Multi-level optimization. Space mapping and manifold mapping
Echeverria, D.

Citation for published version (APA):

General rights
It is not permitted to download or to forward/distribute the text or part of it without the consent of the author(s) and/or copyright holder(s), other than for strictly personal, individual use, unless the work is under an open content license (like Creative Commons).

Disclaimer/Complaints regulations
If you believe that digital publication of certain material infringes any of your rights or (privacy) interests, please let the Library know, stating your reasons. In case of a legitimate complaint, the Library will make the material inaccessible and/or remove it from the website. Please Ask the Library: http://uba.uva.nl/en/contact, or a letter to: Library of the University of Amsterdam, Secretariat, Singel 425, 1012 WP Amsterdam, The Netherlands. You will be contacted as soon as possible.
Chapter 2

Space-Mapping Review

2.1 Introduction

The space-mapping idea was conceived by Bandler [8] in the field of microwave filter design. It aims at reducing the cost of accurate optimization processes by iteratively correcting a sequence of rougher approximations. In technological applications this allows us to couple, for example, simple rules that represent expert knowledge accumulated over the years with the accuracy of expensive simulation techniques based on the numerical solution of partial differential equations. This combination may yield an efficient method with good accuracy of the final solution.

The space-mapping technique has been mainly applied in electromagnetics [7, 13] but, since the underlying principles are quite general, it could also be used in other areas such as structural [56], vehicle crashworthiness [74] or thermal [68] design. Space mapping has developed enormously during the last decade. As can be seen in the rather complete survey [7], the original idea has gone through a large number of changes and improvements. Further, it can be applied in model reduction for the construction of cheap surrogates [16, 80]. But this modeling purpose is in our work of secondary importance.

In this chapter we analyze space mapping from the perspective of defect correction. Defect correction is a well-understood principle in computational mathematics [22] to solve accurate but complex operator equations in an iterative manner with the help of approximate but easier operators. The idea of solving a problem with the help of a sequence of simpler versions is quite old and well established. Famous examples are, e.g., Newton iteration and multigrid methods [49].

Also the space-mapping technique can be considered as defect correction iteration. This viewpoint allows some mathematical analysis and the construction
of a family of more general iteration procedures, generalizing space mapping.

In this chapter we introduce the concept of model flexibility, that gives additional insight and understanding of space-mapping performance. The most significant conclusion derived from that concept is that the best space-mapping techniques are those in which both models involved have equal flexibility, while the less accurate model can be solved much faster.

Traditional space-mapping techniques cannot change the model flexibility. We present manifold mapping, a new space-mapping approach that, by additional left preconditioning [88] with an affine operator, aims at locally endowing the two given models with equal flexibility. Our analysis makes clear that this properly adapted left-preconditioning is essentially more effective than the traditional space mapping, which corresponds with right-preconditioning.

The recently published output space mapping (OSM) [12] and space-mapping based interpolating surrogate (SMIS) scheme [15] are approaches that, following the framework proposed in [1], yield convergence to the accurate solution at the expense of incorporating in the algorithms exact gradient information. The manifold-mapping technique has provable convergence to the true solution and can be used without computing that usually expensive derivative information.

The present chapter is structured as follows. Section 2.2 briefly presents the most fundamental concepts and definitions of space mapping. Then it introduces the two basic types of space-mapping solutions and presents typical algorithms found in literature for their computation. The notions of perfect mapping and flexibility of a model are presented in Section 2.3 and there also, some relations between these are explained. In Section 2.4 we indicate how the space-mapping technique can be efficiently applied in designs where the constraint evaluation is time-expensive. In Section 2.5 the defect correction principle is introduced and its relation with space mapping is shown. Some defect correction iteration schemes can be recognized as space-mapping methods but also generalizations can be found. These processes are analyzed and a procedure is constructed so that locally the model flexibility for the different models coincides. This yields our improved space-mapping method. We conclude by means of a few simple examples in Section 2.6 where we illustrate some essential concepts and lemmas from the space-mapping theory presented along the chapter.

2.2 Space-Mapping optimization

Although space mapping is mostly employed as a means for optimization, it is useful to know that it can be applied with two different objectives in mind — optimization and modeling— and that, with the different use, different emphasis is put on particular aspects of space mapping.

We distinguish between: (1) The space-mapping technique aiming at efficient optimization of complex technical problems. The idea is to combine the
advantages of simple (coarse, less accurate, easy to evaluate) models and more complex (fine, more accurate, expensive) models of a phenomenon, to accelerate the numerical computation of a fine model optimum. Typical examples of such an optimization problem are, e.g. fitting a model to a set of measured data or determining the shape of a device in order to achieve a physical response with certain properties. More generally, (2) space mapping deals with constructing a simpler or surrogate model instead of a more expensive and accurate one, and with calibrating this surrogate.

The main difference between the two purposes of space mapping (optimization or modeling) is the required validity region in which the surrogate approximation should correspond with the fine one. In the first case it is sufficient if the two models correspond in the neighborhood of the optimum where the optimization procedure dwells. In the latter case the surrogate model should be sufficiently accurate over a larger range of its parameters.

Though most of the concepts can also be applied in modeling, our work is primarily concerned with the use of space mapping in optimization.

2.2.1 The space-mapping function

It is supposed to be much easier to compute \( z^* \) than \( x^* \). With this knowledge and with the information regarding the similarity between the two models \( f \) and \( c \), we aim at an efficient algorithm to compute an accurate approximation of \( x^* \). Obviously, the similarity or discrepancy between the responses of two models used for the same phenomenon is here an important property. It is expressed by the misalignment function (see Figure 2.1),

\[
r(z, x) = \| c(z) - f(x) \|. 
\]

In particular, for a given \( x \in X \) it is useful to know which \( z \in Z \) yields the smallest discrepancy. This information would allow us to improve the coarse model. Therefore, the space-mapping function is introduced.

**Definition 6.** The space-mapping function \( p : X \subset \mathbb{R}^n \rightarrow Z \subset \mathbb{R}^n \) is defined by

\[
p(x) = \arg\min_{z \in Z} r(z, x) = \arg\min_{z \in Z} \| c(z) - f(x) \|. \tag{2.1}
\]

If (2.1) does not allow for a unique minimum, no space mapping \( p : X \rightarrow Z \) exists for that \( x \in X \).

The definition implies that for a given control \( x \) the function \( p \) delivers the best coarse model control \( z \) that yields a similar (or the same) response as \( x \) for the fine model. The existence of a unique minimizer is implied by Assumption 3 if \( f(X) \subset Y \), which can be expected in practice.

The process of finding \( p(x) \) for a given \( x \) is called parameter extraction or single point extraction because it finds the best coarse model parameter that
Figure 2.1: Misalignment and space-mapping function.

The left figure shows the misalignment function for a fine and a coarse model for the problem in Section 5.2.1. Darker shading shows a smaller misalignment. The right figure shows the identity function and a few space-mapping functions for different coarse models used in Section 5.2.1. (Only the variable $x_3$ is shown; $x_1 = 5$ mm, $x_2 = 7.5$ mm are kept fixed.)

corresponds with a given fine model control $x$. It should be noted that this evaluation of the space-mapping function $p(x)$ requires both an evaluation of $f(x)$ and a minimization process with respect to $z$ in $\|c(z) - f(x)\|$. Hence, in algorithms we should make economic use of space-mapping function evaluations.

The success of the space-mapping technique relies on the existence of desirable properties for the space-mapping function. One is its injectivity. If the model $f$ has more control parameters than $c$, the space-mapping function cannot be injective. For that reason, together with the fact that both fine and coarse control variables mostly have the same interpretation, we restrict ourselves to spaces $X$ and $Z$ having the same dimension.

For ill-conditioned designs we expect the space-mapping function to be not injective (different design configurations may yield identical responses). In these situations, it makes sense to consider a coarse model based on a smaller number of design variables (selected for example, with a sensitivity analysis). This is the approach followed in partial space-mapping (PSM) [17]. Ill-conditioned problems properly reformulated (for example, with the introduction of additional regularizing constraints) can be solved within the two-level framework of design spaces with the same number of dimensions.

In implicit space-mapping (ISM) [14] some extra coarse model parameters are introduced for a better model alignment. Though this approach seems to be based on a coarse model with a larger number of design variables than the fine one, the space-mapping function considered in ISM is the original one, with
spaces of equal dimension.

In circumstances where the coarse and the fine model reachable aims are one
a subset of the other, i.e. if \( c(Z) \subset f(X) \) or \( f(X) \subset c(Z) \), or if they coincide, the
notion of flexibility of a model can be introduced (Section 2.3) to derive properties
of the corresponding space-mapping function. A priori there is no reason to
assume that these sets have such a relation. Even a non-empty intersection
might occur. However, given that both models describe the same phenomenon,
in some situations such inclusion relation may exist.

**Perfect mapping.** For reasons that will become immediately clear in the next
section, the following definition is introduced [82, Chapter 4].

**Definition 7.** A space-mapping function \( p \) is called a perfect mapping iff
\[
x^* = p(x^*).
\]

Using the definition of the space-mapping function we see that (2.2) can be
written as
\[
\arg\min_{x \in Z} \| c(x) - y \| = \arg\min_{x \in Z} \| c(x) - f(x^*) \|,
\]
i.e., a perfect space-mapping function maps \( x^* \), the solution of the fine model
optimization, exactly onto \( z^* \), the minimizer of the coarse model design. However,
we notice that *perfection* is not a property of the space-mapping function alone
but it also depends on the data \( y \) considered. A space-mapping function can be
perfect for one set of data but imperfect for a different data set.

