Module Interface Specification for STEM Moiré GPA

Alexandre Pofelski
macid: pofelska
github: slimpotatoes

November 17, 2017

1 Revision History

Date	Version	Notes
14/11/2017	1.0	First draft

2 Symbols, Abbreviations and Acronyms

The same Symbols, Abbreviations and Acronyms as in the SRS, the TestPlan and the MG (available in STEM Moiré GPA repository) are used in the Module Interface Specifications document.

addition to document

Contents

3 Introduction

The following document details the Module Interface Specifications for STEM Moiré GPA. The full documentation and implementation can be found in STEM Moiré GPA repository.

4 Notation

The structure of the MIS for modules comes from [?], with the addition that template modules have been adapted from [?]. The mathematical notation comes from Chapter 3 of [?]. For instance, the symbol $:=$ is used for a multiple assignment statement and conditional rules follow the form $(c_1 \Rightarrow r_1 | c_2 \Rightarrow r_2 | ... | c_n \Rightarrow r_n)$.

The following table summarizes the primitive data types used by STEM Moiré GPA.

Data Type	Notation	Description
character	char	a single symbol or digit
integer	$\mathbb{Z}$	an integer number
natural number	$\mathbb{N}$	a natural number
real	$\mathbb{R}$	a real number

The specification of STEM Moiré GPA uses some derived data types: sequences, strings, and tuples. Sequences are lists filled with elements of the same data type. Strings are sequences of characters. Tuples contain a list of values, potentially of different types. In addition, STEM Moiré GPA uses functions, which are defined by the data types of their inputs and outputs. Local functions are described by giving their type signature followed by their specification.

5 Module Decomposition

The following table is taken directly from the Module Guide document for this project.

Level 1	**Level 2**
Hardware-Hiding Module	
Behaviour-Hiding Module	Input STEM Moiré GPA Control STEM Moiré GPA GUI User Input SMH simulation GPA Mask Unstrained region Conversion 2D strain tensor
Software Decision Module	Fourier Transform Least square fitting method Phase calculation Gradient Generic GUI/Plot Data structure

Table 1: Module Hierarchy

LIST ALL MIS to refer them in other document

6 MIS of STEM Moiré GPA Control Module (M 2)

6.1 Module

main

6.2 Uses

- STEM Moié GPA GUI
- Processing modules
  - Unstrained region
  - Conversion
  - SMH Simulation
  - GPA
  - 2D Strain Tensors
- Data Structure

6.3 Syntax

6.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

6.4 Semantics

STEM Moiré GPA is designed to have the different steps of the process flow driven by user directly through GUI_SMG. The STEM Moiré GPA Control Module uses the events in STEM Moié GPA GUI to use the processing modules in the order defined by the user.

6.4.1 State Variables

6.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

7 MIS of STEM Moiré GPA GUI Module (M 3)

7.1 Module

GUI_SMG

7.2 Uses

- Generic GUI/Plot
- Data Structure

7.3 Syntax

7.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

7.4 Semantics

STEM Moiré GPA process flow is driven by user through GUI_SMG. User triggers the events that start the wished processing step.

7.4.1 State Variables

7.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

8 MIS of Input Module (M 4)

8.1 Module

Input

8.2 Uses

- STEM Moié GPA GUI
- Data Structure

8.3 Syntax

8.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

8.4 Semantics

8.4.1 State Variables

8.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

9 MIS of SMH Simulation (M 5)

9.1 Module

SMHSimCalc

9.2 Uses

- Fourier Transform
- Input
- Data Structure

9.3 Syntax

9.3.1 Exported Access Programs

Name	In	Out	Exceptions
SMHsim	$I_{SMH_{\text{exp}}} : \mathbb{R}^2 \rightarrow \mathbb{R}$ $I_{C_{\text{ref}}} : \mathbb{R}^2 \rightarrow \mathbb{R}$, $p \in \mathbb{R}^{+*}$, $p_{\text{ref}} \in \mathbb{R}^{+*}$	$\widetilde{I}_{SMH_{\text{exp}}} : \mathbb{R}^2 \rightarrow \mathbb{C}$ $\widetilde{I}_{SMH_{\text{sim}}} : \mathbb{R}^2 \rightarrow \mathbb{C}$ $N_{\text{lim}} \in \mathbb{N}^*$	Nlim.zero()

9.4 Semantics

9.4.1 State Variables

None

9.4.2 Access Routine Semantics

SMHsim($I_{SMH_{\text{exp}}}, I_{C_{\text{ref}}}, p, p_{\text{ref}}$):

