Neighbor-specific BGP: An algebraic exploration

Alexander J. T. Gurney
Computer Laboratory
University of Cambridge
Email: Alexander.Gurney@cl.cam.ac.uk

Timothy G. Griffin
Computer Laboratory
University of Cambridge
Email: Timothy.Griffin@cl.cam.ac.uk

***Abstract*—There are several situations in which it would be advantageous to allow route preferences to be dependent on which neighbor is to receive the route. This idea could be realised in many possible ways and could interact differently with other elements of route choice, such as filtering: not all of these will have the property that a unique routing solution can always be found. We develop an algebraic model of route selection to aid in the analysis of neighbor-specific preferences in multipath routing. Using this model, we are able to identify a set of such routing schemes in which convergence is guaranteed.**

I. Background

A BGP speaker with a route to a given destination can announce that route to its neighbors. Depending on the configuration, a particular route could be permitted to be sent, or could be filtered out. Each neighbor can be associated with a different filter, but there is no possibility for an alternative route to be sent instead: either the installed route is sent, or no route at all.

Neighbor-specific BGP (hereafter NS-BGP) is an alteration to the standard BGP model, allowing outbound route selection to yield a different outcome for each neighbor [23]. This only makes sense if a router can have multiple installed routes to the same destination, and if there is some supported forwarding mechanism by which the appropriate route can be selected for incoming traffic.

In this paper, we explore some route selection models related to those of NS-BGP, in order to establish which of these schemes can be considered 'safe', in the sense of protocol convergence to an expected optimal state. In so doing, we are able to isolate 'neighbor specificity' as an aspect of protocol design, which can be associated not just with BGP but with other protocols as well. We illustrate NS-BGP ideas with reference to some routing outcomes which are either impossible to acheive without neighbor specificity, or for which the consideration of neighbor specificity leads to a clearer understanding of the routing policy. For correctness, we demonstrate that convergence to a unique locally-optimal routing solution can be guaranteed by the same algebraic condition as in the familiar single-path case. We show that if correctness has been established for the 'global' preferences on which everybody agrees, then any *extension* of these preferences is safe for neighbor-specific use.

The result applies to any situation in which multipath preferences are present, and where they can be refined on a per-adjacency basis. This includes the NS-BGP model as a special case, where multipath is only used within an AS, and external connections use some neighbor-specific single-path preference scheme. A merit of NS-BGPis that an AS can adopt neighbor-specificity without any of those neighbors needing to know, since the new behaviour is confined to the AS interior. Our result would also prove convergence for protocols that do not have this restriction.

There is a long history of related approaches to interdomain routing, particularly with respect to the possibility of offering multiple paths. Many of the proposals take the form of completely new routing architectures [8], [25], [26]. Others represent additions to the standard BGP control plane, in which new components allow additional capabilities for multipath [20], [24]. Meanwhile, contemporary BGP use allows a limited form of multipath routing already, and in the future `ADD_PATHS` may permit a more general capability [18], [21].

On the theoretical side, the possibility of finding multiple paths in a graph has also been extensively studied [2], [9], [10]. Crucially, correctness conditions for this kind of pathfinding are 'inherited' from correctness conditions for conventional single-path search. The same mathematical framework is used in either case. For problems with variable preference (as opposed to variations on the shortest-path theme such as the k-shortest paths problem), combinatorial games such as the stable paths problem [14] allow the consideration of 'optimal' paths where the network participants have different ideas about what 'optimal' really means.

However, the general idea of neighbor-specificity has not been studied. This pattern has been noted as a possible BGP extension [23], but has not been treated as a problem in itself. The proof of unique convergence for NS-BGP is not only specific to BGP, but to a particular family of configurations (the Gao-Rexford conditions [7]). In this paper, we will provide a much more general proof. Neighbor-specific preferences are not limited to BGP, but may arise in many other pathfinding scenarios. Indeed, we argue that some current practices on today's Internet, properly regarded, are *already* examples of neighbor-specificity in action. Identification of this pattern should lead to a clearer understanding of what problem is actually being solved, and therefore of which alternative solutions or extensions might be possible.

We would like to emphasize that in this paper, we are not attempting to make a case for the practical benefits of NS-BGP or any other neighbor-specific routing protocol. Any such design must be assessed in terms of routing stability, ease of

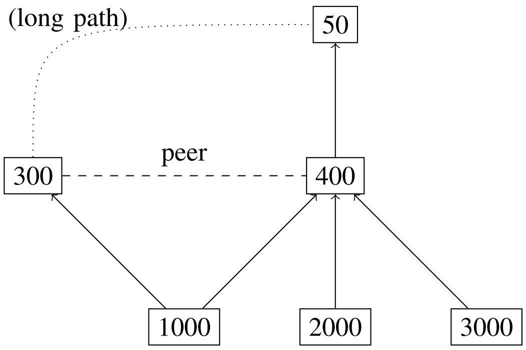

Fig. 1. Neighbor-specificity by neighbor class

management, implementation performance, and other factors that we do not consider here. We do demonstrate that the correctness issue is 'no worse' for neighbor-specific preferences as compared to conventional single-path preferences. We also aim to show some of the variety of neighbor-specific preference models that might be adopted, and how these relate to current practice.

II. Examples of neighbor-specificity

The current interdomain routing system, based on BGP, allows a great deal of policy expressivity. Routing participants are able to exercise considerable control over how their routes are advertised, and how the recipients should treat them. Nonetheless, there are some situations where the present model is limited. Neighbor-specific BGP is one proposed extension that would enable new policies, based on the capability of an autonomous system to advertise different routes to different neighbors, for the same prefix.

A. Class of neighbor

A simplified model of interdomain policy divides the neighbors of an AS into customers, peers, and providers. When a prefix is available through several neighbors, their classes are important in determining which route is to be preferred; customer routes typically being considered the best, followed by peer routes, and finally provider routes. In addition, the onward propagation of routes can be controlled through communities. These include well-known communities in which the further advertisement of a route is prohibited. Many ASes also implement communities to limit route advertisement to specific neighbors or classes of neighbor, such as "all customers" or "European peers".

Consider the network of Figure 1. AS 400 has (at least) two routes to AS 1000: a direct route, and another via AS 300. The direct route would typically be preferred, since it comes from an immediate customer, and furthermore is shorter. If this route is tagged with an appropriate community, then its onward advertisement might be restricted only to the other customers of AS 400. The adoption of this route prevents any other route from being used for other neighbors. In particular, they will not be able to reach AS 1000 through AS 400 and then AS 300: they will be forced to use some other path, which could have worse performance characteristics.

