Title:	Details on Mapping Channel Rays to Samples for System Simulations
File:	SCM-092_Ray_Mapping_Details.doc
Source:	Motorola
Date:	January 8-9, 2003

1. SUMMARY

This contribution provides the details for a method of mapping arbitrary channel rays to discrete receiver samples for use in system simulations. The two key benefits of this method are that the original channel delays are maintained to a high degree of accuracy and that a variety of receiver algorithms may be used. Other methods can introduce inaccuracies in the SINR computations or restrict the selection of receiver algorithms.

2. PROBLEMS WITH OTHER METHODS

Other contributions have been presented on the topic of using the SCM produced channel impulse responses in receiver SINR computations.

In [1], a method of ray splitting was proposed. The major drawbacks of the ray splitting method are covered in [2] and [3]. In summary, the ray splitting method introduces additional rays into the impulse response that can increase the simulation time as well as significantly distort the spatial and temporal nature of the channel. Furthermore, the channel sampling rate is restricted to the chip rate, which does not allow for fractionally spaced equalizers.

In [3], a method of using the unmodified spatial channel model was proposed. The basis of this contribution was a simplified Rake receiver that directly computes the signal and interference power at each Rake finger. The procedure bypasses any sampling of the channel impulse response. While this method faithfully adheres to the original channel model, it is unclear how to handle MMSE equalizers, diversity receivers, or MIMO transmission techniques.

3. DETAILS OF THE CHANNEL SAMPLING METHOD

Figure 1 shows a block diagram of the effective continuous-time channel impulse response. The goal is to produce a sampled version of $g(t)$ that can be used for various receiver implementations.

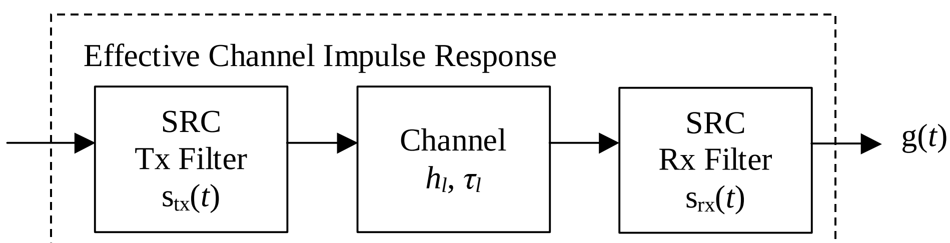

Figure 1. Continuous-time channel impulse response.

The following equation represents the effective continuous time impulse response (convolution of the transmit filter and the channel):

$$g(t) = \sum_{l=1}^{L} h_l \cdot s(t - \tau_l),$$

where L is the total number of modeled rays, and h_l is the complex amplitude of the l-th ray and $\mathrm{s}(t-\tau_l)$ is the combined impulse response of the transmit and receive SRC filters at delay τ_l.

In an MMSE receiver implementation, this channel impulse response needs to be sampled at some rate ($1/T_s$):

$$\mathrm{g}(nT_s) = \sum_{l=1}^{L} h_l \cdot \mathrm{s}(nT_s - \tau_l)$$

The resulting set of samples, **g**, can be directly used in the MMSE SINR computations provided in Appendix A of the SCM text [4]. Note that the SINR calculations take into account both the spatial and temporal properties of the signal, interference, and noise.

The above expression requires a continuous-time description of the filter, s. The continuous-time filtering may be time consuming to compute during the simulation, or may not be defined at all (as in the 3GPP2 transmit filter). To simplify the procedure, the channel delays can be quantized to a very small interval that does not affect the nature of the channel. Figure 2 shows the amount of distortion of the system level SCM channel caused by the ray-splitting, triangular pulse filtering, and sample rate change channel mapping approaches. The distortion is computed as the mean-square-error between an unquantized SCM channel filtered with SRC transmit/receive filters and one of the 3 channel mapping techniques[1]. The ray splitting approach distributes the energy of a ray between chips at the chip rate, as described in [1]. The triangular pulse based mapping method uses a triangular pulse in place of the transmit and receive filters. The sample rate change based method uses an SRC transmit filter (operating at the specified number of samples per chip) and maps the time at which the rays occur to the nearest sampling instant, as described above. Note that the results shown for ray splitting and triangular pulse methods are reference lines. The ray-splitting method always uses channel rays quantized to the chip rate and the triangular pulse method always uses an unquantized description of the channel.

Examining Figure 2, we see that the ray splitting method always has the greatest distortion, while the other two methods' relative performance varies with sample rate. The ray splitting method is always at least an order of magnitude worse in error power over the better of the two other methods. The triangular pulse method is better than the quantized channel method at low numbers of samples per chip, since it has better time resolution. However, we note that for rates higher than 4 samples per chip, the quantized channel method outperforms the triangular pulse. At 16 samples per chip, the error power is almost two orders of magnitude lower than that of the triangular pulse.

The cost of increasing the number of quantization levels per chip is simply the amount of memory required to store the transmit and receive filters. The total simulation time is unaffected, since only one or two samples per chip are actually used in the system simulations. Examining the quantized channel curve, we note that it flattens around 16 samples per chip and

[1] The mean square error calculation is done after normalizing the filtered, unquantized channel and the ray mapped channels to unity power, and is at a rate of 192 times the chip rate.

provides two orders of magnitude less error than the triangular pulse method. This low amount of distortion and the modest amount of memory required to store the filtered SCM channel lead us to use 1/16 of a chip quantization for discussion purposes. For a transmit filter with a length of 6 chips, an array with 96 floating point numbers would be stored in memory.

