Circular Path Autopilot Navigation System

Nicholas DeCicco[1], Kristine Napolitano[1], Russell Trafford[1]
Petia Georgieva[1,2], Nidhal Bouaynaya[1] and Robi Polikar[1]
[1]Department of Electrical and Computer Engineering
Rowan University, Glassboro, NJ U.S.A., (traffo17@students.rowan.edu)
[2]Department of Electronics Telecommunications and Informatics /IEETA
University of Aveiro, Aveiro, Portugal, (petia@ua.pt)

Abstract—**In this paper we propose an autopilot control strategy for a small radio controlled aircraft aiming to follow a circular path with a desired radius. In contrast to the usual way to compute the error as the distance between the plane current space position and the desired flight trajectory path (for example a circle of radius r_0), here we propose an alternative way to compute the scalar quantity $E(t)$) such that the aircraft moves in a spiral path of a decreasing/increasing with time radius to approach the circle with desired radius r_0. Our particular choice of error term is intended to cause only gradual changes in the aircraft's direction so as to prevent unstable flight due to over-control.**

I. Introduction

The major challenge in designing an autopilot control system is to guarantee the aircraft's flight does not become unstable or erratic, as the aircraft can very quickly enter a state from which it is not possible for the autopilot to regain stability [1]. The literature analysis shows that the autopilot navigation systems consist of the following control subsystems, [2], [3]: i) stabilizing flight system (i.e. the aircraft can travel in a straight line), ii) system which can hold a specified bank angle, and iii) system which can hold a specified rate of rotation. The combination of the above subsystems provides an autopilot capable of navigating between waypoints. Waypoints are set of coordinates that identify a point in physical space. For terrestrial navigation the coordinates are usually longitude and latitude, for air navigation also includes altitude. If the autopilot can stabilize the flight, hold a specified bank angle and a rate of rotation it can change the heading of the aircraft and achieve a desired location. The autopilot can achieve this by the deflection of the control surfaces from their center positions. This imparts rotational moments on the airframe which alter the aircraft's orientation in space. A major concern in the autopilot design is to define a control strategy that will prevent the aircraft to drive into an unstable region. For example, continuously applying an up elevator control signal of too high amplitude can cause the aircraft to enter a vertical loop. Similarly, for an aircraft with ailerons, continuously applying aileron control will cause the aircraft to enter a roll. Both of these are undesirable conditions as a result of over-control, [4].

In this paper we propose a particular choice of error term to cause smooth change of the aircraft direction in order to prevent unstable flight due to over-control. The objective of the autopilot navigation system is to follow a circular path with a desired radius. Starting from an arbitrary initial position, the proposed error correction strategy computes a trajectory to smoothly drive the aircraft to encircle a desired location.

II. Test aircraft platform

The components of our test aircraft platform are illustrated in Figure 1, [5].
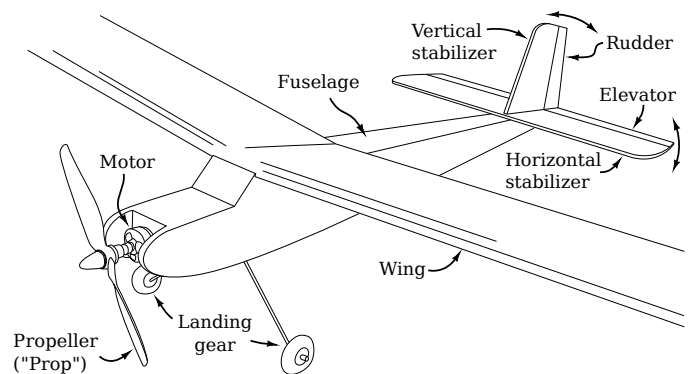

Figure 1: Components of the test aircraft platform .

The test aircraft consists of a *fuselage* to which the *wing* is attached with rubber bands to permit removal to access the interior of the fuselage, in which we have placed the various electronics and servos which operate the plane. At the front of the plane (the *nose*) is the propeller, driven by a *brushless motor*. The motor is driven by an *electronic speed controller* (ESC) which draws power from a lithium polymer battery. At the rear of the plane is the *tail*, which is comprised of a *horizontal stabilizer*, to which the *elevator* is attached, and the *vertical stabilizer*, to which the *rudder* is attached. The elevator and rudder are permitted to rotate at the point at which they attach to the stabilizers. Collectively, the rudder and elevator are referred to as the aircraft's *control surfaces*, and it is the angle of these surfaces relative to the vertical and horizontal stabilizers to which they are attached that we can control using servo motors.

