Available online at www.joac.info

ISSN: 2278-1862

Journal of Applicable Chemistry

2013, 2 (5):1007-1034

(International Peer Reviewed Journal)

E-man Part 4[#]: Tutorial on prospects of charged system search (CSS) algorithm in chemical sciences

K RamaKrishna[1], Ch V Kameswara Rao[1] and R Sambasiva Rao[2*]

1. Department of Chemistry, Gitam Institute of Science, Gitam University, Visakhapatnam, 530 017, **INDIA**
2. School of Chemistry, Andhra University, Visakhapatnam 530 003, **INDIA**

Email: karipeddirk@gmail.com, rsr.chem@gmail.com

Received on 23rd August and finalized on 29th August 2013.

(Dedicated to Dr. P.V. Krishna Rao, our teacher and former professor of physical chemistry, Andhra University on his 75th birth anniversary)

ABSTRACT

CSS (charged system search or Charge_Syst_Serch) algorithm is another nature inspired multi-agent optimisation tool of this decade. It is applicable to variable selection in structure X (: activity, toxicity, property etc.) relationships (SXR), estimation of equilibrium constants of multiple chemical equilibria and rate constants of parallel/ consecutive kinetics profiles. The extensive application in engineering and advances in the algorithm brought CSS to the forefront of bandwagon of physics based swarm approaches viz. gravitational, big-bang_big-crunch, intelligent_water-drop etc. The Charge_Syst_Serch algorithm consists of calculation of the resultant force affecting each of a set of charged particles based on Coulomb and Gauss electrostatic interaction. Here, the magnitude of charge of a charged particle (CP) depends on fitness value of object function. Newton's law of mechanics directs the movement of CPs to refined positions in the search space. The iterative improvements of approximate (random) solutions converge to true optimum. CSS algorithm converged to Pareto optimal solution of non-convex functions and it finds application in parametric models. Incorporation of magnetic forces, chaos and correction for fence crossing during refinement improved efficacy of CSS. The binary hybridization of CSS with another E-man module viz. ant colony optimisation (ACO), particle swarm optimization (PSO), Big-Bang_Big-crunch (BB_BC) is the development with a right perspective of deriving synergistic benefits of both worlds. This combination at the same time diminishes the short comings of individual component algorithms. The concept of fields of forces, again from Physics, mimicking ACO, PSO and CSS is another landmark to probe deep into core mathematics for future prospects. These multi-agent search/optimization tools designed for multi objective multi-dimensional-non-linear-convex functions with constraints/ discontinuities are implementable on parallel software and hardware architectures.

Keywords: Charged System, Multi-object-functions, E-man, Nature mimicking, SXR, Chemical equilibria, Chemical kinetics, Electrostatics, Magnetic force, hybrid E-man.

[#] Part 3: Journal of Applicable Chemistry, 2013, 2 (4), **698-713**; Part 2: Journal of Applicable Chemistry, 2012, 1 (1), 109-124

INTRODUCTION

The wisdom/knowledge in swarm behavior or flocks of birds, parliament of owls, herds of land animals, fish schools and various social insects (bees, wasps, ants, termites, mosquitoes, fireflies, glowworms etc.) paved way for nature inspired algorithms over half a century. In this decade, the inspiration of physical forces in nature resulted in a new era of mathematical optimization/variable search methods [1-5]. The (subset of finite) nature, scientific laws of physics and E-man are described in the table 1.

Table 1. Nature's process to nature-inspired algorithms through laws of physics

	Response	Physics	E-man
Nature	[Charge, mechanics]	Laws of Coulomb Gauss Newton	Charge_Syst_Search
	[Magnetism]	Biot–Savart law	Mag_Charge_Syst_Search
	[Gravity]	Newton's gravitation	[Grav.binary, Grav.Float]
	[river course]	Fluid dynamics	[Intelligent_water_drop]
	[Origin of universe]	Big-Bang_Big- Crunch	[Big-Bang_Big-Crunch]
	[Sound]	Audible Sound	[Cuckoo]
Man made	[Rhythm]	Music	[Harmony search]

a. Forces in nature

The study of properties, transformation of matter and energy are the focus of the scientific and non scientific pursuit from early history of man; the hamosapeon. The particles occupying space at different instants of time exert forces of widely varying magnitude. This results in motion of particles, aggregation, disassociation etc. The space–time mathematical model is a four dimensional physical concept comprehending space and time in a single construct. Newtonian gravitational/electric/magnetic interactions produce typical fields in physics. From classical physics, force between two particles --- charges, magnetic monopoles, or masses--- are inversely proportional to the square of the distance of separation between them and directly proportional to the product of their numerical magnitudes. It is popular as an inverse square law *(Eqn.* A01). The direction of force is along the line joining the particles. The magnitude of the field for particle *i* is obtained by considering magnitude of second particle to be unity (*$magnitude_j$=1)*

b. Translation of charges interaction into laws of physics

The attractive force pools all agents (charges here) in a small area of the search space, while repelling force disperses them far away from each other. The net force is a product of a function of the magnitude of the charges (Appendix 01). The force between two charges bearing a of unit of charge at one meter distance is 9.0e9 N m^2/c^2, while that between unit two masses separated by one meter is 6.67e-11 N m^2/Kg^2. These numerical values are electrostatic (kel) and gravitational (G) constants respectively. Coulomb's law of electrostatic attraction between two charges, its limitations/failures, Gauss's law and integrated form of these two laws accounting for the attraction force/direction inside/ on /outside/ the charged sphere are incorporated in Appendix-2 along with m(atlab)-functions for calculation and numerical illustration.

c. Mathematical optimisation

In yester years, mostly a single object function is used in minimization/maximization employing a range of gradient to direct non-gradient (including simplex) methods. The functions considered are nonlinear and mostly quadratic and rarely curved functions. But, in the last two decades, multiple- object functions/convergence criteria/Pareto optimality of multimodal nonlinear profiles with several global/local optima has become routine in chemical sciences and engineering. The optimization task becomes more challenging if the function is not known/ has breaks/ indefinite, infinity or NAN (not a number) at some points in the search space. Nature inspired algorithms are now a new class of the tools and their

combination with classical mathematical tools are better than either of the categories. The complexity now is in terms of accuracy and reliability rather than the CPU time and memory requirements. Of course, they also play a pertinent role in the case of repeatable multiple optimization tasks in routine industrial/manufacturing. Unfortunately, there is no algorithm to achieve the best solution for all optimization problems like no drug is panacea for all the diseases. In fact, a set of algorithms output a better solution for select tasks compared to others. The initial/approximate search point plays a dominant role in the convergence and escaping from local optima in the case of conventional direct search methods and Cheng et.al. [5] reported limitations of heuristic global optimization algorithms. A brief sketch of mathematical optimization and CSS are represented in pseudo code form in Chart 1.

Chart 1: Bird's eye view of **Optimization algorithm**

Chart 1 (a): Mathematical optimization

- **Input**
  - **Data**
  - Object function (objFn)
  - Convergence limits
  - Stopping rules
  - Search space identification
- **Initialisation**
  - Parameters of optimization method
  - Approximate solution(s)

Iterate until convergence | stopping criteria

Cal objFn
Operate Opt_method
Refine X
X_app ← X_refined

EndIterate
Output

Chart 1 (b): CSS search algorithm

Initialisation
Iterate until convergence | stopping criteria

For each CP
Cal objFnValue, q of CP
Cal Coulomb electrostatic force

Update position and velocity

endFor
endIterate
Output

Chart 1(c): Mag_charge_syst search

Initialisation
Iterate until convergence | stopping criteria

For each CP
Cal objFnValue, q of CP
Cal Coulomb electrostatic force
Cal magnetic force
Update position and velocity
Boundary exceedance of X
endFor
endIterate
Output

2. (Artificial) Charge system search algorithm (Charge_Syst_Serch_Alg)

In 2010, Kaveh and Talatahari [6] proposed a nature inspired charged system search algorithm for single object function minimization tasks. It is a population based stochastic meta-heuristic procedure with multiple-agents. Each agent is a charged particle (CP) based on the fitness value of object function and its distance from the others and this standard algorithm is the basis of an extensive study [7-28].

Chart 1(d): Terminology of CSS and mathematical optimization

Domain		
Nature	**Charged System Search**	Optimization
Moieties	nsol	Number of solutions (agents)
Charges	charged particles	Approximate (possible) solutions
Amount of charge	Charge	Fitness (quality), objective function value
Sequence of processes	Movement of charges based on force of attraction	Gradient/ direct search

Chart 1(e): Types of constants, initial/intermediate variables and system software for CSS

Mathematical	constants		FP	eps		
Optimisation	User chosen		integer	iter_max		
			FP	accuracy_ expected	Max(dist.CPs)	
	Intermediate		integer	iter		
	calculations		FP	fit_i		
CSS	constants		FP	Cof_vel	Radius of influence	coe_acc
			integer	Δt		
			FP	Xmin(i) Xmax(i)		
	initiation		FP	$q0_i$	$X0_{i,d}$	Vel0i
	Intermediate	tensors		**X**iter, dist_sep		
		integer	integer	Prob_CP_movement		
		scalar	scalar	fitbest and fitworst		CMCR
Matlab functions	Rand					
	FP : Floating point					

Assumptions_ Charge_Syst_Search (CSS)

Each CP is a sphere of radius 'rad' with uniform volume charge density. It exerts an attractive electrical force on other charges (agents) [20]. The laws of motion of Newtonian mechanics (Appendix 3) drive the movement of CPs to a new position in objFn search space. Each solution candidate is considered as a charged particle in an n-dimensional space.

Basis: The magnitude of the electric field at a point inside/outside a charged insulating solid sphere is calculated by integrated form of Coulomb and Gauss laws of electrostatics (Appendix 2, fig A2). The electric force operated on CPs (agents) results in the acceleration of their movement to a new position in the multi-dimensional-object-function (m-D-obj-fn) search space (fig.1).

