Metrics For Evaluation of Behavior-Based Robotic Systems

Amol Dattatraya Mali
Dept. of computer science and engg.
Arizona state university,
Tempe, AZ 85287-5406, USA
amol.mali@asu.edu

Amitabha Mukerjee
Centre for Robotics,
IIT Kanpur, India 208016.

Abstract

[1]

This research focuses on robot behaviors which use minimal communication and rely mostly on changes in the environment as their cue for action. The behavior-based paradigm for building autonomous robots has recently become very popular because of its successes, use of the world as an external memory and replacement of classical planning by agent-environment dynamics. However there are no metrics for evaluating and improving the behavior spaces. Our aim here is to bridge this gap. We define novel metrics (power, usefulness, flexibility, modularity and scalability) and investigate the properties of behavior spaces using them. We use these metrics to present results on modifications to individual behaviors and addition of new behaviors to the behavior spaces. We discuss the case of a behavior-based robot operating in kitchen to illustrate the significance of our metrics and discuss how the utility of our metrics remains valid in other behavior representations.

1 Introduction

Behavior-based systems became popular in robotics with the work of Brooks [2], who challenged the deliberative paradigm in AI by building stimulus-response based robots, also known as behavior-based robots. A typical behavior-based robot is a collection of several independent task-achieving modules with a simple distributed control mechanism. Each behavior mediates directly with the external world and behaviors are in a parallel control structure, as opposed to the traditional serial structure where interaction with the world is processed serially through sensors, reasoners, planners, actuators, etc. All behavior systems attempt to reduce or eliminate the centralized shared memory, relying instead on parameter passing and communication between individual behaviors.

[1]This paper appears in the proceedings of ICRA-1998, Belgium (IEEE international conference on robotics and automation), pp. 1122-1127.

Successful results have been achieved using this strategy in the can collection robot [3], navigation of mobile robot [1], 6 legged walking robot [6], autonomous rover Rockey-III that navigates through rough outdoor terrain and collects soil samples [7], behavior-based flying vehicle that won the aerial robotics competition [8], 12 degree of freedom hyper-redundant serpentine robot that navigates complex pipe structures and inspects them [9], walking robot for exploring volcanic craters [10] and a robot that learns how to push boxes [4]. The behavior-based model had a deep impression on Artificial Intelligence and several journal issues have recently been devoted to debating this approach (Artificial Intelligence v.47, 1991, Robotics & Autonomous Systems, v.6:1, 1990, Cognitive Science v.17, 1993, Artificial Intelligence v.73 1995).

However there are no metrics to compare behavior systems. The design of behaviors is an ad-hoc process and different designers come up with different sets of behaviors for the same task. How should these behavior spaces be compared and evaluated ? How should they be modified to change functionality? How should the effects of such modifications be evaluated ? This is not clear. If the behaviors do not work, either the environment is modified or some heuristics are added to the programs. Though testing in real world is essential, it currently lacks formal foundations. We set out to answer these questions and bridge the gap between the design of behavior spaces and their testing in the real world. We develop some metrics for an evaluation of the behavior spaces and use them to arrive at our results. We discuss the case of a kitchen management robot to illustrate the significance of our results. In conclusion, we discuss why these results are important.

2 What Is a Behavior ?

We consider a system of agents in this work. All changes to the world are caused by the action of one of these agents, which are interchangeably called behaviors. Nobody changes the world except these agents.

The essential notion of a behavior is a mapping from a stimulus to a consequence. In this work we followed the psychologists and adopted a *3-tuple* model of behavior: stimulus, action, consequence (also referred to as response in this paper). Thus a behavior β_i is denoted by $< s_i, a_i, c_i >$. In the examples and proofs that follow, the stimulus s_i and the consequence c_i are defined using first order predicates as in Brooks' subsumption architecture. In this article, the stimuli and consequences are assumed to be in pure conjunctive form, though our conclusions apply to non-conjunctive forms as well.

2.1 Behavior Chain

We define a behavior chain and the corresponding task as follows.

- **Behavior Chain**: a temporal sequence of behavior modules $\{\beta_{i_1} : \beta_{i_2} : \beta_{i_3} : \ldots : \beta_{i_n}\}$. Here the action of the earlier module changes the situation in such a way that the newly changed part of the situation provides stimulus for the next module in the sequence.