A small, radio-controlled model aircraft was built from a kit for this project (Figure 2). The specifications of this model plane are listed in Table I and the Bill of materials in Table II.

Figure 2: *Left*: A view of the aircraft used for testing. *Top right*: A view inside the fuselage (directly under the wing) where the servos which control the tail control surfaces reside; the servos (center) are connected to the tail surfaces by piano wire (center to right; the front of the plane is to the left in the image). *Bottom right*: A view from underneath of the tail, showing how the servo control rods attach to the elevator and rudder control surfaces.

Table I: AeroCraft Micro Telemaster model specifications.

Wingspan	35.5 in (90.2 cm)
Wing area	192 in^2 (1238.7 cm^2)
Wing loading	6 oz/ft^2
Length	24 in (61.0 cm)

III. Waypoint Navigation

Our strategy to permit navigating the plane to encircle a fixed location in space is described in this section. Taking into account that the operation region of the autopilot is very small with respect to the Earth, latitude (ϕ) and longitude (λ) variables can be also treated as Cartesian coordinates. The Great Circle distance [6] can be therefore approximated as the Euclidean distance:

$$d \approx \sqrt{(\lambda_1 - \lambda_2)^2 + (\phi_1 - \phi_2)^2}$$

We will henceforth use x and y as synonyms for λ and ϕ. Further, vectoral notation will be used;

$$\vec{v} = v_x \hat{i} + v_y \hat{j} + v_z \hat{k} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$$

Note also that we denote vectors with the arrow ($\vec{\ }$) and normalized vectors with the circumflex ($\hat{\ }$); e.g., $\hat{a} = \vec{a}/||\vec{a}||$. It is also assumed that the magnitude of the plane's velocity is constant:

$$||\vec{v}|| = v,$$

where v is a real positive scalar number. To ensure a constant magnitude of the velocity, we define $\vec{v}$ lying on a circle of radius v, and having an angle θ relative to the $+x$ axis:

$$\vec{v} = v \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix}.$$

A. Computing the Error Term

The error term E is usually computed as the distance between the plane current space position and the desired flight path (for example a circle of radius r_0). Here we propose an alternative way to compute the scalar quantity $E(t)$) such that the aircraft moves in a spiral path of a decreasing/increasing with time radius to approach the circle with desired radius r_0. Our particular choice of error term is intended to cause only gradual changes in the aircraft's direction so as to prevent unstable flight due to over-control.

Let us consider the arrangement in Figure 3: here, the plane is at a location $\vec{p}$, and the goal is to fly towards the circle of radius r_0 circumscribed about the origin. We define the error term as the z-component of the cross product of the plane's normalized velocity vector $\hat{v}$ (i.e. its current heading) with the desired heading direction, $\hat{v}_d$:

$$E(t) = (\hat{v}_d(t) \times \hat{v}(t)) \cdot \hat{k}. \tag{1}$$

With this definition of $E(t)$, if the desired ($\hat{v}_d$) and actual ($\hat{v}$) headings coincide (i.e., if $\hat{v}_d = \hat{v}$), the error $E(t)$ is zero. Otherwise, the error $E(t)$ is the sine of the angle between the vectors $\hat{v}_d$ and $\hat{v}_d$, and the error is maximum if $\hat{v}_d \perp \hat{v}$. Further
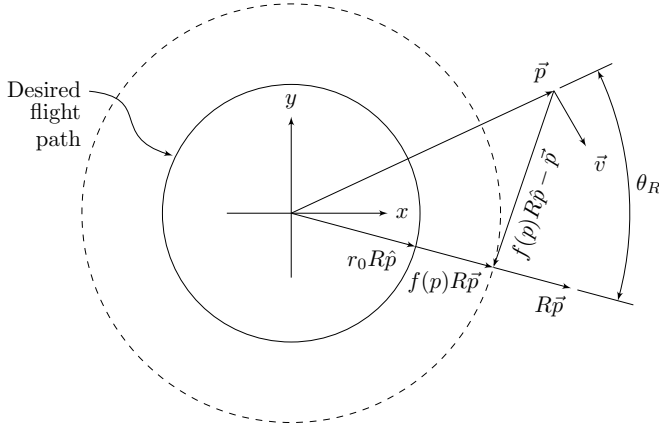
Figure 3: An illustration of the vectors used to compute $E(t)$ for the waypoint navigation control loop.

to that, using the z-component of the cross product is distinct from the *magnitude* of the cross product even in this two-dimensional case as it preserves the sign of the cross product. This definition of the error has the added benefit of indicating if the plane is headed clockwise or counter-clockwise by the sign of the cross product.

