Understanding HPE in the VEMCO Positioning System (VPS)

V1.0, September 27, 2013

Frank Smith
Director of Data Analysis, VEMCO
frank.smith@vemco.com

Contents

Introduction

The VEMCO Positioning System (VPS) is an ultrasonic aquatic fine-scale positioning system used for tracking fish, other aquatic animals, or underwater objects. Transmitters are deployed on the animals or objects being tracked, and receivers are deployed at fixed stations in the area of interest to detect and record their transmissions.

VPS uses hyperbolic positioning, which is a technique based on measuring differences in transmission detection *times* at pairs of time-synchronized receivers, and converting these to *distance* differences using the signal propagation speed. Distance differences are commonly referred to as *range differences*, and we will use that term in this document.

VPS is primarily used with autonomous receivers like the VR2W-69KHz and VR2W-180KHz. Although a VR2W can record detections to millisecond resolution, since it is an autonomous device, its clock is not synchronized. In addition, a VR2W's clock can drift by up to ~4 seconds per day. To address this, VPS employs stationary transmitters at known locations as the basis of a method for measuring clock skew between neighbouring pairs of receivers. These transmitters are also used for calibrating the locations of receiver stations and measuring the effects of positioning error.

In the VEMCO Positioning System (VPS), the nature of error is very complex, and understanding error in VPS results has been a challenge for many customers. VPS does not provide calibrated accuracy estimates for calculated positions measured in terms of distance. Instead, it provides a relative, unitless estimate of how sensitive a calculated position is to errors in its inputs; this is referred to as horizontal position error (HPE). HPE is not comparable between different studies.

<u>Because VPS does not provide calibrated accuracy estimates, determining accuracy in a VPS requires the use of stationary transmitters at known locations for comparison with calculated positions.</u> There is no one number that can be calculated to characterize a VPS's accuracy, as accuracy depends on many factors that are variable over time and location.

The goal of this paper is to help customers understand HPE by

- Outlining VPS theory of operation
- Providing a concise definition of HPE
- Helping the user to determine a way of using HPE to characterize the error of a VPS dataset in terms of distance

This paper discusses

- The general concepts behind *hyperbolic positioning,* and the nature of its *error sensitivity*
- Fundamentals of VPS time synchronization
- Details on how VPS calculates positions and estimates error sensitivity
- Specific sources of error in VPS
- Application of HPE to characterize the error in a VPS dataset

Hyperbolic Positioning

VPS is based on hyperbolic positioning, also known as time difference of arrival (TDOA)[1]. This is the same technique that was used in various radio-navigation systems such as LORAN-C and Decca, and is currently used for locating mobile phones. This section illustrates the basics of hyperbolic positioning and its sensitivity to measurement error.

VPS operates in three dimensions (3D), but for simplicity this section will focus on two dimensional (2D) systems. The details of VPS operation in 3D will be presented in a later section. Also for simplicity, there is assumed to be no error unless explicitly discussed.

A hyperbolic positioning system consists of transmitters to be positioned, and time-synchronized receivers at fixed known locations. Transmissions are detected by receivers and logged with their detection times. The detection time (D) of a transmission from transmitter i at receiver a is determined by the transmission time (T) plus the propagation time (P) from i to a:

$$D_{ai} = T + P_{ai}$$

By convention, receivers will be identified by the letters a, b, and c, and transmitters will be identified by the letters i and j.

When two receivers detect a transmission, the difference in the detection times indicates how much sooner or later it was detected by one receiver than the other. We define *detection-time difference* (DD) as

$$\begin{aligned} DD_{abi} &= D_{bi} - D_{ai} \\ &= T + P_{bi} - T - P_{ai} \\ &= P_{bi} - P_{ai} \end{aligned}$$

Note that the order of the receivers is significant, i.e. $DD_{abi} = -DD_{bai}$.

DD_{abi} is the difference between the two detection times using receiver a as the reference:

- DD_{abi} positive: Detection time at b is apparently *after* detection time at a.
- DD_{abi} negative: Detection time at b is apparently *before* detection time at a.

Hyperbolic positioning systems typically assume *ideal propagation*: that a given transmission spreads spherically from its source at a known constant speed. With ideal propagation, time is proportional to distance, and therefore a difference in detection times converted to distance is the difference in distance, and indicates how much closer or farther the transmitter was to one receiver than the other. A distance difference is typically referred to as a *range*[2] *difference*:

$$RD_{abi} = velocity \times DD_{abi}$$

[1] http://en.wikipedia.org/wiki/Multilateration
[2] Range is a synonym for distance

A range difference (RD) combined with the positions of the two receivers is the fundamental building block of a hyperbolic positioning system. A hyperbola is "the set of points in a plane whose distances to two fixed points in the plane have a constant difference"[3]. In a hyperbolic positioning system, the two fixed points are the receivers, and the constant difference is a range difference.

In two dimensions (2D), a range difference and two receiver positions defines one branch of a hyperbola on which the transmitter was located at the time of the transmission. In other words, it is known that the transmitter was located somewhere on the branch of the hyperbola, but it's not known *where* it was. The branch of the hyperbola is commonly referred to as a hyperbolic *line of position* (LOP).

Transmissions that originate *anywhere* on the same hyperbolic LOP will result in the same range difference. In Figure 1, transmissions originating from positions T1 through T5 will result in the same range difference observed by Ra and Rb.

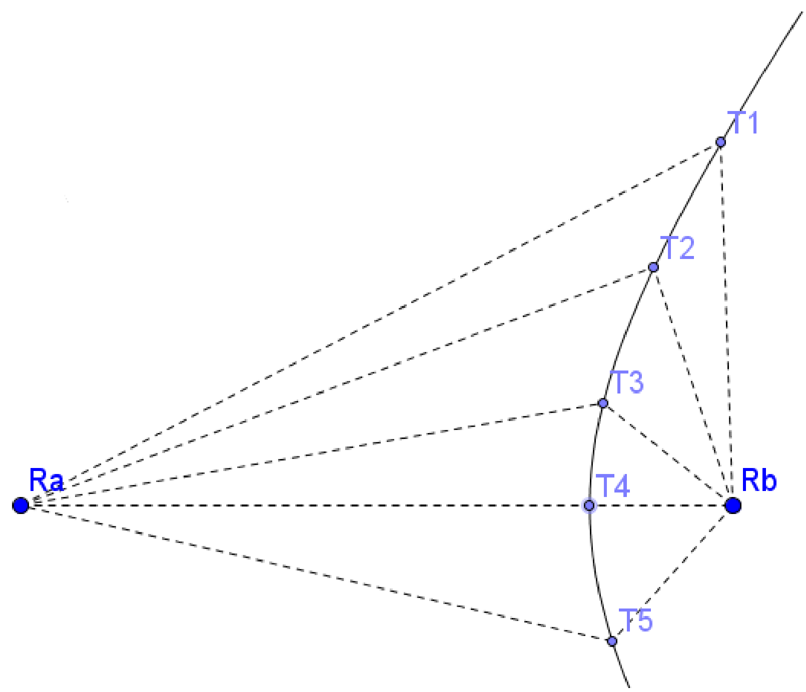

Figure 1

The curvature of the hyperbolic LOP depends on the range difference and the distance between the receivers. We define *normalized range difference* (NRD) as

$$NRD_{abi} = RD_{abi}/dist_{ab}$$

where $dist_{ab}$ is the distance between the receivers. Assuming no error, normalized range difference can range between +1.0 and -1.0. Figure 2 shows the hyperbolic LOPs corresponding to normalized range differences between +1.0 and -1.0 in steps of 0.1.

[3] http://dictionary.reference.com/browse/hyperbola

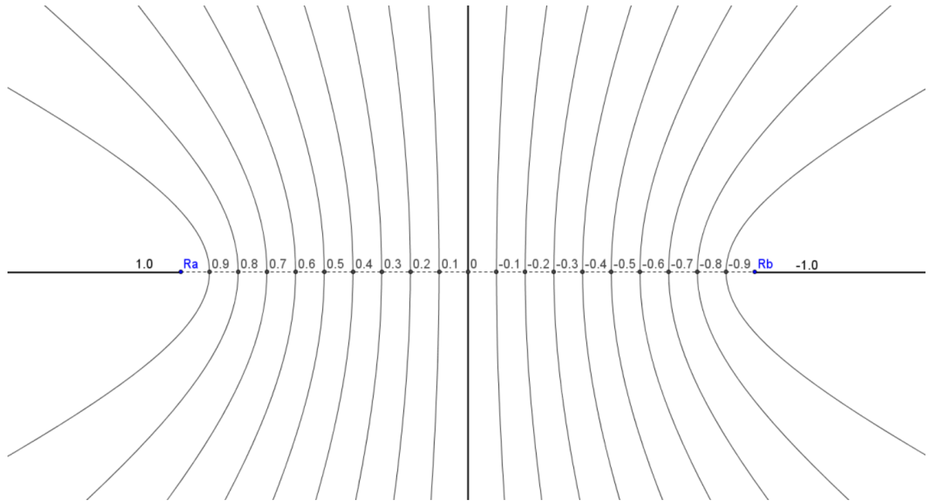

Figure 2

A normalized range difference of 0 means that the transmitter was the same distance from both receivers, and the hyperbolic LOP is a straight line perpendicular to the line that goes through the receivers. As the absolute value of the normalized range difference increases in either direction from 0, the curvature of the hyperbolic LOP increases. Finally, as the absolute value of the normalized range difference approaches 1 at the two receivers, the hyperbolic LOPs approach folded straight lines. Note the relatively large changes from 0.9 to 1.0 and -0.9 to -1.0.

With two receivers and a range difference, the location of a transmitter can be narrowed down to a hyperbolic LOP. To determine the *position* of the transmitter requires an additional receiver.

