INTRODUCTION

This paper describes an integrated system for prioritizing compounds in the context of discovery chemistry projects. The system has its origins in 2001 when we began experimenting with iterative low throughput screening combined with a space-filling design approach, as an alternative to higher throughput screening of molecules in early stage discovery projects. This approach was utilized while testing molecules for potency and selectivity at 15-hydroxyprostaglandin dehydrogenase (15-PGDH). The starting point was a few published documents and about 30–40 molecules that were tested in our in-house assay. In four rounds of low throughput testing, with 250 molecules per round, the effort progressed from having micromolar inhibitors that were not competitive to finding single digit nanomolar competitive inhibitors of 15-PGDH. To our knowledge, there were no examples of competitive inhibition in the literature. One year later, the same approach was used on a different project, where the target was 11β-hydroxysteroid dehydrogenase type 1 (11bHSD1). In three testing rounds of 500 compounds per round, this method led from double digit micromolar to low nanomolar and selective inhibitors.

After these early successes we continued using this new approach and at the same time began evaluating our traditional way of following up on actives from screening efforts in a formal process referred to as actives assessment (AA). We observed that most project teams were taking active molecules and applying filters (potency, molecular weight, etc.) to reduce the set of actives to the most interesting “seed” molecules. By use of these as new starting points, molecules in our collection were identified that had not been tested in the screen but were similar to a seed molecule. This became the expanded set of candidate molecules. At this point, a cluster analysis was performed on those “hit expansion” sets of molecules. Cluster analysis (first coined by Tryon, 1939) includes a number of different methods for grouping objects of similar kind into respective categories. In the current context, molecules are divided into groups by an algorithm, and an algorithm is deemed suitable if the groupings are a good approximation of how a chemist would manually separate the molecules into different groups. Readers unfamiliar with cluster analysis have a wide range of sources for gaining familiarity; perhaps a good starting point is Bacha et al.1 In our actives assessment setting, centroids of clusters became interesting molecules for each round of testing (typically 1000 molecules each round). In reviewing this process, we noted several inefficiencies in the way we were expanding on actives, and we summarized with these three recommendations for actives follow-up:

1. Do not ignore negative information (inactive molecules). Molecules lacking potency give valuable information not to be ignored.
2. Do not use hard filters. This is a brittle approach that can deny one molecule while allowing another with very similar attributes. It leads to lost opportunities.
3. Do not use cluster analysis as a design tool. There was a heavy dependency on cluster analysis as a way to find representative compounds to explore. This tends to bias too heavily toward molecules merely because they have many close analogs.

The third point is especially important at early stages of screening just after medium throughput screening. Often there...
is a fixed number of molecules to select for follow-up, but many cluster analysis algorithms have no control over the number of clusters. If the user needs to select 1000 molecules, a cluster analysis may give an inconvenient number like 2877, leaving the user to tweak the parameters of the cluster analysis over several iterations in search of the 1000 molecules they need. There are cluster analysis algorithms that give control over the number of clusters, e.g., Kmeans. These are often not used because the uniformity of the clusters and what chemists consider to be natural groupings of the molecules tends to suffer when a specific number is forced. Tweaking parameters to control the number of clusters is an inappropriate manipulation of an algorithm to accomplish something that is really not natural for it. (If the goal is to analyze an SAR, cluster analysis may be ideal; in this work the focus is on prioritization.) We implemented the new approach to actives follow-up in a system called the molecule selection toolkit (MST). The MST is now generalized for both early and later stages of discovery chemistry project work including medium throughput screening (MTS), actives assessment (AA) and formal hit assessment (FHA), hit to lead (H2L), lead optimization (LO), and candidate selection (CS).

**Components of the System.** The main components of the MST are the following.

1. Predictive modeling (using all information, including inactive molecules)
2. Utility functions (avoiding hard filters)
3. Weighted desirability (optimizing many things, not just potency)
4. Biased spread design (a powerful design that augments what has been learned)

The topic of predictive modeling deserves an in-depth coverage but is beyond the scope of this paper. Recent advances are built into the MST, but here we will simply say that we use our own implementations of both random forest and support vector machines, and for both methods we have adapted good ideas developed by Sheridan\(^1\) for rescaling of predictions and for reliability of prediction. Depending on the goal, experimental data may be available for candidate compounds. It should be noted that a multiparameter optimization (MPO) system needs a way to deal with missing values. When experimental data are used, there are often molecules that lack experimental values and an imputation system must be used to fill in those missing values. Often this can be done using the same machine learning methods that would be applied in the case where none of the molecules have experimental data, but there are many other ways to do data imputation, outside the scope of this work. It should also be noted that not all values are of equal reliability. An experimental value from an assay with low noise relative to signal has higher reliability than a measure from a different assay with higher noise. A missing value could be treated as yet another measure with even lower reliability. Importance weighting of various measures can reflect the reliability of the measures as well as their importance to the goals of the project. This paper focuses on the implementation of utility functions and weighted desirability, which form the basis of the MPO component, and biased spread design (BSD), which gives a design component to the way screening efforts are iterated.