- **Task:** A task is defined as a transition from an initial state of the world to a new state, achieved through a temporal chain of behaviors. In particular we are interested in defining a measure of the number of tasks that are potentially fulfillable. These correspond to all possible temporal chains of behaviors. Where this is achieved by executing behaviors in a temporal sequence, tasks can be enumerated by the total number of chain fragments that are executable. Thus the chain $\{\beta_1 : \beta_8 : \beta_9\}$ represents a task that is different from $\{\beta_1 : \beta_8\}$.

At this point, we need to address an issue relating to all situation calculus models: the finiteness of the universe. Typically, the set of predicates required for any set of behaviors is finite. Of this set, only a few will be affected by the actions in a behavior chain; the rest constitute the *universe*, which in the rest of this discourse, is considered to be a finite set of entities U. When we write $(c_i \Rightarrow s_{i+1})$, c_i contains the entire finite universe U. For compactness in the explicit statements below, we do not list all predicates from U and limit ourselves to those whose change affects the firing of various stimuli in the chain.

What we mean by the finite universal state can be clarified by an example. Let the state of the Universe be $X \bigwedge Y \bigwedge Z$ and $s_2 = X \bigwedge A$. Let us say that the execution of β_1 makes A true. However c_1 is assumed to contain X which is a part of the universe. Then β_1 leads to β_2 and $(c_1 \Rightarrow s_2)$. Thus when we say that $\{\beta_1 : \beta_2\}$, we mean that a part of s_2 was true in the Universe and execution of β_1 causes the rest of s_2 to become true. This is another version of the frame problem, which arises in any model of such temporal sequencing, such as the add and delete list model used widely in planning and knowledge representation. The model chosen here can be reformulated in terms of the successor state axioms that Reiter[5] has shown to be derivable from the positive and negative effect axiomatic structure.

We define a *behavior space* B as a set of distinct behavior modules, (i.e. no two modules have the same stimulus and consequence). A temporal chain of behaviors C is said to be composable from B (written as $C \lhd B$), if the elements of C are also elements of B. $\mid C \mid$ denotes length of the chain C (which is same as the number of behavior modules in the chain). The members of a behavior space are denoted by $\beta_1, \beta_2, \ldots, \beta_{|B|}$.

2.2 Behavior Metrics

To compare different behavior spaces, we define some metrics that relate to the effectiveness of a behavior space.

- **Power :** A behavior (β := <s, a, c>) is more powerful than (β' := <s', a', c'>) iff $(s' \Rightarrow s) \bigwedge (c \Rightarrow c')$. In other words, it can be triggered at least as frequently as a less powerful behavior and results in a state that subsumes the older one (Figure 1). A behavior space B is more powerful than the behavior space B' iff B' can be obtained from B by replacing some module $\beta \in B$ by less powerful module β'. $B_1 >_p B_2$ denotes that the behavior space B_1 is more powerful than the behavior space B_2.

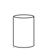
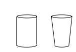
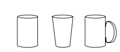
Stimuli for behaviors of increasing power		
(A)	(B)	(C)
		
$graspable(x) \wedge can(x)$	$graspable(x) \wedge (can(x) \vee cup(x))$	$graspable(x)$

Figure 1: *Power:* Behavior B can pick up cups as well as cans (has a more general stimulus). C is even more general. The notion of power is used to capture this, provided all three behaviors have the same consequence.

- **Span**: Behavior space B **spans** task space τ iff and only if $\forall(t \in \tau)\,(\exists(C \lhd B) fulfills(C, t))$.

- **Greatest Potential Task Space** $\tau_G(B)$: Is the set of all tasks that can be fulfilled by behaviors in B, i.e. the largest task space that is spanned by the behavior space B.

- **Usefulness :** The ratio $\frac{|\tau_G(B)|}{|B|}$. $B_1 >_u B_2$ denotes that the behavior space B_1 is more useful than the behavior space B_2. The usefulness of behavior space B_i is denoted by $U(B_i)$.

