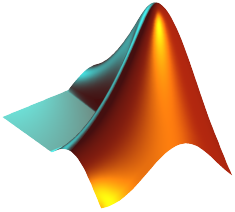
INSTITUTE OF SCIENTIFIC INSTRUMENTS
The Czech Academy of Sciences

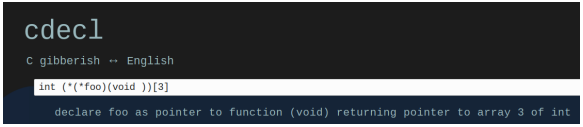
Ladění výpočetní knihovny napsané v jazyce C prostřednictvím MATLABu

Martin Šiler

Akademie Věd ČR, Ústav přístrojové techniky
Královopolská 147, 612 64 Brno

Proč?





`malloc(), ..., free()`

Proč:

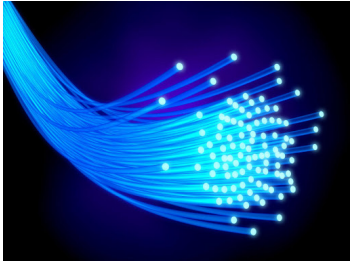
- Překonat omezení
- Slabý hardware (Arduino), real-time, složité a náročné
- CUDA - explicitně

Cena:

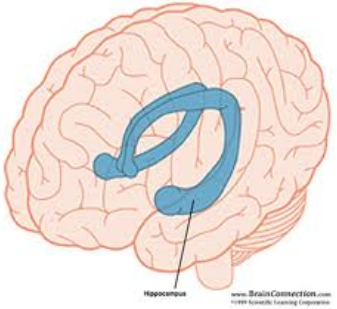
- Složitost
- Práce s pamětí
- Debugování a ověření správnosti

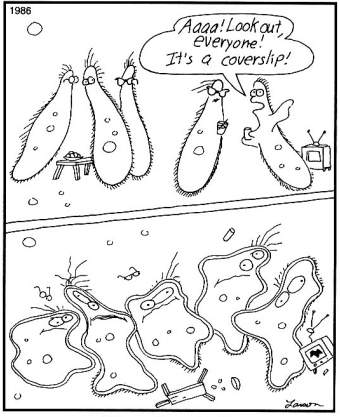
Náš problém:

Zobrazovaní pomocí optických vláken



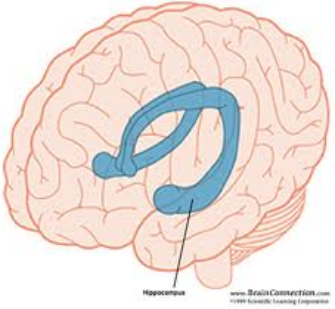
Co se děje uvnitř živého organismu?



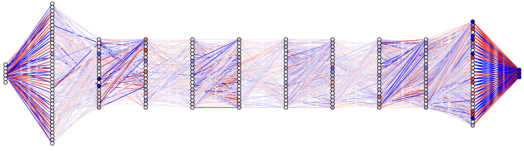


Life on a microscope slide

Co se děje uvnitř živého organismu?

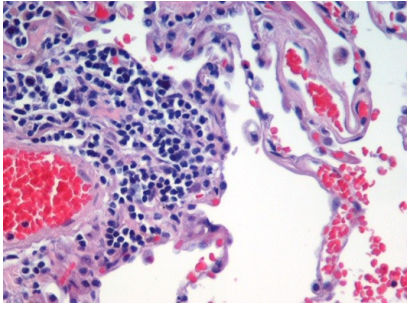


Jak funguje skutečná neuronová síť?

