Fast Method to Find Critical Points of the Electron Density in Large Systems

Jorge Garza

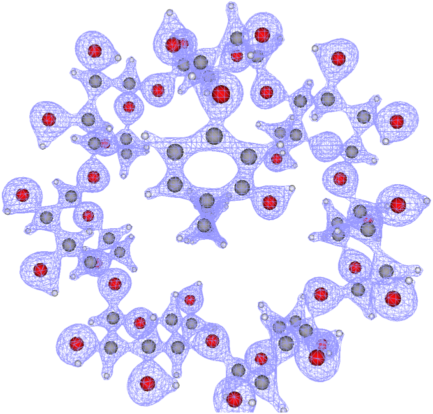
Departamento de Química
Área de Fisicoquímica Teórica
Universidad Autónoma Metropolitana-Iztapalapa.

March 19th, 2015

Casa abierta al tiempo

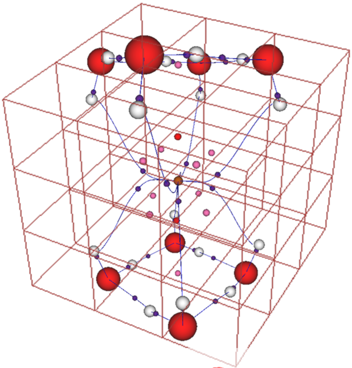
Electron density

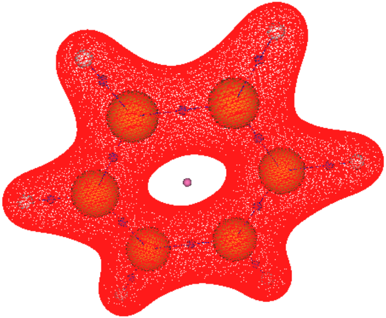
The analysis of the electron density or other scalar field is important to understand the microscopic world

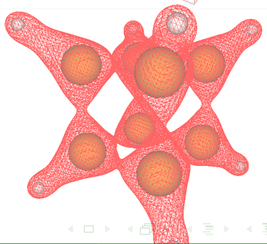


Visualization of orbitals and electron density

In quantum chemistry, orbitals or electron density are evaluated typically on a mesh to be displayed on a screen by using the marching cubes algorithm.







Electron density

For wave-function methods or density functional theory the electron density is obtained from

$$\rho(\vec{r}) = \sum_{i=1}^{occ} \omega_i \psi_i^*(\vec{r}) \psi_i(\vec{r})$$

Electron density

For wave-function methods or density functional theory the electron density is obtained from

$$\rho(\vec{r}) = \sum_{i=1}^{occ} \omega_i \psi_i^*(\vec{r}) \psi_i(\vec{r})$$

For atoms, molecules or extended systems, in general, the orbitals are represented in a basis set functions

$$\psi_i(\vec{r}) = \sum_{\mu=1}^{K} c_\mu^{(i)} f_\mu(\vec{r})$$

- $\{f_\mu\}$: basis set functions.
- $\{c_\mu\}$: coefficients obtained from a quantum chemistry method.
- K : number of the basis functions.

Gaussian functions

In the GTC 2014, I presented one code where the Gaussian functions are used as basis set

$$f_\mu(\vec{r}) = (x-X)^{m_\mu}(y-Y)^{l_\mu}(z-Z)^{n_\mu}e^{-\zeta r^2}$$

with

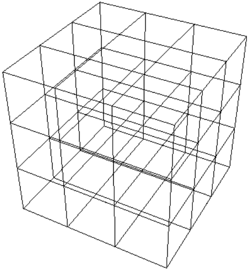
$$r^2 = (x-X)^2 + (y-Y)^2 + (z-Z)^2$$

(X, Y, Z): coordinates of a center (nucleus). There are several codes where gaussian functions are used to describe orbitals or electron density for atoms, molecules or solids.

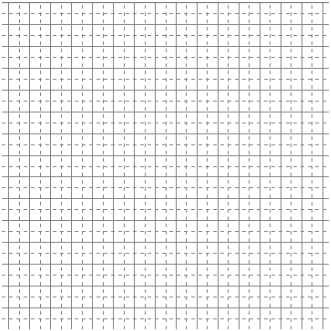
Evaluation of electron density on a GPU

The evaluation of ρ is considered as a reduction problem

- One thread is associated to each point on the mesh.