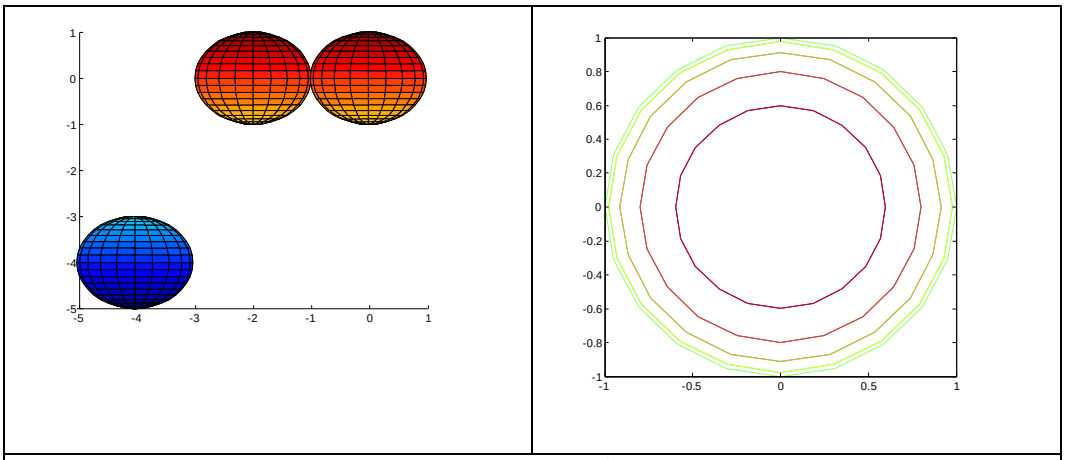


Fig. 1. Multiple (= 3) charges of different charge values

a.Data structure: The data structure of variables and free parameters in CSS in tensorial notation are given in chart 2(a). A bird's eye view of terminology of CSS with intermediate variables and constants are in chart 1(d, e).

Chart 2: Data structure, charge of CPs and forces in Charge_system.alg

(a) Tensorial representation of variables and free parameters of **Charge_system alg**.

vectors

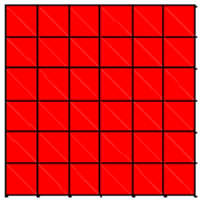
$$fitness : fit_1 \quad fit_2 \quad fit_i \quad fit_{nsol}^T$$

$$charge\ (q) : q_1 \quad q_2 \quad q_i \quad q_{nsol}^T$$

$$velocity\ (v): vel_1 \quad vel_2 \quad vel_i \quad vel_{nsol}^T$$

MATRIX

$$position: \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,d} \\ x_{2,1} & x_{2,2} & x_{2,d} \\ x_{i,1} & x_{i,2} & x_{i,d} \\ x_{nsol,1} & x_{nsol,2} & x_{nsol,d} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}_1 \\ \boldsymbol{x}_2 \\ \boldsymbol{x}_i \\ \boldsymbol{x}_{nsol} \end{bmatrix}$$



superscript ^{T}: Transpose of vector/matrix/tensor

Nsol	:	Number of solutions
fit_i	:	Objective function value or the fitness of *i*th agent (e.g. ESS)
veli	:	velocity of i^{th} CP
q_i	:	Charge of i^{th} CP
x_{id}	:	Coordinates of i^{th} position in d^{th} dimension

Data structure with iterations

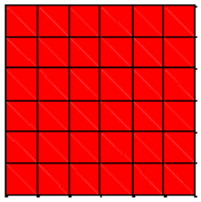
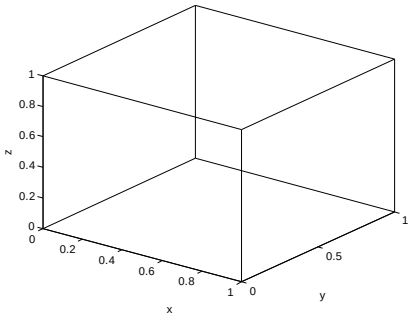
	■ fitness ■ charge ■ velocity		Positio n (X)
Second order tensor (or matrix or 2-way data) Nsol x iter		Third order tensor (3-way data): nsol x ndim x iter	

Chart 2(b):Calculation of charge of CPs

Algebraic equation	Vector/tensor (matlab program)

$q(i) = \frac{fit(i) - fitworst}{fitbest - fitworst} \quad i = 1,2,...nsol$
Eqn.1
fitworst : Worst fitness value
fitbest : Best fitness value
	`function [q,fit]= om_fit2q(fit2)`
	`[nsol,col] = size(fit2);`
	`one = ones(nsol,1);`
	`%`
	`[fit.sort,Ix]= sort(fit2);`
	`%`
	`fit.Ix=Ix;`
	`fit.worst = fit.sort(1,1);`
	`fit.best = fit.sort(nsol,1);`
	`%`
	`q = (fit2 - fit.worst*one)/ ....`
	`(fit.best-fit.worst);`

```
            fitworst = 1        fitbest = 11
ObjFnvalue    6          7          1          5          3          11
    q         0.5        0.6        0          0.4        0.2         1
```

Chart 2(c): Pseudo code for calculation of force in a multi-agent system of charges

Input	Function	output
X, xminmax, q	ESForce	F

cal

for $i = [1,2,...nsol]$

for $j = [1,2,...nsol]$

$i1 = 1,\ i2 = 0 \quad if\ dist\_sep(i,j) < rad\_infl$

$i1 = 0,\ i2 = 1 \quad if\ dist\_sep(i,j) \geq rad\_infl$

$$F(j) = q(j) * \sum_{i=1, i\neq j}^{nsol} \left(\frac{q(i)}{radius^3} * dist\_sep * i1 + \frac{q(i)}{dist\_sep^2} * i2 \right) * prob(i,j) * \ X(i,:) - X(j,:)$$

$$Thus, F_j = fn \quad q, X \ , \ prob, \{ar\}$$

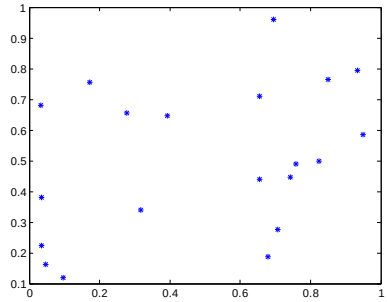
b. Charge of charged particle (CP): The quality of solution (fitness in GA, object function in mathematical optimization) is quantified as charge of the agent, here charged particle (CP). The charge of CP is calculated (Chart 2.b) by comparing the current fitness value with worst in the cycle and normalized with range between worst and best fitness values. It tantamounts to q is fitness vector scaled between zero and one, easing application of numerical techniques. The magnitude of the charge induces competition in the search process.

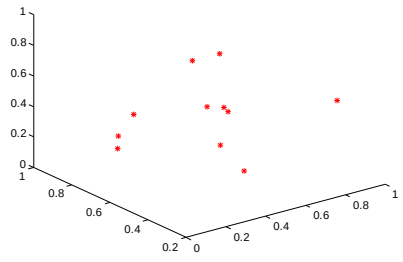
c. Refinement (updating/iteration) of position and velocity of CPs: CSS is a self starting multi-agent parallel search algorithm. The number of charges representing the solutions considered is nsol. The pseudo code for CSS (Chart 1.b) is described along with that major breakthrough of introducing magnetic forces of moving charges (Chart 1.c).

Initialization: An array of positions of charged particles (CPs) is initialized with uniform random numbers. The initial velocities of the CPs are taken as zero.

Table 2: Initiation and separation distance between charges

(a) Initiation





```
function [x0cpLU,x0cp01]= om_initcpx(Xminmax,nsol)
% xi,min : minimum allowed value  for ith dimension
% xi,max : maximum allowed values for ith dimension
%
 [ndim,col] = size(Xminmax);
uniform01 = 0
if  uniform01
    for j = 1:ndim
        Xminmax(j,2)=1;
        Xminmax(j,1)= 0;
    end
end
  Xminmax
x0cp01 = rand(nsol,ndim);
randx = x0cp01;
for i = 1:nsol
    for j = 1:ndim
        x0cpLU(i,j)= Xminmax(j,1) + randx(i,j) * (Xminmax(j,2)-
Xminmax(j,1));
    end
end
```

```
clean, Xminmax = [ 0 1; 0 100;0 1000]
nsol =6;
[x0cpLU,x0cp01]=  om_initcpx(Xminmax,nsol);
```

```
    0.4387   44.5586  655.0980
    0.3816   64.6313  162.6117
    0.7655   70.9365  118.9977
    0.7952   75.4687  498.3641
    0.1869   27.6025  959.7440
    0.4898   67.9703  340.3857

    dim1     dim2     dim3
```

(b) Distance of seperation

```
function [vel0]= initcpvel(nsol,ndim)
         vel0 = zeros(nsol,ndim);
```

$$dist\_sep(i,j) = \frac{\|X(i,:) - X(j,:)\|}{\left\|\frac{X(i,:)+X(j,:)}{2} - Xbest\right\| + eps}$$

Eqn.2

dist_sep (i, j)	:	separation distance betwee charged particles
X*i* and X*j*	:	positions of the *i* th and *j*th C
Xbest	:	position of the best current
eps	:	small positive number (1e- avoid singularity
$\|.\|$	:	*Eucledian norm*

```
X = [ 0. 0. ; 4. 3.; 8. 0.];
y = [11, 9, 1]';
[dist_Eucl,dist_sep] =om_distSep(X,y)
```

```
function [dist_Eucl,dist_sep] =om_distSep(X,objFnValue)
%
%   om_distSep.m    12/6/13
%
function [dist_Eucl,dist_sep]
=om_distSep(X,objFnValue)
%
[nsol,ndim] = size(X) ;
dist_sep    = zeros(nsol,nsol);
dist_Eucl   = zeros(nsol,nsol);
objFnValueXAsc = sortz([objFnValue,X],1,1)
Xbest          = objFnValueXAsc(nsol,2:ndim+1)

for  i = 1:nsol
%%%
    for j = 1:nsol
    %%%
        if i ~= j
        %%
           Xi = X(i,:); Xj = X(j,:);
           dist_Eucl(i,j) = norm (Xi - Xj) ;
           A = dist_Eucl(i,j);
           B = norm((Xi + Xj)/2-Xbest);
           dist_sep(i,j) = A/(B +eps);
           %
        end %%if
    end %%%j
end %%%i
```

```
    [ y      X]
     11      0       0
     9       4       3
     1       8       0
```

```
dist_Eucl =

     0      5      8
     5      0      5
     8      5      0
```

```
dist_sep =

    0    2.0000
2.0000
2.0000       0
0.8085
2.0000    0.8085
0
```

(c) m-function for calculation of radius of influence

```
Xminmax =

 [ 0    1
   0   10
   0  100]
```

```
function [rad_Infl] =radinfl(Xminmax)
%
[ndim,col] = size(Xminmax);
   for j = 1:ndim
      rad_Infl(j,1)= 0.10* ...
         max (Xminmax(j,2),Xminmax(j,1));
   end
```

```
rad_I

 0.1
 1.0
 10.0
```

rad_infl

+ It induces competition in the algorithm
+ Better fitness (great *qi*) can create a stronger attracting force, so the tendency to move toward a good more than toward a bad particle

Iteration: The magnitude of the charge (Eqn. 1) for each CP (agent), and the distance of separation between all pairs (Eqn. 2) are calculated (Table 2). Radius of influence (rad_infl) is set to unity or calculated by Eqn.3.

$$rad\_\text{infl} = 0.10 * \max \left\{ x(i,\max) - x(i,\min) \middle| i = 1,2,\ldots,nsol \right\} \qquad \text{Eqn.3}$$

i) Force between charges: Thus, Force is calculated for each CP (Chart 2c) taking into consideration of effect of all other charges (Fig 2) using superimposition principle.

