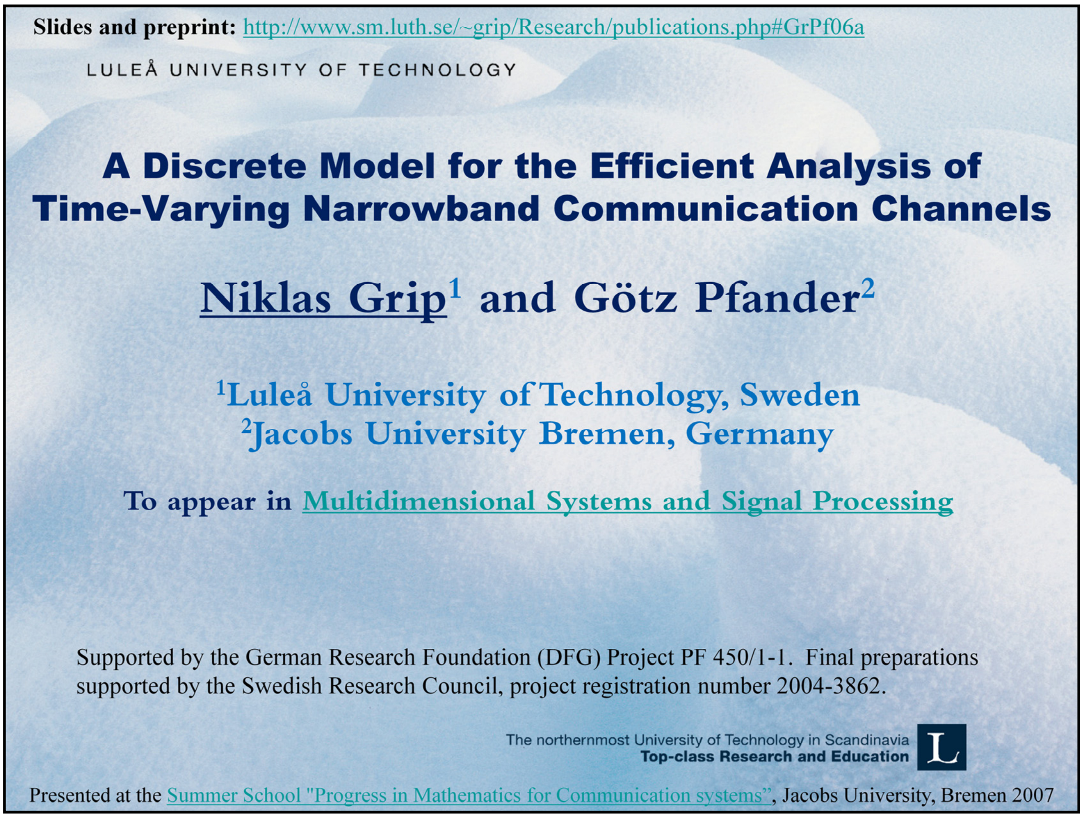
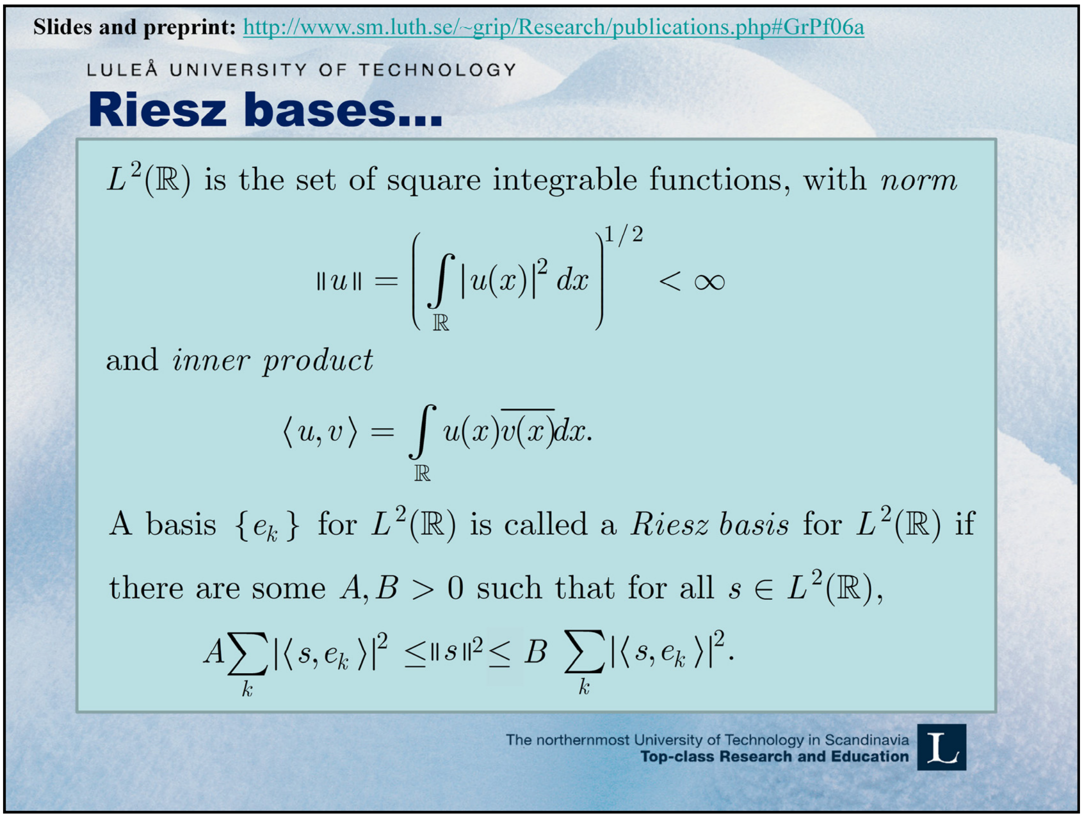


Note to the reader: The accompanying these slides reflects roughly but not exactly what was said in the actual presentation. A more detailed and mathematical presentation of the subject can be found in the preprint linked to above.

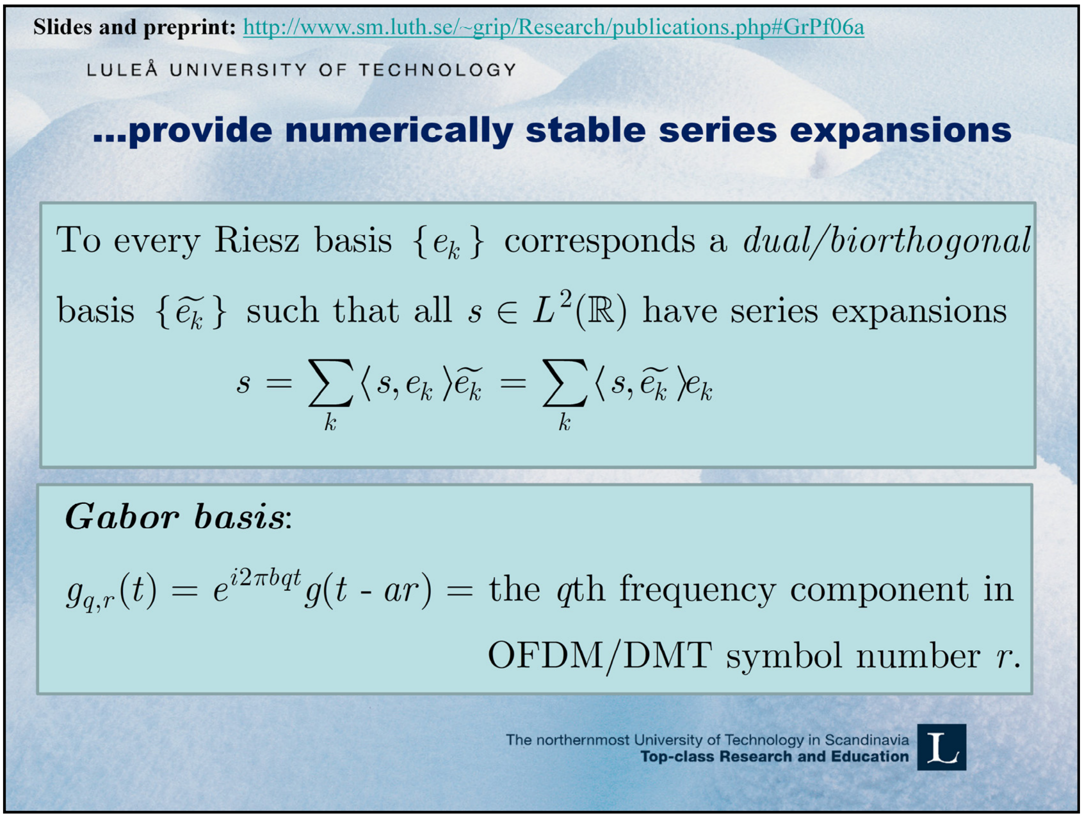
Beginning of the talk:

In this talk I will describe some joint work with Götz Pfander from when I had a PostDoc position here in Bremen. Starting from a Riesz basis setting, I will describe a certain matrix repereresentation of communication channels. Then I will describe an algorithm for efficient computation of this matrix.