Mesh for the electron density

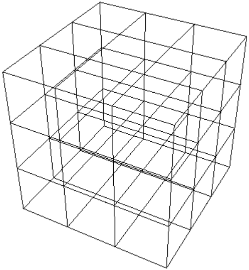


Mesh on the GPU

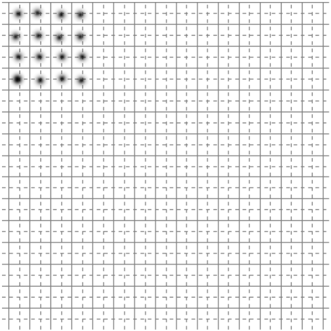
Evaluation of electron density on a GPU

The evaluation of ρ is considered as a reduction problem

- 64 threads are associated to each point on the mesh.



Mesh for the electron density



Mesh on the GPU

Computing and rendering of scalar fields on a GPU

Our code is designed to evaluate :

- Orbitals
- Electron density
- Laplacian
- Reduced gradient
- Electron localization function
- Electrostatic potential

Grid-based methods

Additionally to the visual part related with the electron density, there are tools to understand the chemical bond concept.

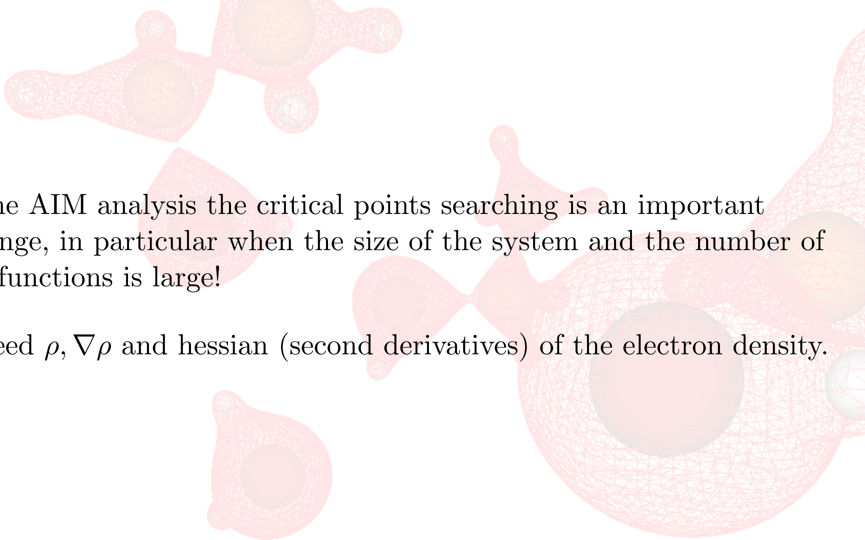
The Atoms in Molecules (AIM) analysis predicts a chemical bond in a molecule if the condition

$$\nabla\rho(\vec{r}) = 0$$

is satisfied.

All points that satisfy this condition are known as **critical points**.

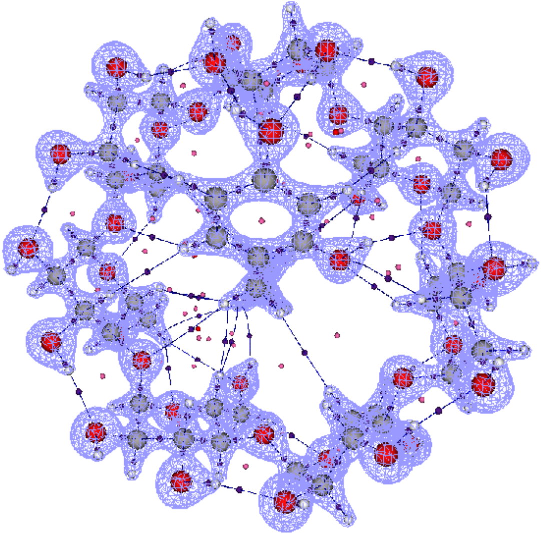
Grid-based methods



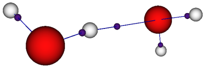
For the AIM analysis the critical points searching is an important challenge, in particular when the size of the system and the number of basis functions is large!

We need $\rho, \nabla\rho$ and hessian (second derivatives) of the electron density.