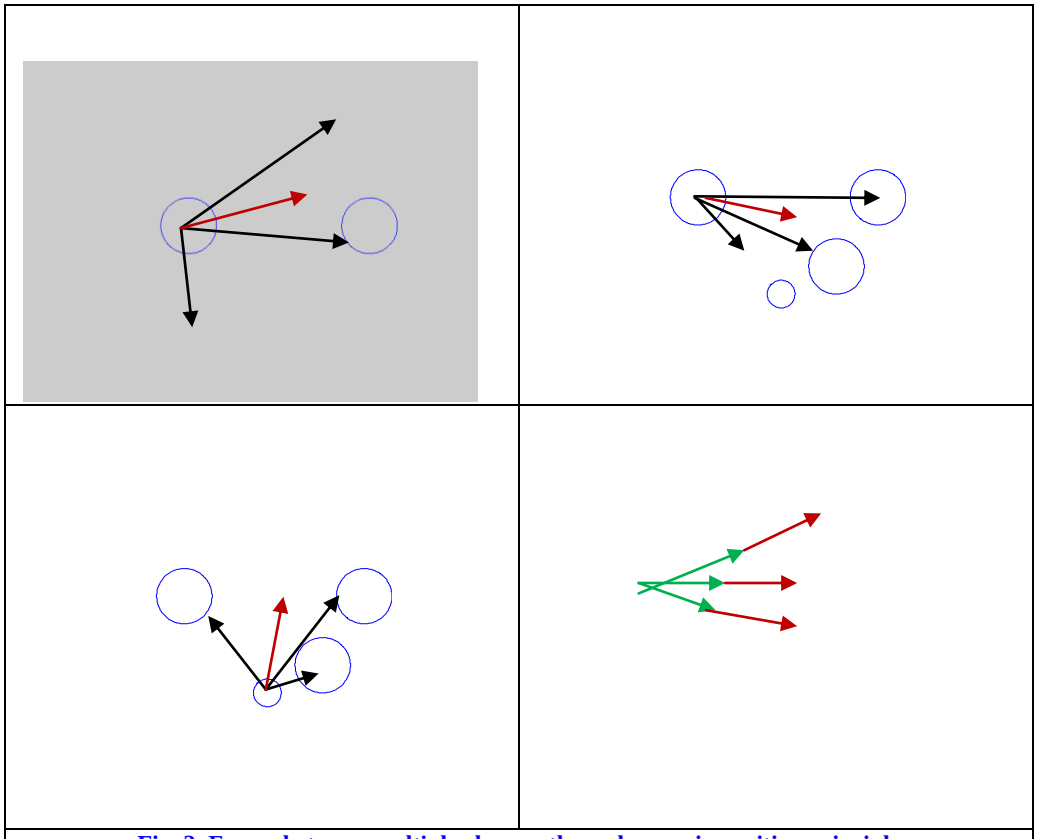


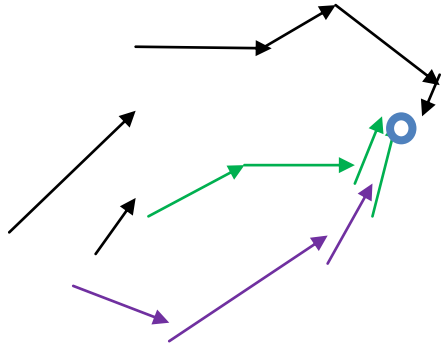
Fig. 2: Forces between multiple charges through superimposition principle

ii) Movement of position of CPs (solution): The new/refined/iterated position of each CP depends upon its velocity (Eqn. 4) in the previous iteration (Chart 3). The values of objFn, charges of refined CPs and current velocities are calculated. The q vector and their corresponding position matrix (Xiter) are sorted in ascending order.

Chart 3. Movement of agent with iteration

$$X(\mathrm{j},\mathrm{iter}+1) = X(j,iter) + \frac{F(j)}{q(j)} * \Delta t^2 + vel(j,iter) * \Delta t \qquad \text{Eqn.4}$$

q_j	:	Equal to mass of the j^{th} CP
Δt	:	time step set to unity (t_← 1)



iii) Terminating criteria for iterative refinement: The iterative cycle is terminated after a fixed number of iterations (iter_max), say 1,000, even if the convergence is not reached. Further, the refinement is discontinued, when there is no improvement in fitness function after some fixed number of consecutive iterations, minimum objective function error (difference between the values of the best objective function and global optimum) is less than a priori anticipated threshold value or difference between the objective values of the best and the worst CPs is less than a specified accuracy (Chart 4).

Chart 4 : Stopping criteria for iterative algorithms

(a) Iteration termination

```
Stop =  .T.

If      Iteration_current          < Max_it                    .or.

        it_without_Improvement     < Max_it_                   .or.
                                     withoutImprovement

Then    Stop _iter= .F.
Else    Stop_iter = .T.
```

(b) convergence

If	fn(obj _fn_value-refined – obj _fn_value-approximate)	< tol	.or.
	ABS (best_obj _fn_value – Glob_opt _value_of_ obj _fn)	< threshold	.or.
	Max_ distance between CPs	< 3 * radius_of sphere	.or.
	Heuristics specific to CSS		
	ABS(obj_fn_value_ best CP – obj_fn_value_worstCP)	< accuracy_expected	.or.
	maximum distance between CPs	< 3 * pre-defined-value	
Then	converge =.T.		
Else	converge =.F.		

(c) Iteration process discontinuation

```
If      converged or stop
Then    continue_iter = .No.
Else    continue_iter = .Yes.
endif
```

APPLICATIONS

A literature search of application of nature_mimicking_meta_heuristic_algorithms in chemistry and chemical engineering/technology grew exponentially in the last two decades. GA, EA, PSO, ACO algorithms found a niche in analytical chemistry, chemometrics, SXR, nano chemistry and nano medicine. HBF, HBMA, gravity, charge_syst_search, BBBC etc. reported in recent times have successfully solved non-linear programming (NLP) with better prospects in engineering. But, the publications in chemical sciences are scanty at the movement. A white box domain independent matlab programs comprising classical and E-man tools will open new vistas in pharmaceutical, environmental, drug discovery, nano sciences and proteomics. Kaveh and Laknejadi [8] extended CSS to multi objective optimization (*MOO*) tasks using clustering and particle regeneration procedures. CSS_MO successfully arrived at solutions for multi-modal object functions.

Chemical equilibria and kinetics: The interaction of a metal ion with polyprotic ligands in solution phase is in the realm of chemical equilibria including proton-ligand/metal-ligand/protonated-metal-ligand complexes. The overall formation constants (chemical parameters) of the equilibria (Chart 5) are determined with experimental data using glass-electrode / ion-selective-electrode/ spectrophotometric/ NMR/ ESR (instrumental) probes.

Chart 5. Overall formation constants of protonated metal complexes

$$m(j)M + l(j)L + h(j)H \xleftrightarrow{\beta} M_{m(j)}L_{l(j)}H_{h(j)}$$

$$\beta_{m(j),l(j),h(j)H} = \frac{\left[M_{m(j)}L_{l(j)}H_{h(j)}\right]}{FM^{m(j)} * FL^{l(j)} * FH^{h(j)}}$$

$m(j), l(j), h(j)H$	:	Stoichiometric coefficients
FM, FL, FH	:	Equilibrium (unbounded or free) concentrations of metal, ligand and hydrogen ions
$\beta_{m(j),l(j),h(j)H}$	:	Overall(cumulative) formation constant of complex $M_{m(j)}L_{l(j)}H_{h(j)}$

The parameters, conventionally, calculated or estimated by non-linear least squares using first and second derivatives (gradient [g], Hessian[H]) by analytical/numerical methods over last half a century. The feasibility study with nature-inspired E-man modules viz., CSS, honey-bee-mating/honey-bee-foraging algorithms showed promising results [29] and details will be published.

The time dependent formation of a chemical compound/complex is the core of chemical kinetics while that at molecular level is in the realm of chemical dynamics. The applications in organic synthesis, analytical estimations of industrial/pharmaceutical/clinical moieties require a fast progress and near completion of reaction. The research/pedagogical pursuits were initially confined to integrated form of rate equations wherein first and second order rate constants (k_1, k_2) were calculated by graphical/linear least squares fit. The non-linear equations (without integration) are exponential in chemical parameters (k) and concentrations of reactants. The unconstrained non-linear least squares (using Gauss Newton, Newton Raphson or Marquardt algorithms) had been in extensive use in chemical/biological/pharmaceutical sciences. This task in mathematical frame consists of a set of non-linear equations with complex multi-dimensional profiles, constraints on rate constants and concentrations. In our laboratory, work is progress in application of ant-colony-optimisation, PSO, BBBC, gravitational and charge system search algorithms in estimation and interpretation of rate constants.

Engineering : For the first time, CSS algorithm was applied to estimate economic power dispatch with prohibited operating zones and power generation limit. Kaveh and Talatahari [7, 14] obtained a viable result in optimum design of geodesic domes and the output of CSS was compared with those of PSO, ACO, HS, BBBC and hybridization of PSO with ACO.