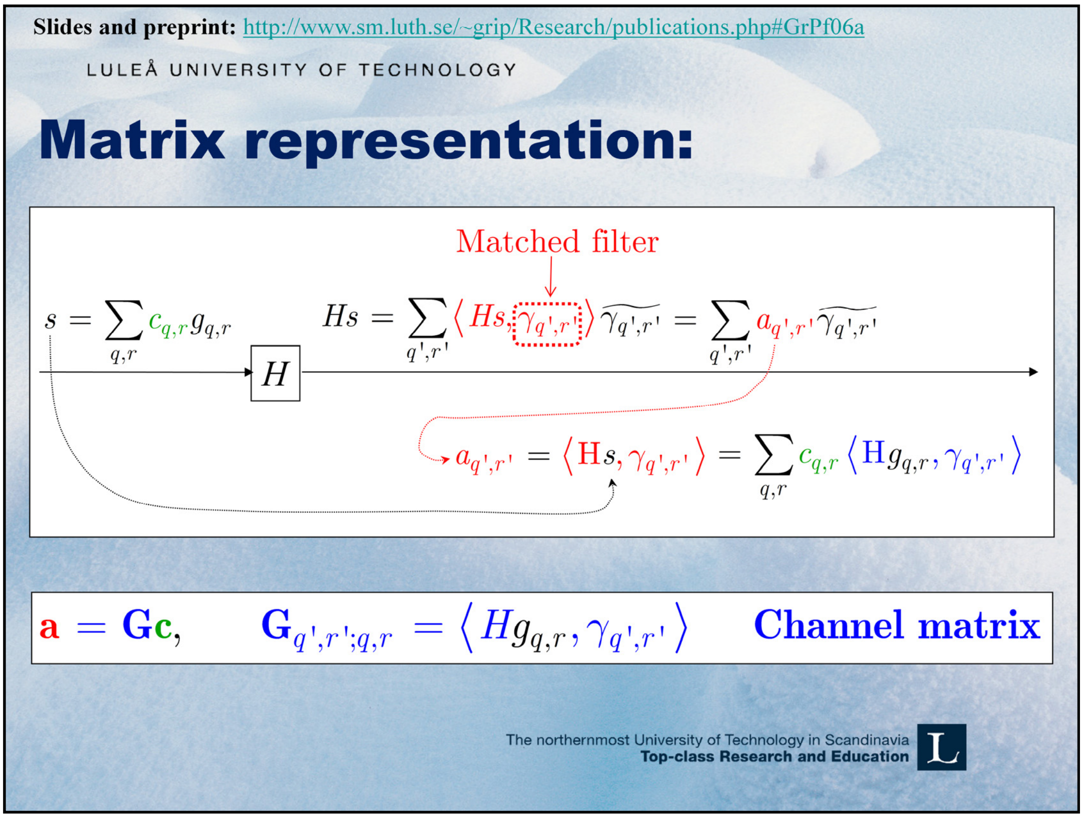


- The mathematical setting for Riesz bases is the space of square integrable functions.

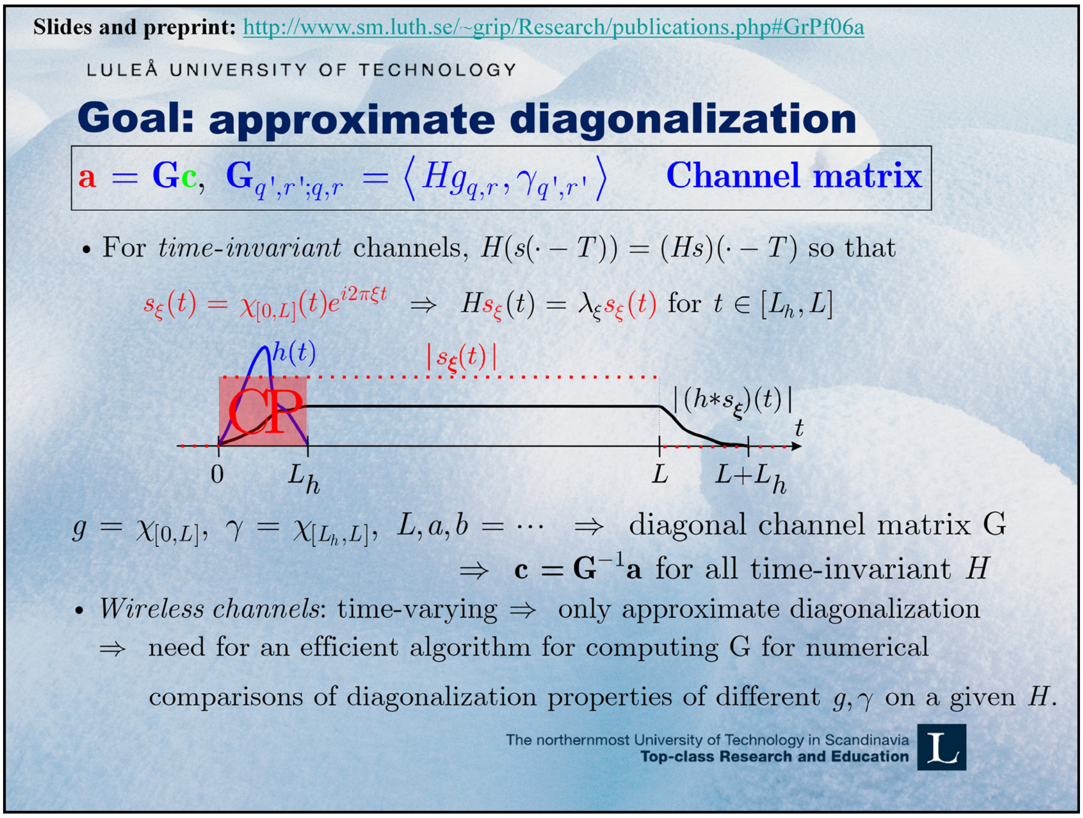
- For orthonormal bases such as the Fourier basis, many or all of you already know of the *Parseval* relation, which tells that the energy of the signal equals the energy of its Fourier series coefficients.
- What you ***should*** remember from this slide is this bracket notation for the inner product or correlation of the functions *u* and *v*.
- This is an unnecessarily strong restriction, and Riesz basis is exactly what we get if we weaken the Paresval equality to a double inequalty.
- This is a numerical stability condition in the sense that the square root of B/A is the condition number of the mappping from functionsto coefficients, so that small B/A means that a small error in the coefficients correspond to a small perturbation of the function.
- “Basis” means that all square integrable functions have a series expansion…