Next, we argue that our choice of a desired heading (desired direction of travel) $\vec{v}_d$ is defined as

$$\vec{v}_d = f(p)R\hat{p} - \vec{p}, \qquad (2)$$

where $f(p)$ is a function which acts on the length of the vector $||\vec{p}|| = p$, and R is the 2D rotation matrix

$$R = \begin{bmatrix} \cos\theta_R & -\sin\theta_R \\ \sin\theta_R & \cos\theta_R \end{bmatrix}.$$

(The derivation of R can be found in any introductory linear algebra text.)

The rationality behind this formulation of the desired heading is as follows: the product $R\vec{p}$ rotates the plane's current position in space through an angle θ_R about the origin. If we define

$$\vec{v}_d = R\vec{p} - \vec{p},$$

(for small θ_R) the plane would forever orbit the origin at its present radius. The function $f(p)$ shortens the vector $R\vec{p}$ if $p > r_0$, lengthens $R\vec{p}$ if $p < r_0$, and keeps $R\vec{p}$ the same if $p = r_0$. In this way, we direct the plane to fly in a circle of a radius which asymptotically approaches r_0. The function $f(p)$ which gives this result is

$$f(p) = r_0 + (1 - \varepsilon)(p - r_0),$$

where ε is some small scalar quantity, a parameter which controls the rate at which the aircraft turns and diminishes to approach r_0.

Now, revisiting Equation (1) and taking into account that we consider the motion of the aircraft in the plane (i.e., the two-dimensional case of motion), the error is expressed as

$$E = \frac{v_{dx}v_y - v_{dy}v_x}{||\vec{v}|| ||\vec{v}_d||}. \qquad (3)$$

where

$$\vec{v} = v_x\hat{i} + v_y\hat{j},$$
$$\vec{v}_d = v_{dx}\hat{i} + v_{dy}\hat{j}.$$

Let reduce this sequence of computations to a single equation. The product $R\hat{p}$ is:

$$R\vec{p} = \begin{bmatrix} \cos\theta_R & -\sin\theta_R \\ \sin\theta_R & \cos\theta_R \end{bmatrix} \left(\frac{1}{||\vec{p}||} \begin{bmatrix} p_x \\ p_y \end{bmatrix} \right)$$

$$= \frac{1}{||\vec{p}||} \begin{bmatrix} p_x\cos\theta_R - p_y\sin\theta_R \\ p_x\sin\theta_R + p_y\sin\theta_R \end{bmatrix}.$$

Then, $\hat{v}_d$ is obtained from Equation (2):

$$\hat{v}_d = \frac{f(p)}{||\vec{p}||} \begin{bmatrix} p_x\cos\theta_R - p_y\sin\theta_R \\ p_x\sin\theta_R + p_y\sin\theta_R \end{bmatrix} - \vec{p}$$

$$= \begin{bmatrix} \frac{f(p)}{||\vec{p}||}(p_x\cos\theta_R - p_y\sin\theta_R) - p_x \\ \frac{f(p)}{||\vec{p}||}(p_x\sin\theta_R + p_y\cos\theta_R) - p_y \end{bmatrix}.$$

Finally, using Equation (3), the error term is

$$E = \left[\frac{f(p)}{||\vec{p}||}(p_x\cos\theta_R - p_y\sin\theta_R) - p_x \right] \frac{v_x}{||\vec{v}||} - \left[\frac{f(p)}{||\vec{p}||}(p_x\sin\theta_R + p_y\cos\theta_R) - p_y \right] \frac{v_x}{||\vec{v}||}.$$