In a neighbor-specific implementation, AS 400 could adopt and offer two different routes to AS 1000. For customers, there is the direct route; for all others, there is the peering link to AS 300. To support this possibility, AS 400 must deploy multipath routing along with multipath forwarding. The *routing* must allow the propagation of at least two classes of route: those available to customers only, and those which are universal. Best-path selection should be done separately for these two classes. On external adjacencies, however, at most one route should be exported. For customer adjacencies, this could be either the best customer-only route, or the best universal route; for all others, it will be the best universal route. Filtering could also take place. Since a BGP route advertisement is a promise to carry traffic, AS 400 must forward incoming traffic to the appropriate egress point. The selection of egress point can be made easily, but some form of tunnelling will be necessary to carry out the transport correctly.

B. Internet exchanges

Many Internet exchange points (IXes) offer participants the option to peer via a *route server*. Conceptually, such a server carries out the routing function for all participants, based on their declared policy [6], [11], [12], [17]. The administrative burden of peering with many parties is then reduced somewhat, as is the technical overhead of supporting many adjacencies, or many VLANs. However, this technology is not only for efficiency's sake, but is also an implementation of neighbor-specific routing.

When IX participants operate a multilateral peering agreement, the route server implementation is simple. The server only has to collate all incoming routes, carry out the BGP best-path selection process, and then advertise the resulting routes to all participants.

As policies become even slightly more complex, this simple model breaks down. The route server, as the center of a star topology, is only able to select a single best route for each destination: alternative routes are suppressed. Suppose that AS 100 has the best route to some destination, but does not wish AS 200 to be able to use it. If the route server selects AS 100's route, and filters out the announcement to AS 200, then the unfortunate AS 200 is left with no route to that destination. Without the route server's intervention, AS 200 might be able to learn an alternative route via another IX participant.

One way to solve this problem is to set up a route server with multiple RIBs, each corresponding to some IX participant, with appropriate import and export rules among them [6]. This amounts to simulating the original partial mesh within the route server. Path selection and filtering are carried out on a per-RIB basis. So if AS 100 wants its routes to be hidden from AS 200, they will not reach the internal AS 200 table—but some other routes might be present instead, even if they would normally be knocked out by BGP best-path selection.

In fact, this situation is another example of neighbor-specific preferences. Imagine the IX route server as a transit AS (after all, it has an AS number and speaks BGP). The policy it operates is, internally, to make no decisions about routes when they come from different neighbors. Only on export are distinctions made: for each external adjacency, a more specific preference is imposed, in which certain routes are removed from consideration, and a conventional BGP best-path process applies to the remainder.

It should now be obvious that the IX is really no different from any other AS in terms of its possible routing architecture. The only point of divergence is that the flavor of BGP being used is a neighbor-specific one, and that the policies in question are being determined by participants rather than by the IX itself. Through a better understanding of neighbor-specific preferences, we can explore which of these designs might be useful in future IX operations, or which could be unsafe.

III. A MODEL OF NEIGHBOR-SPECIFIC PREFERENCES

We now develop a model of route preference that incorporates neighbor-specific choice. Appendix A contains definitions of the standard algebraic objects and terminology used in this section.

The idea behind neighbor-specificity is that *within* a network, routes are selected according to some relatively lax criteria, whereas on *export*, more stringent rules are applied—and these vary according to the adjacency. A modest example along these lines would be the existence of neighbor-specific filters, as are used today: on a particular outgoing connection, some route announcements are forbidden which would otherwise be acceptable. More radically, one can imagine several ways in which route preference might vary in a neighbor-specific manner.

Consider an abstract model of a network as a graph, in which the nodes are routers and the arcs are connections between them. Routers select paths according to some *order* on the set of all paths. Conventionally, this order is taken to be *total*, so that for any two distinct paths, there is a well-defined best one. In such an environment, equal-cost multipath routing is the same as ordinary single-path routing; if multiple paths are desired, then they will be of variable quality. More liberal preference orders admit the possibility that two routes might be equivalent in preference, so they could be equally 'best'. AS path length comparison is an example: the length of the path matters, but usually not the contents. A related notion is that routes might be incomparable, so that a determination of the best path cannot be made. In BGP, incomparability occurs in use of the MED attribute. MED values from different neighbors cannot be compared: the comparison is only meaningful with MEDs coming from the same source. Our neighbor-specific model will make heavy use of incomparability: we have a 'global' order in which many routes will be incomparable, meaning that there is no overall agreement as to their relative preference; alongside this, several 'extended' orders have the freedom to choose different, stricter, preferences for these routes.

In order to admit as many models as possible, we will assume that the underlying route preference order is a *preorder* (a reflexive and transitive relation). So we allow routes to be equivalent or incomparable, as well as strictly ordered. The various ways in which this order might be extended must all follow the definition below.

Definition 1: Let $\preceq$ and $\preceq_R$ be preorders on a set S. Say that $\preceq_R$ *extends* $\preceq$ if $x \preceq y \implies x \preceq_R y$ for all x and y in S.

An extension is *linear* if it is a total order.

It can be seen that if one order extends another, then it must agree with the original order in some respects, but is allowed to make some further distinctions which were not originally present. Two routes that were incomparable according to $\preceq$ are allowed to be strictly ordered one way or the other in the extended order. However, if two routes are equivalent in $\preceq$ then they must still be considered equivalent by $\preceq_R$: the extended order cannot break these ties. Finally, if two routes x and y are strictly ordered ($x \prec y$), then either $x \prec_R y$ or $x \sim_R y$ in the extended order. This means that the extension must also prefer x to y, or at least consider them equivalent: it cannot prefer y to x.

For a given preorder $\preceq$, let $R(\preceq)$ denote the set of all of its extensions. It can be shown that

$$x \preceq y \iff \forall \preceq_R \in R(\preceq) : x \preceq_R y,$$

so that $\preceq$ is equal to the relational intersection of all of its extensions. This is a well-known theorem when $\preceq$ is a partial order [5], and only slightly less well-known for preorders [4]. If we are provided with some set of orders, then their intersection is the order that represents the 'common ground' between them: the subset of preferences on which all of the given orders agree. This further motivates the use of preorders: the intersection of all neighbor-specific orders is unlikely to be total.

