Brackets	Root	Function	# Itr	Root	Function	# Itr
[1.5, 2.5]	None (Err)	None (Err)	None (Err)	None (Err)	None (Err)	None (Err)
[0.0, 1.5]	1.0027348	-4.0853e-8	100000	1.0000…2	0.0	60

Regula Falsi seems to be able to produce fairly accurate roots if the number of iterations is sufficiently high. However, there is an inherent limitation on the method's accuracy depending on the function in question and the bracket, not dependent on the number of iterations being used at a certain point. There is a diminishing return on increase in accuracy as iterations increase past a certain point.

For instance, with the first test function provided with 100000 interactions the Regula Falsi method produces a root of 1.0027348164705234 and a function value at that point of -4.08526563688838e-08. The table below shows how these numbers change with different interaction counts

Number of iterations	Estimated root	Calculated Func value at root
60	1.1039481928993016	-0.00212961467456374
100	1.0820404405855206	-0.001059066907171058
1000	1.0269775732096382	-3.8738309185326614e-05
10000	1.008621484090945	-1.2761446690490175e-06
100000	1.0027348164705234	-4.08526563688838e-08
500000	1.0012239903529578	-3.6652050283692006e-09
1000000	1.0008656487495464	-1.2967822371479087e-09

As one can see, even with a million iterations on the $x^4 - 6x^3 + 12x^2 - 10x + 3$ function, the Regula Falsi method still doesn't converge on a perfectly accurate 1.0 root and value of 0.0 at that point. And for each additional step in iteration count we only get an additional decimal place in accuracy.