The space-mapping function is a crucial element within the space-mapping
technique. Quite often in practice multiple solutions to (2.1) are found [7]. Sev-
eral regularization criteria have been suggested in order to alleviate that difficulty.
Examples of these are Multi-point Parameter Extraction (MPE) [11], Statistical
Parameter Extraction (SPE) [11], Penalized Parameter Extraction (PPE) [10] or
Gradient Parameter Extraction (GPE) [17]. As we will see in the next chapter,
manifold mapping avoids the parameter extraction process by taking instead of \( p \)
a known mapping \( \tilde{p} : X \to Z \) with an easy-to-compute inverse. The significant
problem of non-uniqueness associated to (2.1) is thus avoided in manifold
mapping.

### 2.2.2 Space-Mapping approaches

Many space-mapping based algorithms can be found in literature [6, 7], but they
all have the same basis. In this section we first describe the original space-
mapping idea and two principal approaches (primal and dual). In Section 2.2.3
we show the most important space-mapping algorithms used in practice, and
Figure 2.2: The original space-mapping algorithm.

\[ x_0 = z^* = \arg \min_{x \in Z} \| c(x) - y \|; \]

\[ p_0 = I; \]

for \( k = 0, 1, \ldots, \) while \( \| p(x_k) - z^* \| > \text{tolerance} \)

do \( z_k = p(x_k) = \arg \min_{x \in Z} \| c(z) - f(x_k) \|; \)

from \( \{ (x_j, z_j) \}_{j=0}^k \) determine an updated approximation\(^a\) \( p_{k+1} \) for \( p; \)

\[ x_{k+1} = p_{k+1}(z^*); \]

enddo

\(^a\)the approximation is assumed to be invertible

Figure 2.3: The primal space-mapping algorithm.

\[ x_0 = z^* = \arg \min_{x \in Z} \| c(x) - y \|; \]

for \( k = 0, 1, \ldots, \) while ...

do \( z_k = p(x_k); \)

compute an updated approximation \( p_{k+1} \) by means of \( \{ (x_j, z_j) \}_{j=0}^k \);

\[ x_{k+1} = \arg \min_{x \in X} \| p_{k+1}(x) - z^* \|; \]

enddo

later, in Section 2.3, we analyze what approximations to a solution are obtained under certain conditions.

The basic idea behind space-mapping optimization is the following: if either the fine model allows for an almost reachable design (i.e., \( f(x^*) \approx y \)) or if both models are similar near their respective optima (i.e., \( f(x^*) \approx c(z^*) \)) we expect

\[ p(x^*) = \arg \min_{x \in Z} \| c(x) - f(x^*) \| \approx \arg \min_{x \in Z} \| c(x) - y \| = z^*. \]

Based on this relation, the original space-mapping approach [8] assumes \( p(x^*) \approx z^* \). It first determines \( z^* \) and then tries to solve

\[ p(x_{mn}^*) = z^*. \] (2.4)

The algorithm for its computation is shown in Figure 2.2. However, in general \( p(x^*) \neq z^* \) and even \( z^* \in p(X) \) is not guaranteed, so that the existence of \( x_{mn}^* \) cannot be assumed. Therefore, the primal space-mapping approach seeks for a solution of the minimization problem

\[ x_p^* = \arg \min_{x \in X} \| p(x) - z^* \|. \] (2.5)

This leads to the primal space-mapping algorithm as shown in Figure 2.3.

An alternative approach can be taken. The idea behind space-mapping optimization is the replacement of the expensive fine model optimization by a surrogate model one. For the surrogate model we can take the coarse model \( c(x), \)
Figure 2.4: The dual space-mapping algorithm.

\[ x_0 = z^* = \arg\min_{x \in X} \| c(x) - y \| ; \]
\[ \text{for } k = 0, 1, \ldots, \text{while } \ldots \]
\[ \text{do } z_k = p(x_k); \]
\[ \text{compute an updated approximation } p_{k+1} \text{ by means of } \{(x_j, z_j)\}_{j=0}^k; \]
\[ x_{k+1} = \arg\min_{x \in X} \| c(p_{k+1}(x)) - y \| ; \]
\[ \text{enddo} \]

and improve its accuracy by the space-mapping function \( p : X \to Z \). Now the improved or mapped coarse model \( c(p(x)) \) may serve as the better surrogate model. Because of (2.1) we expect that \( c(p(x)) \approx f(x) \) and hence

\[ \| f(x) - y \| \approx \| c(p(x)) - y \|. \]

Then the minimization of \( \| c(p(x)) - y \| \) will usually give us a value, \( x^*_d \), close to the desired optimum \( x^* \):

\[ x^*_d = \arg\min_{x \in X} \| c(p(x)) - y \|. \tag{2.6} \]

This leads to the dual space-mapping algorithm as shown in Figure 2.4.

We will see in Section 2.3 that all three approaches coincide when \( z^* \in p(X) \) and \( p \) is injective. In Section 2.6 the case where the space-mapping solutions are different will be illustrated. A notable difficulty of all the approaches is the expensive evaluation of \( p(x) \). One way to deal with this problem is via convenient choices for the approximations to \( p(x) \) indicated in each of the generic schemes given in Figures 2.2-2.4. The selection of a suitable approximation is made in the specific algorithms below.

### 2.2.3 Space-Mapping algorithms

The three algorithms given in the Figures 2.2-2.4 are vague in the sense that no choice has been made yet for the approximation of the space-mapping function \( p \). Using a linear approximation gives rise to the more popular space-mapping optimization algorithms. The two most representative examples are shown below. An extensive survey of available algorithms can be found in [7].

**The ASM algorithm [9].** The space-mapping function is approximated by linearization to obtain

\[ p_k(x) = p(x_k) + B_k (x - x_k). \]

In each space-mapping iteration step the matrix \( B_k \) is adapted by a rank-one update. For that purpose a Broyden-type approximation [27] for the Jacobian of
the space-mapping function \( p(x) \) is used,

\[
B_{k+1} = B_k + \frac{p(x_{k+1}) - p(x_k) - B_k h \cdot h^T}{h^T h},
\]

where \( h = x_{k+1} - x_k \). This is combined with original space mapping, so that \( x_{k+1} = x_k - B_k^{-1}(p(x_k) - z^*) \) and the aggressive space mapping (ASM) is obtained (Figure 2.5). Note that Broyden’s method cannot assure that every \( B_k \) is nonsingular, so, the inverse in the original space-mapping scheme should be read as a pseudo-inverse. ASM just solves equation (2.5) by a quasi-Newton iteration with an approximate Jacobian. We notice that only one evaluation of the fine model is needed per iteration.

The TRASM algorithm [3]. Like ASM, the trust-region aggressive space mapping (TRASM) algorithm solves (2.5), now taking into account a trust region. I.e., to stabilize the algorithm, specially in its initial stage, every iterant \( x_{k+1} \) is found in a region not too far away from the earlier one. Further, making use of the Levenberg-Marquardt algorithm [60], TRASM restricts itself to minimizing the residue in the Euclidean norm. By Levenberg-Marquardt the inner loop of TRASM uses the update formula

\[
(B^T_k B_k + \lambda_k I) h_k = -B^T_k (p(x_k) - z^*),
\]

\[
x_{k+1} = x_k + h_k,
\]

where \( B_k \) is the \( k \)-th approximation of the Jacobian of \( p(x) \) at \( x_k \). Notice that the adaptation of the trust region is controlled by adapting \( \lambda_k \). For \( \lambda_k = 0 \) the iteration step reduces to a quasi-Newton one, whereas for larger \( \lambda_k \) the step-length reduces and the method tends to steepest descent for minimizing \( \| p(x) - x^* \| \). Generally \( B_k \) is constructed by rank-one updates as in ASM. The TRASM algorithm is shown in Figure 2.6.
The HASM algorithm [4]. In general the space-mapping function \( p \) will not be perfect, and hence a space-mapping based algorithm will not yield the solution of the fine model optimization. Therefore, space mapping is sometimes combined with a classical optimization method. This can be done in several ways. One obvious way is using a space-mapping solution as an initial guess for an arbitrary other (classical) optimization algorithm. Another approach is by constructing a combination of a space-mapping and a classical optimization method. Such an algorithm is called hybrid aggressive space mapping (HASM). It exploits space mapping when effective, otherwise it defaults to the linearization of the fine model response \( f(x) \)

\[
l_k(x) = f(x_k) + \hat{B}_k (x - x_k),
\]

where \( \hat{B}_k \) is a Broyden rank one update approximation to the Jacobian of \( f(x) \) in \( x_k \). It is proved in [58] that (under mild conditions) the iterative process

\[
x_{k+1} = \arg\min_{x \in X} \| l_k(x) - y \|
\]

converges to \( x^* \).

The combined model \( v_k(x) \) is introduced as a convex combination of the mapped coarse and the linearized model:

\[
v_k(x) = \omega_k \cdot c(p_k(x)) + (1 - \omega_k) \cdot l_k(x), \tag{2.7}
\]

with \( \omega_k \in [0, 1] \). Often \( \omega_k \) is used as a simple switch: either \( \omega_k = 1 \) or \( \omega_k = 0 \). Now \( \| v_k(x) - y \| \) is the functional to be minimized in the HASM scheme (Figure 2.7). Conditions for convergence are given in [59]. It should be noted that the combined model \( v_k(x) \) is only slightly more expensive than the mapped coarse one \( c(p_k(x)) \) since for both, the same number of fine model evaluations is used.
2.3 Perfect mapping, flexibility and reachability

By its definition, perfect mapping relates the similarity of the models and the specifications. If the fine model allows for a reachable design (i.e., $f(x^*) = y$), then it is immediate that, independent of the coarse model used, the mapping is always perfect. Also if the coarse and the fine model optimal responses are identical, the space-mapping function is perfect. These two facts are summarized in the following lemma.