Given any preorder $\preceq$ on a set S, one can define an operator $\min_\preceq$ over subsets of S by

$$\min{}_\preceq(A) = \{x \in A \mid \forall y \in A : \neg(y \prec x)\}\,.$$

Thus $\min_\preceq(A)$ returns the set of $\preceq$-minimal elements of A: those which are not dominated by any other element of A. We can use such an operator to model equal-cost multipath. Given a set of routes, obtained from neighbors, a router applies the $\min$ operation to find the subset of best routes. If there is a total order, then the resulting set will contain only a single path. Otherwise, there will be several elements: and they will all be equivalent or incomparable to one another.

For neighbor-specificity, we can select paths internally by using $\min_\preceq$ for some 'standard' preorder $\preceq$. On external links, however, the order is extended—if $\preceq_R$ is the extension of $\preceq$ to be used on a particular link, then the required operator is $\min_{\preceq_R}$. How can this be incorporated into the model? To answer this question, note that the extended order applies to the link between two routers. Each router will be choosing routes according to the conventional best-path process ($\preceq$); the scope of the extended order is only relevant to the link. The router that advertises a set of routes is certainly not being constrained by any $\preceq_R$ in its own selection procedure—after all, a different extended order could be in use for a different adjacency. The receiving router is constrained by $\preceq_R$, but indirectly: the extended order does not apply to *all* received routes, but only to those received along the link in question. So we view the extended order as being attached to the link, in the same manner as a filter.

This completes the basic picture of neighbor-specific preference. The preference model on routes is that the router-level graph is associated with a global preorder together with a family of extensions of that preorder. Each link in the graph is associated with one of the extensions. Path selection operates as follows. Suppose a is a node with in-neighbors b_1 through b_k. Refer to the extended orders on each link (a, b_i) as $\preceq_i$. Then if B_1 through B_k are the best paths chosen by nodes b_1 through b_k, let

$$A = \min\left(\bigcup_i \min_{\preceq_i} \{(a, b_i)p \mid p \in B_i\}\right).$$

These are the best routes for a. In other words, the sets of best routes available through each b_i are reduced according to $\preceq_i$; all of the resulting sets are the joined to give the candidate set for best-path selection according to the global $\preceq$.

Note that this algebraic model supports a scheme where multipath routing only happens within an AS. If we enforce that all external links use refinements that are total orders, then we are describing a situation much like that envisaged for ADD_PATHS: within an AS, many routes can be propagated (the order involved is not total), whereas on external adjacencies we use traditional eBGP.

IV. SOME ORDER EXTENSIONS AND THEIR USES

Let us see a few examples of extensions of orders. These demonstrate several ways in which neighbor-specific routing could operate. Not all of these, however, will necessarily be safe.

a) Example 1: Discrete order: Consider what happens if $\preceq$ is the discrete order on S, so that $x \sim x$ for all x in S, but elements are otherwise unrelated. This order has a great many possible extensions: in particular, any linear order on S is a extension. That includes linear orders which are the reverse of one another.

In routing, this corresponds to a situation where no best-path selection *at all* is carried out for internal sessions. Instead, every possible path is carried along. Only at the borders, on export, is a choice made.

b) Example 2: Lexicographic order: Let $(S \times T, \preceq)$ be the direct product of $(S, \leq_S)$ and $(T, \leq_T)$, where both are linear orders. Then

$$(s_1, t_1) \preceq (s_2, t_2) \iff (s_1 \leq_S s_2) \wedge (t_1 \leq_T t_2).$$

Two of the possible extensions are the lexicographic orders on $S \times T$:

$$(s_1, t_1) \preceq_1 (s_2, t_2) \iff (s_1 <_S s_2) \vee (s_1 = s_2 \wedge t_1 \leq_T t_2)$$

and

$$(s_1, t_1) \preceq_1 (s_2, t_2) \iff (t_1 <_T t_2) \vee (t_1 = t_2 \wedge s_1 \leq_S s_2).$$

Call these $S \vec{\times} T$ and $T \vec{\times} S$ respectively [16].

This example can be extended to apply to a larger number of components. If $(S_i, \preceq_i)$ are preorders, for i in $I = \{1, 2, \ldots, k\}$, then their direct product consists of all vectors

$$(s_1, s_2, \ldots, s_k) \in S_1 \times S_2 \times \cdots \times S_k$$

with

$$\vec{s} \preceq \vec{t} \iff \forall i \in I : \vec{s}_i \preceq_i \vec{t}_i.$$

We can build an extension of this order by the following procedure. Let J be a subset of I, and let J' be $I \setminus J$. Choose some order for the elements of J. Construct two algebras:

- The lexicographic product of all S_j, for j in J, taken in order.
- The direct product of all $S_{j'}$ for j' in J' (in any order).

The direct product of these two algebras is an extension of the original direct product.

In other words, if we start with a collection of route attributes S_i, an extension is to take a lexicographic order on *some* of them, and retain the others. So the extension is to make a choice about which attributes are more important than which others. In the simple case, with just S and T, it could be that some neighbors would prefer to see routes that are better according to S, using T as a tiebreaker, whereas others want routes that are better according to T, using S as a tiebreaker. Within the network, using the direct product, both of these possibilities can be supported at the border, since the only routes that are removed internally are those which are strictly dominated in both S and T components.

c) Example 3: Linear combinations: Consider the direct product of $(\mathbb{N}, \leq)$ with itself. For parameters α and β in $\mathbb{R}+$, we can define another order by

$$(w, x) \leq_{\alpha,\beta} (y, z) \iff \alpha w + \beta x \leq \alpha y + \beta z.$$

This is a refinement of the direct product, for if $w \leq y$ and $x \leq z$, then $\alpha w \leq \alpha y$ and $\beta x \leq \beta z$, and the conclusion follows.

This example extends to arbitrary linear combinations of numeric attributes. If we have the n-fold product $(\mathbb{R}^+, \leq)^n$ together with a family of vectors $\vec{\alpha}$, with each component positive, then we can define orders by

$$\vec{v} \preceq_{\vec{\alpha}} \vec{w} \iff \vec{\alpha} \cdot \vec{v} \leq \vec{\alpha} \cdot \vec{w}$$

where $\cdot$ is the scalar product of the two vectors. For any $\vec{\alpha}$, this is a refinement of the product order.

These two examples correspond to a routing model based on Pareto optimality. Within the network, a route is preferred if and only if it is better according to all attributes: any resulting set of best routes will form a Pareto frontier. On export links, the linear combination allows various possibilities for trading these attributes off against one another, thereby selecting one or more points from the frontier.

