Architectural evolution through computational intelligence: A module within FILA-SoS

Siddhartha Agarwal

Engineering Management and Systems Engineering Department
Missouri University of Science and Technology
Rolla, Missouri
Outline of the Presentation

- Motivations and background
- **Meta-Architecture generation**
  - Genetic Algorithms
  - Binary Particle Swarm
- **Architecture assessment model** (domain specific)
  - Fuzzy Modular Nets
- **Meta-Architecture Implementation**
- Negotiation protocol
  - Model the opponent
  - Decision Making
  - Generate counter-offer
- Current results
- Future Work
FILA-SoS Highlights

- Flexible and Intelligent Learning Architectures for SoS

- FILA-SoS and the Wave Process address four of the most challenging aspects of system-of-system architecting:
  
  - 1.) Dealing with the uncertainty and variability of the capabilities and availability of potential component systems.
  
  - 2.) Providing for the evolution of the system-of-system needs, resources and environment over time.
  
  - 3.) Accounting for the differing approaches and motivations of the autonomous component system managers.
  
  - 4.) Optimizing system-of-systems characteristics in an uncertain and dynamic environment with fixed budget and resources
SoS Basics

A System of Systems....

– Has operationally independent elements.
  – Component systems are useful in their own right.

– Has managerial independent elements.
  – Component systems maintain current operations independent of the SoS.

– Has evolutionary development.
  – A SoS is scalable.

– Has emergent behavior
  – The SoS performs functions that do not reside in any one component system.
A conceptual overview of System-of-Systems

Acknowledged System of Systems

$\textbf{SoS}$ coordinator needs to build an overall larger capability

ENVIRONMENT VARIABLES
(Other Stakeholders such as government, media, employees, and subcontractors)

SYSTEM A
SYSTEM B
SYSTEM C
SYSTEM D
SYSTEM E
SYSTEM F
SYSTEM G
SYSTEM H
SYSTEM I

Capability TYPE I
Capability TYPE II
Capability TYPE III
Need and Applications of System-of-Systems

- Organizations that acquire large-scale systems have transformed their attitude to acquisition.
- These organizations now want solutions to provide a set of capabilities, not a single specific system to meet an exact set of specifications.
- It is therefore necessary for the acquisition manager to be able to look at the future scenarios and critically assess the impact of technology and stakeholder changes.
- The acquisition manager’s look for options that signify affordable acquisition selections and lessen the cycle time for early acquisition and new technology addition.
- This research impacts a number of areas, such as:
  - Coast Guard Deepwater Program
  - FAA Air Traffic Management
  - Army Future Combat Systems
  - Disaster Management Response Systems
Multi-Objective Optimization Approach

- The multi-objective approach combines multiple objectives into the following single objective:

  - Max or Min $f_k (x)^T \forall k$
  - $g_i (x)^T \leq b_i \forall i$
  - $x^T = \{ x_1, x_2, \ldots, x_n \} \in X$

$x$: vector of the variables; $f$: objective function(s); $g$: inequality constraints;
Difficulties in Handling Many Objectives in Evolutionary Algorithms

- A large fraction of population is non-dominated
- Evaluation of diversity measure becomes computationally expensive
- Representation of trade-off surface is difficult
- Visualization is difficult
- Incorporate viewpoints from multiple stakeholders for choosing the near optimal solution
Theorem 1

- Given are \( m \) randomly selected points in the \( n \)-dimensional hypercube. For the expectation value of the size of the PARETO set of these \( m \) points we have the recursive relation:

\[
\begin{align*}
    e_1(n) &= 1 \\
    e_m(1) &= 1 \\
    e_m(n) &= e_{m-1}(n) + \frac{1}{m} e_m(n-1) \quad (n,m \geq 2)
\end{align*}
\]

- For increasing number of sample points in the hypercube, the number of non-dominated points will also increase.
- For increasing dimension, it will become more and more unlikely to find any dominated point in a population of random sampling points.
- In fact, the probability falls exponentially. The PARETO set of \( m \) points will contain nearly all \( m \) points.
Techniques to Handle Many Objectives

- MCDM-based EMO methodologies
- Using Reference-Point-Based Nondominated Sorting Approach
- Ranking methods for many-objective optimization
- $\varepsilon$ dominance concept and an adaptive population sizing approach
- Performance metrics such as hyper-volume indicator
- Use of scalarizing functions for fitness functions
Architecture Representation

Architecture space

Genotype space = \{0, 1\}^L

Encoding (representation)

<table>
<thead>
<tr>
<th>Fighters (EO/IR)</th>
<th>Fighters (EO/IR)</th>
<th>...</th>
<th>...</th>
<th>BLOS</th>
<th>I_{Fighters with JSTARS}</th>
<th>I_{Fighters with U-2}</th>
<th>I_{JSTARS with U-2}</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systems</td>
<td>Interfaces</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>X_1, X_2, X_i, ...</td>
<td>X_m, X_1 with 2, X_1 with 3, X_1 with m, X_2 with 3, ...</td>
<td>X_i with j, ...</td>
<td>X_{(m-1)} with m</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

A.E. Eiben and J.E. Smith, Introduction to Evolutionary Computing Genetic Algorithms
Key performance attributes (KPA) of the SoS

- Performance denoted by $P^{SoS}$
- Affordability denoted by $A^{SoS}$
- Robustness denoted by $R^{SoS}$
- Modularity denoted by $M^{SoS}$
- Net-Centricity denoted by $NC^{SoS}$
Range of Values of Key Performance Attributes for evaluating SoS

- KPAs of the SoS can be provided with different levels of linguistic granularization such as:
  - Net Centricity: very insufficient, insufficient, sufficient, good, and brilliant
  - Affordability: very costly, costly, medium, cheap, and very cheap
A Kiviat chart describing the SoS Architecture as too risky according to Architecture Attributes Assessment.
Fuzzy Membership Functions