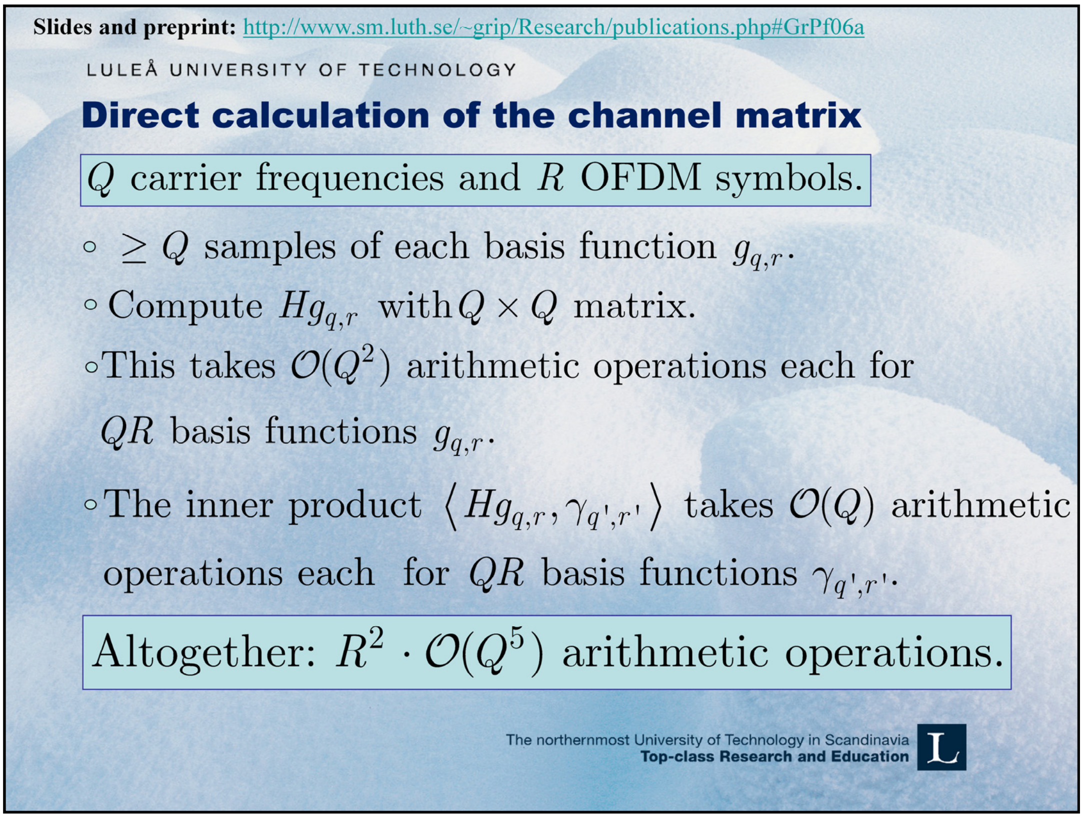
- …which is particularly easy for Riesz bases.
- In fact, to every Riesz basis corresponds a *dual* or *biorthogonal* basis, denoted with tilde, such that all square integrable functions *s* satisfy this series expansion formula:
- For orthonormal bases, e_k equals e_k-~tilde, so that these to formulas coincide.
- In the remaining slides we willonly consider Gabor bases, which are defined via a window function *g* and some lattic indicec *a* and *b* such that … or in engineering terminology …



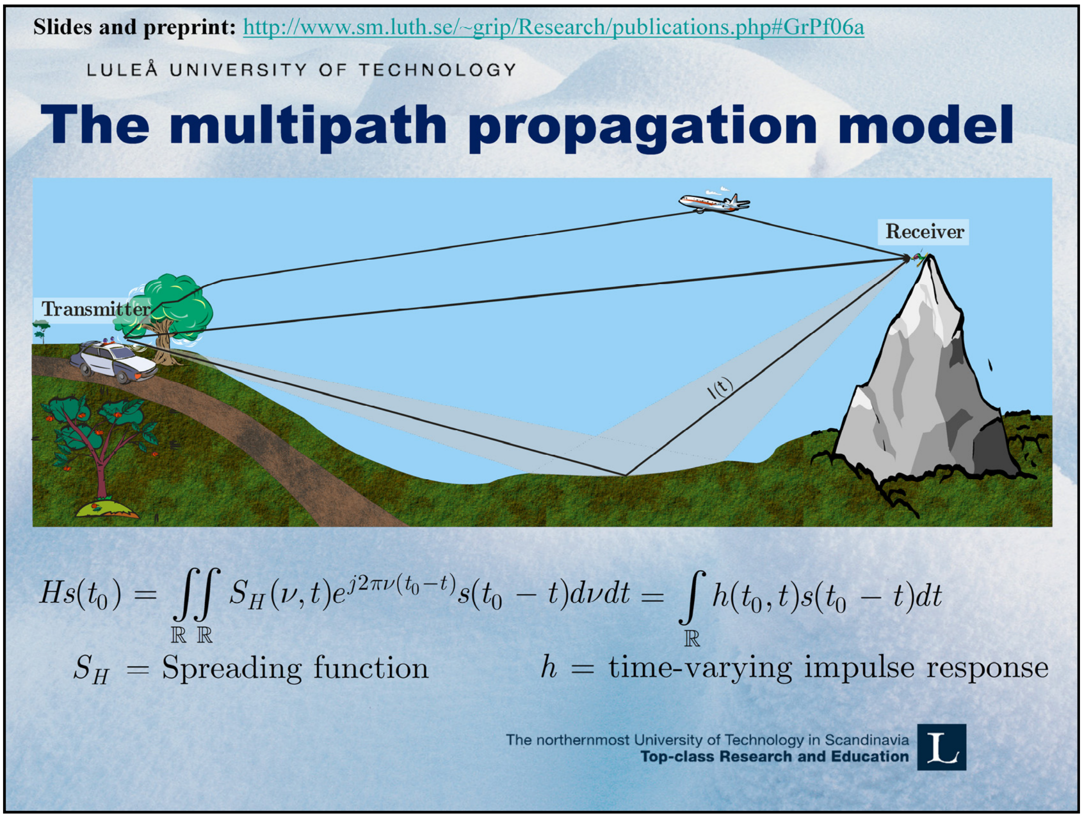
- Suppose now that we send an OFDM signal *s* through a channel *H,* and expand *Hs* in some Riesz basis *gamma.*
- Some of you might, recognise this function as a matched filter, but in this talk, we will rather consider the computation of the coefficients $a_{q',r'}$ obtained by inserting the upper left expression for s and moving the sum and coefficients outside the inner product.
- This sum for obtaining the received coeccicients *a* from the transmitted coefficients *c* we can easily rewrite as a matrix-vector multiplication … with G = …
- We call G the channel matrix.



