 1-7.

[3]. Bansal, Dr. R. K. (2005), A Textbook of Engineering Mechanics. Laxmi Publications. pp 154-162

[4]. Bansal, Dr. R. K., A textbook of Strength of materials, 4th edition. pp 515-521.

[5]. Chapra, Steven C. and Canale, Raymond P. (2010), Numerical Methods or Engineers. Sixth edition.

[6]. Anonymous, Solution of Nonlinear equations. http://www.slideshare.net/nakulrtm/root-37376075 (accessed 20 February, 2015)

[7]. Kumar, Rathish, Convergence of secant method:http://nptel.ac.in/courses/122104019/numerical-analysis/Rathish-kumar/ratish-1/f3node5.html _(Accessed 01 March, 2015

[8]. Kaw, Autar and Kalu, Egwu Eric (2011), NUMERICAL METHODS WITH APPLICATIONS.2nd edition.

[9]. Wikipedia, the free Encyclopedia.