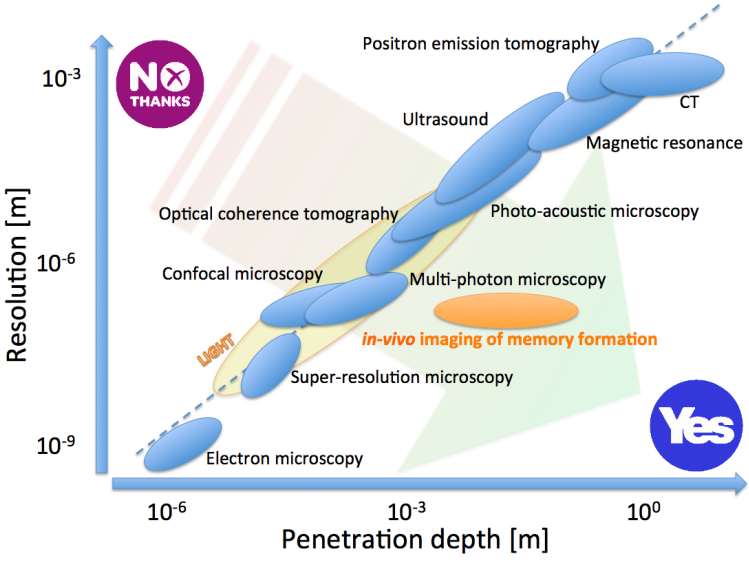


Co se děje uvnitř? Zajímavé je schované

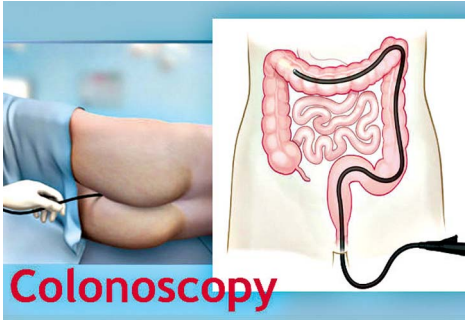
Co se děje uvnitř? Zajímavé je schované

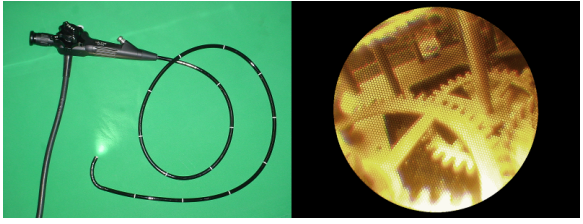


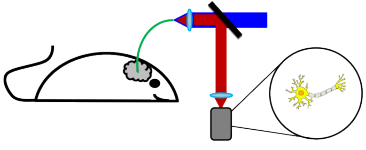
Rozlišení × Jak hluboko zobrazujeme



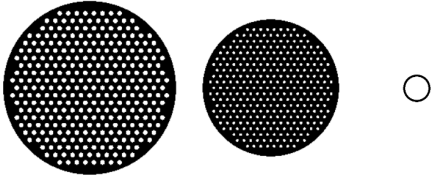
Endoskopy



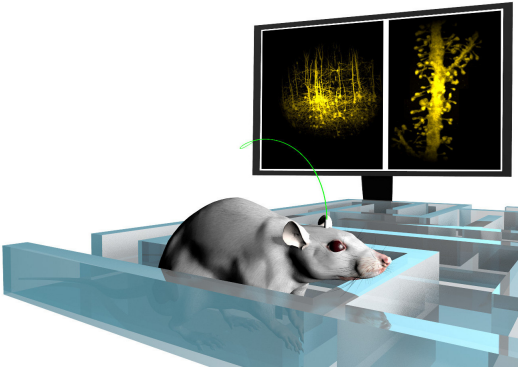