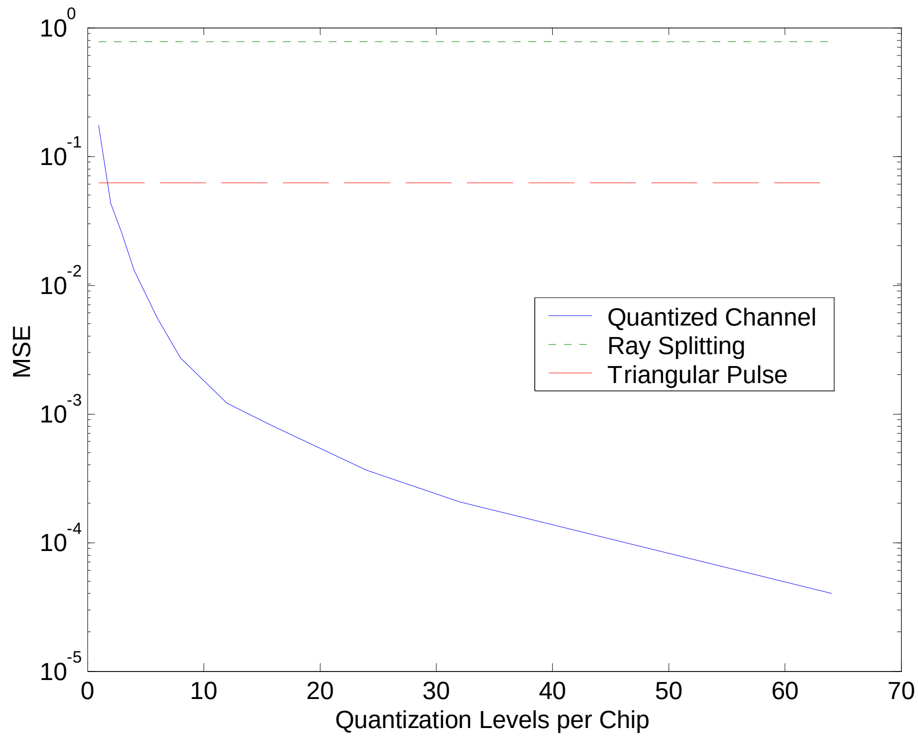

Figure 2. Mean Square Error of various channel representation techniques.

Based on a quantized transmit filter, the received samples can be expressed as:

$$g(n) = \sum_{l=1}^{L} h_l \cdot s(M \cdot n - n_l)$$

where g is the discrete-time version of the effective channel impulse response sampled at M/16 samples per chip, M is the decimation factor (M=16 corresponds to chip spaced receiver sampling), s is the discrete-time version of the transmit filter sampled at 16 samples per chip, and n_l is the delay in 1/16 chip samples of the l-th ray. Figure 3 graphically represents the procedure.

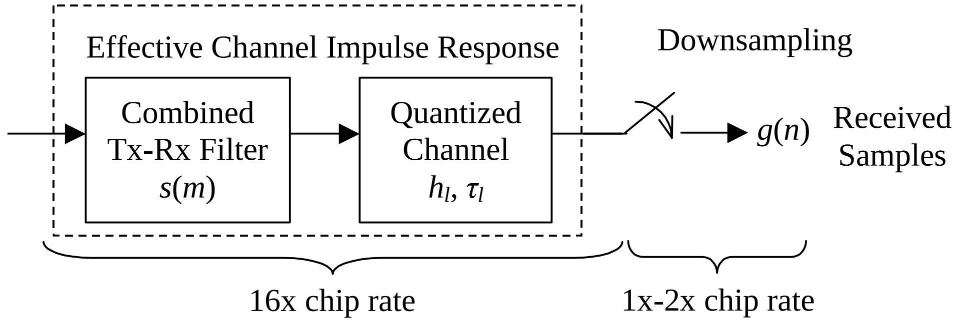

Figure 3. Channel impulse response with sampling.

In general, it is necessary for receiver algorithms to have access to the sampled channel impulse response. With this information, both a Rake and MMSE receiver can be simulated. Also, it is straightforward to account for the spatial nature of multiple transmit antennas and the other-cell interference.

4. CONCLUSIONS

Details were presented on an efficient channel impulse response sampling method that can be seen as a generalization of the methods of [1], [5] and [3], allowing fractionally spaced equalizers and arbitrary choices of transmit and receive filters, while faithfully representing the spatial and temporal characteristics of the channel. The ray splitting approach of [1] can be thought of as applying a triangular filter to the channel responses, then decimating to the chip rate. Alternatively, the tx-rx filters can be approximated by a triangular pulse [5]. The method of [3] uses exact continuous time representations of the tx-rx filter, producing signal and interference powers at the symbol rate. The sample rate change based method discussed here allows any transmit or receive filter to be used and sample rates of more than once per chip.

5. REFERENCES

[1] SCM-087, Ericsson and Nokia, "Splitting Rays into Resolvable Paths," December 2002.

[2] SCM-085, Motorola, "Mapping Channel Rays to Samples for System Simulations," December 2002.

[3] SCM-088, Lucent, "Using the Unmodified Spatial Channel Model for System Simulations," December 2002.

[4] SCM-083, "Spatial Channel Model Text Description," December 2002.

[5] Minutes of the Dec. 19, 2002 SCM conference call.