Comparison

March 14, 2019

0.1 Comparison between Lambert Beers Law and Transfer Matrix absorption and their effects in a two temperature model

In this file the absorption calculated with an exponential decay, according to Lambert Beers law and the local absorption evaluated, using the Transfer Matrix Method are being compared within the framework of a two temperature model calculation originally proposed by Anisimov et al. but more recently used by J. Hohlfeld et al. and D. Schick et al. among others.
The sample case under consideration is :

Layer	length	refractive index	optical penetration depth
Platinum	3nm layer	n = 1.0433 + i3.0855	18 nm
Cobalt	15nm layer	n = 1.0454 + i3.2169	10 nm
Chromium	5nm layer	n = 2.0150 + i2.8488	10 nm
Magnesium Oxide	inf- layer	n = 1.7660	inf

The procedure of this session is to: * Import all the packages needed available at LocalAbsorptionand TTMObject to run the code * Run a local absorption computation to obtain the local absorption profile of the layer (This is just done for reference, in fact the local absorption module is also implemented in the TMMObject package.)

- Run different simulations with the TTMObject, changing the input source from a Lambert Beer description to a Transfer Matrix description of the absorption.

Dependecies: Numpy; Matplotlib B-splines; Progressbar; Numerical Units

```
In [1]: import TMM_abs as atmm      #Import absorption file (The algorithm here is in principle
        from TTMObject import *     #Implement the 2temperature object
        import numericalunits as u #Numerical units to show physcal dimensions. (In principle
        u.reset_units('SI')

In [2]: #Define the complex refractive index for every layer and the length of every layer
        #layers:  Air   Pt          Co                  Cr          MgO
        n_list = [1, 1.0433+3.0855j,1.0454+3.2169j,2.0150+2.8488j,1.766]
        d_list = [np.inf,3,15,5 ,np.inf] #in nm
        th0    = np.pi/4 #in rad. (0 is perpendicular to the surface)
        lam0   = 400     # wavelength in vacuum
        pol    = 'p'     #polarization
```

Based on the suggestions made by Steven J. Byrnes the total absorption, reflection and transmission A, R, T are calculated based on the idea of transfer matrices, where the refractive index, the incident angle and the length of every layer are taken into consideration. The parameters A, R, T are normalized with respect to the incident power.

The results obtained have been compared with Steven J. Byrnes- Sample 4 and to results obtained by calculations, executed via COMSOL Multiphysics, where we find that the simulation done in 1D here matches almost exactly with a 2D simulation done with COMSOL Multiphysics where each layer attributes were extended homogeneously on to one more degree of freedom. Hence our 1D model is a good approximation even to a higher dimensional case.

```
In [3]: plotpoints                  = 500
        [M,M_n,t,r,T,R,A,theta]     = atmm.TM(th0,lam0,n_list,d_list,pol)
        [absorp,grid]               = atmm.absorption(th0,lam0,n_list,d_list,pol,plotpoints)
```

```
In [4]: print(f"Total transmission at the end {T:.2}\n"\
              f"Total reflectance at the first layer {R:.2}\n"\
              f"Total Absortpin at the entire material {A:.2}")
```

```
Total transmission at the end 0.14
Total reflectance at the first layer 0.51
Total Absortpin at the entire material 0.36
```

```
In [5]: #--> Result obtained by calling the function absorption()
        plt.figure()
        plt.suptitle('Local absorbtion profile of multi layer thin film', fontsize=12)
        plt.title(r"$\lambda=400$nm, $\theta_0=45^ř$, polarization =$p$, total Absorbtion = {:
                  .format(np.round(A,2)),fontsize=10)
        plt.xlabel("Distance from surface (nm)");
        plt.ylabel(r"Absorbtion per unit incident power $\left(\frac{1}{nm}\right)$")
        plt.plot(grid,absorp)
        plt.show()
```