In Figure 3, a third receiver Rc has been added which potentially provides another 2 range differences. For calculating a position, only one of these is necessary, as the other is redundant and carries no additional information[4], so for the following we will only illustrate one additional range difference.

If a transmitter at point Ti transmits and is detected by the three receivers, it will result in $NRD_{abi} = -0.2$, and $NRD_{aci} = -0.3$. These define two hyperbolic LOPs (solid line for NRD_{abi}, and dashed line for NRD_{aci}), and the intersection of these is the location of the transmitter.

[4] $DD_{bci} = D_{ci} - D_{bi} = DD_{aci} - DD_{abi}$

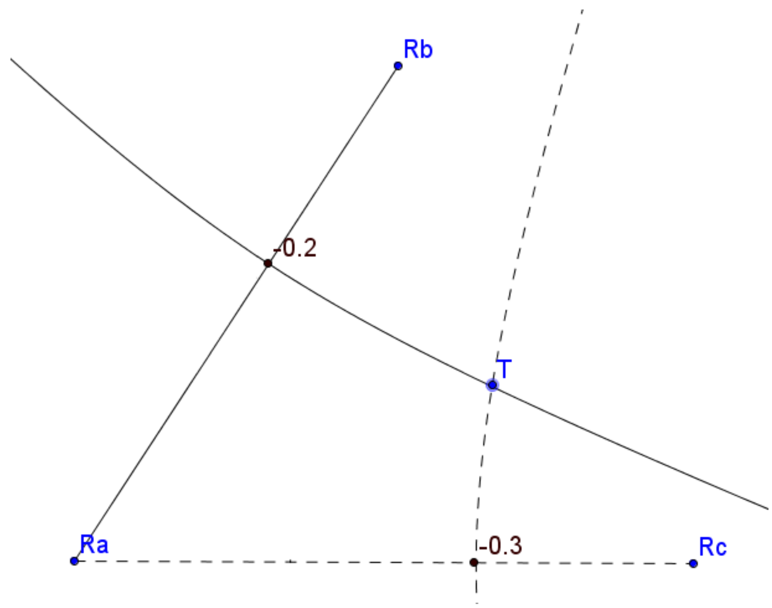

Figure 3

Because of measurement error in the calculated range differences (e.g. propagation speed measurement error, non-ideal propagation, detection time measurement error), the transmitter was likely not *exactly* on either of these two hyperbolic LOPs. Let us assume that each normalized range difference has potential error of ± 0.03.

In Figure 4, the center hyperbolic LOPs are as in Figure 3, and the intersection of these is the location of the transmitter assuming no error. The hyperbolic LOPs on either side of these represent a normalized margin of error of ± 0.03. Based on this error assumption, the transmitter can be assumed to be in the green region, bordered by the two outer pairs of hyperbolic LOPs. Note that the colour green is used to indicate that this example represents *relatively low* error sensitivity.

For simplicity, it is assumed that there is no error in receiver positions.

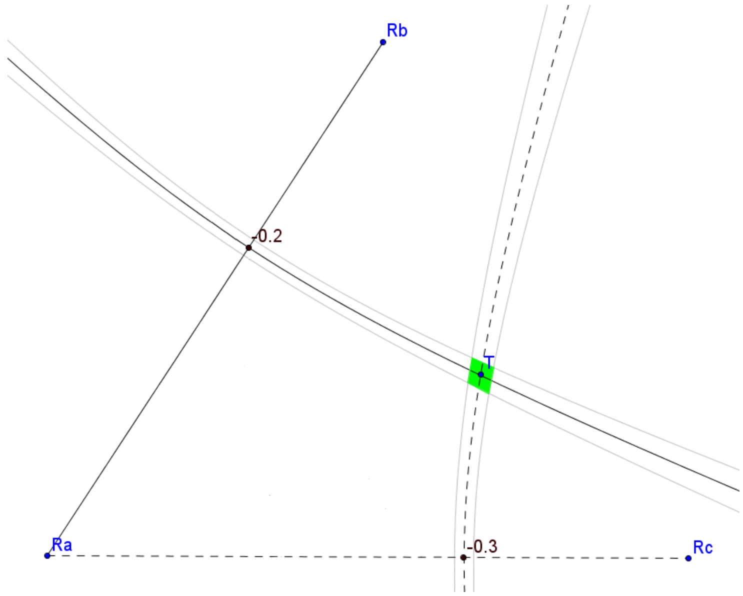

Figure 4

Figure 5 shows a case with relatively high error sensitivity, where $NRD_{abi} = 9.3$, and $NRD_{aci} = 6.0$.

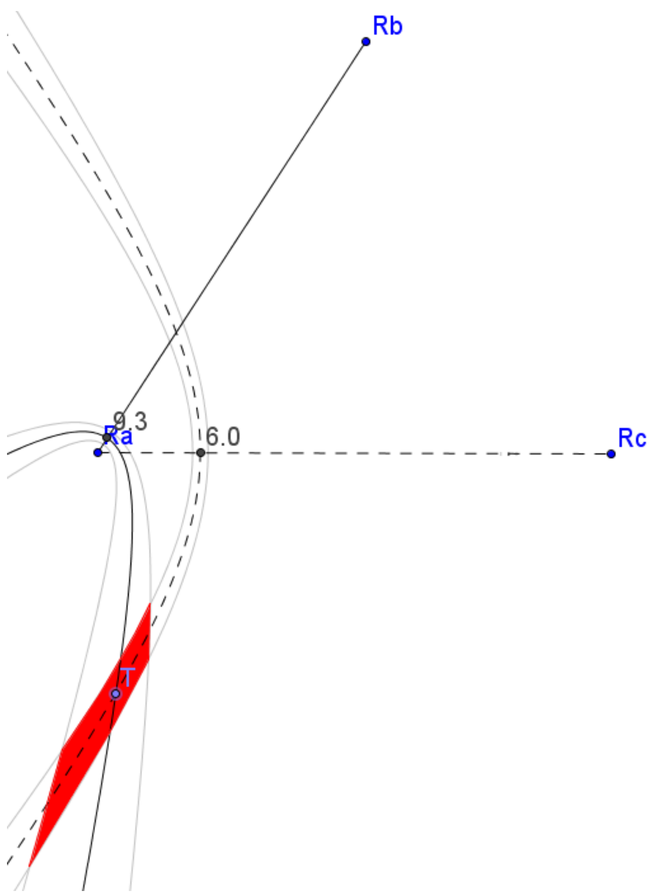

Figure 5

Again, the intersection of the two center hyperbolic LOPs is the position of the transmitter assuming there was no measurement error. The hyperbolic LOPs on either side of these represent a normalized margin of error of ± 0.03. Based on this error assumption, the transmitter can be assumed to be in the red region, bordered by the two outer pairs of hyperbolic LOPs. Note that the colour red is used to indicate that this example represents *relatively high* error sensitivity, because this area is relatively large and elongated.

These two examples were specifically chosen to illustrate that depending on where the transmitter was with respect to the receivers (i.e. geometry), the effect of range difference error can be magnified considerably. This effect is similar to the concept of Dilution of Precision (DOP) in GPS (Enge & Misra, 2006).

In Figure 6, hyperbolic LOPs from -0.9 to 0.9 in increments of 0.1 are shown (as in Figure 2). However, instead of showing hyperbolic LOPs -1.0 and 1.0, LOPs -0.99 and +0.99 are shown to illustrate how the curvature of the LOPs increases quickly as the normalized range differences approach +1.0 and -1.0.