**Lemma 1.** If either a design is reachable for the fine model, or if the fine and the coarse model give the same optimal response, then the corresponding space-mapping function is perfect. In formula: (i) If $f(x^*) = y$ then $p(x^*) = z^*$.

(ii) If $f(x^*) = c(x^*)$ then $p(x^*) = z^*$. 

The fine and the coarse model in space mapping should, by their nature, show important similarities. Normally the coarse model needs only a small correction in order to be aligned with the fine one. Then the space-mapping function is just a small perturbation of the identity. Therefore, we can expect that, with the tolerances handled in practice, there exists a point $x \in X$ such that $f(x) = c(x^*)$ and thus, that $p(x) = z^*$. This is a most interesting situation in which many of the different cases discussed above coincide and space mapping may lead us to the desired optimum. This is shown in the following lemma.

**Lemma 2.** (i) If $z^* \in p(X)$, then $p(x_{sm}^*) = p(x_p^*) = p(x_d^*) = z^*$.

(ii) If, in addition, $p$ is an injective perfect mapping then $x^* = x_{sm}^* = x_p^* = x_d^*$.

*Proof.* In case $z^* \in p(X)$ by definition (2.4) $x_{sm}^*$ exists and $p(x_{sm}^*) = z^*$. Now by definition (2.5) we see $p(x_{p}^*) = p(x_{sm}^*)$. Further, $c(p(x_{sm}^*)) - y = c(z^*) - y$, so that by definition (2.6) we have $p(x_{d}^*) = p(x_{sm}^*)$. Summarizing,
\[ p(\mathbf{x}_{\text{sum}}) = p(\mathbf{x}^*_p) = p(\mathbf{x}^*_d) = \mathbf{z}^*. \] If, further, \( p \) is injective, the original, primal and dual approach yield the same result. If, in addition, \( p \) is a perfect mapping then \( \mathbf{x}_{\text{sum}} = \mathbf{x}^*_p = \mathbf{x}^*_d = \mathbf{z}^*. \)

In some cases we can expect that the sets of fine and coarse reachable aims \( f(X) \) and \( c(Z) \), respectively, may overlap in a region of \( \mathbb{R}^m \) close to their respective optima. The concept of model flexibility is now introduced and from that notion some results concerning properties of the space-mapping function can be derived.

**Definition 8.** A model is called more flexible than another if the set of its reachable aims contains the set of reachable aims of the other. Two models are equally flexible if their respective sets of reachable aims coincide.

Thus, a coarse model \( c \) is more flexible than the fine one \( f \) if \( c(Z) \supset f(X) \), i.e., if the coarse model response can reproduce all the fine model reachable aims. E.g., if \( n \geq m \), linearization makes a flexible coarse model (as long as the Jacobian is regular). Similarly the fine model is more flexible if \( f(X) \supset c(Z) \). Model flexibility is closely related to properties of the space-mapping function. This is shown in the following lemmas, where \( p \) denotes the space-mapping function.

**Lemma 3.** If \( c \) is more flexible than \( f \) then
(i) \( c(p(\mathbf{x})) = f(\mathbf{x}) \) \( \forall \mathbf{x} \in X \);
(ii) \( p : X \to Z \) is a perfect mapping if and only if \( c(\mathbf{z}^*) = f(\mathbf{z}^*) \);
(iii) if \( f : X \to Y \) is injective then \( p : X \to Z \) is injective;
(iv) if \( c(Z) \setminus f(X) \neq \emptyset \), then \( p : X \to Z \) cannot be surjective.

**Proof.**
(i) \( \forall f(\mathbf{x}) \exists c(\mathbf{z}) : c(\mathbf{z}) = f(\mathbf{x}) \Rightarrow c(p(\mathbf{x})) = f(\mathbf{x}) \) \( \forall \mathbf{x} \in X \).
(ii) \( f(\mathbf{x}^*) = c(p(\mathbf{x}^*)) = c(\mathbf{z}^*) \) because of (i) and perfect mapping.
(iii) \( p(\mathbf{x}^*) = \arg \min_{\mathbf{z} \in Z} ||c(\mathbf{z}) - f(\mathbf{x}^*)|| = \arg \min_{\mathbf{z} \in Z} ||c(\mathbf{z}) - c(\mathbf{z}^*)|| = \mathbf{z}^* \) because the optimum is unique.
(iv) \( p(\mathbf{x}_1) = p(\mathbf{x}_2) \Rightarrow c(p(\mathbf{x}_1)) = c(p(\mathbf{x}_2)) \Rightarrow f(\mathbf{x}_1) = f(\mathbf{x}_2) \) because of (i) \( \Rightarrow \mathbf{x}_1 = \mathbf{x}_2 \) because \( f \) is injective.
(iv) Let be \( \bar{z} : c(\bar{z}) \in c(Z) \setminus f(X) \). But then, because of (i), we see that it is not possible to find \( \bar{x} \in X : p(\bar{x}) = \bar{z} \).

**Remark.** Because of (ii) generally we cannot expect space-mapping functions to be perfect for flexible coarse models unless the two models are equally flexible near the optimum. This fact is illustrated in Figure 2.8. However, we notice that if the design is reachable, the perfect mapping property holds, even if \( c(Z) \setminus f(X) \neq \emptyset \).

**Lemma 4.** If \( f \) is more flexible than \( c \) then
(i) \( p : X \to Z \) is surjective;
(ii) if \( f(X) \setminus c(Z) \neq \emptyset \), then \( p \) cannot be injective.
Figure 2.8: Perfect mapping cannot be expected in general for more flexible coarse models.

\[ f(X) \supset c(Z) \Rightarrow \forall z \exists x \in Z : c(z) = f(x) \]

Proof. (i) If f and c are equally flexible and f is injective, then (i) \( p \) is a perfect mapping.

Lemma 5. If f and c are equally flexible and f is injective, then (i) \( p \) is a perfect mapping.

Proof. (i) Combine lemmas 3 and 4.

The conclusions in Lemma 2 can now be derived from assumptions about model flexibility. Therefore, we can express when the basic space-mapping approach yields the true optimum \( \mathbf{x}^* \) in terms of properties concerning the models and their mutual relation (and with independence of the particular problem specifications \( y \)).

Lemma 6. (i) If f is more flexible than c, then \( p(x_m^*) = p(x_p^*) = p(x_s^*) = \mathbf{x}^* \).

Remark. It is not really necessary for the space-mapping function to be a bijection over the whole domain in which it is defined. In fact, perfect mapping is a property that concerns only a point, and it is enough if the function is injective in a (small) neighborhood. Thus, the assumptions for the lemmas above can be relaxed and stated just locally.
2.4 Constrained optimization with space mapping

The space-mapping theory presented in the previous section is general and it can also be applied to constrained optimization. Just rewriting (2.1) with explicit constraints, we obtain

\[ p(x) = \arg\min_{x \in \mathcal{X}} \|c(z) - f(x)\| \quad \text{subject to} \quad k_e(z) \geq 0. \] (2.8)

The space-mapping solutions should be computed according to (2.5) and (2.6) respectively

\[ x_f^* = \arg\min_{x \in \mathcal{X}} \|p(x) - z^*\| \quad \text{subject to} \quad k_f(x) \geq 0, \] (2.9)
\[ x_d^* = \arg\min_{x \in \mathcal{X}} \|c(p(x)) - y\| \quad \text{subject to} \quad k_f(x) \geq 0. \] (2.10)

If the fine constraints \( k_f \) are expensive to evaluate, this approach is not likely to be the most efficient one.

The double space-mapping approach. In [56] an alternative approach is suggested. There, the models are aligned without taking into account the constraints, and with the aid of an additional space mapping. The evaluation of \( k_f \) is replaced by something computationally cheaper. This is an interesting point of view because in many cases the models are not closely related to their respective constraints (for example, a model can be proposed for a certain physical quantity and the constraints are associated with a completely different one). Thus, it makes no much sense to try to align them simultaneously with just one single space-mapping function. In [56] the following two space-mapping functions \( p_m : \mathcal{X} \rightarrow \tilde{Z} \) and \( p_k : \mathcal{X} \rightarrow \tilde{Z} \) are introduced

\[ p_m(x) = \arg\min_{z \in \mathcal{Z}} \|c(z) - f(x)\|, \] (2.11)
\[ p_k(x) = \arg\min_{z \in \mathcal{Z}} \|k_e(z) - k_f(x)\|, \] (2.12)

for the alignment of the models and the constraints, respectively. Regularization might be needed in many situations because the constraints and the model minimization have been dropped in (2.11) and in (2.12), respectively.

The constraint space-mapping solution is defined in this case as

\[ x_{c,m}^* = \arg\min_{x \in \mathcal{X}} \|c(p_m(x)) - y\| \quad \text{subject to} \quad k_e(p_k(x)) \geq 0. \] (2.13)

Algorithms for this approach can be obtained through approximations of both space-mapping functions \( p_m \) and \( p_k \), as it was done in Section 2.2.3 for standard space mapping.

If we define the function \( p_{c,m} : \mathbb{R}^n \rightarrow \mathbb{R}^{n + m_k} \) as a concatenation of \( p_m \) and \( k_e \circ p_k \), i.e., \( p_{c,m} = [p_m; k_e \circ p_k] \), the next result, similar to Lemma 2, can be formulated.
Lemma 7. If
\[ p_{ca.m}(x^*) = [z^*; k_c(z^*)] \]  
(2.14)
and in addition, \( p_{ca.m} \) is injective then \( x_{ca.m}^* = x^* \).

Proof. By definition of the constraint space-mapping solution \( x_{ca.m}^* \) and because of (2.14) we have that \( p_n(x_{ca.m}^*) = z^* \) and \( k_c(p_n(x_{ca.m}^*)) = k_c(z^*) \). If \( p_{ca.m} \) is injective, we then have that \( x_{ca.m}^* = x^* \).