Route choice based on a linear combination of attribute values has been considered by many authors, and is included in the routing protocol EIGRP [3]. In particular, this has been suggested for interdomain routing, including in the context of neighbor-specificity [22], [23]

d) Example 4: Direct sum: Let $(S, \preceq_S)$ and $(T, \preceq_T)$ be two preorders, and consider their disjoint union

$$(S \uplus T, \preceq)$$

where

$$\begin{aligned} s_1 \preceq s_2 &\iff s_1 \preceq_S s_2 \\ t_1 \preceq t_2 &\iff t_1 \preceq_T t_2 \end{aligned}$$

for all s_1, s_2 in S and t_1, t_2 in T. One extension is $\preceq_1$, where $s \prec_1 t$ for all s and t, and its companion $\preceq_2$, where $t \prec_1 s$ for all s and t.

This corresponds to a routing model with two classes of route, 'S routes' and 'T routes'. They are incomparable internally, so that any set of best routes will include some S routes and some T routes. The extensions amount to preferring one class over the other. This could be used for multi-topology routing, where external neighbors are offered best routes from one topology or the other.

The example of Section II-A is related. In that network, the different kinds of route were 'customer-only' and 'universal'. Within the AS, these are incomparable, but on external links the two extensions are

- On customer links, do not distinguish between the classes. The incomparability has become equivalence (for routes which are otherwise the same).
- On other links, customer-only routes are less preferred than universal routes: indeed, they are forbidden. This is the same as having them being worse than the 'null' route.

Other classes can be encoded similarly; for example, we could imagine an adjacency for which customer routes should be considered as worse than peer routes. This would be implemented by an appropriate extension, placing one class above the other.

V. An algebraic model of NS-BGP

Before proving correctness properties, we need to have a proper algebraic model of routes and route selection. Section III established neighbor-specific preference; this section broadens that to a model which incorporates properties about the *extension* of routes, and its interaction with preference.

Our foundation is *semigroup transforms* [15], [16], which are themselves based on *algebras of monoid endomorphisms* [10]. This provides an algebraic model of choice and extension, where choice is implemented by a binary operator and extension by a family of functions. More specifically, a semigroup transform $(S, \oplus, F)$ consists of a set S, a binary operator $\oplus$ on S, and a set of functions F from S to itself. The operator is required to be associative. For problems on graphs, S can be a set of paths or path weights, $x \oplus y$ yields the 'better' path out of x and y, and $f(x)$ represents the extension of path x by a function f associated with some link.

In the multipath case, we can take the elements of S to be sets of paths. Specifically, they will be sets A such that $A = \min_\preceq(A)$: the paths selected by each node *must* be of equivalent (or incomparable) costs. An $\oplus$ operator is given by

$$A \oplus B = \min_\preceq(A \cup B)$$

selecting the best paths out of the combination of A and B. For the functions, we suppose that we already have some set F of functions on paths; these can be extended to sets of paths in the obvious way:

$$f(A) = \min_\preceq \{f(x) \mid x \in A\}\,.$$

In summary, if we are given a triple $(S, \preceq, F)$, then we can define an algebra $(M(S), \oplus, F')$ for multipath routing, where

- $M(S)$ is the set of all subsets A of S for which $A = \min(A)$,
- $A \oplus B = \min(A \cup B)$, and
- F' consists of functions $f'(A) = \min(f(A))$ for each f in F.

Generalized algorithms can use this structure to find multiple best paths, provided that the algebra has appropriate properties. We will call the result of this construction a *multipath algebra* (MPA).

The construction applies equally well when S is not a set of paths, but a set of path weights (and the other data are adapted appropriately). This allows us to draw conclusions which are not limited to a specific graph by their dependence on its path set, but which apply to all graphs whose weights are taken from S. It should be clear that, if $\preceq_w$ is the preorder on paths defined by a weight function w with values in $(S, \preceq)$, so

$$p \preceq_w q \iff w(p) \preceq w(q),$$

then

$$\min_{\preceq_w}(P) = \left\{ p \in P \;\middle|\; w(p) \in \min_{\preceq}(w(P)) \right\}$$

where $w(P) = \{w(p) \mid p \in P\}$. So it does no harm to consider minimization with respect to path weights as opposed to the paths themselves: the same results can be achieved.

Some useful properties of min include:

1) It is idempotent: $\min(\min(A)) = \min(A)$ for all subsets A of S.
2) It is decomposable: $\min(A \cup B) = \min(\min(A) \cup \min(B))$ for all subsets A and B of S.

These two facts mean that there is considerable flexibility about where and how min might be applied in an algorithm. Because of idempotence, it is always safe to apply min several times; equally, multiple applications can be replaced by a single one in order to achieve the same outcome more efficiently.

Similar observations can be made about the decomposability of $\min$. It is always safe to apply $\min$ to a subset of the given set: we never have to worry about losing something that will be needed later. Conversely, we can get away with using just one $\min$ at the top level, instead of repeatedly applying it to subsets.

For neighbor-specificity, we can extend the MPA. Given a (single-path) order transform $(S, \preceq, F)$, the appropriate algebra is simply

$$NS(S, \preceq, F) = (M(S), \oplus, G)$$

where G consists of functions

$$g_{(f, \preceq_R)}(A) = \min{}_{\preceq_R} (\{f(x) \mid x \in A\})$$

for all f in F, and all order extensions $\preceq_R$ of $\preceq$. Call this the *neighbor-specific multipath algebra* (NSMPA) associated with $(S, \preceq, F)$. When $NS(S)$ is used to label a graph, each arc will now be associated with a function $g_{(f, \preceq_R)}$, rather than just f. This means that a set of routes is transformed by applying f to each one, and then minimizing with respect to the extended order $\preceq_R$. Different arcs may use different f functions, or have different orders, or both.

VI. Correctness proof

One possible desirable property for routing algebras is that for any appropriately labelled graph, there is a path assignment that is *globally optimal* [2]. This means that for every source node and every destination node, the assigned path is the best out of all possible such paths. For a semigroup transform $(S, \oplus, F)$, this means that the weight of each assigned path is equal to the *sum*, using $\oplus$, of the weights of all paths between the same source and destination.

The key algebraic property which leads to this kind of optimal solution existing is the *distributivity* of each function f over $\oplus$. This property holds when

$$f(x \oplus y) = f(x) \oplus f(y) \tag{1}$$

for all x and y in S, and all f in F.

