SmartBeaconing

SmartBeaconing (tm) was developed by Tony Arnerich KD7TA and Steve Bragg KA9MVA on the HamHUD. It was originally used with the HamHUD but is now used by many devices such as the TinyTrak3, OpenTracker and APRS clients such Xastir.

At the inception of APRS, the equipment available had no provisions for deciding when to transmit other than through simple timing parameters. Tony and Steve came up with the idea of using the incoming GPS data to determine when it would be appropriate to send out a position update. Rather than simply beacon at a fixed rate, they created an algorithm that would beacon at a slow rate when the unit was sitting still, and beacon at a faster rate when the unit was moving. At this time they also came up with another concept to enhance the track laid down by a moving APRS unit. CornerPegging (tm) involves watching for a change in direction of travel. Again, when a change in direction of travel is detected, the unit is instructed to beacon. CornerPegging also looks at the velocity of the unit, and uses that to adjust the amount of deviation in the direction of travel required to cause a beacon.

Through the use of SmartBeaconing and CornerPegging one is able to leave behind a very detailed track of the route travelled, using a small number of packets. This technique is more channel efficient than simple time based beaconing, as it dynamically reduces the number of packets being sent as the unit velocity decreases.

SmartBeaconing allows the user to set the maximum beacon rate desired, associated with a maximum velocity, as well as a minimum beacon rate desired, associated with a minimum velocity. In comparison, timed beacons occur at the same rate whether the unit is moving or not. With SmartBeaconing, one can set the unit to beacon once every 3 minutes at a velocity of 60 mph or higher, but to only beacon once every 30 minutes at a velocity of 3 mph or less. This is more channel efficient than having the same unit beacon at a set rate of once every 3 minutes regardless of the velocity.

CornerPegging is the portion of the algorithm that increases the accuracy of the track laid down by the unit. While driving in a straight line, there is no need to keep reporting information that can be determined by dead reckoning. However, it is important to report deviations from that straight line track. Using time based beaconing, it is quite common to see tracks that appear to show the tracked unit driving straight through buildings, or across open fields, or bodies of water. With CornerPegging the unit watches for changes in direction of travel and determines when it is appropriate to send a beacon. Minimum change of direction settings control how much of a turn will trigger a beacon. CornerPegging also makes use of velocity information to tailor the amount of change of direction required to trigger a beacon. This allows the unit to send a beacon for a shallow turn at high velocity, while ignoring low speed changes of direction, such as lane changes.

Recommended SmartBeaconing settings for a vehicle, for normal driving:

PARAMETER	VALUE	UNITS	COMMENT
Fast Speed	60	mph	3 miles per posit
Fast Rate	180	sec	3 miles per posit
Slow Speed	5	mph	go to low rate at/below 5mph
Slow Rate	1800	sec	30 min posit interval while going slow/parked
Min Turn Time	15	sec	15 secs dead-time for corner posits
Min Turn Angle	30	deg	30 deg per posit at higher speeds
Turn Slope	255	unitless	requires a turn of more degrees at slower speeds

Turn Slope: Someone wrote recently on the TinyTrak list that 255 is an error that has been carried forward many places, and default should be 25. Stay tuned for any updates!

255 is not an error. It is a perfectly valid value for turn slope. Continue to the CornerPegging page for more detailed information.

CornerPegging

From APRSWiki

Jump to: navigation, search

CornerPegging(tm) is an integral part of a larger concept called SmartBeaconing(tm) developed on the HamHUD by Steve Bragg KA9MVA and Tony Arnerich KD7TA. When the term SmartBeaconing is used, it usually encompasses not only the time/speed based routine, but also the subroutine which watches for deviations in course.

As with SmartBeaconing, CornerPegging is a routine designed to reduce the network load produced by an APRS equipped mobile station while producing accurate representations of the path travelled.

The original time based position reporting system implemented in the APRS network missed out on using the available speed and course information contained within the NMEA strings from the GPS equipment attached. Older packet radio equipment upon which the APRS network was built was never designed to be location aware, and only had provisions to send packets out on a strictly time based schedule.

This worked fine for fixed location stations, but when trying to report the position of a moving target, time based position reports are not that great. To gain accuracy in the location of a station, you must increase the reporting rate. Your location accuracy becomes a larger and larger circle of ambiguity as time goes on. This circle of ambiguity can be estimated and reduced by using the last known direction of travel and speed of travel to give an estimated arc. This technique is known as dead reckoning. However, dead reckoning relies on the last reported information. In a time based position reporting system, the tracked station may have changed direction many times between position reports, which makes the dead reckoned position that much less accurate.