We recognize in (2.14) a condition similar to perfect-mapping, but now involving the constraints. It should be noticed that the assumption of \( k_c \) being injective cannot be made here, because in most cases the number of design variables is larger than the number of constraints (i.e., \( n > m_k \)). As with the general space-mapping theory, condition (2.14) might hold only by approximation. For the following case of practical relevance (see [36] or Section 6.2), we can prove that \( x_{ca.m}^* = x^* \).

A particular case: \( f \equiv c \) and \( n \geq m_k \). In some cases most of the computational effort during an optimization is concentrated in the evaluation of the constraints. For example, when minimizing the mass of a device with a desired functionality. In that case, the model is the mass computation and the function performed by the device can be described by a certain number of constraints. Mass computation is very often a simple task and, thus, the fine and coarse models coincide. The situation \( n \geq m_k \) is also observed in practice [36].

The model space-mapping function \( p_n \) can be taken as the identity since both models are already aligned (\( \hat{X} = \hat{Z} \) and \( f \equiv c \)). Since \( n \geq m_k \), it can be assumed that both constraint models \( k_c \) and \( k_f \) are equally flexible in a neighborhood \( U_{x^*} \subset \hat{X} \) of the true solution \( x^* \). For the same reasons as in Lemma 3 we have
\[ k_c(p_k(x)) = k_f(x) \quad \forall x \in U_{x^*}. \]  
(2.15)

Thus,
\[
\begin{align*}
x_{ca.m}^* &= \arg\min_{x \in \hat{X}} \|c(p_n(x)) - y\| \quad \text{subject to} \quad k_c(p_k(x)) \geq 0 \\
&= \arg\min_{x \in U_{x^*}} \|c(p_n(x)) - y\| \quad \text{subject to} \quad k_c(p_k(x)) \geq 0 \\
&= \arg\min_{x \in U_{x^*}} \|f(x) - y\| \quad \text{subject to} \quad k_f(x) \geq 0.
\end{align*}
\]

We can interpret this result as follows: if the optimization dwells in a neighborhood of the true optimum (for example, with a good initial guess), then the constraint space-mapping solution \( x_{ca.m}^* \) is the fine optimum \( x^* \). The situation in this particular case is essentially the same as for a reachable design, but now stated for the constraints.

When \( n > m_k \), in general, the minimization in (2.12) will not have a unique solution (multiple points in the design space \( \hat{X} \) may satisfy the same constraints).
For example, the following two regularization criteria can be used  
\[ p_{k_1}(x) = \arg\min_{z \in Z} \| c(z) - y \| \text{ subject to } k_c(z) = k_f(x), \]
\[ p_{k_2}(x) = \arg\min_{z \in Z} \| c(z) - f(x) \| \text{ subject to } k_c(z) = k_f(x). \]
In the first case the preferred solution is the closest to the specifications \( y \). The second criterion is based on the proximity to the fine model response \( f(x) \). The respective constraint space-mapping solutions coincide with the fine optimum (if for example, a good enough initial guess can be taken for the associated algorithms; i.e., the assumption of being in a neighborhood of the true solution is satisfied).

2.5 Defect correction and space mapping

The main idea underlying the space-mapping technique, i.e. the efficient solution of a complex problem by the iterative use of a simpler one, is known since long in computational mathematics. Generally, it is known as defect correction iteration (of which, e.g., Newton’s method, relaxation procedures [22, 49], iterative refinement [22] and multigrid methods [49] are examples). The general principles of the idea are well understood [21, 22, 84]. In this section we first briefly summarize the defect correction principle for the solution of operator equations and extend the idea to optimization problems. We compare defect correction with space mapping and show how space mapping can be derived from defect correction. Then, in the general framework of defect correction new space-mapping algorithms are constructed and analyzed.

2.5.1 The defect correction principle

Let us first consider the problem of solving a nonlinear operator equation  
\[ \mathcal{F} x = y, \]  
where \( \mathcal{F} : D \subset E \rightarrow \hat{D} \subset \hat{E} \) is a continuous, generally nonlinear operator and \( E \) and \( \hat{E} \) are Banach spaces. In general, neither injectivity nor surjectivity of the mapping are assumed, but in many cases these properties can be achieved by a proper choice of the subsets \( D \) and \( \hat{D} \).

The classical defect correction iteration for the solution of the equation (2.16) with \( y \in \hat{D} \) is based on a sequence of operators \( \mathcal{F}_k : D \rightarrow \hat{D} \) approximating \( \mathcal{F} \). We assume that each \( \mathcal{F}_k \) has an easy-to-calculate inverse \( \mathcal{G}_k : \hat{D} \rightarrow D \). Actually, it is the existence of the easy-to-evaluate operator \( \mathcal{G}_k \), rather than the existence of \( \mathcal{F}_k \), what is needed for defect correction, and we do not need to assume neither \( \mathcal{F}_k \) nor \( \mathcal{G}_k \) to be invertible.
Defect correction comes in two brands [21], depending on the space, $E$ or $E$, in which linear combinations for extrapolation are made. The two basic iterative defect correction procedures to generate a sequence of approximations to the solution of (2.16) are

$$
\begin{align*}
\begin{cases}
x_0 &= \bar{G}_0 y, \\
x_{k+1} &= (I - \bar{G}_{k+1} \bar{F})(x_k + \bar{G}_{k+1} y),
\end{cases}
\end{align*}
$$

and

$$
\begin{align*}
\begin{cases}
l_0 &= y, \\
l_{k+1} &= (I - \bar{F} \bar{G}_k) l_k + y.
\end{cases}
\end{align*}
$$

In the latter we identify the approximate solution as $x_k \equiv \bar{G}_k l_k$. We see that the two iteration processes are dual in the sense that the extrapolation in (2.17) is in the space $E$, whereas the additions in (2.18) are in $E$. If $\bar{G}_k$ is injective, then an operator $\bar{F}_k$ exists such that $\bar{F}_k \bar{G}_k = I_D$, i.e., $\bar{F}_k$ is the left-inverse of $\bar{G}_k$. Then $\bar{F}_k x_k = l_k$ and (2.18) is equivalent with the iterative procedure

$$
\begin{align*}
\begin{cases}
\bar{F}_0 x_0 &= y, \\
\bar{F}_{k+1} x_{k+1} &= \bar{F}_k x_k - \bar{F} \bar{G}_k \bar{F}_k x_k + y.
\end{cases}
\end{align*}
$$

In order to apply (2.19), the injectivity of $\bar{G}_k$ is not really needed and it is immediately seen that neither (2.18) nor (2.19) converge if $y \not\in D$. However, (2.19) can be modified so that it can be used for $y \not\in D$. Therefore, we require injectivity for $\bar{F}_k$ and take $\bar{G}_k$ its left-inverse, i.e., $\bar{G}_k \bar{F}_k = I_D$. Then (2.19) leads to

$$
\begin{align*}
\begin{cases}
x_0 &= \bar{G}_0 y, \\
x_{k+1} &= \bar{G}_{k+1} \left(\bar{F}_k x_k - \bar{F} \bar{F}_k x_k + y\right).
\end{cases}
\end{align*}
$$

Because (2.20) allows for a non-injective $\bar{G}_k$ this procedure can be used for optimization purposes. In case of an invertible $\bar{G}_k$ both (2.19) and (2.20) are equivalent with (2.18).

Assuming that for large enough $k$ the approximate operators $\bar{G}_k = \bar{G}$ and $\bar{F}_k = \bar{F}$ do not change anymore, upon convergence we see that with $\lim_{k \to \infty} x_k = x$ and $\lim_{k \to \infty} l_k = l$ we find for (2.17), (2.18), (2.19) and (2.20) respectively

$$
\begin{align*}
\bar{G} \bar{F} x &= \bar{G} y, \\
\bar{F} \bar{G} l &= \bar{F} x = y, \\
\bar{F} \bar{F} x &= y, \quad \text{and}
\end{align*}
$$
\[ \mathbf{x} = \tilde{G}(\mathcal{F}\mathbf{x} - \mathcal{F}\mathbf{x} + \mathbf{y}). \]  

(2.24)

The process (2.17) is convergent if the operator \( \mathcal{I} - \tilde{G}\mathcal{F} : D \to D \) is a contraction, i.e., if the Lipschitz constant \( \| \mathcal{I} - \tilde{G}\mathcal{F} \|_{D\subset E} \) is less than one. This implies that \( \mathcal{F} \) should be injective. Similarly, (2.18) converges if \( \mathcal{I} - \mathcal{F}\tilde{G} : \hat{D} \to \hat{D} \) is a contraction, which implies that \( \mathcal{F} \) should be surjective. When \( \mathcal{F} \) is not surjective, (2.22) shows that (2.18) and (2.19) do not always allow for a fixed point. However, the processes (2.17) and (2.20) may allow for one, provided \( \tilde{G} \) is not injective. Further we see that, for constant, non-singular affine \( \tilde{G} = \tilde{G}_h \), all processes (2.17)–(2.20) yield identical sequences of approximants. If a regular \( \tilde{G} \) is not affine, the processes (2.17) and (2.18) give different results.

For our optimization problems, where the design may be not reachable, \( \mathbf{y} \in \hat{D} \), but \( \mathbf{y} \notin \mathcal{F}(D) \), i.e., \( \mathcal{F} \) is no surjection so that no solution for (2.16) exists and (2.18)-(2.19) cannot converge. Therefore, we drop the idea of finding an \( \mathbf{x} \in D \) satisfying (2.16) and we replace the aim by looking for a solution \( \mathbf{x}^* \in D \) so that the distance between \( \mathcal{F}\mathbf{x} \) and \( \mathbf{y} \) is minimal, i.e., we want to find

\[ \mathbf{x}^* = \arg\min_{\mathbf{x} \in D} \| \mathcal{F}\mathbf{x} - \mathbf{y} \|_{\hat{E}}. \]

For a compact non-empty \( D \) and a continuous \( \mathcal{F} \), at least a solution exists. If the operators \( \tilde{G}_h \) are such that (2.17) or (2.20) converges, the stationary point \( \mathbf{x} \) satisfies (2.21) or (2.24), respectively. For a linear operator \( \tilde{G} \) this means that \( \mathcal{F}\mathbf{x} - \mathbf{y} \in \text{Ker}(\tilde{G}) \).