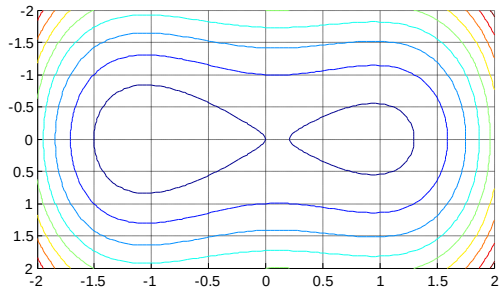
Mathematical functions : Charge_Syst_Serch is now a tool of choice for non-smooth or non-convex functions [19]. This algorithm was used in parameter identification of nonlinear- (NL-) differential equations (DEs) and Pareto optimal solution in economic power dispatch. A few typical systems optimized with Charge_Syst_Serch follow and three dimensional surfaces and 2D-contours are generated with m-files reported from this laboratory [29].

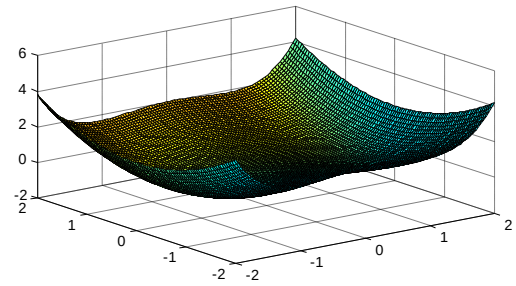
Function.polynomial: Aluffi-Pentiny function is a second order polynomial in two variables and Fig.3a depicts 2D-contour and 3D-surface.

Function.multiModal: Becker and Lago is a multimodal quadratic response function with only positive response values (Fig. 3b). But, the ranges of x ([-10 to 10]) in both dimensions (variables) cover negative to positive values.

Function.uniModal: Fig. 3c is a popular uni-modal single object function. It is an exponential of algebraic quadratic function in two (variable) dimensions.

Function.expoential: widened lamp, twisted-break and pot with a hole are exponential function in two dimensions, but posing varying difficulty in finding optima. The contours are described in Fig. 3d to Fig 3.f.

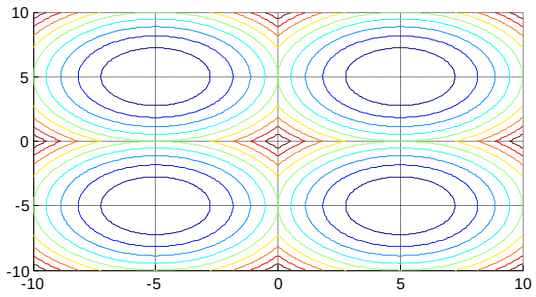


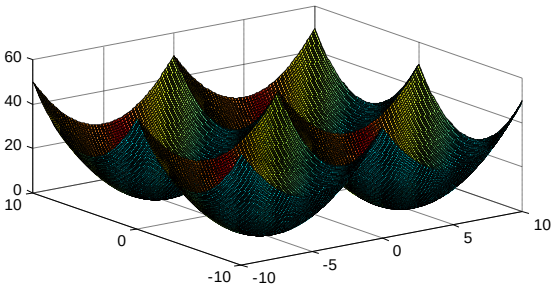


(a) Aluffi-Pentiny function

ll = -2.; ul = 2; inc = .04;

f1 = 0.25*x1.^4 -0.5* x1.^2 + x1/10 +0.5* x2.^2;

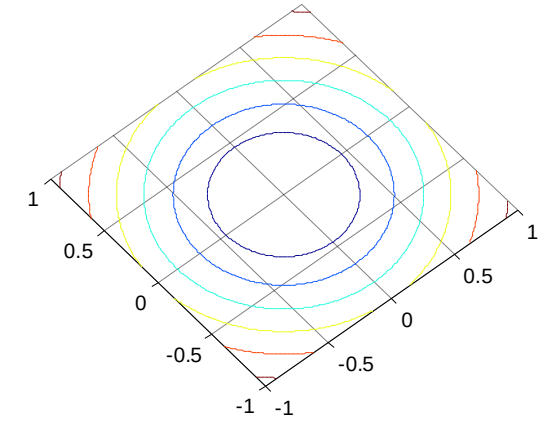


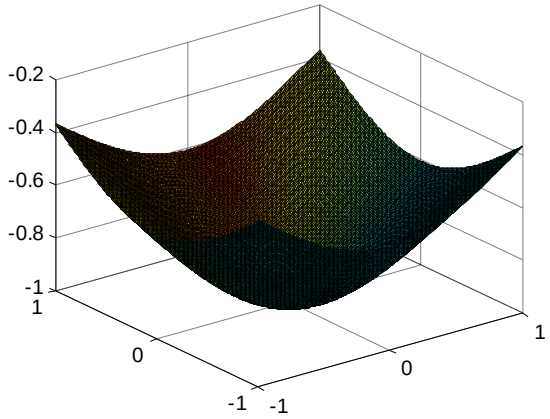


(b) Becker and Lago function

$$fn = (-1)^k * \left[\sum_{i=1}^{n\dim} a_i * |x_i| + b_i \right]^{power} \quad ; \quad k = 1 or 2$$

ll = -10.; ul =10; inc = .1;
f11= [abs(x1)-5].^2 +[abs(x2)-5].^2 ;

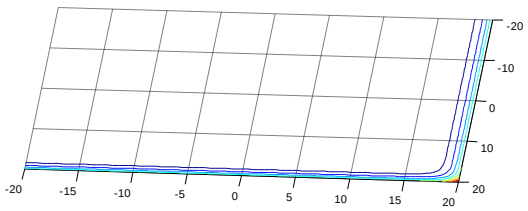


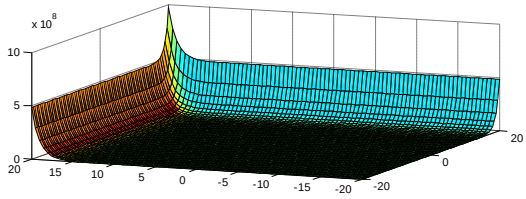


(c) Exponential function

$$fn = (-1)^k * \left[\sum_{i=1}^{n\dim} \exp\ a_i * x_i^2 \right] \quad k = 1 or\ 2$$

ll = -1.; ul =1; inc = .01;
f12= -exp (-0.5*x1.^2 -0.5*x2.^2) ;

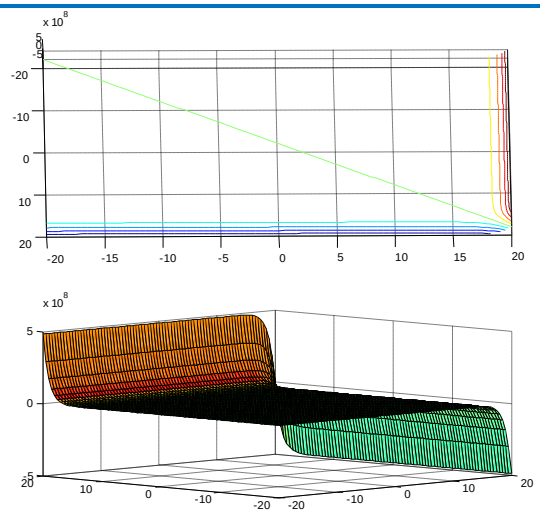




(d) Widened Lamp

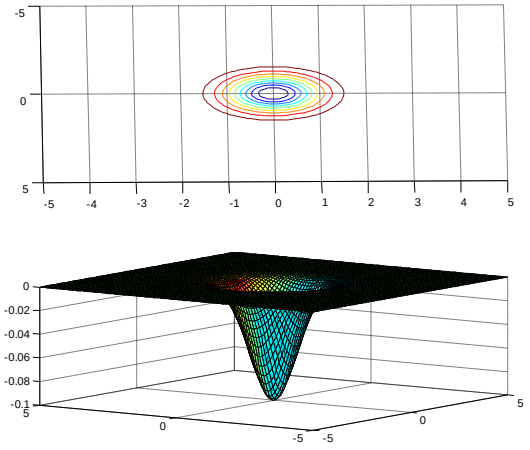
a=20; ll = -a; ul =a; inc = .5;

a1 = 1 ;a2 = 1;
f12= a1*exp (x1) +a2*exp (x2) ;



(e) Twisted function with a break

a=20; ll = -a; ul =a; inc = .5;
a1 = -1 ;a2 = 1;
f12= a1*exp (x1) +a2*exp (x2) ;



(f) Pot with a hole

a= 5;ll = -a; ul =a; inc = .1;
fn= -0.1*exp (-x1.^2 - x2.^2) ;

(g) Structure of function (Fn) base (Fnbase) in matlab

```
...AU Au AU Au >mm>>>>>mmm>>>>mm>>>mm>>

        fnName: 'Aluffi-Pentiny'
                ref: '11.'
             number: 11
             rangeX: [2x2 double]
               incX: [2x1 double]
          min_globX: [1x1 struct]
    min_globFnValue: 99
           min_locX: [1x1 struct]
     min_locFnValue: [3x1 double]
             inform: [1x1 struct]
        methodsUsed: ' '
                 fn: 'objFnValue =
0.25*x1.^4 -0.5* x1.^2 + x1/10 +0.5*
x2.^2'
```

RangeX: [-2.0 to 2.0 with inc of 0.04]

Aluffi Fn '0.25*x1.^4 -0.5* x1.^2 + x1/10 +0.5* x2.^2' is

Unconstrained multi modal Fn.

'Non-linear' ; 'algebraic'; 'Polynomial--order two'

FnValue at globMin is -0.352286

Fig. 3. 2D-contours and 3D-surfaces of typical functions and structure of function base

Hidden features of charge_syst_search

The features viz. self-adaptation/cooperation/competition and exploration/ exploitation are present in CSS algorithm [30]. Moving towards good CPs provides the self-adaptation step. A CP with higher charge (in other words larger fitness value of object function) results in larger force compared to that (bad) CP with lower charge. It should be noted that the heuristic induces the competition step of the algorithm. The magnitude of resultant attractive electrical force affecting a CP is calculated based on cooperative effort of CPs is a compromise between exploration and exploitation.

Advantages and limitations of charge_syst_search

The improvement in core algorithm progresses with finding limitations and fixing them at the first stage followed by integration and even optimizing code from scratch. Harmony search algorithm, probability factor, coefficient for acceleration/ velocity and effect of magnetism are introduced with basic CSS (Chart 6) rendering it more powerful.