We denote this error as the *turn error*. It is a measure of how much the plane needs to turn. This is not the error which describes how far the plane is from its desired path. In contrast, the error with regards to "distance" is simply

$$E_d(t) = ||\vec{p}|| - r_0,$$

The error is ultimately converted to an angle by taking the arc sine, then integrating E to obtain the new heading θ:

$$\theta(t) = \theta_0 + \int_0^t \arcsin E(\tau) \, d\tau,$$

where θ_0 is the initial heading of the plane. The reason for using the arcsine is because the error was computed as a cross product. Recall that the cross product yields the sine of the angle between two vectors:

$$\vec{a} \times \vec{b} = \frac{\sin(\phi)}{||\vec{a}|| ||\vec{b}||}.$$

B. Computing the space position from acceleration

Our autopilot hardware contains a micro-electromechanical (MEMS) accelerometer, which provides the acceleration of the aircraft at discrete time intervals. Using the relation between position $\vec{p}$, velocity $\vec{v}$ and acceleration $\vec{a}$ we obtain the following system of equations

$$\begin{cases} \dfrac{d\vec{p}}{dt} = \vec{v} \\ \dfrac{d\vec{v}}{dt} = \vec{a} \end{cases}$$

3

Using Euler integration, we relate the velocity $\vec{v}_{\text{new}}$ at a timestep i with the velocity at a timestep $i-1$, $\vec{v}_{\text{old}}$:

$$\vec{v}_{\text{new}} = \vec{v}_{\text{old}} + \vec{a}\Delta t,$$

where Δt is the time between successive timesteps. Equivalently

$$v_x^{\text{new}} = v_x^{\text{old}} + a_x\Delta t,$$
$$v_y^{\text{new}} = v_y^{\text{old}} + a_y\Delta t,$$
$$v_z^{\text{new}} = v_z^{\text{old}} + a_z\Delta t.$$

Integrating numerically again yields the plane's position in space:

$$p_x^{\text{new}} = p_x^{\text{old}} + v_x\Delta t,$$
$$p_y^{\text{new}} = p_y^{\text{old}} + v_y\Delta t,$$
$$p_z^{\text{new}} = p_z^{\text{old}} + v_z\Delta t.$$

Thus, obtaining the position of the plane in space is a matter of twice integrating the plane's acceleration. However, due to the inevitable accumulation of error over time in performing this approximate numerical integration, we cannot trust the result of integration to be accurate beyond a short period of time. For this (and other) reasons, our autopilot hardware also has a GPS receiver, which periodically (at a rate of about 1 Hz) gives the plane's position in space with an accuracy of a few meters. Output from the GPS then becomes the initial position, $\vec{p}_0$, of the plane in the integration each time we receive a new GPS reading on the plane's location.

We simulated this flight path control strategy and the results are summarized in Figures 4 to 6. The smoothness of the plane trajectory while moving towards the desired circle is defined by the rotational matrix R. Reducing θ_R $(-\frac{\pi}{2}, -\frac{\pi}{8}, -\frac{\pi}{16})$ makes the change of the plane direction milder and increases the stability margin. However, the price to pay are the slower moves and the nonzero steady-state errors.

Table II: Aircraft Bill of Materials

Mfg.	Mfg. PN	Supplier	Supplier PN	Description	Qty.	Price	Price
Du-Bro	LYT35	–	–	Foam Wheels	1		
Top Flite	TOPQ0227	–	–	6′ "True Red" MonoKote Covering Material	1	$ 16.99	$ 16.99
AeroCraft	–	Hobby Express	HLA192	Micro Telemaster RC Airplane Kit	1	$ 27.90	$ 27.90
Spektrum	SPMR6630	–	–	DSMX 6-channel transmitter	1	$129.99	$129.99
Spektrum	SPMAR500[3]	–	–	AR500 5-channel receiver	1	$ 59.99	$ 59.99
E-flite	EFLM1150	–	–	Park 300 brushless outrunner motor	1	$ 49.99	$ 49.99
E-flite	ELFB8002SJ	–	–	7.4V 800 mAh 2S Lithium polymer battery	1	$ 16.99	$ 16.99
E-flite	EFLA1010	–	–	10A electronic speed control (ESC) with BEC	1	$ 27.99	$ 27.99
Cox	005500	–	–	6 g micro servos	2	$ 11.99	$ 23.98
Du-bro	852	Hobby Express	DUB852	30″, 0.32″ push rod set	1	$ 7.28	$ 7.28
Du-bro	848	Hobby Express	DUB848	Control horn set	1	$ 1.22	$ 1.22
APC	LP 07050EP	Hobby Express	APC07040SF	7×4 composite "slowflyer" prop (for electric)	1	$ 1.89	$ 1.89