• **Flexibility :** A behavior space B is at least as flexible as behavior space B' iff $\forall t \in (\tau_G(B) \cap \tau_G(B')) (\exists (C \lhd B)(fulfills(C,t) \bigwedge \forall (C' \lhd B') (fulfills(C',t) \Rightarrow | C | \leq | C' |)))$. This means that B can fulfill tasks with as short or shorter chains (which can be composed from fewer behaviors). $B_1 \geq_f B_2$ denotes that the behavior space B_1 is at least as flexible as the behavior space B_2.

• **Modularity :** A behavior space is more modular if different modules in that space are more independent, in the sense of minimal interference between them. One measure of interference in a behavior space is the incidence of cyclic behavior (see the definition of cyclic conflict in section 3). We therefore define the modularity of behavior space B as the inverse of the likelihood that cyclic conflicts will arise in the given behavior space B. $B_1 \geq_m B_2$ denotes that the behavior space B_1 is at least as modular as the behavior space B_2. Modularity, Power and Flexibility are relative metrics.

• **Scalability :** We define two disjoint behavior spaces B_i and B_j to be *scalable* together if $\tau_G(B_i \cup B_j) \supset (\tau_G(B_i) \cup \tau_G(B_j))$.

3 Properties of Behavior Spaces

Lemma 1. If $B_1 >_p B_2$ and $B_2 >_p B_3$, then $B_1 >_p B_3$.

Proof - Let the behavior space $B_3 = B \cup \{\beta'\}$. Let β' be replaced by a more powerful behavior β'' to derive the behavior space $B_2 = B \cup \{\beta''\}$, so that $B_2 >_p B_3$. Let β'' be replaced by a more powerful behavior β''' to derive the behavior space $B_1 = B \cup \{\beta'''\}$, so that $B_1 >_p B_2$. Since β''' is more powerful than β'' and β'' is more powerful than β', β''' is more powerful than β'. Hence B_1 can be obtained directly from B_3 by replacing the behavior β' by more powerful behavior β'''. Hence $B_1 >_p B_3$.□

From Lemma 1, we get,

Corollary 1. If a behavior space B' is obtained from the behavior space B by replacing some behaviors $\beta_i \in B$ by behaviors β_i' such that each replacing behavior β_i' is more powerful than the replaced behavior β_i, then $B' >_p B$.□

Lemma 2. If $B_1 \geq_f B_2$ and $B_2 \geq_f B_3$, then $B_1 \geq_f B_3$ if $(\tau_G(B_1) \cap \tau_G(B_3)) \subseteq (\tau_G(B_1) \cap \tau_G(B_2))$.

Proof - Since $(\tau_G(B_1) \cap \tau_G(B_3)) \subseteq (\tau_G(B_1) \cap \tau_G(B_2))$,

$(\tau_G(B_1) \cap \tau_G(B_2) \cap \tau_G(B_3)) = (\tau_G(B_1) \cap \tau_G(B_3))$. Since $(\tau_G(B_1) \cap \tau_G(B_2) \cap \tau_G(B_3)) \neq \phi$, $\forall t \in (\tau_G(B_1) \cap \tau_G(B_2) \cap \tau_G(B_3)) (\exists C \lhd B_1, C' \lhd B_2, C'' \lhd B_3 (fulfills(C,t) \bigwedge fulfills(C',t) \bigwedge fulfills(C'',t)))$. Also, since $B_1 \geq_f B_2$ and $B_2 \geq_f B_3$,

$| C | \leq | C' |$ and $| C' | \leq | C'' |$. Hence $| C | \leq | C'' |$. Hence $B_1 \geq_f B_3$.□

Here we define various types of conflicts that may occur in behavior spaces and refer to them in some results in the next sections. In the broad sense of the word conflict, any behavior chain leading to non-fulfillment of the desired objectives can be said to contain a conflict. Let a chain $C = \{ \beta_{i_1} : \beta_{i_2} : ... : \beta_{i_n} \}$, $1 \leq i_1, ..., i_n \leq | B |$ (the chain is composable from B) be the behavior sequence that achieves a desirable outcome. There are four types of conflicts that can cause the chain C from not being executed. In each case, some sequence $\beta_{i_k} : \beta_{i_{k+1}}$, $1 \leq k \leq (n-1)$ must be broken.