**MULTIPARAMETER OPTIMIZATION: A BRIEF REVIEW**

For more than two decades now, there has been a strong message in pharmaceutical discovery that one must optimize more than just potency. Long gone are the days of sequential design where potency is optimized first, then druggability, and finally safety. Our industry has become a high pressure environment in which one is compelled to address druggability and safety concerns earlier in the process and navigate trade-offs between potency of molecules and other properties. The realization of this goes back to 1997, when the rule of five in the design of oral drugs was initially proposed by Lipinski et al.\(^3\) Although this was not a true multiparameter optimization (MPO), it was one of the earliest publicized attempts at a method to “score” a molecule in a higher dimensional landscape. Soon after that, numerous groups published their own version of “rules” for CNS exposure or for druglikeness of molecules. Along the way there have also been various ratios proposed, such as various ligand efficiency measures LE\(^4\), LLE\(^6\) and LELP\(^7\), and the number of specialty ratios is growing. These are specialty measures for optimizing 2–3 aspects of a molecule. For example in our company, potency and size are captured with a ratio called “LEAN,” which is simply IC\(_{50}\) in molar units divided by the number of heavy atoms. Many of the proposed ratios and “rules” are static, fixed functions, reviewed in Garcia-Sosa et al.\(^6\) We argue that adding more specialty measures and more rules, or refinements of rules, will give diminishing returns. What is needed in this area is a simple system for prioritizing molecules that is flexible and easily tunable to the current need. We also feel that multiparameter optimization is not the end but one step in a process that should include components of predictive modeling and design as well.

A significant step beyond the rule of five came from another group at Pfizer, who developed a “CNS MPO” that uses six properties of a molecule, chosen to improve not only brain penetration but also permeability, safety, and clearance.\(^7\) This is a significant step beyond the rule of five because it avoids the use of hard filters. However, it is a fixed set of functions which, as such, lacks flexibility for other purposes. For example, a team optimizing fragments would not be able to use the tool without modifying it to restrict molecular weight, one of its six components, far more than the CNS MPO does. Such a team could develop their own MPO that is adjusted for that purpose. Another team working on a cancer related target might accept higher molecular weight molecules and would need to loosen the limits on molecular weight beyond the CNS MPO. It is easy to foresee the dispensing of many specialized MPO sets ad nauseam. Indeed, a group has already published another paper with a modified set of functions called the “CNS PET MPO.”\(^8\) One might go farther to imagine various target classes: kinases, GPCRs, cell surface receptors, and different therapeutic areas motivating many different variants of an MPO for use in project work. Even the same project team will have different needs a year in the future than at present. Creating and tracking any number of different specialty MPOs (as static functions) are not needed if there is a sufficiently flexible and powerful system to optimize what is needed at the current time. We have developed a system that automatically creates custom functions to optimize any number of end points (or properties), on the fly, for use in compound prioritization in discovery chemistry.
There are numerous papers on the topic of MPO. Notable for their early contribution, Biswas et al. proposed a drug likeness score called DLS in 2006. The form of the utility functions constructed was driven by distributions of each property among marketed drugs (possibly a disconcerting practice for driving future drug design efforts). Cruz-Monteagudo et al. proposed, in 2008, a method using Derringer’s desirability for optimizing the potency, safety, and bioavailability of compounds. An even earlier contribution is Gillet et al. in 2002, which treats diversity as another parameter to be optimized along with various properties and then uses a genetic algorithm for optimization. A more recent contribution is that of Nissink et al. in 2013, which develops a uniquely different approach motivated philosophically by Karl Popper’s ideas of falsifiability of hypotheses in science. This approach seems to deal with missing data by reducing the dimensionality of the data for molecules that have missing data rather than imputing those missing values. Thus, comparisons between two molecules are made in a context whose dimension varies, based on the number of attributes for which both compounds have experimental data. For example, if there were seven properties of interest, then two compounds, call them A and B, may be compared in a 7-dimensional space, while compounds A and C may be compared in a 5-dimensional space, if compound C is missing data in two of the end points of interest. Thus, the same compound A is compared to B in a 7-D space but compared to C in a 5-D space. This seems a most curious approach. For additional review, numerous publications of note include Ekins et al., Segall et al., Segall et al., Debe et al., and Yusof et al. What we have implemented is equivalent to a smoothed version of the desirability functions of Derringer, perhaps more similar to Harrington’s desirability functions. In this paper we use these terms interchangeably: “utility function”, “desirability”, “score”. These will always refer to a number between 0 and 1 that measures the desirability of a single attribute, or in the weighted aggregation it will be referred to as “overall desirability.” Often in the literature there is a discussion of Pareto optimality. The references given above, as well as this work, refer to Pareto optimality as important for an MPO system. A molecule is Pareto optimal if, when examining the aspects being considered, further improvement (say, in the form of a small substituent change) to any one aspect would come at the detriment of one or more other aspects of that molecule. If a molecule cannot be improved in any one aspect without losing ground in one or more other aspects, then that molecule is a Pareto optimal choice. Note that is only defined within a fixed set of solutions, in this case molecules; it is not defined as a claim about all possible solutions.