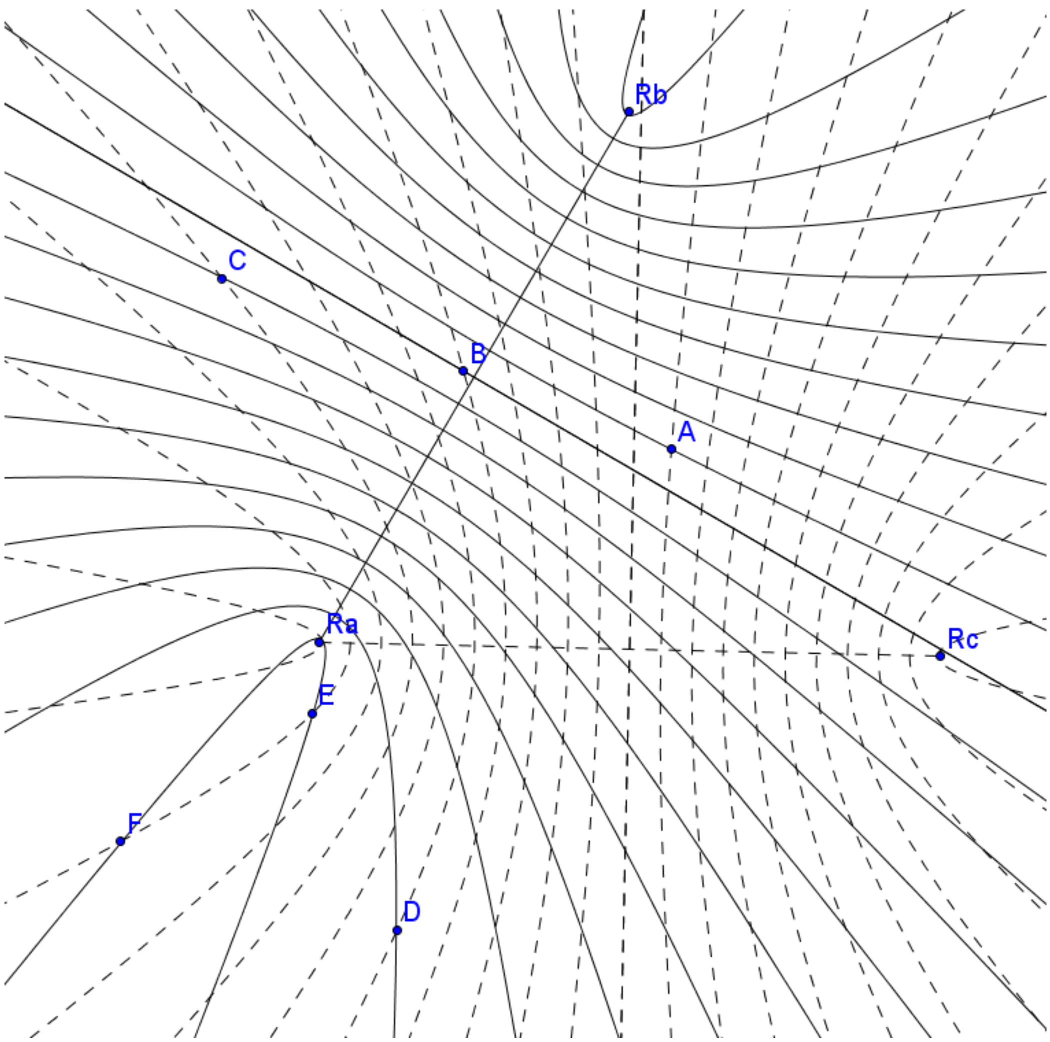

Figure 6

Figure 6 shows how the nature of the intersections of LOPs varies by location. At point A inside of the receiver triangle, the LOPs intersect at an angle of approximately 60° (based on the most acute angle), and at points B, C and D, increasingly farther from the center of the triangle, the angles are increasingly acute. In addition, points E and F are two intersection points of the *same pair* of LOPs, and consequently there is no unique solution.

In addition to the location of the transmitter relative to the receivers, the layout of the receivers has a significant effect on the nature of the hyperbolic LOPs. Note in Figure 7 how A and B are two intersection points of the same pair of LOPs, as are C and D.

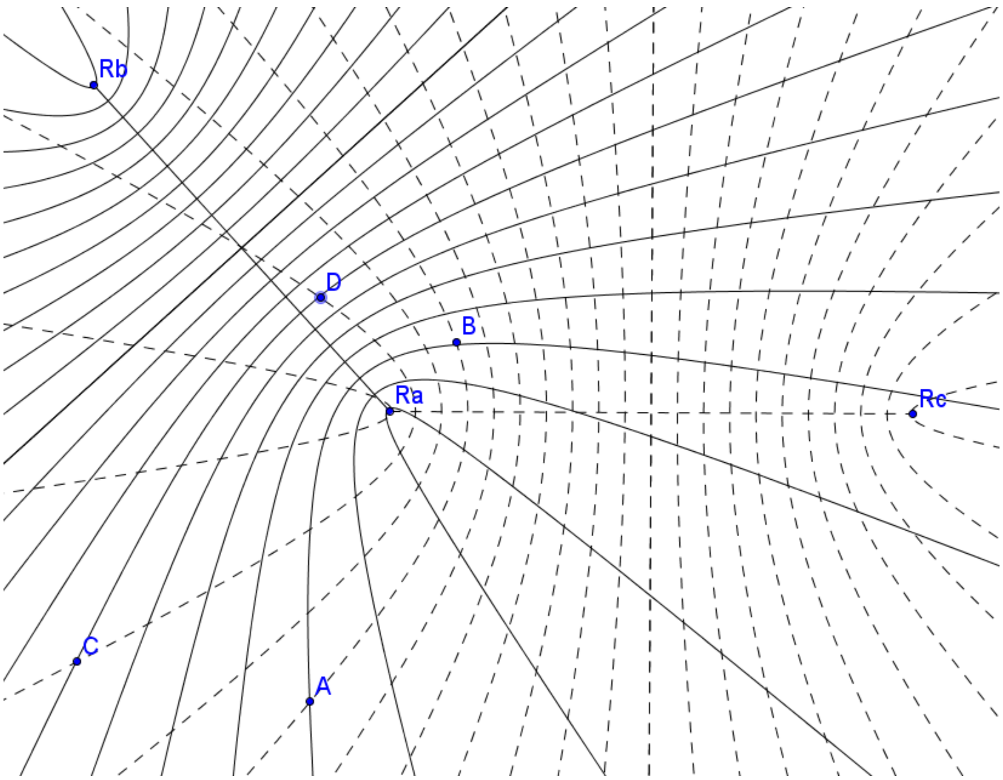

Figure 7

Figure 8 shows that as the normalized range difference approaches its extremes +1.0 and -1.0, changes have increasingly larger effects on the hyperbolic LOP. This causes positions calculated using extreme normalized range differences to be more sensitive to range difference error.

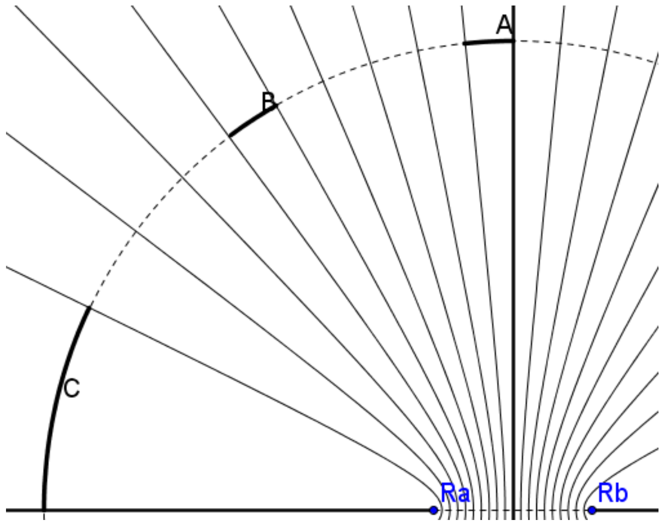

Figure 8

Figure 9 shows how the hyperbolic LOPs diverge with distance from the receiver triangle, which causes precision outside of the array to be significantly lower than inside of the array, where the LOPs are much closer together.

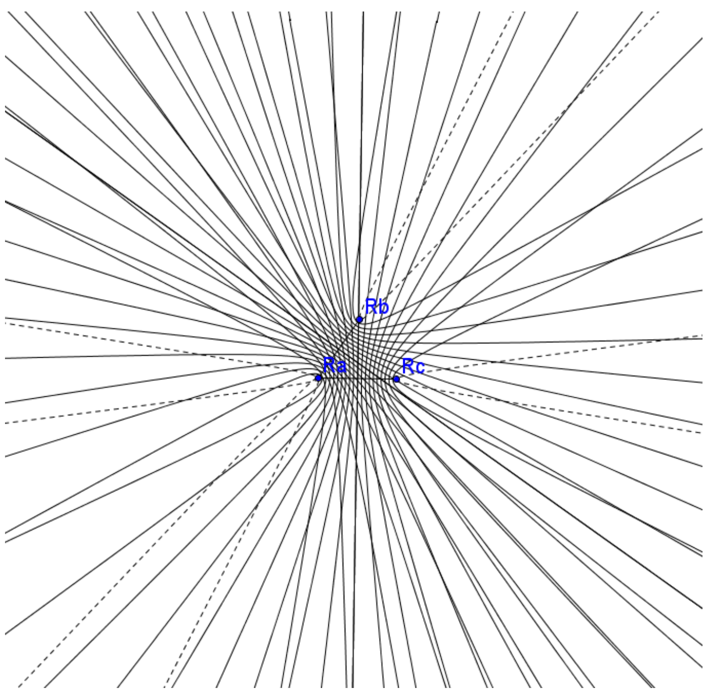

Figure 9

Figure 10 shows a summary of RD error sensitivity for a nearly equilateral receiver triangle, based on a software simulation. Green areas are those where RD error sensitivity is relatively low (as in Figure 4). Light green, yellow, orange, red and purple are regions where the RD error sensitivity is increasingly higher (as in Figure 5). Regions that are not colour-coded are those where unique positioning solutions are not possible (e.g. points E and F in Figure 6); these are called shadow zones. These plots are overlaid on satellite imagery. These plots are similar to those produced by PosSim, a simulation program used to predict positioning error resulting from detection time error, as described in (Klimley, et al., 2001).

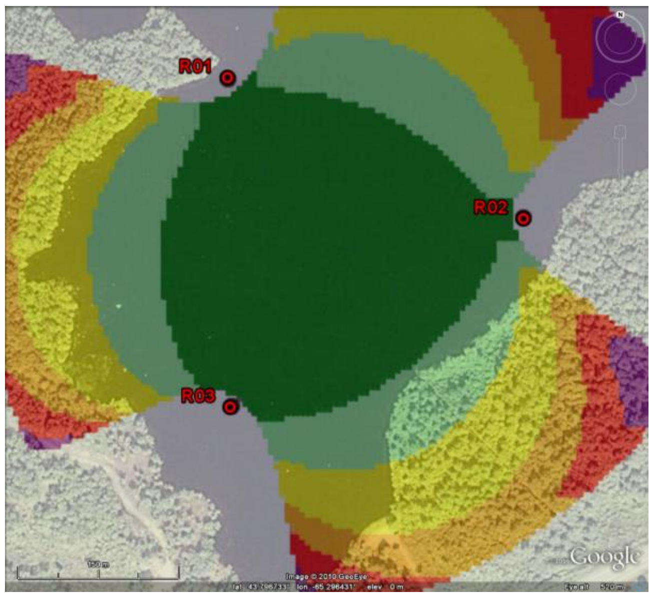

Figure 10

Receiver triangles with different shapes result in different RD error sensitivity profiles. Figure 11 shows a triangle with one angle approximately 100°. Note that the shadow zone extends into the triangle significantly more than in the previous example, and the region of lowest error sensitivity is smaller.

