comment concerning the use of the Vandermonde

The objection to the use of the Vandermonde matrix, whether in interpolation or in least squares approximation, is not that it may have a bad condition number but, rather, that it is the result of the use of the power basis, and this basis can be badly conditioned, even in the quadratic or cubic case, depending on the norm being used or, in more naive terms, depending on just where the interval of interest lies.

Hence, even if an accurate solution of the linear system with Vandermonde coefficient matrix is obtained, its further use, as coefficients in the power form of a polynomial, is doubtful because that power basis can be badly conditioned.

To the extent that the interval $[a \,.\,. b]$ of interest is 'far' from the origin, the power basis is ill-conditioned, with the condition number going to infinity as the ratio a/b approaches 1. Correspondingly, the Vandermonde matrix involving points from that interval is correspondingly badly conditioned.

However, the Vandermonde matrix can also be badly conditioned because of the particular pointset used, even if the power basis is not ill-conditioned on the the interval containing the points. For example, if two points approach each other while other points stay fixed, then, in the limit, the Vandermonde matrix is singular.

Specifically, consider the use of the power basis in polynomial interpolation, at the points x_i, $i = 0, \ldots, k$, to data (y_i). Solving the corresponding linear system

$$Vc = y$$

with

$$V := (x_i^{k-j} : i, j = 0, \ldots, k),$$

by Gauss elimination with partial pivoting, we can be sure that the computed coefficient vector, c, satisfies the linear system as well as possible, i.e., the residual will be of the size of roundoff incurred during calculation of Vc. Hence, for evaluation, near the interpolation points, of the power form of the resulting interpolating polynomial, we expect errors no worse than that.

However, this can be quite bad if there is loss of significance during the calculation of Vc, i.e., if the quantities involved are much larger than the resulting polynomial value. While such cancellation is unavoidable at a zero, it may happen for all points of an interval if the power basis is badly conditioned on that interval, as is the case for the power basis on the interval $[1000 \,.\,. 10001]$ and the polynomial

$$p(x) = (x - z_1)(x - z_2)(x - z_3)$$

with $z_i := 1000 + i/4$, $i = 1, 2, 3$. In this case, p is of size `1e-1` on that interval, while the power coefficients of p are, approximately, $[1,$ `-3e+3`, `3e+6`, `-1e+9`$]$, and these will be multiplied by numbers of sizes $[($`1e+3`$)^3, ($`1e+3`$)^2,$ `1e+3`$, 1]$, hence the summands in Vc all are of size `1e+9`, thus leading to a residual error of `1e-7` in matlab where approximately 16 decimal digits are carried. In other words, this is a relative error in p of size `1e-6` even if the coefficients are computed to full accuracy in this arithmetic.

However, this is not all. For, this error analysis assumes that we do carry those polynomial coefficients to all the digits matlab carries, and carry out the evaluation in the double precision environment of matlab. If, e.g., we merely record the single precision version of the coefficients, i.e., retain only their first eight significant digits (after all, who wants to copy all those digits!), we may totally change our polynomial on the interval of interest. For the resulting change in the computed polynomial may be of the order $10 =$ (`1e+9`)(`1e-8`) since each of the summands in Vc may be changed by that much, while the polynomial itself is of size `1e-1` there.

C. de Boor