The code `fhbvm2`

This code is described in [1-4], and is available at the URL [5]. It is an improvement of the code `fhbvm` (we refer to the description of this latter code, for further details), in that it combines a possible initial graded mesh,

$$\hat{t}_0 = 0, \quad \hat{t}_i = \hat{t}_{i-1} + h_i, \quad h_i = r^{i-1}h_1, \quad i = 1,\dots,\nu, \qquad h_1\frac{r^\nu - 1}{r-1} = nh \equiv n\frac{T}{N},$$

for a suitable $n \in \{1,\dots,N\}$, with a subsequent uniform one,

$$t_j = jh \equiv j\frac{T}{N}, \qquad j = n,\dots,N.$$

Clearly, when $\nu > 1$ and $n = N$, the mesh reduces to a purely graded one, whereas when $\nu = n = 1$ it becomes a purely uniform one. For the graded mesh, the parameter r is fixed as follows [4]:

$$r = \begin{cases} 2, & \text{if} \quad n = 1, \\ \frac{n}{n-1}, & \text{if} \quad n \geq 2. \end{cases}$$

Consequently, by increasing n, the parameter r of the graded mesh becomes smaller and smaller, though always greater than 1. The motivation for using a double mesh, is to cope with the possible nonsmoothness of the vector field at the initial point, with a subsequent oscillatory solution. In this respect, we notice that from the equations defining the initial graded mesh one obtains that, for a given value of n, in order to reduce the initial timestep h_1, one needs to increase the value of ν.

The code `fhbvm2` implements a FHBVM(22,22) method, i.e., a FHBVM(k,s) method with $k = s = 22$ [1]. Consequently, as for the code `fhbvm`, a spectrally accurate solution can be expected. Moreover, it can be proved that, when $k = s$, in each sub-interval the method becomes a collocation method [4]. The calling sequence of the code is:

```
[t,y,etim] = fhbvm2( fun, y0, T, N, n, nu )
```

In input:

- `fun` is the identifier of a function computing:
  - the vector field (`fun(t,y)`) also in vector mode, returning row vectors;
  - the Jacobian (`fun(t,y,1)`);
  - the order α of the fractional derivative, if called without arguments;
- `y0` is a $\lceil\alpha\rceil \times m$ matrix containing the initial conditions, m being the dimension of the problem;
- `T` is the width of the integration interval (the initial time being set at 0);
- `N,n,nu` are the parameters used for the two meshes.

In output:

- `t,y` contain the computed solution (`y` is stored by rows);
- `etim` is an optional parameter containing the elapsed time.

[1] Brugnano, L.; Burrage, K.; Burrage, P.; Iavernaro F. A spectrally accurate step-by-step method for the numerical solution of fractional differential equations. *J. Sci. Comput.* **2024**, *99*, 48. https://doi.org/10.1007/s10915-024-02517-1

[2] Brugnano, L.; Gurioli, G.; Iavernaro, F. Numerical solution of FDE-IVPs by using Fractional HBVMs: the fhbvm code. *Numer. Algoritms* **2025**, *99*, 463–489. https://doi.org/10.1007/s11075-024-01884-y

[3] Brugnano, L.; Gurioli, G.; Iavernaro, F. Solving FDE-IVPs by using Fractional HBVMs: Some experiments with the fhbvm code. *J. Comput. Methods Sci. Eng.* **2025**. https://doi.org/10.1177/14727978251321328

[4] Brugnano, L.; Gurioli, G.; Iavernaro, F.; Vikerpuur, M. Analysis and implementation of collocation methods for fractional differential equations. *arXiv:2503.17719 [math.NA]* **2025**. https://doi.org/10.48550/arXiv.2503.17719

[5] https://people.dimai.unifi.it/brugnano/fhbvm/ (accessed on April 10, 2025).