THE MOLECULE SELECTION TOOLKIT: A HIGH LEVEL OVERVIEW

We will cover in more detail the components of the MST and then show how they are integrated together. Let us begin with the high level architecture of the MST as given in Figure 1. Looking at the figure from left to right, the starting point is the choice of raw end points. These may be properties, potency, metabolism, in short any aspect of a molecule that can be expressed as a number. The word “raw” conveys that the different aspects will have their own units of measurement; for example, metabolism may be in units of percentage from 0 to 100, while potency is often in nanomolar units ranging from single digits into the tens of thousands. These end points represent the optimization opportunities and hurdles that a given team may face at a particular point in time, where in vitro or in silico surrogates are used to capture important aspects of
molecules. The molecules could be virtual or in an existing collection; either way, predictive modeling is often a heavy component in producing the raw numbers for these inputs. The list of end points as shown in Figure 1 is certainly not exhaustive. There are numerous ADME, toxicological, and related in vivo end points that could be (and often are) included in the inputs. Since each of these may have their own scale of measurement, each end point is given a utility function, which converts the raw end point to a number between 0 and 1, representing how a molecule is valued in that end point. This is covered in more detail in the next section.

At this point, there is a set of desirability numbers, each between 0 and 1, and importance weights are assigned to these numbers. As a motivating example, one project team may deem potency to be five times more important than metabolism, while another project team has an overabundance of potent molecules which all suffer from extreme metabolism; thus for this second team the heavier importance may be placed on metabolism. Yet another team may have a set of antitargets for which selectivity of the molecules is the biggest challenge. This process is applied to the entire list of end points, which often includes 15–20 different properties. A later section discusses how to calibrate the assignment of importance weights. The utility functions are combined into a weighted desirability, using a weighted geometric mean. There are numerous ways to aggregate a group of individual desirabilities into an overall desirability. One could envision a spectrum, with “Min” and “Max” representing extreme ends of the spectrum. Using Min means that the overall desirability is the property that is least desirable, ignoring all other properties. Using Max ignores everything except the property that is most desirable. An arithmetic mean is right in the center of that spectrum, with no bias toward either the best or worst aspect of a molecule. A geometric mean is more influenced by the weakest aspects of the molecule than an arithmetic mean, and there are other functions (hypergeometric, inverse geometric, not defined here) that lie between arithmetic mean and Max in this spectrum. In any case, having chosen an aggregation method (geometric mean in our case), there is now a single importance weighted desirability number for each molecule, and one could sort the molecules by this number and select from the top of the list. This will likely be suboptimal, because the molecules at the top of the list are often very similar to each other, and effectively represent testing the same hypothesis repeatedly (this varies depending on the diversity and size of the molecule set). A design component is needed in order to maximize the efficiency of sets of molecules chosen for testing. The use of a biased spread design allows one to explore a wider hypothesis space and select a more efficient set of molecules for testing. As will be shown in the Biased Spread Design section, this not only gives molecules that are different from each other and yet desirable but also strategically augments the set of molecules that have previously been tested. Modern design is a critically important step in enriching the information content of molecules selected for testing. After the molecules are tested, the results are processed, any predictive models are evaluated and rebuilt with the new information incorporated, and the learning cycle begins again.

### Utility Functions

In reviewing the literature, the reader will see terms such as “utility function”, “desirability”, and “scoring function”. Often in the pharmaceutical discovery chemistry community, these terms are synonymous. Utility functions have their origin in economics and finance theory and are used to capture preference with respect to a specific good or service. A desirability function is a mathematically simplified description of a decision maker’s preference. It transforms an objective function to a scale-free desirability value, which measures the decision maker’s satisfaction with the objective value. The distinction between preference and desirability is subtle and not important for the use made of this concept here. A utility function can come in many forms. A “filter” is something commonly used in discovery chemistry; this is a utility function that takes on only two values, for whether a molecule “passes” the filter or not. This is simply a step function, for example, if one requires a molecule to have a molecular weight (MW) of 350 or less; the result is shown in Figure 2. This seems quite brittle, as a molecule with a MW of 351 is then deemed completely unacceptable while another molecule whose MW is 349 is completely acceptable. One of the first things we did in restructuring our AA efforts was to recommend against hard filters like this. A more reasonable utility for MW can be constructed by answering the following three questions:

1. Are larger values desirable or undesirable?
2. What is the smallest value below which two molecules should be considered equivalent so that any further effort is directed to other aspects to improve?
3. What is the largest value above which two molecules should be considered equivalent so that any further effort is directed to other aspects to improve?

