Rethinking the JDL Data Fusion Levels

Alan N. Steinberg
alan.steinberg@comcast.net

Christopher L. Bowman
cbowman@indra.com

Abstract

Recent work has occasioned a rethinking of the well-known JDL Data Fusion levels. The suggested revised partitioning of data fusion functions is designed to capture the significant differences in the types of input data, models, outputs, and inferencing appropriate to broad classes of data fusion problems. In general, the recommended partitioning is based on different aspects of a situation for which the characterization is of interest to a system user. In particular, a given entity − whether a signal, physical object, aggregate or structure − can often be viewed either (a) as an individual whose attributes, characteristics and behaviors are of interest or (b) as an assemblage of components whose interrelations are of interest. From the former point of view, discrete physical objects (the "targets" of level 1 data fusion) are components of a situation. From the latter point of view, the targets are themselves situations; i.e. contexts for the analysis of components and their relationships.

A complementary set of resource management levels − duals of the fusion levels − is also introduced. Among other things, these encompass such fusion-related functions as (a) management of the data collection, signal and data processing, and fusion processes and (b) management of models used in data fusion processes; to include building and refining sensor and system performance models, bias models for calibration or registration, models of target characteristics, and models of the characteristics of situations.

1 Objective

In this paper, refinements to the well-known JDL Data Fusion model are proposed. Specifically, the goals of this paper are

(a) to reiterate the motivation for this and earlier versions of the model: viz. to facilitate understanding of the types of problems for which data fusion is applicable as an aid to recognizing commonalty among problems and determining the applicability of candidate solutions); and

(b) to refine the definition of the levels to establish a clearer basis for partitioning among such problems and concepts.

A companion paper is scheduled for presentation at NSSDF and the International conference on Information Fusion in Stockholm in the summer of 2004 [11]. That paper addresses the additional objectives:

(c) to discuss system design issues, to include functional partitioning and interactions among processes that perform at diverse "levels"; and

(d) to discuss the relationship between data fusion as conceived historically and related information exploitation processes, to include inductive and abductive processes of pattern discovery and explanation (the traditional focus of data mining), management of information collection and processing (to include functions of collection management, and agent-based data retrieval).

2 Background

The Data Fusion model was developed by the Joint Directors of Laboratories Data Fusion Group, a US DoD government committee overseeing US defense technology. The stated purpose for that model and its subsequent revision has been

- to categorize different types of fusion processes;
- to provide a common frame of reference for fusion discussions ;
- to facilitate understanding of the types of problems for which data fusion is applicable;
- to codify the commonality among problems;
- to aid in the extension of previous solutions;
- to provide a framework for investment in automation.

It should be emphasized that the model was conceived as a functional model, not as a process model or as an architectural paradigm.

In 1998 Steinberg, Bowman, and White [1] published the first paper formally addressing various extensions to the model [2]. That paper began by revisiting the basic definition(s) of Data Fusion (DF) both conceptually and in terms of the "Levels" that are characterized in the original JDL model.

Figure 1 depicts the 1998 version of the JDL Data Fusion model. The goals of the 1998 revision – retained in the present proposed revision – were

- to clarify some of the concepts that guided the original model;
- to refine the categorization representing different levels of fusion problems to better reflect different classes of data fusion problems – i.e. systematic differences in types of inputs, create different outputs, and are solved by different techniques; and
- to broaden the definitions of fusion concepts and functions to apply across as wide a range of problems as possible, beyond the initial focus on military and intelligence problems; and
- to maintain a degree of continuity by deviating as little as possible from the concepts and terminology in the data fusion community.

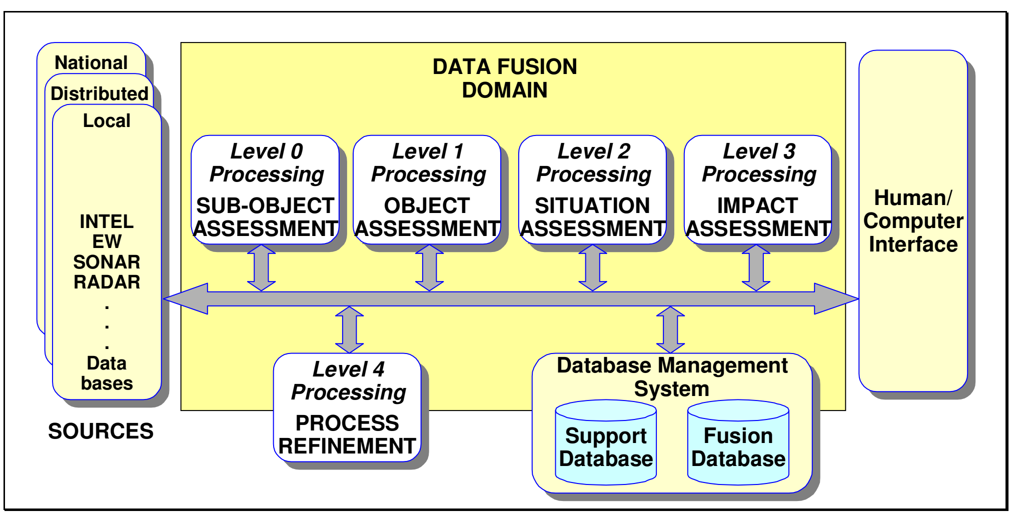

Figure 1: The 1998 Revised JDL Data Fusion Model (with Process Refinement and DBMS Partially Outside the Data Fusion Domain)[1]

Key aspects of the 1998 revision were:

(a) The definitions of levels 1-3 were broadened to accommodate fusion problems beyond military and intelligence ones that had been the focus of earlier versions of the JDL model;

(b) A level 0 was introduced, to address problems of detecting and characterizing signals, whether in a 1-dimension time-series or transform spaces or in multiple spatial dimensions (as in imagery or video feature extraction);

(c) The 1998 revision paper emphasized that the

> "Process Refinement (an element of Resource Management) processing involves planning and control, not estimation … there is a formal duality between estimation and control [; and] there is a similar duality between association and *planning*."

The present paper continues to treat Process Refinement as a resource management function. The DBMS is treated as a support application below the applications program interface (e.g., accomplished within a lower layer of the common operating environment);

(d) The notion of estimating informational and perceptual states in addition to the familiar physical states was introduced, citing the work of Waltz [ref];

(e) An approach to standardization of an engineering design methodology for DF processes was introduced, citing the prior works of Bowman [3], Steinberg and Bowman [4], and Llinas, et al [5] in which engineering guidelines for DF processes were elaborated.