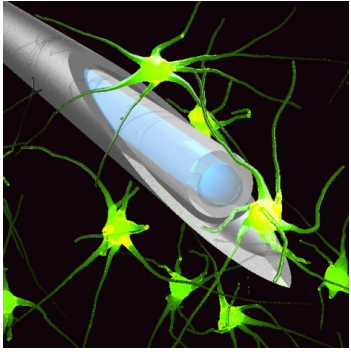


a standard image guide b single–mode image guide c multi–mode fibre

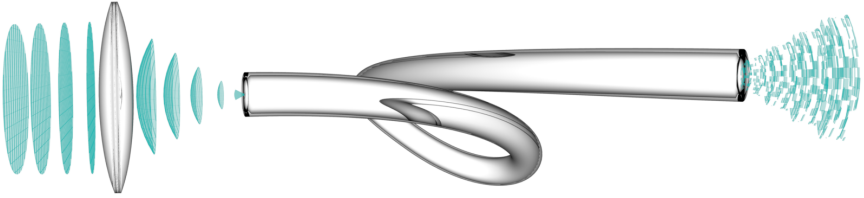


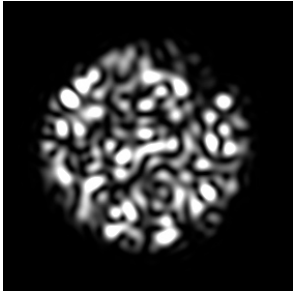
Endoskopy



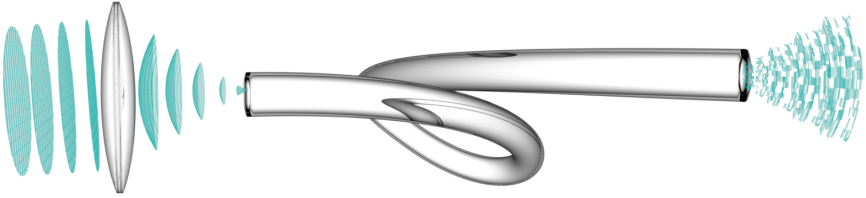


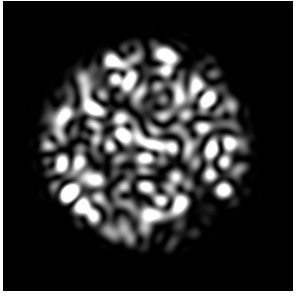
Multimodové vlákno a vznik “*obrazu*”?





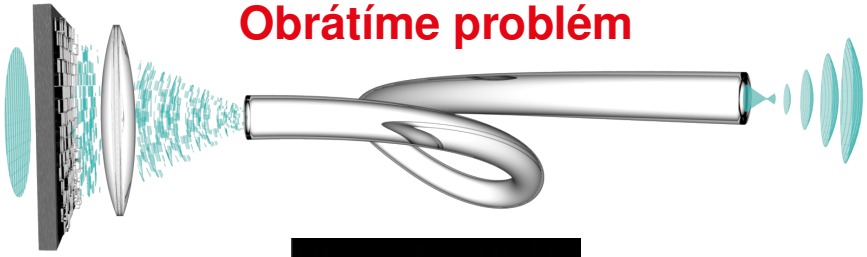
Multimodové vlákno a vznik “*obrazu*”?