Temperature in the range \([T_1, T_2]\) conceived as: (a) a fuzzy variable; (b) a traditional (crisp) variable.

figure from Klir\&Yuan
The fuzzy modular nets used to evaluate an architecture’s quality according to key system attributes.
Interval Type-2 Fuzzy Sets (IT2FS)

- Example of Tall

- Upper MF $\mu_{\text{Tall}}$
- Lower MF $\mu_{\text{Tall}}$
- Type -1 MF

$\sim$

Height (m)

$\mu$

$1$

0

$\mu_{\text{Tall}} = \text{FOU}$
Choosing the Best Architecture After Mutation & Crossover

Meta-Architecture Quality
1. 3.51----Best Architecture
2. 3.10
3. 1.25
The best architecture obtained by BPSO

Maximum, minimum and mean SoS architectural value histories obtained over 100 generations via BPSO

A circular undirected, graph of systems selected in the best SoS architecture as obtained through BPSO
The best architecture obtained by GA

Figure 7 Maximum, Minimum and Best SoS architecture value.

Figure 8 An undirected graph of systems selected the best SoS architecture obtained through GA.
# Systems and the capabilities selected in the best architectures

<table>
<thead>
<tr>
<th>Systems Selected by BPSO</th>
<th>Capabilities Provided</th>
<th>Systems Selected by GA</th>
<th>Capabilities Provided</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systems 1,2-Cutter</td>
<td>2,7,9,10</td>
<td>Systems 1-Cutter</td>
<td>2,7,9,10</td>
</tr>
<tr>
<td>Systems 3,4-Helicopter</td>
<td>1,8,10</td>
<td>Systems 3-Helicopter</td>
<td>1,8,10</td>
</tr>
<tr>
<td>Systems 5,6-Aircraft</td>
<td>3,5,8,10</td>
<td>Systems 4,5-Aircraft</td>
<td>3,5,8,10</td>
</tr>
<tr>
<td>Systems 11,12,16,17-UAV</td>
<td>1,3,4,7,9,10</td>
<td>Systems 7,8,9,10,12,17-UAV</td>
<td>1,3,4,7,9,10</td>
</tr>
<tr>
<td>Systems 20-22 -Fish Vessel</td>
<td>3,4,6,7,9,10</td>
<td>Systems 18-Fish Vessel</td>
<td>3,4,6,7,9,10</td>
</tr>
<tr>
<td>Systems 23 –Civilian Ship</td>
<td>3,4,6,7,9,10</td>
<td>Systems 24 –Coordination Control</td>
<td>5,6,9,10</td>
</tr>
<tr>
<td>Systems 24,25 –Coordination Control</td>
<td>5,6,9,10</td>
<td>Systems 28-Communication</td>
<td>10</td>
</tr>
<tr>
<td>Systems 26,27,28,29-Communication</td>
<td>10</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Three Salient Features of Automated Negotiation

- Model the Opponent
  - Cooperative
  - Non-Cooperative

- Make a Decision
  - Reject
  - Accept
  - Negotiate

- Generate a Counter-Offer
  - Time based constraints
  - Utility based constraints
  - Resource based constraints
Overall Negotiation Protocol

• Send an offer to all systems simultaneously
• Receive a counter-offer from all systems
• Model the opponent behavior-(clustering)
• First make decision on set of systems with capability $i$ where $i = 1$ to $N$
• Need to select at least one system from each capability $i$
• Evaluate the overall architecture quality based on the systems selected in one epoch (may contain multiple negotiation rounds)
• If the architecture is not of a predefined quality then go for a second epoch for systems not yet selected
First Round of Negotiation

<table>
<thead>
<tr>
<th>System j</th>
<th>Performance</th>
<th>Funding</th>
<th>Deadline</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\Delta_p = P_{\text{SoS}} - P_S$</td>
<td>$\Delta_f = F_{\text{SoS}} - F_S$</td>
<td>$\Delta_d = D_{\text{SoS}} - D_S$</td>
</tr>
</tbody>
</table>

Table 1 Concession calculated by SoS manager for each system

- $P_{\text{SoS}}$: Performance demands by SoS to system j
- $P_S$: Performance accepted to be delivered by system j to SoS
- $F_{\text{SoS}}$: Funding provided by SoS to system j
- $F_S$: Funding asked by system j to SoS
- $D_{\text{SoS}}$: Deadline demands by SoS to system j
- $D_S$: Deadline accepted by system j to SoS
Overall Negotiation Protocol

- Send an offer to all systems simultaneously
- Receive a counter-offer from all systems
- Model the opponent behavior-(clustering and training)
- First make decision on set of systems with capability \( i \) \( \text{where} \ i = 1 \text{ to } N \)
- Need to select at least one system from each capability \( i \)
- Evaluate the overall architecture quality based on the systems selected in one epoch (may contain multiple negotiation rounds)
- If the architecture is not of a predefined quality then go for a second epoch for systems not yet selected
Hierarchical clustering

Four red boxes over the major clusters in the dendogram
K-means clustering

plots the within groups sum of squares by number of clusters extracted.
The systems’ behaviors as reflected in the four clusters

- Class 1-ready for participation in lesser time, asks for less funding and provides a stronger performance

- Class 2- Request for more time to participate, asks for less funding, provides a stronger performance

- Class 3- ready for participation in lesser time, asks for less funding and provides a weaker performance than any of the other clusters

- Class 4- Request for more time to participate, asks for more funding, provides a stronger performance

Systems in Class 1 can be referred to very cooperative. Class 2 and 3 systems can be referred to as are semi-cooperative. Class 4 can be referred to as selfish.
Predicting the behavior of incoming offer

- Labelled data obtained from K-means clustering
- Train the labelled data using supervised learning
- Supervised learning is a type of machine learning algorithm that uses a known dataset (called the training dataset) to make predictions.
- The training dataset includes input data and response values.
- From it, the supervised learning algorithm seeks to build a model that can make predictions of the response values for a new dataset.
Learning Vector Quantization

- Learning vector quantization (LVQ) is a supervised learning technique that uses class information to move the vectors slightly, so as to improve the quality of the classifier decision regions.