In addition to refining the Data Fusion model, the present revision recommends a model for Resource Management that is isomorphic to that Data Fusion model. This development reflects the dual structure of Data Fusion and Resource Management functionality. The Data Fusion Tree and Fusion Node functional partitioning were defined in 1980 [6-7]. The *Data Fusion and Resource Management (DF&RM) Dual Node Network (DNN)* technical architecture were proposed by Bowman for affordable DF&RM prior to 1994 [3, 8]. That architecture incorporates software components, interfaces and a software development engineering methodology for DF&RM. This DNN architecture leverages the duality of data fusion and resource management to bootstrap the less mature resource management technology.

3 Motivation for the Present Revision

Recent work has occasioned a rethinking of the well-known JDL Data Fusion levels. The suggested revised partitioning of data fusion functions is designed to capture the significant differences in the types of input data, models, outputs, and inferencing appropriate to broad classes of data fusion problems. In general, the recommended partitioning is based on different aspects of a situation for which the characterization is of interest to a system user. In particular, a given entity – whether a signal, physical object, aggregate or structure – can often be viewed either (a) as an individual whose attributes, characteristics and behaviors are of interest or (b) as an assemblage of components whose interrelations are of interest. From the former point of view, discrete physical objects (the "targets" of level 1 data fusion) are components of a situation. From the latter point of view, the targets are themselves situations; i.e. contexts for the analysis of components and their relationship.

This paper extends the JDL levels of Data Fusion to their dual levels in Resource Management. In addition, this paper adds a new level 4 of Data Fusion and its corresponding dual level 4 of Resource Management. It is proposed that these new DF&RM Levels will serve much the same purpose as the original JDL Fusion levels (e.g., albeit to include Resource Management) and will enable the extension of the DNN technical architecture to performance assessment and design management functions.

The extension of the JDL model into a DF&RM applications-layer architecture will promote the development of affordable (e.g., reusable) software to meet the increasing demand for more automation of fusion functions operating in open distributed environments [10].

4 Suggested Definitions for Data Fusion Functional Levels

The suggested revised data fusion function functional levels are based upon the entities of interest to information-users. The recommended revised model diagram is shown in Figure 2, with fusion levels defined as follows:

- Level 0: Signal/Feature Assessment – estimation and prediction of signal or feature states;
- Level 1: Entity Assessment – estimation and prediction of entity parametric and attributive states (i.e. of entities considered as individuals);
- Level 2: Situation Assessment – estimation and prediction of the structures of parts of reality (i.e. of relations among entities and their implications for the states of the related entities);
- Level 3: Impact Assessment – estimation and prediction of the utility/cost of signal, entity or situation states – including predicted impacts given a system's alternative courses of action;
- Level 4: Performance Assessment – estimation and prediction of a system's performance as compared to given desired states and measures of effectiveness.

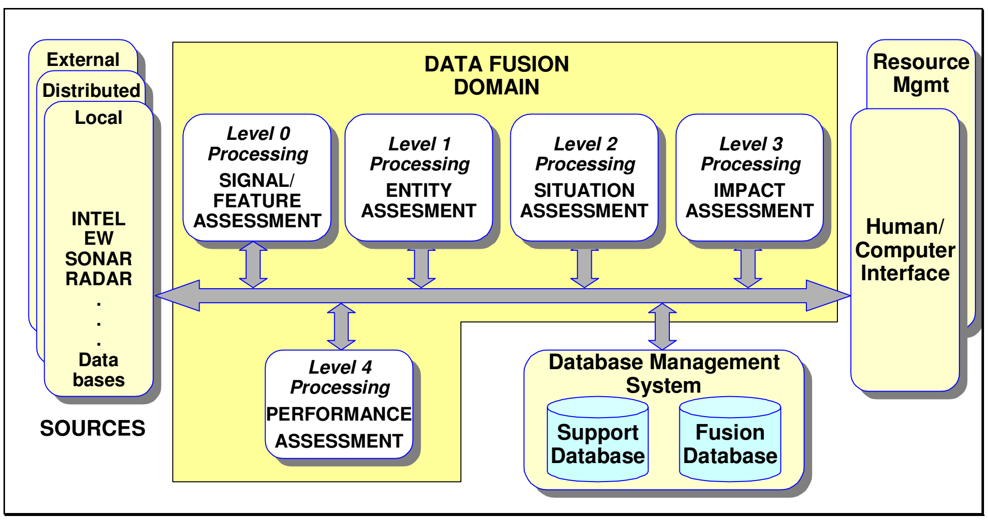

Figure 2: Recommended Revised Data Fusion Model

These concepts are compared and contrasted in Table 1.

Table 1: Characteristics of the Recommended Data Fusion Levels

Data Fusion Levels	Association Process	Estimation Process	Product
L.0 – Signal/Feature Assessment	Observation-to-Signal or Feature*[1]	Feature Extraction	Estimated Signal/Feature States & Confidences
L.1 – Entity Assessment	Signal/Feature-to-Entity or Sensor Entity State	Attributive Entity State Estimation	Estimated Entity States & Confidences

[1] More precisely, where indicated by asterisk, association is with features, signals, entities, relations and situation that are *postulated* by the system.

	Report-to-Entity*		
L.2 – Situation Assessment	Entity*-to-Entity*, Entity*-to-Relationship* or Relationship*-to-Relationship*	Relational State Estimation	Estimated Relationships, Situation (set of Relationships), & Confidences
L.3 – Impact Assessment	Situation* to System Courses of Action	Cost/Benefit Analysis	Estimated/Predicted Entity & Situation Utilities & Confidences
L.4 – Performance Assessment	System States* to Goals	Performance Analysis	Estimated MOPs and MOEs & Confidences

In general, the benefit of this scheme of partitioning fusion functions into these levels is due to the significant differences in the types of input data, models, outputs, and inferencing applicable to problems at the different levels. These levels are not necessarily processed in order and any one can be processed on its own given the corresponding inputs. In addition, more than one fusion level may need to be treated within one fusion node to achieve user fusion needs albeit at higher complexity.

A brief description of each follows. A more detailed treatment of level 2, Situation Assessment, and it relationship to Level 1 (Entity Assessment) is provided in an Appendix.

4.1 Level 0: Signal/Feature Assessment

The 1998 revision [1] recommended defining a Level 0 fusion to encompass various uses of multiple measurements in signal and feature processing. These include processes of feature extraction in Image Fusion and analogous signal detection and parameter estimation in ELINT and MASINT. In the most general sense, these problems concern the discovery of information (in Shannon's sense) in some region of space and time.

Level 0 processes include inferences that do not require assumptions about the presence or characteristics of entities ("targets") possessing such observable features. They concern the structure of measurement sets (their syntax) not their cause (i.e. their semantics). In cases in which signal characteristics are conditioned on the presence and characteristics of one or more presumed entities, the latter are treated as a context, or situation, in which the signals or features are inferable; e.g. via a likelihood function.

