AUTOMATIC SYNTHESIS OF A WIRE ANTENNA

AUTOMATIC SYNTHESIS OF A WIRE ANTENNA

NUMERICAL ELECTROMAGNETICS CODE

- **Maxwell's equations govern the electromagnetic waves generated and received by antennas**
- ***Numerical Electromagnetics Code* (NEC) is a method-of-moments (MoM) simulator for wire antennas**
- **NEC was developed at Lawrence Livermore National Laboratory**
- **NEC simulator is very general, reasonably fast, widely used, reasonably accurate, works from text input, produces text output**
- **Source code (in FORTRAN) is available**

AUTOMATIC SYNTHESIS OF A WIRE ANTENNA

ILLUSTRATIVE PROBLEM

- **This illustrative problem has previously been solved by both conventional antenna design techniques and the genetic algorithm (GA) operating on fixed-length character strings (Altshuler and Linden 1999)**

AUTOMATIC SYNTHESIS OF A WIRE ANTENNA

ILLUSTRATIVE PROBLEM

- **Synthesize the design of a planar symmetric antenna composed of wires of a half millimeter radius that**
  - **has maximum gain in a preferred direction (specifically, along the positive X-axis) over a range of frequencies from 424 MHz to 440 MHz,**
  - **has reasonable value (specifically $\leq$ 3.0) for voltage standing wave ratio (VSWR) when the antenna is fed by a transmission line whose characteristic impedance, Z_0, is 50 Ω,**
  - **fits into a bounding rectangle whose height is 0.4 meters and whose width is 2.65 meters, and**

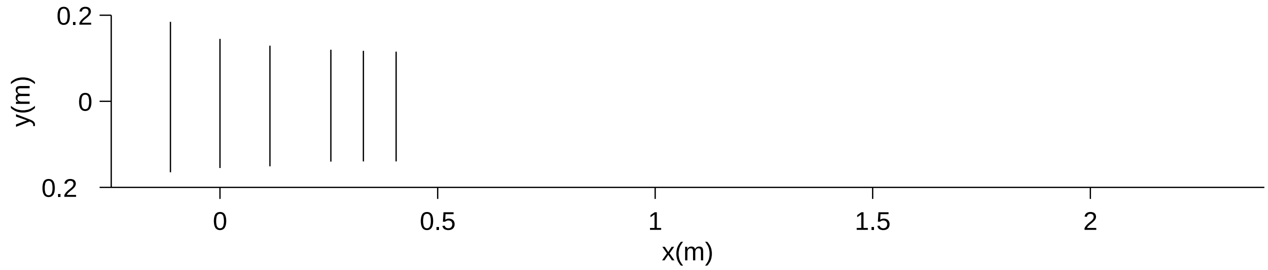
AUTOMATIC SYNTHESIS OF A WIRE ANTENNA

ILLUSTRATIVE PROBLEM

- **is excited by a single voltage source. One end of the transmission line is connected to the source and the other end is connected (at the origin of a coordinate system) to the antenna's driven element. The lower left corner of the bounding rectangle is positioned at (-250, -200) and the upper right corner is at (2400, 200).**

SOLUTION TO ILLUSTRATIVE PROBLEM

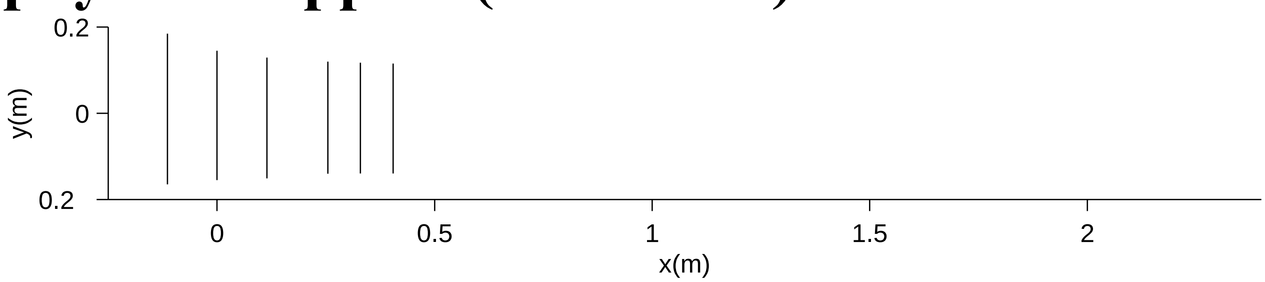
- **Requirements can be satisfied by a Yagi-Uda antenna**