- Training an LVQ network is accomplished by presenting input vectors and adjusting the location of hidden units based on their proximity to the input vector.

- The hidden layer weights are trained in this manner for an arbitrary number of iterations, usually with the learning rate decreasing as training progresses.

- The objective is to place the hidden units so as to cover the decision regions of the training set.
Learning Vector Quantization (LVQ)

aim:

classification of data
learning from examples

classification:

assignment of a vector $\xi$ to the class of the closest prototype $w$

example situation:

3 classes, 3 prototypes
Radial Basis Function Network (RBFN)

input layer (fan-out)

hidden layer (weights correspond to cluster centre, output function usually Gaussian)

output layer (linear weighted sum)
Finding radius of Gaussian Kernels for Radial Basis Functions

- The root-mean squared distance between the current cluster centre and its $P$ nearest neighbours is calculated, and this is the value chosen for $\sigma$.
- So, if the current cluster centre is $c_j$, the value is:

$$\sigma_j = \sqrt{\frac{1}{P} \sum_{i=1}^{P} (c_k - c_i)^2}$$
The confusion matrix indicates that the output performance of LVQN is much better than RBFN.
The numbers of misclassifications by RBFN are illustrated in Figures above.
The misclassifications for LVQN are zero whereas RBFN has
Three Salient Features of Automated Negotiation

- **Model the Opponent**
  - Cooperative
  - Non-Cooperative

- **Make a Decision**
  - Reject
  - Accept
  - Negotiate

- **Generate a Counter-Offer**
  - Time based constraints
  - Utility based constraints
  - Resource based constraints
Overall Negotiation Protocol

- Send an offer to all systems simultaneously
- Receive a counter-offer from all systems
- Model the opponent behavior-(clustering and training)
- First make decision on set of systems with capability $i$ where $i=1$ to $N$
- Need to select at least one system from each capability $i$
- Evaluate the overall architecture quality based on the systems selected in one epoch (may contain multiple negotiation rounds)
- If the architecture is not of a predefined quality then go for a second epoch for systems not yet selected
## Systems, Capabilities and Behaviors

<table>
<thead>
<tr>
<th>Systems Selected by GA</th>
<th>Capabilities Provided</th>
<th>Behavior Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systems 1-Cutter</td>
<td>2-7,9,10</td>
<td>Class 1</td>
</tr>
<tr>
<td>Systems 3-Helicopter</td>
<td>1-8,10</td>
<td>Class 1</td>
</tr>
<tr>
<td>Systems 4,5-Aircraft</td>
<td>3,5,8,10</td>
<td>Class 4, Class 3</td>
</tr>
<tr>
<td>Systems 7,8,9,10,12,17-UAV</td>
<td>1,3,4,7,9,10</td>
<td>Class 2, Class 4, Class 3, Class 1, Class 2, Class 3</td>
</tr>
<tr>
<td>Systems 18-Fish Vessel</td>
<td>3,4,6,7,9,10</td>
<td>Class 4</td>
</tr>
<tr>
<td>Systems 24 –Coordination Control</td>
<td>5,6,9,10</td>
<td>Class 3</td>
</tr>
<tr>
<td>Systems 28-Communication</td>
<td>10</td>
<td>Class 4</td>
</tr>
</tbody>
</table>
## Systems, Capabilities and Behaviors

<table>
<thead>
<tr>
<th>Systems Selected by GA</th>
<th>Capabilities Provided</th>
<th>Behavior Predicted</th>
</tr>
</thead>
<tbody>
<tr>
<td>Systems 3-Helicopter, System 7-UAV</td>
<td></td>
<td>Class 1, Class 2</td>
</tr>
<tr>
<td>Systems 1-Cutter</td>
<td>2</td>
<td>Class 1</td>
</tr>
<tr>
<td>Systems 4,5-Aircraft, Systems 7,8,9,10,12,17-UAV, Systems 18-Fish Vessel</td>
<td>3</td>
<td>Class 4, Class 3, Class 2, Class 3, Class 4, Class 1, Class 2, Class 3, Class 4</td>
</tr>
<tr>
<td>Systems 7,8,9,10,12,17-UAV, Systems 18-Fish Vessel</td>
<td>4</td>
<td>Class 2, Class 4, Class 3, Class 1, Class 2, Class 3, Class 4</td>
</tr>
<tr>
<td>Systems 4,5-Aircraft, Systems 24 –Coordination Control</td>
<td>5</td>
<td>Class 4, Class 3, Class 3</td>
</tr>
<tr>
<td>Systems 18-Fish Vessel, Systems 24 –Coordination Control</td>
<td>6</td>
<td>Class 4, Class 3</td>
</tr>
<tr>
<td>Systems 28-Communication</td>
<td>7</td>
<td>Class 4</td>
</tr>
<tr>
<td>Systems 3-Helicopter, Systems 4,5-Aircraft</td>
<td>8</td>
<td>Class 1, Class 4, Class 3</td>
</tr>
<tr>
<td>Systems 1-Cutter, Systems 7,8,9,10,12,17-UAV, Systems 18-Fish Vessel, Systems 24 –Coordination Control</td>
<td>9</td>
<td>Class 1, Class 2, Class 4, Class 3, Class 4, Class 3, Class 1, Class 2, Class 3, Class 4, Class 3</td>
</tr>
<tr>
<td>Systems 24 –Coordination Control, Systems 28-Communication</td>
<td>10</td>
<td>Class 3, Class 4</td>
</tr>
</tbody>
</table>
Fuzzy Network Architecture and a set of four terms with their semantics