Note that the word “equivalent” in questions 2 and 3 does not imply desirable or undesirable. The answer to question 1 will determine what is desirable and what is undesirable. Once these three questions are answered, it is very easy to compute a smooth utility function. For example, if a team is targeting a receptor with a small molecule, questions 1–3 may be answered as follows: large values of MW are undesirable (Q1), a very good MW is 250 or lower (Q2), and a completely unacceptable MW is 400 or higher (Q3). This leads to a utility function that is sigmoidal downward in shape, with a linear portion between 250 and 400, as shown in Figure 3. A value of 1 is a perfect score, while 0 is a reject. The rationale for doing this is twofold:

1. Setting up to optimize more than one thing requires handling differences of scale. Converting each end point

![Figure 2. MW filter as a step utility function. Here a weight above 350 is a failure.](image-url)
One can accommodate different importance weights to reflect the current priorities of a project. Often at an early stage, priorities are not well established, and so an equally weighted desirability can be used. This is commonly the case when building libraries for medium throughput screening. For project work where strengths and weaknesses of each series being investigated are known, usually there is a sense that some issues are critically important to solve, others are moderately important, and still other issues are either already solved or not considered much of a threat to the goals of the project. Sometimes an end point is just being tracked, without the need to improve that aspect of the molecule but rather just to keep an eye on it.

Suppose for example that there are four end points of interest, numbered 1 through 4. They are assigned importance weights $a$ through $d$. A weighted geometric mean can be used to compute a weighted desirability score as in eq 2.

$$Dscore = (u_1^a \times u_2^b \times u_3^c \times u_4^d)^{1/(a+b+c+d)}$$

If, say, $u_1$ is the utility function for an end point that the team simply wants to track but not use as a criterion for prioritizing molecules, then setting $c = 0$ would accomplish this. The system then computes the end point, but it has no effect on how the molecules are prioritized. As has been pointed out numerous times in the literature, a geometric mean finds Pareto optimal solutions.

It should be noted that the geometric mean is multiplicative: several numbers, each between 0 and 1, are multiplied in the
computation. If any single attribute of a molecule gets a value of 0, then the overall result will be 0, regardless of how the other attributes score. To avoid this, a "shift value" is used in all of our utility functions. In effect, the utility curve bottoms out at 0.10 (our default value) instead of at 0. The shift value can be altered to any number desired and does not affect the shape of the utility function in the location of inflection points. Looking back at eq 1, the reader will notice a "0.0" in the equation in two places. Changing these to 0.10 achieves the 0.10 shift.

Calibrating the importance weights is crucial. Two approaches for this will be described. Both approaches require a set of molecules that have experimental results. One set should be large, say hundreds to thousands of molecules if that many are available. Let this be called set A. Another set should be a more focused, smaller set, which may include 20−50 molecules that are very well understood. Let this be called set B. Both sets A and B should include some highly desirable molecules, some molecules that are highly undesirable, and some molecules with moderate desirability (here we mean a qualitative judgement call, not a desirability number). An easy first approach is to start with an initial estimate of the importance weights and to compute a Dscore for set A. Sorting the set by this Dscore, one can observe where a few (usually 2−5) favored molecules fall in the prioritization. Ideally they will be listed in the top 5% of the molecules and better yet would be in the top 10−20 molecules. (Demanding that they be at the very top of the list runs a risk of “overfitting” to the molecules used to calibrate.) If not, close examination of the molecules that were ranked higher than the favored molecules often suggests one (or both) of two things:

1. The utility function for one or more end points was either too “lenient” or too “strict.” A lenient function would accept molecules that the team considers unacceptable, and a strict function would reject molecules that, though not ideal, the team would want to explore.
2. Some aspect of the molecules was given an importance weight that is too low (or high). Increasing (or decreasing) the importance weight for this aspect will cause the favored molecules to “percolate” higher toward the top of the list.

It is often very helpful to go through the process of calibrating the weights and utility function shapes in order to see the most favored molecules rise in a prioritized list. Often a team comes to realize that what they thought was most important was actually less important than they realized, and other aspects of molecules were more important than they realized, once they see the concrete impact. This is part of human nature, that one is not able to grasp a complex set of values until the impact of them is seen in a concrete way. For an interesting discussion of the impact of abstract versus concrete information on decision making, see Borgida et al.20 There is an interface to a tool that is very helpful for getting initial importance weights. This is an approach created by a mathematician in the 1970s, called the analytical hierarchy process (AHP).21 The method uses pairwise comparisons to measure preferences with an ordinal score. By focusing on only two things at a time, it allows the user to focus attention locally while at the same time constructing a postanalysis that ranks all the choices. A related paper that also uses the AHP is Petit et al.22 The AHP requires constructing a matrix that represents all
pairwise preferences, but we have developed an approximation that is far easier to use. Our interface gives an absolute importance line and allows the user to drag and drop labels onto this line. The distance between items on the line is then used to infer pairwise preferences, and these in turn are used as input to the AHP algorithm. This makes decision making a very easy matter of dragging and dropping labels visually. Figure 6 shows an example of this, where the relative importances (arbitrarily ordered for illustrative purposes only) are toxicity < properties = selectivity < ADME < potency and numbers assigned are used to compute pairwise preferences. Thus, properties is given the same importance as selectivity. Potency is significantly more important than properties, while ADME is mildly more important than properties. These numbers become the weights in a geometric mean that gives the overall desirability ($D_{score}$). How “properties” is defined is not considered at this level; this is handled at a later stage when the user decides which properties will be used (e.g., molecular weight, clogP, PSA, rotatable bonds, $pK_a$, etc.) and how they are weighted versus each other. At this initial level these are high level groupings, or aggregations, of aspects of a molecule, tuned to a team’s needs. For example toxicity may involve several specific end points for one team and a different set for another team, with surrogate measures chosen to estimate each end point. Our approach is to compartmentalize the aspects of molecules at a high level and later break down each grouping and within that group also assign importance weights. In most cases, we have used the less tedious first method of calibration, which also uses a larger data set. It is instructive to see well-known reference molecules embedded in a larger set and to inspect the rankings of the whole set.