These functions include DAI/FEO (Data In/Features Out) and FEI/FEO (Features In/Features Out) functions as defined in Dasarathy's model [9], as well as DEI-FEO (Decisions In/Features Out) in our expanded version of that model [handbook][2]

4.2 Level 1: Entity Assessment

JDL Level 1, as in earlier versions, is concerned with the estimation of the identity, classification (in some given taxonomic scheme), attributes, activities, location, and the dynamics and potential states of entities.

---

[2] Dasarathy's original categories represent constructive, or data-driven processes, in which organized information is extracted from relatively unorganized data. In [ref] we defined additional processes that are analytic, or model-driven, such that organized information (a model) is analyzed to estimate lower-level data (features or measurements) as they relate to the model. Examples include pre-detection tracking (an FEI-DAO process), model-based feature-extraction (DEI-FEO), and model-based classification (DEI-DAO).

Entity-hood implies an inclusion, or boundary function, such as is found in such paradigmatic entities as

- Biological organisms;
- Purpose-made entities; e.g. a hammer, automobile, or bird's nest;
- Other discriminated spatially contiguous uniformities that persist over some practical time interval; e.g. a boulder, coral reef, cloud, raindrop or swarm of gnats;
- Socially (e.g. legally) discriminated entities; e.g. a country, tract of land, family, treaty organization.

A level-1 estimation process can be thought of as the application of one-place predicates $P^{(1)}(a) = x$, where x may be a discrete or a continuous random variable, or a vector of such variables; so that $P^{(1)}(a)$ may be the temperature, location, functional class, an attribute, or activity state of an entity a.

4.3 Level 2: Situation Assessment

Level 2 data fusion concerns inferences regarding relationships among entities and contextual implications concerning entities and relationships.

Developments in Situation Theory [13, 14] can be applied to estimation and prediction problems by generalizing the notion of *situation* to include any structured part of reality: a single- or multi-target state, including attributes of and relations among such entities.

Situations can be represented as sets of relationships, enabling inferencing from one perceived entity in a situation to another and from estimated entities, their attributes and relationships to situations. Attributes of individuals can be treated as1-place relationships.

A more detailed discussion of Situation Assessment and its relation to Entity Assessment is given in the Appendix.

4.4 Level 3: Impact Assessment

Level 3 data fusion deals with the estimation and prediction of the *utility* or *cost* of an estimated world state (i.e. situation) to a user objective (e.g., mission).

Because the utility of a fused state in supporting a user generally needs to be predicted based upon the estimated current situation, known plans and predicted reactions, level 3 fusion processing has different inputs and different outputs (i.e., utility predictions) than the other fusion levels. An example of a level-3 type of *association* problem is that of determining the expected consequence of system plans or courses of action (COAs), given the current estimated signal, entity, and/or situational state. An example of a level 3 type of *estimation* problem is the prediction of the impact of the estimated current situational state on the mission utility.

4.5 Level 4: Performance Assessment

In early versions of the JDL model, a level 4 was designated Process Refinement. This was meant to encompass adaptivity in the fusion process and – in some interpretations – in the data collection process as well. We have argued that a more fundamental partitioning of functionality involves a distinction between Data Fusion functions (involving data association and state estimation) and Resource Management functions (involving planning and control) and that Process Refinement is clearly a Resource Management function [1988 paper].

There is, however, a need for a system-level category of level 4 fusion processes, one that is intimately involved in *associating* system states and responses to desired system states (or, in the case of a Data Fusion system, to *truth*) and then *estimating/predicting* the performance of the system. We therefore propose a Data Fusion level 4, concerning this Performance Assessment. Level 4 data fusion functions combine information to estimate a system's measures of performance (MOPs) and of effectiveness (MOEs) based upon a given desired set of system states and/or responses.

In Level 4, the primary association problem to be solved is that of determining which system outputs correspond to which of the desired responses. For example, in the case of a DF&RM system, to what extent each track associates to each truth and to what extent each resource response corresponds to the set of desired responses over the mission. In Performance Assessment, the hierarchy of MOE/MOPs are then estimated based upon these associations.

5 Information Flow Within and Across the "Levels"

Processing at each of these Data Fusion levels involves the batching the available data for fusion into a network of fusion nodes where paradigmatically each fusion node accomplishes

- Data Preparation (data mediation, common formatting, spatio-temporal alignment, and confidence normalization);
- Data Association (generation, evaluation and selection of association hypotheses; i.e. of hypotheses as to the applicability of specific data to particular aspects of the state estimation problem);
- State Estimation and Prediction (estimating the presence, attributes, inter-relationships, utility and performance or effectiveness of entities of interest, as appropriate to the data fusion node).[3]

A comparison of the association problems that need to be solved at each fusion level is given in Figure 3. In all the fusion levels, the accuracy of the fused state estimates tend to increase as larger batches of data (e.g., time, sources, and types) are fused; however the cost and complexity of the fusion process also increases. Thus, a knee-of-the-curve of performance versus cost fusion node network is sought in the system design and operation.

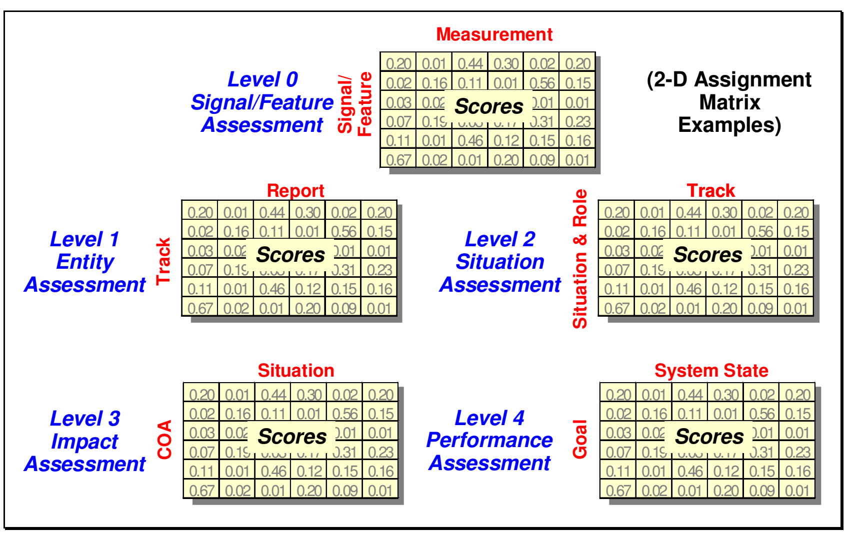

Figure 3: Data Association Problems Occur at Each Fusion Level

---

[3] We use the work 'paradigmatically' because some systems may not require data preparation in every fusion node. Furthermore, not every multi-target state estimation processes requires explicit data association at the target level; data can be determined to be relevant at the situational (e.g. multi-target) level.