AIM on GPUs



AIM on GPUs: Three examples

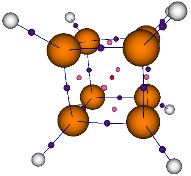


M06-2X

10 occupied orbitals

96 primitive gaussian functions

1 bond critical point



B3LYP

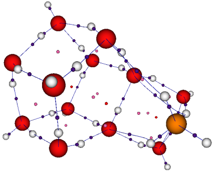
28 occupied orbitals

408 Gaussian functions

20 bond critical points

6 ring critical points

1 cage critical point



MP2

495 occupied orbitals

720 primitive gaussian functions

51 bond critical points

14 ring critical points

3 cage critical points

AIM on CPUs: Total time in seconds

CPU	H_2O-H_2O		C_8H_8		$(H_2O)_{12}CH_4$	
	i7-4770					
1	31	(1.00)	1938	(1.00)	254561	(1.00)
2	16	(0.97)	968	(1.00)	128840	(0.99)
4	09	(0.86)	514	(0.94)	67835	(0.94)
	Xe E5-2670 v2					
1	50	(1.00)	3175	(1.00)	428877	(1.00)
2	25	(1.00)	1591	(1.00)	215960	(0.99)
4	13	(0.96)	793	(1.00)	108103	(0.99)
8	07	(0.89)	398	(1.00)	54211	(0.99)
16	04	(0.78)	200	(0.99)	27170	(0.99)

AIM on GPUs: Total time in seconds

CPU	H_2O-H_2O	C_8H_8	$(H_2O)_{12}CH_4$
Xe E5-2670 v2			
16	04	200	27170

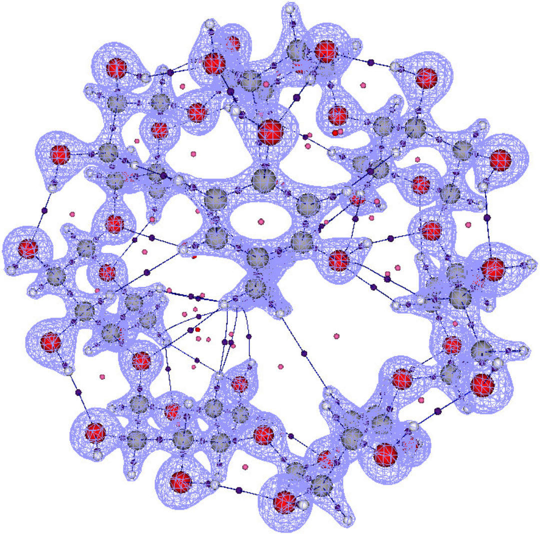
AIM on GPUs: Total time in seconds

CPU	H_2O-H_2O	C_8H_8	$(H_2O)_{12}CH_4$
Xe E5-2670 v2			
16	04	200	27170
GPU	H_2O-H_2O	C_8H_8	$(H_2O)_{12}CH_4$
GeForce GTX 760			
1	01	21	2083
Tesla M2090			
1	01	24	2348
2		15	1381
4		13	820
Tesla K80			
1	00	11	880
2		08	553
4		08	350

AIM on GPUs: single and double precision

	Single-precision			Double-precision		
GPU	H_2O-H_2O	C_8H_8	$(H_2O)_{12}CH_4$	H_2O-H_2O	C_8H_8	$(H_2O)_{12}CH_4$
	GeForce GTX 760					
1	01	21	2083	01	43	4809
	Tesla M2090					
1	01	24	2348	01	37	3783
2		15	1381		25	2386
4		13	820		22	1498

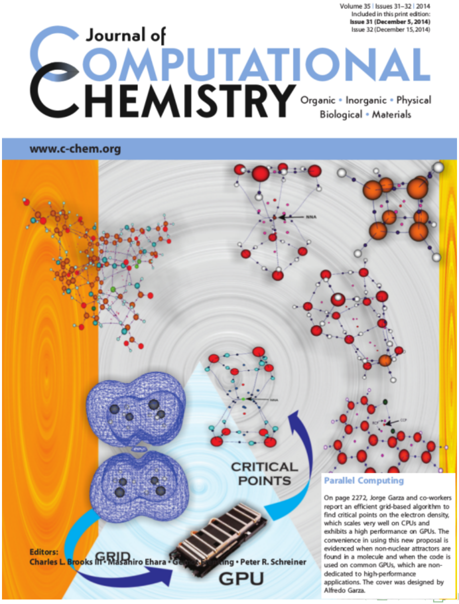
AIM on GPUs



AIM on GPUs

Details about implementation of AIM by grid-based methods
R. Hernández-Esparza, S.- M. Mejía-Chica, A. Martínez-Melchor,
A.- D. Zapata-Escobar, A. Guevara-García, J.- M. Hernández-Pérez,
R. Vargas and J. Garza
J. Comput. Chem. **35**, 2272-2278 (2014).