(a)**Extraneous behavior Conflict :** $\beta_{i_k} : \beta'$, $\beta' \notin$C, it can be resolved by prioritization.

(b)**Cyclic Conflict :** This occurs because of an undesired repeatition of a behavior chain,e.g. $\{\beta_{i_p} : \beta_{i_{p+1}} : ... : \beta_{i_{q-1}} : \beta_{i_q} : \beta_{i_p} : \beta_{i_{p+1}} : ... : \beta_{i_q} : \beta_{i_p} : ...\}$. Here the chain preceding β_{i_q} executes after β_{i_q} and the sequence $\beta_{i_q} : \beta_{i_{q+1}}$ is broken.

(c) **Skipping Conflict :** $\beta_{i_k} : \beta_{i_j}$, $\beta_{i_j} \in$ C, j > (k+1). This type of conflict can be resolved by prioritization.

(d) **Control Conflict :** Stimuli of behaviors β_{i_k}, β_{i_j}, both belonging to the chain, are triggered simultaneously and these behaviors need the same resource to execute. This may lead to a deadlock if there is no arbitration mechanism.

4 Effects of Behavior Modifications

Here we investigate the effects of modifying the existing behaviors in a behavior space.

Many times stimuli are specialized to restrict the situations under which a behavior is triggered. For example, if the stimulus s_i of behavior β_i for cleaning dishes is $\exists x(dish(x) \bigwedge graspable(x))$ and it is desired that only those dishes that are on the table should be cleaned, then this stimulus can be specialized to $\exists x,y(dish(x) \bigwedge graspable(x) \bigwedge table(y) \bigwedge on(x,y))$. The action of a behavior can be modified so that it results in a weaker consequence. We call this modification *response generalization*. For example, consequence c_j of the behavior β_j for painting the side and top of a can is $\exists y,z$ (painted(y) $\bigwedge$ painted(z) $\bigwedge \neg$ equal(y, z)) (s_j is $\exists x,y,z$ (can(x) $\bigwedge$ graspable(x) $\bigwedge$ side(x,y) $\bigwedge$ top(x,z) $\bigwedge \neg$ painted(y) $\bigwedge \neg$ painted(z))). The action of painting can be modified to paint only the top, resulting in the weaker consequence $\exists z$ (painted(z)).

Lemma 3. Whenever a behavior space B is modified to B' by specializing stimulus s of some behavior $\beta \in B$ to s' so that $(s' \Rightarrow s) \bigwedge \neg(s \Rightarrow s')$, then $\frac{|\tau_G(B)|}{|B|} > \frac{|\tau_G(B')|}{|B'|}$.

Proof - Let s be specialized to s' so that $s' \Rightarrow s$. Now tasks or subsequent behaviors fulfillable or triggerable in (s - s') (this difference corresponds to the states of the world in which s holds but s' does not) will no longer be so. Thus we need a new behavior β'' such that ($s'' \cup s'$) = s, so that β' and β'' together serve the stimulus set s. This implies that $\mid B \mid$ increases and the usefulness of B decreases. Without the new behavior, $\mid \tau_G(B) \mid$ reduces. □

Lemma 4. Whenever a behavior space is modified by response generalization, the flexibility or usefulness of the behavior space decreases.

Proof - The consequence c of β is generalized to c' so that $c \Rightarrow c'$. Thus $(c - c')$ is not being made available by β. Let the behavior space containing β' be denoted by B'. Hence other new behaviors are needed to complete the tasks requiring (c - c'), this increases $\mid$ B $\mid$. This implies that the usefulness of B decreases. Addition of new modules increases the lengths of the chains composed to fulfill these tasks resulting in decrease in flexibility. If this not done, then some tasks requiring (c - c') (this should be interpreted as a pure difference of literals) cannot be fulfilled, this implies that $\mid \tau_G(B') \mid \ < \ \mid \tau_G(B) \mid$. This means that the usefulness of the behavior space decreases. □

Stimulus specialization and response generalization can be used to eliminate undesirable cyclic conflicts, e.g. consider the the cycle $\{\beta_{i_p} : \beta_{i_{p+1}} : ... : \beta_{i_{q-1}} : \beta_{i_q} : \beta_{i_p} : ... : \beta_{i_q} : \beta_{i_p} : ...\}$. This can be eliminated either by specializing s_{i_p} to s'_{i_p} such that $\neg(c_{i_q} \Rightarrow s'_{i_p})$ or generalizing the consequence c_{i_q} to c'_{i_q} so that $\neg(c'_{i_q} \Rightarrow s_{i_p})$.