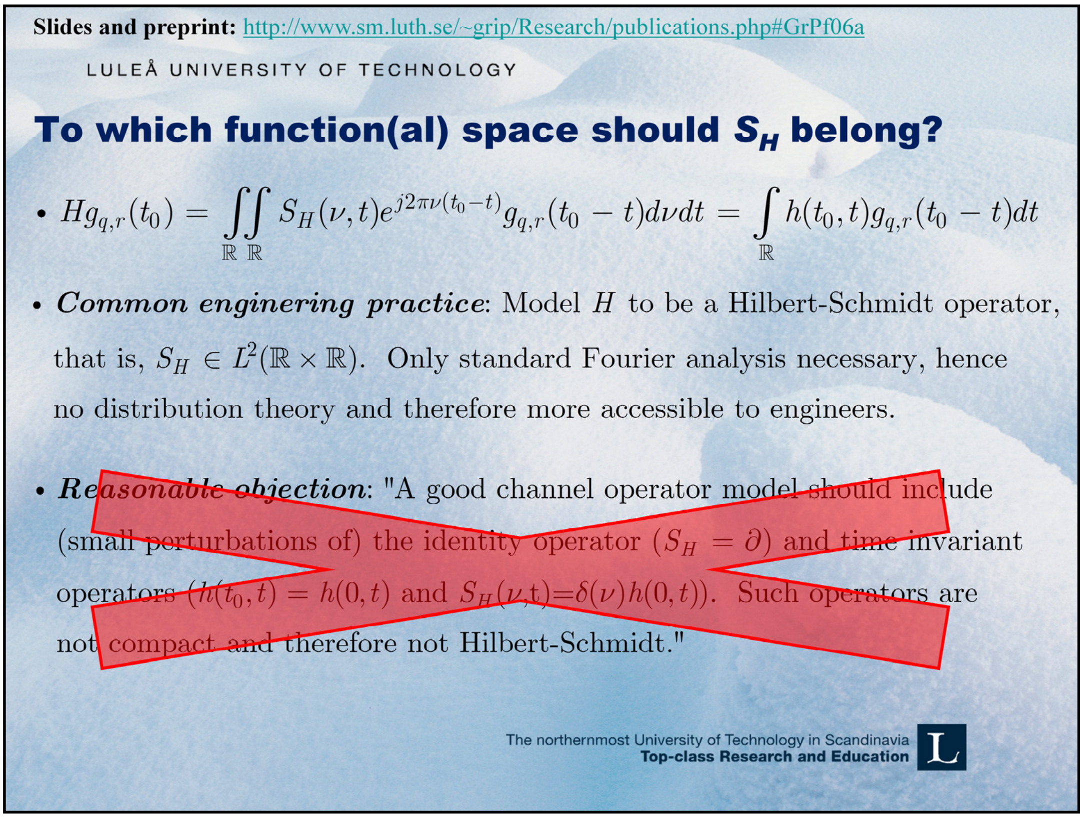
- We would prefer G to have some simple sparse structure, and for *time-invariant* channels, we can enen make it *diagonal*.
  A *time-invariant* operator *H* is one for which doing a time-shift with *T* and then applying *H* gives the same result as first applying *H* and then doing the time shift.
  For time-invariant operators, complex exponentials are eigenfunctions.
  In practice, we can only send complex exponentials restricted to some finite length interval, like here, $[0,L]$.
  The absolute value of this function is plotted as his red box function here.
  Now if we apply a time-invariant operator with impulse response living in $[0,L_h]$, then the output will look something like what's plotted in black here and in the interval $[L,L_h]$ it still holds that the output equals the input multiplied with some eigenvalue.
  Hence, with this choice of Gabor windows and with appropriately chosen *L*, and lattice constants *a,b*, we get a diagonal G for ***all*** time-invariant channels with impulse response length L_h.
  The elements of G will be the channel frequency response, computing $c=G^{-1}a$ is what's referred to as the equalizer in engineering terminology, where the interval of *g* removed in *gamma*, usually is referred to as cyclic prefix (CP).
- *Wireless communication channels* are time-varying, and two different time-varying operators do not usually commute, that is AB is not equal to BA. From this follows that both cannot be diagonalized with the same choice of bases, so the whole machinery up here falls apart and we can at most hope for *approximate* diagonalization.
  There is no simple formulas for computing "how close to diagoal" we get for a given *g*, *gamma* and *H*, so in general, numerical computations will be needed…



- But do we really need an ***efficient*** algorithm for that? Why not just a quick and dirty approach for, say, *Q* frequencies and *R* OFDM symbols?
- Well, then we could expect to need at least *Q* samples of each basis function, leading to a total need of $R^2O(Q^5)$ operations with Q typically being of the order 1000.
- That's *a lot*!
- For a more efficient computation of the channel matrix, we will exploit some useful physical properties of the channel and the windows.
- The first one follows from the well-known…

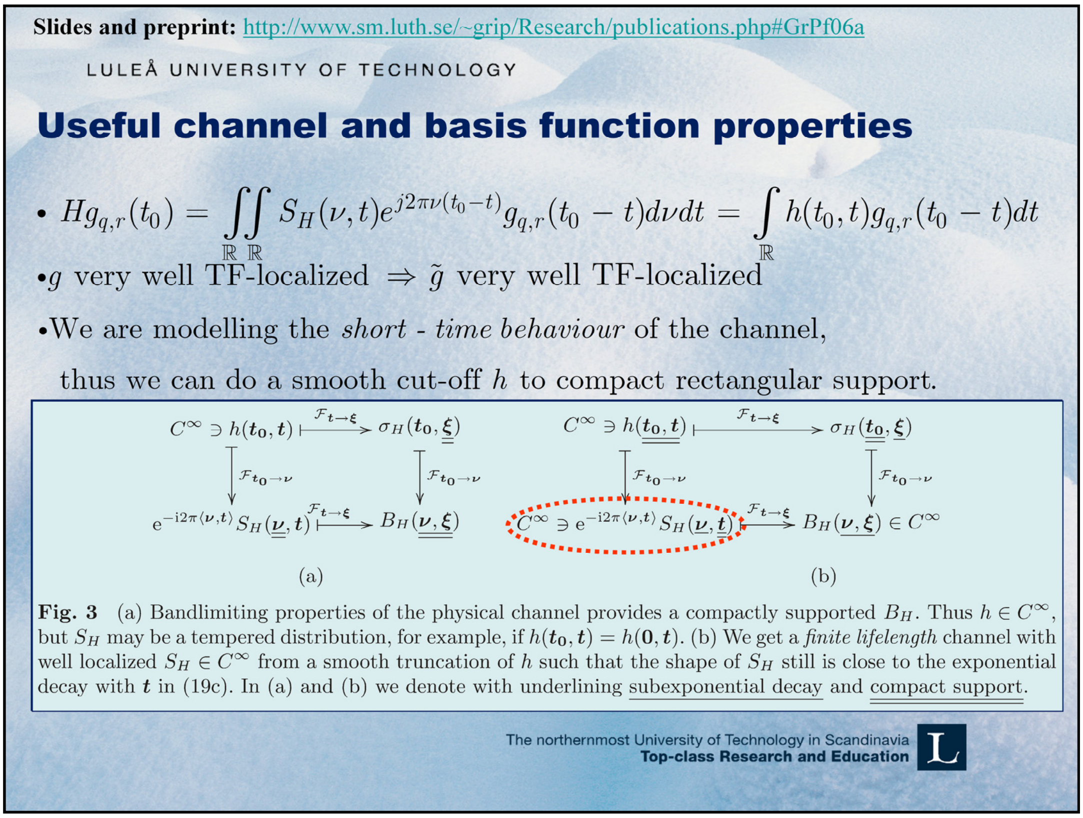


- …***multipath propagation model*** of the channel.
- In this simplified example, a poilce care tries to get a radio message through to some alpine skier.
- The signal reach the receiver along different signal paths, such as this line-of-sight path, this one reflected in a tree and ona plane or this contibuous set of paths reflected in some concave formation in the ground.
- The paths have different dlengths and thus cause different time delays.
- Also, since both the transmitter, reflecting objects and the reciever might be moving, the Doppler effect also causes different frequency shifts for different signal paths.
- Hence the received signal is a superposition of different time- and frequency-shifted copies of the transmitted signal with some weight function called spreading function.
- Another useful integral representation of the channel is via the time-varying impulse response, which describe the channel response at time t_0 to an impulse at time t_0-t.