As noted above, the data fusion levels are not necessarily processed in order and any one can be processed on its own or in combination given the corresponding inputs. Therefore, the term "level" can be misleading in this context. The early DF Model descriptions rightfully avoided the sense of hierarchy by the use of functional diagrams like Figure 1, in which various data fusion functions are represented in a Common Bus architecture.

Nonetheless, there is often a natural progression from raw measurements to entity states, to situation relationships, to utility prediction, and to system performance assessment, in which data are combined as suggested by the level ordering. Such composition of estimated signals (or features), entities, and aggregates per Levels 0-2 is quite natural; with a corresponding reverse flow of contexts, as typified in Figure 3.[4] Utility is generally assessed as a function of an estimated or predicted situational state, rather than of the state of a lone entity within a situation. That is the reason that Level 3 fits naturally "above" Level 2. System performance can also be assessed based on estimated or predicted outcomes of situations and executed or planned system actions; consistent with our ordering of levels.

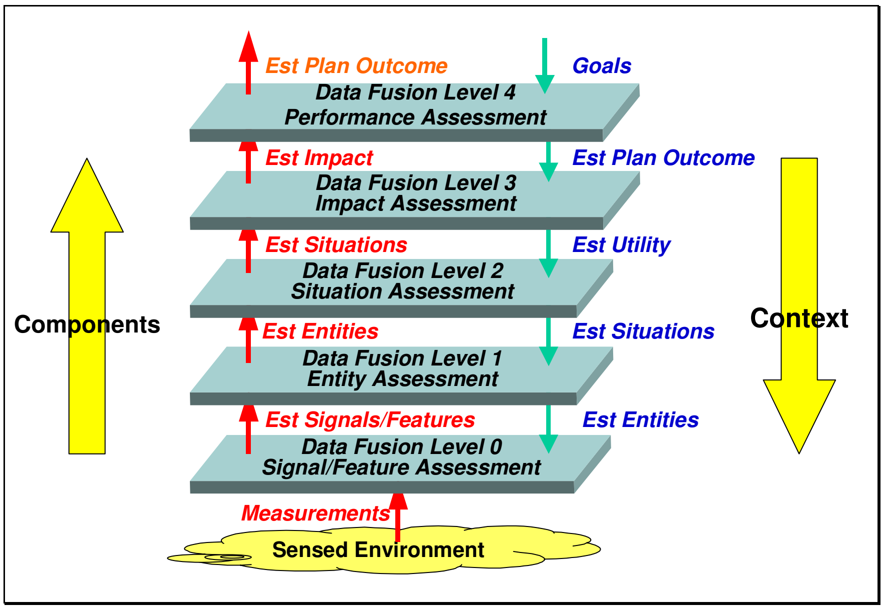

Figure 4 Characteristic Functional Flow across the "Levels"

6 Corresponding Resource Management Levels

The function of *Process Refinement* – level 4 in early versions of the JDL model – is part of the Resource Management function. As discussed by Bowman [10], just as a formal duality exists between estimation and control, there is a more encompassing duality between data fusion and resource management that includes the duality between association and planning. The duality between Data Fusion and Resource Management is summarized in Figure 5.

We now propose a functional model for Resource Management in terms of functional levels that are, accordingly, the duals of proposed Data Fusion Levels. A dual scheme for modeling DF and RM functions has the following beneficial implications for system design and development:

---

[4] In many or most data fusion implementations to date, the predominant flow is upward; any contextual conditioning in a data fusion node being provided by corresponding resource management nodes (e.g., during process refinement).

- Data fusion & resource management (DF&RM) systems can be implemented using a network of interacting fusion and management nodes;
- The dual DF&RM levels provide insight into techniques to support the designs for each;
- The technology maturity of data fusion can be used to bootstrap resource management technology which is >10 years behind; much as estimation did for control theory in the 1960's;
- The levels are partitioned due to the significant differences in the types of data, resources, models, and inferencing necessary for each level.;
- All the fusion levels can be implemented using a fan-in network of fusion nodes where each node performs: data preparation, data association, and state estimation;
- All the management levels can be implemented using a fan-out network of management nodes where each node performs: task preparation, task planning, and resource state control.

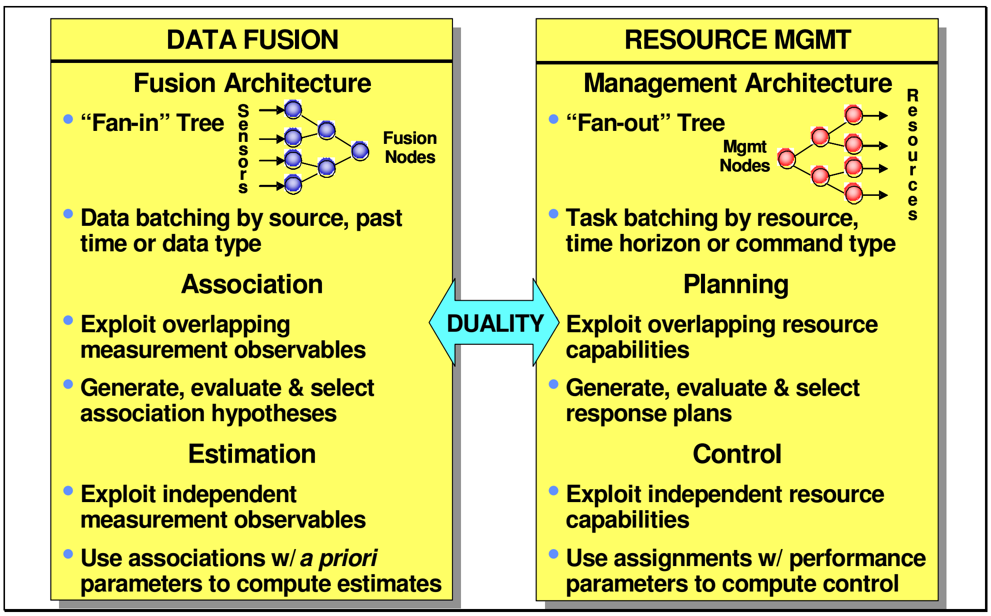

Figure 5: The Data Fusion & Resource Management (DF&RM) Duality: Key Concepts

Given the dual nature of Data Fusion and Resource Management, we would expect a corresponding duality in functional levels. As with data fusion "levels", the dual resource management processing levels are based upon the user's resources of interest. The proposed dual DF and RM levels are presented in Table 2.

Table 2: Dual Data Fusion and Resource Management Processing Levels

Level	Data Fusion Level	Resource Management Level
0	**Signal/Feature Assessment:** estimation of signal/feature states	**Signal Management:** management of resource emissions/observables
1	**Entity Assessment:** estimation of entity attributive states	**Resource Response Management:** management of individual resources
2	**Situation Assessment:** estimation of entity relational/situational states	**Resource Relationship Management:** management of resource relationships
3	**Impact Assessment:** estimation of the impact of fused states on mission	**Mission Objective Management:**

	objectives	management of mission objectives
4	**Performance Assessment:** estimation of MOP/MOE states	**Design Management:** management of system engineering and operational configuration

The duality concepts that were used in defining these Resource Management levels are summarized in Figure 6.