- Preference for Capability
- Cooperative Behavior
- Willingness to Collaborate

2 Tuple Computing With Words

SoS Decision:
Accept
Negotiate
Reject

L
M
H
VH
## Decision by SoS manager for each system

<table>
<thead>
<tr>
<th>Attributes/Alternatives</th>
<th>A1</th>
<th>A2</th>
<th>A3</th>
<th>A4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Willingness to collaborate</td>
<td>(H,0)</td>
<td>(M,0)</td>
<td>(VH,0)</td>
<td>(L,0)</td>
</tr>
<tr>
<td>Cooperative behavior</td>
<td>(L,0)</td>
<td>(M,0)</td>
<td>(VH,0)</td>
<td>(L,0)</td>
</tr>
<tr>
<td>Preference of capability</td>
<td>(VH,0)</td>
<td>(H,0)</td>
<td>(H,0)</td>
<td>(M,0)</td>
</tr>
<tr>
<td>2-tuple Linguistic Aggregation</td>
<td>(H,-0.36)</td>
<td>(M,0.33)</td>
<td>(VH,-0.34)</td>
<td>(L,0.33)</td>
</tr>
<tr>
<td>Decision of SoS</td>
<td>Negotiate</td>
<td>Negotiate</td>
<td>Accept</td>
<td>Reject</td>
</tr>
</tbody>
</table>
Three Salient Features of Automated Negotiation

- Model the Opponent
  - Cooperative
  - Non-Cooperative

- Make a Decision
  - Reject
  - Accept
  - Negotiate

- Generate a Counter-Offer
  - Time based constraints
  - Utility based constraints
  - Resource based constraints
Generate a Counter-offer

- \( V_{i}^{SoS}(t + 1) = V_{i}^{SoS}(t) + |V_{i}^{SoS}(t) - V_{i}^{s}(t)| \times \left( \frac{t}{t_{\text{max}}} \right)^{\beta_{s}} \) : Polynomial

- \( V_{i}^{SoS}(t + 1) = V_{i}^{SoS}(t) + e^{\left((1 + \frac{t}{t_{\text{max}}} \right)^{\beta_{s}}) \ast \ln(|V_{i}^{SoS}(t) - V_{i}^{s}(t)|)} \) : Exponential

\( \beta >> 1 \) : This choice is made if the opponent is Conceder (reluctant) (SoS starts losing ground fairly quickly)

\( \beta = 1 \) : This choice is made if the opponent is Linear (SoS concedes equal amount in each round of negotiation)

\( \beta < 1 \) : This choice is made if the opponent is Boulware (SoS concedes slowly till the deadline is nearly up)

\( \beta = \frac{\text{Number of systems remaining to be selected from capability } i}{\text{Total systems or alternatives for capability } i} \)
Overall Negotiation Protocol

- Send an offer to all systems simultaneously
- Receive a counter-offer from all systems
- Model the opponent behavior-(clustering and training)
- First make decision on set of systems with capability $i \text{ where } i = 1 \text{ to } N$
- Need to select at least one system from each capability $i$
- Evaluate the overall architecture quality based on the systems selected in one epoch (may contain multiple negotiation rounds)
- If the architecture is not of a predefined quality then go for a second epoch for systems not yet selected
The fuzzy modular nets used to evaluate an architecture’s quality according to key system attributes.
Overall Negotiation Protocol

• Send an offer to all systems simultaneously
• Receive a counter-offer from all systems
• Model the opponent behavior-(clustering and training)
• First make decision on set of systems with capability $i$ where $i = 1$ to $N$
• Need to select at least one system from each capability $i$
• Evaluate the overall architecture quality based on the systems selected in one epoch (may contain multiple negotiation rounds)
• If the architecture is not of a predefined quality then go for a second epoch for systems not yet selected
Results

- Although the best architecture obtained by the two techniques is slightly different for the same set of objectives, it means many good architectures exist in the modeled design space.

- The clustering and training techniques did fairly well on this sample data.

- This model is generic and can be applied various multi-attribute negotiations.

- Also can be applied to situations where negotiation takes the form of one to many eg. One buyer and many sellers etc.

- Future work involves the complete testing of the model proposed.
Conclusions

• An architectural search methodology was applied to a generic SAR problem,

• The architectures generated via computational intelligence reduced both complexity and time.

• Both a hybrid k means with an RBFN and a hybrid k means with an LVQN algorithm were implemented in this study.

• The SoS decision maker can first choose to negotiate further with any of the systems based on the importance of their functionality within the collaboration (capability offered), cooperation levels, and their personal willingness.

• The SoS coordinator can hence successfully adapt his negotiation framework based on the inherent behavior of systems owners.
Acknowledgements

This material is based upon work supported, in whole or in part, by the U.S. Department of Defense through the Systems Engineering Research Center (SERC) under Contract H98230-08-D-0171. SERC is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the United States Department of Defense. Thanks to Ms. Elizabeth Roberson for help in editing the paper.
Q & A time
Advantages/Disadvantages

• RBF trains faster than a MLP
• Another advantage that is claimed is that the hidden layer is easier to interpret than the hidden layer in an MLP.
• Although the RBF is quick to train, when training is finished and it is being used it is slower than a MLP, so where speed is a factor a MLP may be more appropriate.
KPA definitions

Performance
• The architecture’s performance is calculated as the sum of over $N$ $P_{s}^S$, $L_{s}^S$, and $S_{p}^S$.

