FWIW, the multiprocessing module has a nice interface for this using the Pool class. And if you want to stick with threads rather than processes, you can just use the multiprocessing.pool.ThreadPool class as a drop-in replacement.

def foo(bar, baz): print 'hello {0}'.format(bar) return 'foo' + baz from multiprocessing.pool import ThreadPool pool = ThreadPool(processes=1) async\_result = pool.apply\_async(foo, ('world', 'foo')) # tuple of args for foo # do some other stuff in the main process return\_val = async\_result.get() # get the return value from your function.