AIM on GPUs



Volume 35 | Issues 31–32 | 2014
Included in this print edition:
Issue 31 (December 5, 2014)
Issue 32 (December 15, 2014)

Journal of COMPUTATIONAL CHEMISTRY

Organic • Inorganic • Physical
Biological • Materials

www.c-chem.org

Parallel Computing

On page 2272, Jorge Garza and co-workers report an efficient grid-based algorithm to find critical points on the electron density, which scales very well on CPUs and exhibits a high performance on GPUs. The convenience in using this new proposal is evidenced when non-nuclear attractors are found in a molecule and when the code is used on common GPUs, which are non-dedicated to high-performance applications. The cover was designed by Alfredo Garza.

Editors:
Charles L. Brooks III • Masahiro Ehara • Gernot Frenking • Peter R. Schreiner

Semiempirical methods

Additionally to the codes based on gaussian functions, there are codes which are implented using Slater Type Orbitals.
For example, semiempirical methods use this kind of basis set

$$f_\mu(\vec{r}) = (x - X)^{m_\mu}(y - Y)^{l_\mu}(z - Z)^{n_\mu} e^{-\zeta r}$$

with

$$r = \sqrt{(x - X)^2 + (y - Y)^2 + (z - Z)^2}$$

(X, Y, Z): coordinates of a center (nucleus).

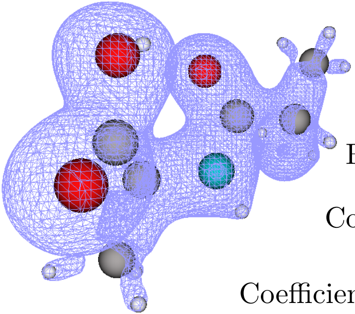
These methods use only valence orbitals.

Semiempirical methods

Semiempirical methods present an important challenge!!

In these methods the number of atoms in the molecule is large, and consequently the number of basis set functions to be used could be huge.

Analysis of the electron density from semiempirical methods implemented in our code



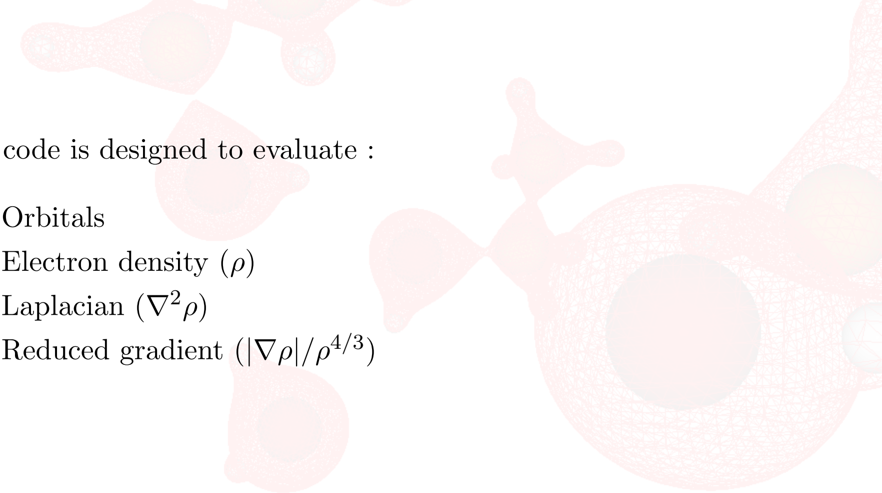
Evaluation of scalar and vector fields on GPUs