Net-centricity
• Interoperability = $\sum_{i=1}^{N} \sum_{j=1}^{N} A_{ii} \ast A_{jj} \ast A_{ij}$

• Communication = $\sum_{k=26}^{29} \emptyset \ast A_{kj} \ast A_{ik}$

Affordability
• Operations cost = $\sum_{s=1}^{N} O_{c}^s \ast \sum_{i=1}^{M} c_{i}^s$

• Interfaces cost = $\sum_{s=1}^{N} I_{c}^s \ast \sum_{k=1,k\neq s}^{N} I_{i}^{s}S_{k}$
KPA continued…

Robustness

• Laplacian \((L)\) is calculated as the difference between the degree matrix (denoted by \(\Delta\)) and the adjacency matrix (denoted by \(A\)).

• The second smallest eigenvalue \(\lambda_2\) of the Laplacian is known as algebraic connectivity [16]. This value is used to assess the robustness of the graphs structure to external perturbations. The algebraic connectivity is equal to zero if and only if the graph is unconnected.

• It is proved that the range of the value of \(\lambda_2\) is \(0 \leq \lambda_2 \leq \frac{N}{N-1} D_{\text{min}}\), where \(N\) is the number of vertices and \(D_{\text{min}}\) is the minimum degree of the graph.

Modularity

• Modularity measures the structure of networks and graphs.

• It is used to compute the maximum possible indivisible graphs (either groups, clusters or communities) within a network.

• Here, \(Q\) (modularity metric) = the number of edges within groups subtracted from expected number of edges within group for a random graph with same node degree distribution as the given network.

• The Newman Girvan algorithm is used to calculate it. The value of modularity is between '-1' and '1'.

• The networks modularity increases as this value increases.
Choosing the Best Architecture After Mutation & Crossover

Meta-Architecture Quality
1. 3.51----Best Architecture
2. 3.10
3. 1.25
Illustration of the first wave in ISR through FILA-SoS

<table>
<thead>
<tr>
<th>System</th>
<th>Type Sub-System</th>
<th>Cap ability Number</th>
<th>Coverage sq mi/hr;</th>
<th>Develop $M/epoch/interface</th>
<th>Operate $K/hr per system</th>
<th>Time to Develop, Epochs</th>
<th>System Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fighter</td>
<td>EO/IR</td>
<td>1</td>
<td>500</td>
<td>0.2</td>
<td>10</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>Trainer</td>
<td>EO/IR</td>
<td>1</td>
<td>2000</td>
<td>2</td>
<td>2</td>
<td>1</td>
<td>2-3</td>
</tr>
<tr>
<td>UAV</td>
<td>EO/IR</td>
<td>1</td>
<td>50000</td>
<td>0</td>
<td>15</td>
<td>0</td>
<td>4-8</td>
</tr>
<tr>
<td>DSP</td>
<td>IR</td>
<td>1</td>
<td>8000</td>
<td>0.1</td>
<td>1</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td>Fighter</td>
<td>Radar</td>
<td>2</td>
<td>3000</td>
<td>0.7</td>
<td>10</td>
<td>1</td>
<td>10-12</td>
</tr>
<tr>
<td>JSTARS</td>
<td>Radar</td>
<td>2</td>
<td>10000</td>
<td>0.1</td>
<td>18</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td>Theatre</td>
<td>Exploit</td>
<td>3</td>
<td>5000</td>
<td>2</td>
<td>10</td>
<td>1</td>
<td>14-15</td>
</tr>
<tr>
<td>CONUS</td>
<td>Exploit</td>
<td>3</td>
<td>25000</td>
<td>0.2</td>
<td>0</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td>Control Station/AOC</td>
<td>Cmd &amp; Control</td>
<td>4</td>
<td>10000</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>17-18</td>
</tr>
<tr>
<td>LOS Link</td>
<td>Comm</td>
<td>5</td>
<td>10000</td>
<td>0.2</td>
<td>0</td>
<td>1</td>
<td>19-20</td>
</tr>
<tr>
<td>BLOS Link</td>
<td>Comm</td>
<td>5</td>
<td>5000</td>
<td>0.5</td>
<td>3</td>
<td>1</td>
<td>21-22</td>
</tr>
</tbody>
</table>
The left side of the figure is the meta-architecture whereas the right side is the negotiated architecture.

- **Assessment for the meta-architecture** = 3.47
- **Key Attribute values:**
  - Performance = 2.16; Flexibility = 4
  - Affordability = 3.5; Robustness = 3.91

- **Assessment for the final-architecture** = 2.5
- **Key Attribute values:**
  - Performance = 1.5; Affordability = 3.72
  - Flexibility = 2; Robustness = 2
### Illustration of the second wave in ISR through FILA-SoS