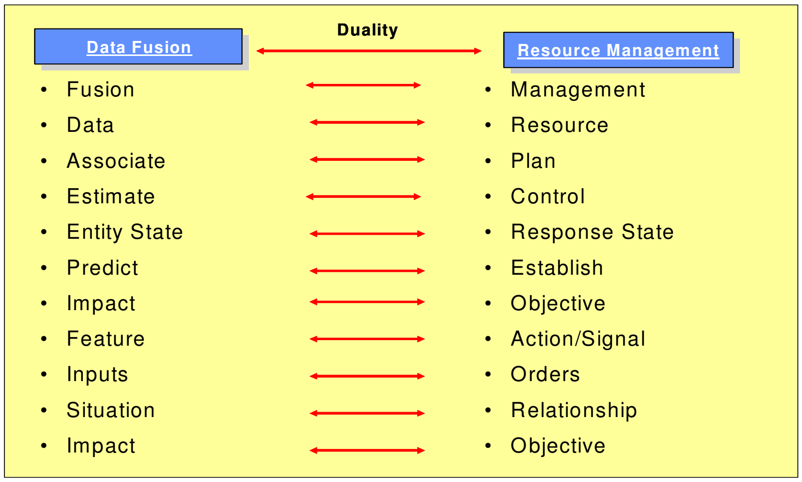

Figure 6: DF&RM Duality Concepts Enable "Jump-Start" of Less Mature Resource Management

As with the corresponding partitioning of data fusion functions into levels, the utility of these management levels is due to the significant differences in the types of resources, models, and inferencing used in each level. Resource Management includes applications-layer functions such as sensor management, target management, weapons management, countermeasure management, flight management, process management, communications management, etc. As with the Data Fusion levels, the Resource Management levels are not necessarily processed in order and any one can be processed on its own or in combination given the corresponding inputs.

All the management levels can be implemented and integrated using a fan-out network of management nodes where each node performs the functions of task preparation, task planning, and resource state control. Data fusion and resource management systems can be implemented using a network of interacting fusion and management nodes. These node interactions can occur across any of the levels. However, for the purpose of showing all the "on-line" levels 0-3, Figure 6 shows a sequential processing flow across the DF and RM levels.[5]

---

[5] Level 4 fusion and management processes are typically performed off-line, during system design and evaluation.

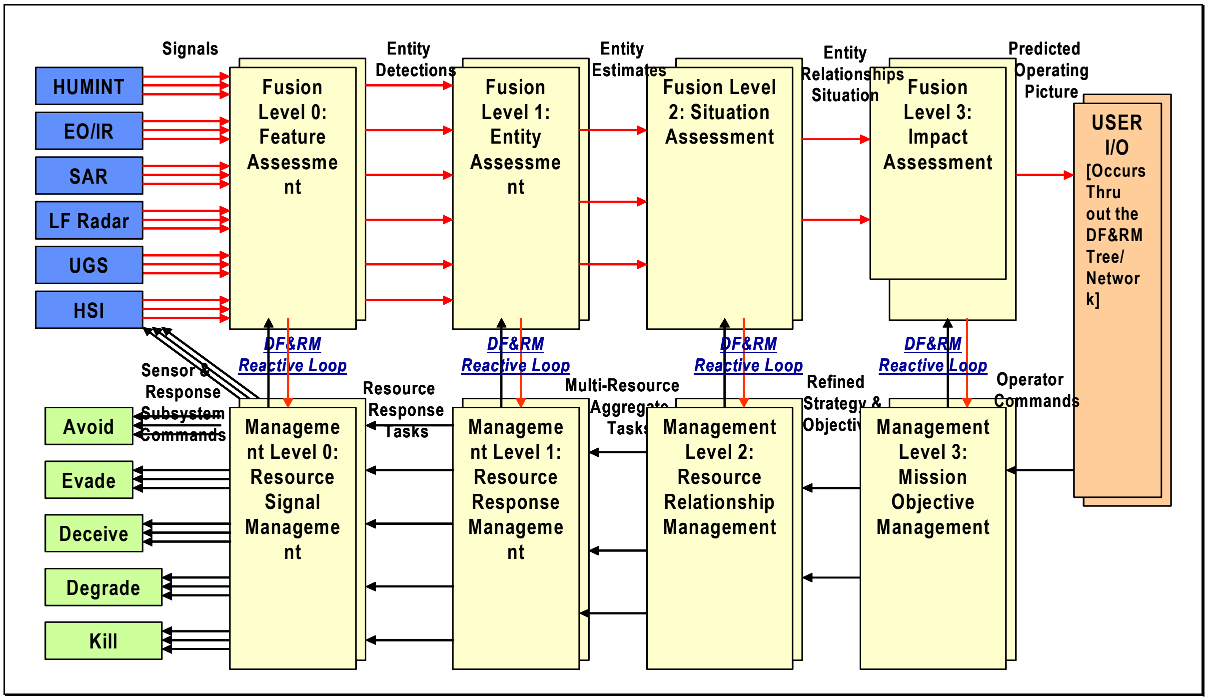

Figure 7: Sample Multi-Level DF&RM System Network with Sequential Level Interactions

The extended partitioning by type of inputs, response planning, and output response states strives to enhance clarity, ease solution development, and preserve respect for the duality. Planning – the dual of Data Association – involves analogous assignment problems that differ by RM level. This is illustrated in Figure 8, which is the RM analog to Figure 3.