Rendering by using GPUs

Code based on CUDA

Coefficients from MOPAC

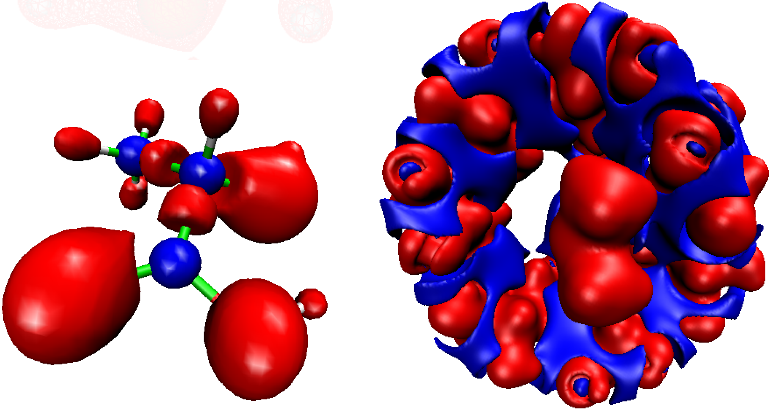
Computing and rendering of scalar fields on a GPU for semiempirical methods



Our code is designed to evaluate :

- Orbitals
- Electron density (ρ)
- Laplacian ($\nabla^2\rho$)
- Reduced gradient ($|\nabla\rho|/\rho^{4/3}$)

Computing and rendering of scalar fields on a GPU for semiempirical methods

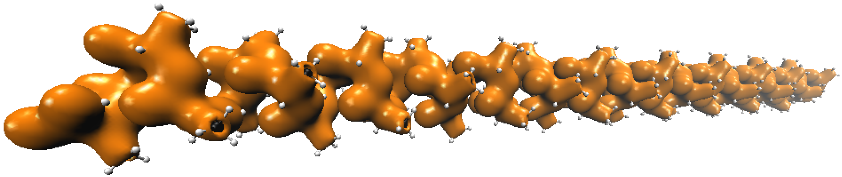


Alanine: 13 atoms, 32 orbitals, 56 primitive functions.

β-cyclodextrin:168 atoms, 252 orbitals, 432 primitive functions.

Computing and rendering of scalar fields on a GPU for semiempirical methods

Poly-alanine/60 residues



603 atoms
844 orbitals, 1506 primitive functions

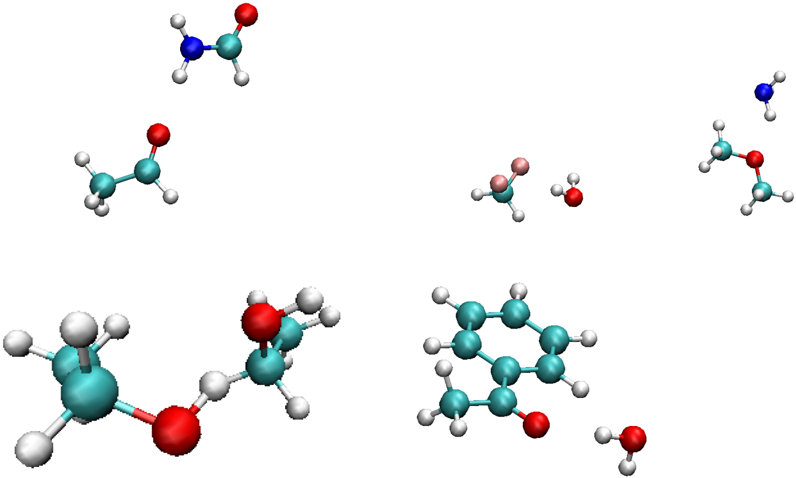
Evaluation of the electron density for poly-(Ala)$_n$ from semiempirical methods

n	Atoms	Func.	Orb.	Points	Time (s)		
					M2090	K80	CPU
4	43	106	60	313,632	1	1	26
5	53	131	74	373,248	2	2	45
10	103	256	144	958,800	15	7	446
15	153	381	214	1,953,504	48	24	1999
20	203	506	284	3,369,600	128	54	6112
30	303	756	424	8,276,400	701	349	33007
35	353	881	494	10,365,264	1295	509	56533