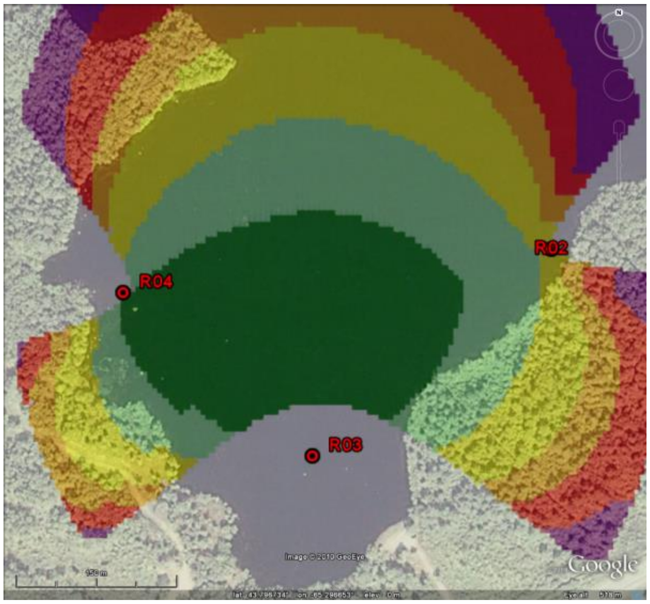

Figure 11

Figure 12 shows a triangle with one angle that is about 130°. Note that the shadow zone extends into the triangle even more than the previous example, and the region of lowest error sensitivity is much smaller. Comparing Figure 12 with Figure 7 will help illustrate the nature of this shadow zone.

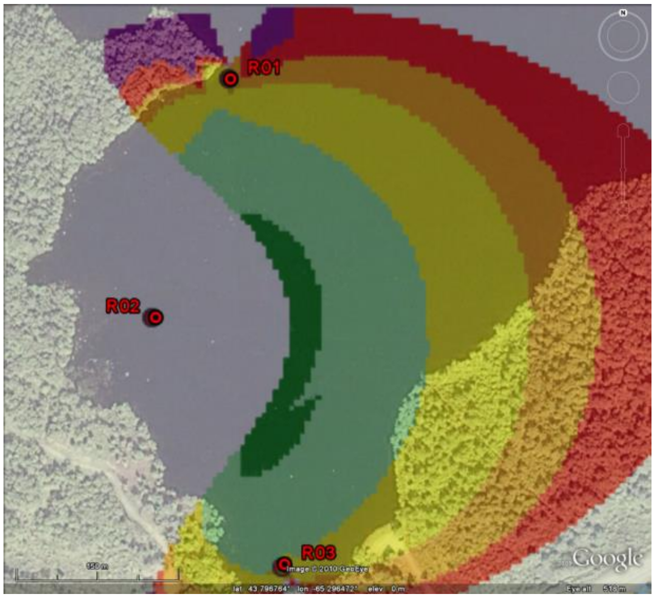

Figure 12

Propagation Model

VPS assumes *ideal propagation* as described in the section *Hyperbolic Positioning*, and does not attempt to model non-ideal propagation. Non-ideal propagation is a source of error.

In VPS, transmissions propagate through the water at ultrasonic frequencies (e.g. 69 KHz, 180 KHz). The speed of sound in water depends primarily on temperature, salinity and depth, and is not dependent on the frequency.

VPS uses the Coppens[5,6] equation (Coppens, 1981) for calculating the speed of sound in water. Its range of validity is 0 to 35 °C, salinity 0 to 45 parts per thousand, and depth 0 to 4000m. The formula is

$$c(D,S,t) = c(0,S,t) + (16.23 + 0.253t)D + (0.213 - 0.1t)D2 + [0.016 + 0.0002(S - 35)](S - 35)tD$$

$$c(0,S,t) = 1449.05 + 45.7t - 5.21t2 + 0.23t3 + (1.333 - 0.126t + 0.009t2)(S - 35)$$

where

- t = T/10 where T = temperature in degrees Celsius
- S = salinity in parts per thousand
- D = depth in kilometres

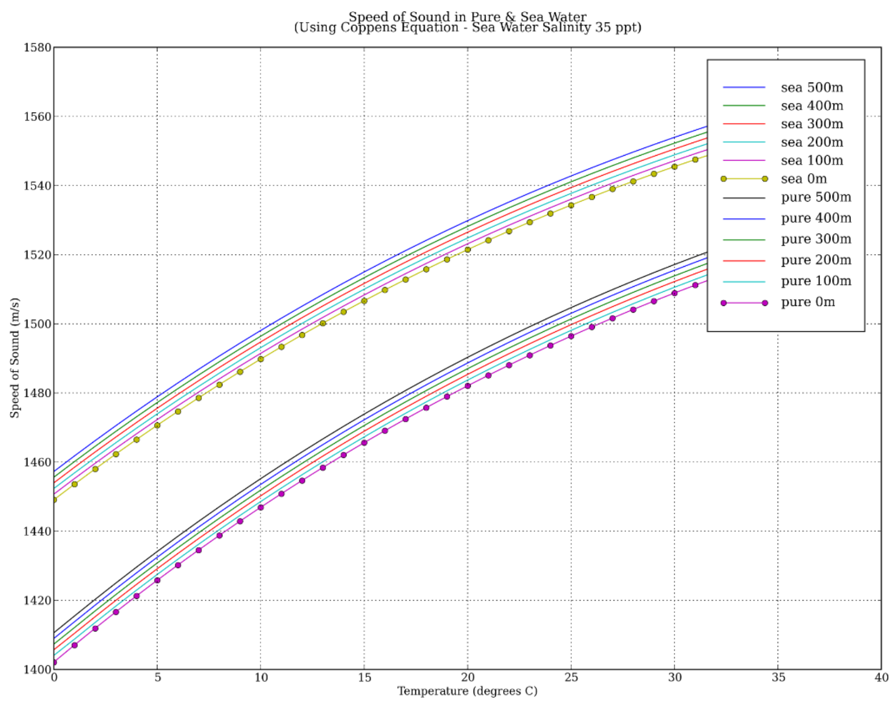

Figure 13

[5] http://www.tsuchiya2.org/soundspeed/coppens.htm
[6] http://www.comm-tec.com/Library/Technical_Papers/speedsw.pdf

Figure 13 shows a summary of the speed of sound in water between 0°C and 40°C as calculated by this equation, for pure (0 ppt salinity) and sea water (35 ppt salinity), and depths from 0 to 500m. As Figure 13 shows, depth is not significant under typical VPS conditions (depth < 100m), so the VPS propagation model is based on temperature and salinity only, and assumes depth = 0.

The VPS propagation model assumes that at any point in time, water temperature and salinity are uniform, but that water temperature and salinity can change over time.

Time Synchronization

Clock Skew and Drift

When a VR2W is initialized, its clock is typically synchronized with the clock of the PC running VUE to within approximately 1 second. Error in the PC's clock will result in error in the receiver's clock at initialization. VEMCO has observed many modes of error in PC clock time, resulting in clock error ranging from seconds to years. In addition, after initialization, a VR2Ws clock can drift by up to approximately 4 seconds per day. Clock *skew* is the difference between clocks at a point in time; clock *drift* is the rate of change in skew.

A typical VR2W's clock has a calibration tolerance of ±20 parts per million (PPM) at approximately 25°C, and as temperature increases or decreases from 25°C, the clock drift will change by approximately –0.04ppm/°C^2, e.g. it will slow down by 16ppm at 5°C, and 25ppm at 0°C.

Correlating Detections to Transmission

VPS is designed to be able to use detections from receivers that exhibit any amount of VR2W clock skew and drift. It does this by analyzing the differences in the detection times of stationary transmitters called synctags that are deployed at known locations. However, in order to calculate these differences, detections at different receivers must be correlated to their originating transmission, which is not trivial because the receiver clocks are not synchronized!

VPS correlates detections for synctags by taking advantage of the pseudo-random transmission delay sequence used by coded transmitters. Transmission delay is the time from the end of one transmission to the beginning of the next. Coded transmitters are configured to transmit with an average delay of avg seconds, but with specific delays that range between min and max seconds according to a pseudo-random sequence. A common configuration is $min = 500s$, $avg = 600s$ and $max = 700s$.

VPS analyses the detection time sequences of a synctag at pairs of receivers and matches the pseudo-random pattern. This results in detections being grouped by originating transmission.

Calculating Skew Between Receivers

Because of clock skew, a detection-time difference (DD) between two receivers a and b will consist not only of the difference in propagation times, but also the relative clock skew S_{ab} (i.e. the skew between the pair of receiver clocks). The relative clock skew at a single point in time can be expressed in terms of the skews S_a and S_b relative to real time:

$$S_{ab} = S_b - S_a$$

A transmission from i is detected by receivers a and b at times

$$D_{ai} = T + P_{ai} + S_{ai}$$
$$D_{bi} = T + P_{bi} + S_{bi}$$

where T is the transmission time, and S_{ai} and S_{bi} are the clock skews with respect to real time at the times that i is detected by a and b. Note that S_{ai} and S_{bi} are clock skews at two distinct times, but these times are close enough that it doesn't matter[7], so we will say that

$$S_{abi} = S_{bi} - S_{ai}$$

Therefore we can derive detection-time difference as

$$\begin{aligned} DD_{abi} &= D_{bi} - D_{ai} \\ &= T + P_{bi} + S_{bi} - T - P_{ai} - S_{ai} \\ &= P_{bi} - P_{ai} + S_{bi} - S_{ai} \\ &= PD_{abi} + S_{abi} \end{aligned}$$

In order for a DD_{abi} to be used to measure PD_{abi}, and therefore range difference, the clock skew component must be eliminated. Clock skew can be measured using

$$S_{abi} = DD_{abi} - PD_{abi}$$

If a transmitter i is detected by receivers a and b, DD_{abi} can be measured. If the *location* of transmitter i and the speed of sound are also known, PD_{abi} can be calculated, and therefore S_{abi}.