- **A Yagi-Uda antenna is a planar symmetric wire antenna consisting of a number of parallel linear elements. The Yagi-Uda antenna is widely used for home TV antennas. The *driven element* is the vertical linear element positioned along the Y-axis. The midpoint of the driven point is connected to a transmission line (at the origin). The other elements of the antenna are not connected to the transmission line. Instead, their currents are induced parasitically by mutual coupling. All of the antenna's elements are symmetric (about the X-axis).**

SOLUTION TO ILLUSTRATIVE PROBLEM — CONTINUED

- **The elements to the left of the driven element act as *reflectors*. Usually (but not necessarily) there is just one reflector. The (numerous) elements to the right of the driven element act as *directors*.**
- **The directors are typically spaced unequally, shorter than the driven element, and of different lengths. A well-designed Yagi-Uda antenna is an *endfire* antenna in that it directs most of its energy toward a point in the farfield of the antenna (along the positive X-axis here). The driven element, directors, and reflector(s) of a Yagi-Uda antenna are attached to a non-conducting physical support (not shown).**



USING GENETIC ALGORITHMS (GA)

- **When the genetic algorithm (GA) operating on fixed-length character strings was used on the problem, the following decisions were made by the human user prior to the start of the run:**
  - **(1) the number of reflectors,**
  - **(2) the number of directors,**
  - **(3) the fact that the driven element, the directors, and the reflector are all single straight wires,**
  - **(4) the fact that the driven element, the directors, and the reflector are all arranged in parallel,**
  - **(5) the fact that the energy source (via the transmission line) is connected only to single straight wire (the driven element) — that is, all the directors and reflectors are parasitically coupled**

USING GENETIC ALGORITHMS (GA)

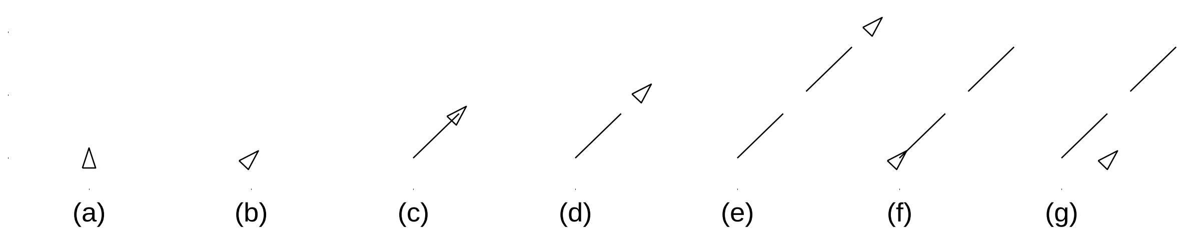
- **Characteristics (3), (4), and (5) are essential characteristics of the Yagi-Uda antenna, namely an antenna with multiple parallel parasitically coupled straight-line directors, a single parallel parasitically coupled straight-line reflector, and a straight-line driven element.**
- **Thus, the GA run assumed that the answer would be a Yagi-Uda antenna.**

AUTOMATIC SYNTHESIS OF A WIRE ANTENNA USING GP

- **We create a 2-dimensional drawing of the antenna**
- **The drawing is made using turtle geometry, including certain features of**
    - **Logo programming language**
    - **Lindenmayer systems**
- **The turtle may (or may not) deposit ink (metal) as it moves.**

EXAMPLE OF TURTLE FUNCTIONS

```
(PROGN3
  (TURN-RIGHT 0.125)
  (LANDMARK
    (REPEAT 2
      (PROGN2
        (DRAW 1.0 HALF-MM-WIRE)
        (DRAW 0.5 NO-WIRE)))
  (TRANSLATE-RIGHT 0.125 0.75))
```