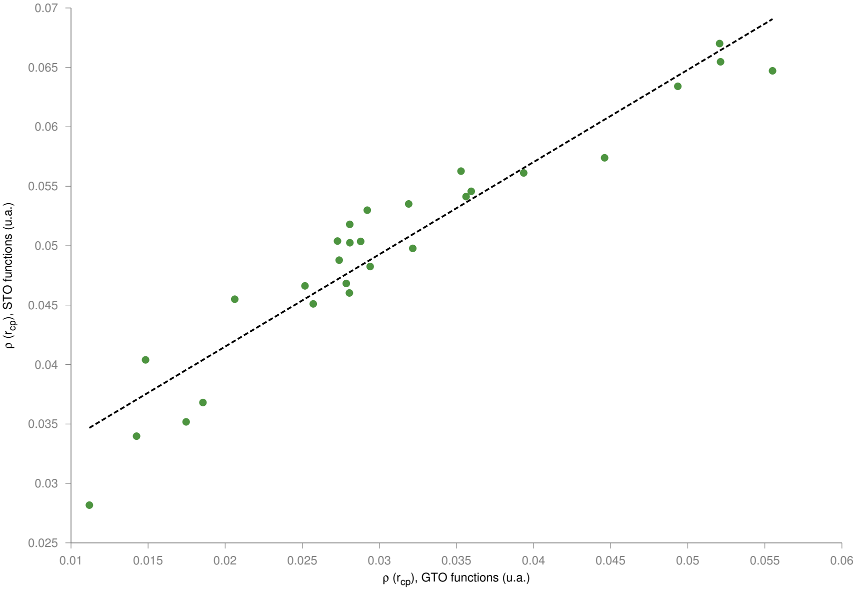
Evaluation of the electron density for poly-(Ala)$_n$ from semiempirical methods

n	Atoms	Func.	Orb.	Points	Time (s)		
					M2090	K80	CPU
4	43	106	60	313,632	1	1	26
5	53	131	74	373,248	2	2	45
10	103	256	144	958,800	15	7	446
15	153	381	214	1,953,504	48	24	1999
20	203	506	284	3,369,600	128	54	6112
30	303	756	424	8,276,400	701	349	33007
35	353	881	494	10,365,264	1295	509	56533
40	403	1006	564	13,893,120	1961	1009	
45	453	1131	634	19,077,120	3523	1537	
50	503	1256	704	24,135,552	5378	2171	
60	603	1506	844	39,387,256	12428	5691	
70	703	1756	984	57,189,888	23405	10509	

Systems linked by hydrogen bonds



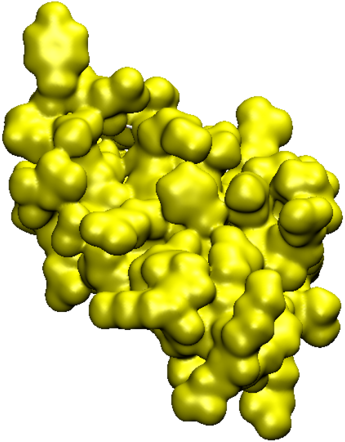
Systems linked by hydrogen bonds



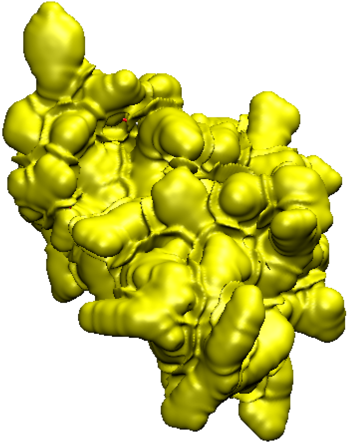
Systems linked by hydrogen bonds

1020 atoms, 1399 orbitals, 2532 primitive functions

ρ



$\frac{|\nabla \rho|}{\rho^{4/3}}$



Systems linked by hydrogen bonds

1020 atoms
1399 orbitals, 2532 primitive functions

7,429,000 points in the mesh

	M2090	K80
ρ	5940	3813
$\frac{\vert\nabla\rho\vert}{\rho^{4/3}}$	21308	9773

Conclusions

- Code on GPUs to evaluate scalar and vector fields in quantum chemistry.
- Wave function from NWChem, G09, GAMESS or MOPAC.
- Critical points searching based on grid-methods for large systems.
- Stable code and tested over several GPUs.

Collaborators and acknowledgments

- Raymundo Hernández-Esparza
- Luis Antonio Soriano-Agueda
- Sol Milena Mejía-Chica
- Apolinar Martínez-Melchor
- Andy D. Zapata-Escobar
- Dr. Rubicelia Vargas (UAMI)
- Dr. Julio Manuel Hernández-Pérez (BUAP)
- Dr. Raquel Váldez (UAMI)
- M. C. Oscar Yañez (UAMI)

Collaborators and acknowledgments

CONACYT: Project 155070

www.fqt.izt.uam.mx