Behavior Modification Theorem. Given two behavior spaces B and B' such that B is more powerful than B' (i.e. B' is obtained from B by replacing some behaviors β of B by the less powerful ones β') then:
(a) The greatest potential task space of behavior space B' is less than that of B, i.e.
$|\tau_G(B')|<|\tau_G(B)|$
(b) Usefulness of B is larger than that of B' i.e.
$\frac{|\tau_G(B)|}{|B|} > \frac{|\tau_G(B')|}{|B'|}$.
(c)Likelihood of a cycle in B is at least as large as that for B'.

Proof (a, b) - Let us consider the case where a single behavior β has been replaced by the less powerful β'. The set of chains of behaviors composable from a behavior space represents a tree with initial point corresponding to the availability of the right initial stimulus and each node in this tree represents a world state which may correspond to the desired state, indicating existence of a task fulfilling chain. The greatest potential task space is proportional to the total size of this tree of behavior chains (this size is defined in terms of the number of all possible chains contained by the tree and the number of such trees, since different trees can be constructed for different initial world states.) Either the behavior β will have more applicability due to smaller stimulus length as compared to the behavior β', or the behavior β will have stronger consequence resulting in more behaviors being triggerable after it or both. In terms of the task tree, either β will have more parent nodes, or it will have more children. In either case, the branching factor in B is larger than that in B' and the size of the task tree will be larger. Hence the result (a). Since |B| has not changed, the usefulness of the behavior space, $\frac{|\tau_G(B)|}{|B|}$ decreases when β is replaced by β', this proves part (b). This treatment can be extended to multiple instances of replacing a strong behavior by a weak one.□

Proof (c) - Let $\beta_i \in B$ and $\beta'_i \in B'$ be two behaviors s.t. β_i is more powerful than β'_i, i.e. $(s'_i \Rightarrow s_i)$ or s_i is at least as weak as s'_i. Now consider any chains of length n composable in B and B', which differ only in that the module β_i is replaced by β'_i. Now consider all behaviors $\beta_j \in C$, $\beta_i \prec \beta_j$, ($\prec$ denotes the temporal precedence operator) with consequence c_j. We define the likelihood of a cycle in B, denoted by $L_{cycle}(B)$ (which is not restricted to 0-1 range like probabilities) to be

$$\sum_{j \geq i}^{n} (prob(c_j \Rightarrow s_i))$$

Then $L_{cycle}(B')$ is

$$\sum_{j \geq i}^{n} (prob(c_j \Rightarrow s'_i)).$$

Clearly, since $(s'_i \Rightarrow s_i)$, $\forall j[prob(c_j \Rightarrow s_i) \geq prob(c_j \Rightarrow s'_i)]$. Similarly $(c_i \Rightarrow c'_i)$ for which similar analysis can be carried out. Thus $L_{cycle}(B) \geq L_{cycle}(B')$. If there is a cycle in B', there will be a cycle in B too. □

Modularity Theorem. Given cycle free behavior spaces B and B' (B' is obtained from B by replacing some behaviors of B by less powerful ones) and a behavior module λ not belonging to B and B' is added to both of them, then the space B$\cup\lambda$ can be at the most as modular as the space $B'\cup\lambda$.
Proof - The theorem follows from the part (c) of behavior modification theorem. As B contains more powerful behaviors, from part (c) of behavior modification theorem, $L_{cycle}(B) \geq L_{cycle}(B')$. Thus $L_{cycle}(B \cup \lambda) \geq L_{cycle}(B' \cup \lambda)$. Hence $\frac{1}{L_{cycle}(B \cup \lambda)} \leq \frac{1}{L_{cycle}(B' \cup \lambda)}$. This decreases modularity as defined in section 2. These results imply that more powerful behavior spaces are consequently less modular.□

Behavior Metrics Theorem. If $B_1 >_p B_2$, then $B_1 >_u B_2$, $B_1 \geq_f B_2$ and $B_2 \geq_m B_1$.