Chart 6: Critical view of functioning of CSS

(a) KB for characteristics of CPs and consequences

True		
Agents (ith jth) attract each other $prob_{i,j} = \begin{cases} 1 & if \ \ fit(j) > fit(i) \\ 0 & else \end{cases}$	**if**	qi better than other (qj)
Adaptation principle operates & improves its performance	**if**	CP moves towards a good agent
Exploitation ability operates	**if**	a good_CP attracts a bad_CP
Exploration operates	**if**	a bad_CP attracts a good_CP
strong exploration & efficient exploitation	**if**	Charged memory tensor used

Chart 6(b): Advantages of CSS

+ A good balance between exploration and exploitation
+ A good global and at the same time a local optimizer

Chart 6 (c): Limitations of CSS and remedial measures

- Repulsive forces (which deteriorate convergence), ignored
  - Remedy: Discrete CSS
- Moving a good CP toward a bad one may cause losing the previous good solution or at least increasing the computational cost to find a good solution
  Remedy: Probability factor
- CPs move in search space at each iteration. A moving charged particle produces a magnetic field which has influence on other CPs, which is not considered in CSS
  - Remedy: Mag_charged_system_search
- Position of refined CPs may be out of range of limits of the variables in allowable search space
  - Remedy: HS-based CP position correction
- Imbalance of exploration and exploitation
  - Remedy: variation of coe_acc and coe_vel

Modifications of CSS

a) Repulsion between CPs : The outcome of simple laws of electrostatics is that each CP influences all others which tantamount to that a bad CP (with low charge value) also can affect a good one (possessing high magnitude of charge) and vice versa. Thus, the probability of movement of a CP (i,j) is one (Eqn. 5). But, bad agents attracting good agents is undesirable, which is surmounted by restricting the probability (Eqn. 6) through the use of a random number.

b) Correction for movement of CPs : The direction of velocity of a CP in the preceding iteration may differ from that of the resultant force.

i)Coefficient of velocity (coe_vel or kv): A control parameter called velocity coefficient (coe_vel or kv) is a linearly decreasing function to zero as iterations proceed (Eqn. 7). It controls exploration by influencing the previous velocity paving way to convergence. Thus, coe _ vel balances righteously the exploration and the fast rate of convergence (Chart 7).

$$prob_{i,j} = 1 \vee q > 0 \qquad \text{Eqn. 5}$$

$$prob_{i,j} = \begin{cases} 1 & if \ \dfrac{fit(i) - fitbest}{fit(j) - fit(\mathrm{i})} > rand \\ & \vee \ fit(j) - fit(\mathrm{i}) \\ 0 & else \end{cases} \qquad \text{Eqn. 6}$$

Chart 7: Coefficient of acceleration and velocity of CPs in iteration process

$$coe\_vel = 0.5 * \frac{1 - iter}{iter\_\max} \qquad \text{Eqn. 7}$$

$$coe\_acc = 0.5 * \frac{1 + iter}{iter\_\max} \qquad \text{Eqn. 8}$$

If acc_coe is large
Then Fast convergence but coarse solution

If acc_coe is small
Then slow convergence & high CPU time

If acc_coe = 1
Then quality of solution is accepted in toto

```
%
 function [ vel_coe,acc_coe ]= coe_velacc(iter,iterMax)
vel_coe = 0.5 * (1-iter)/iterMax
acc_coe = 0.5 * (1+iter)/iterMax
```

$$X(\mathrm{j},\mathrm{iter}+1) = X(j,iter) + rand(j,1)*acc\_coef * \frac{F(j)}{q(j)} * \Delta t^2 + rand(j,2) * vel\_coef * vel(j,iter) * \Delta t \quad \text{Eqn. 9}$$

q_j	:	Equal to mass of the j^{th} CP
Δt	:	time step set to unity (t_← 1)

$$vel_j = fn\ X\ , \{\Delta t\}$$

ii) Coefficient of acceleration (coe_acc or ka): Excessive search in the early iterations is indispensable to improve the exploration ability. But, it should decrease gradually for a thorough search in the neighborhood of accurate solutions. Coefficient of acceleration (Eqn. 8) is a control parameter for the exploitation operated on the resultant force on a CP. This is an incremental function increasing to unity with number of iterations. The result is improvement of the performance. The chart 7 describes the formulae and knowledge bits in calculation of coe_acc and coe_vel during iteration and new position of CPs.

c. Charged memory tensor (ChMem): It is like a backup memory storing local best positions of the agents and corresponding objFn values till completion of that iteration. The CPs in ChMem influence (attract/repel) other CPs and determines the extent and direction of the movement of CPs. It increases the exploitation facet of the algorithm and results in powerful search strategy. It is two dimensional matrix with (nsol/4) rows or increasing linearly with iterations. The refined CPs are in the current iteration replace worst ones in the ChMem. In the process of storing good CPs in ChMem and updating in each iteration is a competitive step (Chart 8) amongst prospective CPs. A balance between exploration and exploitation further increased by choosing two more agents from ChMem along with those in the current iterations.

Chart 8. Benefits and shortcomings of Charged memory

\+ Improves performance
\+ Guides CPs in the direction of global optima
– Increase the computational cost
▶ Remedy: Constant number of the worst particles

Recent advances in charge_syst_search research

a. Discretised CSS: In chemical sciences, discrete search space search is in the wavelength selection, multivariate calibration, choice of molecular descriptor in SXR etc. Kaveh and Talatahari [10] reported discretised CSS to optimize truss structures in civil engineering. The floating point value of X is rounded to nearest integer. In the standard CSS, only attractive forces were taken into account. But, here the repulsive forces are also considered improving efficacy of algorithm. Further, fence crossing i.e. fly off from the boundary of the search spaces is taken care of. If a CP escapes from the allowable search space or swerves off the predefined bounds, its refined position is corrected or brought back within the boundary using the harmony search, another nature inspired technique. Kaveh and Talatahari [11] employed fly-to-boundary approach to avoid the solution crossing the fence of feasible regions in the hybrid PSO-CSS algorithm.

b. Adaptive charged system search: Talatahari et al [19] applied adaptive CSS to identify non-smooth/non-convex regions in parametric models. Niknam et.al [32] enhanced the functioning of standard CSS by considering the optimal solutions found by each particle in the previous iterations. The best Pareto optimal set of a multiple objective (MO) minimization of a mixed integer nonlinear problem in planning the location and operation of Fuel Cell Power Plants (FCPPs) is determined. The Multiple Objectives in this task are minimization of total cost, emissions of FCPPs and voltage deviation.

c. Enhanced CSS (ECSS): Niknam et al. [31-34] introduced a novel mutation strategy enhanced CSS which increases the population diversity as well as altering convergence criteria. The method is applied to small- and large-scale reserve constrained dynamic optimal power flow with 30 and 118 buses test systems.

d. Chaotic charge_ syst_search algorithm: Talatahari et al. [20] substituted random search by chaos in extended-CSS. The application of chaos at several stages of CSS for constrained mechanical design and simulation endorses efficiency of the chaotic_CSS.

e. Breakthrough strategies in CSS: In the basic CSS, the movement of CPs to the refined position are determined after the completion of calculation of forces. It is followed by updating of ChMem matrix. This procedure is referred as a discrete time step here, while in optimisation procedures the concept is popular as iteration. It is important to note that the modifications of time-space for multi-agent algorithms are done when the iteration is complete. However, at this point the new iteration is not started yet.

i) Continuum time model: In the enhanced CSS, the time is considered as continuous and is on as soon as initial CPs or approximate solutions are created. The updating processes are also performed on a continuous time basis, which means the current position of the agent now affects the movement of other CPs. This is in contrast to standard CSS where the effect is not felt until the iteration is complete and new positions already calculated are not made use.

ii) Magnetic charged system search **(Mag_charge_syst):** In this year, Kaveh et.al. [27] proposed magnetic charged system search by incorporating magnetic forces using Biot-Savart law in addition to electrical forces in CSS. The only difference is an additional term in calculation of force (Chart 1c). If a CP traverses/moves in space, a magnetic field is created. This magnetic field exerts magnetic forces on all other CPs. The outcome of Mag_charge_syst is noteworthy and its comparison with other nature mimics and software will be separately detailed [29b].

f. Hybrid charge_syst_search (Hybrid_CSS): Recently, CSS algorithm is hybridized with harmony search [31], PSO [11, 15, 24] and BB_BC [18] with remarkable results. This wave of binary and ternary hybrid systems and incorporating new features from diverse paradigms continues and a breakthrough of hyper-intelligent self adaptive fault tolerant problem solver is awaited. An important observation is that both components are from nature inspired (NI) natural intelligence (NI) category.

i) *CSS + PSO:* Kaveh and Talatahari [24] brought out the best of both worlds viz., CSS and PSO. The charged memory consists of the fitness values in the frame work of electrostatic laws of Coulomb and Gauss. The PSO philosophy brings out the advantage of search using local and global best positions. The same school [15] incorporated CSS in PSO in a novel strategy of selecting best global particle from a set of Pareto optimal solutions. It is applied for MOO where in diversity of search as well as convergence of the solution improved.

ii) *CSS + BB_BC:* Kaveh and Zolghadr [18] proposed a binary hybrid E-man combining BB_BC with CSS. The unique feature is its' local trap (minimum) recognition ability. It is efficient for solutions of non-convex functions with multiple local optima.

iii) Harmony search + CSS: Kaveh and Hosseini [31] hybridized CSS with harmony search (Harm_Serch) which improves the exploitation property of CSS. Here, new goal functions are proposed for continuous optimization and the success of the method is tested with benchmark functions.

Fields of forces (FOF)

Talatahari, profounder of CSS [6] put forward a new version CSS algorithm based on Generalized-fields-of-forces [17]. This concept unifies different meta-heuristic algorithms viz. ACO, PSO and CSS under the same roof. In the continuous- space-time mode of refinement of solutions, all updating processes are performed after creating just one solution (here CP). The mimicking characteristic is akin to Levenberg-Marquardt (LM) algorithm in gradient based optimization and continuous regression (CR) in soft regression. LM algorithm emulates steepest descent, first derivative (gradient) and second derivative (hessian) based methods. CR (continuum regression) adaptively behaves like multiple linear regressions (MLR), principal component regression (PCR), partial least squares regression (PLSR) and modified_CR mimics even multi-layer-perceptron (MLP) neural network (NN). A white box approach of algorithms and

MATLAB functions with SAP (simple as possible) numerical examples promote their routine use in chemical sciences both in research and pedagogical activity.