- output:
  - $\widetilde{I}_{SMH_{\text{exp}}}$ such that
  $$\widetilde{I}_{SMH_{\text{exp}}}(\vec{\nu}) = \mathcal{FT}[I_{SMH_{\text{exp}}}(\vec{r})]$$
  - $\widetilde{I}_{SMH_{\text{sim}}}$ such that
  $$\widetilde{I}_{SMH_{\text{sim}}}(\vec{\nu}) = \frac{1}{p^2} \sum_{\vec{q} \in Q_{lim}} \mathcal{FT}[I_{C_{\text{ref}}}(\vec{\nu} - \frac{\vec{q}}{p})]$$
  $$\text{with } Q_{\text{lim}} = \{\forall(n,m) \in \mathbb{Z}^2 \cap [-N_{\text{lim}}, N_{\text{lim}}]^2,\ \vec{q} = n\vec{u_x} + m\vec{u_y}\}$$
  $$\text{and } N_{\text{lim}} = \Xi(\frac{p}{p_{\text{ref}}}) \text{ with } \Xi \text{ the floor function}$$
- exception:

10 MIS of GPA Module (M 6)

10.1 Module

GPA

10.2 Uses

- Mask
- Fourier Transform
- Phase
- Gradient
- Data Structure

10.3 Syntax

10.3.1 Exported Access Programs

Name	In	Out	Exceptions
Phase	$\widetilde{I}_{SMH_{\exp}} : \mathbb{R}^2 \to \mathbb{C}$, $M : \mathbb{R}^2 \to \mathbb{R}$, $\overrightarrow{g_0} : \mathbb{R}^2 \to \mathbb{R}$	$P_{\vec{g}} : \mathbb{R}^2 \to \mathbb{R}$, $\overrightarrow{\Delta g} : \mathbb{R}^2 \to \mathbb{R}^2$, $P_{\Delta\vec{g}} : \mathbb{R}^2 \to \mathbb{R}$	-

10.4 Semantics

10.4.1 State Variables

10.4.2 Access Routine Semantics

Phase($\widetilde{I}_{SMH_{\exp}}, M, \overrightarrow{g_0}$):

- output:
  - $P_{\vec{g}}$ such that
$$\forall \vec{r} \in \mathbb{R}^2,\ P_{\vec{g}}(\vec{r}) = \arg(i\mathcal{FT}[M \times \widetilde{I}_{SMH_{\exp}}])$$
  - $\overrightarrow{\Delta g}$ such that
$$\forall \vec{r} \in \mathbb{R}^2,\ \Delta\overrightarrow{g}(\vec{r}) = \frac{1}{2\pi}\mathrm{grad}(\mathrm{unwrap}(P_{\vec{g}}(\vec{r}))) - \overrightarrow{g_0}(\vec{r})$$
  - $P_{\Delta\vec{g}}$ such that
$$\forall \vec{r} \in \mathbb{R}^2,\ P_{\Delta\vec{g}}(\vec{r}) = \mathrm{wrap}(\mathrm{unwrap}[P_{\vec{g}}(\vec{r})] - 2\pi\overrightarrow{g_0}(\vec{0}) \cdot \vec{r})$$
- exception:

11 MIS of Mask Module (M 7)

11.1 Module

Mask

11.2 Uses

- Input
- Data structure

11.3 Syntax

11.3.1 Exported Access Programs

Name	In	Out	Exceptions
MCirc	$(x_c, y_c) \in \mathbb{N}^2$, $R \in \mathbb{R}^{+*}$	$M : \mathbb{R}^2 \to \mathbb{R}$, $\overrightarrow{g_0} : \mathbb{R}^2 \to \mathbb{R}$	-

11.4 Semantics

11.4.1 State Variables

11.4.2 Access Routine Semantics

MaskGene(x_c, y_c, R):

- output:
  - M such that
$$M(x,y) = \begin{cases} 1, & (x-x_c)^2 + (y-y_c)^2 \leq R^2 \\ 0, & (x-x_c)^2 + (y-y_c)^2 > R^2 \end{cases}$$
  - $\overrightarrow{g_0}$ such that
$$\forall \vec{r} \in \mathbb{R}^2,\ \overrightarrow{g_0}(\vec{r}) = \begin{bmatrix} x_c \\ y_c \end{bmatrix}$$
- exception:

