Appendix B

The Stability of The ITM for Typical Problems

The ITM is a well known and widely used model for predicting propagation loss in long (greater than one kilometer) outdoor radio links. This model was developed by Hufford *et al.* in [99] for the National Telecommunications and Information Administration (NTIA) Institute for Telecommunications Sciences (ITS). The model predicts the median attenuation of the radio signal as a function of distance and of losses due to refractions at intermediate obstacles. Compared to the vast majority of other models, even those that are similar in approach (e.g., The International Telecommunications Union (ITU) Terrain Model [206]), the ITM is very complicated, requiring the interaction of dozens of functions that implement numerical approximations to theory. Due to this complexity, the question of numerical stability is an obvious one, but has not previously been investigated.

This section takes a systematic empirical approach to the analysis that involves porting the defacto C++ implementation of the ITM [98] to a multiprecision framework. A comparison is made between the predicted path loss values for many randomly generated links over real terrain data. Model parameters are also varied in order to produce a fully factorial experimental design over a range of realistic parameters. In the end, the results show that while the model performs disastrously for half-precision (16 bit) arithmetic, it is well behaved for single-precision (64 bit) and higher precisions. Within the values tested, there are very few isolated cases that result in significantly different (greater than 3 dB) output and these tend to result from a single change in branching decision in the approximation algorithms and not because of massive information loss. While this sort if empirical analysis cannot be used to extrapolate to any parameters and

[0] Work in this appendix has appeared in [173].

any terrain model, the results show that over realistic links the model appears to be well-behaved. This result provides confidence in the stability of the output of the ITM model as well as other similar models that compute diffraction over terrain (e.g, [206, 109]).

B.1 Implementation

The multiprecision framework used here is based on the combination of three open source libraries: MPL, MPC, and MPFR [10, 11, 7]. The MPL library provides basic arbitrary precision support. The MPFR library wraps the MPL library and provides additional necessary features such as a square root function, computation of logs and powers, and trigonometric functions. The MPC library provides support for complex arithmetic. In porting, the ITM source is modified to take an additional command line argument that specifies the precision in bits, which is passed to the multiprecision framework. Otherwise, the functionality and usage is identical to the machine-precision ITM implementation.

The implementation involves a line by line port of the reference ITM implementation to have multiprecision support. By and large, this involves using multiprecision data structures in place of native machine number formats. For instance, The following (commented) equation might be translated into four MPFR function calls:

```
#fhtv=0.05751*x-4.343*log(x);
mpfr_log(tmp,x,R);
mpfr_mul_d(tmp,tmp,4.343,R);
mpfr_mul_d(fhtv,x,0.05751,R);
mpfr_sub(fhtv,fhtv,tmp,R);
```

B.2 Experiment

The experimental design involves generating random link geometries within a latitude and longitude bounding box. For each random link, a path loss prediction is made both with the machine precision (64-bit double precision arithmetic) and multiprecision implementation (at a variety of precisions). After the fact, we can quantify the differences in predictions and investigate any outliers or general trends.

The bounding box is from 39.95324 to 40.07186 latitude and -105.31843 to -105.18602 longitude. This box contains a portion of the mountainous region to the west of Boulder, Colorado, as well as the plains to the east, providing a realistic mix of topographies. 500 links are generated uniformly at random within the box. Antenna heights are also selected uniformly at random between 0 and 35 meters. For each link, the corresponding elevation profile is extracted from a USGS DEM with 0.3 arcsecond raster precision.

B.3 Results

Figure B.1 shows the overall results of this experiment: the error (ϵ) between the multiprecision prediction and the machine precision prediction is plotted. Half-precision arithmetic (11 bits of exponent, 16 bits total) produces results that vary wildly. Above this, however, beginning at single precision (24 bits of exponent, 32 bits total), the two programs make very similar predictions. Figure B.2 provides a more detailed picture of these remaining cases. Much of the small error is negligible as it is presumably a function of differences in rounding[1]. In the results, there is one clear outlier that produces a 6 dB difference. The case was the result of a difference in branching decision that chooses whether or not to make a correction. It is not clear that one direction down the branch offers a better prediction than another, so this case can be safely ignored.

Lastly, figure B.3 shows the performance, in terms of running time for the various precisions. The multiprecision version is not substantially slower than the machine precision implementation. If it were the case that the multiple precision implementation was also safer, then its use would be clearly preferable.

B.4 Discussion

Although it is not possible to extrapolate universally from these results, they demonstrate that the ITM is *not* substantially unstable for typical problems and reasonably precise numeric types (i.e., single and double precision IEEE formats). An analytical investigation of stability would go a long way to determine the stability universally, but is a substantial undertaking that involves the careful dissection of dozens of complex algorithms that combine to create the ITM implementation. An intrepid investigator, may choose to focus his effort on the knife-edge diffraction approximation algorithm, which is almost certainly the most

[1] IEEE 754-2008 requires subnormal arithmetic rounding, which is not done natively by the MPFR library. The majority of rounding (excluding this special case) are identical

numerically complex component of the model. For our purposes, however, the results presented here are sufficient to justify continued use of this model with the confidence that under typical situations it is not significantly affected by rounding and cancellation errors.

