GESELLSCHAFT FÜR ANGEWANDTE MATHEMATIK UND MECHANIK (GAMM)

Resolution on Computer Arithmetic

The elementary floating-point operations +, -, *, / in electronic computers are currently required to be of highest machine accuracy: For any choice of operands, the computed result must coincide with the rounded exact result of the operation, rounded according to the rounding mode in use (if no overflow occurs). For reference, see the IEEE Arithmetic Standards 754 (binary floating-point arithmetic) and 854 (general floating-point arithmetic).

In recent years there has been a significant shift of numerical computation from general-purpose computers towards vector and parallel computers - so-called supercomputers. Along with the 4 elementary operations +, -, *, /, these computers usually offer compound operations as additional elementary operations. This leads to an increase of several orders of magnitude in computing power. Some of these elementary compound operations are:

- **multiply and add:** a * b + c
- **multiply and subtract:** a * b - c
- **accumulate:** computes the sum of the components of a vector
- **multiply and accumulate:** computes the inner (or scalar) product of two vectors

and others.

GAMM requires that all elementary compound operations be implemented by the manufacturer in such a way that guaranteed bounds are delivered for the deviation of the floating-point result from the exact result. It is desirable and usually achievable that for all possible data the computed result of such a compound floating-point operation agrees with the result that would be obtained if the exact result were computed and then rounded by the rounding in use (if no overflow occurs). In this case no explicit error bounds need be delivered. The user should not be obliged to perform an error analysis every time an elementary compound operation, predefined by the manufacturer, is employed.

All elementary compound operations should also be provided with directed roundings, a feature needed both for fast computation of reliable and narrow bounds in numerical algorithms and for verification of the correctness of computed results. It must be ensured that the final floating-point result can differ from the exact result only in the direction defined by the rounding in use. This is already required of the elementary floating-point operations by the arithmetic standards mentioned above.