However, because clocks drift, the calculated skew S_{abi} is only valid at the time of the transmission i used to calculate it, and cannot be applied to transmissions at different times. For instance, 60 seconds after calculating a skew between a pair of receivers, their clocks may have drifted as much as 2.4ms. Synctags are typically programmed to transmit on average every 10 minutes. A method is required to estimate clock skew at arbitrary times, not only at the times of synctag detections.

Figure 14 shows detection-time differences DD_{abi} of a transmitter i at two receivers a and b. This is based on synthetic data, using a synctag with $min = 540s$, $avg = 600s$ and $max = 660s$, assuming no missed detections and no error.

[7] If the receivers are 1000m apart, and the speed of sound is 1500m/s, the difference in detection times would be 667ms, and assuming the two clocks differed by 40 ppm, the skew may have changed by up to 0.027 milliseconds

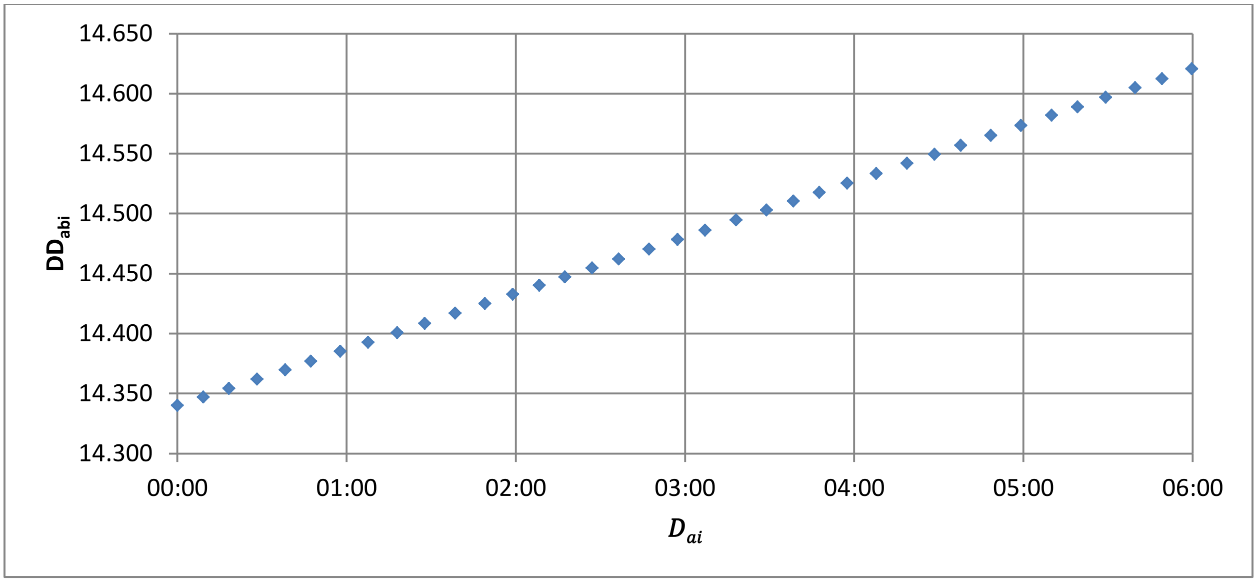

Figure 14

Figure 14 shows that over the period of 6 hours, receiver b's clock was ahead of receiver a's clock because the DD_{abi} values are all significantly[8] positive. It also shows that receiver b's clock drifted ahead by 281 milliseconds, showing that it was 13 parts per million (PPM) faster than receiver a's clock.

If a linear regression is fit to these data points, as shown in Figure 15, it can be used to estimate DD_{abi} at an arbitrary time.

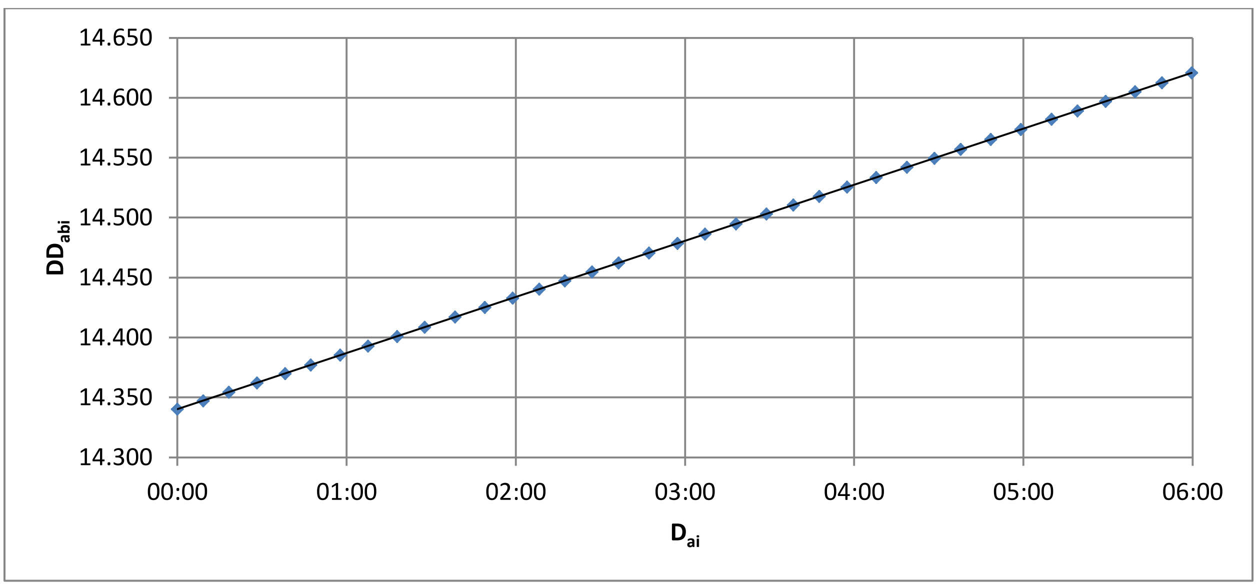

Figure 15

[8] Strictly speaking, a positive *clock skew* is necessary to say this; however the detection-time differences are large enough that clocks skews will also be positive, i.e. $S_{abi} = DD_{abi} - PD_{abi} > 0$.

Assuming the location of the transmitter is known, PD_{abi} can be calculated, and the following can be used to estimate skew at an arbitrary time, using DD_{abi} interpolated from the regression line:

$$S_{abi} = DD_{abi} - PD_{abi}$$

This receiver clock skew calculation can be used to convert a detection time D to a detection time D' expressed using another receiver's clock.

Positioning & Error Sensitivity

In order for VPS to calculate a position for a transmitter, a single transmission must be detected by at least 3 time-synchronized receivers at known locations. Position calculation is performed in two stages:

- Calculate a set of *basic positions* using receiver triangles chosen from the set of detecting receivers, one basic position per triangle
- Calculate a *synthesized position* by combining the basic positions. This is what is included in VPS results

Basic and synthesized positions also include estimates of error sensitivity, and for transmitters with known locations, horizontal distance between calculated position and known location. These are listed in Table 1. Shaded rows indicate the values that are included with VPS results.

Table 1

Value	Definition	Unit	Sync/Ref	Animal
HPE_b	Error sensitivity of a *basic position*	Unitless	Y	Y
HPE_s	Error sensitivity of a *synthesized position*	Unitless	Y	Y
HPE	Error sensitivity of a *synthesized position*	Unitless	Y	Y
HPE_m	Horizontal distance between a *synthesized position* and the known location of the transmitter (e.g. a GPS measurement)	Metres	Y	N

We recognize that the naming convention can be confusing because "HPE" could mean error sensitivity (HPE_b, HPE_s, or HPE), or measured error (HPE_m), depending on the context. Since these terms have been in common use for several years we feel it would be more confusing to change them at this point.

Basic Position Error Sensitivity (HPE_b)

The basics of 2D hyperbolic positioning have been outlined previously. VPS operates in *three* dimensions, so instead of range differences and 2D receiver positions defining hyperbolas on a plane, range differences and 3D receiver positions define *hyperboloids*[9] in 3D space.

The basic position calculation in VPS takes the following parameters, and returns a *basic position*:

- 3D positions for 3 detecting receivers
- Range differences
- Depth of the transmitter

Geometrically, it calculates the intersection point of a set of hyperboloids (defined by the receiver positions and range differences) and a plane (defined by the transmitter depth). The depth used for the transmitter is either a depth sensor reading accompanying the transmission, or an assumed depth when a depth sensor reading is not available. It calculates a 2D point on a horizontal 'depth' plane in 3D space.

[9] Specifically one sheet of a hyperboloid of two sheets

For more on geometrical interpretation of hyperbolic positioning please refer to (Fang, 1990). Note that although VPS operates in three dimensions, it does not *calculate* depth.

A basic position calculation can return 0, 1 or 2 solutions. Only those that return 1 solution (i.e. are unique) are used by VPS, as the error sensitivity calculation is only valid for basic positions with a single solution.

The error sensitivity (HPE_b) of a basic position is calculated by injecting error in the range differences and depth used to calculate the position, and calculating a set of 'errored' positions. The specific error values injected are based on dataset-specific factors such as temperature range, potential transmitter depth range, and distance from transmitter to receivers, and are intended to serve as a reasonable first-order approximation of actual range difference and depth error. The specific values used are not significant, as the goal of the sensitivity analysis is to produce relative unitless error sensitivity estimates, so over- or underestimating actual error is not a significant issue.