PREPARATORY STEPS

PROGRAM ARCHITECTURE

- **Each individual program in the population has one result-producing branch.**
- **Automatically defined functions are not used.**

PREPARATORY STEPS

FUNCTION SET

- **The function set for the result-producing branch is**

F_{rpb} = {`TURN-RIGHT`, `DRAW`, `REPEAT`, `LANDMARK`, `TRANSLATE-RIGHT`, `PROGN2`, `PROGN3`, `PROGN4`}

PREPARATORY STEPS

- **One-argument `TURN-RIGHT` function changes the facing direction of the turtle by turning the turtle clockwise by the amount specified by its argument. The argument is a floating-point number between 0.0 and 1.0. The argument is multiplied by 2π so that it represents an angle in radians.**
- **Two-argument `DRAW` function moves the turtle in the direction that it is currently facing by an amount specified by its first argument (scaled between two bounds such as 16 to 200 millimeters). The turtle may or may not deposit a wire as it is moving, depending on its second argument — `HALF-MM-WIRE` or `NOWIRE`**

FUNCTION SET — CONTINUED

- **Two-argument `REPEAT` function causes its second argument subtree to be executed for the number of times specified by its first argument. The first argument is an integer modulo 100 (to avoid infinite loops).**

- **One-argument `LANDMARK` function is identical to brackets of Lindenmayer systems. It causes the turtle to execute its single argument. The turtle is then restored to the position and facing direction that it had at the start of the evaluation of the `LANDMARK` function.**

PREPARATORY STEPS

- **Two-argument `TRANSLATE-RIGHT` function does three things. First, this function temporarily turns the turtle clockwise by the amount specified by its first argument (in the same manner as the argument of the `TURN-RIGHT` function). Second, this function moves the turtle forward (in the direction in which the turtle has been temporarily turned) by the amount specified by its second argument (in the same manner as the `DRAW` function with `NOWIRE` as its second argument). Third, this function restores the turtle to its original facing direction.**

- **Connective functions `PROGN2`, `PROGN3`, `PROGN4` sequentially execute their 2, 3, and 4 arguments**

TERMINAL SET

- **A constrained syntactic structure is used to restrict certain terminals (`WIRE`, `NOWIRE`, $\Re_{real}$, and $\Re_{integer}$) to be certain arguments of certain functions**

  - **The terminal `HALF-MM-WIRE` denotes a half millimeter wire. The terminal `NOWIRE` represents the absence of a wire. A constrained syntactic structure restricts the location of these two terminals to the second argument of the `DRAW` function.**
  - **$\Re_{real}$ denotes floating-point numbers between 0.0 and 1.0. These terminals can appear only as the first argument of the `TURN-RIGHT` and `DRAW` functions.**
  - **$\Re_{integer}$ denotes integers between 0 and 99. These terminals can appear only as the first argument of a `REPEAT` function. These terminals can appear only as the first argument of the `REPEAT` function.**

TERMINAL SET

- **For all other parts of the program tree, the terminal set is simply**

T_{rpb} = {END}

  - **END terminal appears at all other leaves (endpoints) of the program tree.**

FITNESS MEASURE

- **Each antenna begins with a 7.5-millimeter straight piece of wire beginning at the origin and lying along the positive Y-axis. This stub becomes part of the antenna's driven element. The driven element is excited by a voltage source applied at (0,0). The turtle starts at position (0, 7.5) with a facing direction of north (i.e., along the positive Y-axis).**
- **Each program tree in the population is a composition of the above functions and terminals. The program tree is executed in the usual depth-first order of evaluation (from the left). The number, location, and shape of the antenna's elements are determined by the program tree. As the turtle moves, it may (or may not) deposit metal in the form of straight pieces of wire. Separate elements can be created by the `DRAW` function (when it is executed with the `NOWIRE` argument) and by the `TRANSLATE-RIGHT` function.**

FITNESS MEASURE — CONTINUED

- **Because the statement of this problem calls for the antenna to fit inside a specified area, after execution of the entire program tree, any metal that has been deposited outside the boundary of a clipping rectangle is deleted. The lower left corner of the clipping rectangle for this problem is positioned at (-250, 0) and the upper right corner is positioned at (2400, 200).**
- **Because the statement of this problem calls for a symmetric antenna, all metal deposited above the X-axis is, after clipping, duplicated by reflecting it across the X-axis. The result is the desired symmetric antenna lying inside the bounding rectangle (which is twice the height, but equal in width, to the clipping rectangle).**