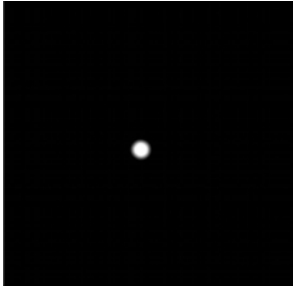


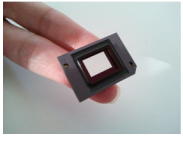


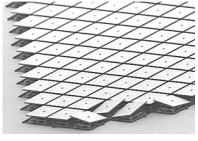
+ Interference

Obrátíme problém

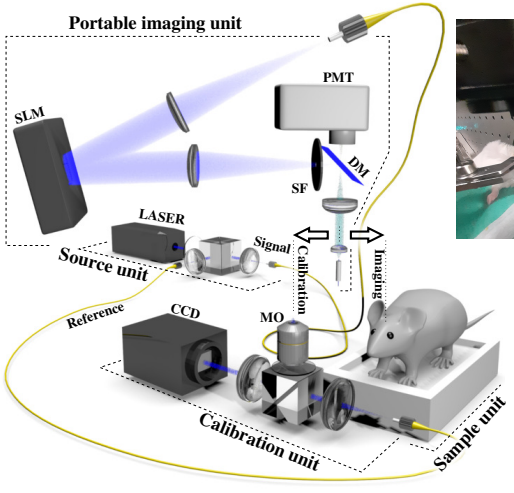


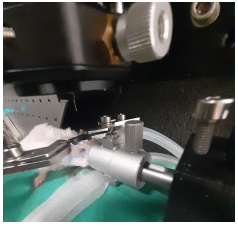


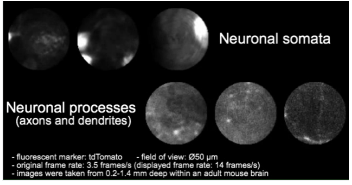


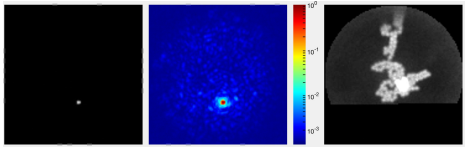


Experimentální systém

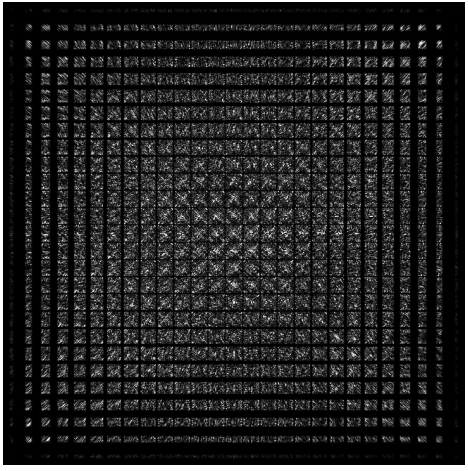


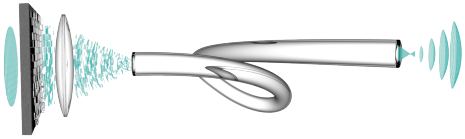






Transmisní matice optického vlákna

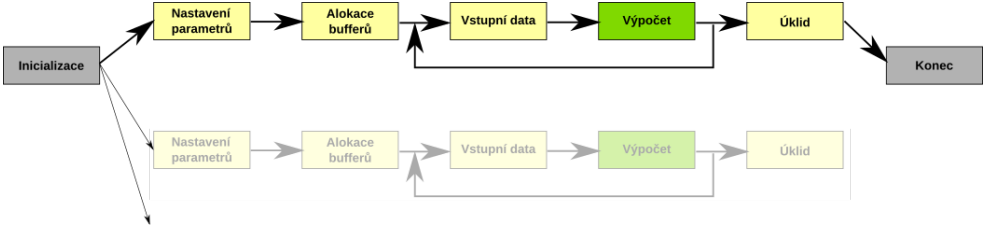




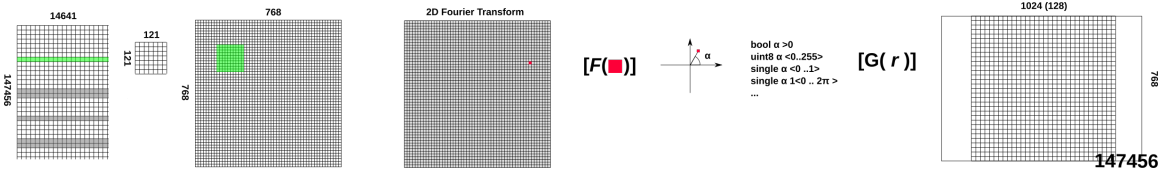
- Propojuje vstup a výstup vlákna
- $181^2 \times 352^2$ `complex float` $\Rightarrow$ 32.5 GB
- vytváří mřížky pro prostorové modulátory
- naměřeno za 1.5 minuty
- zpracování LabVIEW $\sim$ 45 minut