HPE_b is calculated as the largest horizontal distance between the basic position and the errored positions. This approach is similar to the "theoretical accuracy" described in (Klimley, et al., 2001, p. 435). Although HPE_b is *calculated* as a distance, it should not be *interpreted* as a distance because the error being injected is not calibrated.

If any of the errored positions does not return a unique position solution, HPE_b is undefined, as this means that the basic position was geographically close to a shadow zone and more likely to be unreliable. For simplicity, receiver position error is not considered.

Synthesized Position Error Sensitivity (HPE_s)

A *synthesized position* is a weighted average of the valid basic positions for the transmission (i.e. returned a unique solution and have HPE_b defined), where the weight for basic position i is

$$w_i = \frac{1}{HPEb_i^2}$$

There are two error sensitivity values for a synthesized position: an internal error sensitivity HPE_s, and the error sensitivity HPE provided in VPS results. The internal error sensitivity HPE_s of the synthesized position is

$$\sqrt{\frac{1}{\sum w_i}}$$

Note that since HPE_s is based on HPE_b, which is based on injecting an uncalibrated amount of error, it is not valid to interpret HPE_s in terms of distance.

Synthesized Position Error Sensitivity (HPE)

HPE is derived by applying knowledge of measured error from fixed locations to the sensitivity of the system. This takes the form of a linear transformation as follows:

$$HPE = A \cdot HPEs + B$$

The A and B values are the slope and y-intercept of a linear transformation applied to HPE_s in order to derive HPE as a "rough guide" error estimate measured in metres.

Interpretation of HPE as an error estimate in metres should only be used for informal purposes, e.g. casual inspection of results in Google Earth. In practice, due to its simplistic nature, it may significantly under- or over-estimate actual error, so this interpretation is not to be used for formal purposes. The derivation of these values is outlined in Appendix A: Rationalizing Sensitivities to Measured Error.

HPE should only be interpreted as a relative unitless estimate of *error sensitivity.*

Applying HPE

Different methods of analyzing animal position data will have different positional quality requirements. Because of this, there is usually a need to filter out positions that do not meet specific requirements, or at least of verifying that all available positions are of sufficient quality. HPE can serve as the basis of methods of auditing position quality.

Figure 16 shows the relationship between HPE and HPE_m for calculated positions for synctags in a VPS dataset, using only data included in standard VPS results. Each green dot corresponds to a calculated *synthesized position*.

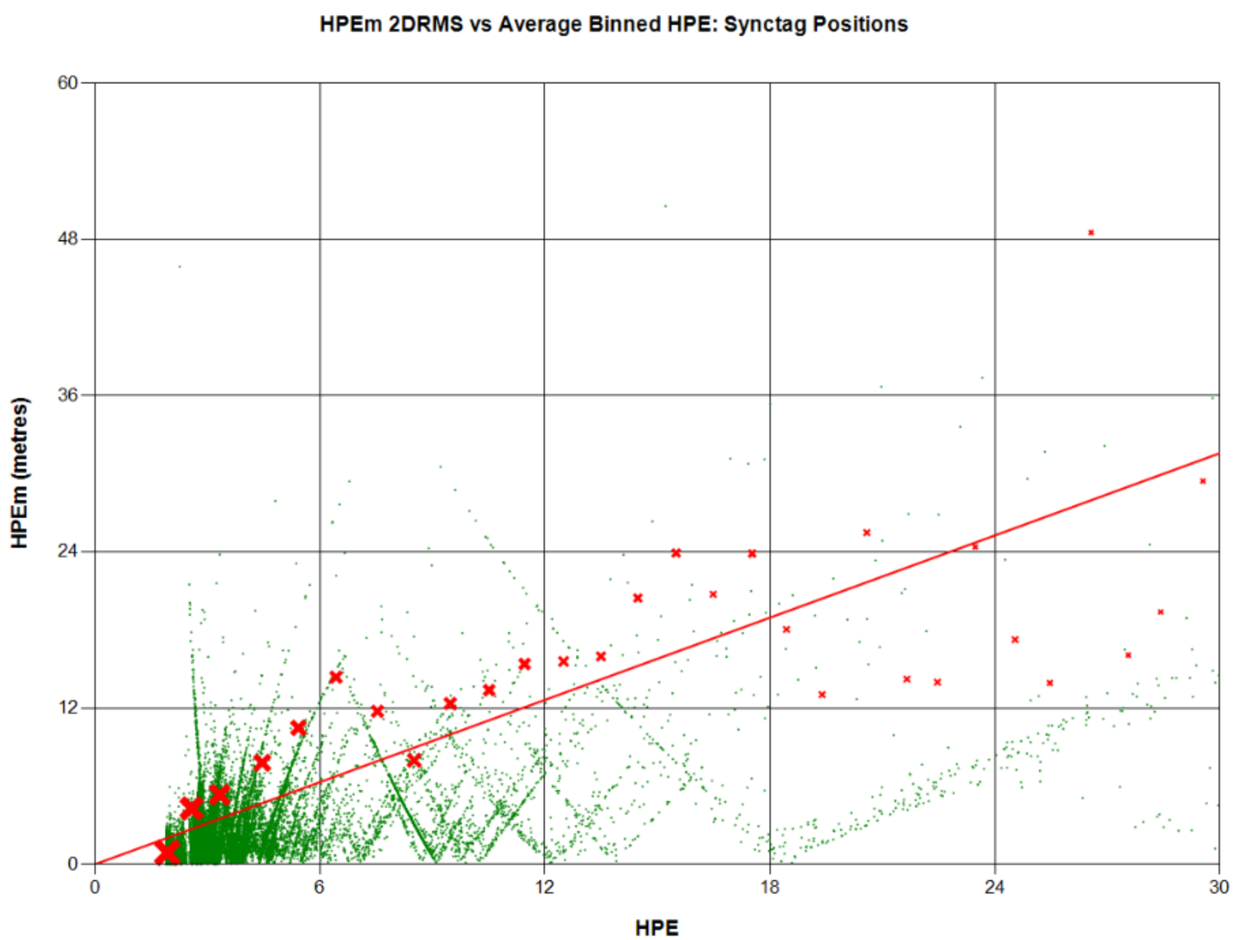

Figure 16

There are clearly visible V-shaped patterns in Figure 16. As HPE increases, they open with larger angles; as HPE decreases, they close. Each of these V-shaped patterns corresponds to the calculated positions for a specific synctag. The reason for the patterns is geometry, i.e. the location of the synctag relative to the detecting receivers. In areas of high error sensitivity, calculated positions can have high error, and if a calculated position has high error, it will appear to be transmitting from a location significantly different than where it was. Some calculated positions will appear to be transmitting from a location of lower error sensitivity, and will have a relatively lower HPE; some will appear to be transmitting from a location of higher error sensitivity, and will have a relatively higher HPE. Figure 17 shows a typical example of this effect. The orange dot is the actual transmitter position; the other dots are calculated positions colour-coded by HPE: green for lower values; red for higher values.

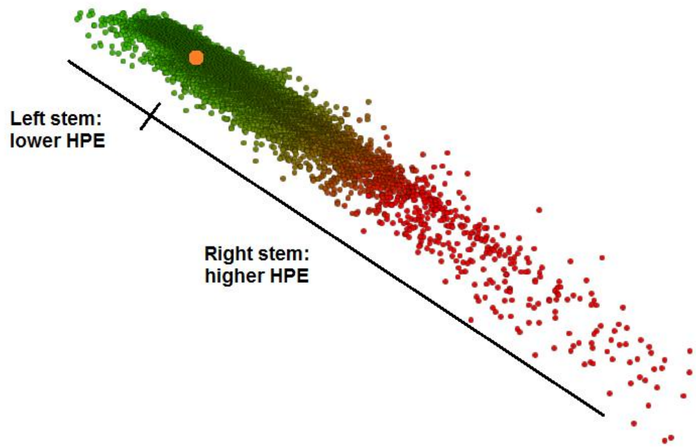

Figure 17

It is often stated by customers that there is no relationship between HPE and HPEm. This is because HPE is not intended to predict what error is equal to for a specific calculated position; it is intended to predict the relative precision or spread of calculated positions that have that value of HPE.

So rather than compare HPE with HPE_m directly, it is more useful to compare HPE with HPE_m statistically, for example by binning groups of calculated positions based on ranges of HPE, and for each bin calculating a statistic such as 2DRMS (twice the distance root mean square, (Enge & Misra, 2006, p. 215) or a percentile.

In Figure 16, the red crosses are 2DRMS values for bins of calculated positions (bin width = 1). Where there may be no obvious relationship by examining HPE with HPE_m directly (the individual green dots), the relationship is discernible based on 2DRMS values.

HPE can be visualized using animal tag positions. Figure 18 (generated using ArcGIS[10]) shows a 107-receiver VPS array covering approximately 10 square kilometers. The white dots are receiver stations, and coloured dots animal tag calculated positions. The colour represents HPE ranges: green < 10, light green < 20, yellow < 30, orange < 50, and red > 50.