In the case of $(M(S), \oplus, F')$, this amounts to the verification of whether

$$\min{}_{\preceq}(f(A)) = \min{}_{\preceq}(f(\min{}_{\preceq}(A)))$$

for all A (with $A = \min_{\preceq}(A)$) and all f. Whether this is true or not will depend on the nature of the $\preceq$ order and its interaction with the functions.

Theorem 1: Let $(S, \preceq)$ be a preorder and F a set of functions over S. Then the semigroup transform $(M(S), \oplus, F')$ has property 1 if, for all x and y in S, and all f in F,

$$x \preceq y \implies f(x) \preceq f(y)$$

Proof: The desired property is equivalent to

$$\uparrow(f(A)) = \uparrow(f(\min{}_{\preceq}(A)))$$

where

$$\uparrow(X) = \{y \in S \mid \exists x \in X : x \preceq y\}$$

We can decompose A as the union of $\min_{\preceq}(A)$ and the other elements. So we are checking whether

$$\uparrow(f(\min{}_{\preceq}(A))) \cup \uparrow(f(A \setminus \min{}_{\preceq}(A))) = \uparrow(f(\min{}_{\preceq}(A)).$$

This is true when, if $f(a) \prec x$ for some x in S and a in $A \setminus \min_{\preceq}(A)$, then there is some a' in $\min_{\prec}(A)$ with $f(a') \prec x$. But if a is in A but is non-minimal, then by definition there is at least one a' in $\min_{\preceq}(A)$ with $a' \prec a$. From the assumption on f, we then have

$$f(a') \preceq f(a) \prec x$$

so a' fulfils the desired property. Therefore, the algebra is distributive. ■

That is, we have convergence to a global optimum for equal-cost multipath whenever the underlying single-path algebra is monotonic.

Another kind of optimum which might exist—and which is closer to the operational model of BGP—is a 'local' one. This is a path assignment in which the chosen paths are not necessarily the best; but they are, at least, the best possible if a node's choice of path must be consistent with its neighbors [14]. That is, if p is a path in a locally optimal assignment, then there can be no path (for the same source and destination) which is simultaneously better than p, *and* an extension of a neighbor's path in the same assignment.

In the case of multiple path algebras, it has been shown that convergence to such an optimum is guaranteed if $(S, \preceq)$ and F satisfy

$$\forall x \in S, f \in F : x \prec f(x) \vee x = f(x) = \top \tag{2}$$

where $\top$ denotes the maximal element of S, if any [15].

The main correctness result we will need is the following theorem. We are interested in local rather than global optimization, because we are investigating neighbor-specific preferences after the example of NS-BGP, in which global optimality is impossible.

Theorem 2: If $x \prec f(x)$ for all x in S and f in F, then $\mathrm{NS}(S)$ supports convergence to a local optimum.

Proof: This can be shown using the methods of Section 4.2 of [15].

For two path assignments A and B, let $A \mathbin{\Delta} B$ denote the set of paths which is in one of A and B but not the other. Let $\sigma(A)$ denote the path assignment obtained from A by simultaneous myopic best response: that is, each node finds its set of possible paths—all the paths with are extensions of the paths selected by neighbors according to A—and makes its selection from these. Any fixed point of σ is a locally optimal path assignment.

The original proof hinges on being able to find a pair of paths (p, q) with q in $A \mathbin{\Delta} B$, p in $\min_{\preceq}(\sigma(A) \mathbin{\Delta} \sigma(B))$, and $q \prec p$. If this is so, then we can show that σ is a *strict contraction* over path assignments, according to a certain metric d on the space of assignments:

$$d(\sigma(A), \sigma(B)) < d(A, B)$$

for all distinct path assignments A and B in M. The Banach fixed point theorem then yields a unique fixed point for σ.

Such a pair (p, q) is shown in [15] to always exist for MPAs where the underlying order transform $(S, \preceq, F)$ is strictly inflationary.

The original argument can be summarized as follows. Let p be any path in $\min_{\preceq}(\sigma(A) \ \Delta \ \sigma(B))$. Without loss of generality, assume that it is in $\sigma(A)$ but not $\sigma(B)$. Let q be its immediate prefix (p must be at least one arc long, or else the two assignments would agree). Certainly q is in A, or else p could not have been selected in $\sigma(A)$. If it were in B, then p could have been chosen in $\sigma(B)$ as well, as it was a candidate path—but it was not. The only reason for this to happen is that in $\sigma(B)$, some other path was chosen instead. This other path would have to be strictly better than p in order to exclude p from the set. But then p would not be minimal in $\sigma(A) \ \Delta \ \sigma(B)$, as we specificed, which is a contradiction. Hence q cannot be in B. So q is in $A \ \Delta \ B$, and by the strict inflationary property we have $q \prec p$ as required.

In the neighbor-specific algebra, there are some more twists to consider. It is still true that $q \prec p$ if we have the strict inflationary property. The problem is with the argument that the only reasons for $\sigma(B)$ not to include p are (1) the absence of q from B, or (2) the presence of some other, better route than p in $\sigma(B)$. With neighbor-specific preferences, there is a third possibility to be accounted for.

It could be that B includes q, but also includes some other path q', which is extended to p' in $\sigma(B)$, such that p and p' are incomparable in $\preceq$, but $p' \prec_R p$ in the extended order. So the presence of both q and q' in B means that $\sigma(B)$ has to choose between p and p' according to the extended order, and p' is chosen even though according to the original order they are incomparable and hence could both be chosen.

We can recover the proof by noting that q' cannot be in A: if it were, then p' would be in $\sigma(A)$ rather than p. So q' is in $A \ \Delta \ B$, and we do have $q' \prec p'$ by the strictly inflationary rule on $\preceq$. In this case, we can use (p', q') as the required witness pair rather than (p, q). ■

The condition in the theorem is the same requirement as for conventional single-path algebra. In moving to a world with not only multiple paths, but also neighbor-specific path selection, additional correctness conditions are not necessarily required. For algebras constructed with NS, proving correctness need not involve *any* reasoning about neighbor-specificity or the presence of multiple paths.