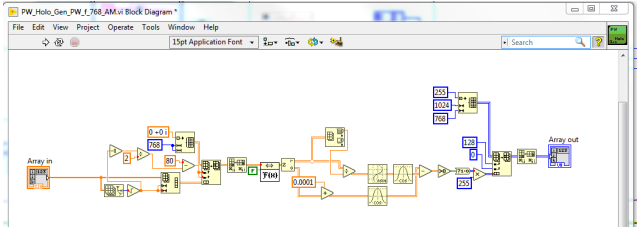
Knihovna

Výpočetní knihovna: flowchart

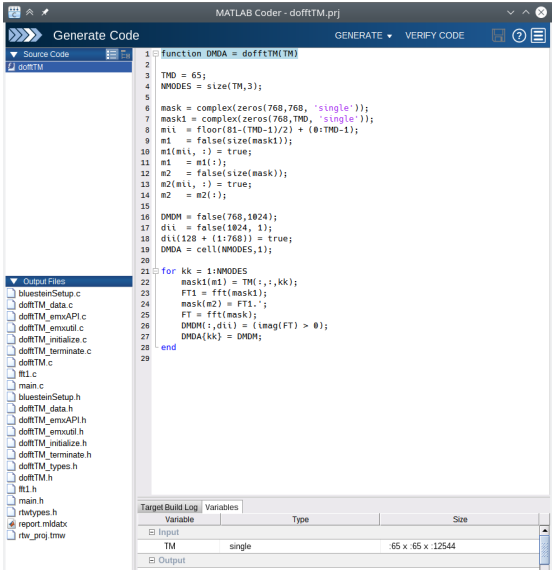


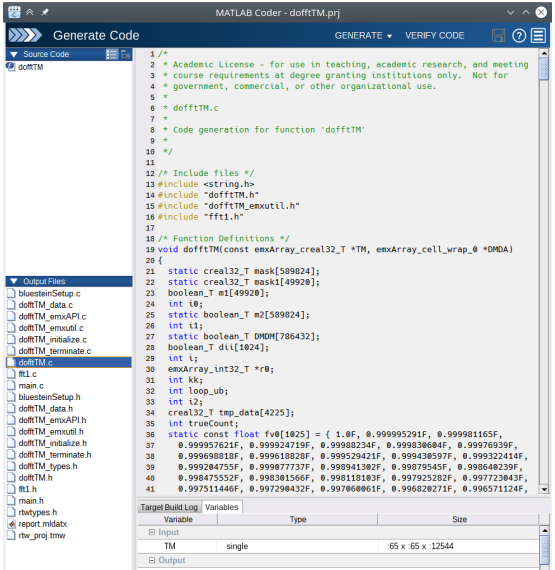
Obsah výpočtu



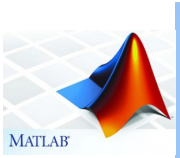


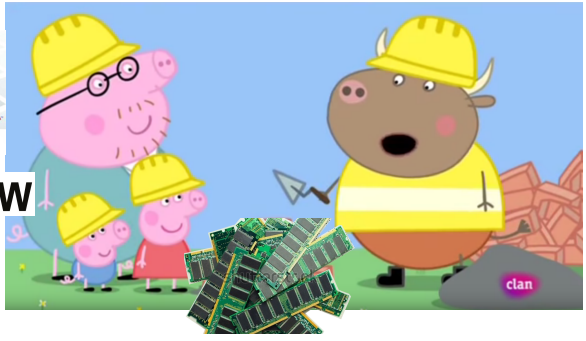
MATLAB Coder





Paměť a alokace



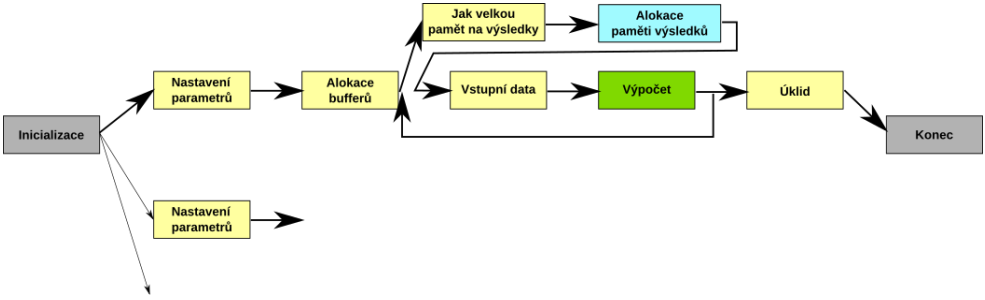


Nadřízené prostředí
vlastní správa paměti
neví kolik má alokovat

×

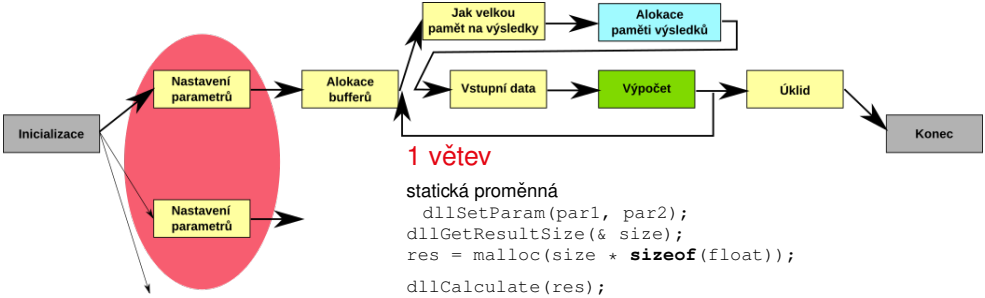
Knihovna
snadno alokuje
kdy a jak uvolnit?

Paměť a alokace II



Paměť a alokace III

Kam uložit parametry



1 větev

statická proměnná

```
  dllSetParam(par1, par2);
dllGetResultSize(& size);
res = malloc(size * sizeof(float));

dllCalculate(res);
```

Paramerty ve vnějším bufferu

alokuje vnější prostředí

```
  size_t parsize = dllParSize();
DLLPAR * params = malloc(parsize);
dllSetParam(params, par1, par2);
dllGetResultSize(params, & size);
res = malloc(size * sizeof(float));

dllCalculate(params, res);
```

Vnitřní správa

knihovna se stará sama

```
  int32_t ID = dllAllocCalculations();
dllSetParam(ID, par1, par2);
dllGetResultSize(ID, & size);
res = malloc(size * sizeof(float));

dllCalculate(ID, res);
```