Future scope

A Scopus search for Charge_Syst_Serch (CSS) algorithm (introduced in 2010) showed around fifty publications till date dealing with applications. The rapid progress in modifications of the method, hybridization with other swarm intelligent techniques like PSO, ACO, BB_BC opened windows for rapid utilization in chemical and biological sciences. The novel extension of CSS with inclusion of magnetic forces foresees its' sure to fire characteristic. The development and application of ternary hybrid systems in sequential, hierarchical or fusion fashion will be the future focus. The geometry representation of tensorial operations in generalized Fields of forces model and deriving Charge_Syst_Serch, grav_alg etc. offers a deep view. The state of art of CSS in research mode awaiting full implementation is in Chart 9 along with typical abbreviations in Chart 10. A parallel implementation of the set of E-man tools for a task on hand will be a reality with open ended academic software. The in house function base (from this laboratory) consisting of around one hundred standard mathematical functions with 3D-surfaces and 2D-contours is under rigorous testing phase. Inter- and intra- disciplinary research probes by experts with "outside the box" knowledge and higher order skills in diverse paradigms with a focus of looking forward for something different will catalyze evolution to realize strategic out-of-the-box tools rather than improvising sure-to-fire time-tested-miracles (Fig. A3).

Chart 9. State-of-art-of-CSS in research mode

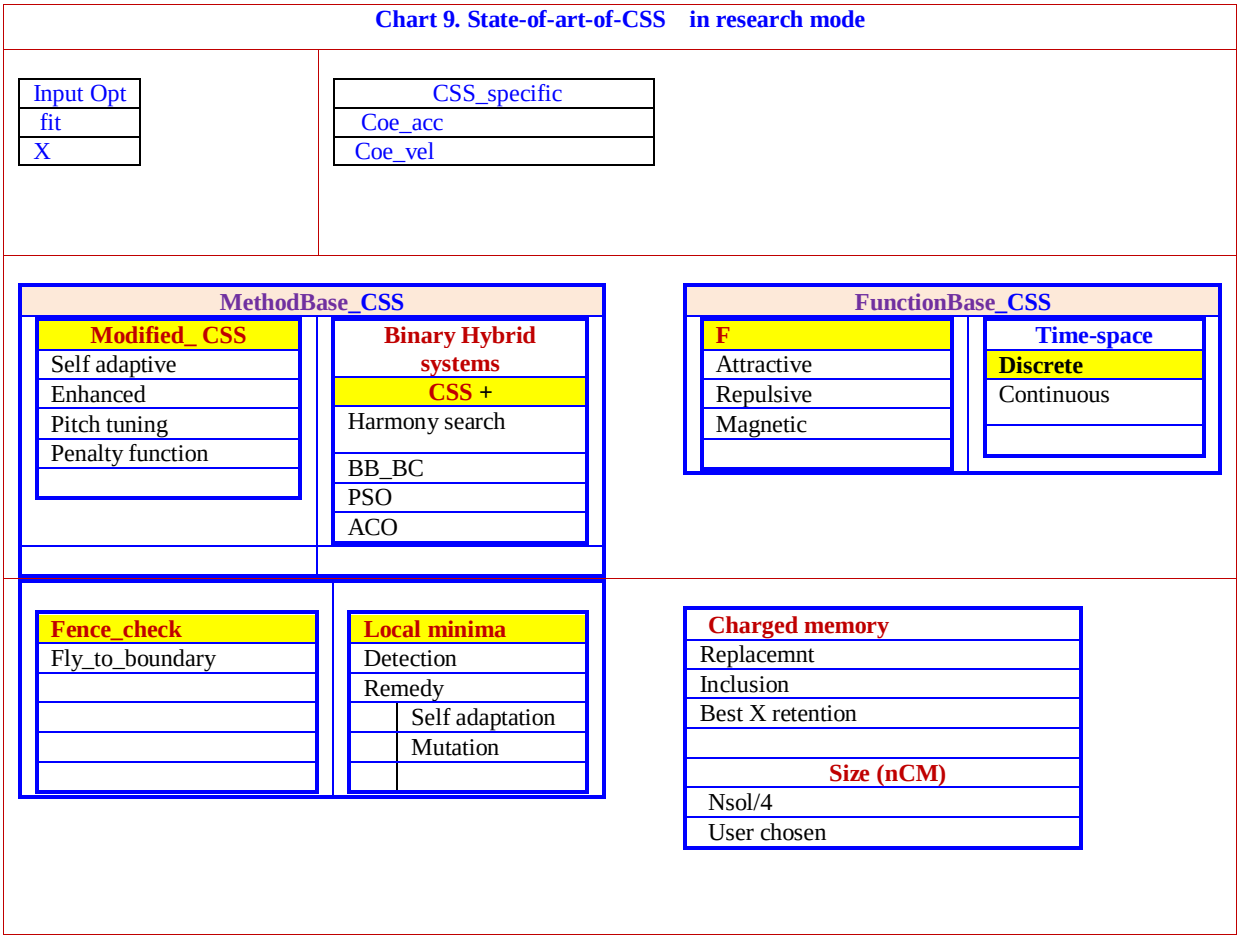


Chart 10: Abbreviation in nature inspired search / optimization algorithms

Abbreviation		Acronym
ACO	:	Ant colony optimisation
BB_BC	:	Big Bang Big crunch

Harmo.serch	:	Harmony search
PSO	:	Particle swarm optimization
HBMA	:	honey-bee-mating alg
HBFA	:	honey-bee-foraging alg
CSS	:	Charged system search (Charge_Syst_Serch)
		Optimisation using Fn (function), g(radient) and/or H(essian)
GN	:	Gauss-Newton method
NR	:	Newton-Raphson technique
LM	:	Levenberg-Marquardt algorithm

Appendix-1: Generalized *force laws*

Table A001: Newton's generalized inverse square law

For point masses / charges / magnetic _ particles			Eqn.
$Force(p_i, p_j) = \frac{magnitude_i * magnitude_j}{\left[dist\ \ x_i, x_j\right]^2} * forceStrength$		$Force(p_i, 1) = \frac{magnitude_i}{\left[dist\ \ x_i, x_j\right]^2} * forceStrength$	
Masses *m1,m2*	$FGrav_{i,j} = \frac{m_i * m_j}{x_i - x_j^{\ 2}} * G$	$FGrav_{i,1} = \frac{m_i}{x_i - x_j^{\ 2}} * G$	A01
charges q1,q2	$FCharge_{i,j} = \frac{q_i * q_j}{x_i - x_j^{\ 2}} * Ke$	$FCharge_{i,1} = \frac{q_i}{x_i - x_j^{\ 2}} * Ke$	A02
magnetic monopoles *M1,M2*	$FMag_{i,j} = \frac{M_i * M_j}{x_i - x_j^{\ 2}} * U$	$FMag_{i,1} = \frac{M_i}{x_i - x_j^{\ 2}} * U$	A03
$dist\_sep = Eucledian\_distance = \left\|x_i - x_j\right\| = \sqrt[2]{\sum_{d=1}^{n\dim} x_{i,d} - x_{j,d}^{\ 2}}$		x_i : coordinates i^{th} position x_j : coordinates j^{th} position	
If ndim = 1, then $norm(x_i - x_j) = \sqrt[2]{\sum_{d=1}^{1} x_{i,d} - x_{j,d}^{\ 2}} = \sqrt[2]{x_i - x_j^{\ 2}}$ $= x_i - x_j$			

Newton's gravitational (force of) attraction exerts between (materialistic) bodies even in empty space (without any material intervention) and even at very large distances (Table A001). Coulomb's electrostatic law proposed in 1785 accounts for the force of interaction (attraction/repulsion) between charged particles/bodies. For example the space around an electric charge possesses electric field. The mutual forces of electrically charged objects on other objects are explained by Coulomb's law. A moving charged particle exerts magnetic fields and quantified by Biot–Savart law.

Particle masses -Gravitation

If Fundamental natural force between masses considered
Then Newton's Law of Gravitation

If Newton's Law of Gravitation
Then (Gravitational) force is always positive (attraction)

Magnetic poles - Biot–Savart law

If Fundamental natural force between magnetic poles considered
Then Biot–Savart law

Charged particles -Coulomb's Law

If Fundamental natural force between charges considered
Then Coulomb's Law

If q1 * q2 > 0
Then Two charges repel each other
Else **if** q1 * q2 < 0

then The charges attract one another

else **if** ABS(q1) =0 | ABS(q2) =0
then Columb's Law is not applicable

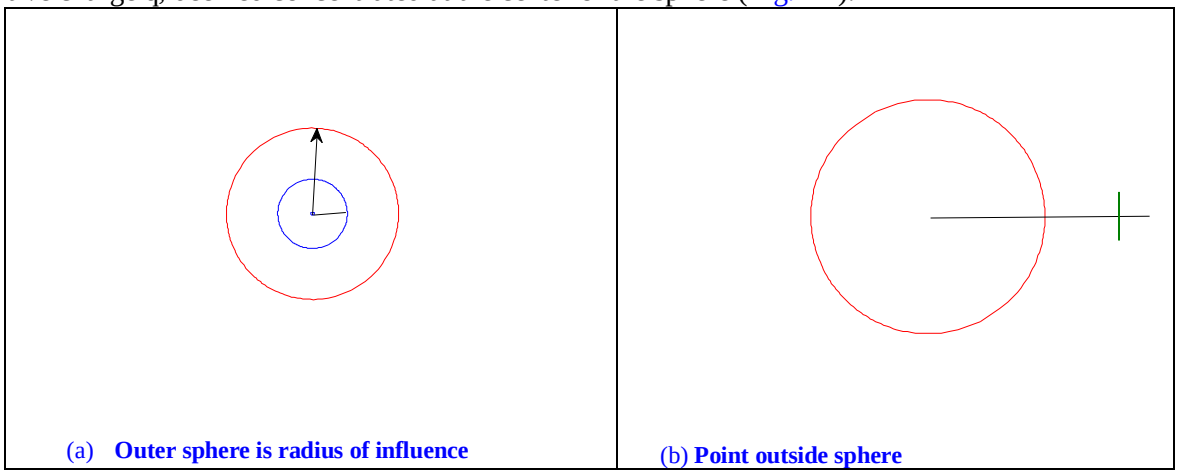
endif

endif

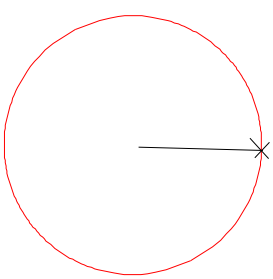
endif

Appendix-2: Electrostatic laws of physics

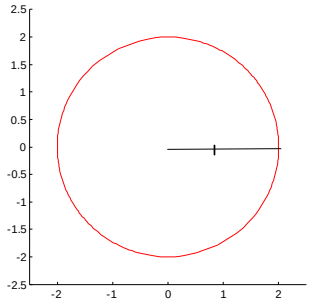
Consider an insulating solid sphere of radius (rad), with a uniform volume charge density. It carries a total positive charge q, deemed concentrated at the center of the sphere (Fig. A1).