In the neighbor-specific setting, an ensemble of different preorders is used. The correctness of the entire system can be verified by checking a property of their relational intersection: the 'global' preorder representing the preferences on which every participant agrees. We suggest that this fact can be interpreted in two ways:

1) If path selection is being done with several different orders in a neighbor-specific way, then correctness can be confirmed by examining their intersection. Thus, the amount of mathematics we need to do is greatly decreased: only one order need be examined.
2) If we start by having verified correctness of some preorder, then we know that any extension of that preorder can be used locally without harming correctness. So once the global preferences have been established, any extension is safe to use, on any adjacency.

These complementary viewpoints suggest that specific designs for neighbor-specific path selection might be developed by (1) identifying which preorders might be useful locally, (2) finding their intersection and proving correctness, then (3) noting that any extension of that global preorder, including extensions not already found in step 1, could be deployed safely. Therefore, we would expect neighbor-specific preference schemes to admit wide variation in local practice, beyond what the original designers might anticipate.

VII. Application to BGP

We have developed an algebraic model inspired by the idea of neighbor-specific preferences in BGP. The original paper on NS-BGP proved correctness for a particular model of route selection, in which economic constraints are used to ensure convergence. This is a variation of the well-known Gao-Rexford model of routing, in which adjacencies are classified as customer-provider or peer-peer.

It is certainly possible to object that the Gao-Rexford conditions are not uniformly observed in current practice, although they do capture a useful pattern of inter-AS interaction. Many other *correct* interactions are possible, including those between entities which are not classifiable in the original scheme, such as IXes or CDNs [13].

In this section, we develop an algebraic description of a path selection scheme similar to that of NS-BGP, but with an explicit connection to the attributes of present-day BGP, and showing where the neighbor-specificity is able to be introduced. Crucially, we will be able to prove correctness of the new scheme.

The BGP route selection process consists of the successive examination of a series of attributes, each acting as a tie-breaker for the last. In algebraic terminology, this is a lexicographic product. According to standard BGP (that is, disregarding extensions due to communities and so on), the first step is to compare routes on the basis of their *local preference* (LOCAL_PREF). This numeric value is, in principle, completely arbitrary. In consequence, essentially no guarantees can be made about correctness. The model of Gao-Rexford is a response to this state of affairs, noting that a principled usage of local preference, conforming to certain economic sanity conditions, *can* be regarded as safe, even though unrestrained usage cannot.

The next important attribute to consider is the AS path; shorter paths are preferred. This attribute provides a form of route optimization, as opposed to the constraints represented by local preference. The overall procedure, then, even without taking any other attribute into account, is a kind of constrained optimization. A decent rule of thumb for constrained optimization problems is that it is often NP-complete to tell whether an optimal solution exists subject to the constraints, and indeed this is the case for BGP. But again, this is not the case if local preferences are decided upon according to sufficiently strong constraints.

Now, let us define some algebras for these two attributes. Of course, this will only provide us with an extremely simplified model of eBGP. Even so, it will serve as a reasonable basis for the introduction of neighbor-specific preferences.

Local preference values range from 0 to $2^{32} - 1$, and can be set arbitrarily. Larger values are better. Let L be the set $\{0, 1, \ldots, 2^{32} - 1\}$. We therefore have an order transform

$$\mathrm{LocalPref} = (L, \geq, K_L)$$

where K_L denotes the set of all constant functions whose range is L, so it contains functions of the form $\kappa(x) = \ell$, where ℓ is in L. While this, as expected, is not inflationary, there are subsets of K_L which are inflationary.

In a (considerably) simplified version of the AS path attribute, we will assume that AS numbers come from a set N and that the metric value is a simple list of these numbers. So we are ignoring aggregation, AS sets, confederations, compatibility between two- and four-byte AS numbers, and the total size of the attribute. Let ∞ be a special value not it in N: we will use this when a loop would otherwise be created. The order transform is

$$\mathrm{ASPaths} = (N^\star \cup \{\infty\}, \preceq, C_N)$$

where

- $N^\star$ denotes the set of sequences over N
- $\preceq$ orders sequences by length, so $p \prec q$ if p is shorter than q and $p \sim q$ if they are the same length; and $p \prec \infty$ for any p in $N^\star$
- C_N is the set of all functions c_n for n in N, where $c_n(p)$ returns np if n is not in P, and ∞ otherwise. Also, $c_n(\infty) = \infty$.

This algebra is strictly inflationary (despite the presence of ∞—this is accounted for in the definition of the property).

Pick some subset $\mathrm{LocalPref}_I$ of LocalPref which is inflationary, and form the lexicographic product of this with ASPaths. The resulting algebra is strictly inflationary.

Now, suppose that we have a series of other attributes S_1 through S_k, as in Example 2. We set S to be

$$(\mathrm{LocalPref}_I \vec{\times} \mathrm{ASPaths}) \times S_1 \times S_2 \times \cdots \times S_k.$$

Now we can form $\mathrm{NS}(S)$, the neighbor-specific preference algebra based on S. This allows certain arcs to choose in which order the extra attributes S_i will be considered, if at all. Meanwhile, internal arcs can be configured to use the standard order, so that the iBGP process spreads a multiplicity of routes around, to be winnowed down later.

We could choose $\mathrm{LocalPref}_I$ to be Sobrinho's algebraic model of Gao-Rexford. But there are many other possible choices which would also yield convergence. Equally, we admit many possible designs for choosing attributes S_i, even before allowing them to be compared in different ways. All of these are safe.

This demonstrates that an extraordinary variety of neighbor-specific preferences can be defined in a 'BGP-like' routing system, so long as some global rules are respected. Everyone has to agree on the economic model and on the preferability of short paths. Beyond that, arbitrary preferences can be established *without harming the possibility of convergence*.

VIII. Implementation outlook

We have now shown that for a very generous model of neighbor-specificity, it is possible to prove correctness in exactly the same way as for conventional path problems. For NS-BGP in particular, the consequence of the theorem above is that not only the Gao-Rexford conditions, but any strictly-inflationary conditions, will suffice for protocol convergence. This brings our attention to the remaining considerations: what kind of neighbor-specific preferences shall we adopt, and how should they be implemented?

The examples of Section II are a starting point at indicating some of the possible diversity of routing schemes that are supported in the neighbor-specific model. We believe that we are not in a position to prescribe any particular one of these as the one true way to do neighbor-specific routing. It is up to network operators and researchers to consider their policy needs and establish some preference model that works for them; the design space is large, but we hope that our examples will be suggestive of the kind of schemes that can be made to work.

Even once a neighbor-specific policy has been determined, significant questions remain about how the routing and forwarding are to be implemented. While one answer is the adoption of an entirely new routing protocol, we prefer to consider these questions from the perspective of current BGP practice, and the associated technological and other constraints.