12 MIS of Unstrained region (M 8)

12.1 Module

UReFCalc

12.2 Uses

- Least Square Fit
- Input
- Data Structure

12.3 Syntax

12.3.1 Exported Access Programs

Name	In	Out	Exceptions
ZeroStrain	$\overrightarrow{\Delta g}^M : \mathbb{R}^2 \to \mathbb{R}^2$, bla	$\overrightarrow{\Delta g}^M : \mathbb{R}^2 \to \mathbb{R}^2$	-

12.4 Semantics

12.4.1 State Variables

12.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

13 MIS of Conversion Module (M 9)

13.1 Module

MtoCConv

13.2 Uses

- Input
- Data Structure

13.3 Syntax

13.3.1 Exported Access Programs

Name	In	Out	Exceptions
conversion	$p \in \mathbb{R}$, $(n,m) \in \mathbb{N}^2$, $\overrightarrow{g}^{M_{\text{exp}}}_{\text{uns}} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$	$\overrightarrow{g_j}^{C_{exp}}_{uns} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$	-

13.4 Semantics

13.4.1 State Variables

13.4.2 Access Routine Semantics

conversion(p, $\overrightarrow{g}^{M_{\text{exp}}}_{\text{uns}}$):

- output: $\overrightarrow{g}^{M_{\text{exp}}}_{\text{uns}}$ such that

$$\forall \vec{r} \in \mathbb{R}^2,\ \overrightarrow{g_j}^{C_{\text{exp}}}_{\text{uns}}(\vec{r}) = \overrightarrow{g_j}^{M_{\text{exp}}}_{\text{uns}}(\vec{r}) + p \times \begin{bmatrix} n \\ m \end{bmatrix}$$

- exception:

14 MIS of 2D Strain Tensor Module (M 10)

14.1 Module

2D_Strain

14.2 Uses

Data Structure

14.3 Syntax

14.3.1 Exported Access Programs

Name	In	Out	Exceptions
CalcStrain	$g_{1_{\text{uns}}}^{C_{\text{exp}}} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, $g_{2_{\text{uns}}}^{C_{\text{exp}}} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, $\Delta g_{1_{\text{uns}}}^{C_{\text{exp}}} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$, $\Delta g_{2_{\text{uns}}}^{C_{\text{exp}}} : \mathbb{R}^2 \rightarrow \mathbb{R}^2$	$T : \mathbb{R}^2 \rightarrow \mathbb{R}^4$	-

14.4 Semantics

14.4.1 State Variables

14.4.2 Access Routine Semantics

CalcStrain($g_{1_{\text{uns}}}^{C_{\text{exp}}}, g_{2_{\text{uns}}}^{C_{\text{exp}}}, \Delta g_{1_{\text{uns}}}^{C_{\text{exp}}}, \Delta g_{2_{\text{uns}}}^{C_{\text{exp}}}$):

- output:
- exception:

15 MIS of Fourier Transform Module (M 11)

 # 2D Fourier transform

15.1 Module

FTCalc

15.2 Uses

Data Structure

15.3 Syntax

15.3.1 Exported Access Programs

Name	In	Out	Exceptions
$\mathcal{FT}$	$f : \mathbb{R}^2 \to \mathbb{R}$	$f : \mathbb{R}^2 \to \mathbb{C}$	-
i$\mathcal{FT}$	$f : \mathbb{R}^2 \to \mathbb{C}$	$f : \mathbb{R}^2 \to \mathbb{R}$	-

15.4 Semantics

15.4.1 State Variables

None

15.4.2 Access Routine Semantics

 # Calculate the 2D Fourier transform of a function f

$\mathcal{FT}(f(x, y))$:

- output: $\widetilde{f}(\nu, \mu)$ such that

$$\forall(\nu, \mu) \in \mathbb{R}^2 \wedge \forall(x, y) \in \mathbb{R}^2,\ \widetilde{f}(\nu, \mu) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x, y)e^{-2i\pi(\nu x+\mu y)}dxdy$$

- exception:

 # Calculate the 2D inverse Fourier transform of a function $\widetilde{f}$

i$\mathcal{FT}(\widetilde{f}(\nu, \mu))$:

- output: $f(x, y)$ such that

$$\forall(x, y) \in \mathbb{R}^2 \wedge \forall(\nu, \mu) \in \mathbb{R}^2,\ f(x, y) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \widetilde{f}(\nu, \mu)e^{2i\pi(\nu x+\mu y)}dxdy$$

- exception:

16 MIS of Gradient Module (M 12)

 # 2D Gradient

16.1 Module

GradCalc

16.2 Uses

Data Structure

16.3 Syntax

16.3.1 Exported Access Programs

Name	In	Out	Exceptions
grad	$f : \mathbb{R}^2 \to \mathbb{R}$	$f : \mathbb{R}^2 \to \mathbb{R}^2$	-