Proof - From behavior modification theorem, it follows that $B_1 >_u B_2$. From modularity theorem, it follows that $B_2 \geq_m B_1$. If $B_1 >_p B_2$, then B_1 is obtained from B_2 by replacing one or more behaviors by more powerful ones. Hence $\mid B_1 \mid = \mid B_2 \mid$. Consider some task fulfilling chain $C \lhd B_2$ such that $fulfills(C, t)$. Let $C = \{\beta_{i_1} : \beta_{i_2} : ... : \beta_{i_n}\}, 1 \leq i_1, ..., i_n \leq \mid B_2 \mid$. For each $\beta_{i_k} \in B_2$, $1 \leq k \leq n$, one can find corresponding $\beta_{i'_k} \in B_1$ such that $\beta_{i'_k}$ is at least as or more powerful than β_{i_k}. Hence $\beta_{i'_k}$ has at least as weak or weaker stimulus and/or at least as strong or stronger consequence. Hence there exists a chain $C' \lhd B_1$ such that $fulfills(C', t)$ and $\mid C' \mid \leq \mid C \mid$. Hence $B_1 \geq_f B_2$.□

5 Augmentation of Behavior Spaces

By augmentation of behavior spaces, we mean adding more behaviors to an existing behavior space.

Without proof we state,

Lemma 5. If disjoint behavior spaces $B_1, B_2, ..., B_n$ are scalable together, then $U(B)* \mid B \mid > (U(B_1)* \mid B_1 \mid + ... + U(B_n)* \mid B_n \mid)$ where $B = B_1 \cup B_2 \cup ... \cup B_n$.

Lemma 6. When p new behaviors are added to a behavior space B, $\tau_G(B)$ must increase by a factor of at least $(\frac{p}{|B|} + 1)$ for the usefulness to increase.

Lemma 7. If a behavior β' is added to behavior space B such that there exists $\beta \in B$ which is less or more powerful than β' and β and β' cannot execute concurrently, there exists a control conflict in the behavior space $(B \cup \beta')$.

Proof - Let us say that β' is more powerful than β. When β is triggered, β' will also be. If β' is less powerful than β, then when β' is triggered, β will also be. Hence in any of the two cases, β, β' will be triggered at the same time, resulting in a control conflict. Hence the proof.□

Behavior Space Augmentation Theorem. Elimination of cyclic conflicts places a bound on the power of behaviors that can be added to a behavior space.

Proof - Let the new behavior to be added to the behavior space B be β_k. There should not exist a chain $\{\beta_{i_1} : \beta_{i_2} : \beta_{i_3} : ... : \beta_{i_{j-1}} : \beta_{i_j}\}$ in the tree corresponding to $\tau_G(B)$ such that new behavior can come after this chain and the chain can immediately succeed the new behavior, resulting in the chain $\{\beta_{i_1} : \beta_{i_2} : \beta_{i_3} : ... : \beta_{i_{j-1}} : \beta_{i_j} : \beta_k : \beta_{i_1} : \beta_{i_2} : \beta_{i_3} : ... : \beta_{i_{j-1}} : \beta_{i_j} : \beta_k : ...\}$. This means that either (1) β_k should not occur next to β_{i_j} or (2) β_{i_1} should not occur next to β_k or (3) β_k should delete some literal a in stimulus of some behavior β_{i_p} in the chain $\{\beta_{i_1} : \beta_{i_2} : ... : \beta_{i_j}\}$ and no other behavior β_{i_q} in the chain should make a true. Satisfying this condition eliminates the cycle. Then intuitively, s_k should be stronger and c_k should be weaker (to fulfill 1, 2), so that β_k cannot both occur immediately after the existing chain and immediately precede the existing chain. This places a bound on the strength of s_k and c_k, hence the result.□

Ideally a behavior space should exhibit the same performance irrespective of the order in which the behaviors are added. The following result shows that this is not always true.