A. Similarity to BGP

An ongoing NS-BGP implementation effort envisages a very general means of providing neighbor-specific routes [19]. This scheme uses MPLS tunnels to provide a cross-connect service across an AS, with neighbor-specific BGP deployed in order to provide neighbors with a choice of tunnels.

In its full generality, the implementation does not necessarily constrain the flexibility of route selection—although some restriction to safe configurations is envisaged [19], [23]. In this most extreme setting, we can view the NS-BGP process as operating at a layer below the conventional BGP computation: it is there to provide connectivity across a network, and does not interact with ordinary BGP route selection. This is analogous to the IX situation, in which forwarding is dealt with by the layer-2 infrastructure, with the route server being present only to mediate between BGP speakers that would otherwise be connected in a mesh.

In practice, this most general capability would be infeasible to offer due to the combinatorial explosion of routes involved, and that customized route selection would therefore employ 'BGP-like' choice at the routing level. We hope that our identification of strictly inflationary correctness conditions will be helpful in marking out possible designs for this feature.

Our Section VII suggested one way of molding BGP into a neighbor-specific protocol. In this picture, eBGP attributes are retained in order to ensure protocol convergence to a local optimum (assuming a valid configuration). Additional attributes are available for iBGP choices to be made in a neighbor-specific way. According to the algebra, this limits the flexibility of neighbor-specificity: a choice can only be made between routes which have the same preference on the external attributes. For example, it would not be possible here for customers to rank routes according to the identity of the upstream provider: we would always have to favor a route with better local preference and AS path length, even if it came from a non-preferred source. This limitation derives directly from Theorem 2: the intersection of preferences, on which everybody agrees, *must* be able to carry the correctness condition on its own. A tradeoff exists between similarity to present BGP (with its many tiebreaking attributes) and degree of neighbor-specific flexibility (where ties are mostly not broken). An analysis of local preference along the lines of the stratified shortest-paths problem may reveal further possibilities for how the externally-agreed attributes could be structured [13].

B. Topological issues

A surprising feature of the algebraic model $\mathrm{NS}(S)$ is that it does not enforce topological constraints on the network. While particular choices of S may be associated with conditions such as valley-freedom, there is no requirement that the graph labelling conform to the NS-BGP model. It might be *expected* that iBGP arcs use the simple $\preceq$ order as opposed to an extension, but this is not necessary for correctness. This should make some intuitive sense, since in the situation where each AS is a single router, we would still anticipate being able to prove convergence.

Furthermore, while we have used the term 'neighbor-specific', there is no requirement that different arcs between the same pair of neighbors be labelled consistently. Of course, this may be important for other reasons than the fact of convergence, since we are also interested in achieving a particular routing state—where the paths found should have additional properties other than stability, or where there are external factors necessitating a particular choice of preference. But we can still guarantee convergence if preferences are only adjacency-specific rather than neighbor-specific.

Similarly, the model does not supply any notion of *who is responsible* for choosing a refined order in the case of a given arc. In BGP, there is no actual concept of 'arc'! Instead, the policy applied is the result of actions taken at both ends, independently. The simple algebraic model, where arc labels are just functions f from a set F, is an abstraction from the true reality, wherein 'f' is the composition of *export policy*, decided by one AS, and *import policy*, decided by the other. For neighbor-specific preferences, a similar split applies. It would make perfect sense to have the sender, the recipient, or both, deciding which order extension to use, in just the same way as they do for their other policies.

In NS-BGP, it is envisioned that the sender is responsible for the application of preferences, though there is a potential mechanism for the other party to influence this choice out-of-band. But even in today's BGP, the recipient can evaluate routes differently depending on which neighbor supplied them. From the point of view of $\mathrm{NS}(S)$, there is no distinction—both mechanisms, and indeed their combination, fit into the algebraic model.

C. Forwarding

As recognized by other researchers [19], [23], traditional hop-by-hop forwarding is not sufficient to support this new means of route selection. An AS can no longer choose the appropriate egress point for traffic based on the destination address alone, but must also consider the identity of the neighboring AS which has sent the traffic. The use of tunnels across the AS suffices for this, along with logic at the border for directing traffic down the correct tunnel.

If neighbor-specific multipath were extended to eBGP, then the forwarding situation would become more complex. There are some similarities with pathlet routing [8], which envisages routes being composed by joining tunnels together, even across AS boundaries (assuming the term 'AS' to still be meaningful). The array of possibilities here is bewildering, including source routing and telco-style circuits as well as familiar Internet routing.

We would argue that it is premature to consider interdomain forwarding issues of this kind before a clearer picture has emerged of how autonomous systems might actually deploy neighbor-specific multipath preferences in eBGP. We do not yet know what business models might be appropriate for this setting, and what the consequences would be for network operators' views on who should be able to use their networks. Experience with NS-BGP on an internal basis may provide some clues about the interdomain possibilities.

IX. Open problems

This paper has explored the idea of neighor-specificity in path preferences. There is considerable scope for finding specific designs within this space which will be useful for operators, or applicable to particular classes of interesting problem. For NS-BGP, there are many ways in which BGP could incorporate neighbor-specificity in its attributes—whether by rewriting routes on export according to a route map, or by including one or more attributes that have a neighbor-specific interpretation as an order extension.

The question of performance is very much open. All metrics of interest (including but not limited to convergence time, amount of network traffic, and routing table size) will vary depending on the specific neighbor-specific scheme chosen, and on the details of the network configuration. It is likely that these cannot be predicted analytically—although some quantities, such as the maximal number of possible equivalent routes, can be—but require experimental determination.

Regarding the theory, it would be useful to find a way of describing particular order refinements that could be interpreted computationally. This would be a component of an implementation of neighbor-specific preferences, because at some point, someone has to tell each router which preferences it ought to be applying. In the IX scenario, this is done via routing registries, with the policies encoded in RPSL [1]. Any neighbor-specific extension of BGP would need to have some reflection in RPSL, and in other related tools.

Acknowledgement

The authors are supported by EPSRC (UK) grant EP/F002718/1. We would also like to thank Randy Bush, Pierre François, Jennifer Rexford, Michael Schapira and Laurent Vanbever for their comments and advice on this paper.

References

[1] L. Blunk, J. Damas, F. Parent, and A. Robachevsky. RFC 2012: Routing Policy Specification Language next generation (RPSLng), 2005.