16.4 Semantics

16.4.1 State Variables

16.4.2 Access Routine Semantics

 # Calculate the 2D gradient of a 2D function f

grad(f):

- output:$\nabla f(x,y)$ such that

$$\forall (x,y) \in \mathbb{R}^2,\ \nabla f(x,y) = \begin{bmatrix} \frac{\partial f}{\partial x}(x,y) \\ \frac{\partial f}{\partial y}(x,y) \end{bmatrix}$$

- exception:

17 MIS of Least Square Fit Method Module (M 13)

 # 2D linear least square method to fit a function f

17.1 Module

LSFMCalc

17.2 Uses

Data Structure

17.3 Syntax

17.3.1 Exported Access Programs

Name	In	Out	Exceptions
lsfm	$f : \mathbb{R}^2 \to \mathbb{R}$	$f : \mathbb{R}^2 \to \mathbb{R}$	-

17.4 Semantics

17.4.1 State Variables

17.4.2 Access Routine Semantics

 # Calculate the 2D fit of a function f using the linear least square method on a domain $U = ([x_0, x_1]; [y_0, y_1]) \in \mathbb{R}^2$

lsfm(f,U):

- output: $fit(x, y) = ax + by$ such that

$$\forall (x, y) \in U,\ E(a, b) = \int_{x_0}^{x_1} \int_{y_0}^{y_1} [f(x, y) - fit(x, y)]^2 dxdy \text{ is minimized}$$

$$\Rightarrow \frac{\partial E}{\partial a} = 0 \wedge \frac{\partial E}{\partial b} = 0 \Rightarrow a = \frac{\int_{x_0}^{x_1} \int_{y_0}^{y_1} xf(x, y)dxdy}{\int_{x_0}^{x_1} \int_{y_0}^{y_1} x^2 dxdy} \wedge b = \frac{\int_{x_0}^{x_1} \int_{y_0}^{y_1} yf(x, y)dxdy}{\int_{x_0}^{x_1} \int_{y_0}^{y_1} y^2 dxdy}$$

- exception:

18 MIS of Phase Operation Module (M 14)

18.1 Module

PhaseCalc

18.2 Uses

Data Structure

18.3 Syntax

18.3.1 Exported Access Programs

Name	In	Out	Exceptions
unwrap	$f : \mathbb{R}^2 \rightarrow]-\pi, \pi]$	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	-
wrap	$f : \mathbb{R}^2 \rightarrow \mathbb{R}$	$f : \mathbb{R}^2 \rightarrow]-\pi, \pi]$	-
arg	$z \in \mathbb{C}$	$\phi \in]-\pi, \pi]$	

18.4 Semantics

18.4.1 State Variables

18.4.2 Access Routine Semantics

wrap(f):

- output: g such that

$$\forall (x,y) \in \mathbb{R}^2, \exists k \in \mathbb{Z} | g(x,y) = f(x,y) + 2k\pi \wedge g(x,y) \in]-\pi, \pi]$$

- exception:

unwrap(f):

- output: g such that

$$\forall (x,y) \in \mathbb{R}^2, \exists k \in \mathbb{Z} | g(x,y) = f(x,y) + 2k\pi \wedge g \text{ is continous}$$
$$\Rightarrow \forall (x,y) \in \mathbb{R}^2, \exists k \in \mathbb{Z} | \lim_{(x,y)\to(x_0,y_0)} g(x,y) = g(x_0,y_0) = f(x_0,y_0) + 2k\pi$$

- exception:

arg(z):

- output: ϕ such that

$$\phi = \arg(z) = \text{atan2}(z) = \text{atan2}(\frac{\text{Im(z)}}{\text{Re(z)}})$$

- exception:

19 MIS of Data Structure Module (M 15)

19.1 Module

DataStruct

19.2 Uses

19.3 Syntax

19.3.1 Exported Access Programs

Name	In	Out	Exceptions
set	Metadata	-	-

19.4 Semantics

19.4.1 State Variables

19.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

20 MIS of Generic GUI/Plot Module (M 16)

20.1 Module

GUIGene

20.2 Uses

Hardware-Hiding Data Structure

20.3 Syntax

20.3.1 Exported Access Programs

Name	In	Out	Exceptions
	-	-	-

20.4 Semantics

20.4.1 State Variables

20.4.2 Access Routine Semantics

():

- transition:
- output:
- exception:

21 Appendix