We described a simple first method to calibrate importance weights. The more difficult method requires a smaller set of 20−50 well understood molecules (described as set B above). This is a team effort. The molecules are laid out in a table, one molecule per row, with the raw end points spread out in columns. Now the team must manually prioritize these molecules, considering trade-offs between the end points. Additionally, the structures are hidden and the identifiers are masked. This forces the team to make tough decisions about trade-offs between the end points, which at times are conflicting; improving one aspect of a molecule comes at the expense of less desirable values in another aspect. It removes personal biases arising from chemists who may be attached to molecules they have synthesized or designed. For example, more potent molecules may have poor solubility or poor permeability, or better selectivity ratios may incidentally increase an unwanted CYP inhibition. Once a consensus prioritization is given for this calibration set of 20−50 molecules, the importance weights can be calibrated to mimic the rankings from this manual prioritization. (This can be done on the log scale using simple linear regression with the manual ranks as the response.) It is usually not possible to exactly replicate the ordering that was manually assigned by the team, but usually the bigger issue to resolve is conflicting orderings from one team member versus another. The discussion needed to resolve these differences of opinions usually adds great value to the project.

At this point one may be concerned, and rightly so, about using past data to calibrate a procedure to be used prospectively. Our rationale is this: if we had infinite time and infinite resources, we might look at every candidate molecule and exert our expert judgment at a fine level. Since we do not, we take a subset of molecules that we can thoroughly understand and calibrate an algorithm to mimic what we would do manually (being careful not to overfit). Then we unleash it on the larger set of candidate molecules, we obtain experimental results for those molecules, and we assess and recalibrate the algorithm. This procedure should reflect the expert judgment of the chemists involved rather than a completely computational approach such as merely looking at the properties in marketed drugs (taken as a surrogate for “desirable” compounds). One may also wonder what should be done in the extreme case where there are no desirable molecules for driving a calibration, such as an orphan disease or target. In that case one can only use a decision tool like the AHP and the expert opinion of team members. A calibration step can (and should) be repeated periodically as experimental data are generated in the project.

**BIASED SPREAD DESIGN**

Spread design is a greedy, sequential approximation to a modern space-filling design, proposed by Kennard and Stone in 1969. The objective of the method is to find a maximally diverse subset of a larger candidate set of molecules. This is done by finding, at each step in an iterative process, the molecule whose nearest neighbor distance among molecules selected thus far is maximized; thus each additional molecule added to the selection set is chosen to maximize a sense of information gained by the addition of that extra molecule. An efficient implementation of this is proposed in Higgs et al., and a heuristic explanation is given in Cummins (both available online). Space-filling designs have origins in spatial problems like sphere packing and oil well drilling. The analogy of oil well drilling is a good way to visualize how this design is used in discovery chemistry. Suppose we drill for oil in a particular location and find none. (Or if oil is found, similar logic applies.) We may ask where the best place to drill next is. If we drill very close to where we just drilled, we may waste time and money finding the same result. However, if we drill too far away, we may miss something in between. There is a trade-off between the cost of the exploration and the information returned. The cost constraints will impose a limit on how many drilling explorations can be done. This is quite analogous to the discovery chemistry setting, where one can only screen (or synthesize) a finite number of molecules in each round of discovery. Consider Figure 7, and suppose that this is a plot of potential places where we could drill for oil. Now suppose a constraint such that we can only drill in 30 places, so we must pick 30 of the light gray dots in the plot that best represent the space we wish to explore. This is shown in Figure 8. Some areas of the plot are far denser than other areas (dots very close together), but the density or sparsity of the dots reflects surface topography and not what lies beneath the surface (oil reserves or not). The topography on the surface creates limitations in where we can drill, but the oil reserves beneath the surface are the only thing that really matters. (Tying this analogy back to our industry, there are limitations in our ability to synthesize some molecules but what really matters is designing the best molecules for the target of interest.) A cluster analysis would draw the points toward the dense areas (often these would be molecules that are easy to synthesize), but a spread design ignores the density of points and optimizes the spatial exploration alone. On page 868 of Higgs et al., there is a modification proposed to allow for accommodation of a “demerit.” This is what the MST uses and
which we refer to as a bias. Thus, biased spread design in the
MST uses a renormalized version of the demerit of Higgs. Continuing with the analogy of oil drilling, if we have
information about where oil may be found, the design shown
in Figure 8 does not reflect that. Suppose that we had found oil
in a prior exploration, and this oil was found in the upper right
corner of the plot in Figure 7. If this were the case, we may no
longer consider the unbiased spread design of Figure 8 to give
the most informative set of 30 locations to explore. Instead, we
may wish to put more emphasis in the area where oil was found
while at the same time also pushing outward from that area to
enhance our learning. In the extreme, if we focus only in that
area, we might have a design as shown in Figure 9; this is
actually not a design at all. Figure 9 represents the opposite
extreme from that of Figure 8. In Figure 8 we have a pure
design with no bias; in Figure 9 we have pure bias with no
design. A compromise is shown in Figure 10, where we have
biased the design points in a direction toward the upper right
area in order to enhance success, but we have also pushed
outward in order to enhance learning.