Rozhraní

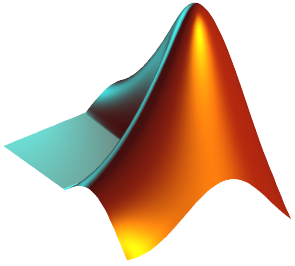
```c
#include <stdint.h>
#include <stddef.h>
#include <fftw3.h>

#include "dll_api.h"
#include "DMDtoolbox.h"

typedef struct {
    int32_t         initialized;   /**< flag showing that structure has been initialize */
    size_t          in_modes;      /**< size of a side of square defining grid of input modes */
    size_t          in_modes2;     /**<  @p INPUTMODES squared (total number of input modes, number of columns of transmission matrix) */
    size_t          out_modes;
    size_t          out_modes2;
    size_t          cout_modes;
    size_t          cout_modes2;

    float           pixel_pitch;
    float           wavelength;

    int32_t         threads;

    float           *gamma;
    int             has_plans;
    fftwf_plan      plan_fw, plan_bk;
    int32_t         proj_center[2];

} tmmtPAR_t;


/*** ERROR code for not enough memory*/
#define TMMTERROR_MEMORY -11
/*** ERROR code for uninitialized parameters */
#define TMMTERROR_INIT -12
/*** ERROR propagation matrix which should be complex */
#define TMMTERROR_PROPAGATION -13


#ifdef __cplusplus
extern "C"
{
#endif

ADDAPI int32_t ADDCALL tmmtGetSettingsSize(void);
ADDAPI void    ADDCALL tmmtCleanSettings(tmmtPAR_t *settings);
ADDAPI int32_t ADDCALL tmmtSetParams(tmmtPAR_t *settings, int32_t input_modes1, int32_t output_modes1, int32_t thread_c);
ADDAPI int32_t ADDCALL tmmtSetPitchWavelenght(tmmtPAR_t *settings, float pitch, float wavelength);
ADDAPI int32_t ADDCALL tmmtGetParamsToArray(const tmmtPAR_t *settings, int32_t pars[7]);

ADDAPI int32_t ADDCALL tmmtCalcAveragedProjection(tmmtPAR_t *settings, const uint32_t N, const tmmtComplex * TM, float *avp_far, float *avp_dist);
ADDAPI int32_t ADDCALL tmmtCalcProjectionCenter(tmmtPAR_t *settings, const float *avp_far, int32_t *slice_rect);

ADDAPI int64_t ADDCALL tmmtCompressTMSize(tmmtPAR_t *settings, int32_t sz[4]);

ADDAPI int32_t ADDCALL tmmtCompressTM(tmmtPAR_t *settings, const tmmtComplex * TM, tmmtComplex *TMC);

ADDAPI int32_t ADDCALL tmmtFTcTM(tmmtPAR_t *settings, const tmmtComplex * TMC, tmmtComplex *TMCft);
ADDAPI int32_t ADDCALL tmmtFTcTMInplace(tmmtPAR_t *settings, tmmtComplex * TMC);
```

1. struktura s parametry
2. funkce pro zjištění velikosti
3. funkce pro nas tavení a čtení parametrů
4. výpočetní funkce

Testování a MATLAB

Integrace s MATLABem – `loadlibrary`

- standardní funkce
  `loadlibrary, calllib, unloadlibrary, libpointer`
- MATLAB automaticky vytvoří rozhraní mezi pro volání funkcí
  `loadlibrary('libDll.so', 'header.h')`
- Volání funkcí: `calllib`
  `[x1,...,xN] = calllib(libname,funcname,arg1,...,argN)`
- Pro návrat do polí nutné použít `libpointer`
- Netriviální příprava struktur
- Komplexní čísla?



Integrace s MATLABem – MEX

```c
#include "mex.h"
#include "matrix.h"

#include "TMManipulator.h"

#define FUNCN "dmdToolbox:tmmtGenRamp"

//[settings, ramp] = tmmtGenRamp(settings)
//int32_t tmmtGenRamp(tmmtPAR_t *settings,  tmmtComplex *ramp)
/* The gateway function */
void mexFunction( int nlhs, mxArray *plhs[],
                  int nrhs, const mxArray *prhs[])
{
    tmmtComplex* trans_matrix = NULL;


    size_t settings_size = tmmtGetSettingsSize();
    /* check for proper number of arguments */
    if ((nrhs != 1 ))
        mexErrMsgIdAndTxt(FUNCN ":prhs","# input parameters must be 1 ");
    if ((!mxIsUint8(prhs[0])) && ( mxGetNumberOfElements(prhs[0]) != settings_size))
        mexErrMsgIdAndTxt(FUNCN ":prhs","Input 1 is not TM manipulator settings");


    if ((nlhs>2)) {
        mexErrMsgIdAndTxt(FUNCN ":nlhs","0-2 output required.");
    }


    tmmtPAR_t * settings = (tmmtPAR_t *) mxGetUint8s(prhs[0]);

    int32_t param[20];
    tmmtGetParamsToArray(settings, param);

    mwSize mdims[2];
    mdims[0] = param[2];
    mdims[1] = param[2];


    mxArray *mRamp;
    tmmtComplex *ramp;

    mRamp = mxCreateNumericArray(2, mdims, mxSINGLE_CLASS, mxCOMPLEX);
    ramp  = (tmmtComplex*) mxGetComplexSingles(mRamp);

    if (nlhs >= 2)
    {
        int32_t err = tmmtGenRamp(settings, ramp);

        plhs[1] = mRamp;
    }
    plhs[0] = mxDuplicateArray(prhs[0]);
}
```