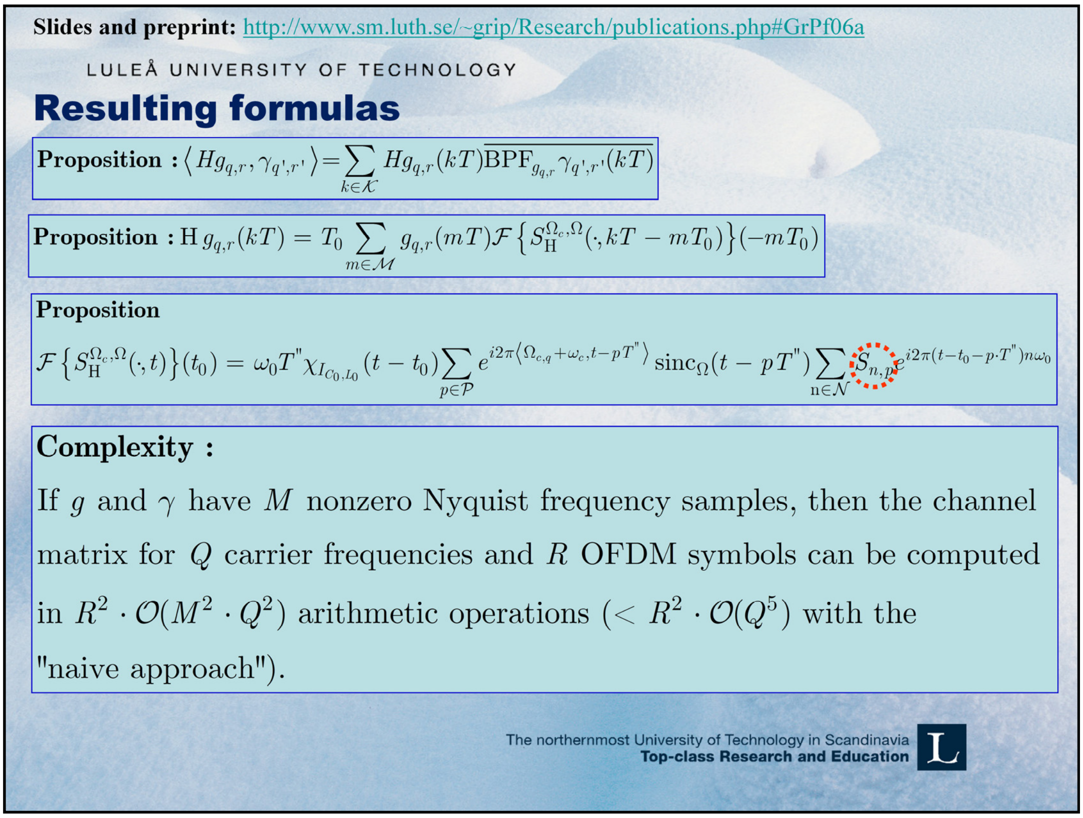


For our mathematical modeling it is important to decide to what function space the spreading function should belong.

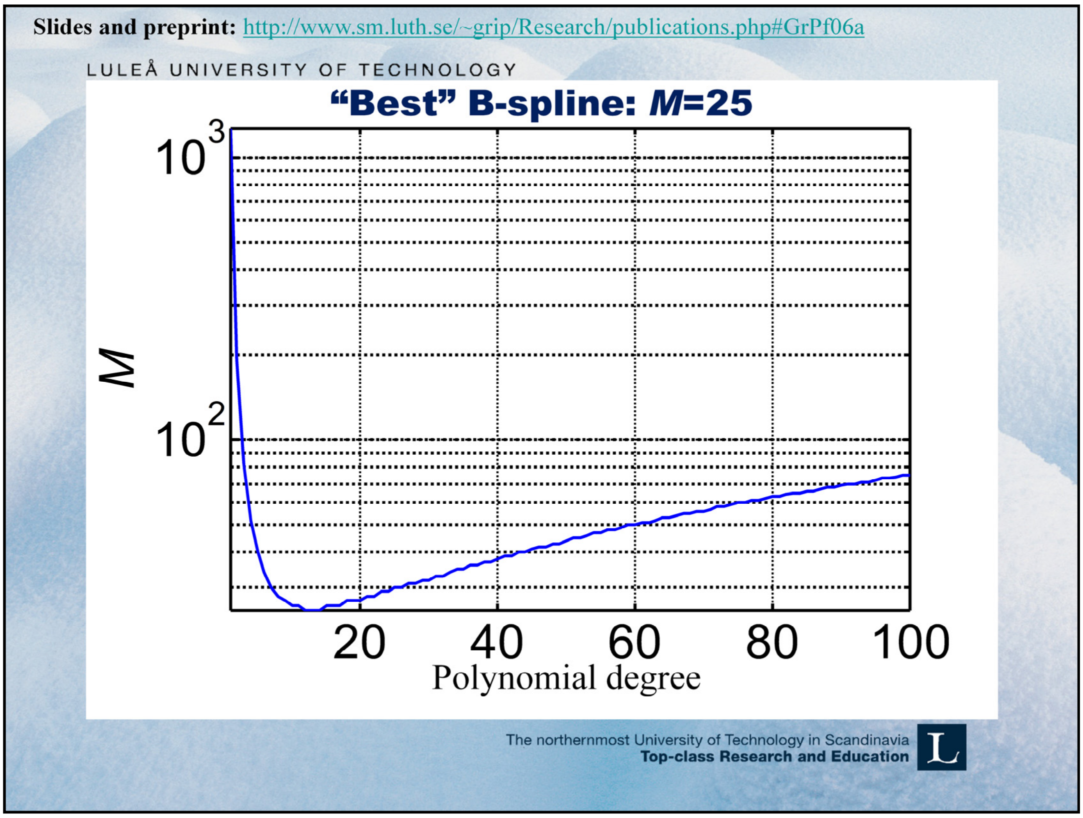
- What one perhaps could call ”common engineering practice” is to…
- However, a mathematically inclined engineer might say something like ”…oh no, you cannot do that …”
- Luckily for us, we do not have to worry about this, for reasons that I will not explain in full detail in next slide…



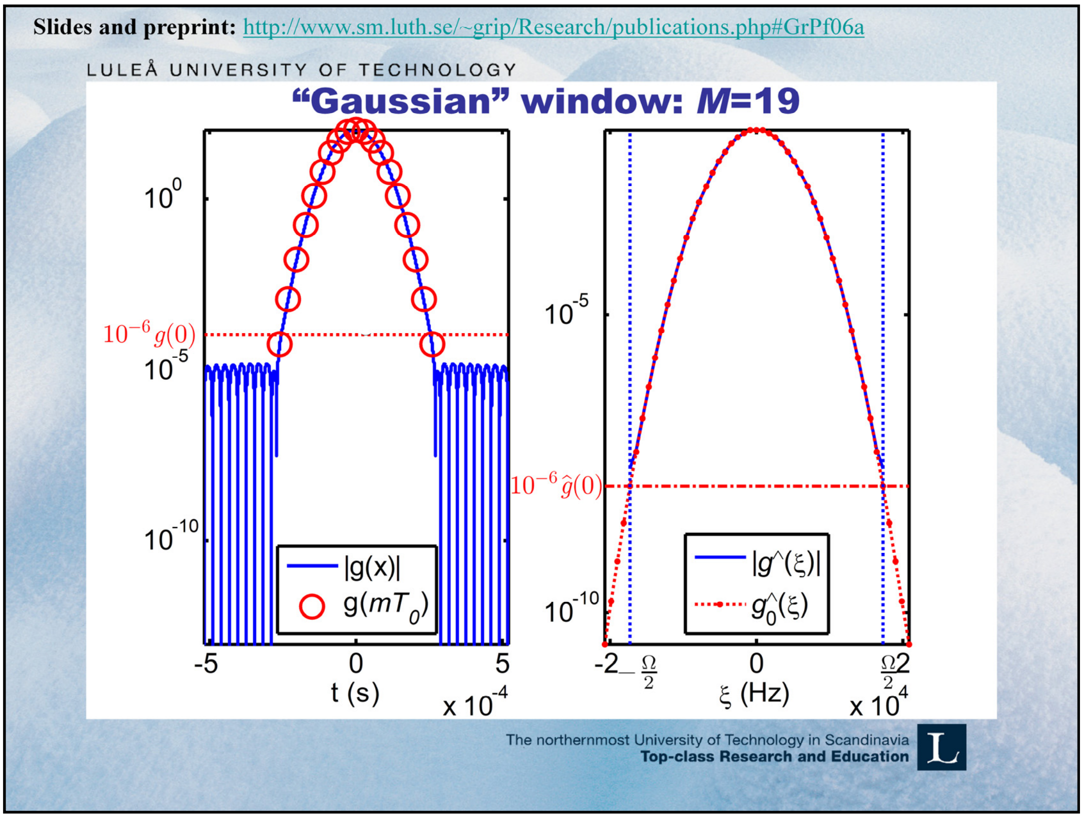
- …but in short, as long as we have well TF-localized basis functions the short-time behaviour of the channel is well modelled as a HS operator. The argument goes roughly as follows.
- We will adopt the philosophy of Matz, Schafhuber, Gröchenig, Hartmann and Hlawatsch who argues in a recent preprint for an IEEE journal that ***the single most important*** factor for obtaining low ISI and ICI is to use a window *g* with good time-frequency localization.
- They show also that if this is stated more formally, for example by requiring the short time Fourier transform to have subexponential decay, then the same property holds also for the dual basis window.
- Many of you do probably remember from some discrete time signal processing course that if a bandlimited signal is sent through a channel that is not bandlimited, the Fourier transform of the impulse response of the channel can be truncated to live on the same interval as the Fourier transform of the signals, thus giving an equivalent discrete time system for that family of input signals.
- Similarly here, from good TF-localization of the Gabor windows one can draw conclusions aboyt the impulse response, the spreading fuction and two of their partial Fourier transforms.
- From this we a Spreading function that is continuou and exponentially decaying in t and compactly supported but possibly containinng Dirac impulses in its nu-dependence.
- Thus we need one more observation, namely that we only are interested in the short-time behaviour of the channel. If the channel changes, for example when a mobile phone user drives into a tunnel, the standard approach is to use, pilote tones etc to make a new estimate of the channel.
- Thus we can multiply the impulse response with some function that equals 1 in the time interval of interest, decays smoothly to 0 and equals 0 outside some other intervals. The smooth decay can be done in such a way that the Fourier transform (the bifrequency function) have subexponential decay, which roughly is very close to but not exponential. For details go to the above URL and download the preprint.
- ***The important conclusion*** for this talk is that ***for well TF-localized Gabor windows (=pulseshaped OFDM symbols)***, we get an equivalent channel operator that ***in the time interval of interest*** has ***the same input-output relationships*** but whose **spreading function is infinitely many times differentiable, compactly supported in *t* and subexponentially decaying in *nu***, which from an engineering point of view basically means that it with only a negligible error can be smoothly truncated to being compactly supported and infinitely many times differentiable.
- For channels with exactly these properties, there is a natural way to apply the Nyquist sampling theorem and the Poisson summation formula …