<table>
<thead>
<tr>
<th>System</th>
<th>Type Sub-System</th>
<th>Cap ability Number</th>
<th>Coverage sq mi/hr;</th>
<th>Develop $M/epoch/interface</th>
<th>Operate n//hr$/K/ hr per system</th>
<th>Time to Develop, Epochs</th>
<th>System Number</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Fighter</strong></td>
<td>EO/IR</td>
<td>1</td>
<td>500</td>
<td>0.2</td>
<td>10</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td><strong>Trainer</strong></td>
<td>EO/IR</td>
<td>1</td>
<td>12000</td>
<td>0.1</td>
<td>8</td>
<td>1</td>
<td>2-3</td>
</tr>
<tr>
<td><strong>UAV</strong></td>
<td>EO/IR</td>
<td>1</td>
<td>8000</td>
<td>0.5</td>
<td>2.5</td>
<td>1</td>
<td>4-8</td>
</tr>
<tr>
<td><strong>DSP</strong></td>
<td>IR</td>
<td>1</td>
<td>8000</td>
<td>0.1</td>
<td>1</td>
<td>1</td>
<td>9</td>
</tr>
<tr>
<td><strong>Blimp</strong></td>
<td>Radar</td>
<td>2</td>
<td>20000</td>
<td>0.5</td>
<td>12</td>
<td>1</td>
<td>10-12</td>
</tr>
<tr>
<td><strong>JSTARS</strong></td>
<td>Radar</td>
<td>2</td>
<td>10000</td>
<td>0.1</td>
<td>18</td>
<td>1</td>
<td>13</td>
</tr>
<tr>
<td><strong>Theatre</strong></td>
<td>Exploit</td>
<td>3</td>
<td>5000</td>
<td>2</td>
<td>10</td>
<td>1</td>
<td>14-15</td>
</tr>
<tr>
<td><strong>MOBExp</strong></td>
<td>Exploit</td>
<td>3</td>
<td>15000</td>
<td>0.1</td>
<td>0.2</td>
<td>0</td>
<td>16</td>
</tr>
<tr>
<td><strong>MOBC2</strong></td>
<td>Exploit</td>
<td>4</td>
<td>12000</td>
<td>1</td>
<td>2</td>
<td>0</td>
<td>17</td>
</tr>
<tr>
<td><strong>Control Station/ AOC</strong></td>
<td><strong>Cmd &amp; Control</strong></td>
<td>4</td>
<td>10000</td>
<td>1</td>
<td>2</td>
<td>1</td>
<td>18</td>
</tr>
<tr>
<td><strong>LOS Link</strong></td>
<td>Comm</td>
<td>5</td>
<td>10000</td>
<td>0.2</td>
<td>0</td>
<td>1</td>
<td>19-20</td>
</tr>
<tr>
<td><strong>BLOS Link</strong></td>
<td>Comm</td>
<td>5</td>
<td>5000</td>
<td>0.5</td>
<td>3</td>
<td>0</td>
<td>21</td>
</tr>
<tr>
<td><strong>Mil-Sat</strong></td>
<td>Comm</td>
<td>5</td>
<td>15000</td>
<td>1</td>
<td>5</td>
<td>1</td>
<td>22</td>
</tr>
</tbody>
</table>

The systems highlighted in yellow were selected at the end of negotiation process in the previous wave. Hence, they are preserved or maintained in the next wave meta-architecture.
The left side of the figure is the meta-architecture whereas the right side is the negotiated architecture.

- **Assessment for the meta-architecture** = 3.61
  - **Key Attribute values:**
    - Performance = 2.28; Flexibility = 4
    - Affordability = 3.09; Robustness = 3.77

- **Assessment for the final-architecture** = 3.1
  - **Key Attribute values:**
    - Performance = 1.94; Affordability = 3.6
    - Flexibility = 4; Robustness = 3.16
Illustration of the first wave in SAR through FILA-SoS

<table>
<thead>
<tr>
<th>SysNo</th>
<th>Type</th>
<th>Capability</th>
<th>I/FDevCost</th>
<th>OpsCost/hr</th>
<th>Perf</th>
<th>DevTime</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cutter</td>
<td>3</td>
<td>0.03</td>
<td>2</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Cutter</td>
<td>3</td>
<td>0.03</td>
<td>2</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Helicopter</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>Helicopter</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Aircraft</td>
<td>4</td>
<td>0.1</td>
<td>5</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>Aircraft</td>
<td>4</td>
<td>0.1</td>
<td>5</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>UAV</td>
<td>1</td>
<td>0.1</td>
<td>0.1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>UAV</td>
<td>1</td>
<td>0.1</td>
<td>0.1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>UAV</td>
<td>2</td>
<td>0.1</td>
<td>0.1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>UAV</td>
<td>2</td>
<td>0.1</td>
<td>0.1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>UAV</td>
<td>2</td>
<td>0.1</td>
<td>0.1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>UAV</td>
<td>2</td>
<td>0.1</td>
<td>0.1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.03</td>
<td>0.5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.03</td>
<td>0.5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.03</td>
<td>0.5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.03</td>
<td>0.5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>Civ Ship</td>
<td>3</td>
<td>0.05</td>
<td>2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>Coord Ctr</td>
<td>5</td>
<td>0.05</td>
<td>0.5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>Coord Ctr</td>
<td>5</td>
<td>0.05</td>
<td>0.5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>Communications</td>
<td>5</td>
<td>0.02</td>
<td>0.03</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>21</td>
<td>Communications</td>
<td>5</td>
<td>0.02</td>
<td>0.03</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>22</td>
<td>Communications</td>
<td>5</td>
<td>0.02</td>
<td>0.03</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
The left side of the figure is the meta-architecture whereas the right side is the negotiated architecture.

- **Assessment for the meta-architecture** = 3.78
  - **Key Attribute values:**
    - Performance = 3.36; Flexibility = 4
    - Affordability = 3.66; Robustness = 3.68

- **Assessment for the final-architecture** = 1.29
  - **Key Attribute values:**
    - Performance = 1.5; Affordability = 3.9
    - Flexibility = 4; Robustness = 1
Illustration of the secondwave in SAR through FILA-SoS