Lemma 8. If a set of behaviors $\{\beta_{i_1}, \beta_{i_2}, \beta_{i_3}, ..., \beta_{i_k}\}$ (such that these behaviors form a cycle free behavior space) is to be added to a behavior space B such that for all $1 \leq m, n \leq k, m < n$, β_{i_n} is more powerful than β_{i_m}, then there exist orders in which the behaviors can be added to detect and avoid cycles in the resulting behavior space.

Proof - One can add the most powerful behavior β_{i_k} first and check if the behavior space $(B \cup \{\beta_{i_k}\})$ contains a cycle. If the new behavior space is cycle free, adding one or more less powerful behaviors will not introduce a cycle and behaviors from the remaining set $\{\beta_{i_1}, \beta_{i_2}, \beta_{i_3}, ..., \beta_{i_{k-1}}\}$ can be added in any order. If addition of β_{i_k} results in a cycle, one can avoid adding it and add remaining behaviors in decreasing order of power. This ensures that the largest subset of $\{\beta_{i_1}, \beta_{i_2}, \beta_{i_3}, ..., \beta_{i_{k-1}}\}$ that does not cause a cycle when added to B, is added to B. One can add the least powerful behavior β_{i_1} first and check for the existence of cycle in the space $(B \cup \{\beta_{i_1}\})$. If $(B \cup \{\beta_{i_1}\})$ contains a cycle, adding any other behavior will also result in a cycle. Hence the proof.□

6 A Case Study

Let us consider a behavior-based robot working in kitchen. Let its behavior space consist of following behaviors - $open\_water\_tap(\beta_1), close\_water\_tap(\beta_2)$, $open\_refrigerator(\beta_3), close\_refrigerator(\beta_4)$, $open\_microwave(\beta_5), close\_microwave(\beta_6)$, $pick\_up\_dish(\beta_7), deposit\_dish(\beta_8)$, $pick\_up\_can(\beta_9), deposit\_can(\beta_{10})$, $open\_trash\_can(\beta_{11}), close\_trash\_can(\beta_{12})$, $pull\_chair(\beta_{13}), push\_chair(\beta_{14})$, $turn\_on\_microwave(\beta_{15}), set\_microwave\_timer(\beta_{16})$, $pick\_up\_box(\beta_{17}), deposit\_box(\beta_{18})$, $avoid\_obstacle(\beta_{19})$ and $wander(\beta_{20})$. This behavior space has the potential to fulfill a number of tasks like washing dishes (by the chain $\{open\_water\_tap : pick\_up\_dish : deposit\_dish : pick\_up\_dish\}$ when the *deposit_dish* behavior puts the dish below the water tap), moving food boxes into and out of refrigerator, trashing objects and preparing lunch and dinner using microwave.

The behaviors β_7, β_9 and β_{17} for picking up dishes, cans and boxes can be replaced by a single more powerful behavior whose stimulus is $\exists x((can(x) \bigvee box(x) \bigvee dish(x)) \bigwedge graspable(x))$. This maintains the greatest potential task space constant and lowers the size of the behavior space. Hence usefulness of the behavior space increases.

Sometimes cans, boxes and dishes may be picked from and deposited at different places of the same empty flat surface. This requires a chain of at least three behaviors, e.g. $pick\_box, wander$ and $deposit\_box$. Similar chains will be used to move cans and dishes. To reduce the lengths of these chains, one can add behaviors like $push\_can, push\_box$ and $push\_dish$. This makes the augmented behavior space more flexible than the original behavior space. However since the size of the behavior space increases by 3, there may be a reduction in the usefulness of the behavior space. We leave it to the user to decide which metrics are important. However the potential loss of usefulness can be curbed by replacing the 3 specific *push* behaviors by a more powerful *push* behavior that can push dishes, boxes and cans. This will increase the size of the behavior space by only 1.

If one unions the existing behavior space with new behavior space $\{turn\_on\_oven, set\_oven\_temperature, turn\_off\_oven\}$, then greatest potential task space increases, indicating that the two behavior spaces are scalable. However there are potential cyclic conflicts like opening and closing water taps, opening and closing refrigerator doors, opening and closing microwave and turning the oven off and on an arbitrary number of times. This indicates that the modularity of the behavior space lowers when it is augmented with behaviors for using the oven.