In Figure 7 and those following, we have maintained an
analogy of oil drilling in order to help the reader visualize the
concepts. In actuality, the plots come from an active project in
our company and are a two-dimensional projection of the
structures of a large set of virtual molecules considered for
synthesis and testing. This kind of plot, often called a "structure
space plot", can be achieved using either a principal
components or multidimensional scaling algorithm, which is
outside the scope of this paper. The projection itself is only
used to aid visualization; the algorithm does not use this
projection but rather uses distance between molecules. Details
of this distance are discussed in Higgs et al.24 and Cummins;25
we will simply describe it as either a Mahalanobis distance or a
Tanimoto based distance, depending on what type of analysis is
done. For visualization purposes, each dot represents a
molecule, and dots close together are molecules that are similar
to each other. The dots could be colored by overall desirability
or by structural series or cluster number if such an analysis is
done. We feel that this kind of visualization increases the
temptation to attempt a manual selection of molecules rather
than a designed approach, which we advocate and implement
with the biased spread design.

We stress that the bias in Figure 10 is one of many possible
trade-offs between desirability of the molecules and pure space-
filling design. There is a whole spectrum possible. This point is
so important that we risk belaboring it with Figure 11 and
Figure 12, which show levels of bias between that of Figure 8
and Figure 10. We have found that the context requires
differing levels of desirability or design, perhaps roughly
summarized as follows:

1. Medium throughput screening (MTS) across targets and
   projects: unbiased spread as illustrated in Figure 8.

Figure 7. Two-dimensional projection of structures of molecules for
an active project. One could think of this in the oil drilling analogy as
potential places to drill for oil, where cost constraints allow only 30
exploratory drillings, and so the most informative 30 points must be
chosen.

Figure 8. One extreme: spread design, unbiased, with 30 design
points.

Figure 9. Other extreme versus Figure 8: top 30 molecules with no
design component.

Figure 10. Compromise: biased spread design with a moderate bias
toward desirability.
2. Actives assessment (AA) following MTS: small bias as in Figure 11.
3. Hit-to-lead efforts (H2L): moderate bias as more is known, Figure 12 or Figure 10.
4. Lead optimization: high bias as much is now known, Figure 10 or Figure 9.

The Supporting Information, available online at http://pubs.acs.org, shows the interface for the MST, and this includes diagnostic plots that help the user visualize, in a practical way, the trade-off between desirability and design (treating “design” and “diversity” as roughly equivalent).

One thing that is evident in the figures with spread design (biased or not) is that the design points are “pushed” apart from each other. The degree to which they are repelled from each other depends on the amount of bias used in the design. The outcome is that the top of the rank ordered list will be molecules that are different from each other, to some degree tunable by the user. Unbiased spread design will very quickly find all the unusual molecules in a supplied set. This is often quite useful for “cleaning” a compound set, as often these molecules are undesirable and sometimes their presence was unknown or overlooked (a different purpose than prioritization but often an important first step).

There is an additional capability to this analysis, and it is very powerful. One can create a “previously selected” list containing molecules that have already been tested in previous rounds of screening. The design will augment this set of previously tested molecules to give an optimal design not only among the candidate molecules from which a subset is desired but also in the sense of augmenting what has already been learned in previous testing. Thus, one maximizes the information returned from the next round of testing. If the goal is to select molecules for testing in an enzyme assay, then the list of previously selected molecules will be all compounds that have been tested in that enzyme assay. If the goal is for testing in a cell-based or an in vivo assay, the list will be the corresponding set that have been tested in the same assay. Molecules that have not yet been tested but are slated for testing would also be in this list.

To illustrate the effect of this, Figure 13 shows an unbiased spread with 30 random starting points (asterisks) and 30 design points (solid boxes). Note that the random selection is largely drawn in to dense areas in the center of the map. With these marked as previously selected, the design points augment that set so that the final set of 60 points has the best possible space-filling properties. The 30 starting points have dashed boundaries overlaid to help visualize how the design points shy away from areas already covered.

Figure 11. Biased spread design with a very small amount of bias.

Figure 12. Biased spread design with a small-to-moderate level of bias.

Figure 13. Unbiased spread with 30 random starting points (asterisks) and 30 design points (solid boxes). The random selection is largely drawn in to dense areas in the center of the map. With these marked as previously selected, the design points augment that set so that the final set of 60 points has the best possible space-filling properties. The 30 starting points have dashed boundaries overlaid to help visualize how the design points shy away from areas already covered.