IV. CONCLUSIONS

The purpose of the waypoint navigation system proposed in this paper is to encircle a fixed location in space. Instead of a distance error between the current and the desired space locations of the aircraft, we propose a new error term denoted as a turn error that computes how much the plane needs to turn at each sampling time. Future work is to embed the new error term into a closed loop scheme with a PID controller to eliminate the steady state error and eventually make the navigation faster.

ACKNOWLEDGMENT

REFERENCES

[1] J. S. Berndt, "JSBSim: An open source flight dynamics model in C++", 2011. http://jsbsim.sourceforge.net/JSBSimReferenceManual.pdf

[2] Sooriyakumara, Ch., S. V. Ragavan, S. Madhavan. Waypoint Navigated Unmanned Aerial Vehicle Autopilot System. Trends in Intelligent Robotics, Automation, and Manufacturing, Series Communications in Computer and Information Science, Vol. 330, pp 149-158

[3] Valavanis, K. (ed). Advances in unmanned aerial vehicles: state of the art and the road to autonomy. Series Intelligent Systems, Control and Automation: Science and Engineering, Springer, 2007.

[4] Pounds, P., Mahony, R., Corke, P. System Identification and Control of an Aerobot Drive System. IEEE Information, Decision and Control Conference 2007, pp. 154-159.

[5] B. Mills. Using MATLAB/Simulink with the JSBSim open source flight dynamics model. [Online]. Available: http://jsbsim.sourceforge.net/matlab.html

[6] Chen, Ch.L., T.P. Hsu Chang, J.R. A Novel Approach to Great Circle Sailings: The Great Circle Equation. Journal of Navigation 57, 311–320, 2004.
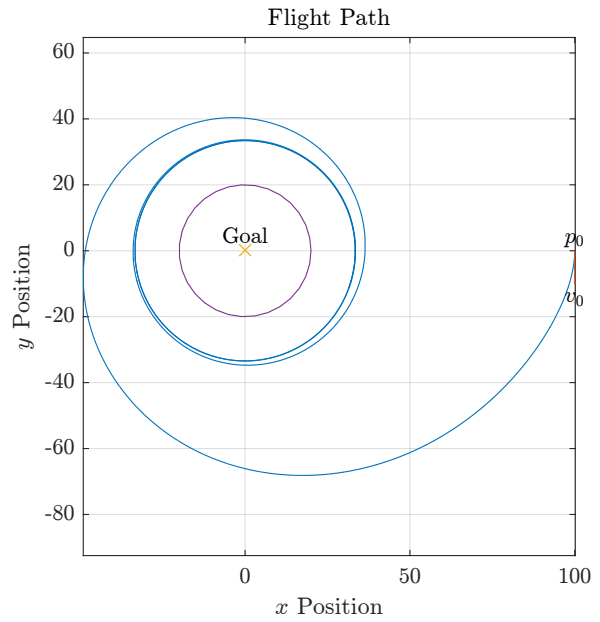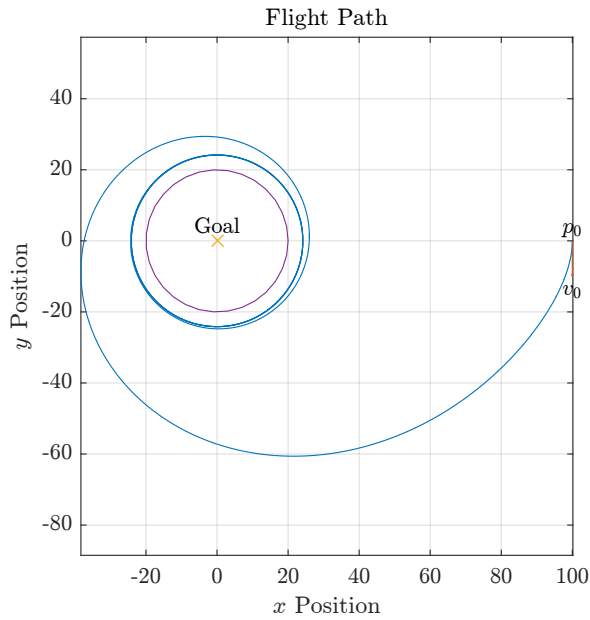
Figure 5: Flight path (blue) with initial parameters $\vec{p}_0 = 100\hat{i} + 0\hat{j}$, $v = 10$, $\theta_R = -\frac{\pi}{8}$, $\theta = -\frac{\pi}{2}$, $\varepsilon = 0.1$, and $r_0 = 20$.
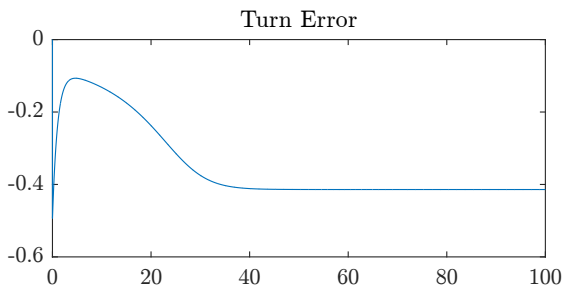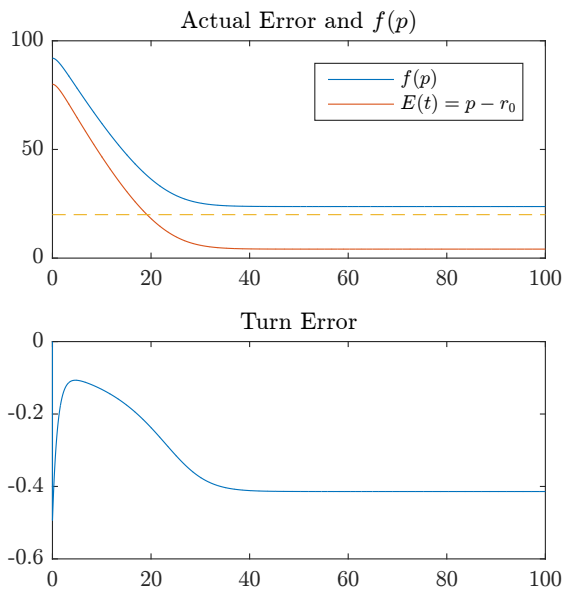