7 Conclusion

The results provided here offer guidelines for the evaluation and modification of the behavior spaces. If one expects different functionality out of a behavior space, one need not always add new behaviors or redesign the behavior space from scratch. One can inspect individual stimuli and consequences and see if modifying them can fulfill the required functionality. Our metrics and proofs apply to stimuli and consequences expressed in conjunctive normal form as well. A weaker stimulus expressed in conjunctive normal form has either fewer clauses or more literals in each clause or both. A stimulus can be made stronger (and hence specialized) either by adding more clauses or dropping literals from existing clauses or both. The consequences can be made weaker or stronger in similar way. The power metric we developed cannot compare stimuli not related by the implication ($\Rightarrow$) relationship. This applies to consequences as well. However we do not see this as a limitation of our metric, since comparing arbitrary stimuli or consequences does not make sense (one can develop other statistical measures for that).

The metrics that we developed can be extended to handle other representations as well, where stimuli and consequences are expressed as values in potential fields or degrees of memberships of fuzzy sets. In our notation, stimuli and consequences are boolean, they are either true or false. In the potential field-based approach, goals and obstacles are modeled by potential functions, the obstacles exert repulsive force on the robot and the goals exert an attractive force. The summation of these forces is used in computing the next velocity vector. Manipulating the clauses in stimuli and consequences to make them stronger or weaker corresponds to changing the potential function definitions and membership functions of the fuzzy sets.

Whatever the chosen representation for stimuli and consequences may be, it will be important to have behaviors of higher power that provide higher functionality by executing under a wider range of situations and by creating stronger consequences. It will be necessary for the behaviors to get chained to fulfill more complex tasks. The amount of chaining will continue to be a measure of the usefulness of the behavior space. Though the behavior chains may be task fulfilling, they cannot be arbitrarily long. The flexibility metric captures this. The behavior spaces may need new behaviors if the owner of a robot expects it to fulfill more goals. However the augmentation should not weaken the original functionality. Hence scalability will continue to be important. Though it is desirable to have highly powerful, useful and flexible behavior spaces, they should not have undesirable interactions among the behaviors. Hence the modularity metric we defined will continue to be relevant. Hence the utility of our metrics is independent of representation. Though one can redefine our metrics to handle other representations, our current representation itself is valuable in measuring the performance of behavior spaces and modifying them intelligently.

References

1. **Arkin, R. C.** 1992. Behavior-Based Robot Navigation for Extended Domains, Adaptive Behavior 1(2): 201-225.
2. **Brooks, R. A.** 1986. A robust layered control system for a mobile robot, IEEE transactions on robotics and automation, 2(1): 14-23.
3. **Connell, J.** 1990. Minimalist mobile robotics, A colony style architecture for an artificial creature, Academic press Inc.
4. **Mahadevan, S., and J. Connell,** 1992. Automatic programming of behavior-based robots using reinforcement learning, Artificial Intelligence, 55: 311-365.
5. **Reiter, R., 1991.** The frame problem in the situation calculus: A simple solution (sometimes) and a completeness result for goal regression. In Lifschitz V., editor, Artificial Intelligence and Mathematical Theory of Computation: Papers in Honor of John McCarthy, pg. 359-380, Academic press, NY
6. **Pattie Maes and Rodney Brooks,** Learning to co-ordinate behaviors, Proc. of AAAI-90, 796-802.
7. **David Miller, Rajiv Desai, Erann Gat, Robert Ivlev and John Loch**, Reactive navigation through rough terrain: Ex-

perimental results, Proc. of AAAI-92, 823-828.

8. **James Montgomery, Andrew Fagg and George Bekey**, The USC AFV-I, IEEE Expert, Vol. 10, No. 2, Apr. 1995.

9. **Rajiv Desai, Charles Rosenberg and Joseph Jones**, Kaa: An autonomous serpentine robot uses behavior control, Proc. of IROS-95, 250-255.

10. **David Wettergreen, Henning Pangels and John Bares**, Behavior-based gait execution for the Dante-II walking robot, Proc. of IROS-95, 274-279.