The dependency on cluster analysis that we have seen is so enormous and the inertia to change so great that we make three points on the topic of cluster analysis versus spread design.

1. Spread design has the useful feature that the entire set of molecules is prioritized and gives an optimal set, however far you drill down the prioritized list. Whether 5 molecules are needed or 5000, the best design is obtained by drawing off the top of the prioritized list. There is no need to control the number by iterating through parameter changes, as often happens with cluster analysis. The spread design gives a ranking that is in some sense the best set of molecules, regardless of the number of molecules to be selected.

2. Cluster analysis has no direct way to accommodate the information from molecules already tested in an assay, what we call the list of previously selected molecules. With spread design, maximum use is made of all the available data in an augmentation sense.
3. Cluster analysis is inherently unable to incorporate a bias to give a desirability gradient. The bias introduced in spread design is intuitive, easy to implement, and effective. Often in order to achieve this effect, scientists have turned to leader analysis (e.g., Spath et al.26), which effectively sorts the data by desirability and drills down the list, skipping molecules that are very similar to a molecule higher in the list (the “previously selected” of item 2). Leader analysis is not a true simultaneous optimization, in that there is no direct management of the trade-off between design and desirability. In contrast, biased spread design chooses each additional design point in a way that optimizes the prioritization as a whole.

We note that the spread algorithm is a greedy approximation; thus it is not strictly the most optimal possible. However, a great deal of experience has shown that it is quite adequate for the context of interest.

■ PUTTING IT ALL TOGETHER

We have evaluated the approach in a number of parallel experiments in real time, in different stages of lead generation. These experiments were done in live projects at our company, where the immediate goal was to advance a scaffold to the next milestone; e.g., if a project was in a pre hit-to-lead state, the next milestone would be hit declaration, and if this is achieved, the project would then transition to a hit-to-lead stage, where the next milestone would then be a lead declaration. A computational chemist and/or lead project chemist selected molecules for testing, using the standard approaches (taken as a baseline for this experiment) and at the same time the MST was used to optimize potency, selectivity, metabolism, and a number of other properties. The same set of properties was the focus for both sides of the experiment. To show results of these studies in all these aspects for each project would consume too much space, and for proprietary reasons we are not able to reveal the targets. We will describe some of the results in terms of the potency hit rates, but similar improvements were seen in metabolism and other aspects for each of these projects. For a CNS target related to cognition in an AA stage (screening follow-up), the standard approach selected a set of molecules that revealed a 5.3% hit rate and the MST selected molecules with a hit rate of 26.3%. If only different scaffolds are counted in the hit rate computation, thus requiring novelty of hits, the hit rates were 5.7% and 32.3%, respectively. This latter comparison shows that novelty of structure was improved in the MST approach, as expected from the design component built into the system. In a cancer project in an AA stage, the standard approach gave a set of molecules with an active rate of 3.1%, where the MST approach gave a hit rate of 24.7%. There was an interesting situation where the purification process created a number of false positives for many series in the sets tested. The molecules selected using the MST were from a series that did not suffer from this effect and in fact were the only series that showed a true SAR. Viewing this effect as an additional source of noise, one may interpret this success as the ability of the predictive models to find the “signal” in the noise. Thus, in spite of training models on data that later were shown to have numerous false positives, the true relationship was uncovered and exploited to select molecules that showed confirmed activity. In another cancer project in a similar stage, the MST doubled the hit rate of active molecules. In a CNS project related to pain, a hit was defined in two dimensions, as potent at the target of interest and not potent at a specific antitarget. The traditional method gave a hit rate of 2.4%, and the MST gave a hit rate of 8.1%.

The power of this system has been shown in H2L stages as well and in prospective design. In one recent H2L project, a molecule was chosen to represent a scaffold of interest (call this the “seed”). The potency, selectivity, and metabolism were known for this molecule, and the team had a clear understanding of what was needed to move this scaffold to the next milestone, with the need to improve all three of these aspects. This seed was divided into four “domains”, and substituent replacements were hypothesized at each area of the
molecule. For example, in one area of the molecule, 45 different substituent replacements were hypothesized, each with a view to improving metabolism or selectivity, etc. A full virtual library was generated with \(45 \times 13 \times 18 \times 38 = 400,140\) virtual molecules. None of these molecules existed in our collection. There were several potency assays, both enzymatic and cellular, and models were trained on available data from these. There were three antitargets of interest, and the team was also looking at solubility and metabolism as well as a few other properties. An MST template was created, and a mere 30 compounds were selected from these 400K. These were synthesized and tested in the assays. The potency found was 2 orders of magnitude greater than the seed in both enzymatic and cellular assays and over a hundred-fold more selective, with improved metabolism across four different species. Yet these molecules were only 60% similar to the seed. From this set of results the team quickly declared a lead molecule and moved on to the next stage of development. This experiment has been done in a number of prospective design contexts. While not always a dramatic success, when the results were discouraging, the traditional methods also had discouraging results.