### 2.5.2 Defect correction and space mapping

Because the theory of defect correction is formulated in terms of Banach spaces, to compare it with the space-mapping paradigm we restrict ourselves to cost functions \( \| \cdot \| \) that take the form of a norm. In fact, for our space-mapping algorithms we simply take \( E = \mathbb{R}^n \), \( \hat{E} = \mathbb{R}^m \) and \( D = X \subset \mathbb{R}^n \) a compact subset. It is tempting to identify the operator \( \mathcal{F} \) with the fine model \( \mathbf{f} \), i.e.,

\[ \mathcal{F}\mathbf{x} = \mathbf{f}(\mathbf{x}), \quad \text{with} \; \mathbf{x} \in D = X \subset \mathbb{R}^n, \]  

(2.25)

but, to be more precise, we denote by \( \mathcal{F}^\dagger \) the operator for the minimization problem

\[ \mathcal{F}^\dagger \mathbf{y} = \arg\min_{\mathbf{x} \in X} \| \mathbf{f}(\mathbf{x}) - \mathbf{y} \|. \]  

(2.26)

Now the operator \( \mathcal{F} : D \to \hat{D} \) in (2.25) is the right-inverse of \( \mathcal{F}^\dagger \). We stress that \( \mathcal{F} \) generally is not surjective. Defect correction for operator equations is extended to optimization problems by approximating the operator \( \mathcal{F}^\dagger \).

The space-mapping idea comes back by identifying the role of the approximate pseudo-inverse. Since the mapped coarse model \( \mathbf{c} \circ \mathbf{p}_h \) acts as a surrogate for the
fine model, one clear choice of the approximate operator can be
\[ F_k x = c(p_k(x)) \quad \forall k. \] (2.27)

Then \( \tilde{G}_k \), the approximate inverse of \( F \), is the mapped coarse model optimization
\[ \tilde{G}_{k+1} y = \arg\min_{x \in X} \| c(p_k(x)) - y \|. \] (2.28)

At this stage we make no particular choice for the functions \( p_k \), but we only assume that (2.28) is easily evaluated. For example, if the spaces \( X \) and \( Z \) can be identified, we simply may take \( p_k = I \), the identity operator. We can try and find better choices for \( p_k \), but—as we assumed that both \( X \) and \( Z \) are in \( \mathbb{R}^n \)—only bijections for the functions \( p_k \) are considered. As for \( \tilde{G}_k \), we also assume for \( p_k \) that they do not to change anymore for \( k \) large enough, i.e., \( p_k = \overline{p} \) for \( k \geq k_0 \).

### 2.5.3 Defect correction in optimization

**Construction of new iterative optimization algorithms**

Using the basic defect correction processes and the relations (2.25), (2.27) and (2.28) that are associated with space mapping, we derive two new defect correction iteration schemes that can be used for optimization. Substitution of the relations into the processes (2.17) and (2.20), respectively, yields the following initial estimate and iteration processes for \( k = 0, 1, 2, \ldots \)

\[ x_0 = \arg\min_{x \in X} \| c(p_0(x)) - y \|, \] (2.29)

\[ x_{k+1} = x_k - \arg\min_{x \in X} \| c(p_k(x)) - f(x_k) \|
+ \arg\min_{x \in X} \| c(p_k(x)) - y \|, \] (2.30)

\[ x_{k+1} = \arg\min_{x \in X} \| c(p_k(x)) - c(p_k(x_k)) + f(x_k) - y \|. \] (2.31)

We denote (2.30) and (2.31) as DeC-X and DeC-Y, respectively. Again, the two processes (2.30) and (2.31) are dual in the sense that extrapolation is applied in the space \( X \) for process (2.30) and extrapolation in \( Y \) for process (2.31). In the case when the spaces \( X \) and \( Z \) can be identified, for the initial estimate (2.29) we take \( p_k = I \), the identity, like in the space-mapping algorithms.

In the above iterations all iterated minimizations are over functions involving the surrogate model, \( c \circ p_k \). However, it is the coarse model that was assumed to be cheaply optimized. Therefore, it is convenient to derive procedures, similar to the ones given, such that direct optimization over the coarse model becomes central. By taking \( F z = f(q(z)) \), \( F_k z = c(z) \) and \( \tilde{G}_k y = \arg\min_{z \in X} \| c(z) - y \| \), with \( q \) and \( q_k \) bijections from \( Z \) to \( X \) fulfilling in every iteration \( q z_k = q_k z_k \),
we obtain for \( k = 0, 1, 2, \ldots \)
\[
\begin{align*}
z_0 &= z^* = \arg\min_{z \in Z} \| c(z) - y \|, \\
z_{k+1} &= z_k - \arg\min_{z \in Z} \| c(z) - f(q_k(z_k)) + z^* \|, \\
z_{k+1} &= \arg\min_{z \in Z} \| c(z) - c(z_k) + f(q_k(z_k)) - y \|.
\end{align*}
\] (2.32)  (2.33)  (2.34)

We denote (2.33) and (2.34) as DeC-Z and DeC-Y, respectively (the reason for the latter being identified with DeC-Y will be given in Lemma 8). Because the solution is wanted in terms of fine model control variables, the procedures are complemented with \( x_k = q_k(z_k) \). The bijections can be interpreted as \( q_k = p_{k}^{-1} \). Again, for \( k > _k \), we assume the iteration process to be stationary: \( q_k = q \).

**Lemma 8.** (i) The initial estimates (2.29) and (2.32) are equivalent, as well as the processes (2.31) and (2.34). (ii) If \( p_k \) is linear and the same for all \( k \), \( p_k = p \), then also (2.30) and (2.33) are equivalent.

**Proof.** (i) We substitute in (2.29) \( p_0(x_0) = z_0 \) to obtain (2.32). In (2.31) we substitute \( p_k(x_k) = z_k \), \( p_{k+1}(x_{k+1}) = z_{k+1} \) and \( q(z_k) = x_k \) to obtain (2.34). (ii) The same substitutions in (2.30) make
\[
q_{k+1} z_{k+1} = q_k z_k = q_k \arg\min_{x \in X} \| c(x) - f(q_k(z_k)) \| + q_k \arg\min_{x \in X} \| c(x) - y \|.
\]

Now linearity of \( q_k \) and \( p_k q_{k+1} z_k = z_k \) complete the proof. \( \square \)

**Recovering space mapping from defect correction**

With a proper choice of \( p_k \) we can recover the space-mapping approaches presented in Section 2.2.2. The primal space-mapping generic scheme is obtained just by applying defect correction to the problem
\[
\mathcal{F} z^* = \arg\min_{z \in X} \| p(z) - z^* \|,
\]
where the easy to calculate pseudo-inverses are
\[
\tilde{G}_k z^* = \arg\min_{x \in X} \| p_k(x) - z^* \|,
\]
and \( p_k \) is an arbitrary approximation of \( p \) satisfying \( p_k(x) = p(x_k) \).

The dual space-mapping approach can be derived as follows. Let \( p_k \) be an approximation of \( p \) satisfying \( p_k(x) = p(x_k) \), then (2.30) reduces to
\[
x_{k+1} = \arg\min_{x \in X} \| c(p_k(x)) - y \|,
\] (2.35)

which coincides with the dual space-mapping algorithm.
A special case. If $X = Z$ and $p_k = I$, then (2.29) and (2.30) reduce to
\[
\begin{align*}
x_0 &= \text{argmin}_{x \in Z} \| c(x) - y \| = z^* , \\
x_{k+1} &= x_k - \text{argmin}_{x \in Z} \| c(x) - f(x_k) \| + z^* , \\
&= x_k - p(x_k) + z^* , \\
&\quad k = 0, 1, \ldots .
\end{align*}
\] (2.36)
This is the iterative method to solve the original space-mapping problem (2.4) which has been published in [20, 67]. Clearly, the iteration can only converge if $p(\mathbf{x}_{\text{opt}}) = z^*$ allows for a solution and the convergence rate is determined by the Lipschitz constant $\| I - p(\cdot) \|$. Hence, a necessary and sufficient condition for convergence of (2.36) is $\| I - p(\cdot) \| < 1$.

Properties of the new algorithms

Convergence of DeC-X. In the case of convergence of (2.30) we obtain, with fixed point $\lim_{k \to \infty} x_k = x$, the equality
\[
\hat{x} := \text{argmin}_{x \in X} \| c(x) - f(x) \| = \text{argmin}_{x \in X} \| c(p(x)) - y \| ,
\] (2.37)
which shows that the design $x \in X$ is associated with $\hat{y} = c(p(\hat{x})) \in c(p(X))$ that is the best approximation of $y$ reachable in $c(p(X)) = c(Z)$, and at the same time such that $f(\hat{x})$ is best approximated in $c(Z)$ by the same $\hat{y}$. Hence, both $f(\hat{x})$ and $y$ belong to the class $W_\Psi \subset \mathbb{R}^m$ of points in $\mathbb{R}^m$ for which $\hat{y}$ is the closest point in $c(Z) \subset \mathbb{R}^m$. Noting that $p$ is a bijection, the equality (2.37) can be written briefly as
\[
\hat{x} := p^{-1}(p(x)) = p^{-1}(z^*) ,
\] (2.38)
where $p$ denotes the space-mapping function (2.1).