Figure 4: Flight path (blue) with initial parameters $\vec{p}_0 = 100\hat{i} + 0\hat{j}$, $v = 10$, $\theta_R = -\frac{\pi}{4}$, $\theta = -\frac{\pi}{2}$, $\varepsilon = 0.1$, and $r_0 = 20$.
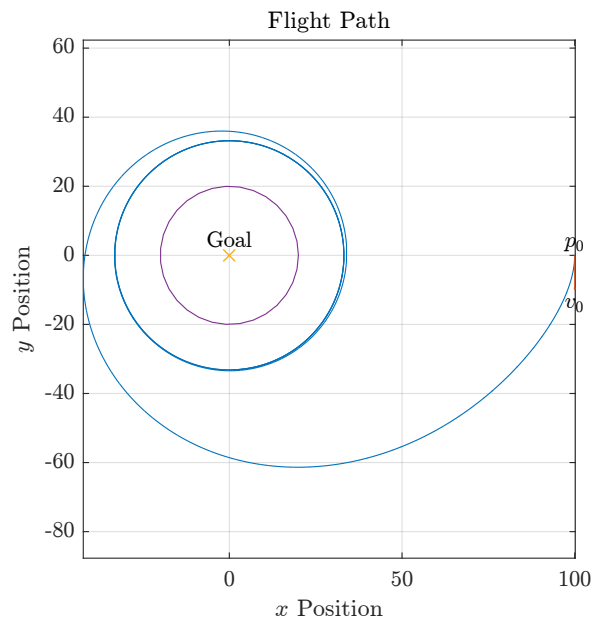

Figure 6: Flight path (blue) with initial parameters $\vec{p}_0 = 100\hat{i} + 0\hat{j}$, $v = 10$, $\theta_R = -\frac{\pi}{16}$, $\theta = -\frac{\pi}{2}$, $\varepsilon = 0.1$, and $r_0 = 20$.