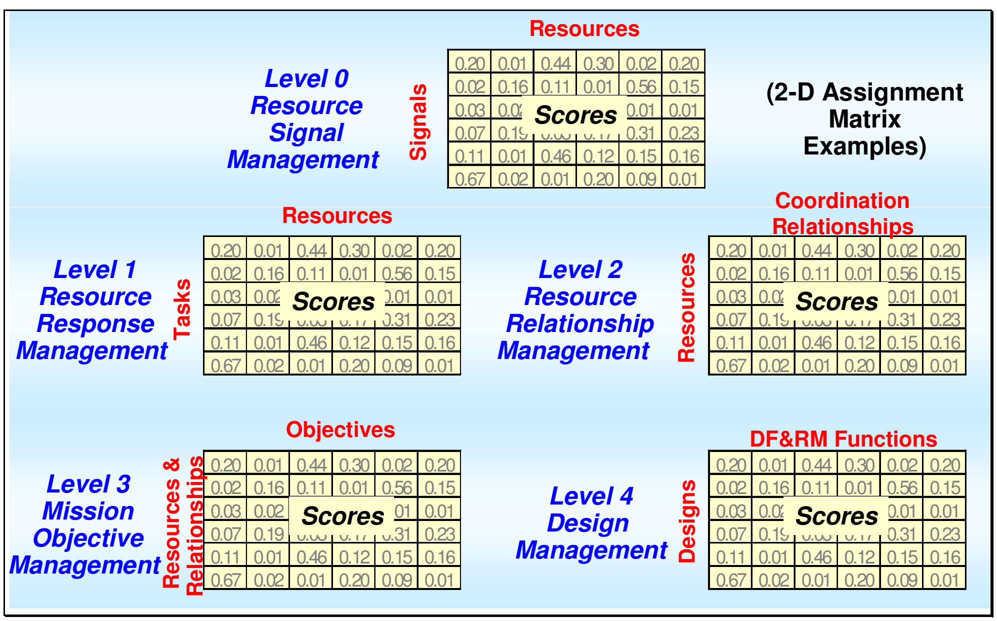

Figure 8: Dual Response Planning Problems Occur at Each Management Level

The five resource management processing levels are described below using the above duality concepts on the data fusion levels given underneath each one:

- Resource Signal Management Level 0: management to task/control specific resource response actions (e.g., signals, pulses, waveforms, etc.);
- Signal/Feature Assessment Level 0: fusion to detect/estimate/perceive specific source entity signals and features.

- Resource Response Management Level 1: management to task/control continuous and discrete resource responses (e.g., radar modes, countermeasures, maneuvering, communications);
- Entity Assessment Level 1: fusion to detect/estimate/perceive continuous parametric (e.g., kinematics, signature) and discrete attributes (e.g., IFF, class, type, ID, mode) of entity states.

- Resource Relationship Management Level 2: management to task/control relationships (e.g., aggregation, coordination, conflict) among resource responses;
- Situation Assessment Level 2: fusion to detect/estimate/comprehend relationships (e.g., aggregation, causal, command/control, coordination, adversarial relationships) among entity states.

- Mission Objective Management Level 3: management to establish/modify the objective of level 0, 1, 2 action, response, or relationship states;
- Impact Assessment Level 3: fusion to predict/estimate the impact of level 0, 1, or 2 signal, entity, or relationship states.

- Design Management Level 4: management to task/control the system engineering (e.g. problem-to-solution space algorithm/model design mapping, model discovery and generalization);
- Performance Assessment Level 4: fusion to estimate the system measures of performance and effectiveness.

7 DF&RM Processing Level Issues

The user's entities of interest can be the basis of all five levels of fusion processing. The features of an entity can be estimated based on attributes inferred from one or more entity signal observations (e.g., via a Level 0 data preparation/ association/estimation process). For example, signal-level association and estimation problems appear in ELINT pulse train deinterleaving or feature extraction of an entity in imagery. This involves inferring the existence and characteristics of the features of an entity by attributive or relational state estimation from observations and measurements specific to each sensor/source modality (e.g., radar pulses, hyperspectral pixel intensities).

The identity, location, track, and activity state of an entity of interest (whether it be a man, a molecule or a military formation) can be estimated on the basis of attributes inferred from one or more signal or entity observations (e.g., via one or a network of data preparation/ association/ estimation fusion nodes). The same entity's compositional or relational state (e.g. its role within a larger structure and its relations with other elements of that structure) can be inferred via Level 2 processes. The behavior of the same entity can also be projected to assess the impact of the utility of an estimated or predicted situation relative to the user's objective.

The fused states can then be compared to desired states at all fusion levels to determine the performance of the fusion system and similarly for response states and the resource management system performance assessment.

The declaration of entity features, entity states, their relationships, their predicted activities, and their correspondence to truth is a data association (i.e., hypothesis generation, evaluation, and selection) function within each fusion level.

Fused state prediction at the Impact Assessment level (DF level 3) needs to determine the impact of alternative projected states. This requires additional information concerning the planned actions and reactions of the entities in the relevant situation (e.g., in the battlespace). Furthermore, utility assessment requires additional information on the mission success criteria; both of which are not necessary for L0-2 fusion.[6]

Utility assessment is the estimation portion of Level 3 processing. Predicting the fused state beyond the current time is of interest for impact assessment whose results are of interest to mission resource management.

The difference between in-mission Impact Assessment and DF&RM system performance estimation is that the former uses only data available to the DF system to assess mission success, whereas the latter makes use of external sources of ground truth and desired states in estimating DF&RM system's performance. For example, in the estimation of the performance of a data fusion process, the system analyst is assumed to have available truth entities and relationships with which to associate the fused state estimates. In contrast, for mission utility estimation, the DF system can only associate its estimate of the current situation to alternative courses of actions to predict the mission impact.

The objective of the lower levels of Resource Management in using impact assessments is to plan responses to improve the confidence in mission success, whereas the objective of Level 4 Design Management in using the Performance Evaluation (PE) outputs is to improve confidence in DF&RM system performance. PE nodes tend to have significant interactions with their duals in Design Management in that they provide the performance estimates for a DF&RM solution that are used to propose a better DF&RM solution. This resource management function for optimizing the mapping from problem to solution space is usually referred to as system engineering (i.e., equated to Design Management herein). The Design Management architecture provides a representation of the system engineering problem that partitions its solution into the RM node processes; i.e. those of

- Problem Alignment (resolving design needs, conflicts, mediation);
- Design Planning (design generation, evaluation, & selection); and
- Design Implementation & Test (output specific resource control commands)

The dual management levels improve understanding of the management alternatives and enables better capitalization of the significant differences in the resource modes, capabilities, and types as well as mission objectives.

Process Refinement – the old DF level 4 – has been subsumed as an element of each level of Resource Management that includes adaptive data acquisition and processing to support mission objectives (e.g. sensor management and information systems dissemination).

User Refinement (which has been proposed as a DF level 5 {ref]) has been subsumed as an element of Knowledge Management within Resource Management. User Refinement includes adaptive determination of which users query information and which have access to information. (e.g. in information operations), as well as adaptive data retrieved and displayed to support cognitive decision making and actions.

Appendix: A Closer Look at Situation Assessment

[6] Generally speaking, L0-2 processes only need to predict the fused state forward in time sufficient for the next data to be fused.

A.1 Situations

Following Devlin, we define a *situation* as a *structured part of reality that is discriminated by some agent* [15, p. 37].

Depending on the way information is used, virtually any entity may be treated either as an individual or as a situation. For example, an automobile may be discussed and reasoned about as a single individual or as an assembly of constituent parts. The differentiation of parts is also subject to choices: we may disassemble the automobile into a handful of major assemblies (engine, frame, body, …) or into a large number of widgets or into a huge assembly of atoms.