[2] B. Carré. *Graphs and Networks*. Oxford University Press, 1979.

[3] Cisco Systems. Enhanced interior gateway routing protocol. White Paper 16406.

[4] D. Donaldson and J. A. Weymark. A quasiordering is the intersection of orderings. *Journal of Economic Theory*, 78(2):382–387, 1998.

[5] B. Dushnik and E. W. Miller. Partially ordered sets. *American Journal of Mathematics*, 63(3):600–610, 1941.

[6] O. Filip, P. Machek, M. Mares, and O. Zajicek. *BIRD Internet Routing Daemon User's Guide*. http://bird.network.cz.

[7] L. Gao and J. Rexford. Stable internet routing without global coordination. *IEEE/ACM Transactions on Networking*, pages 681–692, December 2001.

[8] P. B. Godfrey, I. Ganichev, S. Shenker, and I. Stoica. Pathlet routing. In *Proc. SIGCOMM*, pages 111–122, 2009.

[9] M. Gondran and M. Minoux. *Graphs and Algorithms*. Wiley, 1984.

[10] M. Gondran and M. Minoux. *Graphs, Dioids, and Semirings : New Models and Algorithms*. Springer, 2008.

[11] R. Govindan. Time-space tradeoffs in route-server implementation. *Journal of Internetworking: Research and Experience*, 6, 1995.

[12] R. Govindan, C. Alaettinoğlu, K. Varadhan, and D. Estrin. Route servers for inter-domain routing. *Comput. Netw. ISDN Syst.*, 30(12), 1998.

[13] T. G. Griffin. The stratified shortest-paths problem. In *Proc. COMSNETS*, 2010.

[14] T. G. Griffin, F. B. Shepherd, and G. Wilfong. The stable paths problem and interdomain routing. *IEEE/ACM Transactions on Networking*, 10(2):232–243, April 2002.

[15] A. J. T. Gurney. *Construction and verification of routing algebras*. PhD thesis, University of Cambridge, 2009.

[16] A. J. T. Gurney and T. G. Griffin. Lexicographic products in metarouting. In *Proc. Inter. Conf. on Network Protocols*, October 2007.

[17] E. Jasinska and C. Malayter. (Ab)Using route servers, 2010. Presentation at NANOG 48 (http://www.nanog.org/meetings/nanog48).

[18] V. Van den Schrieck, P. François, and O. Bonaventure. BGP add-paths: The scaling/performance tradeoffs. *IEEE Journal on Selected Areas in Communications*, 2010. (in press).

[19] L. Vanbever, P. François, O. Bonaventure, and J. Rexford. Customized BGP route selection using BGP/MPLS VPNs, 2009. Presentation at Cisco Systems Routing Symposium (http://inl.info.ucl.ac.be/system/files/Cisco_NAG_2009_ns_bgp.pdf).

[20] P. Verkaik, D. Pei, T. Scholl, A. Shaikh, A. C. Snoeren, and J. E. van der Merwe. Wresting control from BGP: scalable fine-grained route control. In *Proc. USENIX ATC*, pages 1–14, 2007.

[21] D. Walton, E. Retana, E. Chen, and J. Scudder. Advertisement of multiple paths in BGP, 2009. IETF Internet-Draft (expired). http://tools.ietf.org/html/draft-walton-bgp-add-paths-06.

[22] Y. Wang, I. Avramopoulos, and J. Rexford. Design for configurability: rethinking interdomain routing policies from the ground up. *IEEE Journal on Selected Areas in Communications*, 27(3):336–348, 2009.

[23] Y. Wang, M. Schapira, and J. Rexford. Neighbor-specific BGP: more flexible routing policies while improving global stability. In *Proc. 11th SIGMETRICS*, pages 217–228, 2009.

[24] W. Xu and J. Rexford. MIRO: multi-path interdomain routing. In *Proc. SIGCOMM*, pages 171–182, 2006.

[25] X. Yang, D. Clark, and A. W. Berger. NIRA: a new inter-domain routing architecture. *IEEE/ACM Transactions on Networking*, 15(4):775–788, 2007.

[26] X. Yang and D. Wetherall. Source selectable path diversity via routing deflections. In *Proc. SIGCOMM*, pages 159–170, 2006.

Appendix

Various mathematical objects are involved in our modelling of routing. We summarize their definitions here.

A *preorder* $(S, \preceq)$ is a set together with a binary relation that is reflexive ($x \preceq x$ for all x in S) and transitive (if $x \preceq y$ and $y \preceq z$ then $x \preceq z$).

A preorder that is antisymmetric (if $x \preceq y \preceq x$ then $x = y$) is a *partial order*. If in addition it is total (for all x and y, either $x \preceq y$ or $y \preceq x$) then it is a *total order* or *linear order*.

A *semigroup* $(S, \oplus)$ is a set with an associative binary operation. It may be commutative ($x \oplus y = y \oplus x$ for all x and y) or idempotent ($x = x \oplus x$ for all x) but need not be either.

If we have a set F of functions over S, then we can combine this with a preorder or a semigroup to make a *preorder transform* $(S, \preceq, F)$ or a *semigroup transform* $(S, \oplus, F)$.

A graph $G = (V, E)$ can be weighted over a preorder transform by providing

1) a function $s : V \to S$, and
2) a function $w : E \to F$.

Then, the weight of a path from node i to node j in G, which uses the arcs e_1 through e_k, can be calculated as

$$(w(e_k) \circ w(e_{k-1}) \circ \cdots \circ w(e_2) \circ w(e_1))\,(s(i)).$$

The s function thus supplies an originated value for each node, and each arc function alters this value in some way. The path weight is, however, still a value in S, which can be compared with other such values via $\preceq$ as expected.

Graph weightings over semigroup transforms are defined in the same way.

Two orders (indeed, any relations) can be combined via intersection. The intersection of $\preceq_1$ and $\preceq_2$ is the order $\preceq$ where

$$x \preceq y \iff x \preceq_1 y \wedge x \preceq_2 y.$$

This naturally extends to more than two orders.

A function f over an order $(S, \preceq)$ is said to be *inflationary* when

$$\forall x \in S : x \preceq f(x).$$

If S has a greatest element, then f is *strictly inflationary* when

$$\forall x \in S : x \prec f(x) \vee x = f(x) = \top$$

Otherwise, the definition of strict inflation only requires that $x \prec f(x)$ in all cases. These definitions extend to sets of functions in the obvious way.