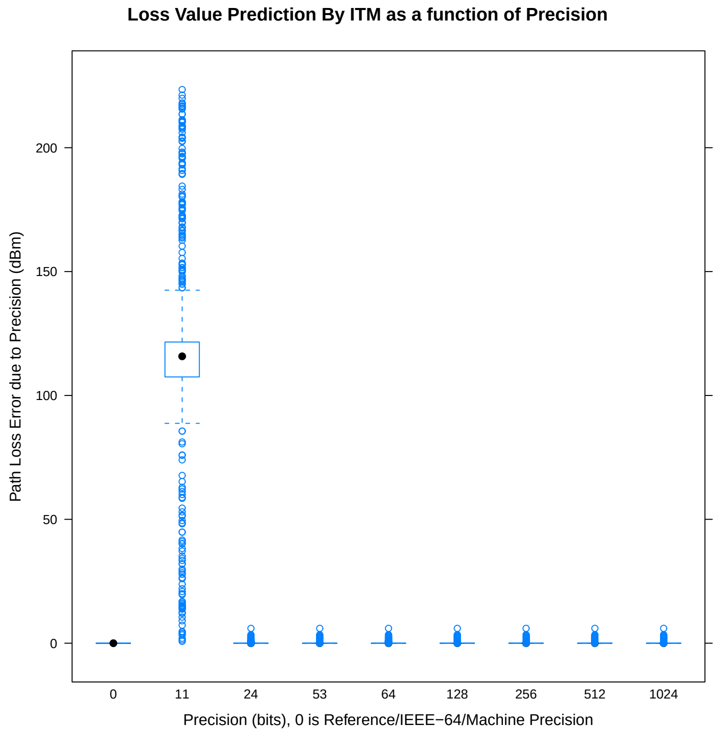

Figure B.1: Box and whiskers plot of error as a function of precision.

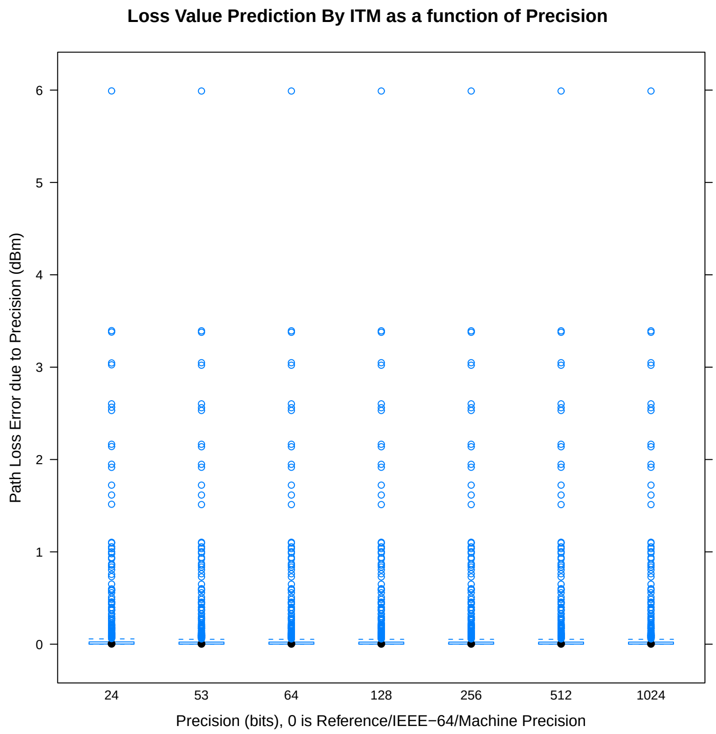

Figure B.2: Box and whiskers plot of error as a function of precision, showing only results for single-precision and greater arithmetic.

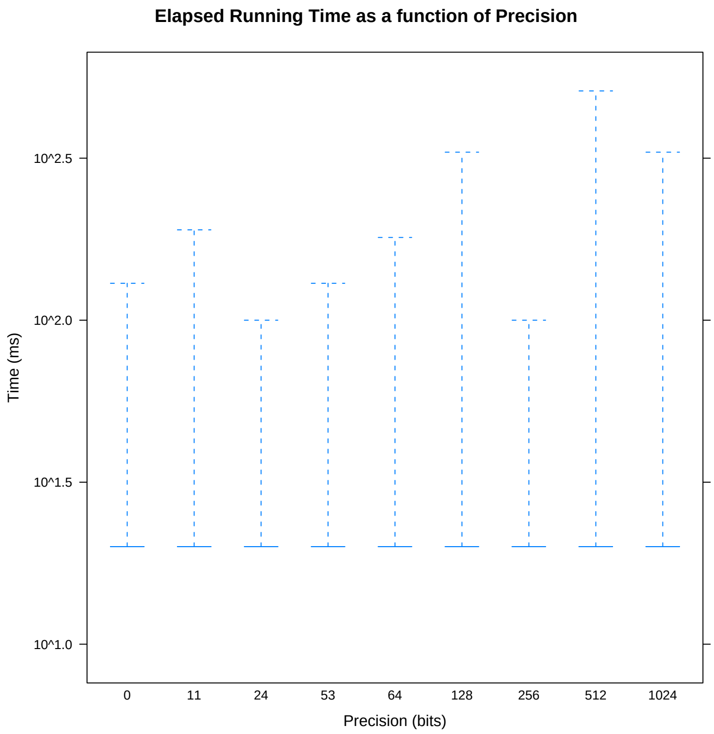

Figure B.3: Running time of ITM algorithm as a function of precision. The 0-bit case is the machine-precision reference implementation.