FITNESS MEASURE — CONTINUED

- **The geometry of the antenna is specified by a data structure containing the coordinates of each wire and the radius of each wire (one half millimeter here).**

- **We embedded version 4 of the *Numerical Electromagnetics Code* antenna simulator in our genetic programming software. The input to NEC consists of text information that describes the geometry of the antenna, the number of segments into which each wire is partitioned for the method of moment calculation, information about the means of excitation, the output, information about the ground plane (or lack thereof), and various commands for controlling the simulation.**

FITNESS MEASURE — CONTINUED

• **Fitness is a linear combination of VSWR and gain.**

• **The VSWR is a measure of how much of the input energy from the source is reflected back down the transmission line from the antenna (rather than radiated by the antenna). The NEC code calculates the complex input impedance, *Z*, at the voltage source.**

$$VSWR = \frac{(1 + |R|)}{(1 - |R|)}$$

Here *R* is the reflection coefficient computed by

$$R = \frac{(Z - Z_0)}{(Z + Z_0)}$$

where Z_0 is the characteristic impedance of the transmission line feeding the antenna.

FITNESS MEASURE — CONTINUED

- **The value of VSWR ranges from 1 (representing no reflection — that is, all energy is radiated), to infinity (i.e., all energy reflected and there is no radiation). In this optimization the maximum value of the VSWR is limited to 2×10^8.**
- **The NEC simulator was instructed to compute the farfield radiation pattern at θ = 90 and ϕ = 0. θ is measured from the positive Z-axis to the X-Y-plane, while ϕ is measured from the positive X-axis to the positive Y-axis. The value for the antenna gain is the magnitude (in decibels relative to an isotropic radiator) of the farfield radiation pattern at θ = 90 and ϕ = 0.**

FITNESS MEASURE — CONTINUED

- **The accuracy of the simulation can be significantly affected by the number of segments into which each straight wire in the antenna is divided for purpose of simulation. There is no a priori way to compute the ideal number of segment. Contrary to intuition, the largest number of segments does not necessarily produce the most accurate simulation. Therefore, we performed the NEC simulation several ways for each antenna and used the worst outcome. Specifically, we performed the simulation using both 15 and 25 as the number of segments per wavelength. In addition, we performed the simulation using both 1 and 2 as the minimum number of segments per straight wire section. In addition, each of these four simulations was performed at 3 frequencies (424, 432, 440 MHz).**

FITNESS MEASURE — CONTINUED

- **The simulator calculates the current at the center of each segment. The value of the VSWR that we used in the fitness calculation is the maximum over all of the above 12 cases. The value of the gain, *G*, is the minimum value over all of the above 12 cases.**

- **Fitness is**

$$-G + C * VSWR$$

A smaller fitness is better. The constant *C* is 0.1 when *VSWR* ≤ 3.0 and is 10 when *VSWR* > 3.0. This is the same fitness measure used by Altshuler and Linden (1999), except that it has a slightly heavier penalty for poor *VSWR*. We increased the penalty because an antenna with poor *VSWR* can produce an artificially high gain (offsetting the VSWR in the overall fitness). *G* appears with a negative sign above because greater gain is more desirable.