(a) **Outer sphere is radius of influence** (b) **Point outside sphere**



(c) Point on the sphere



(d) Point inside sphere

Fig. A1. Single charged sphere

Coulomb law Electric field at a point outside the sphere: Coulomb's law states that magnitude of the electric force (Coulomb force) between the two point-charges is directly proportional to the product of the magnitudes of charges of the two particles and inversely proportional to the square of the distance of separation between the particles. The direction is along the line joining them (Fig A2). If the second charge is of unit magnitude, the electric field $E_{i\,j}$ at a point outside the sphere is given by Eqn. A02. As the point moves nearer to the centre, the field increases and equation tends to become singular corresponding to a physical impossibility. Gauss proposed a model to calculate electric field inside the changed sphere which is in conformity with physical phenomenon and follows Coulomb's law and Superposition Principle. It is applicable if the charge is continuously distributed within sphere, cylinder, or plane which have symmetrical geometry.

$$E_{i,j} = k_e * \frac{q_i}{dist\_sep^{\,2}}$$

- If $dist\_sep_{ij} = 0$, equation is singular.
- If $dist\_sep_{ij} \rightarrow 0$, then a physical impossibility ($E_{ij} \rightarrow inf$) arises
  - Remedy : **Gauss law**

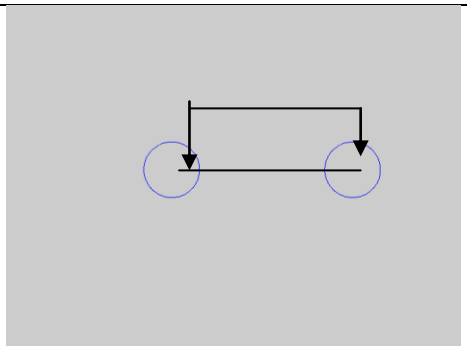


Fig. 2(a) Distance between centers of spheres

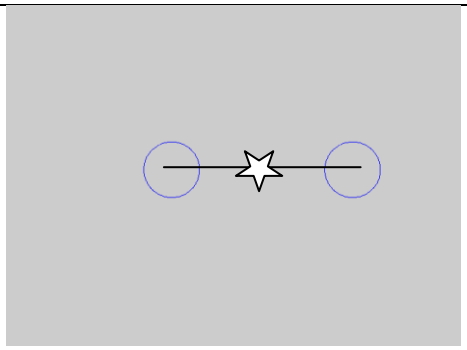


Fig. 2 (b)Point between the centers

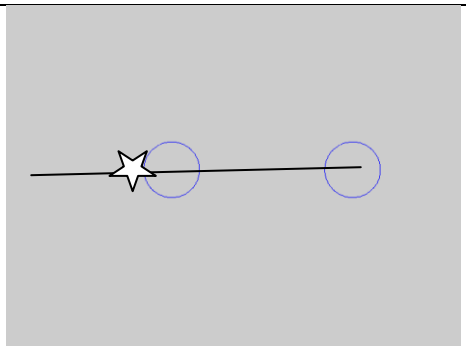
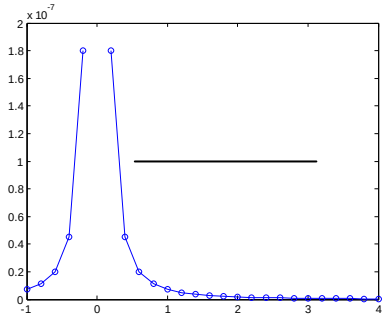


Fig. 2 (c) Point outside the centers

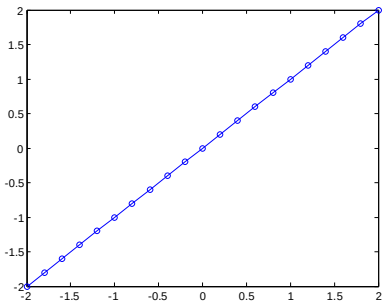



```
%
% om_Fq1q2.m        10/6/13
%
function [FCoulomb]= Om_coulomb(q,dist_sep,kel)

[row,col]=size(dist_sep);

for i = 1:row
   FCoulomb(i,1)=
kel*q(1,1)*q(2,1)/[dist_sep(i,1)^2];
end
```

Fig A2(d): Coulomb law




```
%
% om_Gauss.m        10/6/13
%
function [FGauss]=
Om_Gauss(q,dist,kel,radius_sphere)

[row,col]=size(dist);

for i = 1:row
   FGauss(i,1)= kel*q(1,1)*q(2,1)/radius_sphere.^3 *
dist(i,1);
end
```

Fig A2(e): Gauss law

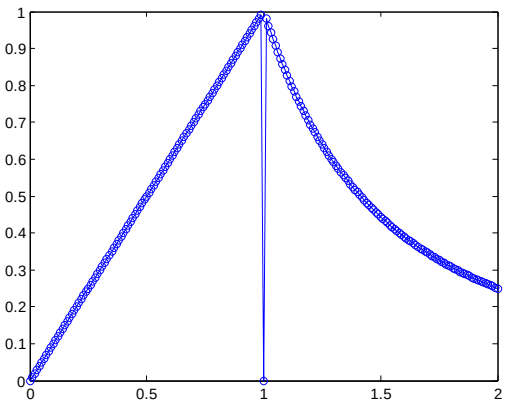


Fig A2(f): Integrated Coulomb-Gauss_law

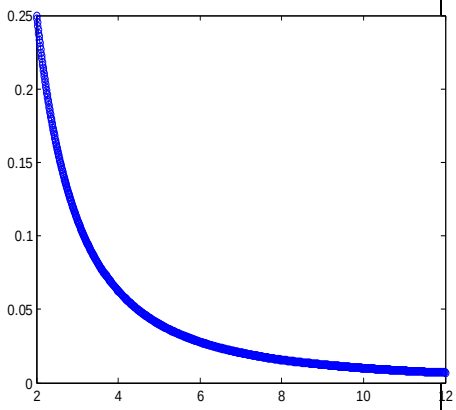


Fig A2(g): Asymptotic influence with distance

```
%
clean
%
kel = 1;
radius_sphere = 1;  dist = [0:0.2: 4]'; q = [1;1]
%
[Fq1q2]= Om_Fq1q2(q,dist,kel,radius_sphere);
%
plot(dist,Fq1q2,'bo',dist,Fq1q2)
```

```
%
% om_Fq1q2.m        10/6/13
%
function [Fq1q2]=
Om_Fq1q2(q,dist,kel,radius_sphere)

[row,col]=size(dist);

for i = 1:row
    %
    if dist(i,1)<radius_sphere
       Fq1q2(i,1)=
kel*q(1,1)*q(2,1)/radius_sphere.^3 * dist(i,1);
    else
       Fq1q2(i,1)= kel*q(1,1)*q(2,1)/dist(i,1)^2;
    end
    %
end
```

Fig A2(h): matlab code

			force	direction		q2 = 1		
q1	q2	F		q1	q2	E		
+5	+4	-10*ke*dist	Repel	←	→	-2.5*ke*dist	←	→
+5	-4	+10*ke*dist	Attract	→	←	-2.5*ke*dist	←	→
-5	+4	+10*ke*dist	Attract	→	←	+2.5*ke*dist	→	←
-5	+4	-10*ke*dist	Repel	←	→	+2.5*ke*dist	→	←

Fig A2(i): Illustration of interaction of two charges

Gauss's law Electric field at a point inside the sphere: The magnitude of the electric field at any point inside a charged sphere is proportional to the separation distance between CPs.

$$E_{i,j} = k_e * \frac{q_i}{radius^3} * dist\_sep$$

+ If $E_{ij} \rightarrow 0$ as $dist\_sep_{ij} \rightarrow 0$, singularity difficulty of coulomb's law surmounted.
+ If $dist\_sep_{ij} = 0$, the point is on the sphere.
+ Approach of surface either from outside or inside the sphere is feasible

Electric field at a point due to a group of objects: The magnitude of the field at a point due to a group of objects is obtained by using the superposition principle.

Appendix-3: Newtonian mechanics

Assumptions: A particle of definite mass is of infinitesimal size. In other words, the mass of spherical body is deemed that its' mass is concentrated at the center. By tracing position of the particle in space with progress of time, the motion profile of the particle is completely known. Of course, this is valid iff (if and only if) the object is small compared to the distance of separation. An advantage is taken in astronomical calculations of orbital motion of a planet around a star even though it is idealized as a particle.

Similarly a charged spherical body is also considered as if the entire charge is located at the center of the sphere. This facilitates the classical analysis of motion irrespective of the size and shape of the interacting moieties.

Newton's second law: Newton's second law explains that the acceleration of an object of mass *m* is directly proportional to the net force acting on that object.

Force = *mass* * acc(eleration)

The displacement of a particle is its change in position as a function of time. The slope of tangent line of the particle position represents the velocity of this particle. The change of velocity is acceleration. If Δt is very small, the displacement of any particle can be obtained approximately by Eqn. A11.