Vlastní rozhraní, funkce pro MEX

1. C a MATLAB prototyp funkce
2. základní část chybové zprávy
3. kontrola vstupních parametrů (typ, velikost)
4. ukazatele na data vstupních polí (`mxGetUint8s`)
   pozor na komplexní data
5. vytvořit výstupní proměnné a ukazatele na data
6. zavolat funkci z knihovny
7. aktualizovat strukturu s parametry

- Pro překlad Makefile
- Linux: `LD_LIBRARY_PATH='.'  matlab&`

Integrace s MATLABem – třída

```
classdef TMManipulator < handle
    %UNTITLED Summary of this class goes here
    %   Detailed explanation goes here

    properties (Dependent = true)
        Error
        InputModes1
        InputModes2
        OutputModes1
        OutputModes2
        COutModes1
        COutModes2
        Threads
    end
    properties (Access = public)
        s        (1,:) uint8 = [];
        e        (1,1) double = 0;
    end
    properties (Access = protected)
        fIM      (1,1) int32 = 0;
        fIM2     (1,1) int32 = 0;
        fOM      (1,1) int32 = 0;
        fOM2     (1,1) int32 = 0;
        fCM      (1,1) int32 = 0;
        fCM2     (1,1) int32 = 0;
        fThreads (1,1) int32 = 0;
    end

    methods
        function obj = TMManipulator(im1, om1, threads)
            if nargin < 1 || isempty(im1)
                im1 = 0;
            end
            if nargin < 2 || isempty(om1)
                om1 = 0;
            end
            if nargin < 3 || isempty(threads)
                threads = 0;
            end

            [obj.s, obj.e] = tmmtSetParams(im1, om1, threads);
            obj.UpdateSettings;
        end

        function delete(obj)
            disp ('delete Manipuator');
            if ~isempty(obj.s)
                tmmtCleanSettings(obj.s);
                obj.s = [];
            end
        end

        function SetPitchWavelength(obj, pitch, wavelengt)
            [obj.s, obj.e] = tmmtSetPitchWavelenght(obj.s, pitch, wavelengt);
        end
```

- Všechny funkce zaobalené do třídy
- Destruktor pro úklid
- Parametry přístupné přes vlastnosti

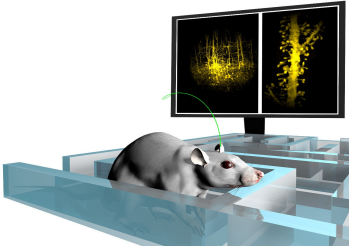
Poznámky:

- pro znovunačtení knihovny `clear functions`
- row/column major vícerozměrná pole
- indexování od 0/1
- `tic;toc`

Shrnutí

Zobrazování vláknem

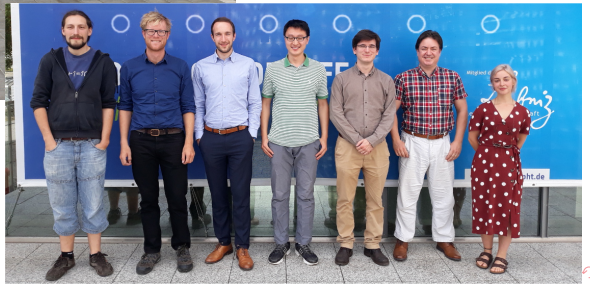
- Flourescencční zobrazování hluboko v tkáních
- LabVIEW $\Rightarrow$ C + CUDA
  alespoň 100 násobné zrychlení
- Další zobrazovací techniky:
  CARS, SGH
- Směřujeme k zobrazování pohybujících se modelů