If we assume $\hat{E}$ to be a Hilbert space, from (2.37) we see that both $c(p(\hat{x})) - f(\hat{x})$ and $c(p(\hat{x})) - y$ are perpendicular to the tangent plane for $c(p(X))$ at $c(p(\hat{x}))$. We denote the linear manifold perpendicular to the tangent plane for $c(p(X))$ at $c(p(\hat{x}))$ by $c(p(X))^\perp(\hat{x})$. Thus
\[
c(p(\hat{x})) - f(\hat{x}) \in c(p(X))^\perp(\hat{x}) ,
\]
c\hspace{2em}(2.39)
\[
c(p(\hat{x})) - y \in c(p(X))^\perp(\hat{x}) ,
\]
and hence
\[
f(\hat{x}) - y \in c(p(X))^\perp(\hat{x}) = c(p(X))^\perp(p^{-1}p(x)) = c(Z)^\perp(z^*) .
\]
This situation is shown in Figure 2.9. We summarize this result in the following lemma.

Lemma 9. Let $\hat{E}$ be a Hilbert space. In case of convergence of (2.30) with a fixed point $x$ we obtain
\[
f(\hat{x}) - y \in c(Z)^\perp(z^*) = c(p(X))^\perp(p^{-1}p(x)) .
\] □
Because $\tilde{p}$ is a bijection, from (2.38) we directly see the following result.

**Lemma 10.** Let $p(x)$ be the space-mapping function as defined in (2.1). If (2.30) converges to $x$ then $p(x) = x^\star$.

**Corollary.** It follows from the definition of a perfect mapping that, with $x$ the convergence result of (2.30), $p(x) = p(x^\star)$ if and only if $p$ is a perfect mapping.

**Remark.** We note that convergence of (2.30) implies $z^\star \in p(X)$. Hence, in the case $z^\star \notin p(X)$ the algorithm cannot reach a stationary point.

In case we have knowledge about the flexibility of the models we can go a bit further with the analysis and state the next lemma.

**Lemma 11.** If the coarse model is more flexible than the fine one, i.e., $c(Z) \supset f(X)$, and $y \in c(Z) \setminus f(X)$, then (2.30) cannot converge.

**Proof.** Because $y \in c(p(X))$ there exists a $x$ such that $y = c(p(x))$. If (2.30) converges, then (2.37) holds and $x = \arg\min_{x \in X} \| c(p(x)) - f(x) \|$. Since the surrogate model is more flexible than the fine model $c(p(x)) = f(x)$.

This yields a contradiction. Hence (2.30) cannot converge.

**Remark.** In the end we have what one would have expected. If $c(Z) = f(X)$ and the process (2.30) converges, the space mapping is perfect and the fixed point of the iteration is the solution of our optimization problem.

**Convergence of Dec-Y.** In Section 2.5.3 Lemma 8, it was shown that the iterations (2.31) and (2.34) are equivalent. Hence, their convergence results coincide. For (2.31) we derive

**Lemma 12.** Let the space $\tilde{E}$ be a Hilbert space. In the case of convergence of (2.31), with fixed point $\lim_{k \to \infty} x_k = x$ we obtain

$$f(x) - y \in c(p(X)) \perp(x).$$
Proof. In the case of convergence of (2.31), with fixed point \( \lim_{k \to \infty} x_k = \bar{x} \) we obtain
\[
\bar{x} := \arg\min_{x \in X} \|c(p(x)) - c(p(\bar{x})) + f(\bar{x}) - y\|.
\] (2.39)
So, it follows that for all \( h \in \mathbb{R}^n \) with \( \|h\| \geq 0 \)
\[
\|c(p(\bar{x} + h)) - c(p(\bar{x})) + f(\bar{x}) - y\| \geq \|f(\bar{x}) - y\|.
\]
With \( \hat{E} \) a Hilbert space, this implies that
\[
f(\bar{x}) - y \perp c(p(\bar{x} + h)) - c(p(\bar{x})) \quad \forall h \in \mathbb{R}^n,
\]
so that \( f(\bar{x}) - y \) is perpendicular to the linear manifold through \( c(p(\bar{x})) \) that is tangential to \( c(p(X)) \). \( \square \)

Remark. Notice the slight difference with the result for DeC-X.

Lemma 13. With a flexible coarse model and \( p_k(x_k) = p(x_k) \) the iteration processes (2.30) and (2.31) yield identical sequences of approximations. Hence, under these conditions, Lemmas 10 and 11 for (2.30) also hold for (2.31).

Proof. The main iteration in DeC-Y is
\[
x_{k+1} = \arg\min_{x \in X} \|c(p_k(x_k)) - c(p_k(x_k)) + f(x_k) - y\|.
\]
From Lemma 3 for a more flexible coarse model
\[
c(p(x_k)) = f(x_k).
\]
Since \( p_k(x_k) = p(x_k) \) we have
\[
x_{k+1} = \arg\min_{x \in X} \|c(p(x)) - y\|
\]
which coincides with DeC-X (2.30) because we satisfy the hypothesis of recovering space mapping from defect correction. \( \square \)

Convergence of DeC-Z. Here the results concerning the analysis of (2.33) are given. Because of the similarity of the lemmas and the proofs for the scheme (2.33) and those for (2.30), we omit the proofs.

Lemma 14. Let \( \hat{E} \) be a Hilbert space. In case of convergence of (2.33) with a fixed point \( \bar{x} = q(\bar{z}) \) we obtain
\[
f(\bar{x}) - y \in c(Z)^\perp(z^*) \equiv c(p(X))^{\perp}(q,p(\bar{x})). \ \square
\]

Lemma 15. Let \( p(x) \) be the space-mapping function as defined in (2.1). If (2.33) converges to \( \bar{x} = \bar{q}(\bar{z}) \) then \( p(\bar{x}) = \bar{z}^* \). \( \square \)
Again it follows that, with \( \bar{x} \) the stationary point of (2.33), \( \mathbf{p}(\bar{x}) = \mathbf{p}(\mathbf{x}^*) \) if and only if \( \mathbf{p} \) is a perfect mapping. If \( \mathbf{c}(Z) = \mathbf{f}(X) \) and the process converges, the fixed point of the iteration is the solution of our optimization problem.

**Lemma 16.** If the coarse model is more flexible than the fine one, i.e., \( \mathbf{c}(Z) \supset \mathbf{f}(X) \), and \( \mathbf{y} \in \mathbf{c}(Z) \setminus \mathbf{f}(X) \), then (2.33) cannot converge. \( \square \)

### The processes with linear coarse model and mappings

A coarse model that is linear in its control variables, in combination with linear mappings for either \( \mathbf{p}_k \) and \( \bar{\mathbf{p}} \) or \( \mathbf{q}_k \) and \( \bar{\mathbf{q}} \), can be attractive in practice and the respective analysis can give some additional insight. Under these linearity conditions, all defect correction processes above coincide. For the analysis we take the iteration (2.33) with the Euclidean norm as a starting point.

Denoting by \( C \) the \( m \times n \) matrix that describes the linear coarse model, we obtain

\[
\begin{align*}
\mathbf{z}_0 & = \mathbf{z}^* = \arg\min_{z \in \mathbf{Z}} \| \mathbf{C}z - \mathbf{y} \|_2, \\
\mathbf{z}_{k+1} & = \mathbf{z}_k - \arg\min_{z \in \mathbf{Z}} \| \mathbf{C}z - \mathbf{f}(\mathbf{q}_k(\mathbf{z}_k)) \|_2 + \mathbf{z}^*,
\end{align*}
\]  

or

\[
\begin{align*}
\mathbf{z}_0 & \in C^\dagger \mathbf{y} + \text{Span}(\mathbf{V}_0), \\
\mathbf{z}_{k+1} & \in \mathbf{z}_k - C^\dagger (\mathbf{f}(\mathbf{q}_k(\mathbf{z}_k)) - \mathbf{y}) + \text{Span}(\mathbf{V}_0),
\end{align*}
\]  

where \( C^\dagger \) is the pseudo-inverse of \( C \) and \( \text{Span}(\mathbf{V}_0) \subset \mathbb{R}^n \) is the kernel of \( C^TC \). The fixed point \( \mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{q}(\mathbf{x})) \) of iteration (2.41) is characterized by

\[ C^\dagger (\mathbf{y} - \mathbf{f}(\mathbf{x})) \in \text{Span}(\mathbf{V}_0). \]

In the case \( m \geq n \) and with a full rank \( C \) operator, the term \( \text{Span}(\mathbf{V}_0) \) vanishes, so that \( C^TC(\mathbf{y} - \mathbf{f}(\mathbf{x})) = 0 \), i.e., the discrepancy between the data and the optimal response is in the kernel of \( C^T \).

**Convergence of the linearized process.** From (2.41) it is clear that for a non-full rank \( C \) matrix or if \( m < n \), the solution is not uniquely determined and the iteration cannot be a contraction, whereas for a full-rank \( C \) with \( m \geq n \), the convergence rate is determined by the Lipschitz constant

\[ \| I - C^\dagger \mathbf{f}(\mathbf{q}(\cdot)) \|. \]

If we consider the Euclidean norm, and the operators \( \mathbf{f} : X \to Y \) and \( \mathbf{q} : Z \to X \) have Jacobians \( F \in \mathbb{R}^{m \times n} \) and \( \bar{Q} \in \mathbb{R}^{n \times n} \) respectively, then the convergence rate is bounded by

\[ \| I - C^\dagger \bar{Q} \|_2 = \| (C^TC)^{-1} C^T (C - F \bar{Q}) \|_2 \leq \| (C^TC)^{-1} \|_2 \| C^T (C - F \bar{Q}) \|_2. \]

This shows how the operators \( \mathbf{c} \) and \( \mathbf{f} \circ \mathbf{q} \) should be related for the iteration process to converge: the coarse model \( \mathbf{c} \) should be full rank and sufficiently stable.
(well-conditioned) and the fine model $f$, preconditioned by $\Omega$, should sufficiently correspond with the coarse model $c$ (at least with respect to the components in its range).