<table>
<thead>
<tr>
<th>SysNo</th>
<th>Type</th>
<th>Capability</th>
<th>I/FFDevCost</th>
<th>OpsCost/hr</th>
<th>Perf</th>
<th>DevTime</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Cutter</td>
<td>3</td>
<td>0.03</td>
<td>2</td>
<td>12</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Cutter</td>
<td>3</td>
<td>0.05</td>
<td>2.5</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Helicopter</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>Helicopter</td>
<td>3</td>
<td>0.1</td>
<td>2</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>Aircraft</td>
<td>4</td>
<td>0.1</td>
<td>5</td>
<td>10</td>
<td>1</td>
</tr>
<tr>
<td>6</td>
<td>Aircraft</td>
<td>4</td>
<td>0.5</td>
<td>8</td>
<td>20</td>
<td>1</td>
</tr>
<tr>
<td>7</td>
<td>UAV</td>
<td>1</td>
<td>0.1</td>
<td>0.1</td>
<td>7</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>UAV</td>
<td>1</td>
<td>0.4</td>
<td>0.2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>9</td>
<td>UAV</td>
<td>2</td>
<td>0.4</td>
<td>0.2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>10</td>
<td>UAV</td>
<td>2</td>
<td>0.4</td>
<td>0.2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>11</td>
<td>UAV</td>
<td>2</td>
<td>0.4</td>
<td>0.2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>12</td>
<td>UAV</td>
<td>2</td>
<td>0.4</td>
<td>0.2</td>
<td>8</td>
<td>1</td>
</tr>
<tr>
<td>13</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.03</td>
<td>0.5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>14</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.03</td>
<td>0.5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>15</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.02</td>
<td>0.3</td>
<td>6</td>
<td>1</td>
</tr>
<tr>
<td>16</td>
<td>Fish Vessel</td>
<td>2</td>
<td>0.03</td>
<td>0.5</td>
<td>4</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>Civ Ship</td>
<td>3</td>
<td>0.05</td>
<td>4</td>
<td>15</td>
<td>1</td>
</tr>
<tr>
<td>18</td>
<td>Coord Ctr</td>
<td>5</td>
<td>0.05</td>
<td>0.5</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>19</td>
<td>Coord Ctr</td>
<td>5</td>
<td>0.03</td>
<td>0.1</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>20</td>
<td>Communications</td>
<td>5</td>
<td>0.02</td>
<td>0.03</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>21</td>
<td>Communications</td>
<td>5</td>
<td>0.02</td>
<td>0.03</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>22</td>
<td>Communications</td>
<td>5</td>
<td>0.01</td>
<td>0.05</td>
<td>1.5</td>
<td>0</td>
</tr>
</tbody>
</table>
The left side of the figure is the meta-architecture whereas the right side is the negotiated architecture

- **Assessment for the meta-architecture = 3.74**
  - Key Attribute values:
    - Performance = 3.49; Flexibility = 4
    - Affordability = 3.67; Robustness = 3.65

- **Assessment for the final-architecture = 1.55**
  - Key Attribute values:
    - Performance = 2.45; Affordability = 1
    - Flexibility = 4; Robustness = 2.06
Evolution of SoS Architecture Through Multiple Waves

Sample Scenario for Toy-Wave 1 results

<table>
<thead>
<tr>
<th>System</th>
<th>Type Sub-System</th>
<th>Capability Number</th>
<th>Value Added to SoS</th>
<th>Time to Develop, Epochs</th>
<th>System Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command</td>
<td>Ground Control</td>
<td>1</td>
<td>5-50</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>SAT</td>
<td>SAT-Type 1</td>
<td>2</td>
<td>20-50</td>
<td>1</td>
<td>2-9</td>
</tr>
<tr>
<td>UAV</td>
<td>UAV Types</td>
<td>3</td>
<td>2.5-10</td>
<td>0</td>
<td>10-15</td>
</tr>
<tr>
<td>SAT</td>
<td>SAT-Type 2</td>
<td>4</td>
<td>20-50</td>
<td>1</td>
<td>16-21</td>
</tr>
<tr>
<td>Carrier</td>
<td>Aircraft Carrier</td>
<td>5</td>
<td>45-50</td>
<td>1</td>
<td>22</td>
</tr>
</tbody>
</table>

FILA-SoS Toy Problem Operational View 1
- Control Station – 2 Sat Links, 1 UAV link
- Carrier – 1 UAV link, 1 Sat link
- Choice of 1 of 2 UAVs
- Choice of 2 of 3 Sats
Evolution of SoS Architecture

Systems selected in the Meta-Architecture

• Meta-Architecture Wave 1

<table>
<thead>
<tr>
<th>Systems Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ground Control (1)</td>
</tr>
<tr>
<td>Satellite Type A (2,6,8,9)</td>
</tr>
<tr>
<td>UAV (11,12,15)</td>
</tr>
<tr>
<td>Satellite Type B (16,18,21)</td>
</tr>
<tr>
<td>Carrier (22)</td>
</tr>
</tbody>
</table>

Assessment for the meta-architecture = 2.98
Key Attribute values:
• Strength of Dependency (SOD) = 0.16
• Criticality of Dependency (COD) = 55
• Performance of Carrier = 79
Evolution of SoS Architecture

• Negotiated Architecture Wave 1

<table>
<thead>
<tr>
<th>Sys no</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>selfish</td>
</tr>
<tr>
<td>2</td>
<td>cooperative</td>
</tr>
<tr>
<td>11</td>
<td>opportunistic</td>
</tr>
<tr>
<td>21</td>
<td>opportunistic</td>
</tr>
<tr>
<td>22</td>
<td>Selfish</td>
</tr>
</tbody>
</table>

Assessment for the meta-architecture =2.1
Key Attribute values:
• Strength of Dependency (SOD)=0.12
• Criticality of Dependency (COD)=55
• Performance of Carrier=60
Evolution of SoS Architecture Through Multiple Waves

Sample Scenario for Toy-Wave 2 results

<table>
<thead>
<tr>
<th>System</th>
<th>Type Sub-System</th>
<th>Capability Number</th>
<th>Value Added to SoS</th>
<th>Time to Develop, Epochs</th>
<th>System Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command Ground Control</td>
<td>1</td>
<td>5-50</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>SAT</td>
<td>SAT-Type 1</td>
<td>2</td>
<td>20-50</td>
<td>1</td>
<td>2-9</td>
</tr>
<tr>
<td>UAV</td>
<td>UAV Types</td>
<td>3</td>
<td>2.5-10</td>
<td>0</td>
<td>10-15</td>
</tr>
<tr>
<td>SAT</td>
<td>SAT-Type 2</td>
<td>4</td>
<td>20-50</td>
<td>1</td>
<td>16-21</td>
</tr>
<tr>
<td>Carrier Aircraft Carrier</td>
<td>5</td>
<td>45-50</td>
<td>1</td>
<td>22</td>
<td></td>
</tr>
</tbody>
</table>