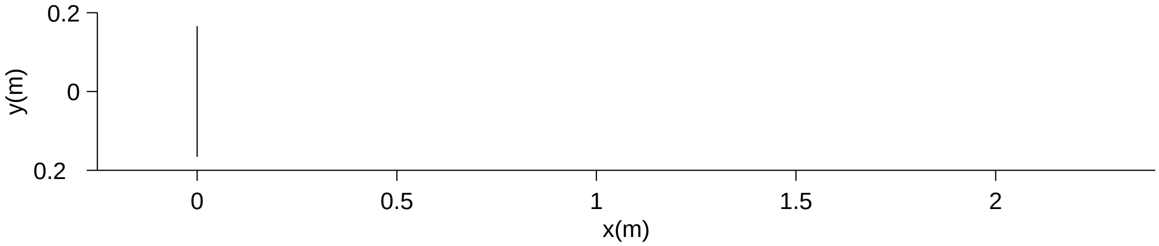
FITNESS MEASURE — CONTINUED

- **As required by the NEC simulator, each pair of intersecting wires is replaced with four new wires (each with an endpoint at the intersection point) prior to running the simulator. In addition, each antenna geometry is pre-checked by a various rules. If the length of each segment divided by its radius is not greater than 2.0 or the center of each segment is not at least 4 wire radii from the axis of every other wire, then the individual is either replaced (for generation 0) or assigned a high penalty value of fitness (10^8).**
- **If more than 10,000 functions are executed, execution of the individual program tree is terminated and the individual is assigned a high penalty value of fitness (10^8).**
- **Antennas that cannot be simulated receive a high penalty value of fitness (10^8).**

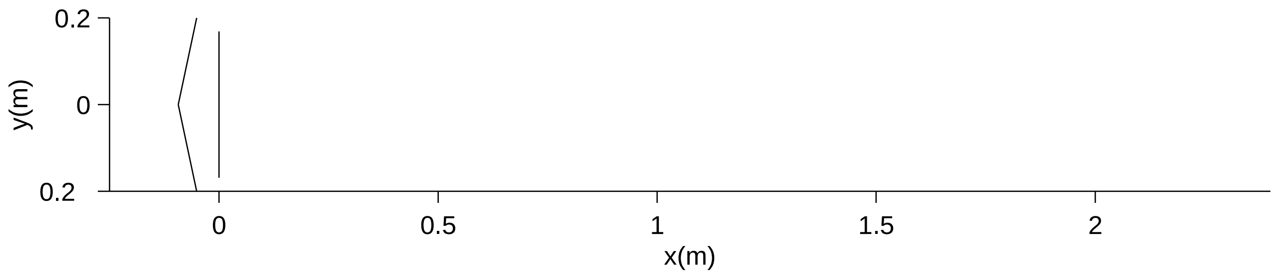
CONTROL PARAMETERS

The population size, *M*, was 500,000.