**Left- and right-preconditioning**

The efficiency of the space-mapping algorithms depends essentially on the similarity between the fine and the coarse model used and their relative computational costs for solution. The space-mapping procedure is then determined by the choice how the space-mapping function is approximated, with ASM and TRASM as obvious choices.

In the defect correction algorithms we also have to deal with the similarity between the fine and the coarse model but, instead of in the selection of the space-mapping approximation, there the freedom is in the preconditioning mapping $p$ or $\Omega = p^{-1}$. This choice can be used to improve the convergence of the procedure. From the linear case we see that better convergence can be expected if the approximation $c(p(x)) \approx f(x)$ is more accurate. We can consider $p$ as a right-preconditioner \cite{88} of $c$ to better approximate $f$.

On the other hand, by the orthogonality relations in the lemmas above, we see that it is advantageous if the manifolds $f(X)$ and $c(Z)$ are found parallel in the neighborhood of the solution. However, by space mapping or by right-preconditioning the relation between the manifolds $f(X)$ and $c(Z)$ remains unchanged. This relation can be improved by the effect of an additional left-preconditioner. Therefore, we have to introduce such a preconditioner $r$ so that near $f(x^*) \in Y$ the manifold $c(Z) \subset Y$ is mapped onto $f(X) \subset Y$:

$$r(c(p(x))) \approx f(x).$$

Hence, in the next section we propose such an operator $S$, which maps $c(Z)$ onto $f(X)$, at least in the neighborhood of the solution. This improves significantly the traditional approach. The recently introduced output space mapping and space-mapping based interpolating surrogate scheme \cite{15} also take such left-preconditioning into account, but in a different manner.

### 2.5.4 Improving space mapping

Except for the hybrid algorithm HASM, all space-mapping methods thus far have the clear disadvantage that, in general, the fixed point of the iteration does not necessarily coincide with the solution of the fine model minimization problem. This is due to the fact that the approximate solution $\bar{x}$ satisfies

$$f(\bar{x}) - y \in c(p(X)) \perp (p^{-1} p(\bar{x})) \text{ for DeC-X and DeC-Z},$$

$$f(\bar{x}) - y \in c(p(X)) \perp (\bar{x}) \text{ for DeC-Y},$$

$$f(\bar{x}) - y \in c(p(X)) \perp (\bar{x}) \text{ for DeC-Y},$$
whereas $x^*$, a (local) minimum for $\|f(x) - y\|$, satisfies
$$f(x^*) - y \in f(X)^+ (x^*) .$$
Hence, differences between $\bar{x}$ and $x^*$ will be larger for larger distances between $y$ and the sets $f(X)$ and $c(Z)$, and for larger angles between the linear manifolds tangential at $c(Z)$ and $f(X)$ near the optima. This disadvantage, however, can be removed by introducing a mapping $S : Y \rightarrow Y$ such that, for a proper $\bar{x} \in Z$, $S \ c(\bar{x}) = f(x^*)$ and the tangent plane for $c(Z)$ at $c(\bar{x})$ is mapped onto the tangent plane for $f(X)$ at $f(x^*)$. Because, in the non-degenerate case when $m \geq n$, both $f(X)$ and $c(Z)$ are $n$-dimensional manifolds in $\mathbb{R}^m$, $S$ can be described by an affine mapping
$$S \bullet = f(x^*) + S (\bullet - c(\bar{x})) ,$$
where $S$ is an $m \times m$-matrix. This approach was introduced in [35] and is the basis of the manifold-mapping technique. We will see in the next chapter that the mapping $S$ depends on the solution of the problem $x^*$ and, thus, that it is not a priori available. Manifold-mapping algorithms iteratively approximate $S$ by $S_k : Y \rightarrow Y$ given by
$$S_k \bullet = f(x_k) + S_k (\bullet - c(\bar{p}(x_k))) , \quad (2.42)$$
with $\bar{p} : X \rightarrow Z$ being a (known) arbitrary mapping and $S_k$ a proper $m \times m$ matrix. We can write the next manifold-mapping iterant as
$$x_{k+1} = \arg\min_{x \in X} \| S_k (c(\bar{p}(x))) - y \| , \quad (2.43)$$
and then, because $S_k (c(\bar{p}(x_k))) = f(x_k)$, we can also interpret it as a defect correction iteration with either $\bar{F}_k = S_k \circ c \circ \bar{p}_k$ and $\bar{F} = f$ in (2.30) or (2.31), or with $\bar{F}_k = S_k \circ c$ and $\bar{F} = f \circ \bar{p}_k$ in (2.33) or (2.34).

In Chapter 4 we will see that, under mild assumptions, if a manifold-mapping based algorithm converges to $\bar{x}$, this fixed point is the fine model optimum $x^*$. Convergence of these manifold-mapping based schemes will be studied further in that chapter.

### 2.6 Simple examples and counterexamples

**Coinciding space-mapping optima and deviation from perfect mapping**

Each of the four points $z^*$, $x^*_p$, $x^*_q$ and $x^*$ introduced in the previous sections is the solution of a minimization problem, the first three being approximations to the last. In this section by a few very simple examples we illustrate that in some cases these approximations coincide, but in other cases they can be all different.
As a first example we consider a simple least squares data fitting problem: the data are \( y = [y_{-1}, y_0, y_1] \) for \( t = [-1, 0, 1] \), so we have \( m = 3 \). The fine model is the family of polynomials over \( X = \mathbb{R}^2 \),

\[
f(x) = f(x_1, x_2) = [x_1^2 t_1^2 + x_1 t_1 + x_2]_{i=1,2,3}.
\]

(2.44)

The coarse model is the linear \( c(x) = c(z_1, z_2) = z_1 t_1 + z_2 \). The number of control parameters is \( n = 2 \) and the two models are similar for small \( x_1 \). The search is made over \( X = Z = \mathbb{R}^2 \). It is immediate that the coarse model minimum is

\[
z^* = \left[ \frac{y_1 - y_{-1}}{2}, \frac{2y_1 + y_0 + y_1}{3} \right].
\]

(2.45)

Using the definition of the space-mapping function, it is easily determined as

\[
p(x) = \left[ x_1, x_2 + \frac{2}{3} x_1^2 \right],
\]

which for small \( x_1 \) clearly resembles the identity operator. It is a bijection between \( X \) and \( Z \) and by applying Lemma 2 in Section 2.3 we see that for this problem all the space-mapping solutions exist and coincide

\[
x^*_m = x^*_p = x^*_d = \left[ \frac{y_1 - y_{-1}}{2}, \frac{2y_1 + y_0 + y_1 - (y_1 - y_{-1})^2/2}{3} \right].
\]

(2.46)

Notice that this is not the true solution for fitting with (2.44). In the case of perfect mapping the solution \( x^*_m \) corresponds with the true optimum \( x^* \).

**Deviation from perfect mapping.** In order to check the value of the space-mapping approach and see the effect of a non-perfect mapping, we perform the following experiment. We create the data \( y = f(x) + n \), with \( x = [0, 1, 0, 1] \) and \( n \) a random vector, uniformly distributed on a sphere with radius \( \| n \| = r \). Figure 2.10 shows the quotient \( \| f(x^*) - y \|/\| f(x^*_{m}) - y \| \). When \( n \) is not significant compared to the fine model response, then the mapping is almost perfect and the space-mapping solution yields a response with a cost function close to the true optimum. With this test we corroborate that a coarse model that can accurately reproduce the response of the fine one is a good candidate for being used within space mapping, and that fine models that meet the specifications well improve the quality of the space-mapping result.

**Different optima \( x^*, x^*_p \) and \( x^*_d \)**

However, in general, if \( z^* \notin p(X) \) the solutions \( x^*_p \) and \( x^*_d \) are expected to be different. By increasing the misalignment between the models, the condition
Figure 2.10: Quality of the space-mapping optimum depending on the deviation from perfect mapping.

Results for 8000 experiments with random perturbations of size \( \|\mathbf{n}\|_2 < 10. \)

In the example \( \|\mathbf{f}(\mathbf{x})\|_2 = 0.2328 \)

\( \mathbf{z}^* \in \mathbf{p}(\mathbf{X}) \) may no longer be satisfied. This is shown by taking another simple fine model:

\[
\mathbf{f}(\mathbf{x}) = \mathbf{f}(x_1, x_2) = [x_1 (x_2 t_i + 1)^2]_{i=1,2,3}.
\]  

We keep the linear coarse model \( \mathbf{c}(\mathbf{z}) = \mathbf{c}(z_1, z_2) = z_1 t + z_2. \) Proceeding as above, the space-mapping function can be determined as

\[
\mathbf{p}(\mathbf{x}) = \left[ 2 x_1, x_2, (1 + \frac{2}{3} x_2^3) \right].
\]

This mapping is not injective and is substantially different from the identity. Furthermore, \( \mathbf{p}(\mathbf{X}) \neq \mathbb{R}^2 \) and it is easily verified that \( \mathbf{p}(\mathbf{X}) = \{ \mathbf{z} \in \mathbb{Z}; 3 z_2^2 \geq 2 z_1^2 \}. \)

For this example we can distinguish four different situations: (1) the design is reachable, (2) the mapping is perfect but the design is not reachable, (3) the primal and the dual solution correspond but the mapping is not perfect, and (4) the four points \( \mathbf{z}^*, \mathbf{p}(\mathbf{x}_p^*), \mathbf{p}(\mathbf{x}_d^*) \) and \( \mathbf{p}(\mathbf{x}^*) \) are all different.