With CornerPegging, the routine watches for deviations from a course, and if the deviation exceeds a dynamic threshold, a new position is reported which contains new speed and heading information. The amount of deviation required is dependant upon a couple user defined variables. Minimum Turn Angle, and Threshold are used to calculate how much deviation from the current heading will be required to trigger an update to the last known position.

When playing with CornerPegging parameters, you have to remember that the minimum turn angle is not the smallest angle that it will report, but rather a value used as a base which is modified by the turn slope divided by speed.

Have a look at the table... The speed you are travelling is down the left side, and I have used a turn slope of 240 simply because it divides evenly. The last three columns simply show the how much you need to turn before a beacon is sent when the minimum turn angle is set to either 10, 20, or 30. The lower your velocity, the more deviation from your travelled path is required to trigger a position report.

SPEED	SLOPE	SLOPE/SPEED	MIN TURN=10	MIN TURN=20	MIN TURN=30
60	240	4	14	24	34
40	240	6	16	26	36
30	240	8	18	28	38
20	240	12	22	32	42
10	240	24	34	44	54
5	240	48	58	68	78

From the table you can see that the course deviation required to trigger a position report never gets down to the minimum turn angle, until your speed is greater than the turn slope value. By setting turn slope to a low value, velocity travelled has much less effect on the amount of deviation required. Setting turn slope to 1 will make the CornerPegging routine attempt to report every deviation greater than the minimum turn angle.

Just for fun let's look at that "correct" value from the TinyTrak list:

SPEED	SLOPE	SLOPE/SPEED	MIN TURN=10	MIN TURN=20	MIN TURN=30

60	25	0	10	20	30
40	25	0	10	20	30
30	25	0	10	20	30
20	25	0	10	20	30
10	25	2	12	22	32
5	25	5	15	25	35

25 as a turn slope is perfectly valid, but as you can see, it does not modify the minimum turn angle until you are travelling at less than 25. Once below that speed, it starts to affect the minimum turn required to trigger a position report.

SmartBeaconing was conceived to decrease the packet load on a network by automatically reducing the number of packets sent as velocity decreases. CornerPegging was added to trigger position reports on significant deviation from a course. When configured properly, the routines do just that.

Some people who don't fully understand the concepts, and who have possibly observed improperly configured units argue that SmartBeaconing causes an undue load on the network. This is simply an uneducated statement from people who are making judgements without the full facts.

APRS was built upon older packet equipment that only had the ability to set timed beacon rates. Setting a timed beacon for one beacon every 3 minutes yields 20 beacons per hour, or 480 beacons per day. This happens regardless of whether the station is moving or is parked all day.

With SmartBeaconing set to beacon at a fast rate of 180 seconds (3 minutes), and a slow rate of 1800 seconds (30 minutes) you will get different results. If the station sits still all day, you get 48 beacons per day. If the station drives to work for 1/2 an hour at or above the fast rate, you will get an additional 9 beacons for a total of 54 beacons. Let's add 20 turns to that 1/2 drive. With a turn slope of 255, minimum turn angle of 30, and mimimum turn time of 20, we should see an additional 20 packets. That brings our total to 74 beacons. Oh yeah, we have to drive home as well... add another 29 packets for a total of 103. Let's add 10 percent more packets due to stopping halfway around the corner, and waiting for traffic, etc. Now we have a total of 113 packets for the day. That's still less than 25% of the packets generated by the station beaconing every 3 minutes. If you drive below the fast rate, the beacon total will be less. Which station is causing more network load?

Yes, of course you can turn off the station that is beaconing every 3 minutes, and reduce it's load on the network. Unfortunately this doesn't always happen. This type of operation can be achieved through the addition of external circuitry such as a Charge Guard, APO3, or a simple ignition interruptor circuit.

With SmartBeaconing, you get automatic adjustment of your beaconing rate as you drive, automatic reduction to a minimum when you park, and highly accurate reproduction of the route travelled with minimal impact on the network. All of this with no user intervention required. Add dead reckoning to the APRS client you are using to view the network upon, and you get a very accurate picture of where the tracked station is. Dead reckoning of a SmartBeaconing/CornerPegging equipped station will keep you very close to the actual location of that station.

Dead reckoning of a time based station can lead you astray. If a time based station makes a turn 30 seconds after a beacon, the dead reckoned position continues along the old course for 2 and a half minutes. A CornerPegging station will report that course deviation when it happens and the dead reckoning station will adjust and show the new course immediately.