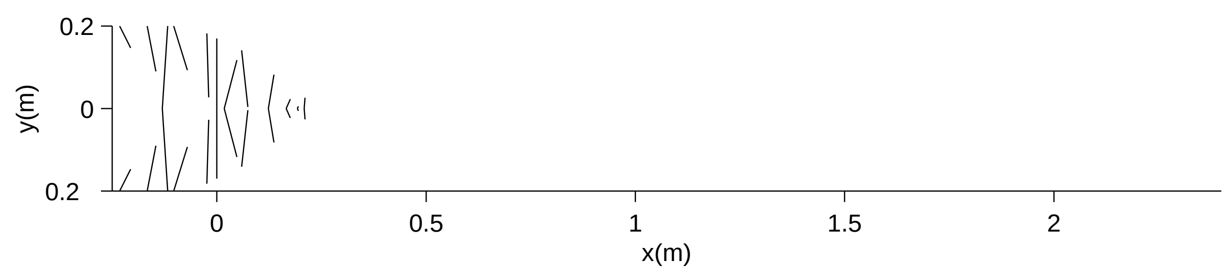
1ST EXAMPLE OF A BEST-OF-NODE ANTENNA FROM GENERATION 0 — FITNESS OF -2.03



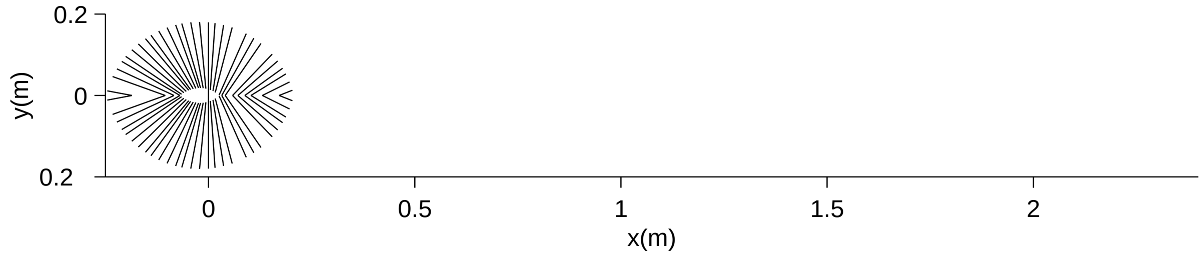
2ND EXAMPLE OF A BEST-OF-NODE ANTENNA FROM GENERATION 0 — FITNESS OF -3.82



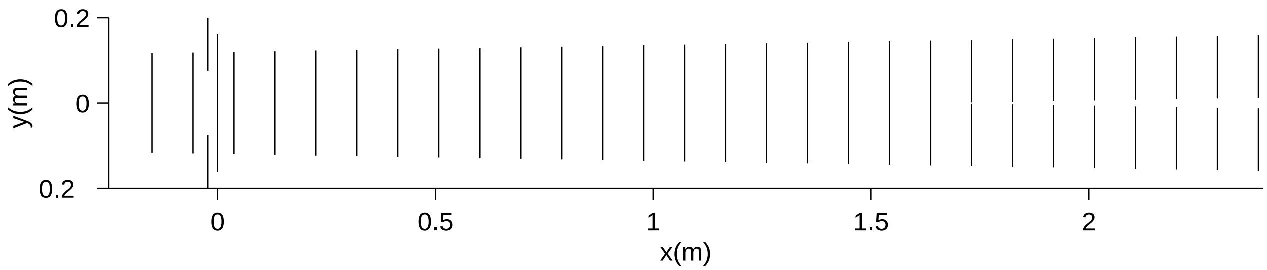
3RD EXAMPLE OF A BEST-OF-NODE ANTENNA FROM GENERATION 0 — FITNESS OF -4.43



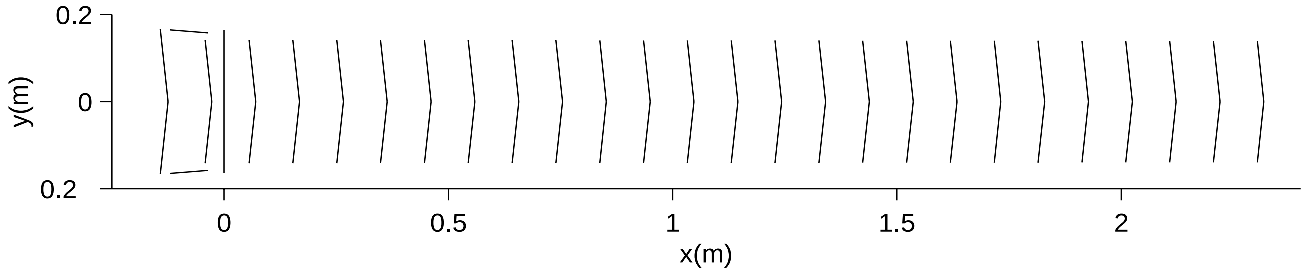
BEST-OF-GENERATION ANTENNA FROM GENERATION 2 — FITNESS OF -5.18



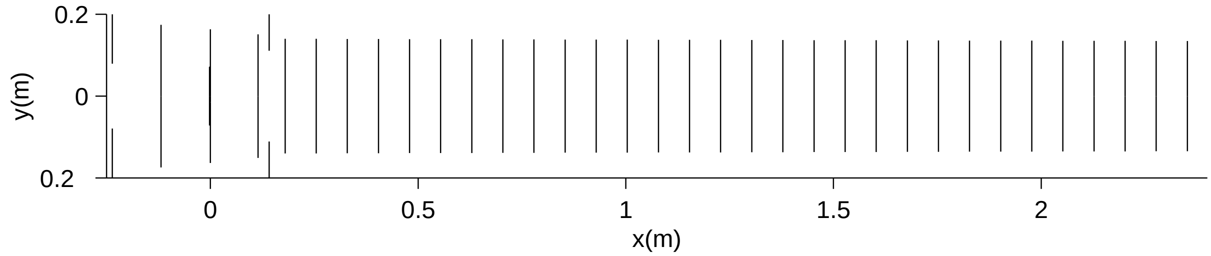
BEST-OF-GENERATION ANTENNA FROM GENERATION 9 — FITNESS OF -7.58



BEST-OF-GENERATION ANTENNA FROM GENERATION 47 — FITNESS OF -14.13



BEST-OF-RUN ANTENNA FROM GENERATION 90 — FITNESS OF-16.04



- **The GP run discovered**

  (1) the number of reflectors (one),

  (2) the number of directors,

  (3) the fact that the driven element, the directors, and the reflector are all single straight wires,