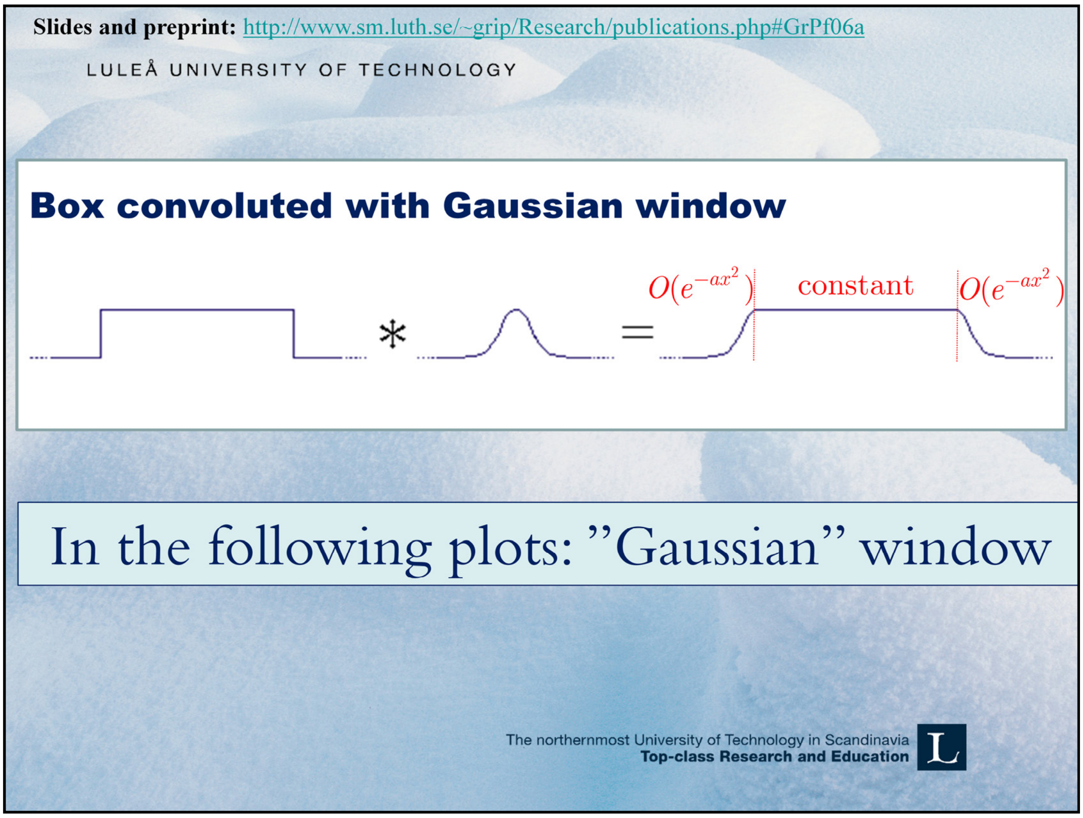
- …to obtain a formula for computing each channel matrix element from four nested sums, depending on some well-known system parameters and with the innermost sums contains some coefficients $S_{n,k}$, that are samples of a smoothly truncated and bandpass filtered copy of the real world spreading function.
- These coefficients can be chosen in two ways. ***One way*** is to compute them from measurements on a real channel or from a more detailed model of a particular channel. In the plots that I will show later we have instead made a ***random*** choice of coefficients under some side constraints that are given by maximum speeds of vehicles etc. (Our underlying thought is that if somebody really wants, (s)he can put a transmitter in a vehicle moving through a landscape with rocks, mountains, trains, planes etc placed and moving in such a way that the spreading function coefficients coincide with our randomly choice of coefficients. The choice of coefficients and the correspondence between coefficients and a corresponding setup of signal paths is also one point where we feel that more can be done.
- Again with Q carrier frequencies and R OFDM symbols, the difference in complexity is that with this approach, the factor Q^5 has changed to M^2Q^2, M being the number of nonzero Nyquist frequency samples of the OFDM pulses *g* and *gamma*.
- Hence we get a much faster algorithm if M is smaller than Q, which typically can be of size 1000 in radio communications. So how much smaller can M be?
- Well, in the already mentioned preprint by Matz, Schafhuber, Gröchenig, Hartmann and Hlawatsch, some B-spline and Gaussian pulse shapes were suggested. Of these,…