[10] http://www.esri.com/software/arcgis

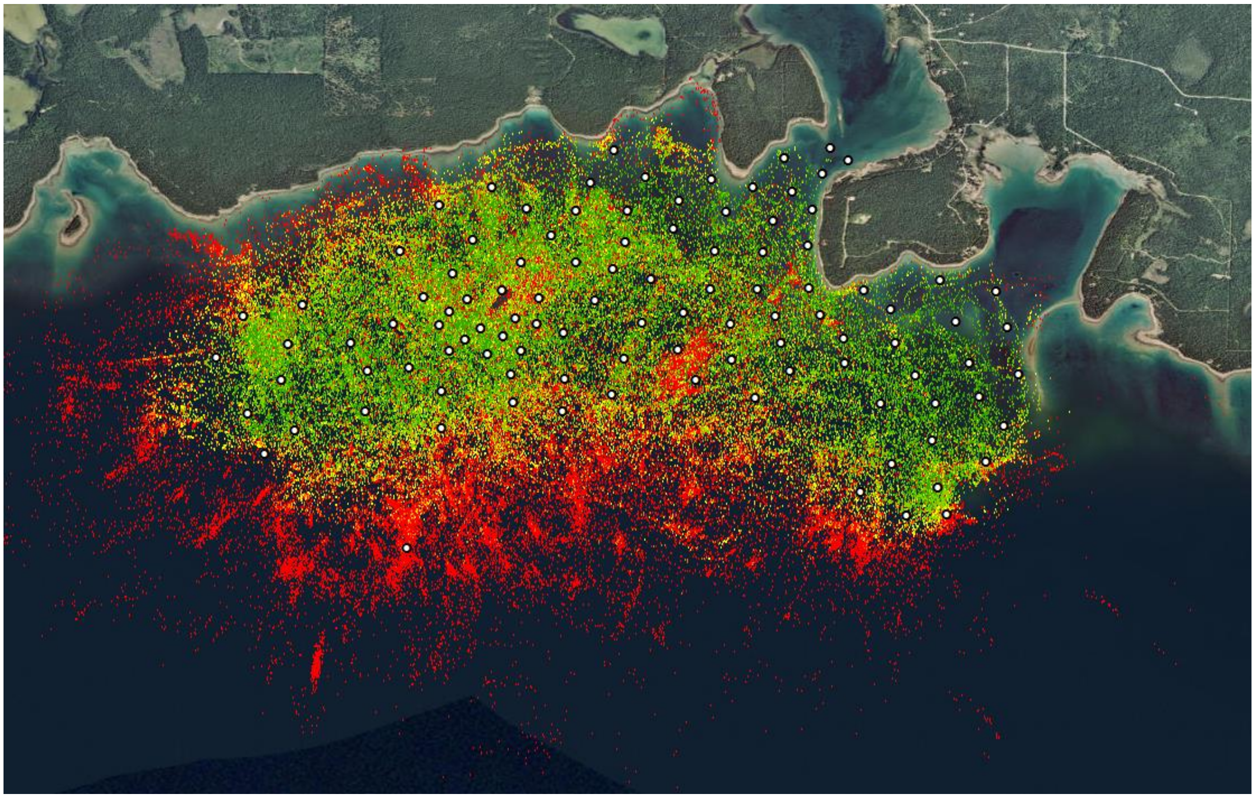

Figure 18

It is apparent that within the boundaries of the array most of the calculated positions are green or light green, meaning that HPE is < 20. There are, however, calculated positions within the array that have HPE ≥ 20; these are caused by transmitters that are inside of the full array, and are inside of at least one well-formed receiver triangle, but due to factors such as acoustics or collisions were not detected by one or more receivers required to complete a good triangle, leaving only poor triangles for positioning.

In general, calculated positions for transmitters inside of an array are more accurate than those for transmitters outside of an array. This has been observed in VPS and other systems (O'Dor, et al., 1998), (Espinoza, et al., 2011), (Piraino, 2011), (Biesinger, et al., 2013). This diagram illustrates how low HPE is correlated with positions calculated inside of an array, and therefore with higher accuracy.

The relationship between HPE and HPE_m is similar to the relationship between GPS horizontal dilution of precision (HDOP) (Enge & Misra, 2006) and GPS horizontal position error. (Enge & Misra, 2006, p. 210) contains an excellent description of how GPS HDOP is often misunderstood. Lower HDOP does not necessarily mean a lower position error, and higher HDOP does not necessarily mean a higher position error. Higher HDOP means that the *scatter* of positions is greater, but that obtaining a position with low error is as likely as obtaining one with high error. This is similar to how higher HPE means that the scatter of calculated positions is greater (i.e. lower precision).

Typical Applications

On its own, HPE is of limited use, but by analyzing the relationship between HPE and the calculated positions for transmitters with known locations, HPE can be characterized in terms of distance. Since HPE is calculated by VPS software in the same way for synctags and animal tags, HPE characterizations are expected to apply to animal tags as well. A similar observation was made in (Coates, Hovel, Butler, Klimley, & Morgan, 2013).

For instance, in (Scheel & Bisson, 2012), it was found that based on the scatter of calculated solutions for fixed-position reference transmitters, HPE<20 represented a positional error average of 5.2m in their dataset. Note that observations such as these are specific to a given dataset.

(Coates, Hovel, Butler, Klimley, & Morgan, 2013, p. supplement) states that if a relationship is found between HPE_m and HPE for synctags, it can be used to filter both synctag and animal tag positions past a threshold level of HPE. Their methodology was based on binning positions by HPE, and examining the median, 90^{th} and 95^{th} percentiles of HPE_m.

It is important to recognize that assessing accuracy is complicated by inaccuracies in the measurements of the station positions. For example, metres of GPS measurement error for receiver and synctag stations will have a significant effect on measuring VPS accuracy if the actual error is 2 metres. Instead of comparing calculated positions with a GPS measurement that has some unknown error, the spread (i.e. precision) of sets of successive calculated positions can be calculated. Higher precision is suggestive of higher accuracy. By examining precision over time, temporal variations in accuracy can be identified, but not systematic biases (Biesinger, et al., 2013).

Note that high accuracy may not always be necessary for a study. In many cases, it may be more important for calculated positions to be accurately positioned *relative to other calculated positions* in a local area, e.g. successive positions over time of the same tag or of multiple tags. If this is the case, precision is more important than absolute accuracy, and is easier to measure.

Error Sources

This section outlines the specific sources of error in VPS. Since hyperbolic positioning in VPS is based on receiver positions, transmitter depth and range differences, we will discuss the sources of error in each of these.

Receiver Positions

A receiver's position is defined by a 3D coordinate. A receiver's *horizontal* position is defined by its geographic coordinate (latitude and longitude). These are most commonly measured using a commercial-grade GPS receiver, and can contain the following sources of error:

A. Intrinsic GPS error (typically 2-3m)
B. GPS error due to poor satellite geometry (can be significantly more than 2-3m)
C. Difficulty positioning GPS antenna directly over the receiver due to mooring line tilt, particularly when the receiver is in deep water and/or being deployed by large vessel
D. Associating GPS measurements with the wrong receiver serial number
E. One unexpected receiver movement, and its new position was not measured before retrieval
F. Two or more unexpected receiver movements resulting in some number of unmeasured receiver positions

A receiver's depth is also required, but it is usually not as significant a source of error as its horizontal position. This is because typical receiver depths (10 – 20 meters) are relatively small compared to the typical distances between neighbouring receivers (250 - 400 meters).

Transmitter Depth

The VPS basic position calculation requires the depth of the transmitter. When a transmission includes a depth sensor measurement, that depth is used in the position calculation. When depth sensor measurements are not available, VPS uses a specified assumed depth, and the difference between actual depth and assumed depth is a source of error. In most VPS systems, where the potential depth range is only a small fraction of the distance between receivers, depth error has minimal effect.

Range Differences

Range differences are calculated by converting differences in time-corrected detection times to distance. This conversion assumes *ideal propagation*: that transmissions travel omnidirectionally in straight lines at a constant finite speed.

All hyperbolic positioning systems are subject to the following sources of range difference error:

G. Detection time measurement error
H. Non-ideal propagation, e.g. non-linear transmission path, non-constant propagation speed over transmission path
I. Propagation speed measurement (e.g. water temperature & salinity measurement error for underwater acoustics)

In VPS, range difference error is significantly more complex in nature than receiver position error. This is because the autonomous receivers typically used by VPS are not time synchronized, and in order to measure differences in detection times based on a common clock, synctags are required. Sources of error due specifically to the usage of synctags are

J. Synctag position measurement (i.e. A – F applied to synctags)
K. Synctag range difference measurement (i.e. G – I applied to synctags)

Conclusion

In this paper, we have covered the general concepts behind hyperbolic positioning, the mechanism used by VPS, and the nature of its sensitivity to error. We examined the specific sources of error in VPS, and provided details on how VPS calculates positions and estimates error sensitivity. Other aquatic positioning systems employ metrics for quantifying relative error sensitivity (Niezgoda, Benfield, Sisak, & Anson, 2002), (Biesinger, et al., 2013), (Ehrenberg & Steig, 2002).

We hope that by providing a concise definition of HPE, and providing guidance on determining ways of applying HPE to characterize the error of a VPS dataset in terms of distance, we have succeeded in our goal of helping customers understand HPE.

Acronyms and Abbreviations

2D	Two dimensions, two dimensional
2DRMS	Twice the distance root mean square
3D	Three dimensions, three dimensional
D	Detection time
DD	Detection time difference
LOP	Line of position
NRD	Normalized range difference
PPM	Parts per million
P	Propagation time
PD	Propagation time difference
RD	Range difference
SOS	Speed of sound
TDOA	Time difference of arrival
T	Transmission time
VPS	VEMCO Positioning System

References

Biesinger, Z., Bolker, B. M., Marcinek, D., Grothues, T. M., Dobarro, J. A., & Lindberg, W. J. (2013). Testing an autonomous acoustic telemetry positioning system for fine-scale space use in marine animals. *Journal of Experimental Marine Biology and Ecology*, 46-56.

Coates, J. H., Hovel, K. A., Butler, J. L., Klimley, P. A., & Morgan, S. G. (2013). Movement and home range of pink abalone Haliotis corrugata: implications for restoration and population recovery. *Marine Ecology Progress Series*, 189-201.