MATLAB pro debugování

- srovnání výsledků výpočtu
  knihovna $\times$ MATLAB
- srovnání výkonnosti
- nutnost použít MEX (někdy)
- pravděpodobnost pádu MATLABu
  99% (ze začátku) $\Rightarrow$ 20 %

Tým



Reference

1. **Subcellular spatial resolution achieved for deep-brain imaging in vivo using a minimally invasive multimode fiber**
   By: Vasquez-Lopez, Sebastian A.; Turcotte, Raphael; Koren, Vadim; et al.
   LIGHT-SCIENCE & APPLICATIONS Volume: 7 Article Number: 110 Published: DEC 19 2018
   Free Full Text from Publisher View Abstract

2. **High-fidelity multimode fibre-based endoscopy for deep brain in vivo imaging**
   By: Turtaev, Sergey; Leite, Ivo T.; Altwegg-Boussac, Tristan; et al.
   LIGHT-SCIENCE & APPLICATIONS Volume: 7 Article Number: 92 Published: NOV 21 2018
   Free Full Text from Publisher View Abstract

3. **Robustness of Light-Transport Processes to Bending Deformations in Graded-Index Multimode Waveguides**
   By: Flaes, Dirk E. Boonzajer; Stopka, Jan; Turtaev, Sergey; et al.
   PHYSICAL REVIEW LETTERS Volume: 120 Issue: 23 Article Number: 233901 Published: JUN 6 2018
   Free Full Text from Publisher View Abstract

4. **Three-dimensional holographic optical manipulation through a high-numerical-aperture soft-glass multimode fibre**
   By: Leite, Ivo T.; Turtaev, Sergey; Jiang, Xin; et al.
   NATURE PHOTONICS Volume: 12 Issue: 1 Pages: 33-+ Published: JAN 2018
   Full Text from Publisher View Abstract

5. **Comparison of nematic liquid-crystal and DMD based spatial light modulation in complex photonics**
   By: Turtaev, Sergey; Leite, Ivo T.; Mitchell, Kevin J.; et al.
   OPTICS EXPRESS Volume: 25 Issue: 24 Pages: 29874-29884 Published: NOV 27 2017
   Free Full Text from Publisher View Abstract

6. **High-speed spatial control of the intensity, phase and polarisation of vector beams using a digital micro-mirror device**
   By: Mitchell, Kevin J.; Turtaev, Sergey; Padgett, Miles J.; et al.
   OPTICS EXPRESS Volume: 24 Issue: 25 Pages: 29270-29283 Published: DEC 12 2016
   Free Full Text from Publisher View Abstract

7. **Multimode fibre: Light-sheet microscopy at the tip of a needle**
   By: Ploeschner, Martin; Kollarova, Vera; Dostal, Zbynek; et al.
   SCIENTIFIC REPORTS Volume: 5 Article Number: 18050 Published: DEC 14 2015
   Free Full Text from Publisher View Abstract

8. **Seeing through chaos in multimode fibres**
   By: Ploeschner, Martin; Tyc, Tomas; Cizmar, Tomas
   NATURE PHOTONICS Volume: 9 Issue: 8 Pages: 529-+ Published: AUG 2015
   Full Text from Publisher View Abstract

9. **Exploiting multimode waveguides for pure fibre-based imaging**
   By: Cizmar, Tomas; Dholakia, Kishan
   NATURE COMMUNICATIONS Volume: 3 Article Number: 1027 Published: AUG 2012
   Free Full Text from Publisher View Abstract

10. **Shaping the light transmission through a multimode optical fibre: complex transformation analysis and applications in biophotonics**
    By: Cizmar, Tomas; Dholakia, Kishan
    OPTICS EXPRESS Volume: 19 Issue: 20 Pages: 18871-18884 Published: SEP 26 2011
    Free Full Text from Publisher View Abstract

11. **In situ wavefront correction and its application to micromanipulation**
    By: Cizmar, Tomas; Mazilu, Michael; Dholakia, Kishan
    NATURE PHOTONICS Volume: 4 Issue: 6 Pages: 388-394 Published: JUN 2010

Děkuji za pozornost

Tomáš Čižmár, . . . , Martin Šiler

www.isibrno.cz/complexphotonics

siler@isibrno.cz

The research was supported by European Structural and Investment Funds Project "Holographic endoscopy for in vivo applications" (No. CZ. 02.1.01/0.0/0.0/15_003/0000476) and MEYS CR (LO1212).