  (4) the fact that the driven element, the directors, and the reflector are all arranged in parallel,

  (5) the fact that the energy source (via the transmission line) is connected only to single straight wire (the driven element) — that is, all the directors and reflectors are parasitically coupled

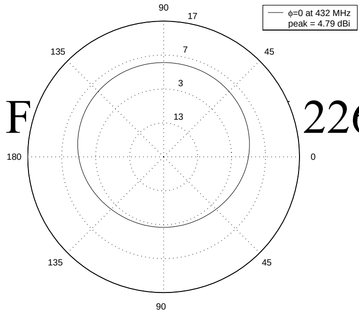
BEST-OF-RUN ANTENNA FROM GENERATION 90

- **Characteristics (3), (4), and (5) are essential characteristics of the Yagi-Uda antenna, namely an antenna with multiple parallel parasitically coupled straight-line directors, a single parallel parasitically coupled straight-line reflector, and a straight-line driven element.**

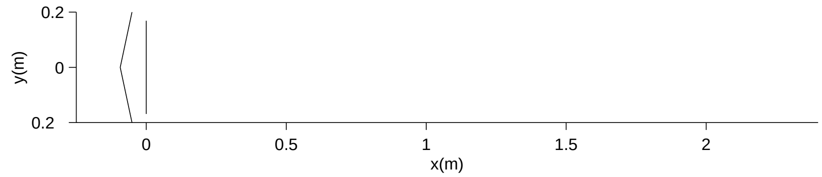
COMPARISON OF YAGI-UDA, ALTSHULER-LINDEN, AND GENETIC PROGRAMMING

- **The conventional Yagi-Uda antenna and the antennas created by the GA and GP are all approximately the same length (i.e., about 3.6 wavelengths along the X-axis)**

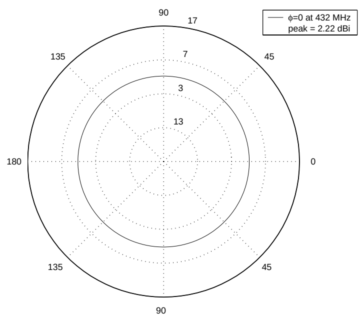
Frequency (MHz)	Yagi-Uda		Altshuler-Linden		GP	
	Gain (dBi)	VSWR	Gain (dBi)	VSWR	Gain (dBi)	VSWR
424	15.5	1.41	15.4	1.88	16.3	2.19
428	15.8	1.11	16.0	1.80	16.4	2.00
432	15.9	1.23	16.3	1.09	16.4	2.02
436	15.7	1.60	16.1	9.50	16.5	2.16
440	15.5	1.85	9.4	39.00	16.3	2.36



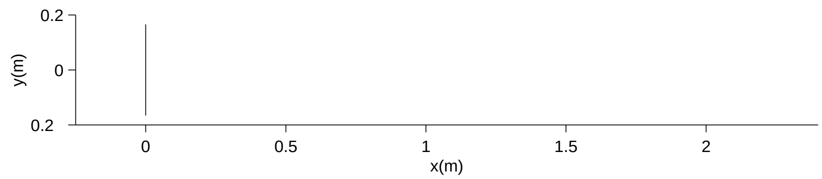
Second example of a best-of-node antenna from generation 0