Coppens, A. B. (1981). Simple equations for the speed of sound in Neptunian waters. *Journal of the Acoustical Society of America, 69*(3), 862-863.

Ehrenberg, J. E., & Steig, T. W. (2002). A method for estimating the "position accuracy" of acoustic fish tags. *Journal of Marine Science*, 140-149.

Enge, P., & Misra, P. (2006). *Global Positioning System.* Lincoln: Ganga-Jamuna Press.

Espinoza, M., Farrugia, T. J., Webber, D. M., Smith, F., & Lowe, C. G. (2011). Testing a new acoustic telemetry technique to quantify long-term, fine-scale movements of aquatic animals. *Fisheries Research*, 364-371.

Fang, B. T. (1990). Simple Solutions for Hyperbolic and Related Position Fixes. *IEEE Transactions on Aerospace and Electronic Systems*, 748-753.

Klimley, P. A., Le Boeuf, B. J., Cantara, K. M., Richert, J. E., Davis, S. F., & Sommeran, S. V. (2001). Radio-acoustic positioning as a tool for studying site-specific behavior of the white shark and other large marine species. *Marine Biology*, 429-446.

Niezgoda, G., Benfield, M., Sisak, M., & Anson, P. (2002). Tracking acoustic transmitters by code division multiple access (CDMA)-based telemetry. *Hydrobiologia* , 275-286.

O'Dor, R. K., Andrade, Y., Webber, D. M., Sauer, W. H., Roberts, M. J., Smale, M. J., & Voegeli, F. M. (1998). Applications and performance of Radio-Acoustic Positioning and Telemetry (RAPT) systems. *Hydrobiologia*, 1-8.

Piraino, M. N. (2011). Fine-scale movements and home ranges of red snapper Lutjanus campechanus around artificial reefs in the northern Gulf of Mexico. *Doctoral Dissertation*.

Scheel, D., & Bisson, L. (2012). Movement patterns of giant Pacific octopuses, Enteroctopus dofleini. *Journal of Experimental Marine Biology and Ecology*, 21-31.

Appendix A: Rationalizing Sensitivities to Measured Error

The linear function that is usually used to map HPE_s to HPE is derived as follows. For all basic positions of synctag and reference transmitters with known positions, bin the positions based on HPE_b using a bin size of 1. Basic positions and HPE_b are used as the basis of the calibration because there are typically significantly more basic positions than synthesized positions, providing more data on which to base the calibration. For each of these bins, calculate X, Y and N as:

- X = average HPE_b from the bin
- Y = 2DRMS of the calculated positions from the bin (see below)
- N = number of calculated positions

The 2DRMS (twice the distance root mean square, (Enge & Misra, 2006, p. 215) of a set of calculated positions is $2 \times \sqrt{{\sigma_x}^2 + {\sigma_y}^2}$, where σ_x and σ_y are the standard deviations of the x and y components of the calculated positions. A weighted linear regression of these (X, Y) points is calculated using N as the weight.

Figure 19, Figure 20 and Figure 21 show the relationship, at three different scales, between HPE_b and HPE_m for synctag basic positions in a typical VPS dataset. Note that these are internal to the VPS software. Each green dot represents a calculated basic position, and its location defined by HPE_b and HPE_m. The blue circles show for each bin the 2DRMS of the positions in the bin. The size of the blue circle is $\ln N$. The blue line represents the weighted linear regression.

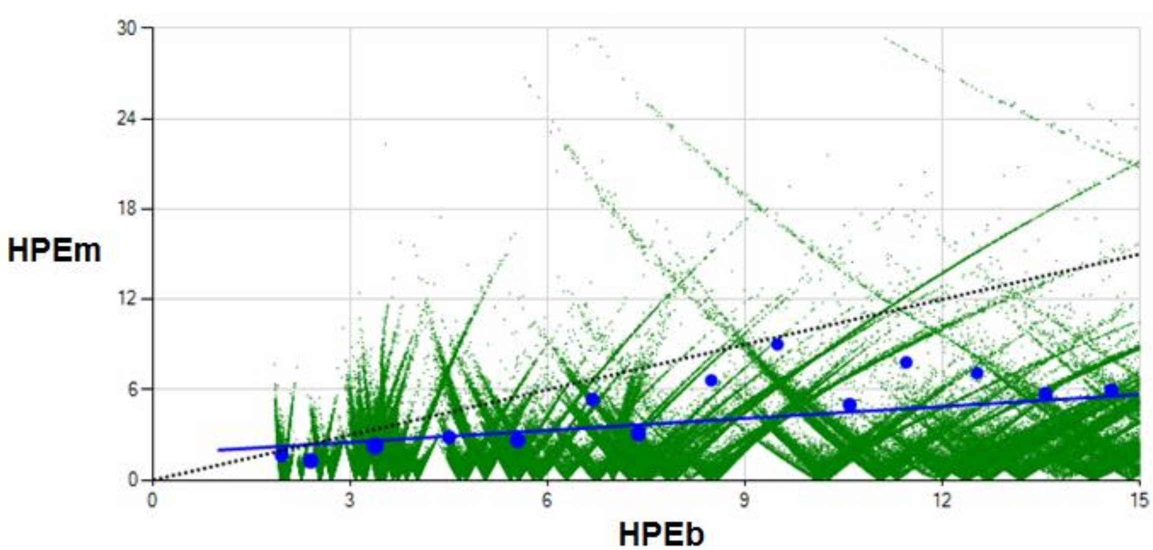

Figure 19

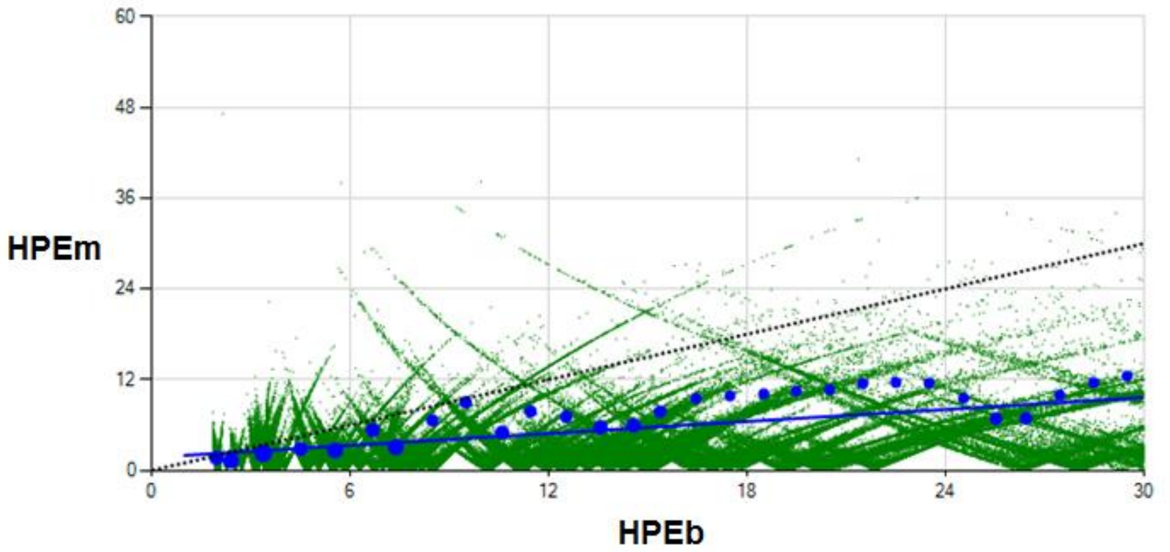

Figure 20

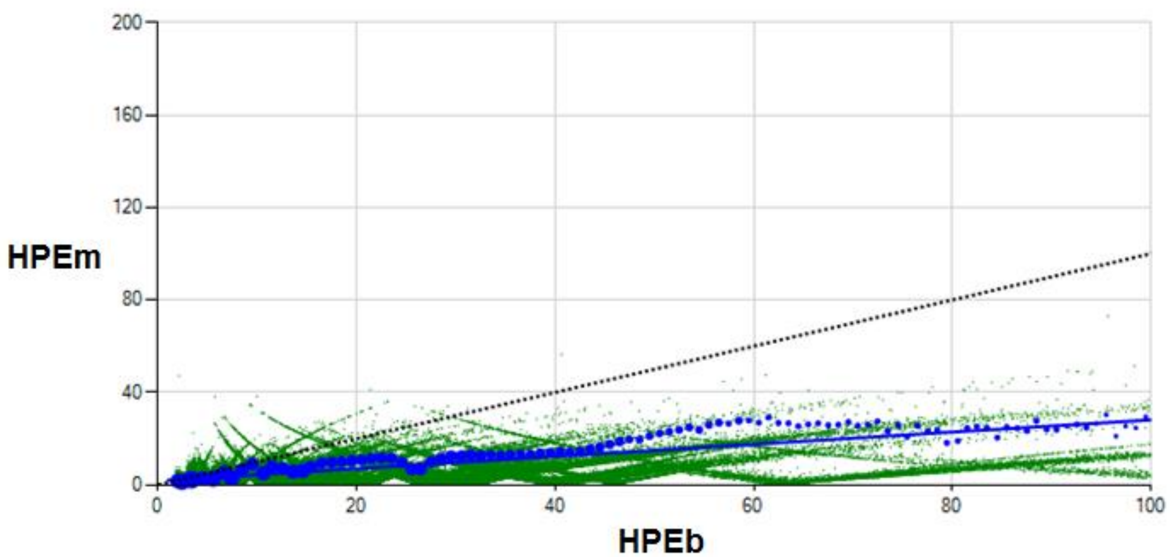

Figure 21