Physical quantity	Finite difference	Calculus	
		Differentiation operator $\lim_{\Delta t \rightarrow 0} \frac{\Delta z}{\Delta t} \rightarrow \frac{d}{dt}(z)$ or $\dot{z}$; $\frac{d}{dt}\left(\frac{dz}{dt}\right) \rightarrow \frac{d^2}{dt^2}$ z or $\ddot{z}$	
Displacement Velocity Acceleration Force	$disp = x_F - x_B = \Delta x$ $vel = \frac{x_F - x_B}{t_{new} - t_{old}} = \frac{\Delta x}{\Delta t}$ $acc = \frac{vel_F - vel_B}{t_{new} - t_{old}} = \frac{\Delta vel}{\Delta t}$ $force = mass * acc$	$vel = \lim_{\Delta t \rightarrow 0} \frac{\Delta x}{\Delta t}$ $acc = \lim_{\Delta t \rightarrow 0} \frac{\Delta vel}{\Delta t}$ $force = mass * acc$	$= \frac{dx}{dt}$ $= \frac{dvel}{dt} = \frac{d^2x}{dt^2}$ $= mass * \frac{dvel}{dt} = mass * \frac{d^2x}{dt^2}$

$$x_{new} = x_{old} + \Delta x \quad \text{… A10}$$

If Δt is small

Then $$\Delta x = x_{old} * \Delta t + \frac{1}{2} * acc * \Delta t^2 \quad \text{… A11}$$

Appendix-4: Reference database structures

Structure	Typical record	
authorL: [11x20 char] authorF: [11x10 char] year: 2010 volume: 213 journal: 'Acta Mech' pageBeg: 267 pageEnd: 289 title: 'A novel heuristic optimization method: charged system search'	ref(1).authorL(1,:)= 'Kaveh ' ref(1).authorF(1,:) = 'A.' ref(1).authorL(2,:)= 'Talatahari' ref(1).authorF(2,:) = 'K.' ref(1).year = 2010 ref(1).volume = 213 ref(1).journal= 'Acta Mech' ref(1).pageBeg= 267 ref(1).pageEnd= 289 ref(1).title= 'A novel heuristic optimization method: charged system search'	Last name of author First name of author Beginning page number Last page number
	css.ref = ref	

Model output

!!!!!!!!!!!!!!!!!!!

In 2010 Talatahari et. al. proposed CSSAlg

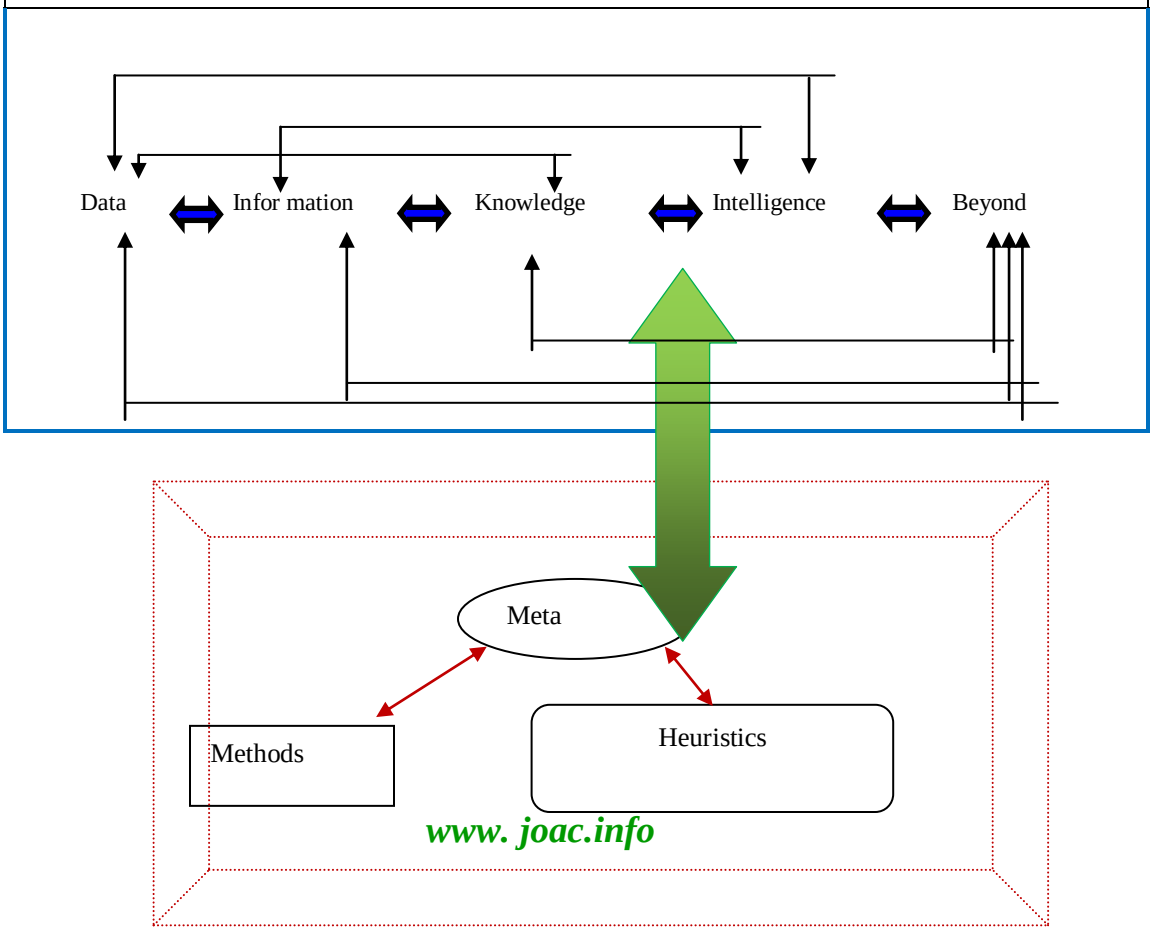
The inspiration is from Laws of Physics by Coulomb, Gauss, Newtonian Mechanics

??????????????????

1. A.Kaveh, K.Talatahari, ,
   Acta Mech, 213(2010)267-289
   A novel heuristic optimization method: charged system search

2. A.Kaveh, K.Talatahari, ,
   Acta Mech, 213(2010)267-289
   A novel heuristic optimization method: charged system search

Fig.A3: A Data to intelligence generation cycle



REFERENCES

[1] C. Blum, *Physics of Life Reviews*, **2005,** 2, 353–373.
[2] S. Chowdhury, G.S. Dulikravich, *Structural and Multidisciplinary Optimization*, **2010,** 41(4), 541-554.
[3] S. Chowdhury, G.S.Dulikravich, R.J. Moral, *Int. Journal of Mathematical Modelling and Numerical Optimisation* **2009**, 1 (1-2), 1-38.Y.M. Cheng, L. Li, *Comput Geotech*, **2007**, 34(6), 462–484.
[4] T.P. Hong, Y.F. Tung, S.L. Wang, Y.L. Wu, M.T. Wu, *Information Sci*., **2012**,182,3-14.
[5] Y.Shi, H.C.Liu, L.Gao, G.H. Zhang, *Infor. Sci*., 2011, 181, 4460-4493.
[6] A. Kaveh, S.Talatahari, *Acta Mech*, **2010**, 213, 267–289.
[7] A. Kaveh, S.Talatahari, *International Journal of Space Structures*, **2010**, 25(4), 217-227.
[8] A. Kaveh, S.Talatahari, *Structural and Multidisciplinary Optimization*, **2010**, 41, 893–911.
[9] A. Kaveh, S.Talatahari, *Structural and Multidisciplinary Optimization*, **2010**, 1-13.
[10] A. Kaveh, S. Talatahari, *Asian Journal of Civil Engineering* **2010**, 11(3), 277-293.
[11] A. Kaveh, S.Talatahari, *Engineering Computations* (Swansea, Wales), **2011**, 28(4), 423-440.
[12] A. Kaveh, S.Talatahari, *Structural and Multidisciplinary Optimization*, **2011**, 43, 215–229.
[13] A. Kaveh, S.Talatahari, *Structural and Multidisciplinary Optimization*, **2011**, 43, 339–351.
[14] A. Kaveh, S.Talatahari, *Structural and Multidisciplinary Optimization*, 2011, 43 (2), 215-229.
[15] A. Kaveh, K. Laknejadi, *Expert Systems with Applications*, **2011**, 38 (12), 15475-15488.
[16] A. Kaveh, P. Zakian, *J. Constructional Steel Research*, **2013,** 82, 111-130.
[17] A. Kaveh, S. Talatahari, *Acta Mech*, **2011**, 221, 99–118.
[18] A. Kaveh, A.Zolghadr, *Computers and Structures*, **2012**, 102-103, 14-27.
[19] S. Talatahari, A. Kaveh, N. Mohajer Rahbari, *J. Mech. Sci. Tech.*, **2012**, 26 (8), 2523-2534.
[20] S. Talatahari, A. Kaveh, R. Sheikholeslami, *Acta Mech*, **2012**, 223, 2269–2285.
[21] A. Kaveh, S. Talatahari, *Applied Soft Computing Journal*, **2012**, 12 (1), 382-393.
[22] A. Kaveh, S. Talatahari, *Applied Soft Computing*, **2012**, 12, 382–393.
[23] A. Kaveh, S. Talatahari, B. Farahmand Azar, *Iranian J. Sci. Techn. – Trans. Civil Eng.*, **2012**, 36 (C1), 67-77.
[24] A. Kaveh, S. Talatahari, *Structural Engineering and Mechanics*, **2012**, 42 (6), 783-797.
[25] A. Kaveh, A.F. Behnam, *Scientia Iranica*, **2012**, 19 (3), 410-416.
[26] A. Kaveh, A.Zolghadr, *Computers and Structures*, **2012**, 102–103, 14–27.
[27] A. Kaveh, Mohammad A. Motie Share, M. Moslehi, *Acta Mech*, **2013**, 224, 85–107.
[28] A. Kaveh, K. Laknejadi, *Advances in Engineering Software*, **2013**, 58, 54-69.
[29] K. RamaKrishna, Ch. V. Kameswara Rao, R. Sambasiva Rao, *J. Applicable Chem*., **2013**, 2 (4), 698-713. (b) Unpublished results
[30] B. Bahmani-Firouzi, E. Farjah, A. Seifi, *Energy*, **2013**, 52, 320-332.
[31] A. Kaveh, O.K. Hosseini, *Periodica Polytechnica: Civil Engineering*, **2012**, 56 (2), 197-212.
[32] T. Niknam, F. Golestaneh, M. Shafiei, *Energy*, **2013,** 49 (1), 252-267.
[33] T. Niknam, M. Bornapour, A. Gheisari, B. Bahmani-Firouzi, *Int. J. Hydrogen Energy*, **2013,** 38 (2), 1111-1127.
[34] T. Niknam, R. Azizipanah-Abarghooee, M.R. Narimani, *Energy*, **2012**, 47 (1), 451-464.
[35] A. Kaveh and O.K. Hosseini, *Periodica Polytechnica: Civil Eng*, **2012**, 56 (2), 197-212.