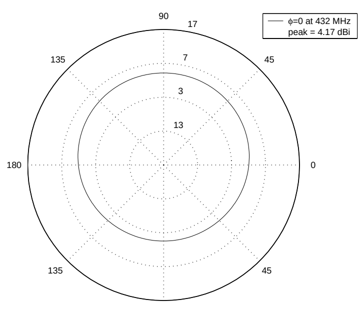


BEST-OF-RUN ANTENNA FROM GENERATION 90

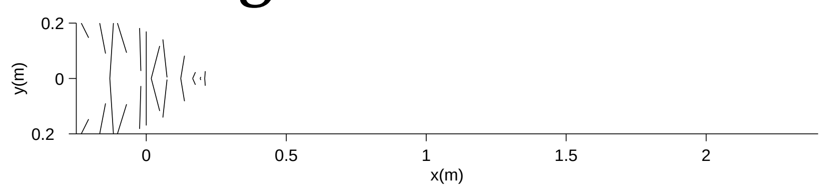


First example of a best-of-node antenna from generation 0

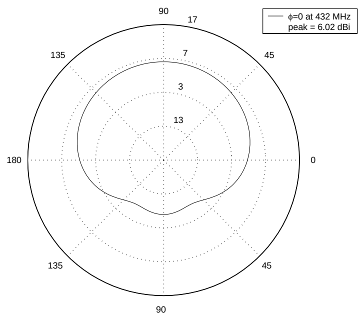




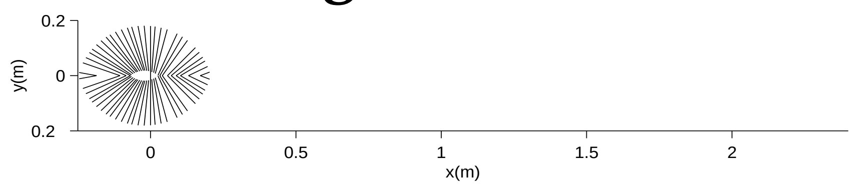
Third example of a best-of-node antenna from generation 0

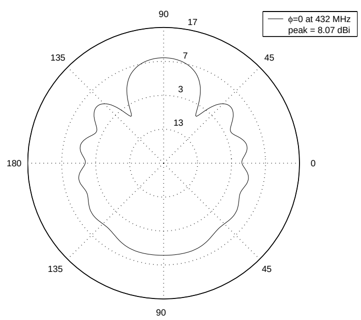


BEST-OF-RUN ANTENNA FROM GENERATION 90

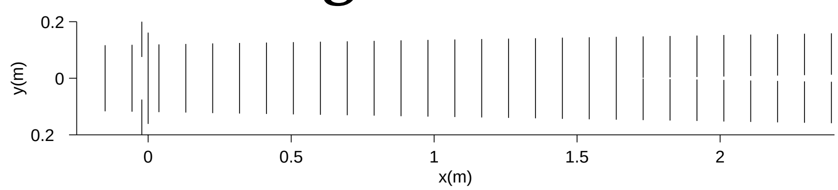


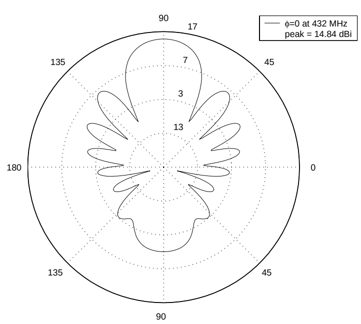
Best-of-generation antenna from generation 2



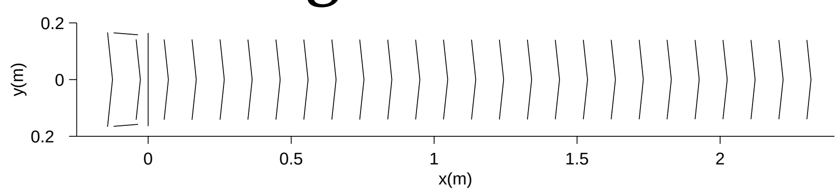


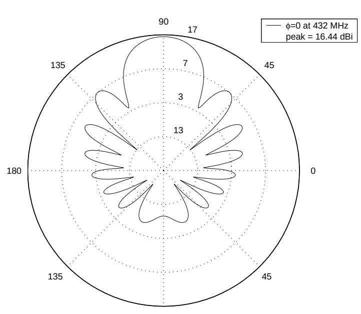
Best-of-generation antenna from generation 9





Best-of-generation antenna from generation 47





Best-of-run antenna from generation 90