Accordingly, the number of entities in a situation can be undecided. That is to say, the same situation can have an indeterminate number of entities, depending on the interests and focus of attention of agents reasoning about or experiencing the situation.

Because of the breadth of our definition, just about any aspect of the world, or any potential aspect thereof, can be considered a situation. We distinguish, in a natural way, between real situations (e.g. the Battle of Midway) and abstract situations or situation types (e.g. Naval Battles, 20th century Naval Battles, encounters at sea, movie subjects).

A *real situation* is a set of *facts*.

Abstract situations are built of "pieces of information", called "infons" in Situation Theory [4]. Infons have the general form $(P, x_1, \ldots, x_n, h, k, p)$, where P is an m-place relation ($m \geq$n), $x_1, \ldots, x_n$, are individuals (entities in our terminology), h and k are a location and time (which may be points or extended regions) and p a polarity, i.e. a truth-value. A *fact* is simply an infon with polarity 1 in the situation constituted by the real world. Therefore, a *real situation* is a set of anchored infons (i.e. those with no unbound variables or parameters) with polarity=1 in the real world.

The symbol '$\models$' is used in representing inferences based on situational context (noting that this is not the same as Tarski's predicate-calculus implicature employing the same symbol). For a situation s and infon σ, '$s \models \sigma$' is read as 'σ is true of s' or 's supports σ''. Given a real situation s, the set of infons $\{\sigma \mid s \models \sigma\}$ is the corresponding abstract situation [4].

Some situations can be crisply defined; e.g., a chess game, of which the constituent entities and their relevant attributes and relationships are explicitly bounded. Other situations may have fuzzy boundaries. Fuzziness is present both in situation types (e.g. the concepts *economic recession* or *naval battle*) and of real-world situations (e.g. the 1930s, the Battle of Midway). Both can naturally be characterized via fuzzy membership functions. For a given situation s, its membership function f_s is a fuzzy indicator function on situational inference: $(s \models \sigma) = f_s,(\sigma)$, so that $\{\sigma \mid (s \models \sigma) > 0\}$ is the corresponding fuzzy situation.

A.2 Attributes and Relationships

Level 1 fusion problems involve estimating or predicting infons headed by 1-place relations $P^{(1)}$ (or, possibly, n-place relations with n-1 bound or parameterized variables). In the familiar data fusion problems, $P^{(1)}$ is often a multi-dimensional state vector of the familiar sort.

In level 2 fusion problems, context is relevant, so that an infon of interest may involve a multi-place relation.

An n-place discrete or continuous *relation* (e.g. Marital status, Distance) is a function that maps within state space $R^{(n)}: X^n \rightarrow Y$, In such a case a $y \in Y$ is a relational state (e.g. y=Marrried, y = 145km); i.e. it is a *Relationship.* An instantation of a relation involves mapping from an instantiated vector to a relational state (e.g. Marital State(Napoleon, Josephine)=Married; Distance (Miami, Havana)=145km) to a relational state (e.g. Napoleon and Josephine's marriage). An (abstract) *attribute* (e.g. Height) is a 1-place relation $R^{(1)}: X^{(1)} \rightarrow Y$. A state $y \in Y$, then, is the value of the attribute (e.g. 1.5m). An instantation of an attribute (an *attribution*) involves mapping from an instantiated state to a relational state (e.g. Napoleon's height=1.5m). Obviously, relations and attributes may be conditional (e.g. Napoleon and Josephine were married ony within a particular span of time; plate tectonics affects terrestrial distances over time).

Relationships of interest in tactical military applications can include

- Relationships among objects in threat complex (deployment, kinetic interaction, organization role/subordination, comms, type similarity, etc.);
- Relationships among blue sensor & weapon platforms (spatio-temporal alignment, measurement calibration, confidence, communication/ coordination, etc.);
- Relationships between sensors & sensed entities (intervisibility, assignment/cueing, sensing, data association, countermeasures);
- Relationships between red & blue tactical entities (targeting, jamming, engaging, etc.);
- Relationships between entities of interest & other entities (terrain features; solar & atmospheric effects; weapon launch & impact points; etc.).

The state of a multi-target set X can be given in terms of infons involving a relation among its members. Because relational roles are not generally symmetrical, this must be given as a relation given on some n-tuple of which the elements are the members of X:

$$P^{(n)}(\underline{X}) \overset{\Delta}{=} P(< x_1,\ldots,x_n >)\,.$$

for some n-placed predicate $P^{(n)}$ and n-tuple of state vectors $\underline{X}^{(n)}$, representing the state of n targets.[7]

We would like to extend the finite random set formulation of multi-target state estimation to level-2 problems. Uncertainties in the truth of a proposition $P^{(n)}(\underline{X}^{(n)})$ can be expressed as distributions of multi-target states. A density function can be given in the notation of Situation Logic as

$$f(P,x_1,\ \ldots\ ,\ x_n,l,t,1).$$

However, there are complications if the number of entities in a situation is unknown.

- A multi-target state of the sort of interest in situation assessment cannot in general be inferred from a set of single-target states $X = \{x_1,\ldots,x_n\}$. For example, we expect that '*x is married to y*' cannot be inferred from any set of statements of the forms '$P(x)$' and '$Q(y)$';
- Relationships of interest are often asymmetrical, such that ordering of elements is significant: infons are vectors. Therefore, a situational state estimate in which the number of targets is unknown needs the use of vectors in an unknown-dimensional space.

However, it is always possible to reduce a vector of any finite length to a set representation, as in the Wiener-Kuratowski definition of an n-tuple

$$< y_1,\ldots, y_n > \overset{\Delta}{=} \{Y \mid \exists i (i \le n \,\&\, Y = \{ y_1,\ldots, y_i \})\};\ (1)$$
$$\text{e.g.,} < y_1, y_2 >= \{\{ y_1\},\{ y_{1},_2 \}\}.$$

---

[7] For succinctness, and not quite legitimately, we abbreviate '$(P,x_1,\ \ldots,\ x_n,\ h,k,1)$' to an elliptical functional form '$P_k(,x_1,\ \ldots,\ x_n,)$', leaving location unbound (i.e. implicit in a given context). This enables us to use familiar predicate calculus notation. It will be important, nonetheless, to remember the original vector form for infons.

Thus, a relationship between two entities can be given by

$$R = \langle P^{(2)}, x_1,x_2\rangle = \{\{ P^{(2)}\},\{ P^{(2)},x_1\},\{ P^{(2)},x_1, x_2\}\};$$

where $P^{(2)}$ randomly varies over binary relations and x_1,x_2 are familiar random entity state vectors.

The set of relationships constituting an uncertain situation $S = \{R_1,\ldots,R_m\}$ involving an unknown number of entities is one in which the R_j can differ in length; i.e., it is a set whose elements have the form given by (1). So S is a finite random set whose elements include both random entity state vectors x_i and random relational state vectors $\langle P, x_{1,\ldots,}x_n\rangle$. The battery of finite set statistic should apply in characterizing such level 2 inference problems.