Figure: FILA-SoS Toy Problem Operational View 1
- Control Station – 2 Sat Links, 1 UAV link
- Carrier – 1 UAV link, 1 Sat link
- Choice of 1 of 2 UAVs
- Choice of 2 of 3 Sats
Evolution of SoS Architecture

Systems selected in the Meta-Architecture

• Meta-Architecture Wave 1

<table>
<thead>
<tr>
<th>Systems Selected</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ground Control (1)</td>
</tr>
<tr>
<td>Satellite Type A (2,5,9)</td>
</tr>
<tr>
<td>UAV (10,11,15)</td>
</tr>
<tr>
<td>Satellite Type B (17,18,21)</td>
</tr>
<tr>
<td>Carrier (22)</td>
</tr>
</tbody>
</table>

Assessment for the meta-architecture = 2.5

Key Attribute values:

• Strength of Dependency (SOD) = 0.4
• Criticality of Dependency (COD) = 35
• Performance of Carrier = 81
Evolution of SoS Architecture

• Negotiated Architecture Wave 1

<table>
<thead>
<tr>
<th>Sys no</th>
<th>Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>selfish</td>
</tr>
<tr>
<td>5</td>
<td>cooperative</td>
</tr>
<tr>
<td>11,15</td>
<td>opportunistic</td>
</tr>
<tr>
<td>21</td>
<td>opportunistic</td>
</tr>
<tr>
<td>22</td>
<td>Selfish</td>
</tr>
</tbody>
</table>

Assessment for the meta-architecture = 1.2
Key Attribute values:
• Strength of Dependency (SOD) = 0.03
• Criticality of Dependency (COD) = 43
• Performance of Carrier = 71
Pareto-Box problem

• Given \( m \) uniformly randomly selected \( n \)-dimensional points. The problem is stated as:

Pareto-Box Problem: What is the expectation value for the size of the Pareto set of these points?

• **Theorem.** For fixed dimension \( n > 1 \) and the number of points \( m \to \infty \), the expectation value \( e_m(n) \to \infty \), the ratio of the non-dominated points \( e_m(n)/m \to 0 \) and for fixed \( m > 1 \) and dimension \( n \to \infty \) it holds \( e_m(n) \to m \).

• With objectives increasing to 5 or more almost 90% of the solutions become non-dominated.

**Theorem 2.** The expectation value for the size of the Pareto set of \( m \geq 1 \) randomly selected points in the \( n \)-dimensional hypercube \( (n \geq 1) \) is

\[
e_m(n) = \sum_{k=1}^{m} \frac{(-1)^{k+1}}{k^{n-1}} \binom{m}{k}
\]

(6)
KPA definitions

Performance
• The architecture’s performance is calculated as the sum of over $N$ $P_{s}$, $L_{s}$, and $Sp_{s}$.

Net-centricity
• Interoperability = $\sum_{i=1}^{N} \sum_{j=1}^{N} A_{ii} * A_{jj} * A_{ij}$
• Communication = $\sum_{k=26}^{29} \emptyset * A_{kj} * A_{ik}$

Affordability
• Operations cost= $\sum_{s=1}^{N} Oc_{s} * \sum_{i=1}^{M} c_{i}^{Ss}$
• Interfaces cost= $\sum_{s=1}^{N} Ic_{s} * \sum_{k=1,k\neq s}^{N} I_{i}^{SsS_{k}}$
KPA continued...

Robustness

- Laplacian (L) is calculated as the difference between the degree matrix (denoted by Δ) and the adjacency matrix (denoted by A).
- The second smallest eigenvalue $\lambda_2$ of the Laplacian is known as algebraic connectivity. This value is used to assess the robustness of the graphs structure to external perturbations. The algebraic connectivity is equal to zero if and only if the graph is unconnected.
- It is proved that the range of the value of $\lambda_2$ is $0 \leq \lambda_2 \leq \frac{N}{N-1} D_{min}$, where N is the number of vertices and $D_{min}$ is the minimum degree of the graph.

Modularity

- Modularity measures the structure of networks and graphs.
- It is used to compute the maximum possible indivisible graphs (either groups, clusters or communities) within a network.
- Here, $Q$ (modularity metric) = the number of edges within groups subtracted from expected number of edges within group for a random graph with same node degree distribution as the given network.
- The Newman Girvan algorithm is used to calculate it. The value of modularity is between '-1' and '1'.
- The networks modularity increases as this value increases.
Comparison between GA and PSO

• Unlike in genetic algorithms, evolutionary programming and evolutionary strategies, in PSO, there is no selection operation.

• All particles in PSO are kept as members of the population through the course of the run.

• PSO is the only algorithm that does not implement the survival of the fittest.

• No crossover operation in PSO.

• Each particle modifies its position according to:
  • its current position (p)
  • its current velocity (v)
  • the distance between its current position and pbest (best position of the particle)
  • the distance between its current position and gbest (best position of the swarm)
  \[ v = v + c_1 \times rand \times (pBest - p) + c_2 \times rand \times (gBest - p) \]
Results

• Although the best architecture obtained by the two techniques is slightly different for the same set of constraints, it means many good architectures exist in the modeled design space.

• Both GA and BPSO try to model the fitness function surface to reach the global maxima.

• The architecture value obtained by BPSO is higher than GA.

• This signifies the PSO was better able to map the surface of the fitness function generated by the fuzzy rules.
Conclusions

• An architectural search methodology was applied to a generic SAR problem, and a set of architectures each with a high fitness, was obtained.
• The architectures generated via computational intelligence reduced both complexity and time.
• The architectures generated were the best combinations possible for the given domain problem.
• The stochastic heuristic techniques can assist in the systems architecting process by providing the systems architects with a set of feasible designs that can be developed into a near optimal architecture.