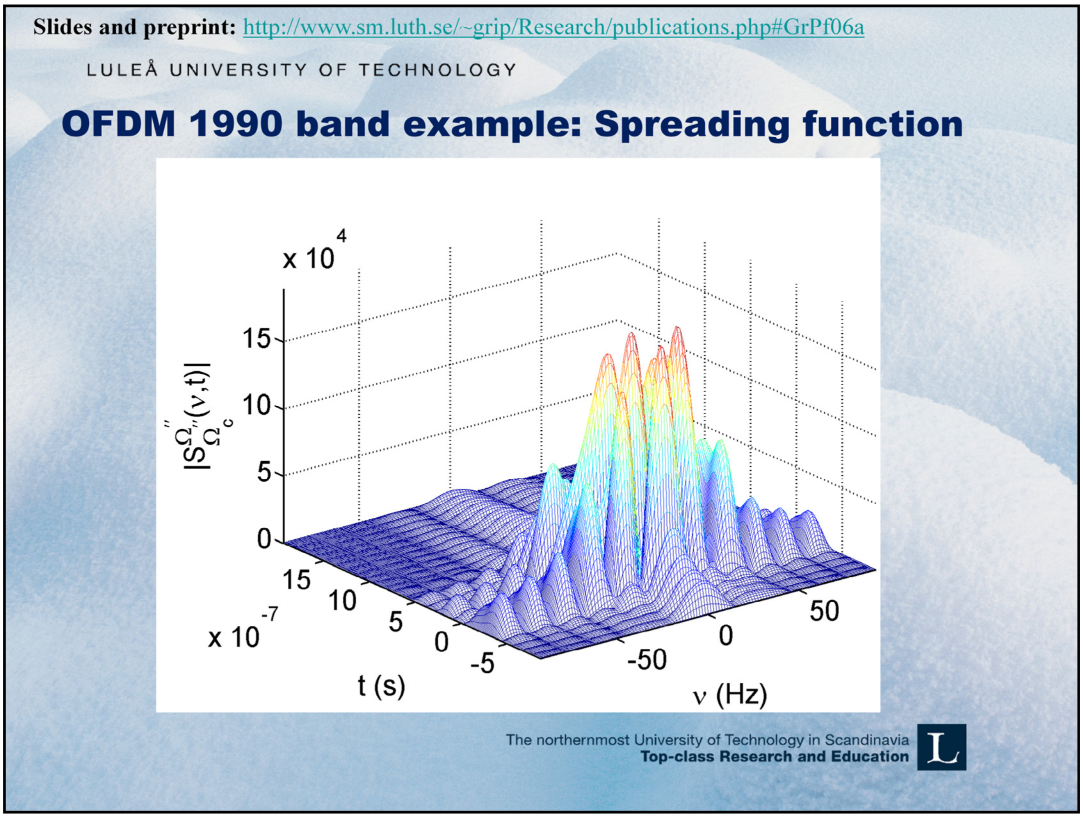
… the "best" B-spline windows has M=25, and for all Gaussian windows …



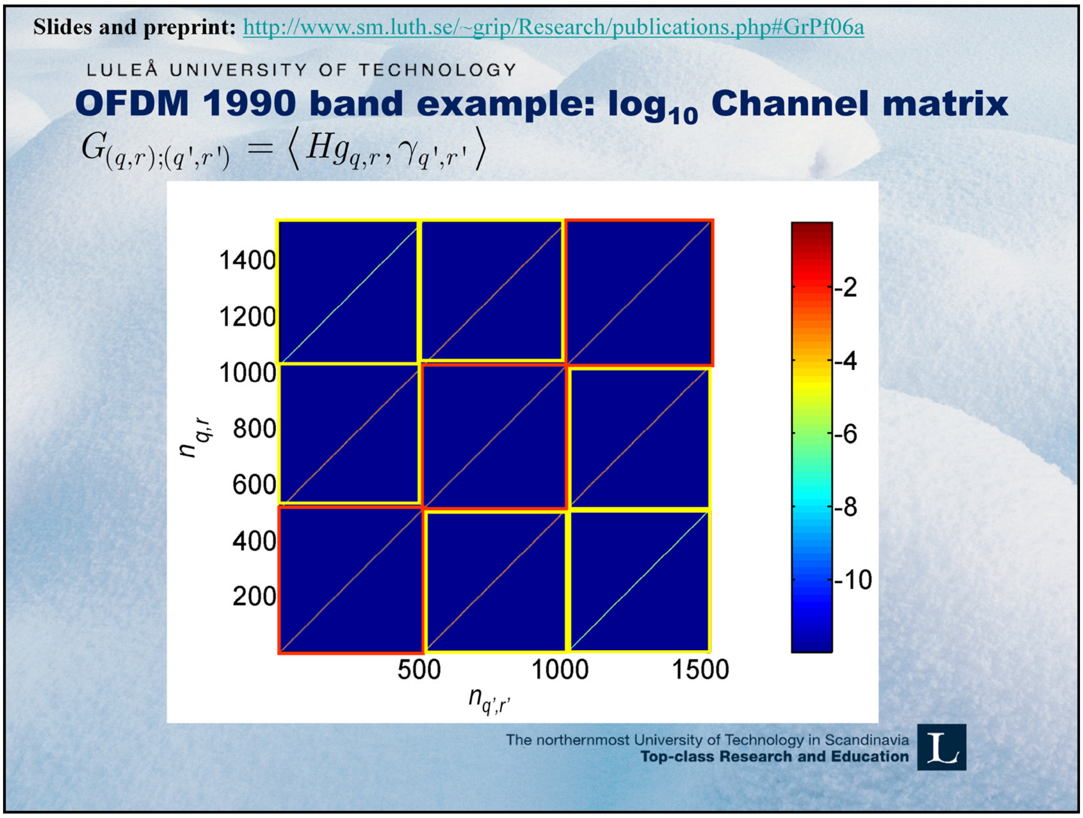
- …*M*=19, both of which are clearly a whole lot smaller than 1000.
- Now a Gaussian is not bandlimited, so *M*=19 holds for an ***approximate*** Gaussian which is obtained from a *true* Gaussian by truncating its (also Gaussian) Fourier transform at relative amplitude 10^{-6}, computing the corresponding Nyquist frequency samples of the Gaussian and truncating also those at relative amplitude 10^{-6}.
- The resulting approximate Gaussian and its Fourier transform are plotted in blue here. We see in the lefthand plot that it decays as a Gaussian to relative amplitude below 10^{-6} and decays slower after that below this threshold, but there with energy and amplitude well below the noise level of typical applications.



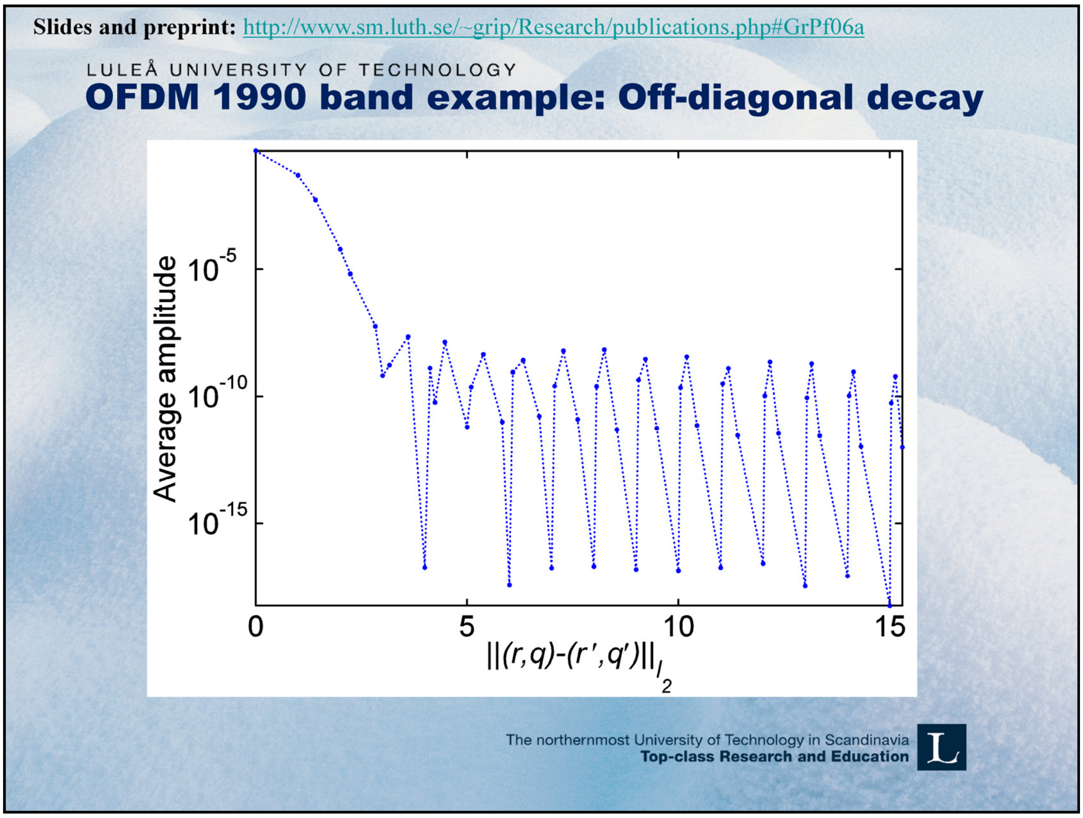
One can also, for example, convolute a box-shaped window with a Gaussian to obtain a window with is constant in some interval and then Gaussian decaying outside. For example for use in VDSL2. In the following slides, however, we'll use the just described approximate Gaussian.