A.3 Situational Inferences

A situation can imply and can be implied by the states and relationships of constituent entities. Just as in Level 1 inferencing (i.e. with 1-place relations), we can write production rules based on logical, semantic, causal, customary or material (etc.) relationships among predicates of any length. The disposition of players and the configuration of the playing field are indicators that the situation is a baseball game, a bullfight, a chess match, an infantry skirmish, an algebra class, or a ballet recital: $P^{(n)}(X^{(n)}) \Rightarrow S$.

Often situational inferences can be given in the form of Boolean combinations of quantified expressions:

$$\exists n\exists x_1,\ldots,\exists x_n\ [P^{(n)}(x_1,\ldots,x_n)] \Rightarrow S;$$

or, more succinctly,

$$\exists n\exists \underline{X}^{(n)}[P^{(n)}(\underline{X}^{(n)})] \Rightarrow S.$$

To be sure, we are generally in short supply of situational ontologies that would enable us to write such rules. This is a job for expert system builders. For many types of situations, fuzzy rules will be required.[8]

A.4 Inferences across the Levels

We can generalize from the familiar level-1 types of inferences, in which entity attributes are inferred from other attributes:

$$f\,(P^{(1)}(x)|Q^{(1)}(x),S); \qquad \text{(L1→L1 deduction)}$$

(e.g. single target likelihood functions or Markov transition densities). Relations between pairs of entities, or among n-tuples of entities, can similarly be inferred from other relations among them:

$$f\,(P^{(2)}(x,y)\,|\,Q^{(2)}(x,y),S) \qquad \text{(L2→L2 deduction)}$$
$$f\,(P^{(n)}(x_1,\ldots,x_n)\,|\,Q^{(n)}(x_1,\ldots,x_n),S)$$

---

[8] Barwise and Perry [refs] list several modes of inference, to include

- logical; e.g. Kiss(x,y) ⇒ Touch(x,y);
- semantic; e.g. Dog(x,y) ⇒ Animal (x,y);
- causal; e.g. Smoke-present (x) ⇒ Fire-present (x);
- customary; e.g. Pope(x) ⇒ unmarried(x);
- material; e.g. Tallest mountain on earth(x) ⇒ in Asia (x).

Any of these may be represented by means of density functions $f(\cdot)$, which may, but needn't, be a probability density function.

This pattern of Level 2 → Level 2 deduction would include, for example, multi-target likelihood functions or multi-target Markov transition densities familiar in finite set statistics of random sets. [5]

Cross-level inference patterns include the following:

- $f(P^{(2)}(x,y)|\, Q^{(1)}(x), R^{(1)}(y),S)$ (L1→L2 deduction);
- $f(\exists y[P^{(2)}(x,y)]|\, Q^{(1)}(x),S)$ (L1→L2 induction);
- $f(P^{(1)}(x)|\, Q^{(2)}(x,y),S)$ (L2→L1 deduction).

A.5 Multi-Target Estimation and Prediction

Thus armed, we may distinguish among problems in state estimation and tracking multiple targets.

We may distinguish the following types of motion models:

a) Independent Target Motion: each target's motion is not affected by that of any other entity; so multitarget prior probability density functions (pdfs) are simple products of the single target pdfs.

b) Context-sensitive Single Target Motion: at each time-step, a target *x* responds to the current situation, i.e. to the states of other entities, which may be dynamic but are assumed not to be affected by the state of *x*.

c) Interacting Multiple Targets: at each time-step, multiple entities respond to the current situation, which may be affected by the possibly dynamic state of other entities.[9]

We may similarly distinguish the following types of measurement models:

a) Independent Measurements: measurements of each target are not affected by those of any other entity; so multitarget posterior probability density functions are simple products of the single target posterior pdfs.

b) Context-Sensitive Multiple Target Measurements: measurements of one target may be affected by the state of other entities as in cases of additive signatures (e.g. multiple targets in a pixel), occluded or shadowed targets. Other cases involve induced effects; e.g. bistatic illumination, electromagnetic interference or disruption of the observing medium.[10]

References

[1] Steinberg, A.N., Bowman, C.L., and White, F.E., "Revisions to the JDL Model", Joint NATO/IRIS Conference Proceedings, Quebec, October, 1998 and in Sensor Fusion: Architectures, Algorithms, and Applications, Proceedings of the SPIE, Vol. 3719, 1999.

[2] White, F.E., "A Model for Data Fusion", Proc. 1st National Symposium on Sensor Fusion, 1988

[9] Tracking targets with type (a) dynamics is clearly a level-1 (i.e. independent-target) problem. Type (b) and (c) dynamics are often encountered, but are generally treated using independent-target trackers; perhaps with a context-dependent selection of motion models, but assuming conditionally independence among target tracks. Type (c) cases, at least, suggest the value of trackers that explicitly model multi-target interactions.

[10] The absent category, Interacting Multiple Measurements, is actually a hybrid of the listed categories, in which entities affect one another's state and thereby affect measurements.

[3] Bowman, C.L., “The Data Fusion Tree Paradigm and its Dual”, Proc. 7th National Symposium on Sensor Fusion, 1994

[4] Steinberg, A.N. and Bowman, C.L., “Development and Application of Data Fusion Engineering Guidelines”, Proc. 10th national Symposium on Sensor Fusion, 1997

[5] Llinas, J., et al. “Data Fusion System Engineering Guidelines”, Technical Report 96-11/4, USAF Space Warfare Center (SWC) Talon-Command project report, Vol.2 1997

[6] Bowman, Morefield, Murphy, “Multi-Sensor Multi-Target Recognition and Tracking”, 13th Asilomar Conference on Circuits, Systems, & Computers, August, 1980

[7] Bowman, C.L. and Morefield, C.L., “Multi-sensor Fusion of Target Attributes and Kinematics”, 19th IEEE Conference on Decision and Control, December, 1980

[8] Bowman, C.L., “The Role of Process Management in a Defensive Avionics Hierarchical Management Tree”, Tri-service Data Fusion Symposium Proceedings, Johns Hopkins Univ., June 1993

[9] Dasarathy, B., “Sensor Fusion Potential Exploitation-Innovative Architectures and Illustrative Applications”, IEEE Proceedings, Vol. 85, No.1, 1997

[10] Bowman, “Affordable Information Fusion Via an Open, Layered, Paradigm-Based Architecture”, Proceedings of 9th National Symposium on Sensor Fusion, Monterey, CA, March ‘96.

[11] Llinas, J. Bowman, C., Galina Rogova, Alan Steinberg, Ed Waltz , and Frank White, “Revisiting the JDL Data Fusion Model II” International Data Fusion Conference, July, 2004