These four cases are shown in Figure 2.11 and in Table 2.1. Case (1) is characterized by the fact that the cost function for the fine model vanishes at \( \mathbf{x}_p^* = \mathbf{x}_d^* = \mathbf{x}^*. \) This occurs if the data \( \mathbf{y} \) fit the model (2.47) so that a reachable design is obtained. In case (2) we also see \( \mathbf{x}_p^* = \mathbf{x}_d^* = \mathbf{x}^*. \) but the cost function for the fine model does not vanish. In case (3) still \( \mathbf{z}^* \in \mathbf{p}(\mathbf{X}) \), but the data are such that \( \mathbf{p} \) is no perfect mapping, so that \( \mathbf{x}_p^* = \mathbf{x}_d^* \) but they do not coincide with \( \mathbf{x}^*. \)

For the case (4) the data \( \mathbf{y} \) are selected such that \( \mathbf{z}^* \not\in \mathbf{p}(\mathbf{X}) \). Now \( \mathbf{x}_p^* \neq \mathbf{x}_d^* \) and, in fact, all values differ.

Because of the quadratic form in \( \mathbf{p}(\mathbf{x}) \), each of the minimization procedures for the primal and the dual approach yields two solutions. In Table 2.1 we omit
Four situations: (1) and (2): coinciding $z^* = p(x^*_p) = p(x^*_d) = p(x^*)$;
(3): coinciding $z^* = p(x^*_p) = p(x^*_d)$, separate $p(x^*)$;
(4): separate $z^*$, $p(x^*_p)$, $p(x^*_d)$ and $p(x^*)$.

The top and bottom quadrant are $p(X)$. In case (4) $z^* \not\in p(X)$.

Figure 2.11: Different space-mapping solutions for problem (2.47).

The one that has the larger fine cost function.

The optimization problem that defines the primal approach finds the point in $p(X)$ that is closest in the Euclidean norm to $z^*$. It lies on a circle centered in $z^*$, with minimum radius so that it has one point of contact with $p(X)$. For the dual approach the contour lines for the cost function are ellipses in $p(X)$ centered in $z^*$. Therefore, $p(x^*_p)$ and $p(x^*_d)$, and consequently $x^*_p$ and $x^*_d$, do not coincide.

Remark. Case (4) has two local optima. The behavior of the fine cost function in the neighborhood of one of them is similar to that of the coarse cost function close to the coarse optimum. But for the other one, the global fine optimum, the difference is substantial. Space mapping finds solutions close to the local optimum with the similar behavior (see Figure 2.12). This observation emphasizes the importance of the similarity between the models used.

The importance of the use of similar models

Now follows an example showing how strong dissimilarities between the models make both space-mapping approaches fail. For the fine model we take the family
Table 2.1: Different space-mapping solutions for the least squares data fitting problem associated with (2.47).

<table>
<thead>
<tr>
<th></th>
<th>$y_0$</th>
<th>$y_1$</th>
<th>$z^*$</th>
<th>$C(z^*)$</th>
<th>$F(z^*)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.08100</td>
<td>0.10000</td>
<td>0.12100</td>
<td>0.0290.101</td>
<td>8.16e-04</td>
</tr>
<tr>
<td>2</td>
<td>0.10011</td>
<td>0.10125</td>
<td>0.10241</td>
<td>0.0010.101</td>
<td>8.16e-06</td>
</tr>
<tr>
<td>3</td>
<td>0.00000</td>
<td>-0.40000</td>
<td>0.10000</td>
<td>0.050-0.100</td>
<td>3.67e-01</td>
</tr>
<tr>
<td>4</td>
<td>0.00000</td>
<td>-0.35000</td>
<td>0.20000</td>
<td>0.1000.050</td>
<td>3.67e-01</td>
</tr>
</tbody>
</table>

$$C(x) = ||c(x) - y||_2$$ and $$F(x) = ||f(x) - y||_2$$.

of exponentials

$$f(x) = f(x_1, x_2) = [x_1 \exp(x_2 t_i)]_{i=1,2,3},$$

and the coarse one is linear as before. In both cases $X = Z = \mathbb{R}^2$. The space-mapping function is again uniquely determined

$$p(x) = \left[ x_1 \sinh(x_2), \frac{x_1}{3} (1 + 2 \cosh(x_2)) \right].$$

Now $p(X) = \{ z \in Z; 9 \ z_2^2 > 4 \ z_1^2 \}$ is an open set and, thus, in the case $z^* \notin p(X)$, the primal and dual space-mapping solutions cannot be found because no minimum exists for the minimization problems that define these approximations. The fine and the coarse model have little in common, to the extent that for certain specifications the space-mapping technique cannot be applied. The conclusion is that in order to have a proper space-mapping scheme, it is crucial to use models that are sufficiently similar.

**Preference for the dual approach**

The dual space-mapping solution for case (4) in Table 2.1 above has a smaller cost function associated than the primal one. Additional experiments with the same problem were carried out in order to check if this better performance can generally be expected, but the results were inconclusive. In [82, section 4.1.2] the following lemma in favor of the dual approach is given and checked experimentally.
Figure 2.12: Level curves and minima of the fine and coarse cost functions for
the case (4) in Table 2.1. The solutions found by space-mapping, $x^*_p$ and $x^*_d$,
correspond with the local optimum for the fine model, which shows similarity
with the coarse model near $z^*$.

**Lemma 17.** If $\exists \epsilon, \delta > 0, \epsilon \leq \delta$ such that

(i) $\| c(p(x^*_p)) - f(x^*_p) \| \leq \epsilon/2$, $\| c(p(x^*_d)) - f(x^*_d) \| \leq \epsilon/2$, and

(ii) $\| c(p(x^*_p)) - y \| - \| c(p(x^*_d)) - y \| \geq \delta$

then $\| f(x^*_p) - y \| \leq \| f(x^*_d) - y \|$. \hfill $\Box$

We give two theoretical examples supporting a choice in favor of the dual
approach, specially in the case where the flexibility concept can be used.

**Example I.** In this example $X$ is a non-trivial subset of $Z$. On $X$ the fine
and coarse model responses are taken equal. Then the space-mapping function
is the identity and the mapped coarse model matches the fine one, i.e.,
$c(p(x)) = f(x) \forall x \in X$ (see Lemma 3 (i) Section 2.3). Under these circumstances
the dual solution $x^*_d$ coincides with the fine optimum $x^*$. But, in general,
is this not the case for the primal approach, as can be seen for the particular sets $X$
and $Z$ shown in Figure 2.13 (left), where $c$ is the identity, the Euclidean norm is
used, and with $y \in Y \setminus c(Z)$ we check that $x^*_p \neq x^*$. This argument, based on
projection onto subsets, is more general and can be used in less obvious situations.

**Example II.** In Figure 2.13 (right) we analyze the possibility of $x^*_p = x^* \neq x^*_d$
with $c$ (locally) more flexible than $f$ in the region near the optimal
responses. This means that we assume the existence of a set $R \subset \mathbb{R}^m$ such that
$R \supset R \cap c(Z) \supset R \cap f(X)$ and that $c(p(x^*_p)), f(x^*), c(x^*) \in R$. We define the
auxiliary set $Q = \{ x \in X | f(x) \in R \}$. By Lemma 3 (i) Section 2.3, applied lo-
Figure 2.13: Left: A simple counterexample for $x_0^* \neq x^* = x_d^*$. The Euclidean norm is taken as cost function. Right: Usually the fine and coarse models are similar enough near the optima that $c(z^*) \notin f(X)$ implies a behavior as shown.

\[
x^*_p = x^* = f(x^*) = y
\]
\[
x^*_d = x^* = f(x^*)
\]
\[
x^* = \arg\min_{x \in X} \|f(x) - y\| = \arg\min_{x \in Q} \|f(x) - y\|
\]
\[
= \arg\min_{x \in Q} \|c(p(x)) - y\| = \arg\min_{x \in X} \|c(p(x)) - y\| = x^*_d,
\]
i.e., the dual space-mapping approach also yields the fine optimum. We conclude that in this case the equality $x^*_p = x^*$ implies $x^*_d = x^*$.

The result does not hold for a more flexible fine model in $R$. Since $x^*_p \neq x^*_d$, by Lemma 2 (Section 2.3) the coarse optimum $z^*$ cannot be in $p(X)$. From $z^* \notin p(X)$ it follows that $c(z^*) \notin f(X)$, otherwise $c(z^*) = f(\bar{x})$ for some $\bar{x} \in X$, would yield $z^* = p(\bar{x})$. Because $c(z^*) \in R$ we have a clear contradiction with the fine model being more flexible than the coarse one in $R$.

2.7 Conclusions

The space-mapping technique aims at accelerating expensive optimization procedures by combining problem descriptions with different degrees of accuracy. In numerical analysis, for the solution of operator equations, the same principle is known as defect correction iteration. Taking the defect correction principle as a starting point, space mapping can be interpreted, extended and analyzed.

Introducing the concept of flexibility of a model, important properties of space mapping can be formulated in these terms, and better understood. Equally flexible models are most desirable in space mapping, but this property does not generally hold for every pair of problem formulations used in practice. As a consequence, the solution found by traditional space mapping may differ from the solution for the fine model. By introducing proper left-preconditioning we can improve the flexibility of the models. Thus, we derive manifold mapping,
a new improved space-mapping algorithm which yields the accurate solution (a precise description and analysis of manifold-mapping will be given in the next two chapters). The phenomena analyzed in this chapter are eventually illustrated by a few simple examples. Practical design problems that confirm the efficiency of the space-mapping/defect correction approach in optimization can be found in Chapters 5 and 6.