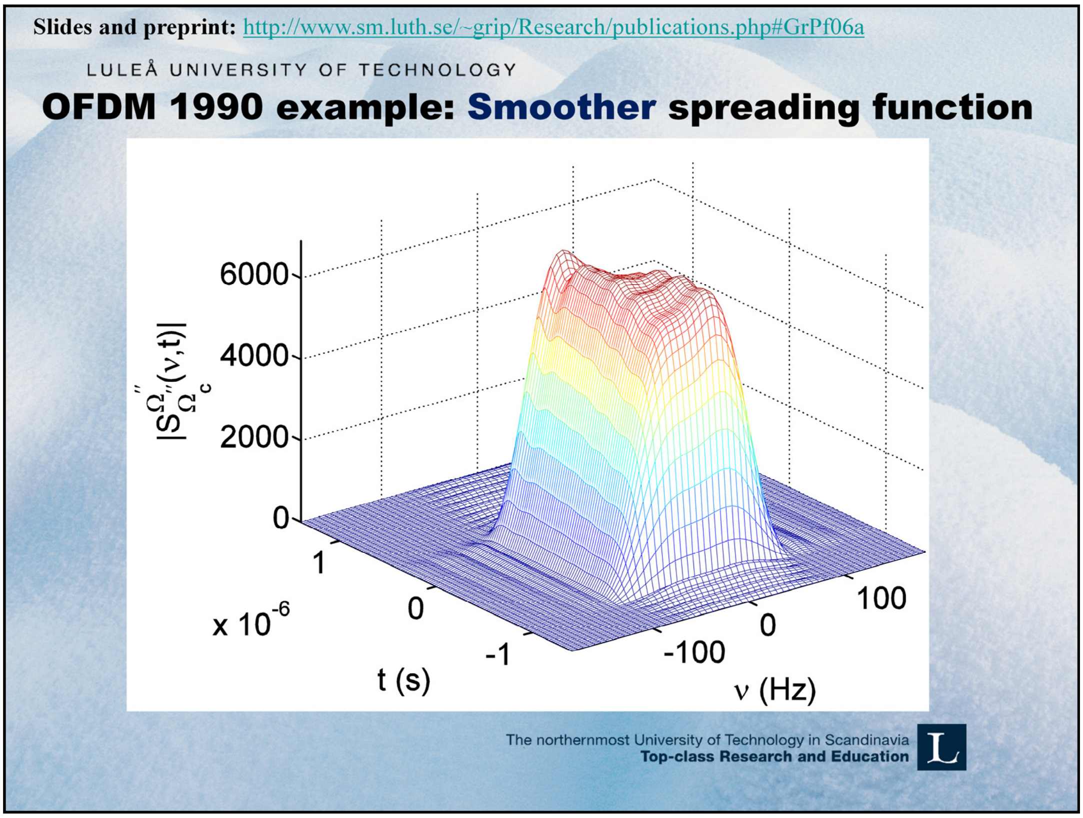
I end the talk with some example plots from a Matlab implementation of our algorithm.   If we choose Gaussian distributed and uncorrelated spreading function coefficients with different system parameters set to values typical for the OFDM 1990 band, then the resulting spreading function looks like this.



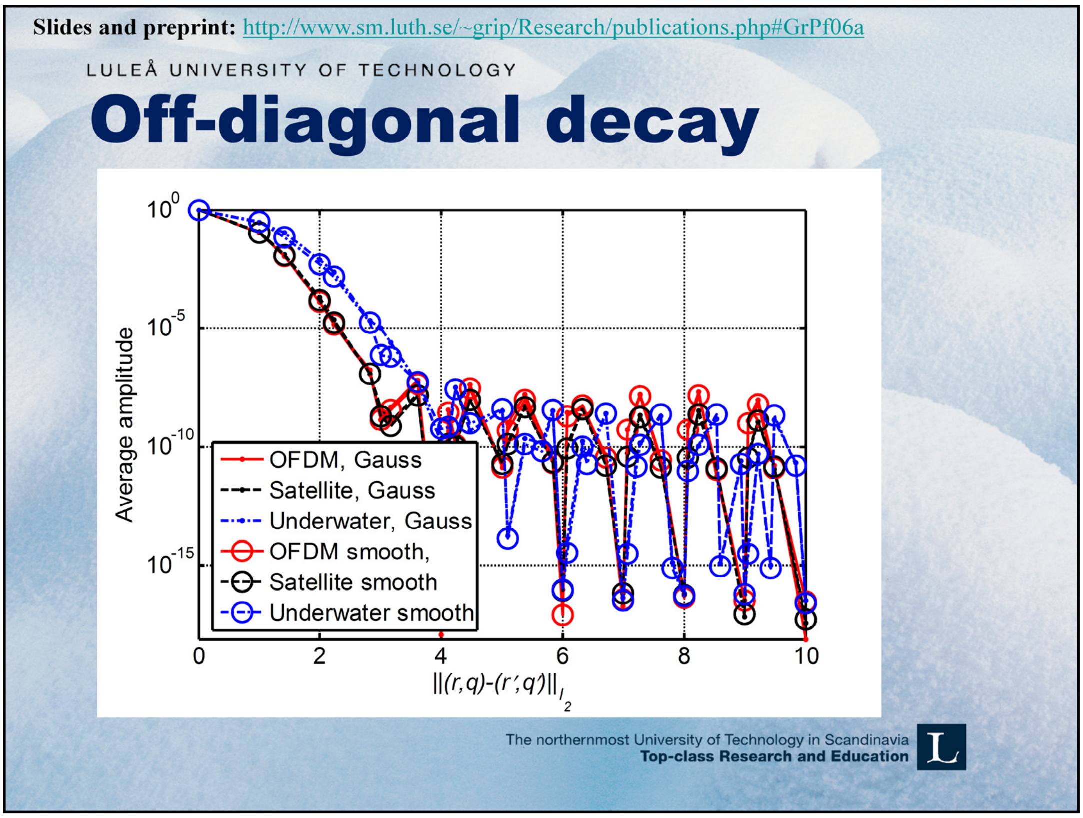
- This is the resulting channel matrix, computed for three OFDM symbols with 512 carrier frequencies each, which took 40 minutes to compute on this laptop.
- Since the matrix has two-dimensional indices, we have ordered them symbol-wise, so that the indices for the first OFDM symbol comes first, and so on.
- Thus the ICI of receivesymbol one is *this* submatrix (=the lowermost and leftmost one)...
- …and similarly for symbol 2 and 3 (=the other two submatrices marked with red borders).
- The ISI of transmission symbol 2 acting on receiver symbol 1 is this submatrix…
- …and similarly for the others.
- One drawback of this kind of plots is that inevitable consequence of this ordering of the two-dimensional indices is that there many elements that are close to the diagonal in the actual matrix ***not*** are close to the diagonal of this reordered and plotted matrix. For example, the center point in this upper left submatrix has distance one from the diagonal, because (q,r) equals (q',r'+2), but it is much more far away from the diagonal of this plotted matrix.
- One countermove is to collect and compute the average of all points that are at the same distence from the main diagonal…



…as is done in this plot- This gives the off-diagonal decay, where you clearly can see both the decay and the effects of having an approximate Gaussian window (=OFDM pulseshape) that is truncated at relative amplitude 10^{-6}.



We have repeated this with system parameters set for some different channels and with both uncorrelated Gaussian Spreading function coefficients, as in the previous plots, and with more correlated coefficients as in this plot of the resulting spreading function.



The corresponding off-diagonal decays show that for the given setup, the optimal (=Gaussian) time-frequency localization of the windows (=OFDM pulseshapes) give similar ISI and ICI for the OFDM and satellite communications setting, whereas the more complicated troublesome underwater communications channel give a much higher ISI and ICI.

And th-th-th-th-**that's all folks**! ...