In the Supporting Information, available online, we illustrate the method with a concrete example using a retrospective set of 912 marketed drugs (proprietary concerns require using a set like this rather than live project data). The set of marketed drugs is dated (prior to 2001) and not comprehensive. For example, it includes morphine analogs but not morphine itself. It also includes (intentionally) drugs taken off the market (e.g., astemizole). The point of the example is merely to illustrate the mechanics of the MST system. In our first example, we generate a scenario in which we seek compounds that enter the brain and are free of two liabilities: hERG blockade and phospholipidosis (PL). In the second example, we create a scenario in which we are looking for biased \(\mu\) opioid receptor agonists, in the same set of 912 marketed drugs. Throughout both examples the reader will see useful diagnostics such as scatter plots and spread diagnostic plots and discussion of pitfalls and other issues related to implementation of a comprehensive system such as this. In particular, the examples contain important details about how to manage a balance between desirability and diversity through the use of what we call the “Dweight.” We do reproduce herein one of the figures found in that document; Figure 14 gives a schematic of the process from start to finish (same as Figure S21 in the Supporting Information). This, coupled with Figure 1, gives a high level “thumbnail” view of the entire system for prioritizing compounds.

**COMMERICAL SOFTWARE**

There are numerous commercial software packages for doing visualization, QSAR and for other individual pieces of an MPO system. There seem to be very few packages that put it all together into one integrated system. One package that puts much of this all together is produced by Optibrium; the package is called StarDrop. This package has several strengths; it uses utility functions (therein called “scoring functions”) aggregated to an overall desirability. It has predictive modeling capabilities, and it has extremely good visualization features. StarDrop also has a diversity component, which uses a genetic algorithm to optimize the trade-off between desirability of the molecules and diversity of the selection. It also accommodates a previously explored set in order to allow the current selection to augment what has already been learned, a capability that we find especially important. The software also offers the user control over the predictive modeling in the same way that the MST does, by allowing the user to supply scripts that call their own modeling software and inputs. All this power and flexibility do come at a cost. A single license for the packages needed to perform these tasks came at a rather hefty price tag (in our view), plus additional costs for optional add-ons. Aside from StarDrop and our in-house MST system, we are not aware of a software package that uses a space-filling design in combination with predictive modeling and an MPO with utility functions to prioritize compounds.

**CONCLUSION**

We have shown a method, and an implementation system, for prioritizing molecules that does not rely on fixed functions or end point sets. The method is simple to use in that a few interpretable parameters will define what a team has determined to be desirable for each end point. The method allows compartmental thinking about each parameter by itself, constructing meaningful and interpretable desirability functions for each parameter, and then through importance weighting, an overall desirability score. We have presented a modern design method that augments what a team has already learned and efficiently prioritizes the entire set of input molecules so that if K molecules are desired, the top K molecules in that prioritized list will be the most information rich and most desirable set of K molecules for the next round of assay testing. Finally, we described parallel experiments demonstrating the effectiveness of the system. We believe that static MPO functions were a good initial step for our industry but that a flexible system such as this is the next logical step in discovery chemistry.

**ASSOCIATED CONTENT**

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jmedchem.5b01338.

Additional figures, technical details, and two worked examples (PDF)

SMILES, predicted values, and properties for 912 marketed drugs (CSV)

**AUTHOR INFORMATION**

Corresponding Author

*Phone: +1 (317) 771-1035. E-mail: Cummins_DJ@lilly.com, DavidJesseCummins@gmail.com.

Notes

The authors declare no competing financial interest.

Biographies

David J. Cummins is a Senior Research Advisor at Eli Lilly Headquarters, Indianapolis, Indiana, U.S. He joined Eli Lilly in 1997, the same year he earned his Ph.D. in Mathematical Statistics from North Carolina State University. Before that he spent 5 years at GS
c (Glaxo-Wellcome at the time) in Research Triangle Park, North Carolina, as a Statistical Research Associate in Drug Discovery Informatics. He is author and co-inventor on over 20 patents, book chapters, and scientific publications.

Michael A. Bell is a Research Scientist at Eli Lilly Headquarters, Indianapolis, Indiana, U.S. In 2001 he earned an M.S. in Computer Science from University of North Carolina, Chapel Hill. He has
supported the Statistical community at Eli Lilly as a Computer Scientist since 2004.

**ACKNOWLEDGMENTS**

We thank Yue Wang Webster and Patty Bragger for early involvement in the graphical user interface. We thank Ian Watson for implementation of biased spread design using molecular fingerprints and many other useful tools that have been integrated in the software and also for many productive conversations. We thank Richard E. Higgs for implementation of biased spread design using descriptors and for the innovation in introducing a bias (penalty) in the spread design algorithm.

**ABBREVIATIONS USED**

MOOP, multiatribute optimization; MPO, multiparameter optimization; Dscore, desirability score; BSD, biased spread; MST, medium throughput screening; AA, actives optimization; Dscore, desirability score; BSD, biased spread design using descriptors and for the innovation in conversations. We thank Richard E. Higgs for implementation in the graphical user interface. We thank Ian

**REFERENCES**


